

Effectual Deep Learning Based Digital Image Forgery Detection Model Using Ameliorate CNN, VGG 16 and VGG 19 for Copy Move and Image Splicing Classification With Enhanced Performance

Preeti Dhiman and Usha Chauhan

Department of Electrical, Electronics and Communication Engineering, Galgotias University, Greater Noida, India

Article history

Received: 07-05-2026

Revised: 21-08-2026

Accepted: 25-08-2026

Corresponding Author:

Usha Chauhan,
Department of Electrical,
Electronics and Communication
Engineering, Galgotias University,
Greater Noida, India
Email:
usha.chauhan@galgotiasuniversity.edu.in

Abstract: In this research, a pioneer passive approach for detecting and localizing tampering in images is proposed. In the past, image forgeries have created a huge impact in the world. Developments of various image processing technologies made it easier to tamper images with the objective to cover up or hide a significant feature in an image. In some situation, such as in a law-suit, it is necessary to prove that images are not tampered. Furthermore, in medical research, the validity of an image is critical, and this need will expand with the future standardization of data sharing among institutions or between patients and practitioners. Several more domains are reported in the literature that need image forgery detection with high accuracy; however, the majority of these works are done in high quality images with only one type of forgery. Many finer features are lost during compression from high to low quality images due to transformations such as smoothing, anti-aliasing, and moreover many techniques can be used for forging the images. Therefore, a new approach is presented to detect such forgeries with accuracy using Deep learning model architecture based on sequential CNN that can precede all previous attempts for images with both type of forgeries (copy paste and image splicing). Among all three evaluated models, Ameliorate CNN model demonstrated the strongest overall performance, achieving 93.11% accuracy, 95.54% precision, 90.63% recall, and 93.02% F1-score on the sampled batch of 833 images. Across the complete dataset of 9,501 images, Model 1 correctly classified 2,020 of 2,064 real images and 6,756 of 7,437 fake images, resulting in 97.87% real-image accuracy, 90.84% fake-image accuracy, and 92.37% overall accuracy. In comparison, Model 2 obtained the highest recall (92.20%) and 91.52% fake-image accuracy among the three models, its substantially lower real-image accuracy and precision reduced its overall performance. VGG 19 based model achieved real and fake-image accuracy comparatively more balanced as compare to VGG 16 based model; however, its overall accuracy and F1-score remained lower than Ameliorate CNN model. Overall, Ameliorate CNN model provided the highest overall accuracy, precision, and F1-score which demonstrate the strongest discrimination of authentic images, establishing it as the best-performing model among the three evaluated architectures under the experimental conditions of the study with improvement in enhancing feature representation and robustness in image forgery detection.

Keywords: Image Forgery Detection, Deep Neural Networks, Convolution Neural Network, VGG 16, VGG 19, Copy Move and Splicing

Introduction

Since the advent of digital media technologies, digital images have steadily permeated all sectors of existence digital due to its simplicity of use, portability, and rich information content. The press, politics, insurance claims, defense, and legal cases, agriculture fields are just a handful of the many major areas where it has evolved into a critical piece of evidence. Data saved or sent in digital form, on the other hand, is open to external incursion, and digital photographs are especially sensitive to illicit access and alteration because of the usage of extensive editing skills in modern technology such as Adobe Premiere, Photoshop Elements, and Pixelmator, GIMP, Pixlr etc. Digital content can be quickly updated by non-professionals in such a way that even an expert gets tricked by the indiscernible forgeries. Once these forged images circulate on the internet, they have the potential to seriously undermine the integrity of person/ institution or even the government. This highlights the importance and necessity of competent Forgery detection procedures capable of assessing the validity, uniqueness, and data integrity. Furthermore, Digital Image Forgery Detection (DIFD) Techniques are generally used to evaluate the genuineness of digital images and to protect image rights, and it may be classified as Active or Passive forensic procedures based on different detection techniques, as shown in Fig. 1.

Furthermore, there are two types of broad classification of forgery detection techniques to evaluate the genuineness of digital images:

- Conventional Techniques: Active forgery forensic techniques (image watermarking (Nikolaidis and Pitas, 2002) and image signatures (Farid, 2009) are used to entrench aforementioned information to preserve the copyright and integrity of the images and then extract the entrenched information to detect whether the digital image is original or not. This method, however, impose the requirement of specialized hardware or software for inserting the authentication code into the picture prior to distribution of the image on public domain
- 1) Passive image authentication/ forensic techniques are the process of validating digital images through images themselves (Fridrich et al., 2003). Passive approaches are categorized into following types:
 - i) Pixel Based: is the most popular technique for forgery detection. Following are the various types of forgeries (Tian-Tsong and Chang, 2005) along

with their detection methods (Mehrzardi et al., 2023):

- Copy Move / Cloning: Copy one / more regions from one image and pasting them onto the same image at different location to conceal some of the information (Piva, 2013; Babu and Rao, 2023) or to replace some of the objects/region is known as cloning or copy-move forgery. DCT (Hayat and Qazi, 2017) and PCA, SIFT (Yang et al., 2018) are two popular computational efficient algorithms to detect cloned image by reducing computational complexity and then ensuring robust detection by sorting the DCT Block Coefficients in lexical order and Similar blocks with the same spatial offset are grouped together in the image to find any slight variations in an image for having additive/ white noise or lossy compression
- Splicing: In this technique, the forgery is done by pasting the objects/ region from another image. The spliced region changes the higher-order Fourier statistics. These discontinuities result in variation in the magnitude and phase of the bi-spectrum which further helps to detect any spliced region in the digital images (Popescu and Farid, 2005; Bay et al., 2008; Meena and Tyagi, 2023)
- Re-sampling: Sometimes, it becomes necessary to resize/rotate/stretch some segment of an image to make it further believable. The process of re-sampling creates precise periodic-correlations in between neighbor pixels to detect this type of forgery (Lyu and Farid, 2005)
- Statistical: Each image contains the statistical properties and any variations in statistical regularities in digital images are used to detect image manipulation. Wavelet decomposition of an image is used to compute first and higher-order statistics where the decomposition splits frequency space into oriented sub-bands with multiple scales. This model's frequency space, which includes multiple scales and orientations, is analyzed for the first four statistical moments of each wavelet sub-band, as well as higher-order statistics to detect correlations between these sub-bands. Another approach extracts nine features representing binary string similarity from the sequential floating forward search algorithm. The descriptive data retrieved are used by a linear regression classifier to distinguish between authentic and altered photos (Lyu and Farid, 2005; Johnson and Farid, 2006)
 - ii) Format Based techniques basically utilize the peculiar properties of JPEG lossy compressed images. One of the techniques is JPEG quantization scheme where each camera manufacturer chooses lossy compression scheme.

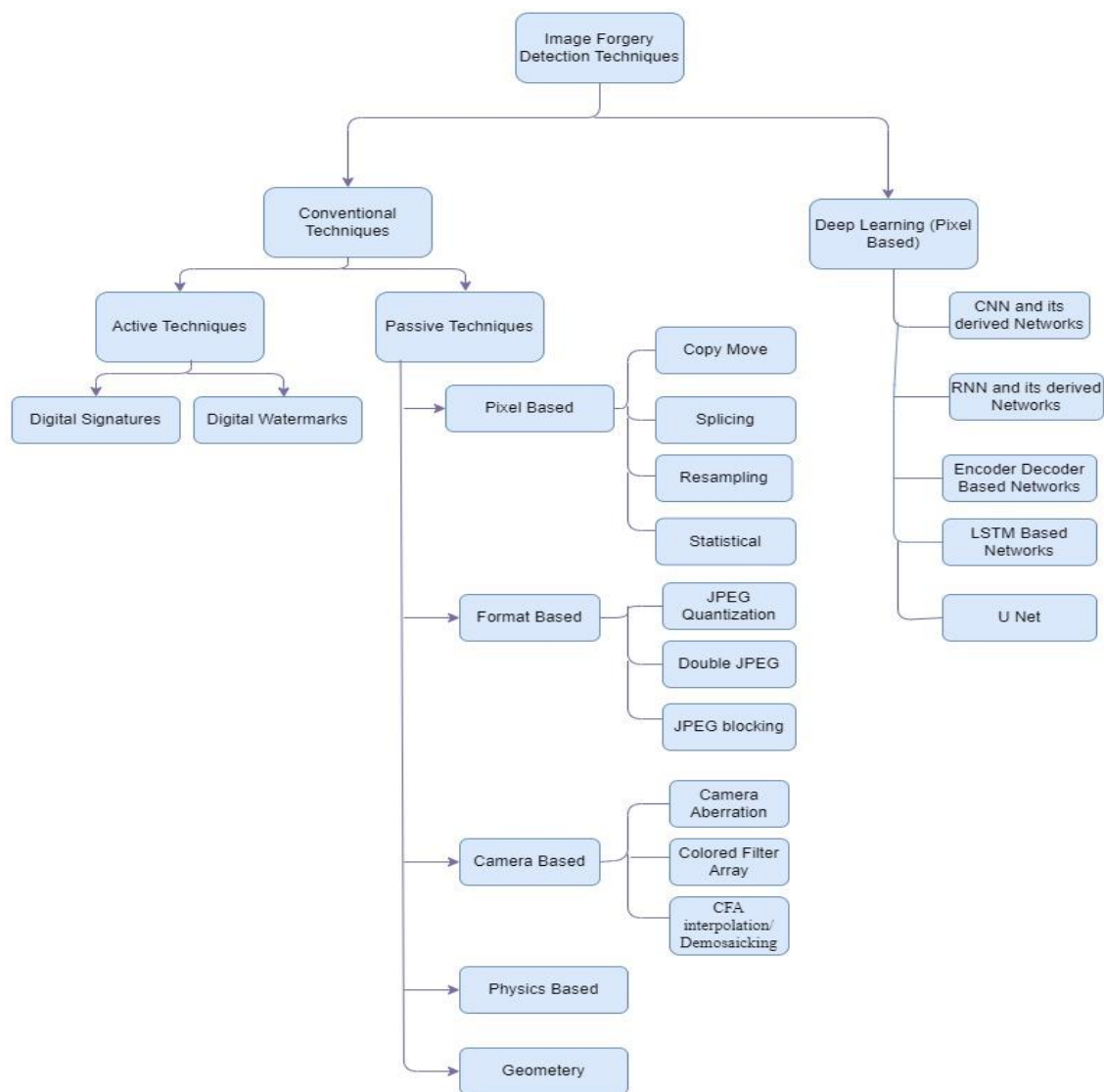


Fig. 1: Classification of Various Images Forensic Techniques

iii) It is based upon how much balance manufacturer wants to maintain between compression and quality of image. Further the difference is used to identify the source camera/ its model (Diallo et al., 2020). Another scheme is known as Double JPEG where the image which is manipulated will be compressed twice. The double compression changes few artifacts of the images which are not present in single compressed image (Fang et al., 2020). However, for this detection technique, both original and its manipulated image must be saved in JPEG format. Third method use Block DCT transforms for JPEG compression and is known as JPEG blocking technique. An image is considered as 8x8 pixel blocks which are transformed and quantized one by one, block artifacts appear in neighboring blocks as horizontal and

vertical edges. Following are the key steps to detect image manipulation:

- New blocking artifacts get introduced which are not available in original image boundaries
 - Further, Supervised pattern classification is used to differentiate original and manipulated image
- iv) Camera based: The camera-based techniques which are mostly used to identify the forgery are based on Camera Aberration where the estimated local lateral aberration is compared with the estimated global aberration in this tampered region to find the inconsistency (Palaz et al., 2019). The other method to find the inconsistency to use the Color Filter Array

(CFA). Digital cameras (mostly) use CFA having three color filters as Red, Blue, and Green which are placed at top of camera sensor. As, a single camera sample is captured for each pixel location, the other two-color samples are estimated using the neighbor sample information to obtain the three-channel colored image and the method of estimation is called CFA interpolation/ Demosaicking. The forgery in any image can be detected through the abrupt changes in the periodic pattern as in the original image the correlation is periodic with the neighbor pixel (Ramamoorthi and Hanrahan, 2001; Johnson and Farid, 2008)

- v) Physics based: This method estimates the light source direction of all its regions and objects in an image and results in detecting forgery through the light inconsistencies in an image (Girshick et al., 2014)
- vi) Geometric-based: Principal Point (the projection of camera centre onto image plane), is one way for spotting anomalies. Any inconsistency in an image's principal point is considered evidence of tampering. Another method employs metric measurements, in which several projective geometry tools are used to allow planar surfaces rectification and, to make real-world measurements of planar surface about polygons of known shape (Ali et al., 2022). Another method for identifying any contradiction is based on the Principal Point, which is essentially the projection

The other way to classify the techniques is:

- i. Source device identification and
- ii. Tamper detection techniques

The former is based on distinguishing camera fingerprints, which are the traces left behind after a camera captures and stores an image. So, it employs camera fingerprints to separate the camera models or exemplars of the camera type that are the same.

The forensic and passive image authentication technique does not require any prior image information; instead, they analyze the image's structure and statistical features to detect purposeful or unintentional tampering. Forgery and tampering alter the statistical qualities of an image, resulting in various abnormalities. These abnormalities are used to identify and locate fake photographs. There are several passive forgery strategies, but current research focuses on two device-dependent pixel based passive forgery approaches, copy-move and spliced picture forgery detection Techniques are most popular.

Copy-move forgeries allow for the copying and pasting of a few areas from the same image, resulting in an extremely exact imitation effect. Strong correlations between copied areas of the image are frequently seen in copy-move attacks; however,

because these regions of any size or form, it is computationally impractical to investigate the entire space of potential shape/size combinations.

Copy Move Forgery Detection (CMFD) algorithms mainly consist of finding similar feature vectors, extracting block-wise feature vectors, and choosing feature pairs with extremely similar shift vectors. The CMFD can also incorporate rotation or scale invariant features.

These forgeries are detected by using different techniques for both Copy-Paste and spliced images. The Conventional feature-based techniques for Copy Move are Discrete Scale-Invariant Feature Transform (SIFT) (Ali et al., 2022) and Speed-up robust feature (SURF) (Castillo and Wang, 2021), Principal Component Analysis (PCA) Yu et al. (2025), Cosine Transform (DCT) Liang et al. (2024), Discrete Wavelet Transform (DWT) Atak and Yasar (2024), Polar Complex Exponential Transform (PCET) (Atak and Yasar, 2024).

Various studies employ the strategies listed above, either alone or in combination, to increase detection accuracy.

In case of splicing, source and target are heterogeneous i.e. the copied contents come from multiple image sources, so it introduces more artifacts. A few of the typical detection algorithms in this type of forgery are:

- CFA interpolation for inconsistency detection
- Noise characteristics

The following are the key drawbacks of traditional CMFD and splicing methods:

- i. Poor generalization
- ii. Poor resilience, and
- iii. Inadequate detection accuracy

Deep Learning (DL) Based Techniques

As illustrated in Fig. 2, a subset of Machine Learning (ML), Deep Learning (DL) draws inspiration from the human brain's capacity to identify patterns in any given data or information. Although DL does not need any explicit rules to work, it does need a lot of data in order to translate a given input into a particular output or label. Each of Dense layers generates a unique feature map of the incoming picture data. Furthermore, DL-based detection algorithms (Ali et al., 2022; Atak and Yasar, 2024) are better suited for many applications than typical approaches since they utilize autonomous learning features:

- Natural Level Processing (NLP)
- Semantic Analysis (SA)
- Computer Vision (CV)
- Audio/ Video Processing, and
- Medical applications and many more

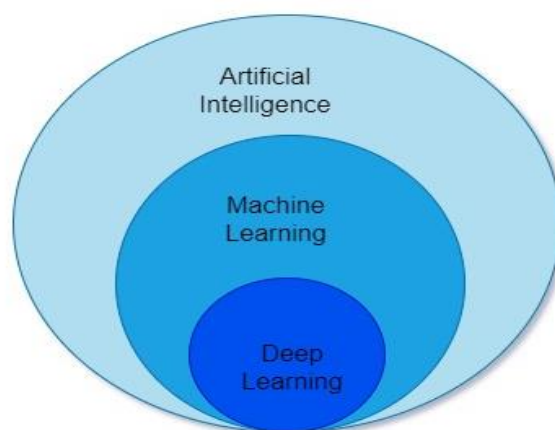


Fig. 2: Venn diagram representing relationship between AI, ML and DL

Table 1: Comparison of Techniques used by Various Researchers

Year	Author	Method	Limitation
2018	Yang et al.	Weakly supervised relation learning	Poor unseen manipulation performance
2025	Yu et al.	Knowledge distillation	Complex training
2024	Castillo and Wang	Vision Transformer + VAE	Weak generalization
2025	Yu et al.	LBRT Transformer	High memory usage
2023	Mehrpour et al.	Deep learning survey	High computational cost
2023	Meena and Tyagi	Noise estimation	Sensitive to enhancement
2023	Babu and Rao	PCET + GDP	Computational overhead
2022	Ali et al.	Recompression CNN	High preprocessing time
2022	Qazi et al.	Deep CNN	Low interpretability
2022	Mallick et al.	CNN forgery detection	Poor cross-dataset performance
2021	Castillo and Wang	DL forensic review	Limited lightweight models
2020	Diallo et al.	CNN supervision	Needs large datasets
2020	Li et al.	Lightweight CNN	Reduced feature richness
2019	Palaz et al.	End-to-end CNN	Task-specific architecture
2018	Yang et al.	CMFD-SIFT	High complexity
2017	Hayat and Qazi	DWT-DCT	Weak under compression
2014	Girshick et al.	R-CNN	Slow inference
2013	Piva	Image forensics survey	Handcrafted features
2009	Farid	Forgery survey	Limited deep learning
2008	Bay et al.	SURF	Sensitive to distortions
2007	Johnson and Farid	Composite detection	Limited robustness
2006	Johnson and Farid	Chromatic aberration	Camera dependency
2005	Popescu and Farid	Resampling detection	JPEG sensitivity

The reason for utilizing DL models in diverse applications is to provide highly precise predictions that human expertise cannot provide in some circumstances. Furthermore, DL models update themselves over time and can also provide transfer learning.

The results of DL models for various applications are widely accepted because to their generalizability, as different data sets for distinct applications can apply same DL approach. Furthermore, DL Models requires large amounts of data to provide sufficient robustness, which is made feasible by their capacity to extract optimized features that these models learn over time. Following are the advantages of having DL Models over ML models:

1. Universal learning approach refers to developing the models that can learn and generalize across multiple applications, domains, or data modalities with minimal need for fine-tuning or retraining the model. These versatile models can adapt to new problems without requiring significant modifications or large amounts of task-specific training data. This approach contrasts with traditional Machine learning Models, which are often designed for and specialized in specific tasks
2. Robustness is ability of a model to sustain high performance and accuracy under noisy or corrupted data, adversarial attacks and some real-world variability. Thus, a Robust DL model can generalize

well, even when it encounters inputs or environments that are significantly different from the training data

3. Generalization is most crucial property for DL models which guarantees the model's success well in practical scenarios. By addressing over fitting, using regularization, increasing data diversity, and employing architectural innovations, researchers can improve a model's ability to generalize further. The overall objective is creating models that not only function well on training data but also effectively solve novel problems and adapt to unseen environments
4. Scalability: The frameworks used for DL models are highly scalable. It is an ability to efficiently train and deploy models as data, computational resources, and model complexity grow. As deep learning models become larger and more complex, handling increasing amounts of data and computation while maintaining performance and minimizing bottlenecks is critical. Scalability is crucial in large-scale recommendation systems

Widely used DL Networks in the field of Image forensic are mostly supervised Learning Models where variants of Convolution Neural Networks (CNNs) (Fang et al., 2020; Li et al., 2020) are commonly employed in forgery detection. CNN's primary advantage over its predecessors is its ability to recognize pertinent elements automatically and without human oversight. Numerous other domains, such as computer vision (Palaz et al., 2019), speech processing, face recognition (Li et al., 2020), semantic analysis (Girshick et al., 2014) and others, have also made substantial use of CNNs. Architecture of CNN can recognize both low and high-level elements in pictures. The CNN offers three main advantages: Parameter sharing, sparse interactions, and equivalent representations.

CNN networks with fully connected layers, shared weights and local connection utilize 2-D data for input images. Due to simple structure of CNN, the model generates the smaller number of parameters and hence computation time will be reduced. The local filters used in these models scans the small regions of the images to identify local correlations and these correlations are analyzed to predict authentication of an image in image forgery detection applications. Table 1 summarizes the techniques used by various researchers along with some findings.

Proposed Methods

Dataset

There are several datasets which are available for the forgery detection field. For the research work, CASIO V2 dataset has been used. Since, it is imbalanced data set

having more forged images thus model selection and validation is required to ensure that model generalizes well and does not favor the majority class i.e. forged images. In this scenario, few measures are taken considering problems involved and model selection/ validation.

There are few of problems with imbalanced data set which varies from Skewed Predictions where model tends to focus on the majority class i.e. forged images because it is easier to optimize for overall accuracy by predicting the majority class more frequently which further leads to poor performance on the minority class (authentic images in this case) to misleading metrics where Standard metrics like accuracy become less informative since a model predicting only the majority class could still have high accuracy without being effective at detecting the minority class. The strategies which are used to select the model include re-sampling (Over sampling/ under sampling), Cost-sensitive Learning assigns higher penalty (cost) for misclassifying the minority class (authentic images). This approach encourages the model to focus on reducing false negatives for authentic images. Anomaly detection treats the minority class (authentic images) as anomalies and involves training the model predominantly on the majority class.

Further, the Validation of model may utilize Stratified Cross instead of traditional k-fold cross-validation to ensure that each fold has the same proportion of authentic and forged images as the original dataset. It helps the model evaluate performance more fairly on both classes and Precision, Recall, F1 Score, Confusion Matrix, may be used as the metrics to validate the model.

Proposed Models

Unlike the conventional CNN models which detects only one type of forgery, the Ameliorate CNN model detects two or more types of forgeries with better accuracy and F1 score. Three different types of ameliorated CNN models are compared with their validation scores. All the proposed models will use the same steps as mentioned in Figure 3.

Fig. 3 illustrates the typical workflow of developing and evaluating Ameliorate CNN model. Following is the step-by-step explanation of the process:

- Data Collection: The process starts with gathering raw data from various sources. The famous datasets are CASIA TIDE (Tampering Detection Evaluation Dataset) CASIA v1.0, CASIA v2.0, Columbia Image Splicing Detection Dataset, CoMoFoD (Copy-Move Forgery Detection Dataset), GRIP Dataset, MICC-F220 and MICC-F600 etc. The proposed method has used CASIA v2.0
- Data Pre-processing: Collected data is cleaned and normalized, and formatted appropriately for training. Most popular techniques are resizing images,

augmenting data, or removing outliers which are applied during this step

- **Data Splitting:** The pre-processed data is divided into training and testing sets where CNN model being trained using training data and tested with testing data
- **Model Training:** During this process of training, the model learns patterns, features, and representations from the data
- **Training Results:** Metrics for loss and accuracy are computed to provide the Model's learning progress during training
- **Proposed Model:** The trained model is tested on testing data to validate performance
- **Performance Evaluation Results:** The model's predictions are compared with actual labels in the testing data to compute evaluation metrics
- **Output:** Finally, when the results are satisfying, the trained model is deployed for real-world use to generate predictions (e.g., classify as "Real" or "Fake")

Implementation of Proposed Models

The first step for all the models is ELA (Error Level Analysis) which is applied to reveal regions in an image that may have been manipulated. Due to saving image in in JPEG format, compression artifacts are introduced, and these artifacts are consistent throughout the image if it is not altered. But when an image is edited (e.g., spliced/ cloned/ copy-pasted), the tampered regions often have different compression levels compared to the original parts of the image and these inconsistencies are exposed by ELA. The dissimilarity between the original image and its recompressed image at a lower quality is visualized as a heatmap. Bright regions in the heatmap often indicate potential tampering in an image.

It becomes important to use ELA when Forgery artifacts are often subtle and hard to detect with raw pixel data and sometimes differences in lighting, shadows, or textures may fool the model. ELA bridges this gap by pre-processing the image to highlight possible forgery artifacts as shown in Fig. 4a and b. The heat map provides the Deep learning model with explicit cues about regions that may be altered.

To integrate ELA as shown in Fig. 5 with Deep learning Models for detecting (Mallick et al., 2022; Qazi et al., 2022)/ Splicing/ Retouching type of forgeries, following steps are used.

However, there are few limitations of using ELA as it has JPEG Dependency ELA is most effective on JPEG images, as the technique relies on compression artifacts. For formats like PNG or BMP, alternative preprocessing methods might be needed.

Also, Images that have undergone multiple compressions (e.g., uploaded to social media) may lose their original artifacts, making ELA less effective. Further sometimes Natural variations (e.g., edges, high-contrast regions) in an image may also show high error levels, leading to false positives. The next step is to apply DL Model. Each type of DL reflects the standard steps followed for any model used in the application and highlights the iterative nature of model for improvement through tuning and validation.

Ameliorate CNN Model

Figure 6 outlines the architecture of an Ameliorated Convolution Neural Network (CNN) model designed for fake image detection. The key components and their roles in the pipeline are as follows.

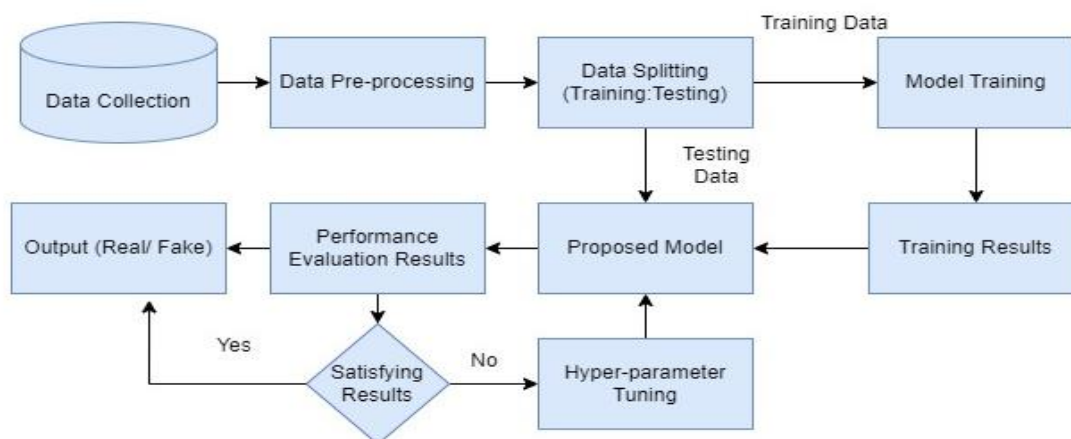


Fig. 3: Data Flow in Proposed systems

Input Layer takes an input image of size $128 \times 128 \times 3$ (Height $\times$ Width $\times$ RGB channels). This preprocessing step ensures all images are resized and normalized to fit the expected dimensions. The second is consecutive Convolution Layers where the first convolution layer consists of 32 Filters, Kernel Size of (5×5) , Padding = "Valid" (no padding is applied to reduce spatial dimensions), Activation layer with ReLU (Rectified Linear Unit) function helps the model to learn complex features. Further, Second Convolution Layer having similar parameters as the first (32 filters, 5×5 kernel, valid padding, and ReLU activation is used to identify high-level features from the input image. Next, Max Pooling Layer by taking the maximum value in 2×2 regions minimize the spatial dimensions.

It helps in down-sampling, reducing the computation

while retaining essential features. Further, Dropout Layer is used as Regularization technique drops 25% neurons during training to prevent over-fitting. This layer ensures the model generalized to unseen data. The Flatten Layer Converts the 2D feature maps from the previous layers into a 1-D vector. This flattened vector is fed into the dense (fully connected) layers. The second last layer consist of fully connected (Dense) Layers with Dense Layer units of 256, Activation function as ReLU which extracts high-level, abstract features and patterns from the flattened input. The Dropout Layer Drops 50% of neurons to further reduce over-fitting during training. The last Layer with SoftMax generates the output probabilities for two classes: Real or Fake. The final output indicates whether the image is real or fake based on the SoftMax probabilities.

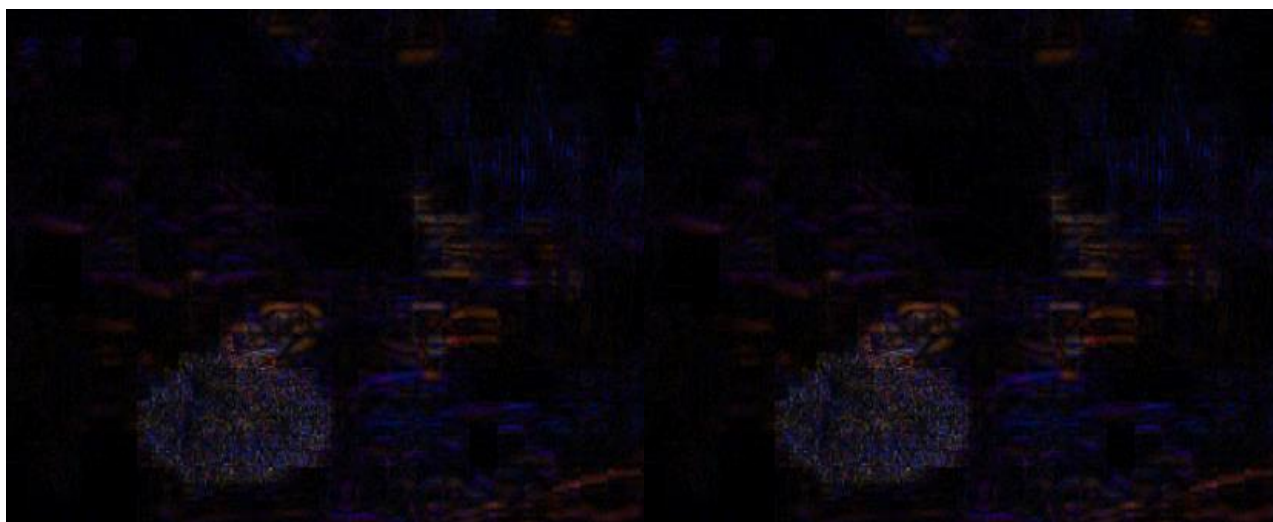


Fig. 4a: Tempered Image and its ELA Heatmap



Fig. 4b: Tempered Image and its ELA Heat map

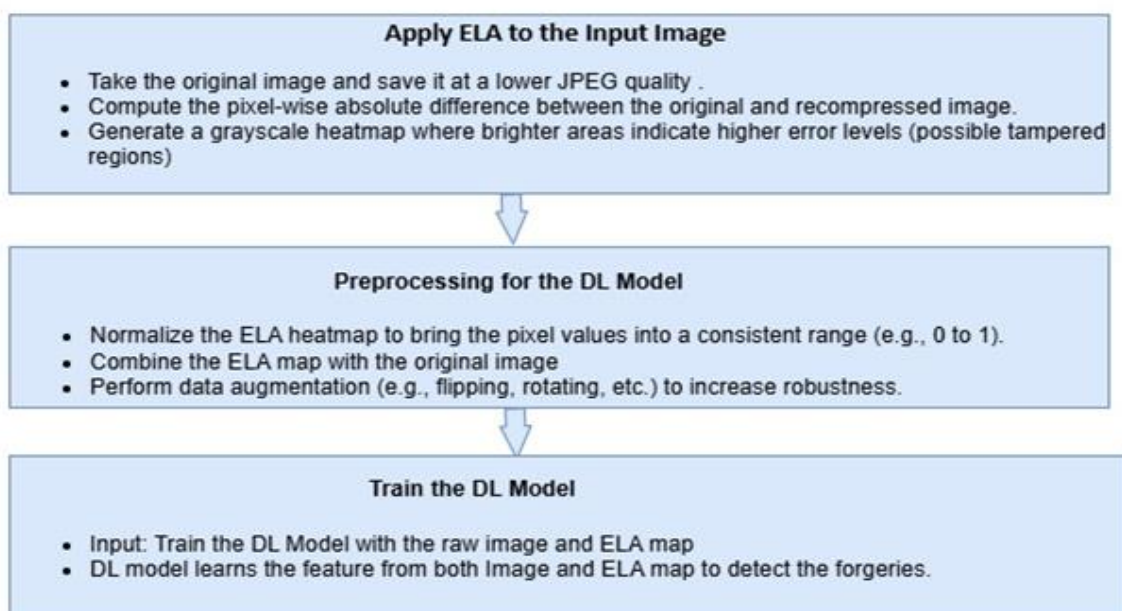


Fig. 5: ELA Implementation

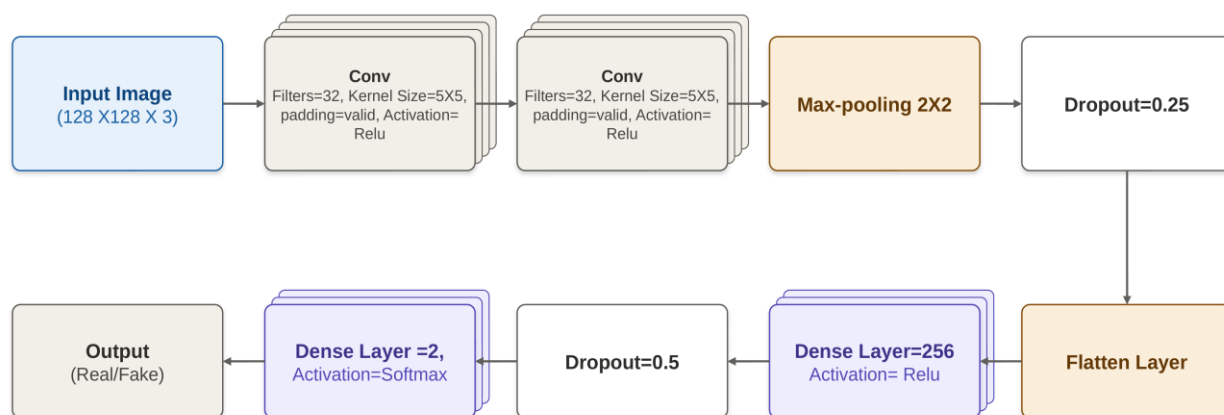


Fig. 6: Ameliorate CNN model

VGG 16 Pre-Trained Model With Ameliorate CNN

The image represents VGG 16 (Fig. 7) based model architecture for forgery detection using the VGG16 model, which is a pre-trained deep learning model designed for image classification tasks. Here's how it works step by step.

The model takes a same input with specification (128x128) pixel image with 3 color channels (RGB) as input. These images represent the data to be classified as either real or fake. VGG16 is CNN which is pre-trained on the ImageNet. It serves as a feature extractor, leveraging its pre-trained weights to extract high-level features. Frozen Layers in the convolutional layers of

VGG16 are frozen (not trainable) to use the learned features from the ImageNet dataset. After the convolutional layers of VGG16, the output (a multi-dimensional feature map) is flattened into a 1D vector. Fully connected dense layer with 1024 neurons is applied to learn more complex patterns from the extracted features. Rectified Linear Unit (ReLU) learns non-linear decision boundaries. Further, Dropout randomly deactivates 25% neurons during training to generalize better. Dense Layer 2 (2 neurons, SoftMax activation) has 2 neurons, corresponding to the two possible output classes: Real and fake. SoftMax Activation ensures the output probabilities as real or fake. The advantage of using VGG16 in Forgery

Detection is to help identify subtle inconsistencies or anomalies in forged images, such as tampering artifacts, mismatched textures, or unnatural transitions.

VGG 19 Pre-Trained Model With Ameliorate CNN

This next architecture combines the robustness of VGG19 based model with additional fully connected layers tailored for the forgery detection task. The model takes a 128x128 RGB image as input. Further it utilizes a pre-trained VGG19 network (trained on ImageNet) to extract high-level features of an input image. The convolution layers of VGG19 are leveraged for robust

feature extraction. Further the flatten Layer converts the multi-dimensional feature map from to prepare it for fully connected layers.

The Dense Layer (1024 neurons, ReLU activation) is used to learn additional patterns from the extracted features. The ReLU captures complex relationships. The dropout deactivates 25% of neurons during training. The Dense Layer (2 neurons, SoftMax activation) generates outputs which predict whether the input image is real or fake. Here SoftMax activation function ensures the output is interpretable as probabilities.

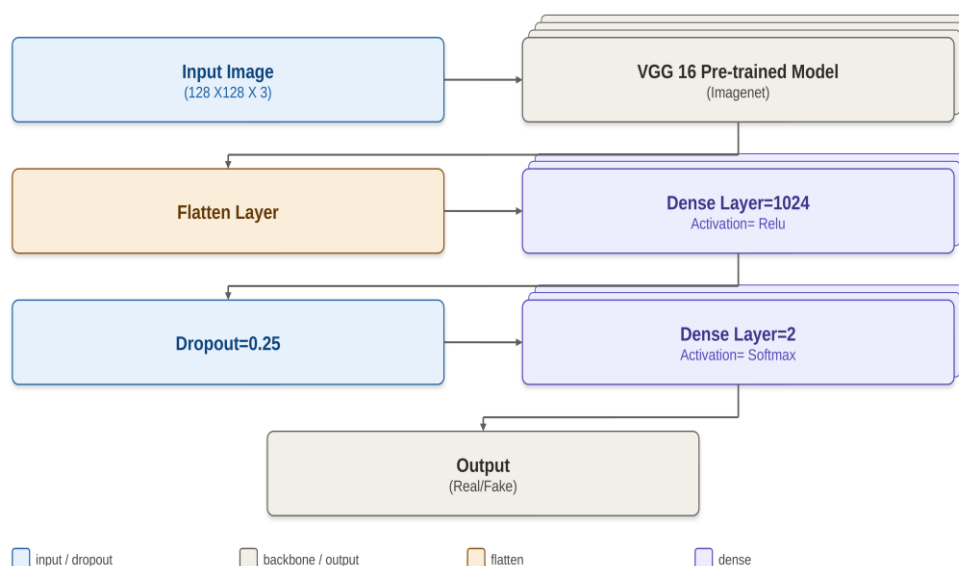


Fig. 7: VGG 16 Pre-trained Model with Ameliorate CNN

The purpose of VGG19 being deeper than VGG16 is to capture more intricate details and features from images. Its feature extraction capabilities combined with additional layers make it effective for forgery detection tasks.

Results and Discussion

Study Design: The study design evaluates the effectiveness of a Ameliorate CNN against two well-known transfer-learning baselines i.e. VGG16 and VGG19 for the task of real vs. fake image classification. All three architectures were trained and evaluated for CASIA V2 dataset with same dataset splits, same preprocessing pipeline, same evaluation metrics, and same software environment (executed on NVIDIA GPU with sufficient RAM) in order to analyze the difference in performance and computation time which could attribute to architectural choice rather than confounding with experimental variables. The design followed performance assessment for both smaller sampled batch

(833 images) for fine-grained metric assessment and on the full CASIA V2 dataset (9,501 images) for analyzing overall generalization performance.

Process

Images were pre-processed (resized, normalized) and ELA processed. The CASIA V2 was divided into training, validation, and test subsets to allow per-class accuracy analysis. VGG 16 and VGG 19 based models were initialized with ImageNet-pretrained weights and fine-tuned (transfer learning). Training time, per-class inference (detection) time, and total computation time were logged for each model to support the efficiency comparison alongside accuracy.

Data Analysis

Model outputs were evaluated based on accuracy, precision, recall, and F1 score. Firstly, it was computed on the 833-image sampled batch. And then, Overall accuracy was computed across the full dataset along with

computation time by comparing training time and per-class detection time. With all these parameters, classification performance was assessed for accuracy-efficiency trade-off.

3.1 For Ameliorate CNN model, taking a batch of samples, the confusion matrix is shown as.

The above results are for the Ameliorate CNN model where Fig. 8a shows Confusion Matrix with the following Metrics Calculation:

- Accuracy which is measures the proportion of correctly classified instances: $\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{386 + 389}{386 + 389 + 18 + 40} \approx 93.11\%$
- Precision (Positive Predictive Value): $\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{386}{386 + 18} \approx 95.54\%$
- Recall (Sensitivity or True Positive Rate): $\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{386}{386 + 40} \approx 90.63\%$
- F1 Score: Harmonic mean of precision and recall. $\text{F1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = 93.020$

Where, True Positives (TP, bottom-right = 386) represents correctly predicted label 1 by model, True Negatives (TN, top-left = 389) represents correctly predicted label 0, False Positives (FP, top-right = 18) represents the prediction as 1 by model when the actual label was 0 and False Negatives (FN, bottom-left = 40) represents the model prediction is 0 when the actual label was 1. For the proposed model, F1 score is a robust and

meaningful metric for evaluating model performance on unbalanced datasets as it highlights the trade-off between precision and recall, ensuring that the minority class is not overlooked. Thus, by focusing on the F1 score, it ensures the model's predictions are meaningful and actionable, especially in high-stakes applications like fraud detection, medical diagnosis, or anomaly detection.

From Complete Data set, total 2064 fake images were used for training and validation while 7437 real images were used for training and validation. The results are:

- Total Real Images: 2064, Correct: 2020, Accuracy: 97.86821705426357
- Total Fake images: 7437, Correct: 6756, Accuracy: 90.843081887858
- Total images: 9501, Correct: 8776, Overall Accuracy: 92.36922429217978
- Further Fig. 8b shows Training and Validation Loss/Accuracy Curves where Training and validation loss decrease over epochs, indicating improved predictions. However, the slight increase near the end suggests minimal over fitting. Both training and validation accuracy increase over epochs and converge, demonstrating consistent learning as validation metrics closely track training metrics. Thus, the model generalizes well on unseen data.

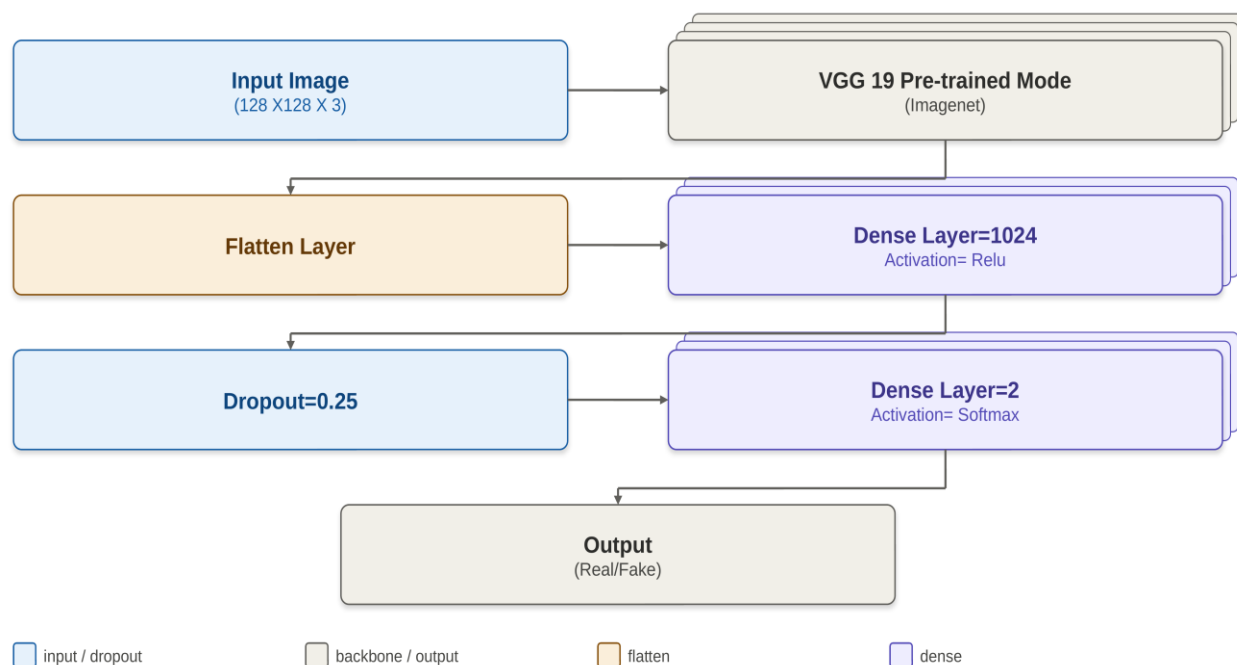


Fig. 8: VGG 19 Pre-trained Model with Ameliorate CNN

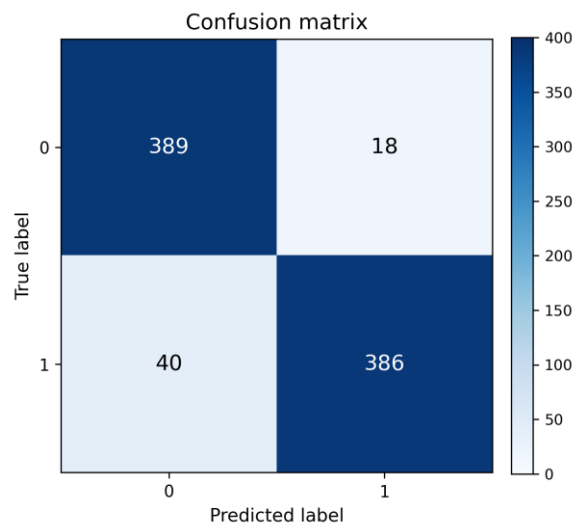


Fig. 8a: Confusion Matrix of Ameliorate CNN model

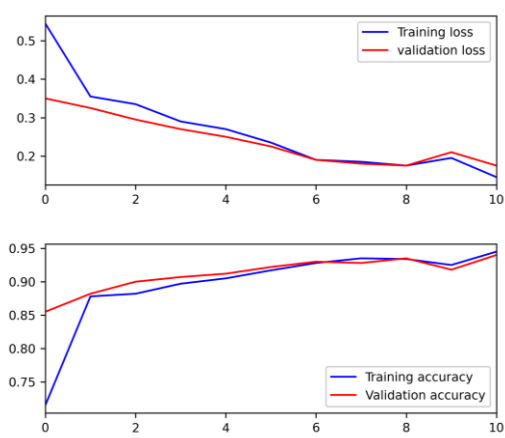


Fig. 8b: Loss and Accuracy of Ameliorate CNN model

Results for VGG 16 Based Models

From sampled Data set, Evaluation results for VGG 16 model are:

- Accuracy: $TP+TN / TP+TN+FP+FN = 393+306 / 393+306+101+33 = 699/833 \approx 83.9\%$
- Precision = $TP / TP+FP = 393 / 393+101 = 393/494 \approx 79.6\%$
- Recall (Sensitivity) = $TP / TP+FN = 393 / 393+33 = 393/426 \approx 92.2\%$
- F1 Score = $2 \cdot \frac{0.796 \cdot 0.922}{0.796 + 0.922} = 2 \cdot 0.7331718 \approx 86.0\%$

From Complete Data set, total 2064 fake images were used for training and validation while 7437 real images were used for training and validation. The results for VGG 16 model are:

- Total Real Images: 2064, Correct: 1633, Accuracy: 79.11821705426357
- Total Fake images: 7437, Correct: 6806, Accuracy: 91.51539599300793
- Total images: 9501, Correct: 8439, Overall Accuracy: 88.82222923902748

This plot in Fig. 9b shows the training and validation performance metrics (loss and accuracy) for a VGG 16 model over a training process for 6 epochs. In the above blue Line (Training Loss) starts with 1 value and steadily decreases as the epochs progress, indicates the model is learning to fit the training data better over time. Red Line (Validation Loss) loss starts lower as compare to the training loss but increases as training progresses around first epoch where the model is starting to memorize the training data instead of generalizing well to unseen data however it further demonstrates consistent learning as validation metrics closely track training metrics.

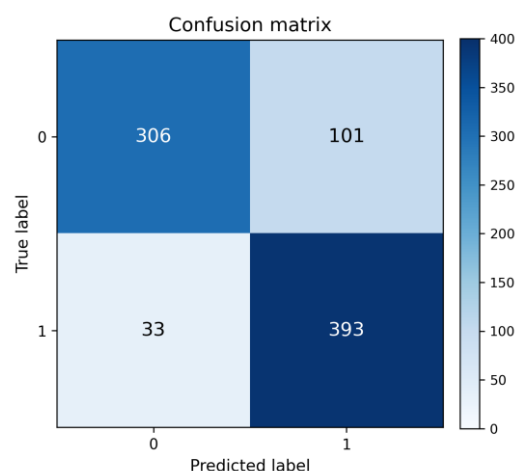


Fig. 9a: Confusion Matrix for Sampled Images of VGG 16 Model

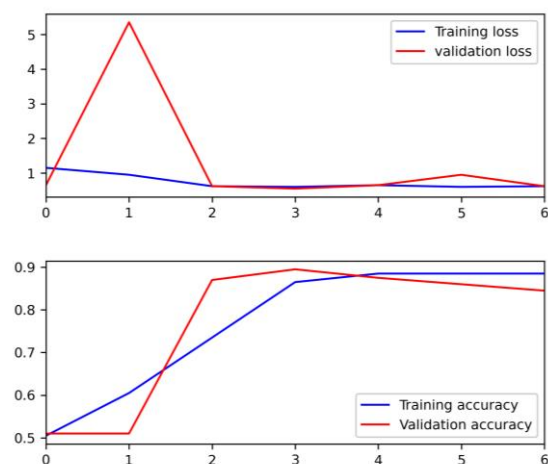


Fig. 9b: Loss and Accuracy of VGG 16 Model

In the bottom plot, blue Line (Training Accuracy) starts from 50% and steadily increases to approximately 86% by the end of training. This suggests the model is improved well its performance on the training dataset. The Red Line (Validation Accuracy) starts lower than the training accuracy but follow it with further training. At last, it declines a bit due to over fitting, or due to complexity of model.

For VGG 19 based Models:

- Accuracy: $TP+TN / TP+TN+FP+FN = 370+360 / 370+360+47+56 \approx 87.63\%$
- Precision = $TP / TP+FP = 370 / 370+47 = 393494 \approx 79.6\%$
- Recall (Sensitivity) = $TP / TP+FN = 393 / 393+33 = 393/426 \approx 92.2\%$
- F1 Score = $2 \cdot 0.796 \cdot 0.9220 / 0.796 + 0.922 = 2 \cdot 0.7331 \cdot 718 \approx 86.0\%$

From Complete Data set, total 2064 fake images were used for training and validation while 7437 real images were used for training and validation. The results for VGG 19 Model are:

- Total: 2064, Correct: 1867, Accuracy: 90.45542635658916
- Total: 7437, Correct: 6569, Overall Accuracy: 88.32862713459728
- Total: 9501, Correct: 8436, Overall Accuracy: 88.7906536154089

The plot in Fig. 10b shows the training and validation performance metrics (loss and accuracy) for a VGG 19 model over a training process spanning 2 epochs. The blue Line (Training Loss) starts higher (around 0.4) and steadily decreases as the epoch's progress, indicating that the model is learning to fit the training data better over time. Red Line (Validation Loss) loss starts lower than the training loss but increases as training progresses. This shows a bit of over-fitting, where the model tries to memorize the training data instead of generalizing well to testing data. In The bottom plot, blue Line (Training Accuracy) starts from 84% and steadily increases to approximately 88% by the end of training. This suggests the model is improved well its performance on the training dataset. The Red Line (Validation Accuracy) starts higher than the training accuracy but decreases with further training. The validation loss increases while validation accuracy decreases as the training progresses, indicating that the model may be over-fitting to training dataset or due to model complexity.

In sampled batch performance (Fig. 11), Ameliorate CNN achieves the highest accuracy (93.11%) and F1

score (93.02%), highlighting its ability to balance precision and recall effectively while VGG 16 based model stands out in recall (92.20%), demonstrating strength in detecting all relevant instances but has lower precision (79.60%) and VGG 19 based model offers balanced performance with good precision (88.74%) and recall (87.32%).

In comparing Real and Fake image (Fig. 12), Ameliorate CNN excels in real image detection with an accuracy of 97.87% while VGG 16 performs best on fake image detection (91.52%) and VGG 19 balances performance for real (90.46%) and fake (88.33%) images but falls short compared to Ameliorate CNN in real image detection.

As shown in Table 2, in overall accuracy, Ameliorate CNN achieves the highest overall accuracy (92.37%), showing its robustness and computational efficiency while VGG 16 based model (88.82%) and VGG 19 based model (88.79%) have comparable overall performance but require significantly more computational resources.

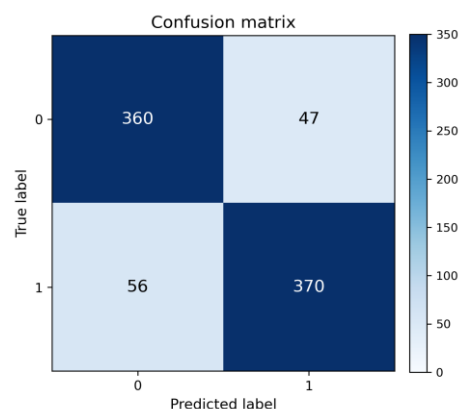


Fig. 10a: Confusion Matrix for Sampled Images of VGG 19 Model

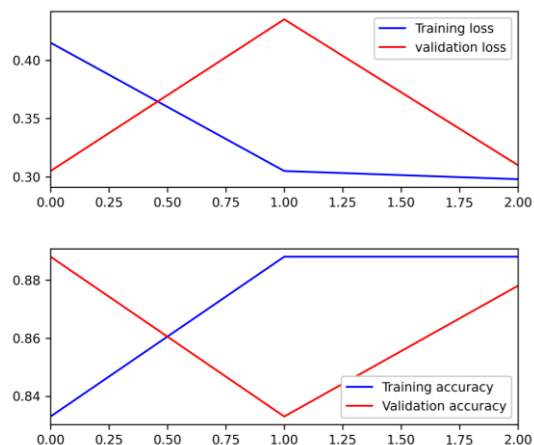


Fig. 10b: Loss and Accuracy of VGG 19 based Model

Model performance comparison — Ameliorate CNN vs VGG16 vs VGG19

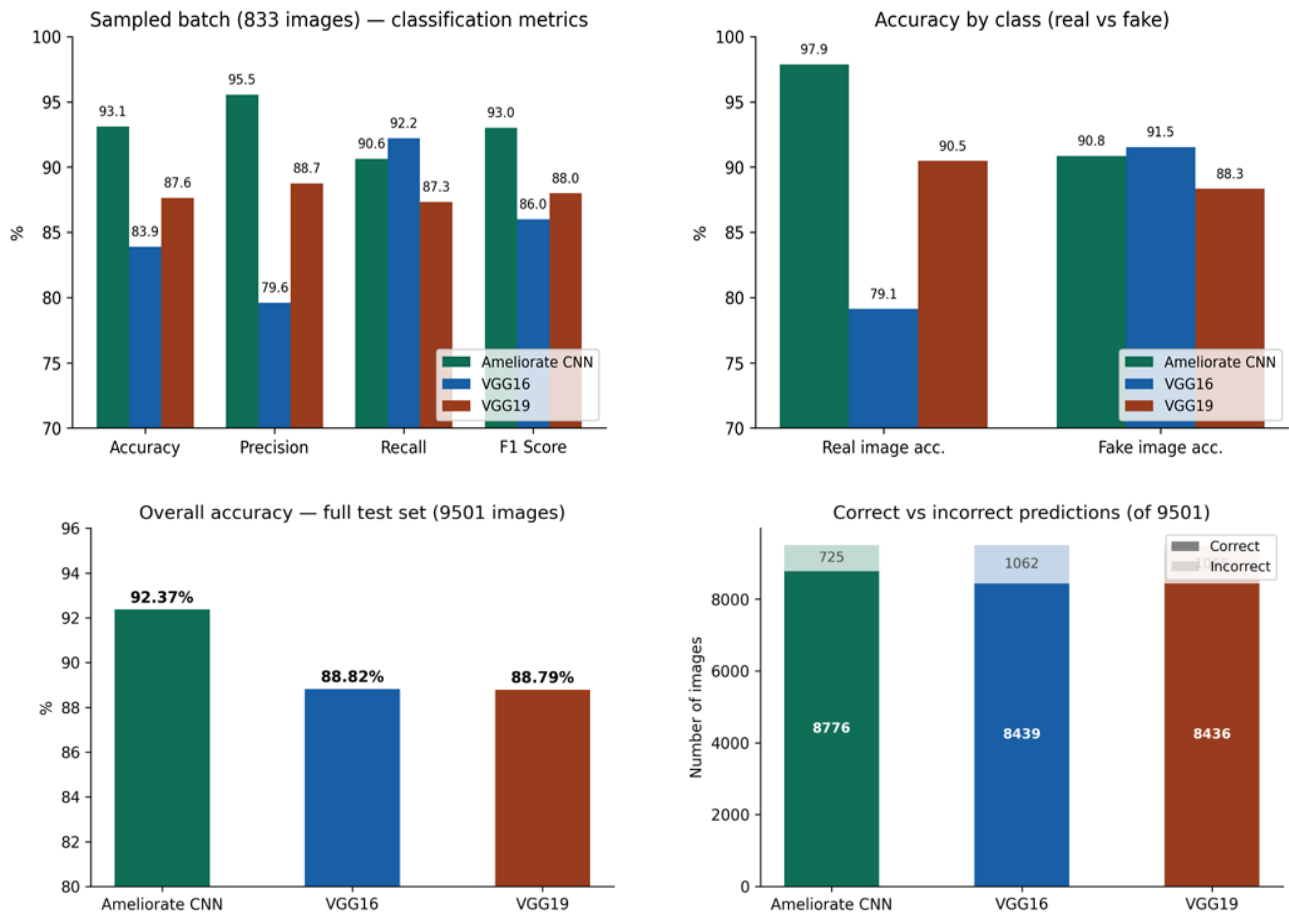


Fig. 11: Comparison of Performance matrices for sampled data, by class and Overall Accuracy with correct and incorrect predictions for all three models

Table 2: Comparison Table

Metric	Ameliorate CNN model	VGG 16 Model	VGG 19 Model
Accuracy for sampled batch of 833 images (%)	93.11	83.90	87.63
Precision for sampled batch of 833 images (%)	95.54	79.60	88.74
Recall for sampled batch of 833 images (%)	90.63	92.20	87.32
F1 Score for sampled batch of 833 images (%)	93.02	86.00	87.99
Total Real Images: 2064	Correct: 2020	Correct: 1633	Correct: 1867
Real Image Accuracy (%)	97.87	79.12	90.46
Total Fake Images: 7437	Correct: 6756	Correct: 6806	Correct: 6569
Fake Image Accuracy (%)	90.84	91.52	88.33
Total Images	9501	9501	9501
Correct Predictions	8776	8439	8436
Overall Accuracy (%)	92.37	88.82	88.79

Table 3: Comparison in Computation Time

Model	Training Time (s)	Real Image Detection Time (s)	Fake Image Detection Time (s)	Total Computation Time (s)
Ameliorate CNN	164.251	432.05	148.137	744.438
VGG 16 based	6474	432.387	164.406	7070.793
VGG 19 based	4255	665.917	1621.479	6542.396

The Ameliorate CNN model as shown in Table 3, has a relatively low training time (164.251s) compared to VGG models due to its lighter architecture and fewer parameters. This model processes real and fake images (432.05s and 148.137s) efficiently, leading to the lowest total computation time of 744.438 seconds. The VGG 16 based model has a significantly higher training time (6474s) due to its deeper architecture and larger number of parameters. Detection times are slightly higher than the CNN model but still manageable. The total computation time sums to 7070.793 seconds, reflecting the impact of its lengthy training. The VGG 19 based model has a slightly lower training time (4255s) compared to VGG 16, possibly due to optimization techniques or resource allocation. The detection times for VGG 19 are the highest, particularly for fake images, contributing to a total computation time of 6542.396 seconds.

Comparison in Computation Time

Key findings: Following key contributions in the field of Image forgery detection:

1. A lightweight, high-accuracy custom architecture i.e. Ameliorate CNN Model that outperforms two well-established transfer-learning baselines (VGG16 and VGG19) on real vs. fake image classification, achieving 92.37% overall accuracy, 95.54% precision, and 93.02% F1 score on a CASIA V2 complete dataset while requiring roughly 9 times less total computation time than either of VGG-based models. This demonstrates that a purpose-built, smaller architecture can match or exceed the accuracy of deeper pretrained networks for this task without their computational overhead
2. A systematic comparative evaluation of transfer learning vs. custom architectures for deepfake detection. The modified and customized versions of basic CNN, VGG16, VGG19 models under identical experimental conditions for reporting accuracy, precision, recall, F1 score, per-class (real/fake) accuracy, and computation time were analyzed and provided a reproducible baseline for future work to compare against. Also, it clarifies the specific trade-offs (e.g., VGG16's higher recall at the cost of precision and speed) that practitioners face when choosing between architectures for image forgery detection problem
3. Empirical evidence of the precision recall trade-off across architectures. Our per-class breakdown shows that VGG16 based model wrongly predict real images as fake (79.12% real-image accuracy) while achieving the best fake-detection recall (91.52%), whereas Ameliorate CNN minimizes both error types

simultaneously. These finding provides better guidance in terms of deployment contexts

4. A computation-efficiency analysis alongside accuracy metrics, which is often omitted in general deepfake-detection literature. By reporting training time as well as per-class inference time alongside classification performance, this work supports more realistic model-selection decisions for resource-constrained or real-time deployment scenarios, rather than optimizing for accuracy alone

Limitation

The reported results demonstrate strong performance under the evaluated experimental conditions but to establish robustness for all possible image manipulation scenarios and cross-dataset images may be included as future aspect.

Conclusion

It is concluded that Ameliorate CNN model is ideal for scenarios prioritizing accuracy and speed, particularly in real image detection. It is the fastest model overall due to its lightweight design, making it suitable for scenarios with time constraints. VGG 16 based model is preferable for applications emphasizing recall or fake image detection. VGG 19 based model offers a balanced approach but doesn't surpass the Ameliorate CNN model in overall effectiveness. Overall, VGG 16 and VGG 19 based models provides higher accuracy and feature extraction capabilities but require significantly more computation time, especially during training. It demonstrates that a customized lightweight Ameliorate CNN based model can outperform deeper pretrained architectures like VGG16 and VGG19 for real/fake image classification, achieving higher accuracy, precision, and F1 score while requiring substantially less computation time, making it a practical, resource-constrained deployment model. The comparative results exhibit a clear precision recall trade-off, offering useful guidance for selecting models based on minimizing false positives and/or false negatives for a given application. Future work could be extended to larger and more diverse datasets, incorporate newer architectures such as Vision Transformers or EfficientNet variants.

Acknowledgment

The First author acknowledges the support and guidance provided by the institution Guide (Co-author) for the completion of research work.

Funding Information

The authors declare that for this research work, no external funding was received.

Author's Contributions

The first author worked on the conceptualization, implementation, analysis, and preparation of this manuscript under the guidance of Co-author.

Ethics

The authors declare that manuscript work is original and free from ethical issues/ conflicts of interest.

References

- Ali, S. S., Ganapathi, I. I., Vu, N.-S., Ali, S. D., Saxena, N., & Werghi, N. (2022). Image Forgery Detection Using Deep Learning by Recompressing Images. *Electronics*, 11(3), 403.
<https://doi.org/10.3390/electronics11030403>
- Atak, I. G., & Yasar, A. (2024). Image forgery detection by combining Visual Transformer with Variational Autoencoder Network. *Applied Soft Computing*, 165, 112068.
<https://doi.org/10.1016/j.asoc.2024.112068>
- Babu, S. B. G. T., & Rao, C. S. (2023). Efficient detection of copy-move forgery using polar complex exponential transform and gradient direction pattern. *Multimedia Tools and Applications*, 82(7), 10061–10075. <https://doi.org/10.1007/s11042-022-12311-6>
- Bay, H., Ess, A., Tuytelaars, T., & Gool, L. V. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3), 346–359.
<https://doi.org/10.1016/j.cviu.2007.09.014>
- Castillo, I. C., & Wang, K. (2021). A Comprehensive Review of Deep-Learning-Based Methods for Image Forensics. *Journal of Imaging*, 7(4), 69.
<https://doi.org/10.3390/jimaging7040069>
- Diallo, B., Urruty, T., Bourdon, P., & Fernandez-Maloigne, C. (2020). Robust forgery detection for compressed images using CNN supervision. *Forensic Science International: Reports*, 2, 100112.
<https://doi.org/10.1016/j.fsir.2020.100112>
- Fang, W., Love, P. E. D., Luo, H., & Ding, L. (2020). Computer vision for behaviour-based safety in construction: A review and future directions. *Advanced Engineering Informatics*, 43, 100980.
<https://doi.org/10.1016/j.aei.2019.100980>
- Farid, H. (2009). Image forgery detection. *IEEE Signal Processing Magazine*, 26(2), 16–25.
<https://doi.org/10.1109/msp.2008.931079>
- Fridrich, J., Soukal, D., & Lukas, J. (2003). Detection of copy-move forgery in digital images. *Proceedings of the Digital Forensic Research Workshop*. DOI: Not Available, 5–8.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA. <https://doi.org/10.1109/cvpr.2014.81>
- Hayat, K., & Qazi, T. (2017). Forgery detection in digital images via discrete wavelet and discrete cosine transforms. *Computers & Electrical Engineering*, 62, 448–458.
<https://doi.org/10.1016/j.compeleceng.2017.03.013>
- Johnson, M. K., & Farid, H. (2006). Exposing digital forgeries through chromatic aberration. *Proceedings of the 8th Workshop on Multimedia and Security*, 48–55. <https://doi.org/10.1145/1161366.1161376>
- Johnson, M. K., & Farid, H. (2008). Detecting Photographic Composites of People. *Digital Watermarking*, 19–33. https://doi.org/10.1007/978-3-540-92238-4_3
- Li, H.-C., Deng, Z.-Y., & Chiang, H.-H. (2020). Lightweight and Resource-Constrained Learning Network for Face Recognition with Performance Optimization. *Sensors*, 20(21), 6114.
<https://doi.org/10.3390/s20216114>
- Liang, P., Li, Z., Tu, H., & Zhao, H. (2024). LBRT: Local-Information-Refined Transformer for Image Copy-Move Forgery Detection. *Sensors*, 24(13), 4143. <https://doi.org/10.3390/s24134143>
- Lyu, S., & Farid, H. (2005). How realistic is photorealistic? *IEEE Transactions on Signal Processing*, 53(2), 845–850.
<https://doi.org/10.1109/tsp.2004.839896>
- Mallick, D., Shaikh, M., Gulhane, A., & Maktum, T. (2022). Copy Move and Splicing Image Forgery Detection using CNN. *ITM Web of Conferences*, 44, 03052. <https://doi.org/10.1051/itmconf/20224403052>
- Meena, K. B., & Tyagi, V. (2023). Image splicing forgery detection using noise level estimation. *Multimedia Tools and Applications*, 82(9), 13181–13198. <https://doi.org/10.1007/s11042-021-11483-x>
- Mehrjardi, F. Z., Latif, A. M., Zarchi, M. S., & Sheikhpour, R. (2023). A survey on deep learning-based image forgery detection. *Pattern Recognition*, 144, 109778.
<https://doi.org/10.1016/j.patcog.2023.109778>
- Nikolaidis, N., & Pitas, I. (2002). Copyright protection of images using robust digital signatures. *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, 2168–2171.
<https://doi.org/10.1109/icassp.1996.545849>

- Palaz, D., Magimai-Doss, M., & Collobert, R. (2019). End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition. *Speech Communication*, 108, 15–32. <https://doi.org/10.1016/j.specom.2019.01.004>
- Piva, A. (2013). An Overview on Image Forensics. *ISRN Signal Processing*, 2013, 1–22. <https://doi.org/10.1155/2013/496701>
- Popescu, A. C., & Farid, H. (2005). Exposing digital forgeries by detecting traces of resampling. *IEEE Transactions on Signal Processing*, 53(2), 758–767. <https://doi.org/10.1109/tsp.2004.839932>
- Qazi, E. U. H., Zia, T., & Almorjan, A. (2022). Deep Learning-Based Digital Image Forgery Detection System. *Applied Sciences*, 12(6), 2851. <https://doi.org/10.3390/app12062851>
- Ramamoorthi, R., & Hanrahan, P. (2001). On the relationship between radiance and irradiance: determining the illumination from images of a convex Lambertian object. *Journal of the Optical Society of America A*, 18(10), 2448. <https://doi.org/10.1364/josaa.18.002448>
- Tian-Tsong, N., & Chang, S.-F. (2005). A model for image splicing. *2004 International Conference on Image Processing, 2004. ICIP '04.*, 1–25. <https://doi.org/10.1109/icip.2004.1419512>
- Yang, B., Sun, X., Guo, H., Xia, Z., & Chen, X. (2018). A copy-move forgery detection method based on CMFD-SIFT. *Multimedia Tools and Applications*, 77(1), 837–855. <https://doi.org/10.1007/s11042-016-4289-y>
- Yu, Z., Ni, J., Zhang, J., Deng, H., & Lin, Y. (2025). Reinforced Multi-teacher Knowledge Distillation for Efficient General Image Forgery Detection and Localization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(1), 995–1003. <https://doi.org/10.1609/aaai.v39i1.32085>