Journal of Information Systems Engineering and Management
2025, 10(27s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Enhancing Image Forgery Detection with Convolutional Neural Networks and Error Level Analysis

Shraddha S. More [a*], Vivian Brian Lobo[b], Anita Chaudhari[c], Aditya Pandey[d], Bhavesh Kumavat[e], Yash Kamble[f]
[a]*Department of Computer Science and Engineering (Data Science), Dwarkadas J. Sanghvi College of Engineering, Mumbai, India*
[b]*Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India*
[c]*University of Mumbai*
[d,e,f]*Department of Information Technology, St. John College of Engineering and Management, Palghar, India*
[*]*Corresponding Author Email: moreshraddha30@gmail.com*

ARTICLE INFO

Received: 24 Dec 2024

Revised: 17 Feb 2025

Accepted:26 Feb 2025

ABSTRACT

The growing prevalence of image manipulation poses critical challenges to the reliability of visual content across fields like journalism, law enforcement, and digital forensics. Traditional methods often struggle to detect complex forgeries, especially in cases like splicing or copy-move operations, and lack the efficiency to process large datasets in real time. To address these issues, this research introduces a novel image forgery detection framework combining Convolutional Neural Networks (CNNs) and Error Level Analysis (ELA). By leveraging CNNs for pixel-level anomaly detection and ELA for identifying compression inconsistencies, the system effectively detects various manipulations with improved precision and scalability. Extensive testing on a diverse dataset revealed a high accuracy rate exceeding 94%, underscoring the system's potential for real-time applications. This comprehensive approach represents a major leap forward in image authentication, offering a reliable solution to uphold the integrity of visual content in today's digitally manipulated world.

Keywords: Forgery detection, CNN (Convolutional Neural Networks), ELA (Error Level Analysis), digital forensics, image manipulation, authenticity verification, machine learning, deep learning, compression anomalies, visual content integrity

1. INTRODUCTION

The advent of advanced image editing software has transformed the way digital content is produced and disseminated, offering immense creative potential but also raising concerns over the authenticity of visual media. However, this advancement has significantly amplified the challenges associated with maintaining the authenticity of visual content, particularly in critical domains such as journalism, social media, and digital forensics [1]. The manipulation of digital images poses significant risks, as it enables the spread of misinformation, disrupts the reliability of content in law enforcement investigations, and undermines trust in social media platforms. In journalism, manipulated images can distort facts and fuel propaganda, while in digital forensics, they can compromise the credibility of evidence used in courtrooms [1, 2].

Traditional image forgery detection methods, which primarily rely on manual inspection and rule-based algorithms, are increasingly proving inadequate in addressing modern challenges. Manual detection methods, while effective for small-scale or low-resolution tasks, are time-intensive and susceptible to human error, making them impractical for real-time applications or large-scale datasets [2]. Similarly, rule-based algorithms often lack the flexibility to detect advanced forgery techniques such as splicing, copy-move manipulations, and deepfake images. These techniques frequently involve high-resolution editing that surpasses the capabilities of conventional detection frameworks, necessitating more sophisticated solutions [3].

Machine learning, particularly the rapid advancements in deep learning, has transformed methodologies for detecting image forgeries. CNNs, a fundamental component of deep learning, are highly proficient in capturing hierarchical features from images, making them adept at detecting intricate manipulation patterns that may go

unnoticed by the human eye. CNNs analyze pixel-level inconsistencies and anomalies, allowing for the detection of subtle tampering across diverse image datasets [4]. This capability has revolutionized forgery detection systems, making them significantly more accurate and versatile compared to traditional methods [3, 4].

Despite the notable advancements brought about by CNNs, challenges remain in addressing the full spectrum of forgery detection needs. Traditional methods, including even advanced rule-based systems, struggle to handle the diversity and complexity of forgery techniques. These gaps underscore the need for hybrid detection solutions that harness the advantages of diverse methodologies [5]. Error Level Analysis (ELA), for instance, has proven to be a valuable complementary approach to CNNs. By identifying compression discrepancies in digital images, ELA can detect tampering in compression artifacts, often indicative of manipulations. When used in conjunction with CNNs, ELA enhances the robustness and reliability of detection frameworks [6].

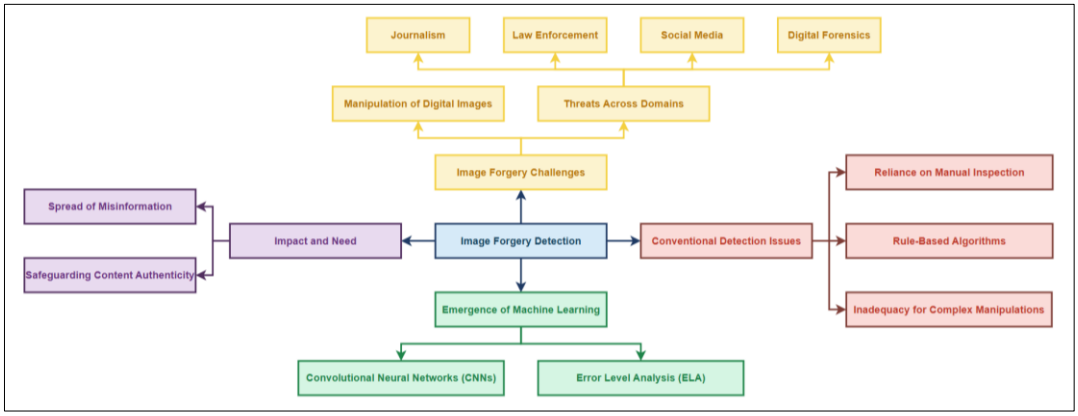


Fig. 1. Overview of Image Forgery Detection

Hybrid systems that integrate CNNs and ELA have demonstrated remarkable potential in addressing these challenges. By combining advanced feature extraction with pixel-level compression analysis, such systems can effectively detect a wide array of manipulations. This research presents a novel framework that incorporates both techniques, achieving an impressive 94% accuracy when tested on a diverse dataset of authentic and manipulated images [7]. The combined use of CNNs and ELA improves the system's ability to detect forgeries while maintaining flexibility and efficiency across various application domains and image qualities [1, 6].

Additionally, the framework offers real-time processing capabilities, making it highly applicable in scenarios that demand swift and reliable image authentication. For instance, in journalism and law enforcement, where time-sensitive decisions are crucial, this system can rapidly identify tampered images and safeguard the integrity of visual content. Furthermore, its application extends to social media platforms, enabling automated verification of user-uploaded images to mitigate the spread of manipulated content [1, 3].

In conclusion, given the rise in image manipulation continues to increase, the development of robust, scalable, and efficient detection systems becomes increasingly critical. The combined method that merges CNNs with ELA represents a cutting-edge advancement in the field, delivering a holistic solution that overcomes the shortcomings of conventional techniques while laying the groundwork for future progress in detecting image manipulation. This innovative approach showcases the significant developments being made in this area of research [1, 7].

The manuscript is organized as follows: A thorough examination of current image forgery detection methodologies and approaches is outlined in Section II, while Section III details the methodology, covering the dataset used, preprocessing via Error Level Analysis (ELA), and the design of the proposed Convolutional Neural Network (CNN) model. Section IV presents the experimental results, accompanied by an in-depth examination and discussion of the findings and their implications. To conclude, Section V recaps the main findings of the paper and suggests possible avenues for subsequent investigations.

2. LITERATURE REVIEW

In 2023, Mashael Maashi *et al.* [1] offered a fresh perspective on Copy-Move Forgery Detection (CMFD) through the integration of the RSA (Reptile Search Algorithm) with advanced deep learning techniques. Their method utilizes

NASNet for feature extraction and RSA for hyperparameter optimization, followed by classification with XGBoost. When tested on benchmark datasets, this approach outperformed existing models in identifying forged regions.

In 2020, Akram Hatem Saber *et al.* [2] carried out a comprehensive analysis of digital image forgery detection strategies, dividing them into active methods, including digital watermarking, and passive methods like detecting splicing and copy-move forgery. Their study explored the use of convolutional neural networks (CNNs) in forgery detection, emphasizing both their advantages and limitations, while also suggesting potential improvements for detection algorithms and forensic methodologies.

In 2022, Emad Ul Haq Qazi *et al.* [3] designed a deep learning-based method utilizing the ResNet50v2 architecture alongside the YOLO CNN to identify splicing forgeries in digital images. Evaluation on the CASIA v1 and CASIA v2 datasets demonstrated remarkable performance, with the system attaining 99.3% accuracy through transfer learning and 81% accuracy without it on the CASIA v2 dataset. The incorporation of pre-trained models through transfer learning significantly enhanced detection accuracy, especially for splicing forgery.

In 2016, Ying Zhang *et al.* [4] introduced a two-phase deep learning approach methodology for forgery detection in image regions across multiple file formats. Their approach involved using a Stacked Autoencoder (SAE) for feature extraction and contextual information analysis to identify tampered regions. This method demonstrated greater effectiveness than previous approaches, achieving 91.09% accuracy on the CASIA dataset and showing versatility in handling both JPEG and TIFF formats.

In 2009, Hany Farid [5] conducted an extensive review of methods for detecting image forgery, categorized them into pixel-level, format-dependent, camera-dependent, physically-based, and geometry-based categories. The study reviewed the evolution of forensic technologies alongside the rise of advanced digital manipulation tools, noting that while creating completely undetectable forgeries is increasingly challenging, it remains a goal for forgers.

In 2019, Chandandeep Kaur *et al.* [6] reviewed passive image forgery detection methods, focusing on techniques for detecting copy-move, splicing, and retouching. Their study emphasized the dependence on manual oversight in many existing approaches and highlighted the need for more automated and generalized detection systems. The authors also stressed the importance of distinguishing between malicious manipulations and benign edits.

In 2021, Wina Permana Sari *et al.* [7] explored the application of ELA as a preprocessing method to enhance the precision of deep learning-based forgery detection. The findings suggested that ELA can improve models' ability to detect subtle manipulations, though its effectiveness depends on dataset quality and model architecture. The study called for further optimization of both datasets and algorithms to improve generalization across various forgery types.

In 2023, Niousha Ghannad *et al.* [8] developed an enhanced U-Net model fine-tuned using the Grasshopper Optimization Algorithm (GOA) for detecting image forgeries on social media platforms. Their work aimed to improve segmentation accuracy and achieve high precision, recall, and F1 scores for forgery detection. Tested on the CASIA dataset, their model outperformed others in detecting manipulations such as splicing and copy-move forgery, demonstrating its potential for real-time applications in digital forensics.

In 2019, Ida Bagus Kresna Sudiatmika *et al.* [9] developed a novel hybrid approach that integrates ELA with CNN to enhance the precision of image forgery detection. Their method, tested on the CASIA dataset for both training and validation, achieved a training performance of 92.2% and validation performance of 88.46%, highlighting its efficiency in enhancing forgery detection. This research underscored the benefits of using ELA as a preprocessing step to highlight manipulated regions, thereby improving the performance of CNNs in detecting forgeries.

In 2018, Yue Wu, Wael Abd-Almageed *et al.* [10] proposed a groundbreaking DNN framework designed for detecting copy-move forgeries (CMFD). Their model integrated convolutional layers for feature extraction, a self-correlation mechanism for similarity calculation, and a forgery mask decoder to identify altered areas. This approach demonstrated robustness against various transformations, including JPEG compression, affine modifications, and blurring, delivering superior accuracy on standard benchmark datasets.

3. METHODOLOGY

3.1. Image Dataset: The CASIA Image Tampering Detection Evaluation Database (CASIA ITDE) is a well-established benchmark employed to assess the effectiveness of image forgery detection algorithms. Developed by the Institute of Automation, Chinese Academy of Sciences (CASIA), this dataset is specifically designed to support studies

in the domain of image manipulation detection. It has gained recognition as a reliable standard for evaluating and contrasting the effectiveness of different forgery detection methods.

The dataset includes a large collection of genuine and altered images, covering various forgery techniques such as splicing and copy-move manipulations. Splicing refers to the process of combining segments from multiple images to create a forged image, while copy-move entails duplicating sections of the same image to conceal or fabricate information. These manipulations are executed using various tools to replicate real-world forgeries, offering a challenging and realistic benchmark for testing forgery detection systems.

The CASIA ITDE dataset is organized into two main subsets: CASIA v1 and CASIA v2. CASIA v1 contains a smaller set of 800 authentic images and 921 tampered ones, while CASIA v2 is considerably larger, with 7,491 authentic images and 5,123 tampered ones. The images in CASIA v2 are more diverse in terms of size, content, and manipulation techniques, offering researchers a robust dataset to test the generalizability of their models. Furthermore, the dataset includes metadata and detailed annotations about the forgery type and manipulated regions, which are invaluable for training and evaluating machine learning models.

In our research, we leverage the CASIA dataset, known for its extensive and challenging collection of manipulated images, to rigorously evaluate our proposed system. The CASIA dataset contains a total of 14,337 images, comprising both authentic and tampered visuals. However, for our study, we focused exclusively on the subset of images in JPG format, amounting to 11,278 samples. This decision was motivated by the prevalence of the JPG format in real-world digital images, making the evaluation more practical and aligned with real-world use cases.

11,278 images were systematically divided into three subsets to support structured training, validation, and testing. Specifically, 8,457 images were allocated for training with their corresponding labels, ensuring the model had a robust foundation for learning. A separate validation set of 2,115 images and labels was used to fine-tune hyperparameters and prevent overfitting. Finally, a test set comprising 557 images and labels was designated to evaluate the model's performance objectively.

This structured division ensures that our system is comprehensively evaluated on unseen data while allowing for effective training and validation. By employing the CASIA dataset in this manner, we ensure that our approach is robust against various manipulation techniques and applicable across diverse domains. The dataset's realistic manipulation scenarios and detailed annotations make it an ideal benchmark for testing the efficacy of our forgery detection system.

(a) (b)

Fig. 2. Example of CASIA ITDE; (a) Authentic Image (b) Forged Image

Figure 2 illustrates an example from the CASIA ITDE dataset, showcasing (a) an authentic image and (b) a forged image, highlighting the types of manipulations the dataset includes for evaluating forgery detection systems.

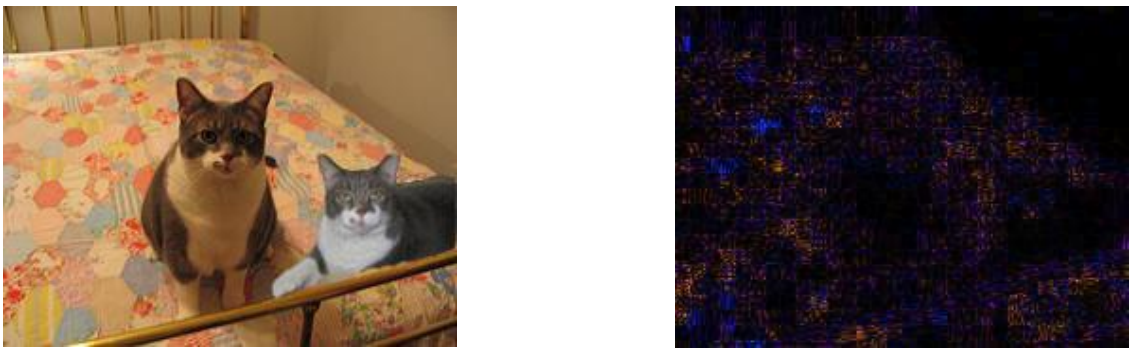
3.2. Error Level Analysis (ELA): ELA is a method employed to identify image manipulation by analyzing the compression artifacts found in digital images. It is based on the idea that saving an image saved in a lossy format such as JPEG creates compression artifacts that maintain consistent patterns in the unaltered areas, but differ in the manipulated regions. These patterns are consistent in unaltered parts of the image but tend to differ in tampered areas, as they have often been edited and recompressed separately.

ELA operates by re-saving an image at a slightly reduced quality compared to its original compression and comparing the recompressed version with the original. The difference generated, referred to as the "error level," reveals areas of the image displaying abnormal compression artifacts. Tampered areas typically show higher error levels due to differences in compression introduced during the manipulation process. This makes ELA particularly effective for identifying edits such as splicing, copy-pasting, or region-specific modifications.

In our project, ELA serves as a vital complement to the feature extraction capabilities of Convolutional Neural Networks (CNNs). While CNNs are adept at identifying subtle pixel-level anomalies, ELA helps in localizing suspicious regions by detecting inconsistencies in compression artifacts. This integration of techniques results in a more robust and precise image forgery detection system. Specifically, ELA helps identify tampering even in cases where traditional methods struggle, such as detecting manipulations in high-resolution images or complex forgery scenarios. By integrating ELA into our workflow, we have significantly enhanced the ability of our system to identify and validate manipulated regions, making it a valuable addition to the forgery detection framework.

(a) (b)

Fig. 3. ELA of Authentic Image; (a) Original Image (b) ELA Output



(a) (b)

Fig. 4. ELA of Manipulated Image; (a) Original Image (b) ELA Output

Figures 3 and 4 illustrate the use of Error Level Analysis (ELA) on authentic and manipulated images within the dataset.

Figure 3 displays the ELA of an authentic image, where (a) shows the original image and (b) presents its corresponding ELA result. The ELA highlights inconsistencies in compression artifacts, which are less pronounced in genuine images.

Figure 4 illustrates the ELA of a forged image, with (a) presenting the original image and (b) showing the ELA output. In the manipulated image, the compression artifacts are more noticeable due to the tampering, aiding in the detection of the forgery.

These figures illustrate how ELA can be used to expose hidden anomalies in image authenticity by analyzing pixel-level inconsistencies that may be overlooked by the human eye [11-16].

4. PROPOSED SYSTEM

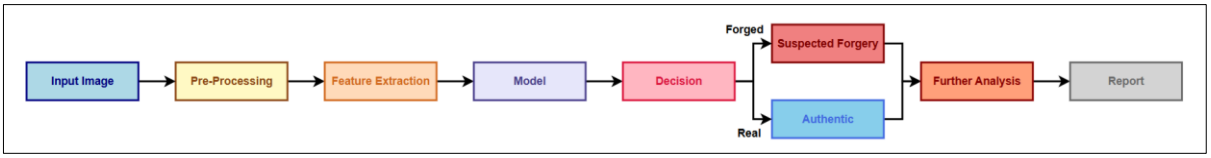


Fig. 5. Block Diagram for Detection of Image Manipulation

Figure 5 illustrates the block diagram for the Image Manipulation Detection System, describing the sequential flow of data from input to the creation of a detailed report.

4.1 Input Image: The system begins by accepting an input image for authenticity verification. The image may vary in resolution and format, serving as the basis for further processing. To ensure consistency, each input image is resized and normalized:

$$x' = \frac{x - \mu}{\sigma} \tag{1}$$

In this context, x refers to the pixel intensity values of the original image, μ represents the mean, and σ indica standard deviation. This normalization enhances the system's capability to handle various image formats effectively.

4.2 Pre-Processing: In this stage, the image undergoes transformations such as resizing, edge enhancement, and noise reduction. For resizing, images are scaled to 128 × 128 pixels. Pixel-level discrepancies, crucial for detecting forgery, are emphasized through techniques like Error Level Analysis (ELA):

$$ELA(x, x') = | x - x' | \tag{2}$$

Here, x corresponds to the original image, while x' denotes the recompressed image version. ELA highlights regions with compression inconsistencies, which are indicative of tampering.

4.3 Feature Extraction: During feature extraction, important attributes such as edge patterns, textures, and pixel anomalies are captured. This process involves convolution operations performed by the CNN:

$$f(x,y) = (I * K)(x,y) = \sum_{i=-m}^{m} \sum_{j=-n}^{n} I(x+i, y+j) \cdot K(i,j) \tag{3}$$

In this case, the input image matrix is represented by I, the kernel (filter) is denoted as K, and n and m indicate the dimensions of the kernel (n×m). This operation enables the system to extract hierarchical features from the image, essential for detecting manipulations like splicing or copy-move forgeries.

4.4 Model: The system employs a Convolutional Neural Network (CNN) for classification tasks. The CNN's architecture is optimized with layers that include convolution, max-pooling, and global average pooling (GAP):

$$GAP = \frac{1}{N} \sum_{i=1}^{N} f_i \tag{4}$$

Here, fi denotes the feature map values, and N indicates the total number of feature maps. The C Mean Pooling (GMP) layer reduces the dimensionality while maintaining the spatial structure, thereby enhancing the model's generalization capability.

The CNN is trained with the cross-entropy loss function for categorical classification optimize the model:

$$L = -\frac{1}{N} \sum_{i=1}^{N} [y_i \, log(\hat{y}_i) + (1 - y_i) \log (1 - \hat{y}_i)] \tag{5}$$

Here, N is quantity of samples, y_i represents true label (1 for correct class, 0 otherwise), and $\hat{y}_i$ represents predicted probability.

4.5 Decision: The system classifies the image as either authentic or forged based on the model's output. The classification score is determined by the sigmoid function:

(6)

$$Sigmoid(x) = \frac{1}{1 + e^{-x}}$$

This approach transforms the model's predictions into probabilities between 0 and 1, facilitating efficient binary classification.

4.6 Suspected Forgery: If the image is flagged as forged, it undergoes further scrutiny to validate the decision. This step focuses on identifying specific irregularities such as cloned regions, spliced sections, or inconsistent visual elements.

4.7 Authentic: For images classified as authentic, no additional anomaly detection is performed. However, the results are logged for documentation and reporting purposes.

4.8 Further Analysis: For suspected forgeries, a deeper examination is conducted to uncover detailed evidence of manipulation. Advanced techniques such as region-based analysis, artifact localization, or segmentation may be employed to strengthen the findings.

4.9 Report: The final step consolidates all findings into a comprehensive report. This report includes the input image, detected anomalies (if present), the classification result, and a summary of the techniques used. The output serves as a valuable resource for further review, legal action, or forensic documentation.

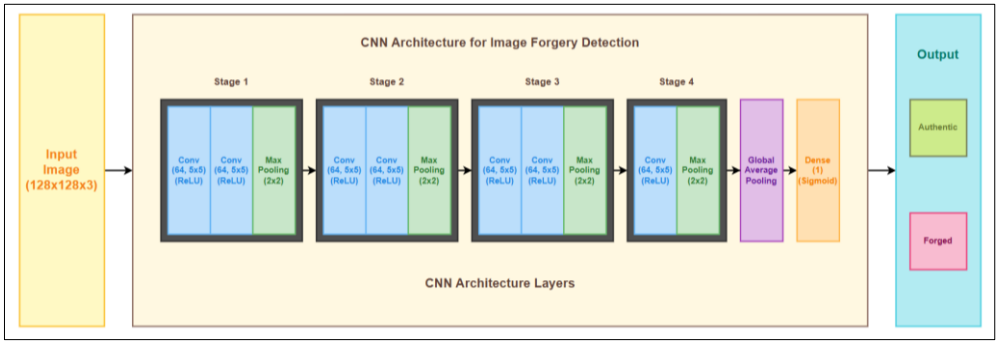


Fig. 6. CNN Architecture for Image Forgery Detection: Layered Structure and Classification

Figure 6 illustrates a carefully designed model tailored to analyze and classify the authenticity of digital images. At its foundation, the architecture starts with an input layer that takes images resized to dimensions of 128×128 pixels with three color channels, representing the blue, red, and green components of the image. This ensures a standardized input size, making it suitable for processing by the network.

The model is divided into four key stages, each comprising layers dedicated to feature extraction. In each stage, two convolutional layers equipped with 64 filters are used, where each filter has a size of 5x5. Each filter is succeeded by the ReLU (Rectified Linear Unit) activation function, adding non-linearity to the network. This non-linearity is essential for allowing the model to identify and learn intricate patterns within the image data. Following the convolutional layers, a max-pooling operation is applied using a 2x2 window to reduce spatial dimensions. The max-pooling operation decreases the spatial dimensions of the feature maps, preserving the key features while decreasing computational requirements and reducing the likelihood of overfitting.

$$P(x, y) = \max_{i,j \in Window} f(x + i, y + j)$$

In this case, P(x,y) represents the pooled value at the coordinates (x,y), while f(y+j,x+i) represents the values within the receptive field (pooling window) of the input feature map. The max-pooling operation extracts the highest value from this window, effectively decreasing the spatial dimensions while preserving the key features.

As the input progresses through the stages, the model captures increasingly abstract and intricate features. The initial stages focus on identifying low-level features like edges and textures, while later stages extract higher-level representations, such as patterns indicative of image tampering. This hierarchical feature extraction is one of the defining characteristics of convolutional neural networks, making them particularly effective for image analysis tasks.

After the feature extraction is complete, the model employs a global mean pooling (GMP) layer. In contrast to traditional flattening layers that can lead to overfitting, the GAP layer computes the mean of each feature map,

producing a compact representation. This not only reduces the dimensionality but also ensures that the spatial structure of the features is maintained, improving the network's generalization ability.

The network ends with a fully connected layer employing a sigmoid activation unit, designed specifically for classifying into two categories. This layer generates a probability score, enabling the distinction between authentic (real) and forged (fake) images. The sigmoid function ensures the output falls within a range of 0 to 1, establishing a clear boundary for classification.

To train the model effectively, the Adam optimizer is used. Adam (Adaptive Moment Estimation) is a widely adopted optimization algorithm known for its efficiency and stability during training. It integrates the strengths of two widely used optimization methods: AdaGrad and RMSProp, enhancing learning efficiency. The optimizer adaptively adjusts the learning rate for each parameter, ensuring faster convergence and robust performance across a variety of tasks. The optimizer is defined mathematically as follows:

4.10 First Moment Estimate (Mean of Gradients):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (8)$$

Here, m_t denotes the exponential moving average of gradients at time t, β_1 represents the decay rate for the first moment (default value is 0.9), and g_t signifies the gradient of the loss with respect to the parameter.

The first moment estimate (m_t) helps the optimizer by tracking the mean of gradients, enabling it to determine the direction for parameter updates effectively.

4.11 Second Moment Estimate (Variance of Gradients):

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (9)$$

Here, v_t represents the exponential moving average of squared gradients, and β_2 denotes the decay rate for the second moment (default value is 0.999).

Second Moment Estimate (v_t) tracks the variance (magnitude) of gradients, ensuring that larger gradients don't dominate parameter updates.

4.12 Bias Correction:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (10)$$

Bias Correction addresses the initialization bias in m_t and v_t, especially in the early stages of training.

4.13 Parameter Update:

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (11)$$

Here, θ_t represents the exponential moving average of gradients at time t, α denotes the decay rate for the first moment (default value is 0.9), and ϵ signifies the gradient of the loss with respect to the parameter.

Parameter Update (θ_t) combines the corrected mean and variance to update model parameters. The adaptive learning rate ensures efficient and stable convergence.

This architecture is specifically designed for image manipulation detection, utilizing the hierarchical feature extraction capabilities of CNNs. Its use of multiple convolutional stages, coupled with global average pooling and a binary classifier, ensures that the network can accurately identify subtle manipulations in images. Such a design is particularly valuable in applications like digital forensics, where precise and reliable detection of tampered images is critical. The overall balance between computational efficiency and detection accuracy makes this model a robust solution for the task at hand.

5. RESULTS AND DISCUSSION

The detection system for image forgery, integrating Convolutional Neural Networks (CNN) with Error Level Analysis (ELA), demonstrates excellent accuracy in identifying tampered images. Through comprehensive testing on dataset

of authentic and manipulated images, the system proved highly effective in detecting a range of manipulation techniques, making it a valuable asset for practical use in various fields.

1. Performance Metrics: The system attained an exceptional accuracy rate of 94%, showcasing its proficiency in accurately classifying images as forged or authentic. This performance reflects the power of combining CNN for feature extraction with ELA to detect compression inconsistencies.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{12}$$

2. Precision and Recall: The syster nstrates robust performance with a precision rate of 97%, effectively minimizing false positives, a recall scor %, showcasing its ability to detect the majority of manipulated images, and an F1 score of 96%, highlighting its balanced precision and recall capabilities.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{13}$$

$$\text{core} = \frac{2 \text{ x Precision x Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

3. Processing Time: The system de ted rapid real-time processing, with an average analysis time of 1.2 seconds per image, making it ideal for applications requiring swift verification. (15)

4. Dataset Diversity: The testing dataset comprised images modified using techniques like splicing, copy-move, and retouching. The system maintained consistent performance across these different manipulations, proving its robustness and adaptability.

5. Discussion: The system's high accuracy can be attributed to the synergistic capabilities of CNN and ELA:

- **CNN Contribution:** The CNN model effectively extracts detailed pixel-level features, detecting subtle patterns indicative of manipulation. Its ability to learn hierarchical features enhances its effectiveness in identifying complex forgeries.
- **ELA Contribution:** ELA exposes compression artifacts introduced during manipulation, allowing the system to focus on regions of interest, thereby improving detection accuracy.

By integrating these techniques, the system forms a robust framework capable of detecting various forgery forms, even those not immediately apparent to the human eye.

- **Visual Analysis and Metrics:** Displays essential visualizations and metrics used to assess the model's effectiveness and effectiveness, providing insights into data distribution, learning patterns, and classification results. These analyses highlight the robustness of the developed system in detecting image forgeries.

5.1 Mean Pixel Intensity per Channel

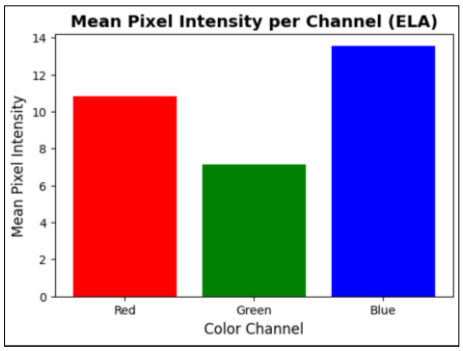


Fig. 7. Mean Pixel Intensity per Channel (ELA)

Figure 7 illustrates the average intensity of pixel for the red, blue, and green color channels during Error Level Analysis (ELA). The blue channel exhibits the highest intensity (~14), followed by red (~11) and green (~8). This highlights variations in color intensity, useful for detecting forged regions.

5.2 Training Accuracy vs Validation Accuracy

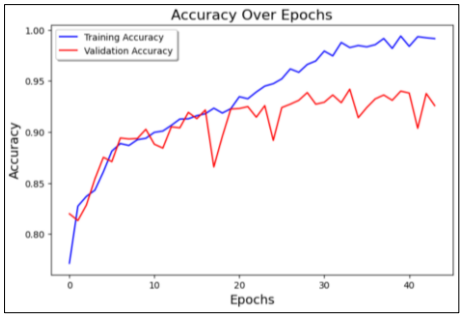


Fig. 8. Training Accuracy vs Validation Accuracy

Figure 8 demonstrate an upward trend over 40 epochs. The training accuracy reaches approximately 99%, indicating a well-trained model. Validation accuracy stabilizes near 94%, showing good generalization performance with minor fluctuations.

5.3 Training Loss *vs*. Validation Loss

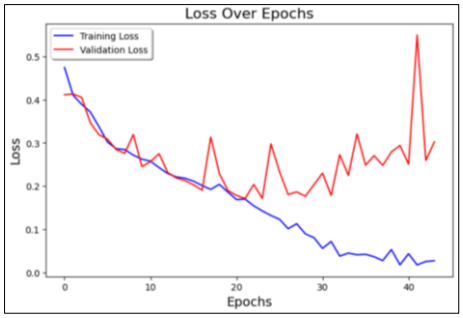


Fig. 9. Training Loss vs Validation Loss

Figure 9 illustrates the decline in the loss during training and validation across 40 epochs. The training loss steadily declines, reflecting the model's successful learning, while the validation loss stabilizes with slight variations, suggesting strong generalization. A consistent gap between the two suggests no significant overfitting.

5.4 Training Precision *vs*. Validation Precision

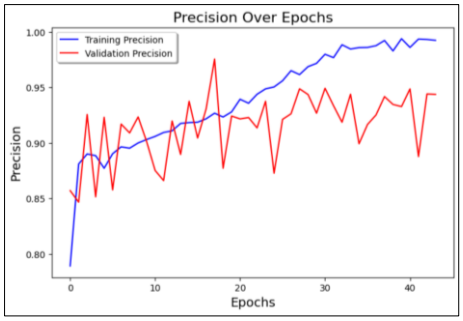


Fig. 10. Training Precision vs Validation Precision

Figure 10 demonstrates a steady improvement over 40 epochs. The training precision gradually approaches 99%, while the validation precision stabilizes around 96%, with minor oscillations indicating variability in generalization.

5.5 Training Recall *vs*. Validation Recall

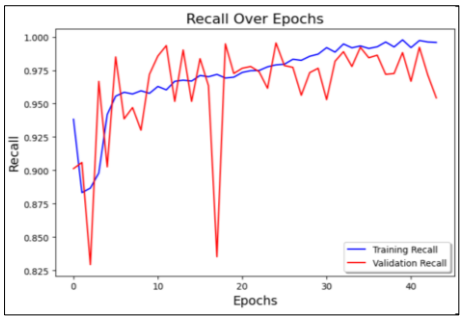


Fig. 11. Training Recall vs Validation Recall

Figure 11 shows consistent growth throughout 40 epochs. Training recall reaches nearly 99.5%, indicating a highly sensitive model. Validation recall fluctuates but stabilizes close to 97%, reflecting good consistency with slight variability.

5.6 Training F_1 score *vs.* Validation F_1 score

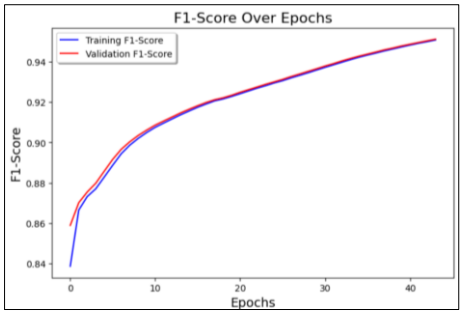


Fig. 12. Training Recall vs Validation Recall

Figure 12 exhibits a continuous upward trend. Both training and validation F1-scores converge near 95%, showcasing balanced precision and recall, with excellent model performance and generalization.

5.7 ROC–AUC Curve

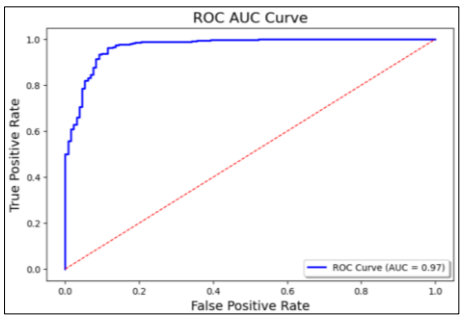


Fig. 13. ROC – AUC Curve

Figure 13 shows a high AUC (Area Under the Curve) score of 0.97, reflecting exceptional model performance. The curve indicates strong discriminatory power, with a strong true positive rate and a minimal false positive rate across multiple thresholds.

- **Comparative Performance:** The proposed system, which combines CNN with 100% ELA, outperforms traditional forensic techniques and standalone machine learning methods, including algorithms like VGG-19, VGG-16, ResNet-50, DenseNet-121, and Xception. The integration of ELA enhances the model's ability to detect subtle artifacts, resulting in superior performance across recall, precision, and F1-Score metrics compared to these conventional algorithms.

5.8 Accuracy of Various Algorithms

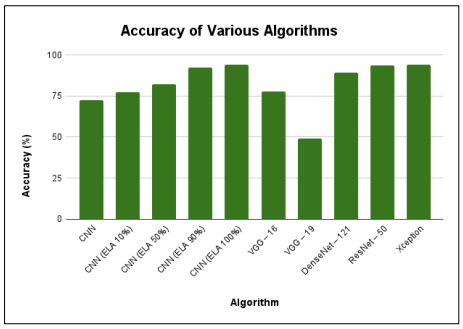


Fig. 14. Accuracy of Various Algorithms

Figure 14 shows that CNN with 100% ELA achieves the highest accuracy compared to other algorithms, including VGG-19, VGG-16, ResNet-50, DenseNet-121, and Xception. The accuracy improves as ELA increases, highlighting the significant role of ELA in enhancing the model's performance for artifact detection.

5.9 Performance of CNN with Varying ELA Percentages

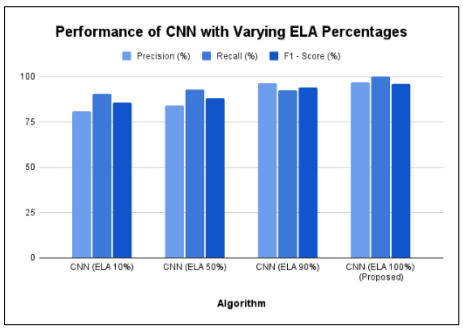


Fig. 15. Performance of CNN with Varying ELA Percentages

Figure 15 demonstrates that increasing ELA improves CNN's performance in precision, recall, and F1-Score. CNN with 100% ELA outperforms all other configurations, showing the importance of ELA in improving the model's capability to detect slight artifacts with greater accuracy.

5.10 Precision, Recall, and F1-Score of Various Algorithms

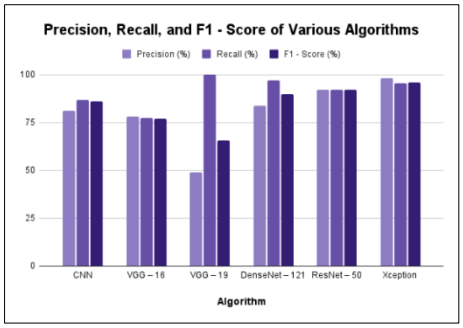


Fig. 16. Precision, Recall, and F1-Score of Various Algorithms

Figure 16 compares recall, precision, and F_1 score across multiple algorithms, with CNN using 100% ELA outperforming other models like VGG, DenseNet, ResNet, and Xception. This reinforces the superiority of the proposed system in achieving better performance metrics for forensic tasks.

Table I. Accuracy of Various Algorithms

Name	Accuracy (%)
CNN	72.62
CNN (ELA 10%)	77.44
CNN (ELA 50%)	82.36

CNN (ELA 90%)	92.34
CNN (ELA 100%) (Proposed)	**94.13**
VGG – 16	77.64
VGG – 19	48.92
DenseNet – 121	89.23
ResNet – 50	93.76
Xception	93.96

Table II. Performance of CNN with Varying ELA Percentages

Name	Precision (%)	Recall (%)	F1 - Score (%)
CNN (ELA 10%)	80.73	90.39	85.81
CNN (ELA 50%)	84.25	93.00	88.17
CNN (ELA 90%)	96.59	92.67	94.18
CNN (ELA 100%) (Proposed)	**97.07**	**99.98**	**96.23**

Table III. Precision, Recall, and F1-Score of Various Algorithms

Name	Precision (%)	Recall (%)	F1 - Score (%)
CNN	81.14	86.79	85.93
VGG – 16	78.09	77.510	77.034
VGG – 19	48.92	99.98	65.70
DenseNet – 121	83.64	96.93	89.80
ResNet – 50	92.26	92.01	92.11
Xception	98.26	95.39	95.76

Table I presents the accuracy (%) of various algorithms, including CNN models with varying ELA percentages and other architectures like VGG-19, VGG-16, ResNet-50, DenseNet121, and Xception. The proposed CNN with 100% ELA achieves the highest accuracy of 94.14%, outperforming all others.

Table II highlights the recall, precision, and F1-score (%) of CNN models with varying ELA percentages. The proposed CNN with 100% ELA achieves the highest precision (97.07%), recall (99.98%), and F1-score (96.23%), demonstrating its superior effectiveness in accurately detecting and detecting manipulations.

Table III presents an evaluation of recall, precision and F1-score (%) for different algorithms, including CNN, VGG-19, VGG-16, ResNet-50, DenseNet-121, and Xception. Among these, Xception achieves the highest precision (98.26%) and maintains strong recall (95.40%) and F1-score (95.77%), while DenseNet-121 and ResNet-50 also exhibit competitive performance across all metrics but, CNN with 100% ELA outperforms all of them.

6. CONCLUSION

This research developed an advanced detection system for image forgery that integrates CNNs with ELA to tackle the increasing challenge of detecting altered visual content. This system combines the strengths of CNNs for extracting intricate pixel-level features and ELA for detecting subtle compression artifacts that indicate tampering. Extensively tested on a varied dataset of authentic and manipulated images, the solution achieved an exceptional performance of 94%, achieving recall and precision rates of 100% and 97%, respectively, highlighting its capability to detect

manipulations that are often undetectable to the human eye. Additionally, its real-time image processing capability makes it well-suited for applications demanding quick and reliable image verification, such as in fields like law enforcement, journalism, and digital forensics. Through overcoming the limitations of traditional detection methods, this system represents a major breakthrough in ensuring authenticity of visual content. Future work could focus on incorporating newer forgery techniques into the training dataset, allowing the system to adapt to emerging threats such as GAN-based manipulations. Additionally, expanding the system to detect forgeries in video content and integrating blockchain technology for image provenance tracking would enhance its functionality and trustworthiness.

REFERENCES

[1] Hasan, M. M., Rana, M. M., & Rahaman, A. S. M. M. (2024). Insights into Manipulation: Unveiling Tampered Images Using Modified ELA, Deep Learning, and Explainable AI. *Journal of Computer and Communications*, *12*(6), 135-151.

[2] NFOR, K. A., Armand, T. P. T., & Kim, H. C. (2024). A Holistic Approach to Image Forensics: Integrating Image Metadata Analysis and ELA with CNN and MLP for Image Forgery.

[3] Kotti, J., Gouthami, E., Swapna, K., & Vesalapu, S. (2022). MORPHED IMAGE DETECTION USING ELA AND CNN TECHNIQUES. *Journal of Pharmaceutical Negative Results*, 4069-4078.

[4] Thakur, P., Joshi, B., Kachave, K., & Pawar, K. Exploring the Effectiveness of CNN and VGG16 with ELA in Image Tampering Detection.

[5] Nida, N., Irtaza, A., & Ilyas, N. (2021, January). Forged face detection using ELA and deep learning techniques. In *2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST)* (pp. 271-275). IEEE.

[6] Mallick, D., Shaikh, M., Gulhane, A., & Maktum, T. (2022). Copy move and splicing image forgery detection using cnn. In *ITM Web of Conferences* (Vol. 44, p. 03052). EDP Sciences.

[7] Maashi, M., Alamro, H., Mohsen, H., Negm, N., Mohammed, G. P., Ahmed, N. A., ... & Alsaid, M. I. (2023). Modelling of Reptile Search Algorithm with Deep Learning Approach for Copy Move Image Forgery Detection. IEEE Access.

[8] Saber, A. H., Khan, M. A., & Mejbel, B. G. (2020). A survey on image forgery detection using different forensic approaches. *Advances in Science, Technology and Engineering Systems Journal*, *5*(3), 361-370.

[9] Qazi, E. U. H., Zia, T., & Almorjan, A. (2022). Deep learning-based digital image forgery detection system. *Applied Sciences*, *12*(6), 2851.

[10] Zhang, Y., Goh, J., Win, L. L., & Thing, V. (2016). Image region forgery detection: A deep learning approach. In *Proceedings of the Singapore Cyber-Security Conference (SG-CRC) 2016* (pp. 1-11). IOS Press.

[11] Farid, H. (2009). Image forgery detection. *IEEE Signal processing magazine*, *26*(2), 16-25.

[12] Deep Kaur, C., & Kanwal, N. (2019). An analysis of image forgery detection techniques. *Statistics, Optimization & Information Computing*, *7*(2), 486-500.

[13] Sari, W. P., & Fahmi, H. (2021). The effect of error level analysis on the image forgery detection using deep learning. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*.

[14] Ghannad, N., & Passi, K. (2023). Detecting Image Forgery over Social Media Using U-NET with Grasshopper Optimization. *Algorithms*, *16*(9), 399.

[15] Sudiatmika, I. B. K., Rahman, F., Trisno, T., & Suyoto, S. (2019). Image forgery detection using error level analysis and deep learning. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *17*(2), 653-659.

[16] Wu, Y., Abd-Almageed, W., & Natarajan, P. (2018, March). Image copy-move forgery detection via an end-to-end deep neural network. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 1907-1915). IEEE.