Steganographic application with encoding and decoding capabilities using Bit-Plane Complexity Segmentation Algorithm

J Component Project

Arshdeep Singh Bhatia

19BCB0086

B.Tech. Computer Science and Engineering with specialization in Bioinformatics

Kartik Nahta

19BCE2009

B.Tech. Computer Science and Engineering

Shivam Singhal

19BCE2112

B.Tech. Computer Science and Engineering

School of Computer Science and Engineering

Vellore Institute of Technology

Vellore

August 2022

Contents

Abstract

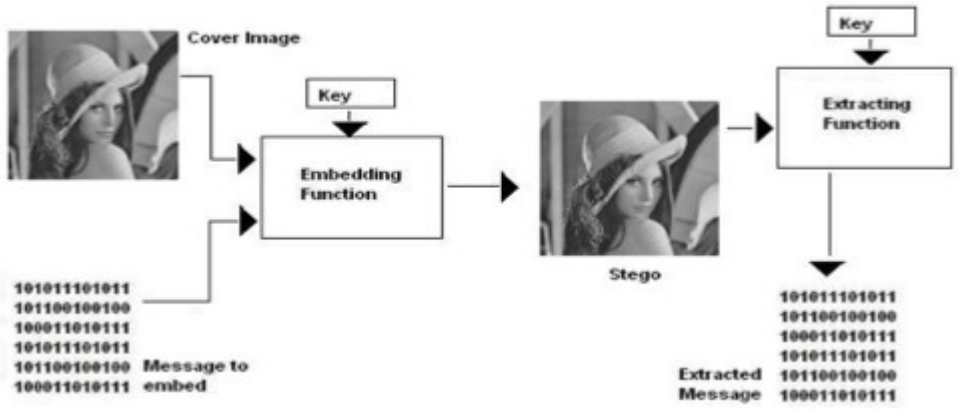
This document aims to implement a steganographic system that is able to perform the standard steganographic functions using the Bit-Plane Complexity Segmentation approach. The motivation for the same stems from the immense amount of piracy and multimedia in today's world and we think that such a system has wide ranging implications. At the end of this we have developed a python-based project that is able to encode and decode files into images. The default threshold used is 0.45 but the encryption can be done using custom alpha metrics as well.

Keywords: Information hiding, Steganography, Encryption, Bit plane slicing, complexity

Introduction

Motivation

Steganography is the science or art of hide the messages into other sources of information like text/ documents, audios, videos and images etc. so that it is not visible to unauthorized users The following picture represents different types of steganographic techniques. The internet allows the ease of spreading information over large areas. This is a blessing as well as a curse as your friends all over the world can view your information but everyone else also can view your information.
Encrypting data is the most popular approach to protect information but the protection can be broken with enough computational power. So, an alternate approach to encrypting data would be to hide it by making this information look like something else. This way only friends would be able to realize its true content. In particular, if some important data is hidden inside of an image then everyone except your friends would view it as an image. At the same time your friends would still be able to retrieve the true information.



Proposal

We propose to implement a steganographic system that is able to perform all the standard steganographic functions using the Binary Pattern Complexity approach. The disadvantage of Least Significant Bit is that it is vulnerable to steganalysis and is not secure at all and thus we don't plan to use this implementation.

Methodology

A Steganography system made up of three components:

- Cover-object means which hides the secret message,
- Secret message (information which is meant to be hidden),
- Stego-object means a cover object with a message embedded inside it.

The main principle of BPCS technique is that, the binary image is divided into informative region and noise-like region. The secret data is hidden into noise-like region of the vessel image without any deterioration.

Spatial Domain Methods: Spatial domain Steganography technique refers to methods in which data hiding is performed directly on the pixel value of cover image in such a way that the effect of message is not visible on the cover image. The following are further classification of this method.

1. BPC: The Binary Pattern complexity approach is used to measure the noise factor in the image complexity. The noisy portion is replaced by binary Pattern and it is mapped from the secret data. The image will remain same when the reverse noise factor will determine.

2. LSB: LSB is one the technique of spatial domain methods. LSB is the simple but susceptible to lossy compression and image manipulations. Some bits are change directly in the image pixel values in hiding the data.

Algorithm

1. Convert the carrier image from any format into png format.
2. Segmentation on carrier image is performed i.e., each bit-plane of the carrier image into informative and noise-like regions by using a threshold value (α0). That means complexity of image is calculated.
3. Group the bytes of the secret file into a series of secret blocks.
4. If a block is less complex than the threshold (α), then conjugate it to make it a more complex block.
5. The conjugated block must be more complex than α.
6. Embed each secret block into the complex regions of the bit-planes (or, replace all the noise-like regions with a series of secret blocks) where maximum colour changes are observed.
7. Convert the embedded dummy image and store.

Literature Survey

Table 1-Literature survey on Steganography

Title of research	Conclusions	Research Gaps
A Survey on different techniques of steganography Harpreet Kaur1, a and Jyoti Rani1 1 CSE	• Introduces the domains of steganography • Introduces various techniques of steganography	• The survey involves few to no statistics and simply illustrates the various techniques • No implementation parameters were used
KHAIRE, SHRIKANT & Nalbalwar, Sanjay. (2010). Review: Steganography – Bit Plane Complexity Segmentation (BPCS) Technique. International Journal of Engineering Science and Technology.	• Very detailed explanations for fundamental concepts and algorithms are discussed • Statistics and tables and good to represent proposed information	• Only specialized for a particular algorithm ie Bit plane complexity segmentation technique • Poor information about security and scalability algorithms • Implementation is explained on a general level and lacks few specifications
V. Verma, Poonam and R. Chawla, "An enhanced Least Significant Bit steganography method using midpoint circle approach," 2014 International Conference on Communication and	• Very detailed explanations for fundamental concepts and algorithms are discussed • Various techniques related to each other are compared	• Detailed about only a particular algorithm (Least Significant bit) and its sister techniques which have slight modifications • Implementation details are very brief and only theoretical
Amandeep kaur, manpreet, “Improved Security Mechanism of Text in Video using Steganographic Technique,” Int. J. Adv. Res. Comput. Sci. Softw.	• Appropriately addresses security prospects and aspects about the steganography as a whole • Discussions are extended to video files too	• The technical parameters are simply described and more detail could have been helpful • Survey based parameters are not completely described

J. Gupta, "A Review on Steganography techniques and methods," vol. 1, no. 1, pp. 1–4,2015)	• Review was detailed and covered all modern techniques • Statistics provided us a clear idea to select BPCS for implementation	• More implementation details could have been appreciated
A.Habes, (Feb 2006): Information Hiding in BMP image Implementation, Analysis and Evaluation, Information Transmission in Computer Networks.	• Detailed discussion and information about the topic is provided • The paper also covers security concerns about the techniques used	• Discussion is limited to only one technique and lacks many implementation details • Since the method discussed is not related to our topic the papers section about security alone was studied
N. Johnson and S. Jajodia, (Feb 1998): Exploring steganography: seeing the unseen, IEEE Computer, pp.26-34	• Introduces the domains of steganography • Introduces various techniques of steganography	• The survey involves few to no statistics and simply illustrates the various techniques • No implementation parameters were used

Table 2-Literature survey on BPCS Algorithms

Title of research	**Conclusions**	**Research Gaps**
A Review of Comparison Techniques of Image Steganography By Stuti Goel, Arun Rana & Manpreet Kaur Kurukshetra University	• Introduces the domains of steganography • Introduces various techniques of steganography	• The survey involves few to no statistics and simply illustrates the various techniques • No implementation parameters were used

Gonzalez, R., Woods, R., Pearson, P., & Hall. (n.d.). Digital Image Processing Third Edition Pearson International Edition prepared by Pearson Education.	• Very detailed explanations for fundamental concepts and algorithms are discussed • Statistics and tables and good to represent proposed information	• Only specialized for a particular algorithm ie Bit plane complexity segmentation technique • Poor information about security and scalability algorithms • Implementation is explained on a general level and lacks few specifications

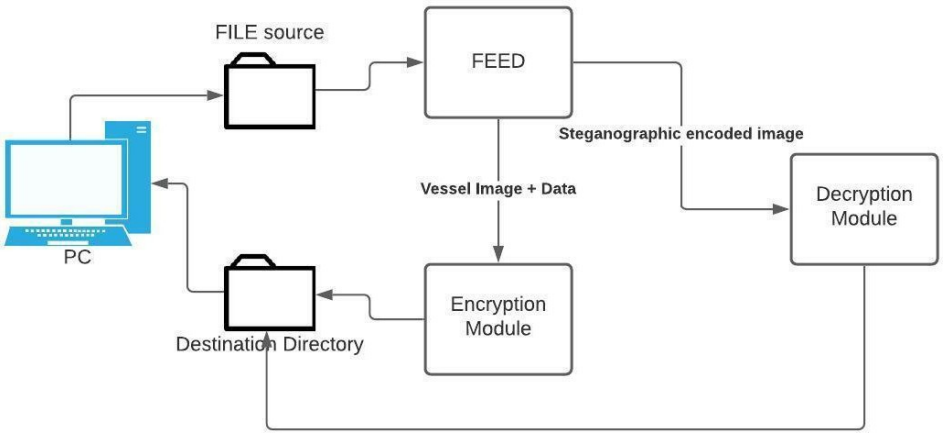
Habes, A. (2006). Information Hiding in BMP image Implementation, Analysis and Evaluation.	• Very detailed explanations for fundamental concepts and algorithms are discussed • Various techniques related to each other are compared	• Detailed about only a particular algorithm (Least Significant bit) and its sister techniques which have slight modifications • Implementation details are very brief and only theoretical
Hirohisa, Hioki. (2002). A data embedding method using BPCS principle with new complexity measures.	• Appropriately addresses security prospects and aspects about the steganography as a whole • Discussions are extended to video files too	• The technical parameters are simply described and more detail could have been helpful • Survey based parameters are not completely described
J. Gupta, "A Review on Steganography techniques and methods," vol. 1, no. 1, pp. 1–4,2015)	• Review was detailed and covered all modern techniques • Statistics provided us a clear idea to select BPCS for implementation	• More implementation details could have been appreciated
Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. Computer	• Detailed discussion and information about the topic is provided • The paper also covers security concerns about the techniques used	• Discussion is limited to only one technique and lacks many implementation details • Since the method discussed is not related to our topic the papers section about security alone was studied
Kawaguchi, E., & Eason, R. O. (1999). Principles and applications of BPCS steganography.	• Introduces the domains of steganography • Introduces various techniques of steganography	• The survey involves few to no statistics and simply illustrates the various techniques • No implementation parameters were used

Table 3 – Literature survey on SDLC

Title of research	Conclusions	Research Gaps
A Study of Importance of UML diagrams: With Special Reference to Very Large-sized Projects – 2013	• Introduces all standard techniques used for UML diagrams • Properly discusses the various scenarios where the techniques can be used	• Sticks to standard rules only • Does not discuss how hybridization can take place • Poorly describes the other steps of development
WATERFALL Vs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC - 2012	• Describes in detail the scenarios where each technique is useful • Provides a basis on which further research of Agile model and Waterfall model was chosen	• Only sticks to the standard rules as defined and no room for cherry picking • Involves many steps which may not be needed for a small scale and is only useful for large scale systemwide development
A Review Paper on Human Computer Interaction - 2018	• Review paper perfectly discusses modern HCI practices • Provides all insights to improvise UI for optimal frontend development	• Lacks implementation details • Frontend needs to adapt to functionality • Involves user interaction to be effective

Overall Architecture

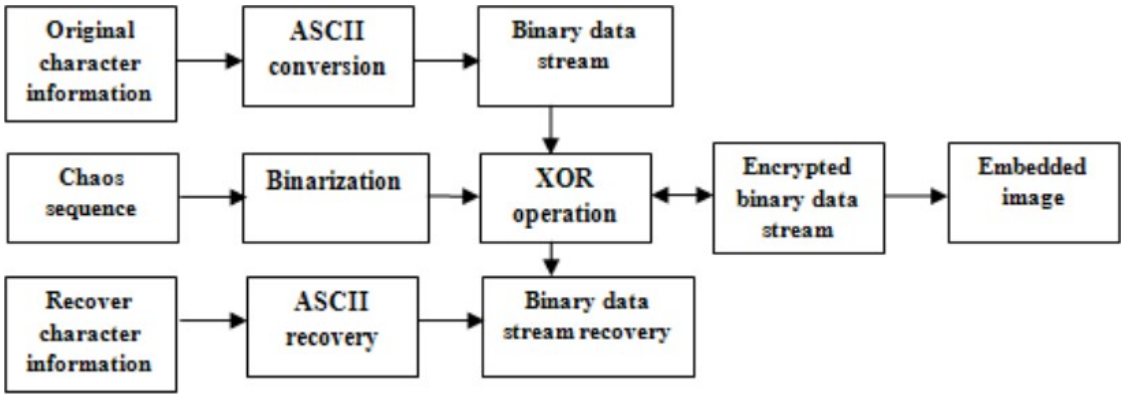
SYSTEM ARCHITECTURE



Module wise explanation

SNO	MODULE	INPUT	PROCESSING/FUNCTION	OUTPUT
1	FEED	VESSEL image + text	Checks data for consistency and forwards to encryption module	Data forwarded to encryption module
2	FEED	Steganographic image	Data is sent to decryption module	Data forwarded to decryption module
3	Encryption Module	FEED	• Libraries in Python split the vessel image into segments • Data is embedded into the segments • Output image is generated into local storage	Steganographic image which has data embedded
4	Decryption Module	FEED	• The image is spliced and each segment is manually checked for embedded data • All the data is written into a separate file • The spliced image without the embedded data is stored as another file.	Vessel Image and data is extracted

Detailed Flow diagram for encryption and decryption modules



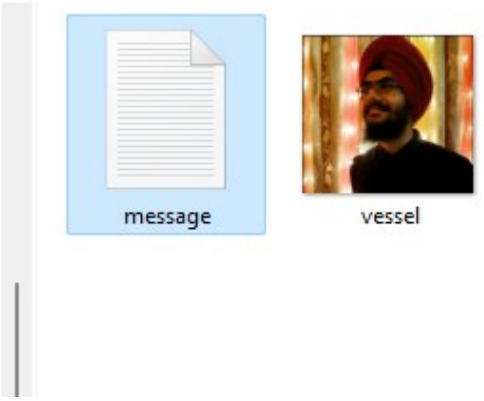
Proposed Methodology

1. Convert the carrier image from any format into png format.
2. Segmentation on carrier image is performed i.e., each bit-plane of the carrier image into informative and noise-like regions by using a threshold value (α0). That means complexity of image is calculated.
3. Group the bytes of the secret file into a series of secret blocks.
4. If a block is less complex than the threshold (α), then conjugate it to make it a more complex block.
5. The conjugated block must be more complex than α.
6. Embed each secret block into the complex regions of the bit-planes (or, replace all the noise-like regions with a series of secret blocks) where maximum colour changes are observed.
7. Convert the embedded dummy image and store.

Results and Outputs

Encoding process

1. Initial state of the system - the image and the message to be encoded is present in the examples directory.



2. Performing the encoding

   the command accepts the following arguments

   -m module (encode)

   -i input or vessel image

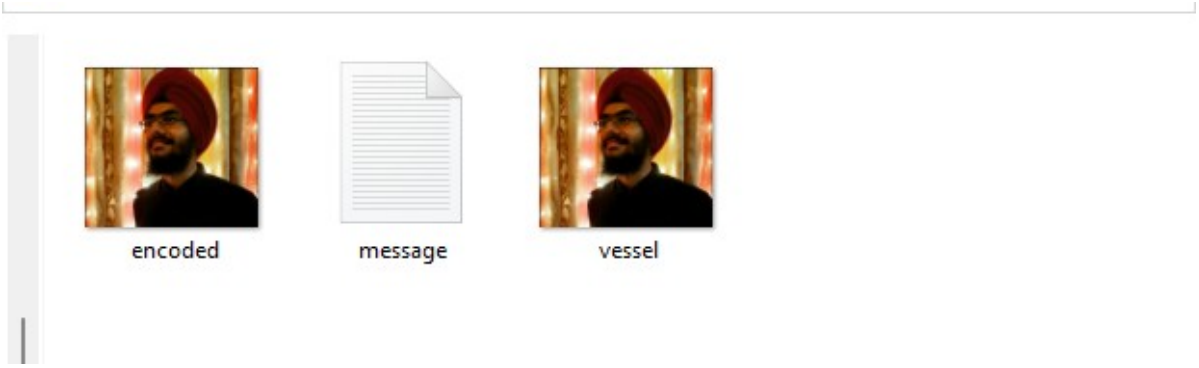
   -m message file

   -a alpha/threshold value

   -o output file(encoded file)

```
Slicing...
Graying...
Loaded image as array with shape (798, 862, 3, 8)
Found 259200 grids
Grid 10000 of 259200
Grid 20000 of 259200
Grid 30000 of 259200
Grid 40000 of 259200
Grid 50000 of 259200
Grid 60000 of 259200
Grid 70000 of 259200
Grid 80000 of 259200
Grid 90000 of 259200
Grid 100000 of 259200
Grid 110000 of 259200
Grid 120000 of 259200
Grid 130000 of 259200
Grid 140000 of 259200
Grid 150000 of 259200
Grid 160000 of 259200
Grid 170000 of 259200
Grid 180000 of 259200
Grid 190000 of 259200
Grid 200000 of 259200
Grid 210000 of 259200
Grid 220000 of 259200
Grid 230000 of 259200
Grid 240000 of 259200
Grid 250000 of 259200
Embedded 349 message grids and 10 conjugation maps
Ungraying...
Stacking...
Loaded new array as image
```

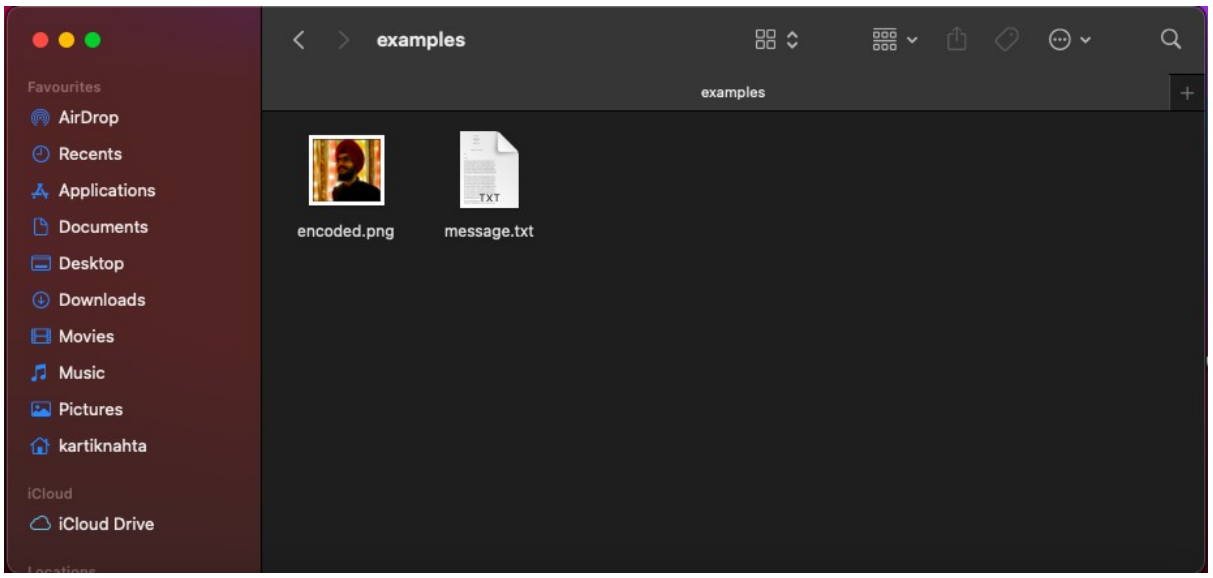
3. Updated system state - after generation of encoded.png it gets added to the examples directory.The encoded and vessel image is the same in appearance.



Decoding Process

Decoding in a new system to which image was transferred.

4. State of the new system - the encoded image to be decoded is present in the examples directory. Message.txt file to validate the message_decoded.txt file.



5. Performing the decoding
   the command accepts the following arguments

   -m module (decode)

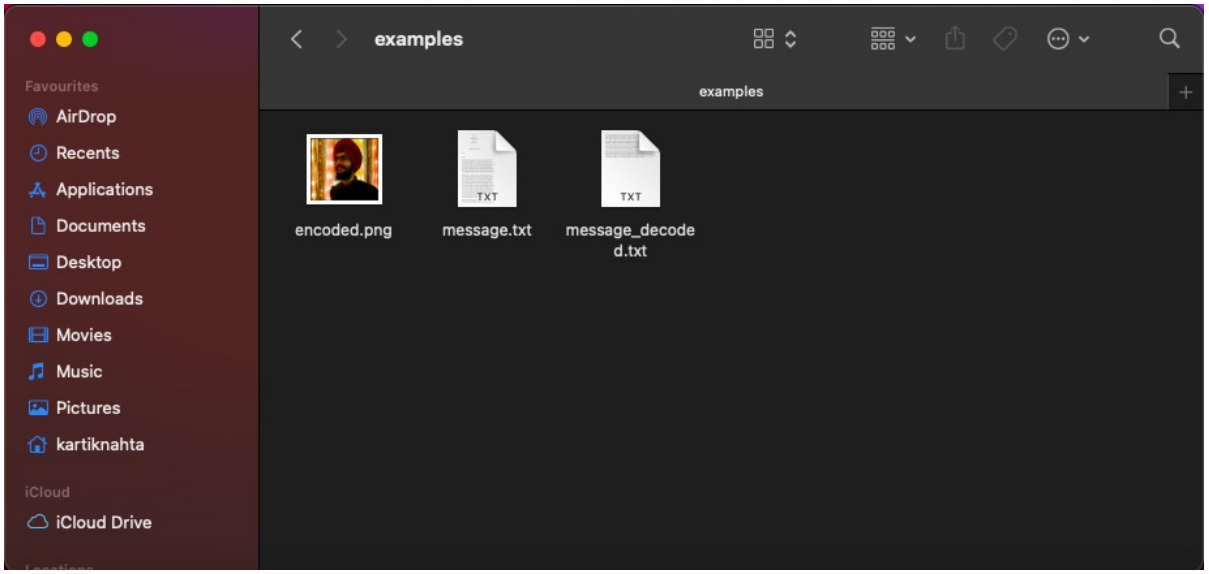
   -i encoded image

   -a alpha/threshold value

   -o output file(decoded text file)

```
Slicing...
Graying...
Loaded image as array with shape (798, 862, 3, 8)
Found 259200 grids
Grid 10000 of 259200
Grid 20000 of 259200
Grid 30000 of 259200
Grid 40000 of 259200
Grid 50000 of 259200
Grid 60000 of 259200
Grid 70000 of 259200
Grid 80000 of 259200
Grid 90000 of 259200
Grid 100000 of 259200
Grid 110000 of 259200
Grid 120000 of 259200
Grid 130000 of 259200
Grid 140000 of 259200
Grid 150000 of 259200
Grid 160000 of 259200
Grid 170000 of 259200
Grid 180000 of 259200
Grid 190000 of 259200
Grid 200000 of 259200
Grid 210000 of 259200
Grid 220000 of 259200
Grid 230000 of 259200
Grid 240000 of 259200
Grid 250000 of 259200
Found 359 out of 259200 grids with complexity above 0.45
Found 349 message grids and 10 conjugation maps
```

6. Updated system state - after generation of message_decoded.txt it gets added to the examples directory. You can verify whether the message_decoded.txt is the same as message.txt.



Analysis

a) LSB Based Steganography

Algorithm to embed text message:-

Step 1: Read the cover image and text message which is to be hidden in the cover image.

Step 2: Convert text message in binary.

Step 3: Calculate LSB of each pixels of cover image.

Step 4: Replace LSB of cover image with each bit of secret message one by one.

Step 5: Write stego image.

Step 6: Calculate the Mean square Error (MSE), Peak signal to noise ratio (PSNR) of the stego image.

Algorithm to retrieve text message:-

Step 1: Read the stego image.

Step 2: Calculate LSB of each pixels of stego image.

Step 3: Retrieve bits and convert each 8 bit into

character.

b) BPCS Based Steganography:

Algorithm to embed text message:-

1. Convert the carrier image from any format into png format.
2. Segmentation on carrier image is performed i.e., each bit-plane of the carrier image into informative and noise-like regions by using a threshold value (α0). That means complexity of image is calculated.
3. Group the bytes of the secret file into a series of secret blocks.
4. If a block is less complex than the threshold (α), then conjugate it to make it a more complex block.
5. The conjugated block must be more complex than α.
6. Embed each secret block into the complex regions of the bit-planes (or, replace all the noise-like regions with a series of secret blocks) where maximum colour changes are observed.
7. Convert the embedded dummy image and store.

Algorithm to retrieve text message:-

Step 1: Read stego image.

Step 2: Each block is compressed through quantization table.

Step 3: Calculate Complexity of each alpha coefficient.

Step 7: Retrieve and convert each 8 bit into character.

c) DWT Based Steganography

Algorithm to retrieve text message:-

Step 1: Read the cover image and text message which is to be hidden in the cover image.

Step 2: Convert the text message into binary. Apply 2D Haar transform on the cover image.

Step 3: Obtain the horizontal and vertical filtering coefficients of the cover image. Cover image is added with data bits for DWT coefficients.

Step 4: Obtain stego image.

Step 5: Calculate the Mean square Error (MSE), Peak signal to noise ratio (PSNR) of the stego image.

Algorithm to retrieve text message:-

Step 1: Read the stego image.

Step 2: Obtain the horizontal and vertical filtering coefficients of the cover image. Extract the message bit by bit and recomposing the cover image.

Step 3: Convert the data into message vector. Compare it with original message.

Table of comparisons

Features	**LSB**	**BPCS**	**DWT**
Invisibility	Low	High	High
Payload capacity	High	Medium	Low
Robustness against image manipulation	Low	Medium	High
Complexity of Development	Low	Medium	Very high

Conclusion and Future Work

Steganography transmits secrets through apparently innocuous covers in an effort to conceal the existence of a secret. Digital image steganography and its derivatives are growing in use and application. In areas where cryptography and strong encryption are being outlawed, citizens are looking at steganography to circumvent such policies and pass messages covertly. As with the other great innovations of the digital age: the battle between cryptography and cryptanalysis, security experts and hackers, record companies and pirates, steganography and Steganalysis will continually develop new techniques to counter each other. In the near future, the most important use of steganographic techniques will probably lie in the field of digital watermarking. Content providers are eager to protect their copyrighted works against illegal distribution and digital watermarks provide a way of tracking the owners of these materials. Steganography might also become limited under laws, since governments already claimed that criminals use these techniques to communicate.

The possible use of steganography technique is as following:

- Hiding data on the network in case of a breach.
- Peer-to-peer private communications.
- Posting secret communications on the Web to avoid transmission.
- Embedding corrective audio or image data in case corrosion occurs from a poor connection or transmission.

References

1. Anderson, R. J., & Petitcolas, F. A. P. (1998). On the limits of steganography. *IEEE Journal on Selected Areas in Communications*, *16*(4), 474–481. https://doi.org/10.1109/49.668971

2. Balaji, S., & Sundararajan, M. (2012). International Journal of Information Technology and Business Management WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC. *International Journal of Information Technology and Business Management*, *2*(1). https://mediaweb.saintleo.edu/Courses/COM430/M2Readings/WATEERFALLVs%20 V-MODEL%20Vs%20AGILE%20A%20COMPARATIVE%20STUDY%20ON%20SDLC.pdf

3. A Review of Comparison Techniques of Image Steganography By Stuti Goel, Arun Rana & Manpreet Kaur Kurukshetra University https://globaljournals.org/GJCST_Volume13/2-A-Review-of-Comparison.pdf
4. Bansal, H., & Khan, R. (2018). A Review Paper on Human Computer Interaction. *International Journal of Advanced Research in Computer Science and Software Engineering*, *8*(4), 53. https://doi.org/10.23956/ijarcsse.v8i4.630
5. Eiji Kawaguchi and Richard O. Eason. (n.d.). *Principle and applications of BPCS-Steganography*. Citeseerx.ist.psu.edu. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1018.3664&rep=rep1&type=pdf
6. English, R. (2010). Comparison of high capacity steganography techniques. *2010 International Conference of Soft Computing and Pattern Recognition*. https://doi.org/10.1109/socpar.2010.5686507
7. Gonzalez, R., Woods, R., Pearson, P., & Hall. (n.d.). *Digital Image Processing Third Edition Pearson International Edition prepared by Pearson Education*. http://sdeuoc.ac.in/sites/default/files/sde_videos/Digital%20Image%20Processing%203rd%20ed.%20-%20R.%20Gonzalez%2C%20R.%20Woods-ilovepdf-compressed.pdf
8. Habes, A. (2006). Information Hiding in BMP image Implementation, Analysis and Evaluation. *Èíôîðìàöèîííûå ïðîöåññû, Òîì*, *6*(1). http://www.jip.ru/2006/1-10-2006.pdf
9. Hirohisa, Hioki. (2002). *A data embedding method using BPCS principle with new complexity measures.* Research Gate. https://www.researchgate.net/publication/228695979_A_data_embedding_method_using_BPCS_principle_with_new_complexity_measures

10. Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. *Computer*, *31*(2), 26–34. https://doi.org/10.1109/mc.1998.4655281

11. Kaur, H., & Rani, J. (2016a). ASurvey on different techniques of steganography. *MATEC Web of Conferences*, *57*, 02003. https://doi.org/10.1051/matecconf/20165702003

12. Kaur, H., & Rani, J. (2016b). ASurvey on different techniques of steganography. *MATEC Web of Conferences*, *57*, 02003. https://doi.org/10.1051/matecconf/20165702003

13. Kawaguchi, E., & Eason, R. O. (1999). Principles and applications of BPCS steganography. *Multimedia Systems and Applications*. https://doi.org/10.1117/12.337436

14. KHAIRE, S., & Nalbalwar, Sanjay. (2010). *Review: Steganography – Bit Plane Complexity Segmentation (BPCS) Technique*. *Vol. 2(9), 2010, 4860-4868*.

15. Lin, E., & Delp, E. (n.d.). *A Review of Data Hiding in Digital Images*. Retrieved November 23, 2021, from https://www.imaging.org/site/PDFS/Papers/1999/PICS-0-42/1043.pdf

16. Nagendra M S, M., Manjula Yerva, M., & Kurian, M. (2021). *A Comparative Study on LSB methods for Image Steganography*. http://www.spjmr.com/gallery/spjmr%201515.pdf

17. Niimi, M., Noda, H., & Kawaguch, E. (1977). An image embedding in image by a complexity based region segmentation method. *Proceedings of International Conference on Image Processing*. https://doi.org/10.1109/icip.1997.631986

18. Steve Beaullieu, Jon Crissey, Ian Smith. (n.d.). *BPCS Steganography*. Citeseerx.ist.psu.edu; University of Texas at San Antonio.

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.87.5754&rep=rep1&type=pdf

19. Verma, V., Poonam, & Chawla, R. (2014). An enhanced Least Significant Bit steganography method using midpoint circle approach. *2014 International Conference on Communication and Signal Processing*. https://doi.org/10.1109/iccsp.2014.6949808
20. Waykar, Y. (2013, March). *(PDF) "A Study of Importance of UML diagrams: With Special Reference to Very Large-sized Projects."* ResearchGate. https://www.researchgate.net/publication/322991896_A_Study_of_Importance_of_UML_diagrams_With_Special_Reference_to_Very_Large-sized_Projects
21. Yeuan-Kuen Leea and Ling-Hwei Chen. (n.d.). *Secure Error-Free Steganography for JPEG Images*. Citeseerx.ist.psu.edu. Retrieved November 23, 2021, from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.15.7265&rep=rep1&type= pdf

Appendix

To view the code for the above project you can refer to the following GitHub link.
https://github.com/shivam24-2000/Image-Steganography