WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Secure Image Reconstruction using Deep Learning-based Autoencoder with Integrated Encryption Layers

Wurood Salih Abd Ali[1,*]

[1]Department of Computer Techniques Engineering, Alsafwa University College, Karbala 56001, Iraq

*Corresponding Author: Wurood Salih Abd Ali

DOI: https://doi.org/10.31185/wjcms.316
Received 6 November 2024; Accepted 17 December 2024; Available online 30 December 2024

ABSTRACT: This study presents an autoencoder model designed for secure image reconstruction through the integration of encryption and decryption layers within its framework. The major goal is to achieve more effective image reconstruction while safeguarding data integrity. A convolutional neural network (CNN) is first utilized as the primary architecture, attaining a reconstruction accuracy of 90.63% with 2.3737×10^{-4} losses. This brought an opportunity for further improvement, and thus we propose the improved model with the integration of CNN and bidirectional gated recurrent unit (BiGRU) as hybrid model. The integration of CNN-BiGRU leverages the feature extraction advantage of CNN and the temporal processing ability of BiGRU to a great improvement of reconstruction accuracy, reaching 95.57% and validation accuracy stabilizing around 0.85 at the end of training. The model exhibits great accuracy without significant overfitting, thus acquiring robust characteristics crucial for precise image reconstruction. In this work, the hybrid model outperforms the conventional CNN-only architectures for secure image reconstruction and can thus be considered a potential approach when high fidelity with security is required in processing image data.

Keywords: Autoencoder (AE), Convolutional neural network (CNN), Bi-directional gated recurrent neural network (BiGRU), Deep learning (DL).

1. INTRODUCTION

Image data, due to the rapid development of the digital era, has become highly essential in many fields such as health care, national security, media, and personal privacy. This wide usage, on one hand, brings enormous advantages, but on the other hand, it also increases the vulnerability with respect to data theft and manipulation [1]. Security and confidentiality of user image data is paramount. Protection strategies regarding image data have been developed over time to fulfill these requirements. Encryption protects sensitive data against illegal access and interception, while hash functions detect transmission faults and malicious attacks. In image encryption, chaotic systems act as pseudo-random number generators, and the advanced encryption standard (AES) algorithm has been adopted in secure communication protocols, Wi-Fi password encryption, and data compression in order to achieve high security [2]. These approaches are, therefore, still far from achieving an appropriate balance between security performance and efficiency of encryption in the light of efficacy.

Recently, DL finds much important applications in the domain of image processing [3], [4], [5]. Deep learning has attracted considerable attention as a promising alternative, leveraging multi-layer neural networks to extract hierarchical features from raw image data. CNNs have been exhibiting huge advantages in both comprehensive computer vision jobs and image domain translation applications [6], [7]. While it previously took longer, the development in DL has more than accelerated research in the field of artificial intelligence, making it an exceptional tool that offers numerous benefits across a wide range of industries and scientific disciplines. This fast progress has allowed the application in many scientific areas [8]. The end-to-end DL methodology, especially autoencoders (AEs), receives increased interest due to the fact that they are capable of improving the performance of an entire system. It is very useful in places where the exact

models are either unavailable or too expensive to compute [9]. Stacked AE neural networks are used for compressing images, and subsequently, a chaotic logistic map encrypts them [10].

Employing parallel computing with stacked autoencoders reduces runtime complexity and concurrently improves the efficacy of the encryption process [11]. A bidirectional diffusion-based image encryption algorithm is utilized to encrypt 8-bit RGB color images. Thereafter, lossless compression utilizing autoencoders is employed on the encrypted image to enable compression [12]. The autoencoder primarily functions to convert the image into a feature matrix, subsequently encrypted with the AES technique. The decryption and reconstruction of the original image take place on the receiver's end [13]. Encryption methodology for color images utilizing a convolutional AE [14]. The application of convolutional AE for image compression and subsequent encryption produces significant results in maintaining image quality while enabling safe transmission [15]. In previous studies, dense layers and CNNs were employed to develop AEs for encryption and decryption tasks. It is important to note that these studies did not investigate the hybrid structure of DL techniques. In this work, we initially construct an AE using CNNs. Subsequently, we create a hybrid structure that combines CNN with BiGRU for the purpose of encrypting and decrypting images. We employ the CIFAR-10 dataset in this research, the CIFAR-10 dataset contains relatively simple images and is used in this work due to the extensive use as a benchmarking dataset and its appropriateness for preliminary assessment of model performance. However, we note that this may limit the generalization of the model to more complex or diverse datasets. Future work will alleviate this limitation by training and evaluating the model on a wide range of datasets so as to ensure applicability in real-world settings. A substantial enhancement in accuracy is evident in the results, which are achieved through the integration of CNN and BiGRU in the hybrid structure. This work is executed in the Python programming language on the Google Colab platform, which provides a friendly environment supported with specific libraries that facilitate the replication of the experiments and the analysis of results.

The subsequent sections of this work are organized as follows: Section 2 explains the architecture of the CNN-based AE model. This is followed by a description of a hybrid AE architecture that integrates CNN with BiGRU layers. Section 3 presents the simulation results accompanied by an extensive discussion of the findings. Ultimately, Section 4 encapsulates the principal conclusions drawn from this work.

2. DESIGN OF THE DEVELOPED AUTOENCODER

This section consists of two primary subsections: the first discusses the design of the CNN-based AE, and the second explains the design of a hybrid AE architecture that integrates both CNN and BiGRU layers.

2.1 DESIGN OF CNN-BASED AE

In this work, a convolutional autoencoder is designed to reconstruct images from the CIFAR-10 dataset with a deep neural network architecture tuned to capture high-order image features with high data compression and reconstruction. The approach will be subdivided into four major stages: data preprocessing, AE architecture design, model training, and performance evaluation. CIFAR-10 data loading in the data preparation process is composed of 32x32x3 RGB images; here, only the images have been retained and labels discarded since this is an unsupervised learning activity. Normalization of image data consists of scaling the pixel values in the range of [0, 1], which, for reasons of computing efficiency and improving model convergence, serves to diminish input variance. Normalization is a critical step in any deep learning workflow, and in particular in cases where an activation function like ReLU and sigmoid is involved, which are sensitive to the input scales.

The architecture of AE mainly comprises two vital parts: an encoder and a decoder, along with a middle transformation layer for increasing the security of data, which is visualized in Fig. 1. An encoder is formed by three consecutive convolutional layers, each followed by MaxPooling layers that decrease the spatial dimensions progressively while maintaining the most salient information intact. Activation for the encoder layers is done by ReLU, a nonlinear function which will enable the model to learn complex feature hierarchies by selectively activating neurons. The encoder outputs a feature map, a compact representation of the most important information in the input; this maps the 32x32 image onto a 4x4 feature map with eight channels. An encryption layer is integrated after the encoder to enhance data security and introduce non-linear change to the latent space, consisting of a dense layer with 128 units. This layer functions as an "encryption" layer, adding complexity and non-linearity to the encoded representation, which may improve privacy for some applications where data security is paramount. This transformation maps the encoded feature set onto a higher-dimensional space, enhancing information density and further condensing the representation.

After encryption, a corresponding dense layer (256 units) functions as a "decryption" process, transforming the latent representation into a format suitable for the succeeding decoder. This layer reshapes the data appropriately for decoding, corresponding to the encoder architecture. The decoder replicates the encoder's architecture in reverse, progressively UpSampling the compressed features to recover the original image dimensions. Convolutional layers utilize UpSampling, substituting MaxPooling with UpSampling layers to incrementally recover the spatial dimensions. The up-sampling procedure reinstates detail, whilst convolutional layers enhance the restored characteristics. The last layer utilizes a sigmoid activation function to revert pixel values to the normalized [0, 1] range, aligning with the original input format. This AE employs padding in its convolutional layers to preserve the spatial dimensions of the feature maps

across both the encoder and decoder components of the AE. Padding='same' is employed in every convolutional layer to guarantee that the output dimensions correspond to the input dimensions of each layer.

The CAE model employs the Adam optimizer, which adaptively modifies the learning rate throughout training, and utilizes mean squared error (MSE) [16] as the loss function, selected for its effectiveness in quantifying pixel-level reconstruction error. The MSE significant deviations between the original and reconstructed pixels, directing the network to successfully minimize reconstruction loss. The CAE is trained on the CIFAR-10 dataset for 50 epochs using a batch size of 256, optimizing memory efficiency and gradient estimation precision. Model generalization is assessed by recording validation loss and accuracy on a distinct test set following each epoch. This configuration facilitates monitoring the model's capacity to reduce reconstruction error while preserving accuracy in both training and validation datasets. Simultaneously, an auxiliary convolutional model is developed employing a comparable methodology, integrating batch normalization layers subsequent to each convolutional layer. Batch normalization stabilizes the learning process by mitigating internal covariate change, expediting convergence, and enhancing model performance [17].

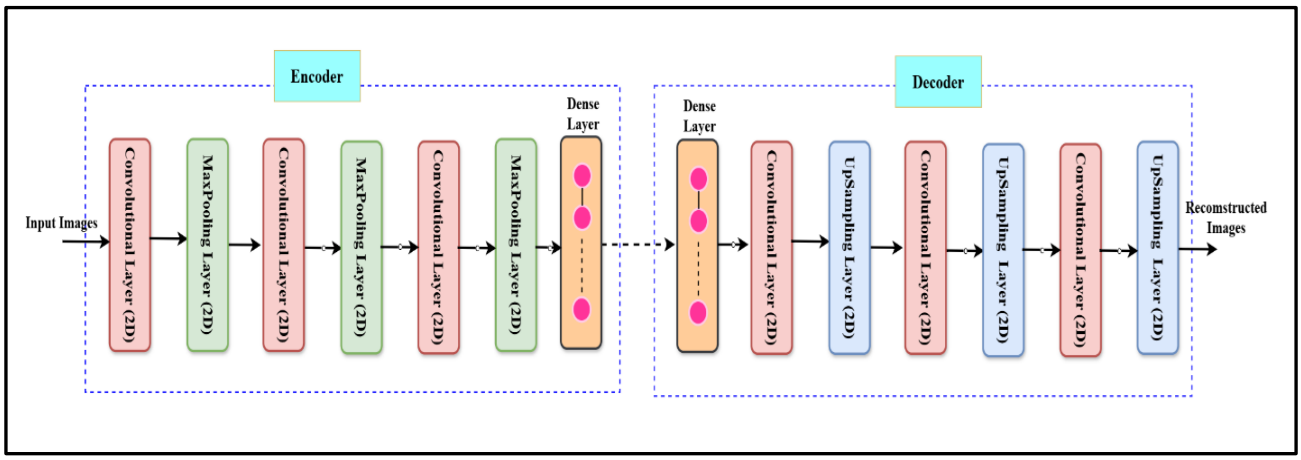

FIGURE 1. - The structure of CAE.

2.2 DESIGN OF HYBRID BIGRU-CNN-BASED AE

This work implements an advanced AE model that integrates CNN and BiGRU layers to accurately reconstruct images from the CIFAR-10 dataset, while incorporating an intermediate encryption layer to improve feature robustness and security as shown in Fig. 2. The CIFAR-10 dataset is first preprocessed by normalizing pixel values to the range [0, 1] to enhance the efficiency of neural network training. Each image, initially a 32x32 RGB representation, is reconfigured to conform to the input dimensions specified by the AE. The model's architecture comprises three primary components: a CNN encoder, a bidirectional GRU-based encoding layer featuring encryption and decryption, and a CNN decoder. Within the encoder, a series of convolutional layers systematically extracts hierarchical feature representations. The initial convolutional layer has 32 filters with a 3x3 kernel with ReLU activation, succeeded by max pooling to reduce the feature map's dimensions. Another convolutional layer with 64 filters picks up the finer features, which are further reduced by another max pooling layer. The output is flattened and reshaped into the dimension (16, 256) as a means of facilitating this transfer from CNN to BiGRU layers for sequential processing. A Bidirectional GRU layer consisting of 128 units subsequently processes these features, both forward and backward temporal dependencies, hence enriching the encoding with sequential context necessary for higher reconstruction quality. The encoded features traverse a dense layer, functioning as an "encryption" method by mapping to a lower-dimensional latent space of 128 units with ReLU activation, so assuring a resilient and compressed feature representation. A second decryption layer with 256 units symmetrically re-expands these features, so restoring the requisite dimensionality for decoding. A second Bidirectional GRU layer, analogous to the encoding GRU, reformats the decoded features into a spatial arrangement of (16, 16, 16), facilitating efficient CNN-based reconstruction. The CNN decoder, analogous to the encoder, utilizes convolutional layers and UpSampling to invert the encoding processes, gradually reinstating the original image resolution. The concluding convolutional layer, equipped with three filters and a sigmoid activation function, produces pixel values ranging from 0 to 1, so reconstructing the image with maximal accuracy. The model is trained utilizing the Adam optimizer with MSE as the loss function, across 50 epochs and a batch size of 256, incorporating both training and validation datasets to assess performance.

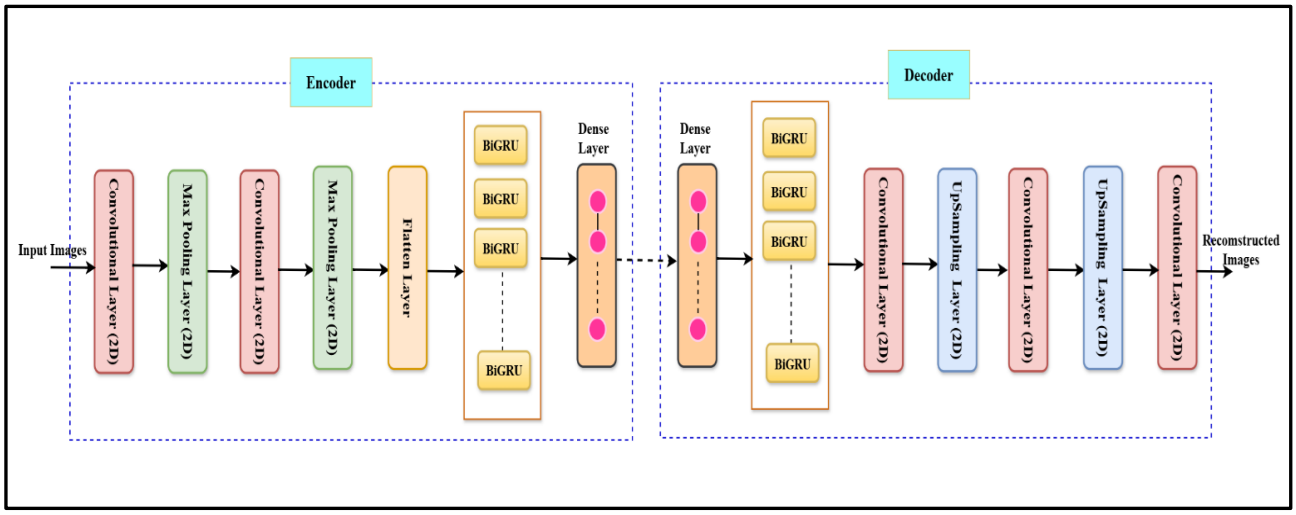

FIGURE 2. - The structure of CNN-BiGRU-AE.

Moreover, recurrent neural networks (RNNs) are commonly employed to discern relationships in sequential data characterized by temporal dependencies. The long short-term memory (LSTM) model is especially good at handling problems of vanishing gradient and gradient explosion, commonly seen in standard RNNs. GRU is a variant of LSTM, where the issue of gradient explosion is prevented, and it uses gate cells in order to control the input flow within the network. This makes GRU simpler than LSTM in its implementation. On the other hand, GRU is the simplified version of LSTM that also uses a version of gated cells to control the information flow within a network and hence is easier to implement than LSTM [18]. Each GRU cell comprises two gates: an update gate and a reset gate. An update gate governs the influx of control information into the subsequent moment, whereas a reset gate manages the retention of information. The two gates collectively determine the output of the concealed state [19]. Fig. 3 elucidates the architecture of the GRU unit, which derives the final output by integrating the current input tx_t and the preceding state h_{t-1}, while accounting for the cumulative effect of these gates. A summary of the internal gate outputs of the GRU unit is presented below [20].

$$
\begin{aligned}
r_t &= \sigma\,(W_r\ [h_{t-1}, tx_t] + b_r) \\
z_t &= \sigma\,(W_z\ [h_{t-1}, tx_t] + b_z) \\
\tilde{h}_t &= tanh\,(W_h\ [r_t \odot h_{t-1}, tx_t] + b_h) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t
\end{aligned}
\qquad (1)
$$

where W_r, W_z and W_h represent the weight matrices for the reset gate, the update gate, and the fresh memory computation, respectively. The bias vector b_r, b_z, and b_h are interrelated. The sigmoid function σ is employed for both the reset and update gates. In memory computation, the hyperbolic tangent activation function is referred to as tanh, while the Hadamard product is symbolized as ⊙.

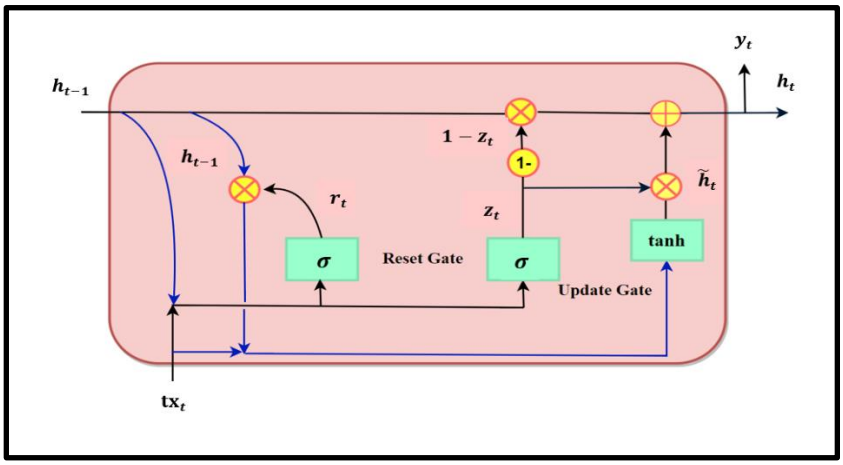

FIGURE 3. - Diagram illustrating the structural elements of a GRU memory unit, modified from [19].

3. RESULTS AND DISCUSSION

At training, the CAE's performance is assessed by quantitative and qualitative evaluations. the model's performance as depicted by the loss and accuracy curves suggests that it has effectively learned the task while avoiding overfitting as explain in Fig. 4. The rapid convergence of both losses to near-zero and the stabilization of both accuracy curves around 90% demonstrate efficient learning and good generalization capabilities. After the CAE is trained, the value of accuracy = 90.63% and losses = $2.3737 \text{x} 10^{-4}$, The small difference between training and validation metrics across epochs reflects a well-tuned model that can perform reliably on unseen data, as it has not merely fit the training data but also learned features that generalize across the dataset. As shown in Fig. 5, the similarity between the two histograms indicates that the reconstructed image aligns with the original for overall brightness and contrast distribution. Using CAE preserves a high level of accuracy, accurately reflecting the original image's intensity distribution while minimally sacrificing finer features. This is a favorable result, particularly if the objective is to preserve the overall visual integrity of the original image with little perceptual degradation. The reconstructed images are qualitatively produced by processing test images via the trained CAE as shown in Fig. 6. Original and rebuilt images are presented adjacently to evaluate reconstruction fidelity visually. This visual assessment aids in determining the extent to which the CAE maintains critical visual attributes following compression and reconstruction.

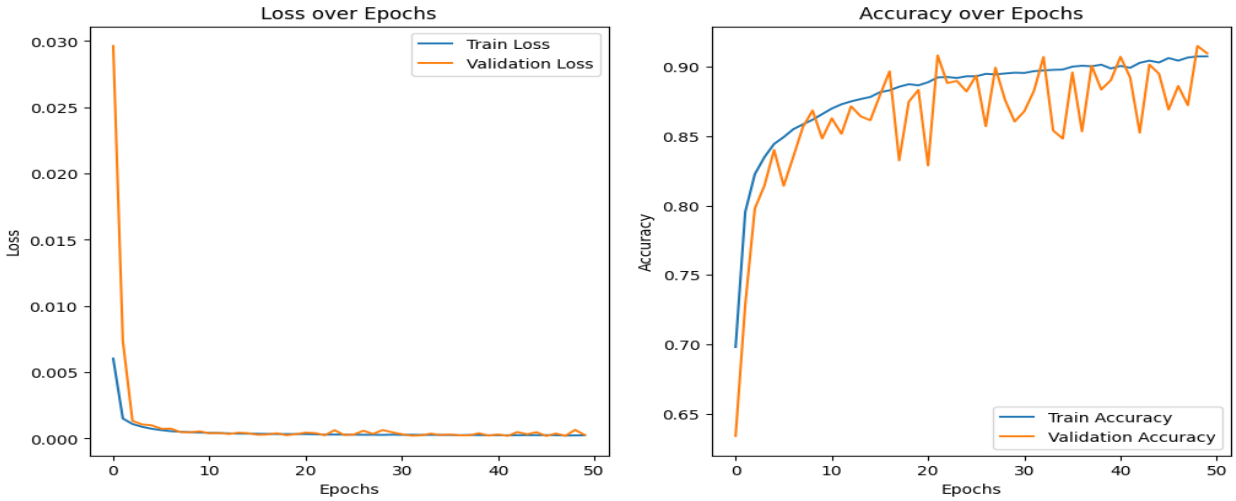

FIGURE 4. - The variation of (a) losses, (b) accuracy with respect to epochs, respectively, using CAE.

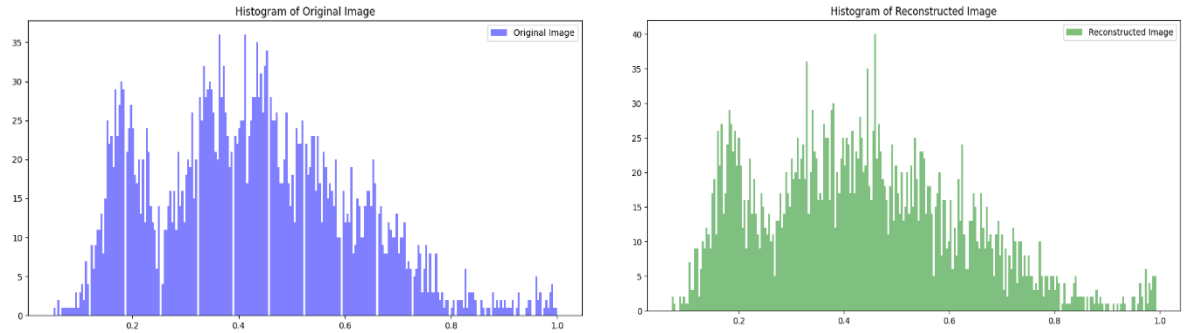

FIGURE 5. - The histogram of (a) original image, and (b) reconstructed image, respectively, using CAE.

FIGURE 6. - The original images, and reconstructed image, using CAE.

Figure 7 (a) depicts the training and validation loss, whereas Figure 7 (b) displays the training andvalidation accuracy across 50 epochs for the CNN-BiGRU AE model developed for secure image reconstruction. At the outset, both training and validation losses are elevated, indicating the model's untrained condition. As training advances, the loss on both datasets markedly diminishes, with the training loss consistently nearing zero, indicating successful learning. The validation loss initially declines, then stabilizes at roughly 0.5 after around 15 epochs. This peak signifies effective generalization, exhibiting limited deviation from the training loss, indicating only marginal overfitting. The accuracy plot demonstrates a swift improvement in both training and validation accuracy, with training accuracy approaching 95.57% and validation accuracy stabilizing around 0.85 at the end of training. The near alignment of these curves signifies that the model exhibits great accuracy without significant overfitting, thus acquiring robust characteristics crucial for precise image reconstruction.

Figure. 8 illustrates the histogram of the reconstructedimage; the reconstructed image roughly replicates the original in terms of overall brightness and contrast distribution. Employing CNN-BiGRU AE maintains a high degree of precision, faithfully representing the original image's intensity distribution. In Fig.9, the original and reconstructed images are displayed side by side to visually assess reconstruction fidelity. This visual evaluation assists in assessing the degree to which the CNN-BiGRU AE preserves essential visual characteristics after compression and reconstruction.

This work focuses primarily on reconstruction accuracy and loss metrics in assessing the performance of the proposed model. However, I ware that other performance measures, such as processing speed and robustness against various types of noise, are also important. Such indicators are necessary for a well-rounded evaluation, particularly for practical applications. Future research will involve such assessments, where the model will be further tested under various noise conditions and computational constraints to provide a fuller description of its performance.

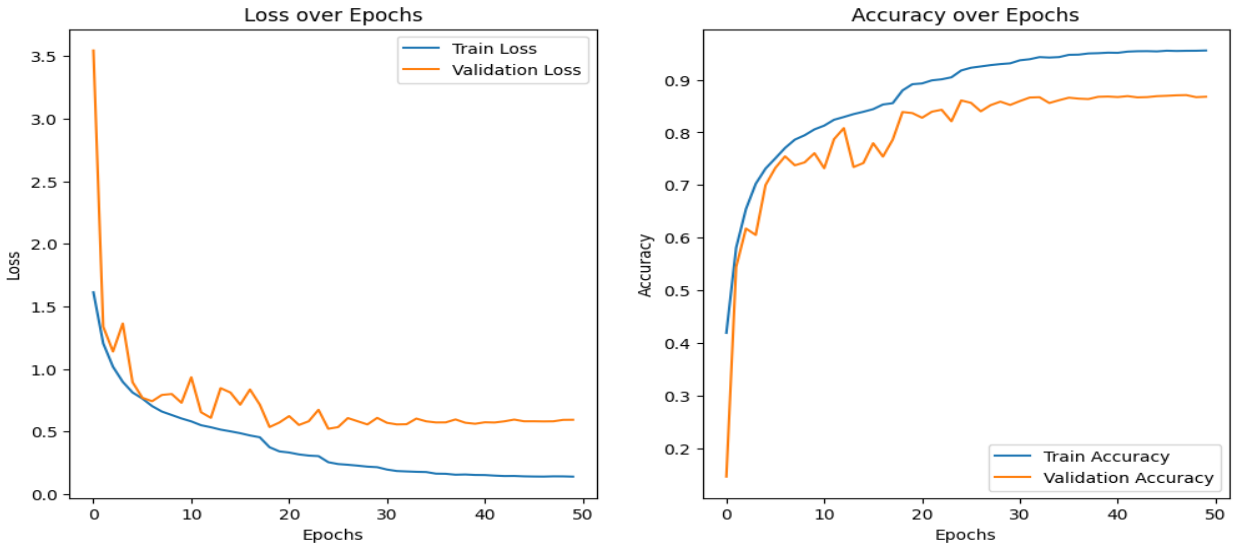

FIGURE 7. - The variation of (a) losses, (b) accuracy with respect to epochs, respectively, using CNN-BiGRU AE.

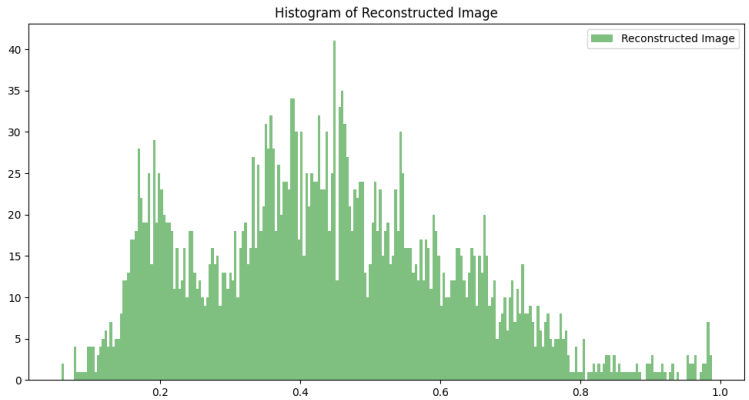

FIGURE 8. - The histogram of reconstructed image, respectively, using CNN-BiGRU AE.

FIGURE 9. - The original images, and reconstructed image, using CNN-BiGRU AE.

4. CONCLUSION

This work introduces an AE model with encryption and decryption layers for secure image reconstruction. Data integrity and enhanced images reconstruction are the main goals. The initial architecture, a CAE, achieved 90.63% reconstruction accuracy. We proposed integrating CNN with BiGRU to create a hybrid architecture for further development. The CNN-BiGRU integration uses CNN's feature extraction and BiGRU's temporal processing to improve reconstruction accuracy to 95.57%. Our hybrid model outperforms CNN-only solutions in secure image reconstruction tasks, making it a possible method for high-fidelity, secure image data processing, with the objective of generating high-quality image reconstructions while minimizing loss and maximizing accuracy. This work can be expanded to implement another encryption technique such as chaotic encryption and incorporate it in developed AE to achieve higher security.

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] L. Dai, L. Hu, L. Chen, C. Wang, and F. Lin, “An Image Double Encryption Based on Improved GAN and Hyper Chaotic System,” *IEEE Access*, vol. 12, pp. 135779–135798, 2024, doi: 10.1109/ACCESS.2024.3462547.

[2] M. Abdelmalek, A. Harhoura, I. Elaloui, M. Madani, and E.-B. Bourennane, “Visually Image Encryption based on Efficient Deep Learning Autoencoder,” Academy and Industry Research Collaboration Center (AIRCC), Jun. 2024, pp. 57–64. doi: 10.5121/csit.2024.141206.

[3] K. Panwar, A. Singh, S. Kukreja, K. K. Singh, N. Shakhovska, and A. Boichuk, “Encipher GAN: An End-to-End Color Image Encryption System Using a Deep Generative Model,” *Systems*, vol. 11, no. 1, Jan. 2023, doi: 10.3390/systems11010036.

[4] R. Archana and P. S. E. Jeevaraj, “Deep learning models for digital image processing: a review,” *Artif Intell Rev*, vol. 57, no. 1, Jan. 2024, doi: 10.1007/s10462-023-10631-z.

[5] S. S. Roy, C.-H. Hsu, and V. Kagita, “Deep Learning Applications in Image Analysis.”

[6] Y. Ding *et al.*, “DLEDNet: A Deep Learning-based Image Encryption and Decryption Network for Internet of Medical Things”, doi: 10.48550/arXiv.2004.05523.

[7] Z. Wang, M. Qin, and Y.-K. Chen, “Learning from the CNN-based Compressed Domain,” in *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, pp. 4000–4008. doi: 10.1109/WACV51458.2022.00405.

[8] A. M. Abbass and R. S. Fyath, “Performance investigation of geometric constellation shaping-based coherent WDM optical fiber communication system supported by deep-learning autoencoder,” *Results in Optics*, vol. 15, p. 100629, 2024, doi: https://doi.org/10.1016/j.rio.2024.100629.

[9] A. M. Abbass and R. S. Fyath, “Simulation Platform for End-to-End Deep Learning Based Coherent Fiber Communication System,” in *2023 3rd International Scientific Conference of Engineering Sciences, ISCES 2023 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 59–63. doi: 10.1109/ISCES58193.2023.10311492.

[10] F. Hu, C. Pu, H. Gao, M. Tang, and L. Li, “An image compression and encryption scheme based on deep learning,” Aug. 2016, doi: 10.48550/arXiv.1608.05001.

[11] T. V, N. Mantripragada, A. P. Singh, and N. Bhasin, "Encrypting Multiple Images using Stacked Autoencoders," in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, 2019, pp. 1–6. doi: 10.1109/ViTECoN.2019.8899495.

[12] K. Sreelakshmi and R. V Ravi, "An Encryption-then-Compression Scheme Using Autoencoder Based Image Compression for Color Images," in *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, 2020, pp. 1–5. doi: 10.1109/ICSSS49621.2020.9201967.

[13] Y. Alslman, E. Alnagi, A. Ahmad, Y. AbuHour, R. Younisse, and Q. Abu Al-haija, "Hybrid Encryption Scheme for Medical Imaging Using AutoEncoder and Advanced Encryption Standard," *Electronics (Switzerland)*, vol. 11, no. 23, Dec. 2022, doi: 10.3390/electronics11233967.

[14] F. Ahmed *et al.*, "A DNA Based Colour Image Encryption Scheme Using A Convolutional Autoencoder," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 19, Nov. 2022, doi: 10.1145/3570165.

[15] A. K. Naveen, S. Thunga, A. Murki, M. A. Kalale, and S. Anil, "Autoencoded Image Compression for Secure and Fast Transmission," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.03990

[16] S. Kato and K. Hotta, "MSE Loss with Outlying Label for Imbalanced Classification," Jul. 2021, [Online]. Available: http://arxiv.org/abs/2107.02393

[17] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

[18] X. Hu, Y. Huo, X. Dong, F. Y. Wu, and A. Huang, "Channel Prediction Using Adaptive Bidirectional GRU for Underwater MIMO Communications," *IEEE Internet Things J*, vol. 11, no. 2, pp. 3250–3263, Jan. 2024, doi: 10.1109/JIOT.2023.3296116.

[19] X. Yin, C. Liu, and X. Fang, "Sentiment Analysis based on BiGRU Information Enhancement," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jan. 2021. doi: 10.1088/1742-6596/1748/3/032054.

[20] Z. Liu, X. Liu, S. Xiao, W. Yang, and W. Hu, "Bi-GRU Enhanced Cost-Effective Memory-Aware End-to-End Learning for Geometric Constellation Shaping in Optical Coherent Communications," *IEEE Photonics J*, vol. 16, no. 1, pp. 1–10, Feb. 2024, doi: 10.1109/JPHOT.2023.3344184.