International Information and Engineering Technology Association

Advancing the World of Information and Engineering

Image and Text Steganography Using Convolutional Neural Network

Vijay Kumar*, Saloni Laddha, Aniket, Nitin Dogra

Computer Science and Engineering, NIT Hamirpur, H.P. 177005, India

Corresponding Author Email: vijaykumarchahar@gmail.com

ABSTRACT

Steganography has been used since centuries for concealment of messages in a cover media where messages were physically hidden. The goal in our project is to hide digital messages using modern steganography techniques. An N * N RGB pixel secret message (either text or image) is to be transmitted in another N * N RGB container image with minimum changes in its contents. The cover image also called the carrier can be publicly visible. In this project, along with LSB encoding, deep learning modules using the Adam algorithm are used to train the model that comprises a hiding network and a reveal network. The encoder neural network determines where and how to place the message, dispersing it throughout the bits of cover image. The decoder network on the receiving side, which is simultaneously trained with the encoder, reveals the secret image. The main aspect of this work is it produces minimal distortion to the secret message. Thus, preserving its integrity. Also, other steganography softwares cannot be used to reveal the message since the model is trained using a deep learning algorithm which complicates its steganalysis. The network is only trained once, irrespective of the different container images and secret messages given as inputs. Thus, this work has wide and secure applications in many fields.

1. INTRODUCTION

Secure transmission of messages is done using various cryptography and encryption techniques. The term steganography was coined centuries ago when messages were physically hidden. It is the strategy used for concealment or hiding of a message in another cover media. On observing the quilt media, one cannot work out if there is presence of any secret message embedded in it whereas in cryptography, the transmitted message over a channel is encrypted with a key and on observing it, it is known that the message is encrypted or not [1]. The goal of modern steganography is to securely transmit a digital message embedding it in a cover media. The steganography process exploits the properties of the human eye. The minute differences within the pixels of the quilt media from the initial media are not noticeable to the human eye when transmitted through a carrier. The carrier is also publicly visible [2, 3]. These hidden messages done using steganography are revealed by steganalysis. Both processes can be done on many different types of digital media such as image file, text messages, audio files, video messages etc. On applying the steganographic operations on the original media, a stego object is created. It may or may not be similar to the original media. The more similarity, the better is the steganography. Steganalysis can be of two types-passive and active. On investigating the stego object, if only the absence or presence of secret message is detected, it is passive. If that embedded secret message is retrieved from the stego object, it is said to be active steganalysis [4].

Steganographic techniques have been used as a medium to hide information since long. Terrorists use this technique to communicate among themselves without getting caught by the military. Secret messages, codes are embedded in cover media

and posted on public sites. The recipient has access to various steganography softwares or trained networks to extract the desired message. In 2011 Mumbai blasts, terrorists used an old photograph of a known Indian actress to hide important information. Later, on analyzing it completely, it was discovered to have some messages embedded in it. The photograph appeared completely normal to the naked eye initially. However, beside these misuses of steganography, it is also used widely in many fields such as biometrics- to match and hide fingerprints in respective identity cards of the employees, in digital watermarking, in military applications to transmit information during wartime, etc. The main focus is to preserve the integrity of the message to be transmitted.

The main challenge of image steganography is to keep the basic properties and visuals of the carrier image unaltered.

In this work, two main aspects are mainly considered- First being the amount and length of the message to be sent. Usually, in embedding a textual message in an image file, the message is measured in terms of bits-per-pixel (bpp) which is kept 0.4 or lower. Longer message means more bits need to be altered in the carrier image which may lead to some undesired visual changes. The second factor considered here is the type of carrier image chosen. It is more difficult to hide information in high frequency, high contrast, noise filled images than in flat curved regions of the same image.

In the methodology followed in the later sections, the first step is to use LSB encoding. In this method, the least significant bit of randomly selected pixels in the carrier image are replaced with the bits of the secret message. The final stego object is not visually different from its original version but on doing statistical analysis, it can be detected whether it is altered or unaltered. To overcome this, we later use a neural network to train our model on deep learning Adam algorithm.

Deep Neural Networks (DNNs) are used to decide which bits to alter in LSB encoding and how to replace those bits with the text message [5, 6].

In this work, in addition to determining where to hide the information bits in the carrier, we can also embed a larger message, almost as equal in size as the carrier image. A N * N RGB pixel secret message (both text or image) is transmitted in another N * N RGB container image with minimum changes in its contents. The model comprises a hidden network which performs operations to hide the message and a reveal network that retrieves the message back. The encoder and decoder networks are trained simultaneously. However, the secret message may not be completely retrieved. The secret message is made equal in size as the cover media. On steganalysis, it is still possible to detect the presence or absence of another message through deep second order statistical analysis. The network model need not be trained again and again for different input images. It does not depend on secret and cover images. Also, steganalysis cannot be done simply by using other steganography softwares. Thus, making it more secure and efficient.

This project mainly comprises of 2 modules-
- Concealment of a digital image in another cover image.
- Embedding of a textual message in an image.

2. RELATED WORK

Over the years, from hiding critical information physically in ancient times, we have now advanced to modern steganography with increasing digitization. Mainly, it can be categorized in 4 classes-
1. Digital media steganography
2. Linguistic steganography
3. Layered file system steganography
4. Network steganography

In textual steganography, after conversion of text to binary representation, their structure and positions are also altered before carrying it further.

In the previous works studied during the course of this project, a common notion observed was to develop a relationship between the original inputs given and the stego object obtained after steganography. This is a challenging task because of the changes undergone in the image file and its pixels after training data loss.

In this application based study by Karaman [7], different techniques used for steganography are described and compared based on simplicity measures. An application was developed for the guests attending a ceremony in the Republic of Turkey Presidency. The identity information of the guests-their IDs, ownership information along with RFID tags are first encrypted using cryptographic softwares. This crypted code was then hidden in their identity photos using LSB transformation on images. This made the identity cards visually similar to the original photographs; though the former one contained sensitive information embedded in it. Later in this paper, their performance was measured using VSL tool which compared the stego object with the input images and it was found that this technique was secure against all the major known cryptanalytic attacks except sharpening of image. On sharpening, the pixels might change their characteristics and reveal the presence of some additional bits. Thus, this approach was widely accepted in watermarking and securing

data. But, to make it more protected, further image processing could be done along with steganography to this application.

Another use of steganographic techniques was for the protection of biometric data [8]. Biometrics are used widely in institutions now for verification and security purposes. They work on the basics of matched fingerprints of the users and hence are practical to use. But this biometric data once hacked, can cause havoc to organisations and its people as once altered, it cannot be changed again. This vulnerability causes a need for its security. This paper presented ways to tackle this problem and provide solutions. One was using watermarks to make the fingerprints and iris scans more authentic. Additional information bits are added to the cover image in watermarking whereas steganography hides information bits in the cover image itself. The spatial domain and transform domain methods were analyzed and their pros and cons were compared. The transform domain technique is more used than spatial domain due to its strength against attacks. Another recent trend is to use Singular Value Deposition technique along with DCT and DWT. Both blind and targeted steganalysis methods were described on the stego object.

In 2002, Fridrich [9], presented a new approach to deal with steganography in JPEG lossy compression images. It cropped the stego image by a certain amount of pixels, re-compressed and quantized it to calculate new DCT coefficients. The DCT histograms were then analyzed and the deviations obtained were solved as discontinuities in basic LSB operations. An Outguess steganographic algorithm was proposed to counter major steganalysis attacks. They calculated the changes in 70 test JPEG images obtained using a digital Kodak DC camera. The success rate was 94%.

Generally, steganalysis is classified into two phases- the first phase is separating and analyzing the distinctive features and properties of image. The second is the classification phase that uses classifiers such as SVM (Support Vector Machines) and Ensemblers. In recent times, deep learning techniques have made it possible to merge these two phases and carry them out parallel to obtain positive results. The paper showed that development of Deep Learning techniques overcame the use of classical methods that are Rich Models with Ensemble Classifiers - both in the spatial and frequency (JPEG) domains. Within the past decade, researchers have used CNN (Convolutional Neural Networks) to detect the presence of information in carrier image. WOW, HUGO, S-UNIVARD, J-UNIVARD are some of the traditional strategies used in the evolution of steganalysis [10].

However, the previous works are still prone to steganalysis attacks with now available advanced steganography softwares. So, trained encoders are used now that provide maximum similarity index between the stego image and the initial input message.

3. PROPOSED MODEL

The proposed system is divided into 2 main modules-
1. Image Steganography
2. Text Steganography

3.1 Image in image steganography

A popular cover object for steganography is images. A message is embedded during a digital image through an embedding algorithm, using the key. The resulting stego

image is sent to the receiver. On the opposite side, it's processed by the extraction algorithm using an equivalent key. During the transmission of encrypted images even if anyone gets their hands on the image, they will not be aware of the hidden message and can only see the cover picture in transmission.

In digital image steganography, the main focus of this proposed work will be on the following methods-
1. Using LSB watermarking encoding
2. Masking and Filtering
3. Empirical Evaluation of the model using Deep Learning

3.1.1 LSB encoding

The LSB encoding works on the principle of weakness of human eyes to distinguish between the colors with very low distinctions. In LSB steganography method the least significant bit of the pixel is modified to accommodate the message at its place. The trick is to not overdo the effect other human eyes are very powerful at noticing abnormalities. Each color bit in the pixel values contains 8 bits, so the method takes the least significant bit from three primary colors RGB and modifies that. Hence, a single pixel can hold up to 3 bits of information with very low changes in the image.

The message file should be shorter than the container file to get the best results; otherwise it could distort the image.

There are some steps:
- Security through obscurity.
- Hiding a message so that nobody is aware of its actuality.
- Most commonly used to hide text or an image within a cover image.
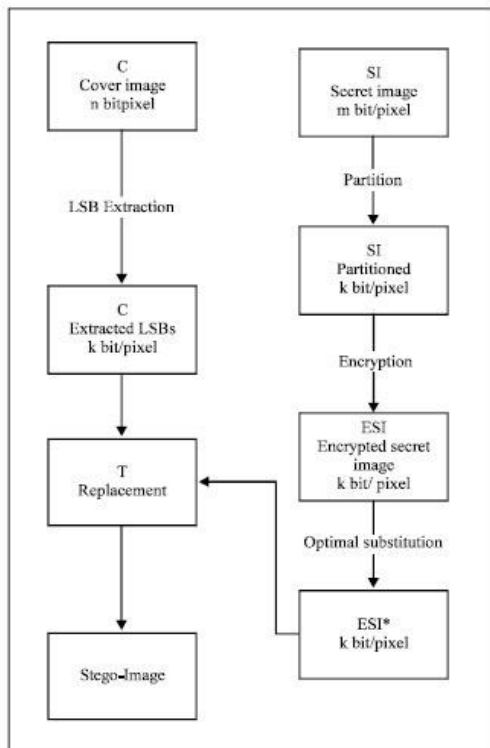
Figure 1. Steps involved in LSB encoding

Most of the images captured, transferred and used over the internet have more details in them than a human eye can see. There are about 5 million various colors possible that can be obtained by changing the least significant bits of an image pixel that human eyes would not be able to recognize the difference. These are things an LSB encoding algorithm takes advantage of and modifies those bits to encode the secret message in between [11, 12].

The LSB encoding sometimes also considers the right most bit as the least significant bit because the positional writing notation of languages like Hindi and Arabic. The LSB encoding is a very simple technique hence even though the changes are invisible to the human eye; a simple computer system can decrypt the secret message with the help of some simple steganalysis algorithms or softwares.

This technique is effective if malicious people in between are not aware of the fact that there's secret messages being transmitted in the images. Otherwise, attackers can easily get the secret message by doing simple calculations from the beginning of the image. An improved system called Video Steganography was introduced to overcome these shortcomings. The secret message is embedded into a random set of pixels scattered all over the cover image [13].

The secret message is converted into the bits and then those bits are placed at the least significant bit (also known as, the 8th bit) of all or some random pixels of the cover image to get the maximum capacity and performance of information hiding, as a result LSB method is the better performing information hiding method than some other options of embedding messages [14-16].

3.1.2 Masking and filtering

This kind of processing can be done on both RGB and gray scale images. It is mainly used to insert watermarks on original images so that no one can illegally copy or use that content without credits. It changes the visual properties of the image by altering the opacity and luminance. This is not a concealment technique but it extends the cover image further by adding information such as copyright information, digital signature, certificates etc. [17]. It is more powerful than LSB encoding in terms of image processing as it can also be done with lossy compression such as in JPEG images. Two images are simply overlapped over each other and grouped together as a single file.

3.1.3 Deep learning using Adam

The two networks (Hiding network and the Reveal network) were trained using Adam [18-20]. Unlike a usual three network structure: Prep-Network, Hiding-Network and Reveal-network. This paper suggests a two network structure otherwise: Hiding-network and Reveal-network, getting rid of the prep-network.

The Hiding network takes on the responsibility of getting the cover image and secret image to the compatible sizes. Secret images either be smaller or the same size than the cover image or the image to the size of the cover, the smaller secret image is stretched across the N x N pixels of the cover image. The hiding network then embeds the secret image into the cover image, creating the Container image. A 6-channel tensor is obtained by concatenating cover image and secret image which is then passed as an input to the Hiding network. There are different utilities for managing image normalization, cropping, adjusting brightness and hue etc. operations on images. This network is adapted from pytorch vision transform.py utility.

The Reveal network's task is to extract secret images from the container images. It doesn't have any access to the secret

image or the Cover image. The whole network operates only on container image without any access to secret or cover image and reveals the secret image as an output. There's a very little distinction between the original secret image and the revealed secret image, unnoticeable to naked human eye.
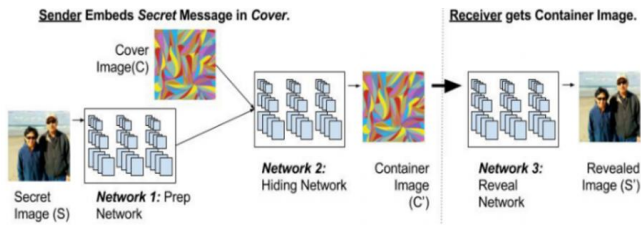
Figure 2. Flowchart of proposed network model

Adam [21-23] is an adaptive learning rate optimization algorithm that's been designed specifically for training deep neural networks. It is an adaptive learning rate method, which means, it computes individual learning rates for different parameters. For simplicity, the reconstructions minimized the sum of squares error of the pixel difference, although other image metrics could have easily been substituted. The networks were trained using randomly selected pairs of images from the ImageNet training set [24].

This paper proposes a deep learning method to achieve Image Steganography. The model is trained against a dataset using Pytorch implementation. The training process is time and power consuming and takes about 3 to 4 days of computation on a 4 GB graphics card. The plus point of this implementation is that the training process can be paused for some time and can be resumed later as the calculations are stored in the form of checkpoints which can be merged together after every calculation is done.

The framework is trained using two images in every calculation: the cover image (image on which the secret image will be superimposed) and the secret image (the image that will be hidden inside the cover image).

The hiding network (H-net) encodes the secret image inside the cover image in a way making the secret image invisible. The hiding network (H-net) outputs a container image. Then that container image is transferred to a reveal network (R-net) whose task is to decode the container image and drag the secret image out from the container image separating the cover and secret image. The output of R-net is called a revealed secret image and can simply be decoded to get the initial secret image.

The complete process works in two parts:

1. Hiding Network: To embed the secret image in a cover image the project proposes a U-net structured convolution network. A tensor concatenation of Cover image and secret image is provided as input to the hiding network. A U-net structured network is proposed because of its specialty in processing images. The network keeps the input and output image of the same size, on which the project paper proposes capitalization. The secret image is stretched to the same size as the cover image, which increases the hiding capabilities since the entire cover image is available to hide the secret image information with minimal distortion to the cover image.

2. Reveal Network: The reveal network is trained simultaneously with the hiding network and it takes the container image produced by Hiding Network directly. The paper proposes a 6 convolution layer network with 3×3 kernel size where every layer is followed by a batch normalization and ReLU (Rectifier Linear Unit) except the last layer for this

network. Some scholars are still debating over whether to use ReLU after or before the batch normalization. But this project proposes to use batch normalization before ReLU. Batch normalization standardizes the inputs to the network and helps in accelerating the training process by having epochs and by providing some secularization, and by reducing generalization error. Rectifier Linear Unit (ReLU) is used as an activation function to train this layered network which simplifies the model training computation requirements by overcoming the vanishing gradient problem.

3.2 Text in image steganography

In text in image steganography, the format and encoding of the text is altered to make it hide more efficiently. The task can be achieved by various methods some of which are namely, Random generation method, Linguistic and format based method. Different coding techniques are implemented in this section-

3.2.1 Line shift coding
This technique is used to modify the text by shifting the lines vertically. The shifting of lines led to a specific pattern in the text which is used to generate a cover text. Shifting is incorporated by moving the lines of the plain text by some degree in the vertical direction. Bits like 0, 1 and -1 can be used to denote the unmoved, shifted up and shifted down lines. At the decoders end either baseline shifting is found or the centroid shifting is found. Baseline shifting includes the hidden message in the baseline of the adjacent lines in text. If centroid shifting is done, then the decoder searches for the hidden message in the centroids of the adjacent lines of the cover text. This technique has problems that if in the transmission of the cover text the lines are shifted or if OCR techniques are applied the hidden information may get destroyed.

3.2.2 Word shift coding
The method introduces the idea of shifting the words of the text in a horizontal manner to hide a meaningful message. The distance in between the words is altered so that information is hidden. The spaces within the words should be different in order to generate a cover text. The decoder should also know about how to shift the words to extract a message hidden using steganography. If there are variable places involving the word shifting method then the decoder should be given the original text so that detection of correct information at the decoder's end is possible. This method has similar limitations as line shifting. If retyping of the cover text is done or if the OCR techniques are used then the message will be destroyed and this method will fail.

3.2.3 Feature encoding
The feature encoding deals with the change in the features of the text in such a manner that a meaningful message is hidden to produce a cover text. Features like height of the text, color of the text, font of the text are some of the ways which are used. A large volume of information is hidden using feature encoding. When the features of the text are altered then only the sender and the intended recipient detects the hidden message. The third party does not catch the attention of something concealed inside a text. OCR techniques and retyping are responsible for changing the text features and damaging the message.

4. RESULTS AND DISCUSSION

4.1 Image in image steganography

The image in image steganography is achieved by Deep learning tools, hence providing better results than using the simple techniques. The container image looks similar to the cover image and the size of the image is not changed to susceptible levels. There is a deviation between the revealed secret image and the original secret image due to all the operations performed on it, but it is not noticeable to the naked eye.

(a) Cover image

(b) Container image

Figure 3. Tiny differences between cover and container images

Figure 3(b) depicts the final image containing the stego image object in it. It can be observed that this image is visually similar to the one shown in the original image. After various statistical methods and steganalysis, our desired secret image can be retrieved from it.
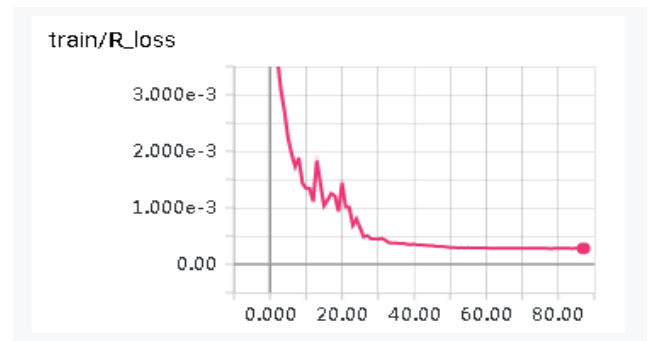

(a) Original secret image

(b) Revealed secret image

Figure 4. Tiny differences between original secret image and revealed secret image

Tiny differences not noticeable to the human eye are present in figure 4 above. The secret message (image) remains almost unaffected on transmission.

4.1.1 Hiding network and reveal network loss curves

The algorithm used two networks i.e. hiding network and Reveal network which were trained with a hyper-parameter with an empirical value 0.75 to balance the visual performance of cover images and revealed secret. The loss curves are shown below.

Figure 5. MSE Loss on Secret Image and Revealed Image (Reveal Network)
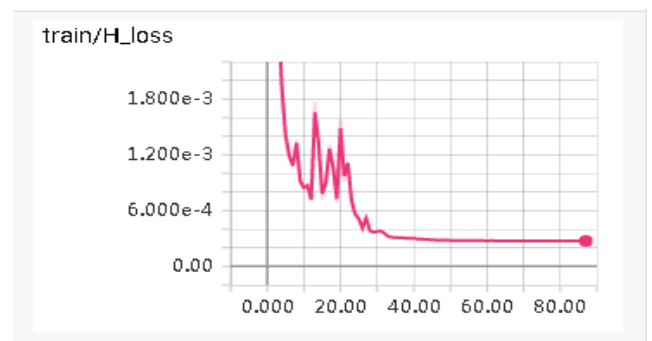
Figure 6. MSE Loss on Cover Image and Container Image (Hiding Network)

4.1.2 Average pixel wise difference

Table 1 Column 1 depicts the average discrepancy between the original pixel and the modified pixel of container image and cover image during Training and testing sessions. Table 1

Column 2 depicts the average pixel wise discrepancy between the original image i.e. the secret image pixels and the fetched image i.e. the revealed secret image. This is calculated by subtracting the RGB values (in 0-255 range) of every pixel in the first image from consecutive RGB values of the second image. Take the modulus of that difference of every value and sum it all and then divide it by the total number of pixels in the image (The images compared are of the same size hence, the total number of pixels value will be the same for both images.)

Table 1. Average pixel wise difference

Dataset	Container - Cover	Secret - Revealed
Training	4.30	4.68
Validation	4.25	4.45

As depicted above, the average pixel difference is about 4 bits change for every pixel, which will be unnoticeable to naked human eyes for most part with some exceptions sometime. A computer system can detect the changes but it needs a lot of computational power to extract some parts of the secret image from the container image. Even with a lot of computational power without the reveal network no system will be able to extract all the details of a secret image as it is from the container image.

4.2 Text in image steganography

A usual sized image has enough bits to go around hiding a few lines of text in its bits easily, without showing any kind of noticeable changes to its size, structure or color pallet.

(a) Secret Text : Steganography project is working very well!

(b) Revealedt Text : Steganography project is working very well!

Figure 7. Original and final vessel images obtained in text steganography

Figure 7(a) is the original container image without any message. We later inserted a text message in it "Steganography project is working very well!" and the output resultant image obtained is visually the same as shown in 7(b). Statistical analysis like observing the increased size of the image may reveal the presence of a message in it.

Steganalysis can be done to decrypt this text message again securely.

5. CONCLUSION

This work demonstrates image steganography using deep learning. Along with LSB encoding, Adam's algorithm was used to implement it on various input images and text messages. In the first part, a hiding network was used to embed the secret image media into a cover image file. Then, the reveal network at the receiving end undoes this process and thus safe transfer of secret message is achieved. Adam's algorithm was used to train the model. In the second part, a text message is overlapped onto the image media with non-noticeable changes visible to the human eye. Steganalysis has to be done at the other end to retrieve the secret text message.

Other steganography softwares cannot be used to decode and reveal the secret message from the cover media. Thus, making this application more secure. The network is trained once only, irrespective of the different cover images and secret messages given as inputs. This process has wide applications in many fields such as hiding sensitive military information, using watermarking techniques to hide identity, in fingerprinting and biometrics for improved security.

REFERENCES

[1] Satish, K., Jayakar, T., Tobin, C., Madhavi, K., Murali, K. (2004). Chaos based spread spectrum image steganography. IEEE Transactions on Consumer Electronics, 50(2): 587-590. https://doi.org/10.1109/TCE.2004.1309431

[2] Bhadra, J., Bojamma, A.M., Prasad, C.N., Nachappa, M.N. (2014). An insight to steganography. International Journal of Innovative Science, Engineering & Technology, 1(10): 29-42.

[3] Dunbar, B. (2002). A detailed look at Steganographic Techniques and their use in an Open-Systems Environment. SANS Institute, 1.

[4] Tomar, G. (2012). Effect of noise on image steganography based on LSB insertion and RSA encryption. IOSR Journal of Engineering, 2(3): 473-477. https://doi.org/10.9790/3021-0203473477

[5] Chaumont, M. (2020). Deep learning in steganography and steganalysis. In Digital Media Steganography, pp. 321-349. https://doi.org/10.1016/B978-0-12-819438-6.00022-0

[6] Jung, K.H. (2019). A study on machine learning for steganalysis. In Proceedings of the 3rd International Conference on Machine Learning and Soft Computing, pp. 12-15. https://doi.org/10.1145/3310986.3311000

[7] Karaman, H.B., Sagiroglu, S. (2012). An application based on steganography. In 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 839-843. https://doi.org/10.1109/ASONAM.2012.152

[8] Douglas, M., Bailey, K., Leeney, M., Curran, K. (2018). An overview of steganography techniques applied to the protection of biometric data. Multimedia Tools and

Applications, 77(13): 17333-17373. https://doi.org/10.1007/s11042-017-5308-3

[9] Fridrich, J., Goljan, M., Hogea, D. (2002). Attacking the outguess. In Proceedings of the ACM Workshop on Multimedia and Security.

[10] Reinel, T.S., Raul, R.P., Gustavo, I. (2019). Deep learning applied to steganalysis of digital images: A systematic review. IEEE Access, 7: 68970-68990. https://doi.org/10.1109/ACCESS.2019.2918086

[11] Das, S., Das, S., Bandyopadhyay, B., Sanyal, S. (2011). Steganography and Steganalysis: different approaches. arXiv preprint arXiv:1111.3758.

[12] Provos, N., Honeyman, P. (2003). Hide and seek: An introduction to steganography. IEEE Security & Privacy, 1(3): 32-44. https://doi.org/10.1109/MSECP.2003.1203220

[13] Wang, H., Wang, S. (2004). Cyber warfare: steganography vs. steganalysis. Communications of the ACM, 47(10): 76-82. https://doi.org/10.1145/1022594.1022597

[14] Bairagi, A.K., Mondal, S., Debnath, R. (2014). A robust RGB channel based image steganography technique using a secret key. In 16th Int'l Conf. Computer and Information Technology, pp. 81-87. https://doi.org/10.1109/ICCITechn.2014.6997309

[15] Ray, R., Sanyal, J., Das, D., Nath, A. (2012). A new Challenge of hiding any encrypted secret message inside any Text/ASCII file or in MS word file: RJDA Algorithm. In 2012 International Conference on Communication Systems and Network Technologies, pp. 889-893. https://doi.org/10.1109/CSNT.2012.191

[16] Singh, S., Datar, A. (2015). Improved hash based approach for secure color image steganography using canny edge detection method. International Journal of Computer Science and Network Security (IJCSNS), 15(7): 92.

[17] Saraswat, P.K., Gupta, R.K. (2012). A review of digital image steganography. Journal of Pure and Applied Science & Technology, 2(1): 98-106.

[18] Dozat, T. (2016). Incorporating nesterov momentum into adam.

[19] Kingma, D.P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[20] Reddi, S.J., Kale, S., Kumar, S. (2019). On the convergence of adam and beyond. arXiv preprint arXiv:1904.09237.

[21] Al Faruque, M.A., Krist, R., Henkel, J. (2008). Adam: run-time agent-based distributed application mapping for on-chip communication. In 2008 45th ACM/IEEE Design Automation Conference, pp. 760-765. https://doi.org/10.1145/1391469.1391664

[22] Fitzpatrick, A.P., Gonzales, R.P., Lesh, M.D., odin, G.W., Lee, R.J., Scheinman, M.M. (1994). New algorithm for the localization of accessory atrioventricular connections using a baseline electrocardiogram. Journal of the American College of Cardiology, 23(1): 107-116. https://doi.org/10.1016/0735-1097(94)90508-8

[23] Szilagyi, L., Benyo, Z., Szilágyi, S.M., Adam, H.S. (2003). MR brain image segmentation using an enhanced fuzzy c-means algorithm. In Proceedings of the 25th Annual International Conference of the IEEE Engineering in medicine and Biology Society (IEEE Cat. No. 03CH37439), 1: 724-726. https://doi.org/10.1109/IEMBS.2003.1279866

[24] Baluja, S. (2017). Hiding images in plain sight: Deep steganography. In Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 2066-2076.