# Removal of Image Steganography using Generative Adversarial Network

**Ritvij Vejare, Abhishek Vaish, Kapish Singh, Mrunali Desai**

***Abstract*: Secret messages can be concealed in ordinary media like audio, video and images. This is called as Steganography. Steganography is used by cyber attackers to send malicious content that could harm victims. Digital steganography, or steganography in photographs, is exceedingly difficult to detect. The detection of steganography in images, has been investigated in thoroughly by a variety of parties. The use of steganographic techniques to send more malware to a compromised host in order to undertake different post-exploitation operations that affect the exploited system. Many steganalysis algorithms, on the other hand, are limited to working with a subset of all potential photos in the wild or have a high false positive rate. As a result, barring any suspected image becomes an arbitrary policy. Filtering questionable photos before they are received by the host machine is a more practical policy. In this paper, a Generative Adversarial Network based model is proposed that may be optimized to delete steganographic content while maintaining the original image's perceptual quality. For removing steganography from photos while keeping the maximum visual image quality, a model is built utilizing a combination of Generative Adversarial Network (GAN) and Image Processing. In the future, utilizing a generator to synthesize a picture will become more popular, and detection of steganography in images will become very difficult. In comparison to other models that have been addressed further, the proposed model is able to give a mean square error of 5.4204 between the generated image and the cover image, as well as better outcomes based on several metrics. As a result, a GAN-based steganography eradication method will aid in this endeavor.***

***Keywords: Removal of steganography, steganalysis, Generative Adversarial Network, Neural Network, Deep Learning***

## I. INTRODUCTION

Steganography is described as the use of various algorithms or approaches to hide data within an entirely separate collection of data (such as photos) so that only the source entity and destination entity are aware of the hidden content during file transfer [1]. The content to be hidden by steganography is frequently encrypted before embedding it into the data. If the hidden data is not encrypted, it is usually treated in some way to make it more difficult to find it. However, hackers use this technique for malpractices like sending malware hidden in images which get activated by the victim unknowing when they open the image. Steganography has attracted various organizations due to the secrecy involved and the achievement of the goals along with it [4]. After contents have been embedded into another data using steganography, various methods are also used to detect the steganographic content present which is basically termed as steganalysis. If there is any unwanted steganographic content, that content has to be destroyed which is possible by different methods which are used for destruction of steganographic content [14]. Steganographic image is the output image after adding hidden information in an original image or cover image using steganography. The remainder of the paper is laid out as follows. The prior work in picture steganography and GAN will be discussed in Section II. In Section III, the dataset used to train the model will be presented, followed by Methodology in Section IV. Section V will contain the experimental outcomes. In Section VI, the conclusion and future work will be given.

## II. BACKGROUND

Various methods or techniques have been studied and implemented to attack steganographic systems. These approaches have either fully removed the steganographic content from the image or minimally changed the steganographic content, rendering it unusable while maintaining the image's quality. Machine learning [9] and non-machine [4] learning techniques are among the various steganography eradication strategies.

Various digital filters and wavelet transforms are used in some of the non-machine learning techniques.[4] Overwriting method is a method in which the least significant bit of the image pixel is changed. Denoising approach is a method in which the corrupted pixel is selected and is replaced with a predicted value which removes the steganographic content. Denoising approach is of two types: filtering technique and discrete wavelet technique. These techniques are simple to use and do not require training on a dataset like their machine learning counterparts. However, they do not erase the artifacts and patterns left by steganographic algorithms. These strategies try to filter out content with a high frequency, resulting in image quality reduction due to a lower focus placed on perceptual quality. [4] The Pixel Steganalysis approach, which is based on an architecture called Pixel CNN++ [22], is a machine learning technique.

**Manuscript received on 10 May 2022 | Revised Manuscript received on 24 May 2022 | Manuscript Accepted on 15 June 2022 | Manuscript published on 30 June 2022.**

***** Correspondence Author

**Ritvij Vejare*,** Department of Computer Engineering, K.J. Somaiya Institute of Engineering and Information Technology, Mumbai (Maharashtra), India.

**Abhishek Vaish,** Department of Computer Engineering, K.J. Somaiya Institute of Engineering and Information Technology, Mumbai (Maharashtra), India.

**Kapish Singh,** Department of Computer Engineering, K.J. Somaiya Institute of Engineering and Information Technology, Mumbai (Maharashtra), India.

**Mrunali Desai,** Assistant Professor, Department of Computer Engineering, K.J. Somaiya Institute of Engineering and Information Technology, Mumbai (Maharashtra), India.

In this procedure, pixel and edge distributions are created for the image, which are then deleted from the suspected image. This method has been compared to three steganographic algorithms: Deep Neural Network (DNN)[24] based algorithm, which is a technique in which artificial intelligence (AI) i.e. neural networks is used to embed steganographic content, Invisible Steganography GAN(ISGAN)[23], which is a novel CNN architecture named that conceals a secret grey image into a color cover image, and Least Significant Bit(LSB) algorithm, which is a technique in which least significant bit of the image is replaced with data bit. GANs for removing steganographic material while keeping excellent visual quality are based on GANs for single image super resolution (SISR). The GAN framework was used in Ledig et al s research [20] to optimize a ResNet to raise the resolution of low-resolution photos to make them as aesthetically comparable to their high-resolution counterparts as feasible.

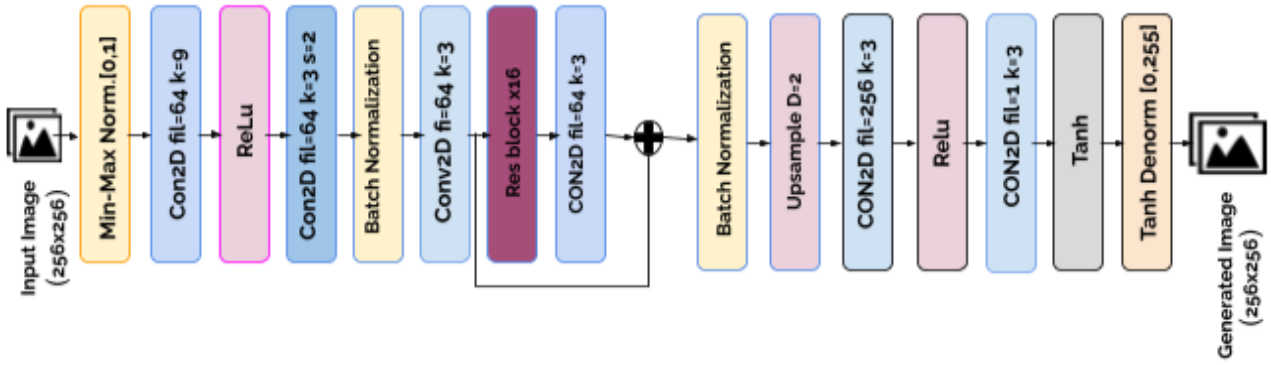


**Fig. 2a: Generator Architecture**

Additionally, using the GAN framework rather than a pixel-wise distance loss function, such as mean squared error (MSE), resulted in more high frequency texture detail being restored, according to their research.

## III. DATA

For the project, we have taken the BossBASE dataset [21]. The BossBASE dataset contains 10000 grayscale images of size 256x256. These 10000 images are divided into 5 groups of 2000 images each. On this dataset, three different Steganographic algorithms named SUniward[26], Hugo[25] and WOW[27]. The three steganographic contents are then applied on the divided dataset with different embedding rates of 10%, 20%, 30%, 40% and 50% applied on each group of 2000 images. Here 10% is the easiest to crack and 50% is the most difficult to crack. The final dataset contains about 30000 images containing steganography and 10000 images without steganography.

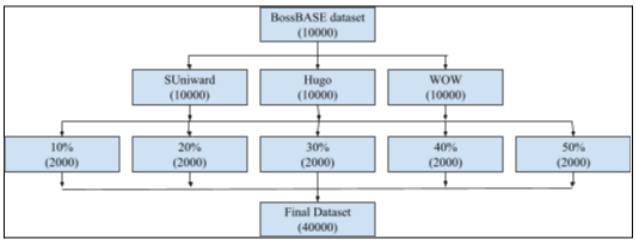


**Fig. 1: Dataset Structure**

## IV. METHODOLOGY

### A. Model

For picture to image translation between the Stego Image and the Cover Image, a model based on generative adversarial networks is proposed. The model design is similar to that of the models used in SRGAN [20], but with fewer parameters. SRGAN is a super resolution GAN that is used to expand images.

SRGAN is a single picture super-resolution generative adversarial network. There are two pieces to it: a generator and a discriminator. The generator generates data using a probability distribution, and the discriminator attempts to determine whether the data comes from the dataset or the generator. The goal of the SRGAN architecture is to recover finer features from an image when it is upscaled, ensuring that the image's quality is not compromised.

### B. Generator Architecture

The generator takes in a 256 x 256 grey scale image as an input and produces a 256 x 256 grey scale image as an output. The input image is first passed through a minmax Normalization layer that scales the input image between 0 and 1.Then it is passed through a series of convolutional, Relu, Batch normalization layer and then it is passed through 16 residual block after which the image gets downsized by a factor of 2 so we have a 128 x 128 output which we then up sample by a factor of 2 to get to the original dimension which is then passed to the tanh activation layer which scales the output between -1 and 1.Finally the tanh denormalization layer which is then scaled to 0 and 255 for the output image

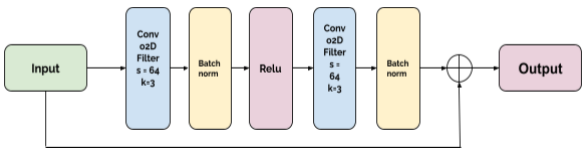


**Fig. 2b: Residual Block Architecture used in fig 2a**

*Retrieval Number: 100.1/ijainn.D1054062422*
*DOI:10.54105/ijainn.D1054.062422*
*Journal Website: www.ijainn.latticescipub.com*



*Published By:*
*Lattice Science Publication (LSP)*

### C. Discriminator Architecture

The discriminator accepts a 256 x 256 grayscale image as input and returns a single value between 0 and 1, indicating whether the image is authentic or bogus.

The term "genuine picture" refers to an image that is part of a series of cover images, while "fake image" refers to an image that is part of a group of generated images. The input image is initially scaled between -1 and 1 using a max. abs. normalisation layer. After that, a sequence of Convolutional, Batch Normalization, and Relu Layers are applied. There are seven different sorts of blocks, each with a different number of filters and strides, before being flattened and connected to a dense layer, then a sigmoid layer for output between 0 and 1.

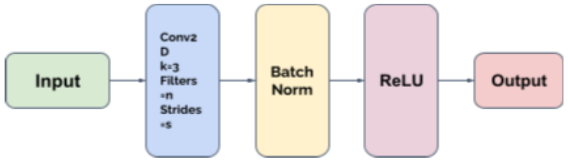


**Fig. 3b: Discriminator Block Architecture used in fig 3a**

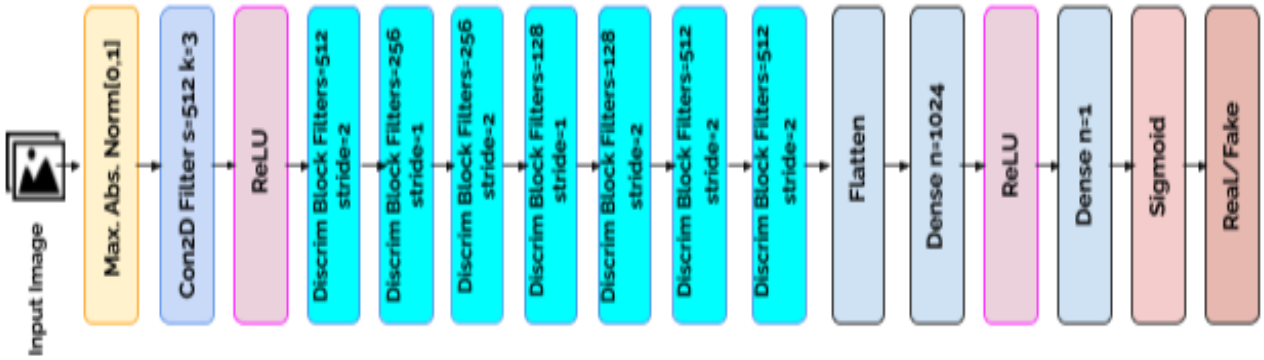


**Fig. 3a: Discriminator Architecture**

## V. TRAINING

First pretrain the generator on the mean square error between the input image and the output image; this is done so that the generator converges quickly while adversarial training. After pre- training the generator Start the adversarial training. In this take stego image (S) and the corresponding cover image (C) then pass the Stego Image(S) in the generator (Gen), and then pass the output of Generator Gen(S) and C to the discriminator (Disc) which tries to distinguish between them. For the generator, the ADAM optimizer is employed with a learning rate of $\alpha$=10-3 $\beta$1=0.5 and $\beta$2=0.9. The loss function for generator is as follows

$$G = Gen(S) \quad (1)$$

Where G is the purified Image that is generated from the generator by passing stego Image (S)

$$DG = Disc(G) \quad (2)$$

Where $D_G$ is the discriminator output for Generated Images(G)

$$Dc = Disc(C) \quad (3)$$

Where Dc is the discriminator output for Cover Images (C)

$$Gen\_Loss = MSE(C, G) + BCE(1, DG) \quad (4)$$

Gen_loss is calculated as the sum of mean square error between the cover images(C) and generated images (G) and the Binary Cross Entropy between 1 and Discriminator output for generated images

$$Disc\_Loss = BCE(0, DG) + BCE(1, DC) \quad (5)$$

Disc_loss is calculated as the sum of Binary Cross Entropy between 0 and Discriminator output for generated images and the Binary Cross Entropy between 1 and Discriminator output for cover images As the adversarial training continues the generator becomes more and more efficient in removing the stego content and discriminator becomes more and more accurate in distinguishing between the generated/purified image and the original cover image.

## VI. RESULT

To compare the results of our model we have calculated over 5000 images and to compare the quality between the generated image and the corresponding cover image. Mean Square Error (MSE) is used, Structural Similarity Index (SSIM), Spectral Angle Mapper (SAM), Universal Quality Index (UQI), Peak Signal to Noise Ratio (PSNR), Visual Information Fidelity (VIFP). Different models have been compared like Auto Encoder, Bilinear Interpolation, Lanczos3, Bicubic Interpolation, Gaussian, Nearest Neighbor Interpolation. It can be observed that the proposed model provides good results for the different metrics used.

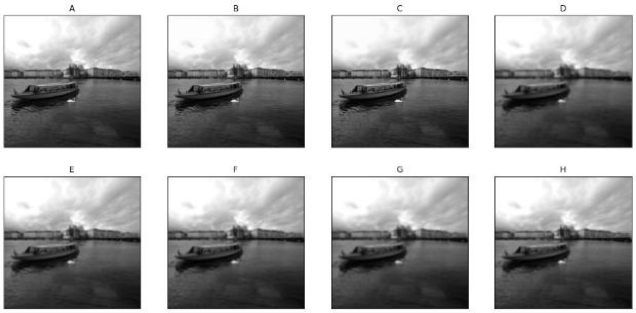


**Fig.6: A:cover images, B:generated images, C:autoencoder images, D:bilinear images, E:lanczos3 images, F:bicubic images, G:gaussian images, H:nearest neighbor images**

**Table 1: Comparison of various model**

| Model | MSE | SSIM | PSNR | UQI | SAM | VIFP |
|---|---|---|---|---|---|---|
| Proposed Model | 5.4204 | 0.9897 | 43.294 | 0.9895 | 0.0265 | 0.8553 |
| Auto Encoder | 34.1621 | 0.9435 | 34.529 | 0.0605 | 0.0691 | 0.6487 |
| Bilinear In. | 95.8807 | 0.8526 | 30.734 | 0.0804 | 0.1169 | 0.4660 |
| Lanczos 3 | 74.3341 | 0.8806 | 31.246 | 0.9841 | 0.1022 | 0.5204 |
| Bicubic In. | 80.3011 | 0.8730 | 31.145 | 0.9832 | 0.1065 | 0.5045 |
| Gaussian | 102.560 | 0.8432 | 30.409 | 0.9790 | 0.1211 | 0.4511 |
| NN In. | 94.7576 | 0.8672 | 30.593 | 0.9814 | 0.1174 | 0.4566 |

## VII. CONCLUSION

In this Paper, a Generative Adversarial Networks (GANs) and Image Processing based model for removing steganography It is proposed to create photos with the highest visual image quality possible. The proposed model is able to provide a mean square error of 5.4204 between the generated image and cover image and better results based on different metrics as compared to other models which have been discussed. Although there are many different methods to remove steganography from an image, most of them are incapable of removing advanced steganography techniques Synthetic images can now be created easily using technologies like GAN with which steganography can be added without distorting much content of the image so traditional steganalysis method may not work in all cases therefore using a Neural Network model to filter images can provide a general method for removing steganography from images.

## REFERENCES


1. Fridrich, Jessica. *Steganography in digital media: principles, algorithms, and applications*. Cambridge University Press, 2009. [CrossRef]
2. Marvel, Lisa M., Charles T. Retter, and Charles G. Boncelet Jr. "Hiding Information in Images." *ICIP (2)*. 1998.
3. Hamid, Nagham, et al. "Image steganography techniques: an overview." *International Journal of Computer Science and Security (IJCSS)* 6.3 (2012): 168-187.
4. Ameen, Siddeeq Y., and Muthana R. Al-Badrany. "Optimal image steganography content destruction techniques." *International Conference on Systems, Control, Signal Processing and Informatics*. 2013.
5. Shrestha, Pradhumna Lal, et al. "A general attack method for steganography removal using pseudo-cfa re-interpolation." *2011 International Conference for Internet Technology and Secured Transactions*. IEEE, 2011.
6. Quan, Xiaomei. "JPEG Steganalysis Based on Local Dimension Estimation." *2021 6th International Conference on Image, Vision and Computing (ICIVC)*. IEEE, 2021. [CrossRef]
7. Amritha, P. P., M. Sethumadhavan, and R. Krishnan. "On the Removal of Steganographic Content from Images." *Defense Science Journal* 66.6 (2016). [CrossRef]
8. Zou, Ying, Ge Zhang, and Leian Liu. "Research on image steganography analysis based on deep learning." *Journal of Visual Communication and Image Representation* 60 (2019): 266-275. [CrossRef]
9. Ke, Qi, Liu Dong Ming, and Zhang Daxing. "Image steganalysis via multi-column convolutional neural network." *2018 14th IEEE International Conference on signal processing (ICSP)*. IEEE, 2018. [CrossRef]
10. Tang, Yong-He, et al. "A review on deep learning-based image steganalysis." *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. IEEE, 2018. [CrossRef]
11. Qian, Yinlong, et al. "Deep learning for steganalysis via convolutional neural networks." *Media Watermarking, Security, and Forensics 2015*. Vol. 9409. SPIE, 2015. [CrossRef]
12. Xu, Guanshuo, Han-Zhou Wu, and Yun-Qing Shi. "Structural design of convolutional neural networks for steganalysis." *IEEE Signal Processing Letters* 23.5 (2016): 708-712. [CrossRef]
13. Aggarwal, Alankrita, Mamta Mittal, and Gopi Battineni. "Generative adversarial network: An overview of theory and applications." *International Journal of Information Management Data Insights* 1.1 (2021): 100004. [CrossRef]
14. Liu, Jia, et al. "Recent advances of image steganography with generative adversarial networks." *IEEE Access* 8 (2020): 60575-60597. [CrossRef]
15. Zhang, Zhuo, et al. "A generative method for steganography by cover synthesis with auxiliary semantics." *Tsinghua Science and Technology* 25.4 (2020): 516-527. [CrossRef]
16. Naito, Hiroshi, and Qiangfu Zhao. "A new steganography method based on generative adversarial networks." *2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST)*. IEEE, 2019. [CrossRef]
17. Zhou, Lingchen, et al. "On security enhancement of steganography via generative adversarial image." *IEEE Signal Processing Letters* 27 (2019): 166-170. [CrossRef]
18. Corley, Isaac, Jonathan Lwowski, and Justin Hoffman. "Destruction of image steganography using generative adversarial networks." *arXiv preprint arXiv:1912.10070* (2019).
19. Wang, Huaqi, et al. "Defeating data hiding in social networks using generative adversarial networks." *EURASIP Journal on Image and Video Processing* 2020.1 (2020): 1-13. [CrossRef]
20. Ledig, Christian, et al. "Photo-realistic single image super-resolution using a generative adversarial network." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017. [CrossRef]
21. Bas, Patrick, Tomáš Filler, and Tomáš Pevný."" Break our steganographic system": the ins and outs of organizing BOSS." *International workshop on information hiding*. Springer, Berlin, Heidelberg, 2011. [CrossRef]
22. Jung, Dahuin, et al. "Pixelsteganalysis: Destroying hidden information with a low degree of visual degradation." *arXiv preprint arXiv:1902.11113* (2019).
23. Zhang, Ru, Shiqi Dong, and Jianyi Liu. "Invisible steganography via generative adversarial networks." *Multimedia tools and applications* 78.7 (2019): 8559-8575. [CrossRef]
24. Baluja, Shumeet. "Hiding images in plain sight: Deep steganography." *Advances in neural information processing systems* 30 (2017).
25. Pevný, Tomáš, Tomáš Filler, and Patrick Bas. "Using high-dimensional image models to perform highly undetectable steganography." International Workshop on Information Hiding. Springer, Berlin, Heidelberg, 2010. [CrossRef]
26. Holub, Vojtěch, Jessica Fridrich, and Tomáš Denmark. "Universal distortion function for steganography in an arbitrary domain." EURASIP Journal on Information Security 2014.1 (2014): 1-13. [CrossRef]
27. Holub, Vojtěch, and Jessica Fridrich. "Designing steganographic distortion using directional filters." 2012 IEEE International workshop on information forensics and security (WIFS). IEEE, 2012. [CrossRef]


## AUTHORS PROFILE

**Ritvij Vejare,** Pursing Bachelor's degree In Computer Engineering at K.J. Somaiya Institute of Engineering and Information Technology. A highly driven individual, seeking the role of a Software Engineer where I can contribute towards the organizational goals through my analytical skills and knowledge acquired by pursuing a degree in Software Engineering. Proficient in C/C++, Python, JavaScript, HTML, CSS and ReactJS.

*Retrieval Number: 100.1/ijainn.D1054062422*
*DOI:10.54105/ijainn.D1054.062422*
*Journal Website: www.ijainn.latticescipub.com*



*Published By:*
*Lattice Science Publication (LSP)*

**Abhishek Vaish**, Pursing Bachelor's degree In Computer Engineering at K.J. Somaiya Institute of Engineering and Information Technology A motivated individual with a keen interest of working with new technologies and constantly try to learn new things and apply that knowledge to gain experience to become a successful expert in the field of Information Technology

**Kapish Singh** Pursing Bachelor's degree In Computer Engineering at K.J. Somaiya Institute of Engineering and Information Technology. Strong in design and integration with intuitive problem-solving skills. Proficient in C++, PYTHON, JAVASCRIPT, and REACT. Passionate about implementing and launching new projects. Ability to translate business requirements into technical solutions. Looking to start the career as an entry-level software engineer with a reputed firm driven by technology.

**Mrunali Desai.** I have completed my M.E. in Computer from Thadomal Sahani Engineering College and am currently pursuing PhD from KJ Somaiya College of Engineering. I am Currently working as an Assistant Professor in KJ Somaiya Institute of Engineering and Information Technology, Dept. of Computer Engineering. I have 15 years of teaching experience, I am disciplined and prepared for any new challenges.