 **UNIVERSITY OF MUMBAI**



**A PROJECT REPORT ON**

**“STEGANOGRAPHY USING MATLAB”**

**SUBMITTED BY**

PURACKAL RIBA SEBASTIAN PATEL ABHILASHA PARESH

VARADKAR PRATIBHA MANGALDAS WANKHEDE RAJ BHANUDAS

**GUIDED BY**

PROF. AMIT KUKREJA

**DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION ENGINEERING**

**K. J. SOMAIYA INSTITUTE OF ENGINEERING AND INFORMATION TECHNOLOGY**

**SION (E), MUMBAI 400022**

**YEAR 2014-2015**

**K. J. SOMAIYA INSTITUTE OF ENGINEERING AND INFORMATION TECHNOLOGY**



**SION (E), MUMBAI 400022**

**UNIVERSITY OF MUMBAI**

**CERTIFICATE**

This is to certify that the project titled “STEGANOGRAPHY USING MATLAB” has been successfully completed under our supervision and guidance by the following students:

PURACKAL RIBA SEBASTIAN

PATEL ABHILASHA PARESH

VARADKAR PRATIBHA MANGALDAS

WANKHEDE RAJ BHANUDAS

In the partial fulfillment of semester VII of Bachelor of Engineering in Electronics and Telecommunication branch as prescribed by the University of Mumbai during the academic year 2014-2015. The said work has been assessed and is found to be satisfactory.

Prof. Amit Kukreja Prof. Mrs. Namrata Gharat

(Internal Guide) (Head of Department)

Dr. Dilip R.Pangavhane College Seal

(Principal) (External Examiner)

**K. J. SOMAIYA INSTITUTE OF ENGINEERING AND INFORMATION TECHNOLOGY**

**SION (E), MUMBAI 400022**

**Department Electronics and Telecommunication**

**Vision**

To become one of the leaders in the field of Electronics and Telecommunication education by promoting the aptitude of inquiry and study among the budding engineers for customizing and popularizing technology for the masses.

**Mission**

To provide conductive teaching learning environment for the stakeholders thereby creating suitable workforce for the service sectors.

**ACKNOWLEDGEMENT**

“...........the beauty of the destination is half veiled and the fragrance of the success half dull until the traces of all those enlightening the path are left to fly with the wind spreading word of thankfulness………”

Keeping this in view; it would be unfair on our part if we don’t thank the mentioned few. We express our sincere thanks to our mentor and guide Prof. Amit Kukreja. His dedication and keen interest and above all his overwhelming attitude helped us deliver our best.

We also owe a great thank you to Prof. Mrs. Jayashree Khanapuri for her valuable suggestions with kindness and enthusiasm which helped us a lot throughout the project.

We also thank Prof. Mrs. Namrata Gharat (Head of Department EXTC) and Dr. Dilip R. Pangavhane (Principal) for their constant encouragement throughout our project.

**TABLE OF CONTENTS**

1. Certificate ………………………………………………………………II
2. Vision and Mission …………………………………………………….III
3. Acknowledgement ……………………………………………………..IV
4. List of figures…………………………………………………………..VI
5. List of tables…………………………………………………………...VII
6. Abstract……………………………………………………………….VIII
7. Introduction …………………………………………………………….1
   1. About steganography …………………………………………...1
   2. Applications of steganography …………………………………2
8. Literature survey ………………………………………………………..5
   1. Least significant bit method …………………………………….5
   2. LSB using secret key ……………………………………………5
   3. Randomized algorithm ………………………………………….5
   4. Dynamic steganography…………………………………………6
   5. Edge detection method ………………………………………….6
   6. Filtering algorithm ………………………………………………7
9. Analysis and design steps completed till end of semester VII …………10
   1. Analysis and design steps ………………………………………10
   2. Embedding process ……………………………………………..10
      1. Embedding message in lighter area …………………….11
      2. Embedding message in darker area …………………….11
   3. Extracting process ………………………………………………13
      1. Extracting message from lighter pixel ………………….13
      2. Extracting message from darker pixel ………………….13
10. Implementation …………………………………………………………16
11. Conclusion ………………………………………………………………18
12. References ………………………………………………………………19

**LIST OF FIGURES**

1. Basic steganography algorithm (1.1) ……………………………………………..2
2. Lena (2.1) …………………………………………………………………………8
3. Baboon (2.2) ………………………………………………………………………8
4. Classification of lighter and darker picture (3.2.1) ………………………………11
5. Embedding flowchart (3.2.2) …………………………………………………….12
6. Extracting flowchart (3.3.1) ……………………………………………………...14

**LIST OF TABLES**

1. Embedding Locations based on the data embedded in the previous pixel (2.4.1) ……….6
2. Comparison of various Algorithms studied on the basis PSNR and MSE ratios (2.1) …..9
3. Embedding a message in lighter pixel (3.2.1) ……………………………………………12
4. Embedding a message in darker pixel (3.2.2) ……………………………………………13
5. Extracting message from lighter pixel (3.3.1) ……………………………………………15
6. Extracting message from darker pixel (3.3.2) …………………………………………....15

**ABSTRACT**

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information. Many different carrier file formats can be used, but digital images are the most popular because of their frequency on the Internet. For hiding secret information in images, there exists a large variety of steganographic techniques, some are more complex than others and all of them have their strong and weak points. Different applications have different requirements of the steganography technique used. For example, some applications may require absolute invisibility of the secret information, while others require a larger secret message to be hidden. This report represents a technique for secret communication using steganography. We plan to design a system that will allow an average user to securely transfer text messages by hiding them in a digital image file using the local characteristics within an image. In this project we are using cryptography along with steganography to add another level of security. This technique has an advantage over other information security techniques because the text is hidden in an image, which is not an obvious text information carrier. The project contains several challenges that make it interesting to develop. The main advantage of this project is a simple, powerful and user-friendly GUI that plays a very large role in the success of the application.

**CHAPTER 1**

**INTRODUCTION**

**CHAPTER 1**

**INTRODUCTION**

Since the rise of the Internet one of the most important factors of information technology and communication has been the security of information to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique used to implement this, is called steganography.

**1.1 Introduction**

Steganography is the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. The word steganography is derived from the Greek words “stegos” meaning “cover” and “grafia” meaning “writing” defining it as “covered writing”. Though the concept of steganography and cryptography are the same, but still steganography differs from cryptography. Cryptography focuses on keeping the contents of a message secret while steganography focuses on keeping the existence of a message secret. Generally, the hidden messages will appear to be (or be part of) something else. The advantage of steganography over cryptography alone is that the intended secret message does not attract attention to itself as an object of scrutiny. Plainly visible encrypted messages—no matter how unbreakable—will arouse interest, and may in themselves be incriminating in countries where encryption is illegal. Thus, whereas cryptography is the practice of protecting the contents of a message alone, steganography is concerned with concealing the fact that a secret message is being sent, as well as concealing the contents of the message.

Steganography includes the concealment of information within computer files. In digital steganography, electronic communications may include steganographic coding inside of a transport layer, such as a document file, image file, program or protocol. Given the large amount of redundant bits present in the digital representation of an image, images are the most popular cover objects for steganography.

The image algorithms can be categorized into two categories, namely, spatial domain and frequency domain. In spatial domain we deal with images as it is. The message to be concealed in the image is embedded directly into the pixels whereas in frequency domain we first transform the image to its frequency distribution, after processing the image the output is not an image but a transformation after performing inverse transform it is then converted into an image which is viewed in the spatial domain. For our project we are performing image steganography in the spatial domain.

**1.2 Problem description**

Steganography refers to concealing data in a covert carrier file in such a way that it is difficult for unauthorized third parties to detect and recover the hidden data. As a result the effectiveness of a steganography algorithm is determined by three properties: the capacity of the data that can be hidden without distorting the carrier file; the imperceptibility of the carrier file after hiding the secret data into it; and the irrecoverabilty of the hidden data in case they were detected.

Hiding capacity: It determines the number of bytes that can be covered within the carrier file without distorting or damaging it. For instance, in image steganography, it is important to hide as much as possible data inside the carrier image without increasing its brightness, without making it blurry, without pixelizing it, and without changing its size. This would be a key element in making the hidden data imperceptible and the carrier image innocent and unsuspicious.

Imperceptibility: It refers to the ability of the steganography algorithm to hide data in an undetectable way so much so that no one can see any visible artifacts or distortions in the carrier file. It therefore avoids drawing suspicions and obscures the fact that a secret communication is taking place.

Irrecoverability: It refers to how much an intercepted carrier file can be easily decoded and reverse-engineered so as to extract the data hidden inside it. An irrecoverable steganography algorithm makes it hard for eavesdroppers and unauthorized third parties to recover the hidden data from the carrier file despite knowing that steganography has been employed.

**1.3 Scope of the project**

To maintain the confidentiality of the data it is very useful if the data being transferred is not visible to an unauthorized user. Steganography provides us with potential capability to hide the existence of confidential data, hardness of detecting the hidden data and strengthening of the secrecy of the encrypted data. Steganography along with providing secrecy to the embedded data also unifies two types of data into one. Media data can have some association with other information. So, data can easily be transferred from one system to another without hitch.

Our project concentrates on the filtering algorithm for the process of steganography. We now give you an in depth explanation of the project. But before we go in details, we will need you to have an understanding of various techniques of steganography other than filtering technique, and to know how they vary from each other. The next section gives you an idea of the same.

**CHAPTER 2**

**LITERATURE SURVEY**

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1 Parameters to determine the image quality:**

For comparing stego image with cover image results it requires a measure of image quality. Commonly used measures are Mean-Square Error, Peak Signal-to-Noise Ratio, Signal to Noise, Average Absolute Deviation and histogram.

* + 1. **Mean Square Error (MSE):**

The mean square error is equal to the expected value of the square of the difference between the value taken by the estimator and the true value of the parameter.

Description: \mathit{MSE} = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2

MSE value should be as low as possible.

* + 1. **Peak Signal-to-Noise Ratio (PSNR)**:

As a performance measurement for image distortion, the well-known Peak-Signal-to-Noise Ratio (PSNR) which is classified under the difference distortion metrices can be applied on the stego images.

Description: \begin{align}\mathit{PSNR} &= 10 \cdot \log_{10} \left( \frac{\mathit{MAX}_I^2}{\mathit{MSE}} \right)\\ 
&= 20 \cdot \log_{10} \left( \frac{\mathit{MAX}_I}{\sqrt{\mathit{MSE}}} \right)\\ 
&= 20 \cdot \log_{10} \left( {\mathit{MAX}_I} \right) - 10 \cdot \log_{10} \left( {{\mathit{MSE}}} \right)\end{align}

Here, MAXi is the maximum possible pixel value of the image.

PSNR value should be as high as possible.

* + 1. **Signal to noise ratio (SNR):**

It is a physical measure of the sensitivity of an imaging system.



SNR value should be as high as possible.

* + 1. **Average absolute deviation (AAD):**

For a data set the average absolute deviation is the average of the absolute deviations from a central point and is a summary statistic of statistical dispersion or variability.



AAD value should be as low as possible.

**2.2 Least Significant Bit method**:

The technique converts image into shaded Gray Scale image. This image will act as a reference image to hide the text. In a gray scale image each pixel is represented in 8 bits. The last bit in a pixel is called as Least Significant Bit as its value will affect the pixel value only by “1”. So, this property is used to hide the data in the image. The last two bits can also be considered as the LSBs. This helps in storing extra data. This approach is very simple but is vulnerable to steganalysis.

**2.3 LSB using a secret key**

In basic LSB Algorithm hidden information is stored sequentially in the LSBs of image pixels. For this reason, knowing the retrieval methods, anyone can extract the hidden information. This problem of security of the hidden data is solved by the algorithm LSB with a secret key. In this technique the hidden information is stored into different position of LSB of image depending on the secret key. As a result, it makes the extraction of the hidden information knowing the retrieval method. In this steganography algorithm the secret key is used to hide the information into an input pixel of cover image without producing perceptible distortions. Here a bit of hidden information is placed in either LSB of Green or Blue matrix of a specific pixel which is decided by the secret key. So anyone cannot exactly make a decision that the bit of hidden information is placed in either LSB of Green or Blue matrix. As a result, the security level of image steganography is attained.

**2.4** **Randomized algorithm**

A good technique of image steganography aims at three aspects.

1. Capacity (the maximum data that can be stored inside cover image).

2. Imperceptibility (the visual quality of stego-image after data hiding).

3. Robustness.

The LSB based technique is good at imperceptibility but hidden data capacity is low because only one bit per pixel is used for data hiding. Simple LSB technique is also not robust because secret message can be retrieved very easily once it is detected that the image has some hidden secret data by retrieving the LSBs. Therefore, Randomized Algorithm is proposed to improve the security of LSB scheme. This method overcomes the sequence-mapping problem by embedding the message and stego key into a set of random pixels, which are scattered over the entire cover image. The bits of the secret message are embedded in pixels of the cover image that are generated by RC4 algorithm. The security of the data is improved by using Randomized algorithm by randomly dispersing the bits of the message in the image and thus making it harder for unauthorized people to extract the original message. The imperceptibility of the stego image is improved even more by using a bit inversion technique. In bit inversion technique, certain least significant bits of cover image are inverted after LSB steganography that co-occur with some pattern of other bits and that reduces the number of modified LSBs. Thus, less number of least significant bits of cover image is altered in comparison to plain LSB method, improving the PSNR of stego-image thus the quality of the stego image.

**2.5 Dynamic steganography**

Two cipher text bits are to be embedded in each pixel of the image. Each pixel is 8 bits. The embedding locations in a pixel are: 6th and 7th bit locations or 7th and 6th bit locations or 7th and 8th bit locations or 8th and 7th bit locations depending upon the cipher text bits. The 8th bit means the least significant bit (LSB). As the embedding locations are decided at the run time of the algorithm, so it is called as dynamic steganography. This proposed algorithm is highly secure as it uses dynamic embedding i.e. message dependent embedding. The amount of message that can be embedded is also very good. The technique is not susceptible to histogram based attacks.

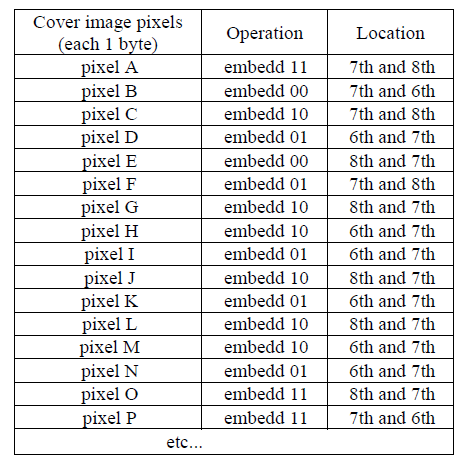


Table 2.1: Embedding locations in dynamic steganography

**2.6 Edge detection method:**

Multiple Edge Detection and Multiple Embedding increase the data hiding capacity of the image.

Along with Multiple Edge Detection, Minimum Error Replacement method is used to increase the PSNR. So data hiding capacity and PSNR both are increased. By this the security is maintained along with achieving more capacity and good visual quality.

Edge detection is a method used to detect the image edges. Image edge is the area in the image in which the gradient is steeply changing. The gradient in the image can be detected using operators like Sobel operator, Canny operator etc. In edge detection variable embedding is employed as there are two types of pixels, edge pixels and non-edge pixels. Edge pixels are heavily loaded and non-edge pixels are normally loaded. So this increases the data hiding capacity. Here we come across a factor known as the Variable Embedding Ratio [VER]. This ratio specifies the number of bits embedded in edge and non-edge pixels. It's specified as the ratio like 4:2, 4:1 etc. Variable embedding can also be used to increase the security as well. The receiver should know the Variable Embedding Ratio without which extraction is not possible. Variable Embedding Ratio has to be used sensibly. According to the information, Variable Embedding Ratio is chosen so as to increase the PSNR. At the same time it can be used to improve security. Only the authorized person who knows the VER can extract the data.

**2.6.1 Minimum Error Replacement Method**: Multiple Edge Detection and Multiple Embedding increase the data hiding capacity of the image. For that it compensates the PSNR. As data hiding capacity is important, the PSNR also important for good Steganography. Hence this paper employs Minimum Error Replacement method to increase the PSNR of the image. In this method the Stego-image is compared with the original image and the difference in the pixel value is calculated. Accordingly in the pixels in which the error vale is high the immediate higher bit of the pixel is changed. This brings down the error range. Hence the overall error reduces which in turn increases the PSNR.

**2.7 Filtering algorithm**:

The proposed algorithm changes very small number of bits when embedding a large cipher. Another point is that we do not just put the message bit into the LSB of the pixel value but we use LSB for status bit while retrieving data from stego image. From the table given below, we see that the PSNR value of the proposed technique is better than other methods providing an efficient way to embed a message into the image without producing clear distortion. The concept of this method of steganography is not to increase the capacity of the message but we try to make it difficult to the unauthorized person to determine the presence of a secret cipher.

In ordinary LSB steganography technique only message bit will be replaced with the LSB bit of the message but this algorithm does not just replace the message bit but it would replace the status of the message bit. Moreover, cryptography will be merged with it so that the secret message can be secured by two security layers. This technique fulfills the requirement of steganography technique.

**2.8 Comparison of the algorithms**

A Comparative study of these algorithms based on the PSNR values led us to the selection of the filtering algorithm. The results of the study along with the images used are shown below:



Fig 2.1:Baboon Fig 2.2:Leena

|  |  |  |
| --- | --- | --- |
| Algorithms | Images | PSNR |
| Dynamic Steganography | Leeena | 50.93 |
| LSB using a secret key | Leena  Baboon | 53.62  53.76 |
| Randomization Algorithm | Baboon | 56.05 |
| Edge Adaptive Algorithm | Leena | 68.38 |
| Filtering algorithm | Leena  Baboon | 74.38  74.53 |

Table 2.2: comparison of various algorithms studied on the basis of PSNR and MSE ratios

**CHAPTER 3**

**FILTERING ALGORITHM**

**CHAPTER 3**

**FILTERING ALGORITHM**

**3.1 Proposed Method**

This technique consists of two parts:

1. Embedding of the secret message

2. Extraction of the secret message

In a color image, color information is arranged byte after byte as R-G-B-R-G-B-R-…. etc. In the proposed technique, a data bit is embedded in the LSB of every such byte. We always choose the blue component of the pixel to embed the status bit as the changes on the blue component are less visible to the human eye. The choice of the bit with which the message bit is compared is decided by the MSB of the 3 components of the corresponding pixel.

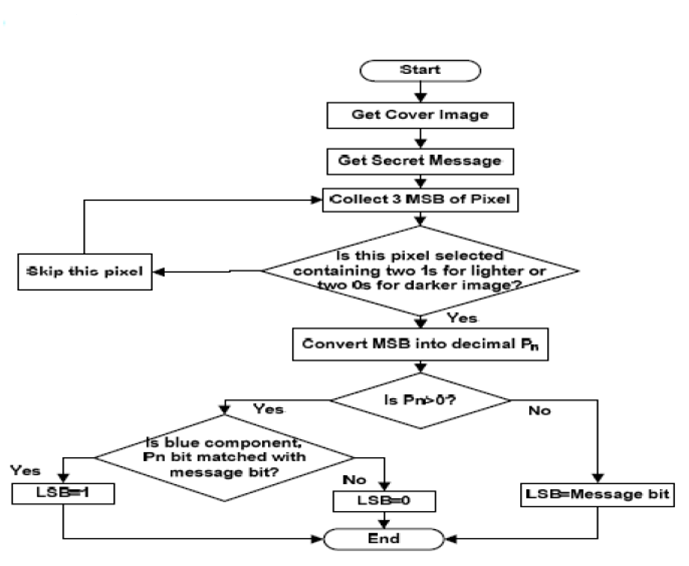


Fig 3.1: Embedding flowchart

**3.2 Embedding of the secret data**

For embedding the data we first need to find out if the image is a lighter image or a darker image

**3.2.1 Embedding data in lighter area**

If the MSB bits of the R-G-B components contain at least two 1’s, then this pixel is a lighter one and hence is chosen to hide the data bit. Let Pn be the decimal representation of the 3 MSB bits which is used to compute the new LSB bit. If the data bit Mn and the bit at the position Pn of 3rd byte i.e. blue component is same, then the LSB of 3rd byte is changed into 1, or 0 otherwise. In this way the data bit is inserted into the cover image.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pixel | Color Component | Component Value | MSB (3Bit) | Message Bit | Resulting Color Value |
| 1st | R | 00001111 | 0112= 310 | 1 | 00001111 |
| G | 10000101 | 00000101 |
| B | 10000000 | 10000000 |
| 2nd | R | 00001010 | 0102 = 210 | SKIP | 00001010 |
| G | 10011100 | 10011100 |
| B | 01001110 | 01001110 |
| 3rd | R | 11010000 | 1102 = 610 | 0 | 11010000 |
| G | 10001111 | 10001111 |
| B | 01001110 | 01001110 |
| 4th | R | 01001010 | 0112=310 | 0 | 01001010 |
| G | 10001101 | 10001101 |
| B | 10110000 | 10110001 |
| 5th | R | 10001011 | 1102=610 | 1 | 10001011 |
| G | 10111101 | 10111101 |
| B | 01001110 | 01001111 |

Table 3.1: Embedding message in lighter pixels

**3.2.2 Embedding message in darker area**

If the MSB bits of the R-G-B components contain at least two 0’s, then this pixel is a darker one and hence is chosen to hide the data bit. . Let Pn be the decimal representation of the 3 MSB bits which is used to compute the new LSB bit. . If the data bit Mn and the bit at the position Pn of 3rd byte i.e. blue component is same, then the LSB of 3rd byte is changed into 1, or 0 otherwise. In the darker algorithm we need to consider one more situation where all 3 bits of MSB are 0 and the decimal value of MSB bits is 0. In this case data bit will be directly embedded into the Blue color component. In this way the data bit is inserted into the cover image.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pixel | Color Component | Component Value | MSB (3Bit) | Message Bit | Resulting Color Value |
| 1st | R | 00001111 | 0102 = 210 | 1 | 00001111 |
| G | 10000101 | 10000101 |
| B | 00000000 | 00000000 |
| 2nd | R | 10001010 | 1102=610 | SKIP | 10001010 |
| G | 10011100 | 10011100 |
| B | 01001110 | 01001110 |
| 3rd | R | 11010000 | 1002 = 410 | 0 | 11010000 |
| G | 00001111 | 00001111 |
| B | 01001110 | 01001111 |
| 4th | R | 01001010 | 0012=110 | 0 | 01001010 |
| G | 00001101 | 00001101 |
| B | 10110000 | 10110001 |
| 5th | R | 00001011 | 0002=010 | 1 | 00001011 |
| G | 00111101 | 00111101 |
| B | 01001110 | 01001111 |

Table 3.2: embedding message in darker pixels

**3.2.3 Embedding algorithm**

1. Get the Original message.
2. Convert the original message into binary number.
3. Calculate the length of the message into a 32 bit string.
4. Concatenate the 32 bit string with the original message.
5. Get the cover image.
6. Check the image whether it is a lighter or darker image.
7. Collect the MSB bits from a pixel (Red, Green, Blue color component).
8. Select the lighter or the darker pixel depending on whether the image is lighter or darker. If it is a lighter image, while embedding the data into the image, skip the darker pixels.
9. Convert MSB into a decimal number ( Pn).
10. If Pn=0, for the darker image, embed the message bit directly into the Blue color component of the pixel.
11. For the lighter image:
    1. Check Pn bit position of the Blue color component with message bit.
    2. If it matches then change the LSB of Blue color component with 1(indicates the status as true).
    3. If it does not match with message bit then change the LSB of Blue color component with 0(indicates the status as false).

**3.3 Extracting of the secret data**

For extracting the hidden data, it has to be found out whether the image is a lighter image or a darker image

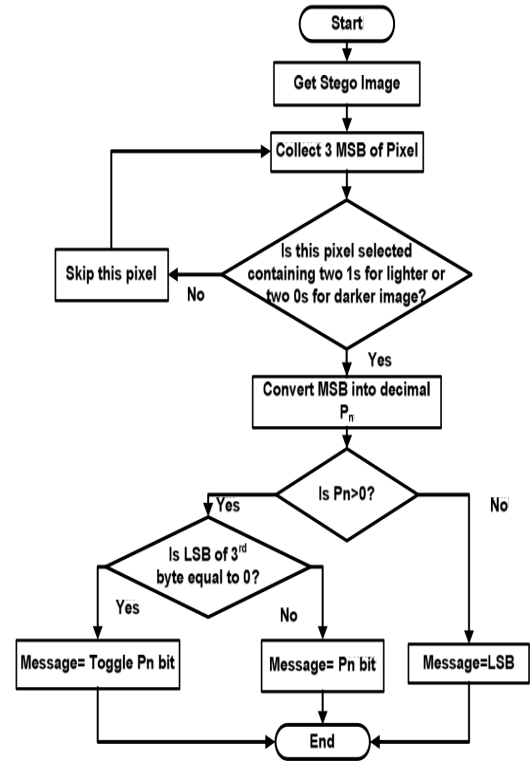


Fig 3.2: Extraction flowchart

**3.3.1 Extracting message from lighter pixel**

Collect the three bytes of the pixel of the image. Then check the MSB bit of the collected 3 bytes and check whether it contains at least 2 bits with value 1.Thisindicates that the pixel is a lighter pixel. The decimal equivalent (Pn) of this value is obtained as shown in the TABLE IV.The LSB bit of the third byte of the lighter pixels is checked to be a one or a zero. If it is a one, then the value in the Pn bit position will be extracted as it is and if the value is zero then the Pn bit position value is inverted before extraction. This will be continued till all the message bits are extracted from the stego image.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pixel | Color Component | Component Value | MSB  (3Bit) | Extracting Message Bit |
| 1st | R | 00001111 | 0112 = 310 | 1 |
| G | 10000101 |
| B | 10000000 |
| 2nd | R | 00001010 | 0102=210 | SKIP |
| G | 10011100 |
| B | 01001110 |
| 3rd | R | 11010000 | 1102 = 610 | 0 |
| G | 10001111 |
| B | 0100111**0** |
| 4th | R | 10001011 | 1102=610 | 1 |
| G | 10111101 |
| B | 0100111**1** |
| 5th | R | 00001111 | 0112 = 310 | 1 |
| G | 10000101 |
| B | 10000000 |

Table 3.3: Extracting message from lighter pixels

**3.3.2Extracting message from darker pixel**

Collect the 3 bytes of the pixel of the image. Then check the MSB bits of the collected 3 bytes and check whether it contains at least 2 bits with 0 value. This indicates the pixel is a darker pixel. Then the decimal equivalent (Pn) of this value is obtained as shown in Table 3.3.The LSB bit of the third byte of the darker pixels is checked to be a zero or one. If it is a one, then the value in the Pn bit position will be extracted as it is and if the value is zero then the Pn bit position value is inverted before extraction. If all the three bits of the MSBs of the pixel are zero then the value of the LSB is taken as it is as the message bit. This will be continued till all the message bits are extracted from the stego image.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pixel | Color Component | Component Value | MSB  (3Bit) | Extracting Message Bit |
| 1st | R | 00001111 | 0112 = 310 | 1 |
| G | 10000101 |
| B | 00000000 |
| 2nd | R | 10001010 | 0102=210 | SKIP |
| G | 10011100 |
| B | 01001110 |
| 3rd | R | 11010000 | 1102 = 610 | 0 |
| G | 00001111 |
| B | 00001111 |
| 4th | R | 00001011 | 0012=110 | 1 |
| G | 00111101 |
| B | 1100111**1** |
| 5th | R | 01001010 | 0012=110 | 0 |
| G | 00001101 |
| B | 10110001 |

Table 3.4: Extracting message from darker pixels

**3.3.3 Extracting algorithm**

* 1. Get the Stego image.
  2. Check whether the Stego image is lighter or a darker image
  3. If the image is a lighter image then consider only the lighter pixels and similarly for a darker image.
  4. The first 32 lighter or darker pixels will give the length of the message.
  5. Collect the MSBs of the Red, Green, Blue color component of the pixel.
  6. Convert MSB into a decimal equivalent Pn.
  7. If Pn=0 only get the binary value from the LSB of the blue component of the pixel.
  8. If Pn>0, check the LSB (Status bit) whether it is 0 or 1.
  9. If the LSB equals to 0, then collect the binary value by toggling the Pn bit of the Blue component of the pixel.
  10. If the LSB equals to 1, then collect the binary value at the Pn bit of the blue component of the pixel.

**CHAPTER 4**

**IMPLEMENTATION**

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 Graphical User Interface**

A graphical user interface (GUI) is a graphical display in one or more windows containing controls, called components that enable a user to perform interactive tasks. The user of the GUI does not have to create a script or type commands at the command line to accomplish the tasks. Unlike coding programs to accomplish tasks, the user of a GUI need not understand the details of how the tasks are performed.

GUI components can include menus, toolbars, push buttons, radio buttons, list boxes, and sliders just to name a few. GUI’s created using MATLAB tools can also perform any type of computation, read and write data files, communicate with other GUI’s, and display data as tables or as plots.

The following Figure 4.1 illustrates GUI for the system.

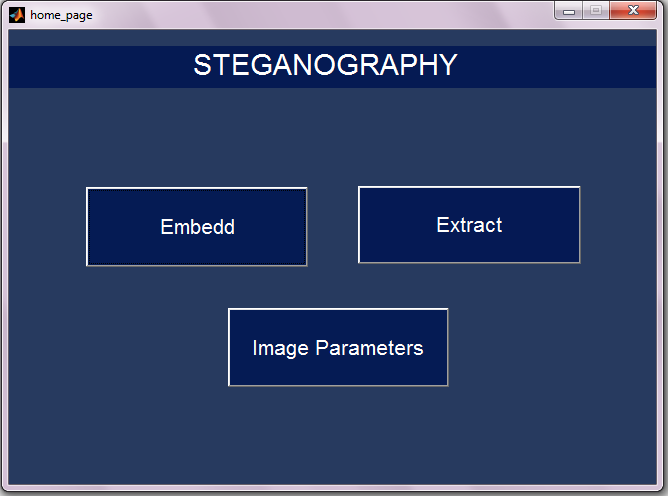


Fig 4.1: GUI for steganography

To enter the secret message the user will enter the embed option upon which a new pane will be opened as displayed below in Fig 4.2



Fig 4.2: GUI for embedding the secret message

The user has to enter the secret message and enter any cover image of his/her choice and click the create stego image option. On clicking the create stego image option the window is displayed to choose the desired location for saving the image as displayed below in Fig 4.2

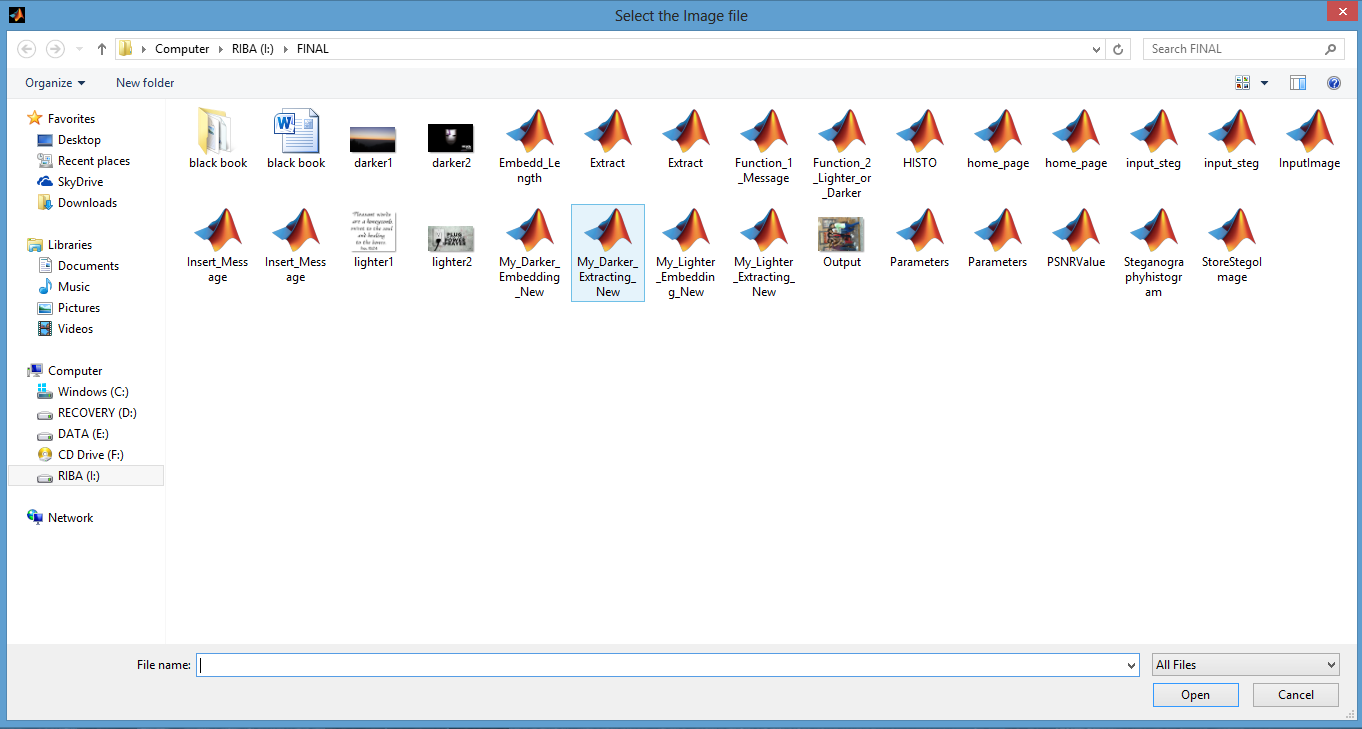


Fig 4.3: Location displayed for storing the stego image

The user has to save the image in the desired location.

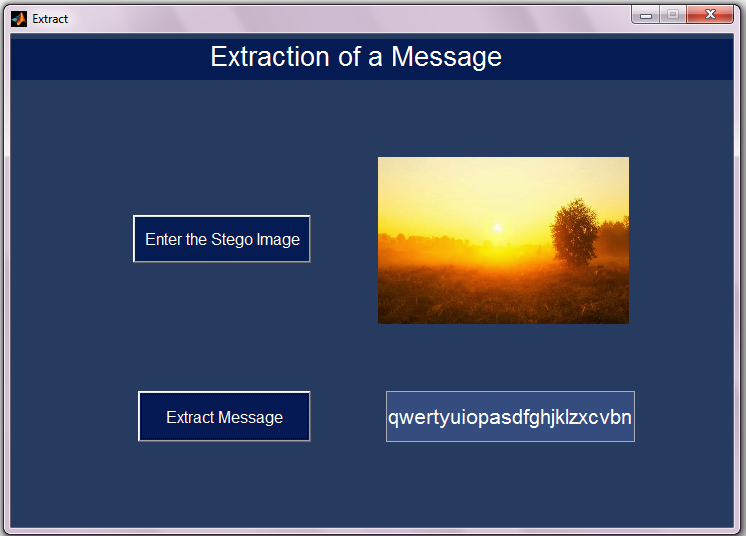


Fig 4.4: GUI for extracting the hidden message

For the extraction of the image the user has to click the extract option as shown in Fig.4.1.A new pane is displayed where the user has to enter the stego image. On clicking the extract option the embedded data is retrieved.

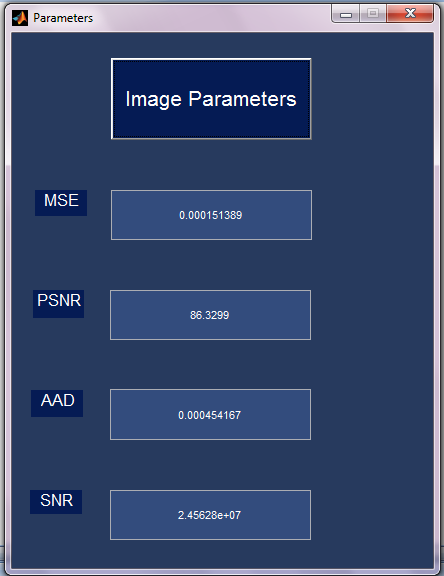


Fig 4.5: GUI for calculating the image parameters

This pane gets displayed on clicking the option of the image parameters in Fig4.1.The values of the image parameters will help the user to test the quality of the stego image.

**4.2 Technology Used:MATLAB version R2012b**

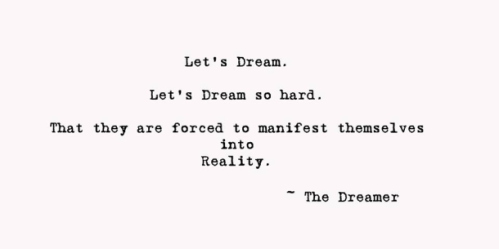
MATLAB is a high performance language for technical computing.it integrates Computation,Visualization,and programming in an easy to use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include math and computation algorithm development data acquisition modeling, simulation ,and prototyping data analysis,exploration,and visualization scientific and engineering graphics application development, including graphical user interface building. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning.

**CHAPTER 5**

**RESULTS AND ANALYSIS**

**CHAPTER 5**

**RESULTS AND ANALYSIS**

**5.1 Evaluation of image quality**





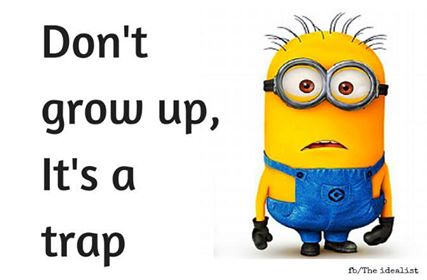


Fig 5.1:(From top -left)Lighter2.jpg,Darker2.jpg,newop.jpg,thetwotrees.jpg,Sareeday.jpg,index.jpg,index1.jpg,Baboon.jpg,Shoot.jpg,Kalaghoda.jpg,Cowboys.jpg,spirestorm.jpg,sgtr.png,IVfun.jpg,pleasantsun.jpg

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IMAGE | MSE | PSNR | AAD | SNR |
| Sareeday.jpg | 33.1828 e-05 | 82.2122 | 99.5483 e-05 | 1.84382 e+06 |
| Cowboys.jpg | 33.2913 e-05 | 82.1981 | 99.8739 e-05 | 1.51978 e+06 |
| Lighter2.jpg | 71.2889 e-05 | 78.8911 | 213.867 e-05 | 1.39290 e+06 |
| Kalaghoda.jpg | 77.1605 e-05 | 78.5474 | 231.481 e-05 | 1.40853 e+06 |
| Darker2.jpg | 84.8639 e-05 | 78.1341 | 254.592 e-05 | 1.33142 e+06 |
| Shoot.jpg | 129.718 e-05 | 76.2914 | 389.154 e-05 | 5.54041 e+06 |
| IVfun.jpg | 256.471 e-05 | 73.331 | 769.413 e-05 | 2.45100 e+06 |
| Newop.jpg | 326.172 e-05 | 72.2869 | 978.516 e-05 | 6.15199 e+06 |
| Baboon.jpg | 608.953 e-05 | 69.5755 | 1826.86 e-05 | 0.90452 e+06 |
| Pleasentsun.jpg | 654.306 e-05 | 69.2636 | 1962.92 e-05 | 0.56828 e+06 |
| Thetwotrees.jpg | 590.565 e-05 | 69.7087 | 1771.69 e-05 | 1.08848 e+06 |
| Spirestorm.jpg | 760.921 e-05 | 68.608 | 2282.76 e-05 | 0.13575 e+06 |
| Sgtr.png | 1371.7 e-05 | 66.0488 | 4115.09 e-05 | 1.19999 e+06 |
| Index.jpg | 3163.36 e-05 | 62.4199 | 9490.08 e-05 | 0.06539 e+06 |
| Index1.jpg | 3170.72 e-05 | 62.4098 | 9512.17 e-05 | 0.09510 e+06 |

Table 5.1:Image parameters

The results show that the quality of the image does not get degraded after the addition of the message as the PSNR,SNR values are high and low MSE and AAD values.

**5.2 Histogram analysis**

Once the message has been embedded in cover image changes in the pixel value should not be noticed in the stego image.it is very hard to find the small color change in the cover and stego image by the human eye. For this histogram comparison is used by steganalyst to identify the stegoimage by comparing the histogram of cover image and stego image. Based on this experiments have been conducted and separate histograms are drawn for cover and stego image.it is observed that there is no major difference visible while comparing the stego image and the cover image.

Fig 5.2: Lighter2-The input(Left) and the corresponding output(Right)

Fig 5.3: Lighter2-Histogram of th input(Left) and the corresponding histogram of output(Right)

Fig 5.4: Darker2-The input(Left) and the corresponding output(Right)



Fig 5.5:Darker2-Histogram of th input(Left) and the corresponding histogram of output(Right)

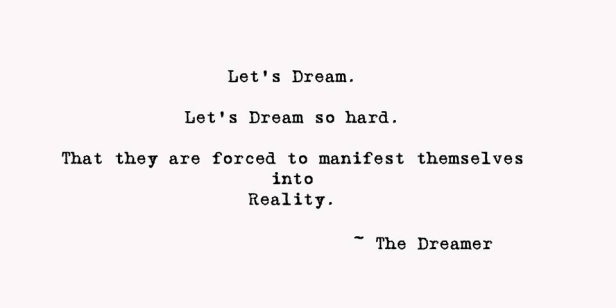
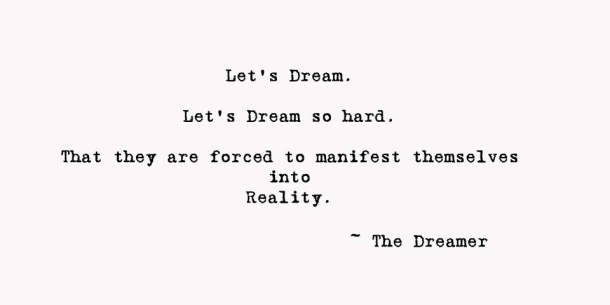


Fig 5.6:Image newop-The input(Left) and the corresponding output(Right)

Fig 5.7:Image newop-Histogram of the input(Left) and the corresponding histogram of output(Right)

Fig 5.8:Image Sareeday-The input(Left) and the corresponding output(Right)

Fig 5.9: Image Sareeday -Histogram of the input(Left) and the corresponding histogram of output(Right)

**5.3 Capacity**

Capacity is a key parameter in evaluating the strength of the steganographic technique. Capacity tells us the amount of data that can be hidden in the image.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IMAGE | LIGHTER/DARKER | CAPACITY(no. of characters) | SIZE(m\*n) | % capacity |
| Sareeday.jpg | LIGHTER | 314326.25 | 1920\*2560=4915200 | 51.15 |
| Cowboys.jpg | LIGHTER | 314587 | 2560\*1920=4915200 | 51.20 |
| Lighter2.jpg | LIGHTER | 232920 | 2000\*1125=2250000 | 82.81 |
| Kalaghoda.jpg | LIGHTER | 164750 | 1920\*1080=2073600 | 63.56 |
| Darker2.jpg | DARKER | 206650 | 1680\*1050=1764000 | 93.71 |
| Shoot.jpg | LIGHTER | 102500.875 | 1280\*956=1223680 | 67.01 |
| Europp.jpg | DARKER | 120810 | 1000\*1000=1000000 | 96.64 |
| IVfun.jpg | LIGHTER | 47689 | 1056\*592=625152 | 61.02 |
| Newop.jpg | LIGHTER | 056512 | 960\*480=460800 | 98.11 |
| Baboon.jpg | LIGHTER | 017693 | 512\*512=262144 | 53.99 |
| Pleasentsun.jpg | LIGHTER | 22375.5 | 600\*400=240000 | 74.58 |
| Thetwotrees.jpg | LIGHTER | 24996.125 | 688\*391=269008 | 74.33 |
| Spirestorm.jpg | DARKER | 23661.875 | 386\*541=208826 | 90.64 |
| Sgtr.png | LIGHTER | 013009 | 427\*280=119560 | 87.04 |
| Index.jpg | DARKER | 5396.3 | 300\*168=50400 | 85.65 |
| Index1.jpg | DARKER | 5410.6 | 275\*183=50325 | 86.01 |

Table 5.2:Image parameters

The results show appreciable capacity for the filtering technique implemented in our project.

**CHAPTER 6**

**CONCLUSION**

**CHAPTER 6**

**CONCLUSION**

**6.1 Conclusion**

The filtering technique helps in the transfer of messages by hiding it in an image so as to prevent its access from an unauthorized user. The embedding of the secret message should be in such a way so as to not degrade the quality of the image as well as to not introduce any visual artifacts in the stego image. Our proposed solution fulfills both criteria with high MSE and PSNR values for high quality and low MSE and AAD which fulfills the criteria for preventing visual artifacts. The main goal of the project was to make it difficult to the unauthorized user to detect the presence of a hidden message. The histogram analysis shows that the difference between the original and the stego image is negligible which fulfills the criteria of imperceptibility. Even If an unauthorized user realizes the presence of a secret message in the image it will be difficult for him/her to retrieve the data is not being inserted directly into the image but only the status bit is being inserted.Thus making the algorithm highly efficient.

**6.2 Future scope**

The algorithm can be improved by finding out methods to increase the capacity of the image for hiding the secret data.

Methods for retrieving the data in case of addition of noise to the stego image also could be worked upon.

**BIBLIOGRAPHY**

|  |  |
| --- | --- |
| [1] | T.Morkel,J.H.P.Eloff,and M.S.Olivier, “An overview of image steganography”New knowledge today conference,Sandton,pp.1-11,2005. |
| [2] | C.Kurak and J.McHugh A Cautionary, “Note on Image Downgrading”,Proc.IEEE 8th Annual Computer Security Applications Conference,San Antonio,USA,pp.153-155,1992. |
| [3] | E.Cole,Hiding in plain sight:Steganography and the Art of Covert communication,Indianapolis:Wiles publishing,2003. |
| [4] | S. Katzenbeisser and F. Petitcolas, Information hiding techniquesfor steganography and digital watermarking, Artech House Books,1999. |
| [5] | K.Hempstalk,“Hiding behind corners: using edges in images for better steganography”, Computing Womens Congress Conference,Hamilton, New Zealand, 2006. |
| [6] | Shejul and U.L. Kulkarni, “A Secure Skin Tone basedSteganography (SSTS) using Wavelet Transform”, International Journal of Computer Theory and Engineering, Vol.3, No.1, pp. 16-22, 2011. |
| [7] | GJ. Simmons,"The Prisoners' problem and the subliminal channel," in proc. CRYPTO'83, pp. 51-67, 1983. |
| [8] | C. K. Chan, L. M. Cheng, “Hiding data in image by simple LSB substitution”, pattern recognition, Vol. 37, No. 3, 2004, pp. 469-474. |
| [9] | Abbas Cheddad,Joan Condell,Kevin Curran and Paul McKevitt, “Digital image Steganography:Survey and analysis of current methods,” signal processions.vol 90,pp.727-752,2010 |
| [10] | E.Franz and A.Schneidewind,“Adaptive Steganography based on dithering”,workshop on multimedia and security,pp.56-62,2004 |
| [11] | J. F. Canny, “A computational approach to edge detection,” IEEE Trans.Pattern Anal. Machine Intell., vol. PAMI-8, no. 6, pp. 679-697, 1986. |
| [12] | D. Sandipan, A. Ajith, S. Sugata, An LSB Data Hiding Technique Using Prime Numbers, The Third International Symposium on Information Assurance and Security, Manchester, UK, IEEE CS press, 2007 |
| [13] | Westfeld, A., Pfitzmann, A.: Attacks on Steganographic Systems. In: 3rd International Workshop on Information Hiding (IHW 99), pp. 61-76. (1999) |
| [14] | Fridrich, J., Goljan, M., Du, R.: Detecting LSB Steganography in Color and Gray Images. Magazine of IEEE Multimedia (Special Issue on Security),October-November, pp. 22-28. (2001) |
| [15] | Dumitrescu, S., X. Wu and Z. Wang, Detection of LSB steganography via sample pair analysis, Springer LNCS, vol.2578, pp.355-372, 2003. |
| [16] | C.C. Thien, J.C. Lin, A simple and high-hiding capacity method for hiding digit-bydigit data in images based on modulus function, Pattern Recognition 36 (2003) 2875–2881. |
|  |  |
|  |  |
|  |  |