Data Hiding Using Style Transfer and Diffusion

Pritish Priyadarshi Patra[1], Kshitij Prasad[1], Prakhar Sahu[1] , Punith Kumar H[1] and Vibha TG[2]
[1]*Student, Department of Electronics and Communication, DSCE, Bengaluru, India*
[2]*Assistant Professor, Department of Electronics and Communication, DSCE, Bengaluru, India*

***Abstract* — Traditional image hiding often leaves signs. Changes in pixels or frequency can create distortions that can be spotted by methods. This work checks if information can be hidden well so that advanced tools can't find it. To do this we suggest a two-layer approach. It combines neural style transfer and diffusion-based inpainting to hide information. Secret data, like text or images is first turned into a grid using a code. This grid then goes through a VGG-19 style transfer model. The cover image adds its patterns to the grid. The information gets embedded at a feature level not by changing pixels A second stage uses Stable Diffusion inpainting to remove style transfer artifacts. It rebuilds changed areas in latent space with a mask and prompt. The final images look natural and consistent. Tests show the results of our method. The SSIM is up to 0.78. The PSNR is 30 dB. The LPIPS is below 0.18. Our method is strong, against attacks. It hides information well.**

***Index Terms* — Steganography, Neural Style Transfer, Diffusion Inpainting, VGG-19**

I. INTRODUCTION

Steganography is really important for security. It is a way to hide information. This is different from encryption. Encryption just keeps the content safe. Steganography actually hides the information itself so people do not even know it is there. Steganography is a way to keep things secret. Old methods like Significant Bit and Discrete Cosine Transform are not as safe as they used to be. This is because they leave behind clues that can be detected.

To make steganography better we propose a method called Dual Layer Steganography. Dual Layer Steganography hides information in the features of networks. It uses Neural Style Transfer and Diffusion Inpainting to make the hidden information hard to detect. The secret data is turned into a grid. Embedded into an image using a VGG19 model. The VGG19 model balances the content and style of the image so the image still looks real. We then use Diffusion Inpainting to make the image look more real. This helps to get rid of any parts that do not look right and makes the image look consistent.

We test Dual Layer Steganography using metrics like Structural Similarity Index and Peak Signal-to-Noise Ratio. We also check if Dual Layer Steganography can resist detection by steganalysis techniques. Our results show that Dual Layer Steganography is better at hiding information and the image still looks good.
In addition the proposed Dual Layer Steganography framework explores the trade-off between imperceptibility and robustness by analyzing parameters across both embedding and refinement stages. The integration of feature-space encoding with generative reconstruction not only enhances resistance to conventional steganalysis but also ensures adaptability to complex image structures. Dual Layer Steganography demonstrates the potential of combining learning and generative models to redefine secure data hiding techniques, for modern digital communication systems.

II. RELATED WORK

Recent developments in steganography have changed from methods to deep learning approaches to make hidden messages less noticeable and more secure. The old ways like LSB, DCT and DWT are not safe anymore because they can be detected by tools that analyze hidden messages. To fix these problems researchers have tried using networks to hide data in a way that is more secure and harder to detect. For example Lin and others made a steganography system that uses neural style transfer and Denoising Diffusion Probabilistic Models to hide messages in images.

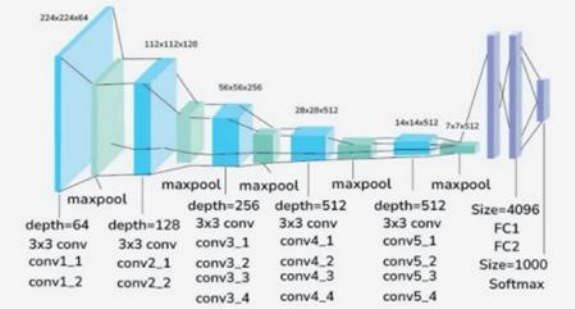

Fig 1. VGG-19 feature extraction layer diagram

Other methods like feature-based steganography try to stop messages from being seen when images are changed by carefully choosing which parts of the network to use and how to hide the messages. These methods show that using feature representations is a good way to hide data in a secure way that looks natural. Some important work by Gatys and others showed that convolutional neural networks can separate what is in an image and what it looks like, which is the basis for style transfer methods. Also diffusion models are very good at making images that look real because they are stable and can make high-quality pictures.

All these new ideas together are the reason for the proposed dual-layer framework, which combines feature-space embedding with diffusion-based refinement to make hidden messages look better, be more secure and be easier to recover. The dual-layer framework is a steganography method that uses deep learning approaches, like neural style transfer and diffusion models to hide messages in a way that is more secure and harder to detect which is the main goal of steganography.

III. METHODOLOGY

The process is divided into multiple stages, starting from encoding the secret data and followed by embedding and refinement steps to achieve a visually natural and secure stego image.

Secret Data Encoding and Grid Formation:-

The system handles two types of secret input: plain text and images. For text, each character is converted into its ASCII value and then into hexadecimal form. For images, the input is downsampled to a fixed resolution and pixel values are similarly translated into a hexadecimal sequence. From here, both inputs follow the same path. Each hexadecimal digit is mapped to a semantic class from a labeled dataset, giving every digit a visual identity tied to a real world object. Using bounding box annotations, representative patches are extracted from each class, resized, and arranged into a structured N by N grid. The result is a spatially organized visual encoding of the secret, where meaning lives in the arrangement rather than in raw numerical values.

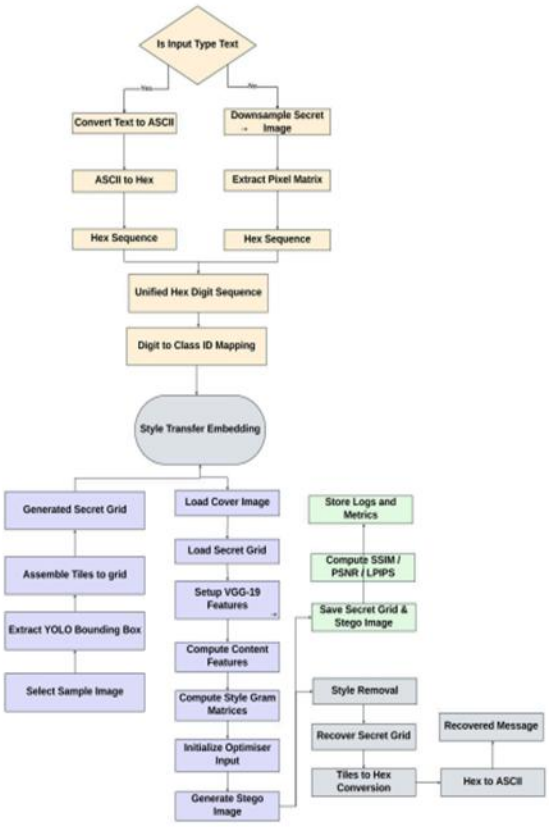

Fig 2. Flow Chart of Implementation

Neural Style Transfer Embedding:-

The secret grid that is made is put into a cover image using a way of changing the style of the image. At this point the grid is like the image and the cover image gives it things like texture and color. A special tool called VGG-19 is used to get the features of the image. This tool is already trained on a lot of images so it can find the things in the grid and the cover image. The important things about the grid are found in a part of the tool, which is called relu4_2. This part finds the high-level structure of the grid. The things that make the cover image look nice are found in parts, like relu1_1, relu2_1, relu3_1, relu4_1 and relu5_1. These

parts help the model find the texture and how the image looks at sizes.

To make the style of the image a special math thing called a Gram matrix is used. This is done with a feature map, which's like a map of the image. The map is changed into a matrix, which's like a table. Then the Gram matrix is calculated to find the connections between the parts of the feature map, which is the style information.

The secret grid is put into the cover image by using a formula that combines the important things about the grid and the cover image. This formula makes sure the new image has the structure as the secret grid and looks like the cover image. The secret. The cover images are combined to make a new image that has the secret grid hidden inside. This new image is made by changing the grid and the cover image, in a special way so the secret grid is embedded into the cover image.

$$G(F) = \frac{1}{CHW} F_{flat} \cdot F_{flat}^{T} \in R^{C \times C} \qquad (1)$$

The generated secret grid is embedded into a cover image using neural style transfer. In this stage, the grid is treated as the content image, while the cover image provides stylistic features such as texture, color distribution, and overall visual appearance. A pre-trained VGG-19 convolutional neural network is used to extract feature representations. Content features are obtained from deeper layers, while style features are computed using Gram matrices across multiple layers. The embedding process minimizes a weighted loss function as shown in equation

$$L = \alpha \cdot L_{content} + \beta \cdot L_{style} \qquad (2)$$

Diffusion-Based Inpainting for Secondary Concealment:-

To enhance concealment and reduce artifacts introduced during style transfer, a second embedding layer based on diffusion inpainting is applied. A Stable Diffusion inpainting model is employed to selectively reconstruct a masked region of the intermediate stego image. A binary mask is generated to define the region of interest, typically a centrally located square region. Gaussian noise is introduced within this masked area, and the model iteratively refines it through a denoising process. The reconstruction is guided by a textual prompt and performed in latent space using a combination of a CLIP text encoder, a UNet-based diffusion network, and a variational autoencoder. This stage improves the naturalness of the image by smoothing inconsistencies and aligning the modified region with real-world image distributions. As a result, as shown in fig 3 the final stego image exhibits improved visual coherence and reduced detectability.

Fig 3. Stego Image Creation

IV. RESULTS AND ANALYSIS

Text and Image to Stego Image:-

The proposed system supports both text and image inputs, which are first converted into a structured visual representation before embedding. For text input, characters are encoded into hexadecimal form through ASCII mapping, while for image input, pixel values are converted into RGB and then hexadecimal format. In both cases, the resulting hex values are mapped to corresponding dataset classes to form a structured grid of object patches representing the secret information. This grid is used as the content input for neural style transfer, with the cover image acting as the style reference. The generated stego image preserves the overall appearance of the cover while embedding the hidden structure within its deep feature space. The effectiveness of this embedding depends on the choice of parameters (α and β). Under optimal settings, high similarity is achieved with SSIM values up to 0.87, PSNR around 33 dB, and LPIPS below 0.2, indicating strong imperceptibility. However, when the content influence is increased, the similarity decreases (SSIM ≈ 0.62, PSNR ≈ 27 dB), highlighting the trade-off between embedding strength and visual similarity. To further improve visual quality, a second layer based on diffusion inpainting is applied. This stage refines a masked region of the stego image through iterative denoising, helping to smooth artifacts introduced during style transfer and making the image appear more natural. Although this refinement slight

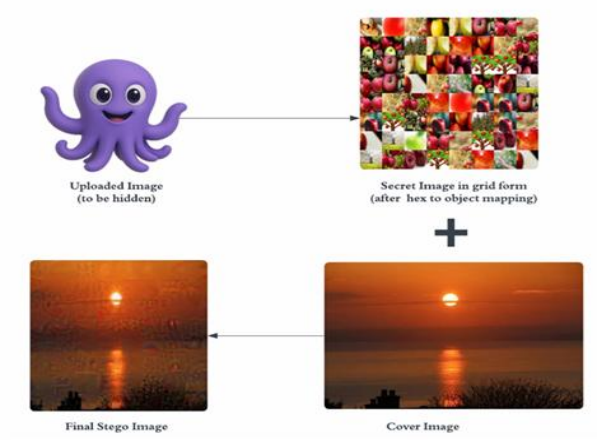

Fig 4. Representation of Stego Flow

Resistance to Statistical Attacks:-

Resistance to Statistical Attacks To evaluate the robustness of the proposed method, we analyze its performance against both first-order and second-order steganalysis techniques.

First-Order Analysis: We perform histogram analysis to compare the pixel intensity distributions of the cover and stego images. The results show only minor differences, suggesting that the embedding process does not significantly disturb the overall statistical properties of the image. Further validation using difference imaging and noise analysis indicates that the changes introduced are small and mostly localized, without creating noticeable anomalies. *Second-Order Analysis:* Edge detection is used to examine the structural consistency between the cover and stego images. The resulting edge maps are largely similar, indicating that the spatial relationships between neighboring pixels are well preserved after embedding.

Additionally, SSIM maps are used to highlight localized differences. These maps show that any modifications are confined to specific regions and do not impact the overall structure of the image.

V. DISCUSSIONS

The results show that hiding information in images using a technique called neural style transfer works really well. This method makes it hard to notice the information, which is better than old methods of hiding data in images. The images with information look very similar to the original images, which is good.The pictures with information look the same as the original pictures, which is great.The pictures have something called SSIM and PSNR values. That means the pictures are really similar to the original pictures.

This happens because the method uses something called the VGG-19 network to separate what the images are, about and what they look like the content and style of the images. Another technique called diffusion-based inpainting is used to make the images with information look even better. This technique helps to fix any mistakes that were made when hiding the information. It makes the output more difficult to detect using statistical methods. Compared to methods that only use style transfer or hide information at the pixel level this method is more robust.It can resist methods of detecting hidden information, such as looking at histograms and edges.

This method is in line with research in hiding information in images and generating new images.It improves the ability to recover the hidden information. Makes the images look more realistic. Unlike methods that only focus on hiding information this method balances hiding information being robust and being able to extract the information reliably.These findings suggest that using learning and generative refinement is a good way to develop secure and practical systems for hiding information.

VI. CONCLUSION

This paper presents a Dual Layer Steganography framework. It is used for hiding data in images. The framework uses neural style transfer and diffusion-based inpainting. This is different from methods that change the pixel values or frequency components of an image. The Dual Layer Steganography framework hides information in the feature space of a VGG-19 network. This makes it harder to detect.

The results of the experiments show that the images with data look a lot like the original images. They are also hard to detect using techniques. The diffusion inpainting helps to make the images look better by reducing artifacts and making the structure more consistent. This makes the Dual Layer Steganography framework good for real-world applications. It balances how hidden the data is, how robust it is and how well the data can be recovered.

The future work can look at making the decoding accuracy better by using object detection or generative reconstruction techniques. This model can also be made to support payloads and things that need to happen in real time. There are a few things that can be done too like making the parameters better and testing the system against advanced steganalysis methods. This will make the system more secure. It will work better. The framework that we are talking about is really good for hiding data in a secure way and it can be made even better if we keep doing research on Data Hiding and Data Hiding techniques.

REFERENCES

[1] Y. Lin, F. Fawad, M.M. Nasralla, M.A. Esmail, H. Mostafa, M. Jia, A steganographic message transmission method based on style transfer and denoising diffusion probabilistic model, IEEE Access 9 (2021) 23409–23423.

[2] C.P. Huang, C.H. Hsieh, delivering messages over user-friendly code images grabbed by mobile devices with error correction, Proc. IEEE Int. Conf. Multimedia and Expo Workshops, Turin, Italy (2015)

[3] L.A. Gatys, A.S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA (2016).

[4] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, Adv. Neural Inf. Process. Syst. 33 (2020) 6840–6851.

[5] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, Proc. European Conf. Computer Vision (ECCV) (2016).

[6] S. Baluja, Hiding images in plain sight: Deep steganography, Adv. Neural Inf. Process. Syst. 30 (2017) 2069–2079.

[7] J. Zhu, R. Kaplan, J. Johnson, L. Fei-Fei, HiDDeN: Hiding data with deep networks, Proc. European Conf. Computer Vision (ECCV) (2018) 657–672.

[8] M. Tancik, B. Mildenhall, R. Ng, StegaStamp: Invisible hyperlinks in physical photographs, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2020) 2117–2126.

[9] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with latent diffusion models, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2022) 10684–10695.

[10] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, RePaint: Inpainting using denoising diffusion probabilistic models, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2022).

[11] X. Zhang, R. Zhang, P. Isola, A.A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2018).

[12] J. Song, C. Meng, S. Ermon, Denoising diffusion implicit models, Int. Conf. Learning Representations (ICLR) (2021).

[13] P. Dhariwal, A. Nichol, Diffusion models beat GANs on image synthesis, Adv. Neural Inf. Process. Syst. 34 (2021).

[14] V. Badrinarayanan, A. Kendall, R. Cipolla, SegNet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39(12) (2017) 2481–2495.

[15] D. Ulyanov, A. Vedaldi, V. Lempitsky, Deep image prior, Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2018).