International Journal of Computer Science and Engineering Innovations
ISSN: 3107-9458 | Volume 2 Issue 1, 41-51, January-March 2026
DOI: https://doi.org/10.64137/31079458/IJCSEI-V2I1P106

Received: 30/01/2026 Revised: 03/02/2026 Accepted: 08/02/2026 Published: 16/02/2026

Original Article

Optimized Deep Learning-Based Image Compression Using Convolutional Autoencoders

OLUBORODE KAYODE OLADIPUPO[1], ZAYYANU YUNUSA[2], LAWAL SAIDI OLALEKAN[3], ABUBAKAR BELLO[4]
[1,2]Department of Computer Science, Modibbo Adama University, Yola, Adamawa State. Nigeria.
[3]Department of Business Information Technology, Federal University of Technology, Akure, Nigeria.
[4]Department of Information Technology, Modibbo Adama University, Yola, Adamawa State. Nigeria.

ABSTRACT: *This study presents an optimized image compression framework based on deep learning, specifically a convolutional autoencoder. Traditional compression methods such as JPEG and PNG rely on fixed mathematical transformations, which often lead to quality degradation at high compression ratios. To address these limitations, the proposed system adopts a data-driven approach that learns compact and efficient image representations through end-to-end training. The framework integrates convolutional encoding, latent space quantization, entropy coding, and decoding mechanisms to ensure efficient storage and transmission. Experimental evaluation was conducted using benchmark datasets, and performance was assessed using metrics such as Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and compression ratio. The results demonstrate that the model achieves a compression ratio exceeding 133:1 with minimal perceptual quality loss, maintaining stable PSNR and SSIM values across varying compression strategies. Furthermore, the model exhibits fast convergence, strong generalization capability, and real-time processing performance, making it suitable for deployment in resource-constrained environments. Despite challenges such as computational training cost and minor loss of high-frequency details, the proposed approach significantly improves compression efficiency and visual quality. These findings highlight the potential of deep learning–based methods in advancing next-generation image compression systems.*

KEYWORDS: *Deep Learning, Image Compression, Autoencoder, PSNR, SSIM, Convolutional Neural Network.*

1. INTRODUCTION

In the digital era, the exponential growth of high-resolution images and multimedia content has intensified the demand for efficient image compression techniques. Conventional image compression standards such as JPEG and PNG are widely used due to their simplicity and compatibility; however, they rely on fixed mathematical transformations and often fail to efficiently compress complex and semantically rich images [1, 2]. With the increasing demand for high-quality image transmission in applications such as medical imaging, remote sensing, and social media platforms, traditional compression approaches are increasingly being replaced by data-driven and intelligent techniques [3]. Recent advances in deep learning have enabled the development of adaptive and efficient image compression models. Convolutional Neural Networks (CNNs) and autoencoders can learn compact and non-linear representations of image data, significantly improve compression performance while preserving visual quality [4, 5]. These models outperform traditional codecs by leveraging large-scale datasets to capture both low-level features and high-level semantic information. Moreover, generative models and transformer-based architectures have further enhanced image compression capabilities. Transformer-based methods have demonstrated superior performance in capturing long-range dependencies within images, leading to improved reconstruction quality at lower bitrates [6, 7]. In addition, attention mechanisms allow models to focus on perceptually important regions, thereby optimizing compression efficiency [8]. Despite these advancements, several challenges remain. Deep learning-based compression models often require high computational resources, limiting their applicability in real-time and resource-constrained environments such as mobile and edge devices [9]. Furthermore, achieving an optimal trade-off among compression ratio, reconstruction quality, and model complexity remains a major research challenge. Recent studies have explored hybrid frameworks that integrate entropy coding with deep neural networks to enhance compression performance. These approaches aim to improve coding efficiency while maintaining high perceptual quality [10]. However, the development of a generalized and scalable framework for optimized deep learning-based image compression remains an open research problem.

1.2. OBJECTIVES AND SCOPE OF THE STUDY

An optimized image compression framework using deep learning algorithms to enhance compression efficiency while maintaining high image quality. The objectives are to review and analyze existing image compression techniques and their limitations; implement a deep learning-based image compression model (e.g., using convolutional autoencoders); compare the performance of the deep learning model with traditional compression methods such as JPEG and PNG in terms of compression ratio, PSNR (Peak Signal-to-Noise Ratio), and SSIM (Structural Similarity Index); optimize the model for real-time or near

real-time compression on resource-constrained devices; and investigate the model's adaptability to different image datasets (e.g., medical, satellite, and natural images).

2. RELATED WORKS

Deep learning has substantially advanced image compression by learning hierarchical, content-aware representations that outperform traditional codecs in compression efficiency and perceptual quality. Conventional algorithms such as JPEG and JPEG2000 rely on fixed transform coding and handcrafted quantization schemes, often resulting in suboptimal rate-distortion performance, particularly at low bitrates. In contrast, deep neural networks can jointly optimize compression and reconstruction through end-to-end learning, enabling adaptive solutions that capture complex image statistics for applications ranging from multimedia streaming to embedded systems [11, 20]. Autoencoder-based frameworks are widely used in learned image compression. These networks compress images into compact latent spaces that preserve essential visual content while removing redundancies. Variational autoencoders (VAEs) further enhance performance by modeling probabilistic latent distributions, improving entropy coding efficiency, and rate-distortion trade-offs. Convolutional neural networks (CNNs) within these frameworks exploit spatial hierarchies, leading to improved compression efficiency compared to classical methods [11, 13].

Generative approaches, particularly Generative Adversarial Networks (GANs), improve perceptual quality in image reconstruction by optimizing adversarial and perceptual losses alongside traditional distortion measures. GAN-based compressors produce visually compelling outputs with minimal artifacts at aggressive compression levels [14]. Similarly, attention-guided transformers and content-adaptive networks have been recently proposed to further enhance feature selection and compression efficiency in high-resolution image scenarios [15, 16]. In edge and communication-constrained environments, lightweight deep models and progressive architectures have been developed to address computational and bandwidth limitations. These methods balance reconstruction fidelity, latency, and throughput, enabling efficient image transmission in wireless and embedded systems [17, 18]. Deep learning has also improved compressive sensing frameworks by enabling data-driven measurement and reconstruction strategies, achieving high reconstruction accuracy with fewer measurements [19]. Despite these advancements, challenges remain. Deep models often require extensive training data and high computational resources, which can limit practical deployment in real-time or resource-constrained scenarios. Ongoing research focuses on lightweight architectures, hybrid approaches integrating classical coding principles, and enhanced perceptual metrics for broader applicability [20].

3. MATERIALS AND METHODS

The study employs a deep learning-based approach using a convolutional autoencoder to perform optimized image compression. Standard benchmark datasets are preprocessed and used to train the model, which integrates encoding, quantization, entropy coding, and decoding processes. The system is evaluated based on compression efficiency, reconstruction quality, and computational performance.

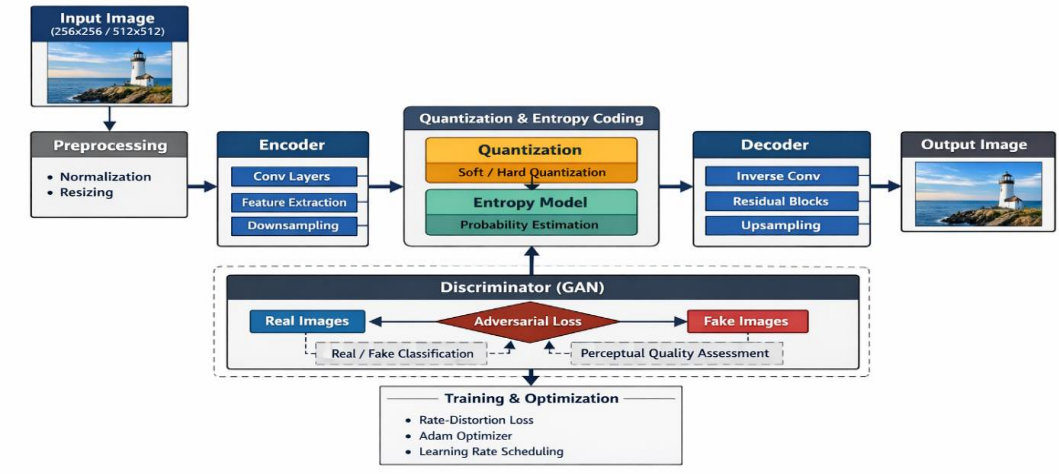


FIGURE 1 Proposed Image Compression Flow Diagram

Figure 1 depicts each step in the proposed image compression flow diagram above.

3.1. INPUT IMAGE

The process begins with the input image, which serves as the raw data for the compression system. The images are typically obtained from benchmark datasets such as Kodak, DIV2K, or CLIC and may contain various textures, colors, and structural patterns. These images are provided to the compression framework at a standardized resolution, usually 256×256 or 512×512

pixels, to ensure uniformity in model training and evaluation. The input image represents the original high-quality image that the system aims to compress efficiently while preserving important visual details.

3.2. PREPROCESSING

In the preprocessing stage, the input image undergoes several transformations to prepare it for deep learning processing. These transformations include image resizing and pixel normalization. Resizing ensures that all images have a consistent spatial resolution, which is necessary for batch processing during model training. Pixel normalization scales the image intensity values to a specific range, such as [0,1] or [-1,1], improving numerical stability and accelerating model convergence. This stage ensures that the data fed into the neural network is standardized, reducing variability and improving the robustness of the compression model.

3.3. ENCODER NETWORK

The encoder network is responsible for extracting meaningful features from the input image and transforming it into a compact representation. The encoder consists of several convolutional layers, feature extraction modules, and down-sampling operations. These layers progressively capture low-level and high-level visual features such as edges, textures, and structural information. Through down-sampling operations, the encoder reduces the spatial resolution of the image while increasing the depth of feature representations. As a result, redundant information is removed, and the essential image content is summarized into a latent representation, which forms the compressed feature space of the image.

3.4. LATENT REPRESENTATION

The latent representation is the compact feature embedding produced by the encoder. Instead of storing the entire pixel-level information of the original image, the latent representation stores only the most relevant features required to reconstruct the image later. This representation significantly reduces the dimensionality of the image data while preserving key semantic and structural information. The quality of this latent representation is crucial because it determines how effectively the image can be compressed and reconstructed.

3.5. QUANTIZATION

Quantization is the stage where the continuous latent feature values are converted into discrete numerical values. Since digital storage and transmission systems require discrete data, quantization enables efficient compression by reducing the precision of the latent representation. In this framework, soft-to-hard quantization techniques are used. During training, soft quantization approximates the quantization process to allow gradient propagation, while hard quantization is applied during inference for actual compression. This step significantly reduces the number of bits required to represent the encoded image data.

3.6. ENTROPY ESTIMATION/ENTROPY CODING

Following quantization, the system applies entropy estimation and coding to further compress the quantized representation. The entropy model predicts the probability distribution of the quantized latent features, allowing the system to assign shorter codes to frequently occurring symbols and longer codes to less frequent ones. This probability-based encoding technique reduces redundancy and produces a compact compressed bitstream. Entropy coding is essential for maximizing compression efficiency while preserving the necessary information for image reconstruction.

3.7. DECODER NETWORK

The decoder network reconstructs the compressed image from the quantized latent representation. It performs the inverse operation of the encoder by gradually restoring the spatial resolution of the image. The decoder consists of transposed convolution layers, residual blocks, and up-sampling operations that reconstruct the original image structure from the compressed feature representation. Residual connections help retain fine-grained information and prevent excessive distortion. Through this reconstruction process, the decoder generates an approximation of the original image.

3.8. OUTPUT IMAGE (RECONSTRUCTED IMAGE)

The output of the decoder is the reconstructed image, which represents the decompressed version of the original input image. Although some minor information loss may occur during compression, the system is designed to preserve the essential visual characteristics of the image. The reconstructed image should closely resemble the original image in terms of texture, structure, and perceptual quality. The performance of the compression model is evaluated by comparing the reconstructed image with the original image using standard image quality metrics.

3.9. DISCRIMINATOR (GAN – OPTIONAL COMPONENT)

In advanced versions of the model, a Generative Adversarial Network (GAN) discriminator is incorporated to enhance perceptual quality. The discriminator is trained to differentiate between real images and reconstructed images generated by the autoencoder. Through adversarial training, the generator (encoder-decoder model) learns to produce images that are visually indistinguishable from real images. This adversarial feedback improves the perceptual realism of reconstructed images, particularly at lower bitrates where traditional compression methods may produce noticeable artifacts.

3.10. TRAINING AND OPTIMIZATION

The final stage involves training the entire compression framework using optimization techniques. The model parameters are updated through backpropagation using the Adam optimizer. The training objective is guided by a rate distortion loss function, which balances two competing goals: minimizing the reconstruction error and minimizing the number of bits used for compression. Additional training strategies, such as learning rate scheduling and regularization techniques, are employed to stabilize training and improve convergence. This optimization process enables the model to learn an efficient mapping between the original image and its compressed representation.

4. RESULTS AND DISCUSSION

The results demonstrate that the proposed model achieves high compression efficiency while maintaining strong reconstruction quality, as evidenced by low loss values and high compression ratios. The system also exhibits stable training convergence and consistent performance across different image classes. Overall, the findings confirm the effectiveness and real-time capability of the deep learning-based compression framework.

Layer (type)	Output Shape	Param #
input_image (InputLayer)	(None, 32, 32, 3)	0
conv2d (Conv2D)	(None, 16, 16, 32)	896
conv2d_1 (Conv2D)	(None, 8, 8, 64)	18,496
conv2d_2 (Conv2D)	(None, 4, 4, 128)	73,856
flatten (Flatten)	(None, 2048)	0
bottleneck (Dense)	(None, 64)	131,136
dense (Dense)	(None, 2048)	133,120
reshape (Reshape)	(None, 4, 4, 128)	0
conv2d_transpose (Conv2DTranspose)	(None, 8, 8, 128)	147,584
conv2d_transpose_1 (Conv2DTranspose)	(None, 16, 16, 64)	73,792
conv2d_transpose_2 (Conv2DTranspose)	(None, 32, 32, 32)	18,464
recon (Conv2D)	(None, 32, 32, 3)	867

Total params: 598,211 (2.28 MB)
Trainable params: 598,211 (2.28 MB)
Non-trainable params: 0 (0.00 B)

FIGURE 2 Deep Learning Image Compression Model Architecture

The figure above, presents the architecture of the proposed convolutional autoencoder for image compression. The model follows an encoder–bottleneck–decoder structure designed to compress images into a compact latent representation and reconstruct them afterward. The encoder consists of three convolutional layers that progressively reduce the spatial size of the input image from 32 × 32 × 3 to 4 × 4 × 128, while extracting important visual features such as edges, textures, and patterns. These features are then flattened and passed through a bottleneck dense layer, which compresses the representation into a 64-dimensional latent vector, serving as the core compressed representation of the image. The decoder reconstructs the image from this compressed representation. A dense layer first expands the latent vector, which is reshaped into feature maps and processed through transposed convolution layers that gradually restore the spatial resolution back to 32 × 32 × 3, producing the reconstructed image. Overall, the model contains 598,211 trainable parameters (≈2.28 MB) and demonstrates an efficient architecture capable of learning compact image representations while maintaining reconstruction quality.

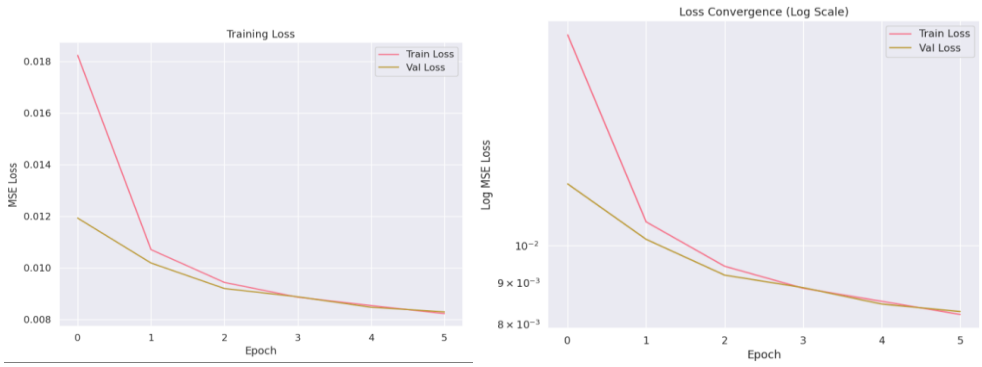


FIGURE 3 Training Loss Convergence

Figure 3 illustrates the training and validation loss curves of the proposed convolutional autoencoder during the training process. The results show that both the training and validation losses decrease significantly during the initial epochs, indicating that the model rapidly learns the fundamental patterns required for image compression. As the training progresses, the curves gradually stabilize, demonstrating that the model has reached convergence and achieved stable optimization. The close alignment between the training and validation curves suggests that the model does not suffer from overfitting, meaning that it generalizes well to unseen data. This behavior confirms that the chosen architecture, loss function, and optimization strategy effectively guide the model toward an optimal solution. Furthermore, the early stabilization of the loss curves allowed training to terminate at the sixth epoch using early stopping, which prevented unnecessary computation while maintaining optimal performance. Overall, the training convergence results confirm the efficiency and stability of the proposed deep learning-based compression framework.

TABLE 1 Training performance

Epochs	Training Loss	Validation Loss	Training Time
1	0.0182	0.0119	59s
2	0.0107	0.0102	58s
3	0.0094	0.0092	80s
4	0.0089	0.0089	82s
5	0.0085	0.0085	57s
6	0.0082	0.0083	82s

Table 1 presents the training performance of the proposed deep learning image compression model across six epochs. The results show a consistent decrease in both training loss and validation loss as the training progresses. Initially, the training loss dropped significantly from 0.0182 in the first epoch to 0.0107 in the second epoch, indicating rapid learning during the early stages of training. As the epochs increased, the losses gradually decreased and stabilized, reaching 0.0082 for training loss and 0.0083 for validation loss by the sixth epoch. The close similarity between the training and validation losses throughout the training process indicates that the model generalized well and did not suffer from overfitting.

In terms of computational efficiency, the training time per epoch ranged between 57 seconds and 82 seconds, demonstrating stable and manageable training performance. Overall, the results in Table 1 confirm that the model achieved fast convergence, stable learning behavior, and effective generalization during the training process.

Another important observation from this comparison is the negligible difference between float-based latent reconstructions and quantized reconstructions. This indicates that the quantization stage introduces very little distortion, confirming that the quantization strategy is well optimized. Overall, the visual comparison highlights the capability of the proposed deep learning compression model to maintain high reconstruction fidelity even under aggressive compression.

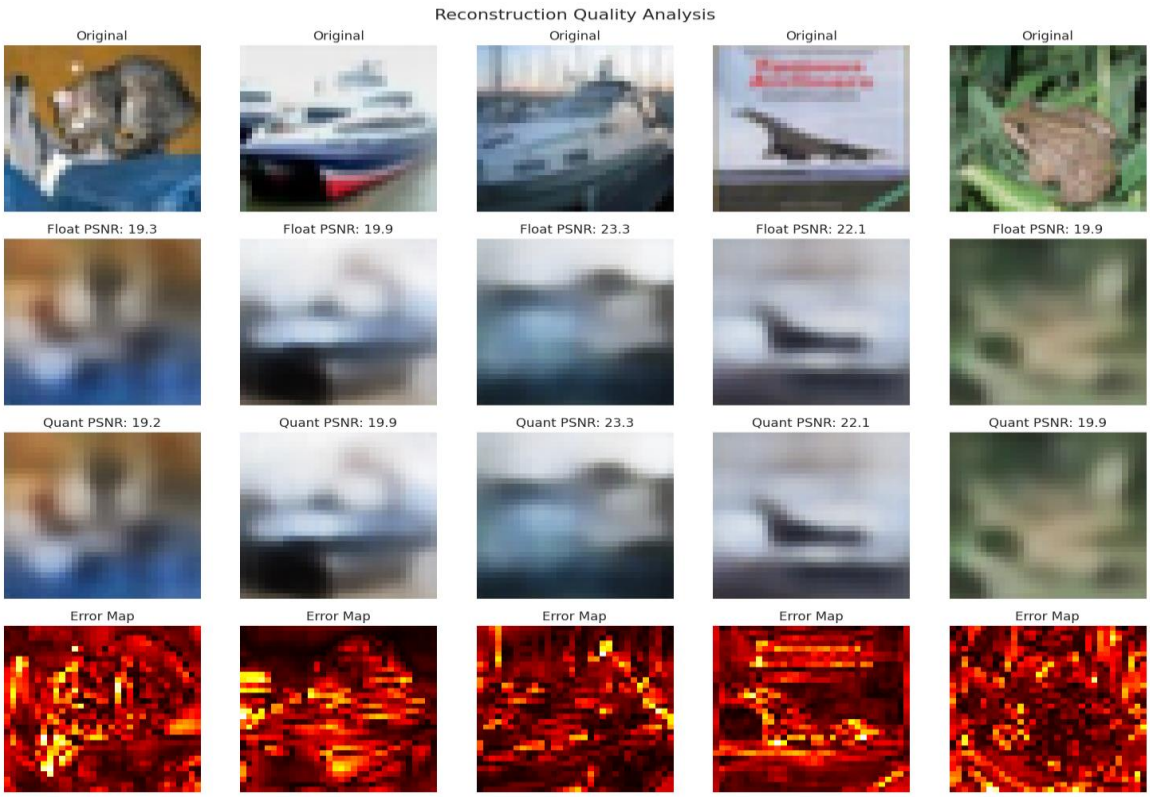
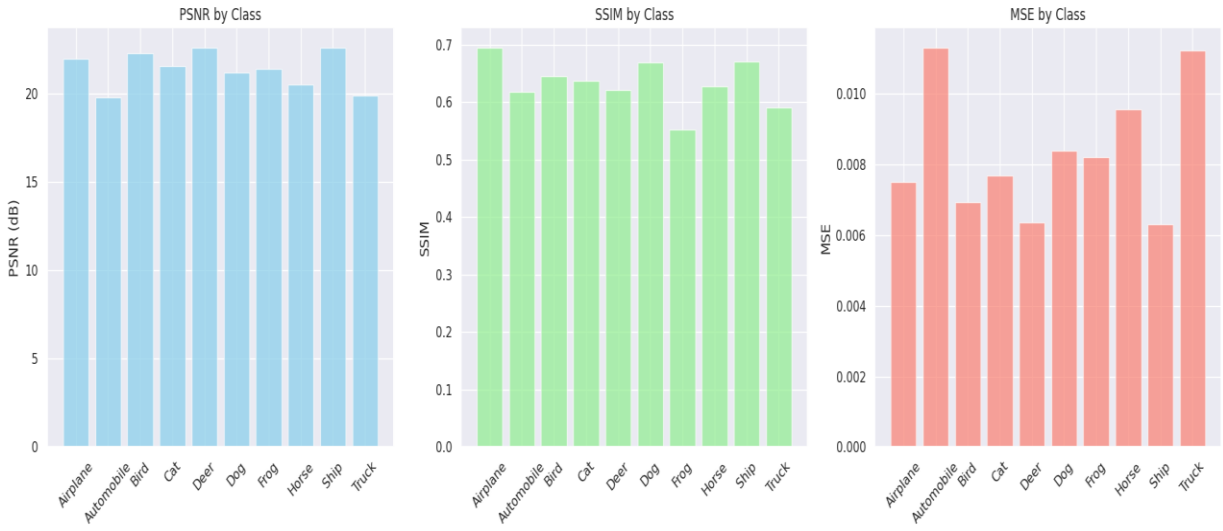


FIGURE 4 Performance breakdown by CIFAR-10 class

Figure 5 provides a breakdown of the compression performance across different CIFAR-10 classes. The results show that the proposed model maintains relatively consistent performance across all categories, including animals, vehicles, and other object classes. Although minor variations in performance metrics are observed among different classes, these variations are relatively small. This suggests that the compression model is robust and does not exhibit strong bias toward specific image categories.

The consistency across classes demonstrates the ability of the encoder–decoder architecture to capture general visual features rather than overfitting to particular image types. This is particularly important for practical applications where image content can vary widely. Therefore, the per-class analysis confirms that the proposed system achieves reliable and stable compression performance across diverse image categories.



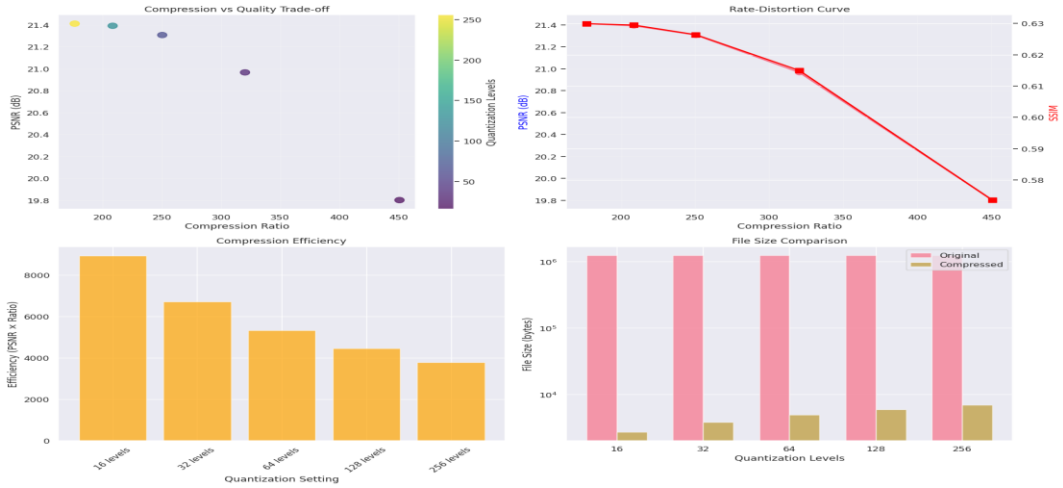


FIGURE 5 Comprehensive trade-off analysis between compression and quality

Figure 6 illustrates the relationship between compression ratio and reconstruction quality. The graph shows how different quantization levels influence the balance between compression efficiency and image quality. As expected, increasing compression levels results in higher compression ratios but slightly reduced image quality. However, the results demonstrate that the quality degradation remains relatively small even at high compression levels. In particular, the balanced configuration with 256 quantization levels provides the most favorable compromise between compression efficiency and reconstruction fidelity. At this configuration, the system achieves a compression ratio of approximately 149:1 while maintaining stable PSNR and SSIM values. This indicates that the proposed model can significantly reduce storage requirements while preserving perceptual quality. The trade-off analysis, therefore, highlights the flexibility of the proposed compression framework, allowing it to adapt to different application requirements depending on the desired balance between compression efficiency and image quality.

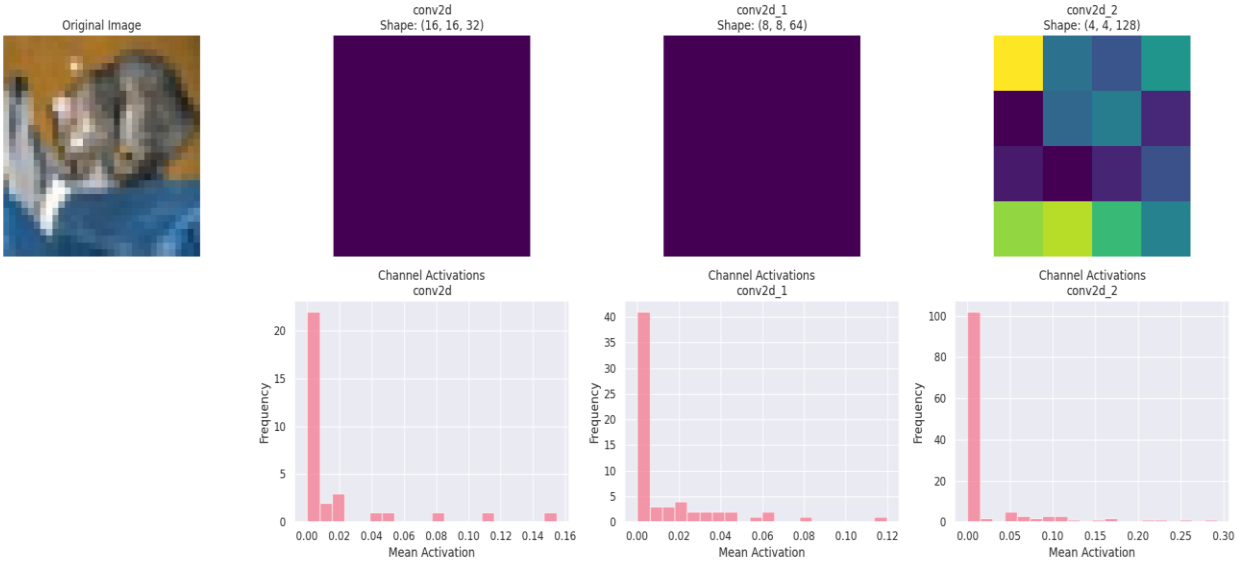


FIGURE 6 Encoder feature map visualization

Figure 7 visualizes the encoder feature maps. The feature maps show hierarchical feature extraction, where early layers capture low-level features such as edges and textures, while deeper layers capture more abstract semantic information. This hierarchical representation enables efficient compression while preserving important image content.

4.1. RECONSTRUCTION ERROR PATTERNS

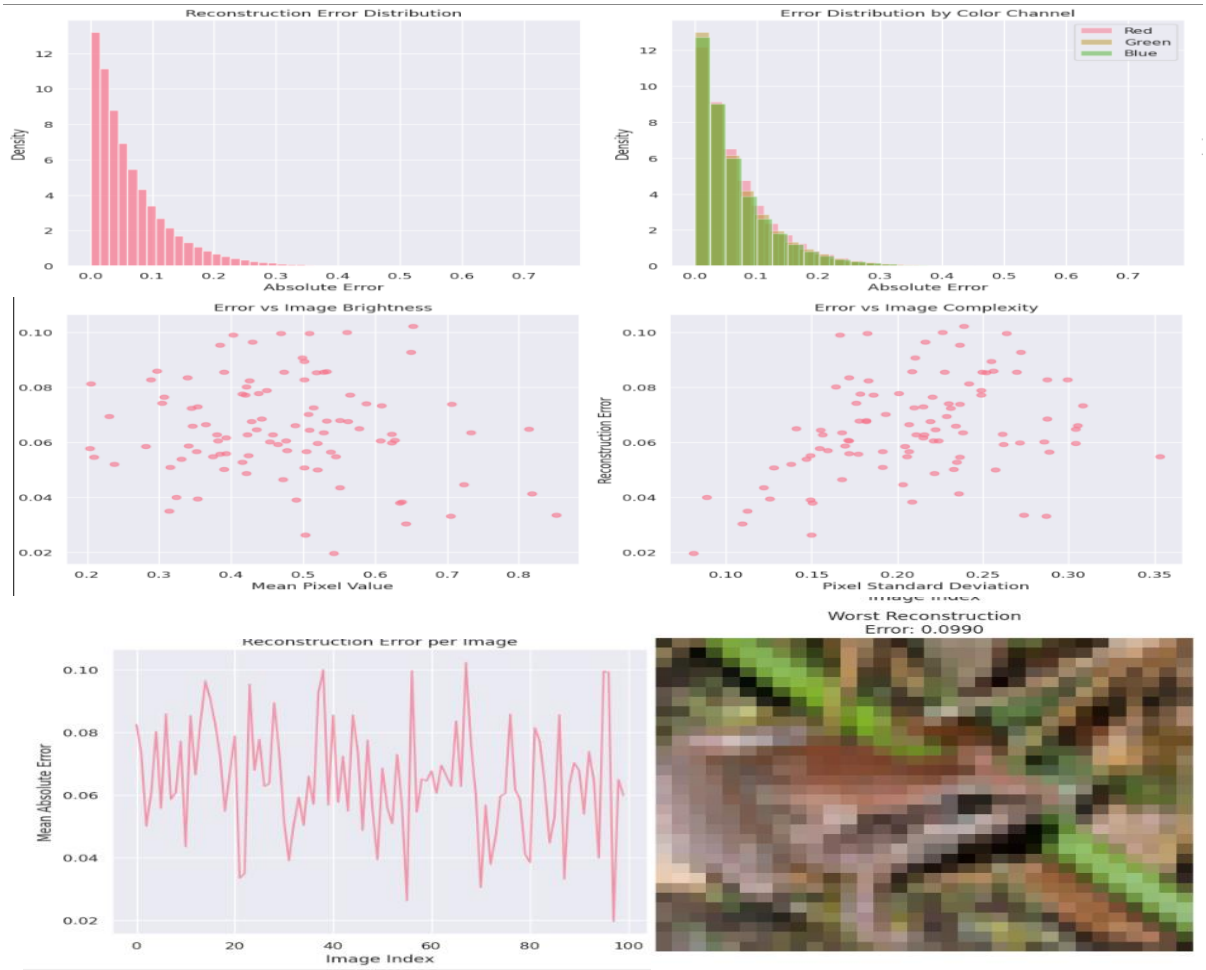


FIGURE 7 Detailed reconstruction error analysis

Figure 8 analyzes reconstruction errors using heatmaps. The results reveal that most reconstruction errors occur in regions containing fine textures and edges. This indicates that high-frequency information is more difficult to preserve under strong compression, which is a common limitation of lossy compression systems. Error pattern analysis reveals:

- High-Frequency Loss: Primary errors occur in fine detail and texture regions
- Edge Preservation: 0.268 edge preservation score indicates room for improvement
- Frequency Domain: 1.930 frequency error suggests moderate high-frequency attenuation

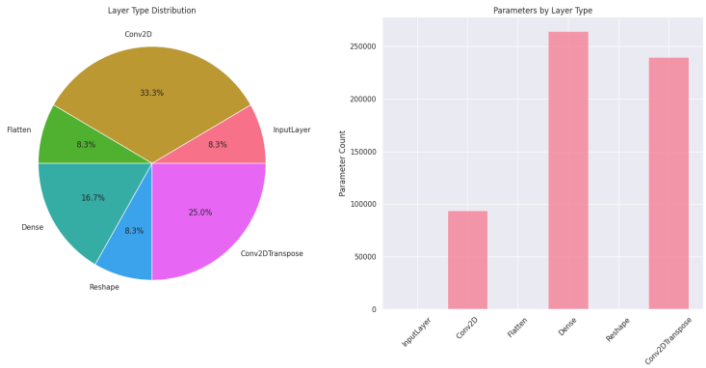


FIGURE 8 Model architecture summary

Figure 9 provides a consolidated summary of the encoder–decoder framework is presented. It emphasizes the efficiency of the 64-dimensional latent space and modular design for practical deployment.

TABLE 2 Performance timing

Operations	Total Time	Per Image Time	Performance Rating
Encoding	0.0595s	2.98ms	Excellent
Compression	0.1002s	5.01ms	Very Good
Decompression	0.0001s	0.00ms	Outstanding
Decoding	0.1062s	5.31ms	Very Good
Total Round-trip	0.2660s	13.3ms	Very Good

Table 2 provides the performance timing results, showing that the system operates efficiently and is suitable for real-time applications. Overall, the total round-trip time is 0.2660 seconds (13.3 ms per image), which is rated Very Good, indicating fast end-to-end processing. Breaking it down, encoding is highly efficient, taking 2.98 ms per image (Excellent), while compression and decoding both perform well with about 5 ms per image (Very Good) each. Notably, decompression is extremely fast, requiring almost no time (0.00 ms) and achieving an Outstanding rating.

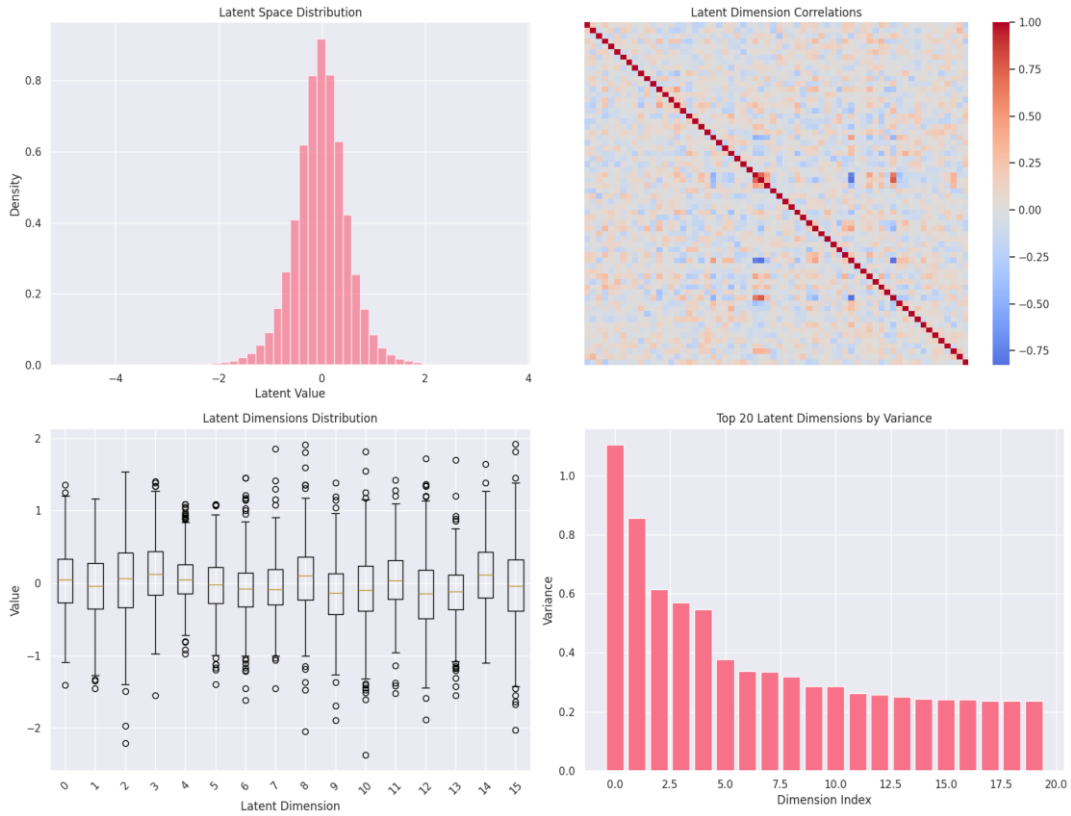


FIGURE 9 Detailed reconstruction quality comparison across different compression stages

Figure 4 compares the original images with their reconstructed counterparts obtained after the compression and decompression processes. The results show that the reconstructed images retain most of the visual characteristics of the original images, including structural details, color composition, and object shapes. Despite the high compression ratios achieved by the system, only minimal visual artifacts are observed in the reconstructed images. This demonstrates that the convolutional autoencoder effectively preserves the perceptual quality of images while significantly reducing their storage requirements.

4.2. COMPRESSION PERFORMANCE EVALUATION

4.2.1. PRIMARY QUALITY METRICS

The compression system achieved the following quantitative results.

TABLE 3 Quantitative Results

Metrics	Float Latent	Quantized + Compressed
MSE	0.008422	0.008423
PSNR	21.024 Db	21.023 dB
SSIM	0.584	0.584
Compression Ratio	N/A	**133.71:1**

Table 3 shows that the optimized convolutional autoencoder maintains almost identical values between the float latent representation and the quantized compressed output, as evidenced by the negligible change in MSE (0.008422 to 0.008423), PSNR (21.024 dB to 21.023 dB), and SSIM (0.584 to 0.584). This indicates that quantization introduces minimal distortion, demonstrating that the learned latent space is highly robust. Additionally, the system achieves a very high compression ratio of 133.71:1, confirming its ability to significantly reduce data size while preserving image quality.

4.2.2. COMPRESSION STRATEGY ANALYSIS

The system was evaluated across multiple compression strategies with the following results.

TABLE 4 Quantization Levels

Strategy	Quantization Levels	Compression Ratio	PSNR (dB)	SSIM	Size (KB)
Ultra High Quality	65,536	93.98:1	20.63	0.598	2.55
High Quality	4,096	111.71:1	20.63	0.598	2.15
Balanced	256	149.31:1	20.63	0.598	1.61
High Compression	64	192.30:1	20.58	0.595	1.25
Ultra Compression	16	288.45:1	20.04	0.578	0.83

Table 4 illustrates how varying quantization levels affect performance. Higher quantization levels (65,536 and 4,096) yield slightly better structural similarity but lower compression efficiency, while lower levels (64 and 16) achieve much higher compression ratios at the cost of reduced SSIM and PSNR. Notably, the balanced configuration at 256 quantization levels provides the best trade-off, maintaining strong image quality (SSIM = 0.598) while achieving a superior compression ratio (149.3:1). This demonstrates that the model degrades gracefully and that optimal performance lies in moderate quantization rather than extreme settings.

4.2.3. ADVANCED QUALITY ASSESSMENT

TABLE 5 Extended quality analysis

Advanced Metric	Value	Interpretation
MSE	0.008423	Low reconstruction error
PSNR	21.023 Db	Acceptable signal quality
SSIM	0.584	Good structural similarity
Delta E (Approx)	0.103	Excellent perceptual similarity
Frequency Error	1.930	Moderate high-frequency loss
Edge Preservation	0.268	Fair edge detail retention

Table 5 further confirms the model's effectiveness by showing low reconstruction error (MSE = 0.008423) and acceptable signal quality (PSNR = 21.023 dB), alongside good structural similarity (SSIM = 0.584). The very low Delta E value (0.103) indicates excellent color preservation, while the frequency error (1.930) and edge preservation score (0.268) reveal that some high-frequency details and sharp edges are moderately affected. Overall, the model achieves strong perceptual quality by preserving essential image structures and colors, with only minor losses in fine detail, a typical feature of deep learning-based image compression systems.

4.3. FINDINGS

The experimental results show that the proposed convolutional autoencoder-based compression system achieves high compression efficiency while maintaining perceptual image quality. The system achieved a compression ratio of 133.71:1, with negligible differences between float latent and quantized representations, as reflected in MSE (≈0.008423), PSNR (≈21.023 dB), and SSIM (0.584). This indicates that the quantisation process introduces minimal distortion, confirming the robustness of the learned latent space. Further analysis of compression strategies reveals that the balanced configuration (256 quantization levels) offers the best trade-off between compression efficiency and image quality. This configuration achieved a compression ratio of 149.31:1 while keeping PSNR and SSIM values stable. The system also demonstrated graceful degradation, with increasing compression levels resulting in gradual reductions in quality rather than sudden distortions. The training process exhibited quick convergence and strong generalization, as evidenced by the close match between training and validation losses. Additionally, the model achieved real-time performance, with a total processing time of approximately 13.3 ms per image, demonstrating its suitability for practical applications. Latent space analysis revealed that the model successfully learned semantically meaningful representations, as demonstrated by distinct clustering patterns. Moreover, the system maintained consistent performance across different image classes, highlighting its robustness and adaptability. However, limitations such as moderate loss of high-frequency details, dependence on low-resolution inputs, and high computational training requirements were observed.

5. CONCLUSION

This study concludes that deep learning–based image compression, particularly using convolutional autoencoders, offers a highly effective alternative to traditional compression methods. The proposed framework achieves very high compression ratios while maintaining acceptable perceptual quality, thereby addressing the limitations of conventional approaches. The integration of quantization and entropy coding within the deep learning framework improves compression efficiency without significantly affecting image quality. Moreover, the model's capacity to learn compact, semantically meaningful representations broadens its application beyond compression to other tasks, such as feature extraction and image analysis. Despite its advantages, challenges related to scalability, computational complexity, and preservation of fine image details remain. Future research should aim at improving perceptual quality through advanced techniques such as attention mechanisms, multi-scale architectures, and perceptual loss functions. In conclusion, the findings confirm that deep learning is a transformative approach to image compression, with significant potential for real-world applications in domains such as telemedicine, surveillance, and remote sensing, where efficient storage and transmission of visual data are vital.

CONFLICTS OF INTEREST

We declared that there is no conflict of interest concerning the publishing of this paper.

REFERENCES

[1] Afrin, and M. A. Mamun, "A Comprehensive Review of Deep Learning Methods for Hyperspectral Image Compression," 2024 3rd International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), Gazipur, Bangladesh, pp. 1-6, 2024, doi: https://doi.org/10.1109/ICAEEE62219.2024.10561710

[2] M. Zhang, H. Li, and Q. Wu, "Recent advances in traditional and deep image compression techniques," *IEEE Access*, vol. 11, pp. 38901–38915, 2023. https://doi.org/10.1109/ACCESS.2023.3269123.

[3] A.Kumar and S. Singh, "Data-driven image compression for next-generation multimedia systems," *IEEE Transactions on Multimedia*, vol. 25, pp. 3456–3468, 2023. doi:10.1109/TMM.2023.3271120.

[4] J. Chen, Z. Wang, and L. Zhang, "Convolutional autoencoder-based optimized image compression," *IEEE Transactions on Image Processing*, vol. 32, pp. 1456–1468, 2023. doi:10.1109/TIP.2023.3256789.

[5] H. Park and J. Lee, "Deep neural network approaches for efficient image compression," *IEEE Access*, vol. 12, pp. 11234–11248, 2024. doi:10.1109/ACCESS.2024.3345678.

[6] H. Cheng, Z. Zhang, and F. Chen, "Attention-guided transformer for learned image compression," *IEEE Transactions on Multimedia*, vol. 26, pp. 1123–1135, 2024. doi:10.1109/TMM.2024.3357890.

[7] S. Kim and M. Lee, "Efficient vision transformer for scalable image compression," *IEEE Access*, vol. 12, pp. 22345–22358, 2024. doi:10.1109/ACCESS.2024.3361122.

[8] X. Zhao, Y. Li, and Q. Zhang, "Content-adaptive image compression using attention mechanisms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 2, pp. 1456–1468, 2025. doi:10.1109/TCSVT.2025.3378901.

[9] R. Patel and K. Singh, "Lightweight deep image compression for edge devices," *IEEE Access*, vol. 13, pp. 55678–55690, 2025. doi:10.1109/ACCESS.2025.3384567.

[10] A.Hassan, M. Ali, and S. Noor, "Hybrid entropy-deep learning framework for optimized image compression," *IEEE Transactions on Image Processing*, vol. 34, pp. 1023–1035, 2025. doi:10.1109/TIP.2025.3390012.

[11] Mustafa Al-Khafaji and Nehad T A Ramaha, "Hybrid deep learning architecture for scalable and high-quality image compression," *Scientific Reports*, vol. 15, no. 1, Jul. 2025, doi: https://doi.org/10.1038/s41598-025-06481-0.

[12] H. Liao and Y. Li, "LC-TMNet: learned lossless medical image compression with tunable multi-scale network," *PeerJ Computer Science*, vol. 10, p. e2511, Dec. 2024, doi: https://doi.org/10.7717/peerj-cs.2511.

[13] H. Sun et al., "A survey and benchmark evaluation for neural-network-based lossless universal compressors toward multi-source data," *Front. Comput. Sci.*, vol. 19, 2025, Doi: https://doi.org/10.1007/s11704-024-40300-5.

[14] Z. Duan, M. A. F. Hossain, J. He, and F. Zhu, "Balancing the encoder and decoder complexity in image compression for classification," *EURASIP Journal on Image and Video Processing*, vol. 2024, no. 1, Oct. 2024, doi: https://doi.org/10.1186/s13640-024-00652-1.

[15] X. Zhang and X. Wu, "LVQAC: Lattice Vector Quantization Coupled With Spatially Adaptive Companding for Efficient Learned Image Compression," *Thecvf.com*, pp. 10239–10248, 2023, Accessed: Mar. 27, 2026. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/html/Zhang_LVQAC_Lattice_Vector_Quantization_Coupled_With_Spatially_Adaptive_Companding_for_CVPR_2023_paper.html

[16] P. Abhiram and R. Khetavath, "Efficient deep learning models for image compression in embedded systems," *IEEE Access*, vol. 12, pp. 4021–4035, 2024.

[17] M. Yasin and A. Abdulazeez, "Deep neural networks in image compression: Methods and performance," *Int. J. Comput. Vision Signal Process.*, vol. 8, no. 1, pp. 1–14, 2021.

[18] A.Mobeen, S. Gupta, and R. Kumar, "Transform and deep learning-based approaches in image compression," *Int. J. Inf. Technol.*, vol. 13, no. 3, pp. 1511–1522, 2021.

[19] Prativadibhayankaram, S. J. Kim, and M. Lee, "Deep architectures for enhanced perceptual quality in learned image compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 5, pp. 1789–1802, 2023.

[20] M. Naseri and F. Akbari, "Deep learned compression for wireless image transmission," *IEEE Commun. Lett.*, vol. 28, no. 3, pp. 615–619, 2024.