Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 10, No. 3, June, 2026

Detecting Fake Images Generated by Artificial Intelligence Using Deep Learning Approach

Ahmet E. Topcu [1], Yehia I. Alzoubi [2*], Emre Camalan [3], Ersin Elbasi [1], Mohammad K. I. AlQallaf [1]

[1] *College of Engineering and Technology, American University of the Middle East, Egaila 54200, Kuwait.*

[2] *College of Business Administration, American University of the Middle East, Egaila 54200, Kuwait.*

[3] *Department of Computer Science Engineering, Işık University, İstanbul, Turkey.*

Abstract

The rapid progress in artificial intelligence has enabled the creation of highly realistic images, leading to concerns about the credibility and genuineness of visual content. This study aims to address the growing challenge of verifying the authenticity of digital images in the era of advanced generative artificial intelligence by developing an effective method to detect AI-generated (deepfake) images. To achieve this objective, we employed a Machine Learning (ML) framework based on Convolutional Neural Networks (CNNs), evaluating five established architectures, VGG19, ResNet50, Xception, DenseNet-121, and InceptionV3, through a systematic pipeline involving dataset compilation, image preprocessing, feature extraction, model training, and rigorous validation. Our experimental analysis demonstrates that DenseNet-121 and InceptionV3 achieve state-of-the-art performance, both attaining 98% accuracy in distinguishing AI-synthesized images from real ones, despite a non-negligible error rate observed in other models. These findings highlight the viability of CNN-based approaches for reliable deepfake detection. The novelty of this work lies in its comparative assessment of multiple CNN architectures on a curated dataset of AI-generated imagery, offering practical insights into model selection for forensic and security applications. The proposed method contributes a robust, scalable solution with significant implications for digital content moderation, cybersecurity, and multimedia forensics, where timely and accurate identification of synthetic media is increasingly critical.

Keywords:

Image;
Deepfake;
Artificial Intelligence;
Machine Learning;
Convolutional Neural Networks.

Article History:

Received:	17	July	2025
Revised:	24	May	2026
Accepted:	27	May	2026
Published:	01	June	2026

1- Introduction

Artificial Intelligence (AI) has made remarkable progress recently, notably by developing sophisticated ML algorithms [1, 2]. These advancements have empowered the generation of remarkably lifelike images. Examples of AI-generated images include projects like Midjourney, Dall-E, Stable Diffusion, and ChatGPT [3, 4]. These initiatives have produced images that closely mimic those captured by human photographers. The evolution of AI technology has introduced exciting possibilities in creative realms, spanning virtual reality, high-resolution imagery, and the rejuvenation of historical visuals [5]. However, this surge in AI's capabilities also brings about substantial concerns surrounding the potential misuse of AI-generated images [6]. Such images could be harnessed to fabricate deceitful or harmful content, such as forged news stories, manipulative social engineering schemes, and deepfake videos [7]. Furthermore, AI-generated images have the capacity to invent visual depictions and identities of entirely fictional individuals, thereby amplifying the risk of identity-related fraud.

* **CONTACT**: yehia.alzoubi@aum.edu.kw

DOI: https://doi.org/10.28991/ESJ-2026-010-03-07

An illustrious illustration of AI-generated images can be found in DALL-E, an AI system crafted by OpenAI [8]. This program crafts images guided by textual descriptions. DALL-E's capabilities span an extensive spectrum, encompassing lifelike entities and even surreal vistas [9]. These images are generated in accordance with the provided textual input. To illustrate, DALL-E can conjure up visuals like a harp-fashioned snail, a teapot atop a stack of pancakes, or a turtle adorned with a pattern resembling a traffic light, all in response to descriptive prompts. Another case in point is MidJourney, an AI application devised by a group of researchers at MIT [10]. This program operates by scrutinizing a provided image and subsequently manipulating it to generate a fresh image that retains similarities while introducing distinct variations from the original. MidJourney showcases its prowess by producing an array of images, spanning from lifelike portraits to abstract motifs, all derived from a solitary input image.

While these instances highlight the captivating potential of AI-generated images and text, they also underscore apprehensions regarding their possible misuse [11]. AI-generated content, encompassing images and text, can craft deceptive news, manipulate public sentiment, and perpetrate fraudulent activities. Hence, the urgency to cultivate dependable techniques for identifying AI-generated content is escalating to forestall its exploitative utilization [7]. The proliferation of AI-generated images has brought about anxieties concerning its potential societal implications [12]. For example, these images could be leveraged to manipulate public sentiment or fabricate false evidence in legal proceedings, engendering grave repercussions [13]. Consequently, a mounting demand for trustworthy methodologies to discern AI-generated images is surfacing, intended to thwart their malevolent deployment.

Recent studies on deepfake detection predominantly frame the problem as a binary classification task, using CNN-based models to differentiate authentic from fabricated images or videos [14]. These approaches effectively identify visual or temporal inconsistencies but rely heavily on large, labeled datasets [15] and show declining performance in cross-dataset evaluations [16]. Although several methods have achieved high accuracy, such as quaternion-based image forensics [17], deep face recognition [16], and advanced CNNs like InceptionResNetV2 [11], they remain limited by dataset dependency, poor generalization, and minimal robustness testing. Recent works, including Huang et al. [18] and OpenAI [9], introduced fusion and graph-based architectures that improved performance but increased model complexity. Similarly, previous studies [19-21] emphasized that most research still optimizes for accuracy alone, overlooking explainability, adversarial resistance, and real-world image degradations. The newest literature, [22-26], further emphasized generalization challenges, computational inefficiency, and the lack of cross-modal evaluation, reinforcing the need for a standardized comparative framework.

To address these gaps, the present study compares five CNN architectures (VGG19, ResNet50, Xception, DenseNet-121, and InceptionV3) under identical conditions using multiple evaluation metrics and cross-dataset testing. By integrating Explainable AI (XAI) visualization to enhance interpretability, this research provides a transparent and generalizable framework. The findings reveal that DenseNet-121 and InceptionV3 achieve 98% accuracy, demonstrating superior robustness and validating the study's contribution to advancing reliable detection of AI-generated images. Accordingly, the novelty of this study lies in its adoption of this approach, utilizing both image processing techniques and ML algorithms to tackle deepfake images. By providing a pragmatic solution for identifying AI-generated images, this research significantly contributes to developing reliable tools to prevent the potential misuse of this technology. The following are the contributions of this study.

- Deepfake techniques continually evolve, leading to various creation methods and quality levels. This study's approach combines image processing and ML, making detection models more robust. They can adapt to different deepfake generation techniques and varying image qualities, ensuring that deepfakes are detected across various scenarios.
- Image processing techniques can extract fine-grained features and anomalies in images that might not be evident to the naked eye. Using these techniques in conjunction with Deep Learning (DL) algorithms enables the creation of extremely precise detection models. Deepfake images can include subtle patterns and discrepancies that the ML model can learn to recognize, increasing the overall detection accuracy.
- This approach has the potential to produce real-time or near-real-time deepfake detection, which is essential for stopping the online spread of false information and fraudulent content.
- Image processing techniques can preprocess images to identify probable signs of deepfakes and lessen the computational load on DL algorithms. This preprocessing can aid feature extraction, thereby increasing the effectiveness of subsequent ML analysis.
- This combination enables transdisciplinary discoveries by bridging the gaps between image processing, computer vision, and DL. Researchers from these domains may collaborate to create more powerful deepfake detection tools, leading to creative solutions and a better comprehension of the issue.
- This approach may be expanded to identify deepfakes in various media, such as text, audio, and video. Due to its versatility, it can be used in various situations to address broader problems caused by AI-generated content.

The rest of this paper is organized as follows. Section 2 provides background on the challenges posed by deepfake images and evaluates current academic writing on the subject. In Section 3, the technique used in this study is explained, and a plan to address the challenges of identifying deepfake photos is outlined. The code developed for this investigation is presented in Section 4, along with an example of the results obtained when the algorithm is applied to real-world data. The conclusions and revelations gathered from this investigation are discussed in Section 5. Finally, Section 6 reviews the findings, highlights the most critical lessons learned, and suggests future research directions.

2- Background and Related Literature

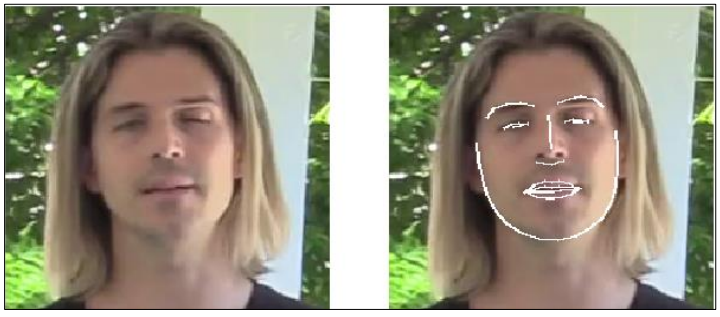
This section discusses the background problem of deepfake images created by AI to shed light on the research problem. It also discusses the power of ML in addressing sophisticated issues such as deepfake images. Moreover, it discusses recent related literature and compares it to the focus of this study.

2-1- Deepfake Images Generated by Artificial Intelligence

The ability to discern images generated by AI has emerged as a significant concern across diverse domains, including online content moderation, security systems, and forensic inquiries [27]. To tackle this challenge, researchers have proposed various strategies to differentiate between AI-generated images and those produced by humans. Among these strategies, two promising avenues involve using image-processing techniques and ML algorithms [28]. Image processing techniques hold the potential to extract pertinent attributes from images, facilitating the recognition of distinctive patterns inherent in AI-generated images. In tandem, ML algorithms can be deployed to categorize images based on these discernible features.

A key hurdle in identifying AI-generated images is their ability to mimic human-created images, making them hard to distinguish visually. However, recent studies have highlighted certain traits inherent to AI-generated images, setting them apart from their human-crafted counterparts [27]. One such example is the absence of artifacts or patterns in human-generated images. These unique patterns can be revealed through image processing techniques that analyze an image's structure and texture to extract distinctive attributes. An equivalent source to the https://ai.facebook.com/ site is obtained from the website, as illustrated in Figure 1. In this given instance, the depicted countenance resembles an authentic face. Yet, the question arises: How does one discern the authenticity of this image? Initially, it has been established that the regions around the mouth, eyes, and nose are key areas for distinguishing genuine from fabricated faces.

(a) Real vs fake image



(b) Fake image vs analysis of fake image

Figure 1. Fake images created by AI and analysis of fake images using face measurements [29]

2-2- Related Literature

Recent research has explored diverse strategies for detecting AI-generated and deepfake images, adopting both traditional image-processing approaches and advanced DL architectures. Across studies, detection performance has improved significantly, but methodological diversity and dataset dependence remain high. Table 1 summarizes the literature's findings and limitations.

Table 1. Findings of the previous research.

Study	Models used	Accuracy	Key findings	Limitations
Wang et al. [17]	Quaternion Central Moments in CQWT	Improved by 19% over Farid's model	CQWT processes color images as a unit, providing more forensic information.	Limited comparison with other state-of-the-art models.
Khalil et al. [30]	Capsule Networks within LSTM	Not specified	Effective in detecting subtle spatial and temporal inconsistencies.	Requires further validation on diverse datasets.
Ramachandran et al. [16]	Deep Face Recognition	AUC of 0.99, EER of 2.04% on FaceForensics++	Deep face recognition outperforms two-class CNNs and ocular modality.	Requires biometric facial recognition technology.
Kolagati et al. [31]	MLP-CNN, LSTM	Not specified	Proposed a hybrid model using MLP-CNN and LSTM.	Limited dataset evaluation (WLDR and DeepfakeTIMIT.
Hasan Abir et al. [11]	InceptionV3, ResNet152V2, DenseNet201, InceptionResNetV2	99.87% (InceptionResNetV2)	Effective advanced CNN models	Generalization not addressed
Huang et al. [18]	CNN, RNN, EfficientNet, WSDAN	Not specified	Introduced FG-TEFusionNet model combining facial geometry and texture features.	Potential complexity in model integration.
Tan et al. [9]	Facial Action Dependency Estimation (FADE)	Not specified	MDGM captures dependencies among facial action units, improving detection performance.	Requires extensive experiments for validation.
Byeon et al. [21]	GANs, CNNs	Not specified	GANs are effective in generating realistic images that are challenging to detect.	CNN-based solutions are complex and time-consuming for training.
Heidari et al. [19]	Various DL models (e.g., CNNs, GANs)	Not specified	DL models show superior accuracy compared to other methods.	Room for improvement in detecting sophisticated and realistic deepfakes.
Kumar et al. [32]	MLP, CNN, LSTM	Not specified	Introduced IDL-DDM model combining MLP, CNN, and LSTM.	Limited information on specific performance metrics.
Passos et al. [20]	CNNs, LSTMs, Hybrid models	Up to 94% in controlled environments	Hybrid models combining CNNs and LSTMs.	High computational cost.
Singh et al. [33]	CNN, Xception, DenseNet-121, EfficientNetB0	96.28% (CNN)	High accuracy with CNN	Variability in model performance
Almalki et al. [23]	VGG16, VGG19, MobileNetV2, ResNet50V2	98.4% (VGG16-based)	Transfer learning potential	Limited dataset comparison
Jin et al. [24]	Swin-Transformer, ResNet-18	Over 99%	High accuracy with Swin-Transformer	Specific image set reliance
Kumar et al. [25]	MobileNetV2, CAFFE block, CMNV2	CMNV2: 99.10%	The CMNV2 model demonstrated superior resilience against complex real-world conditions	High performance across diverse and evolving deepfake techniques
Lagsoun et al. [22]	ResNet50, Random Forest, KNN	97.3% (Random Forest)	Effective for low-quality data	Focus on low-quality data
Wani et al. [26]	VGGFace16, DenseNet-121, Custom CNN	DenseNet-121: 97%	Dataset selection significantly affects model performance	Resource-efficient training is necessary for practical applications

2-2-1-Overview and Key Findings of Reviewed Studies

Wang et al. [17] utilized quaternion central moments within the Color Quaternion Wavelet domain (CQWT), improving classification accuracy by 19% over Farid's model. Their approach effectively captures color information as a unified entity, offering rich forensic cues for distinguishing authentic from synthetic images. However, the study primarily focused on color images, limiting generalization to other image types. Furthermore, Khalil et al. [30] proposed a framework that combines Capsule Networks with LSTM architectures to detect subtle spatial and temporal inconsistencies in deepfakes. Their model was promising for real-time detection tasks, but required validation on larger, more diverse datasets because the performance metrics were not explicitly stated. Also, Ramachandran et al. [16] evaluated deep face recognition for deepfake detection using challenging datasets like Celeb-DF and FaceForensics++. Their model achieved an AUC of 0.99 and an EER of 2.04%, outperforming two-class CNNs and ocular-based methods. However, performance dropped during cross-dataset evaluations, reflecting overfitting to specific datasets.

Moreover, Kolagati et al. [31] introduced a hybrid model integrating MLP-CNN and LSTM layers, enhanced with CLAHE for contrast improvement and VJA for face detection. Although it outperformed models like DeepVision, DNN, and RNN in limited tests, accuracy values were not reported, and evaluation was restricted to only the WLDR and DeepfakeTIMIT datasets. On the other hand, Hasan Abir et al. [11] combined multiple CNNs (InceptionV3, ResNet152V2, DenseNet201, and InceptionResNetV2) with XAI for deepfake detection. The InceptionResNetV2 model achieved 99.87% accuracy, confirming the power of advanced CNN architectures. Yet, the study did not

investigate generalization across datasets, raising questions about robustness. Additionally, Huang et al. [18] proposed FG-TEFusionNet, integrating CNN, RNN, EfficientNet, and WSDAN networks to combine facial geometry and texture features. Their model, enhanced with diffusion-based denoising and IMTCCN feature extraction, demonstrated resilience to adversarial examples. Despite its robustness, model complexity and integration cost were major drawbacks.

Tan et al. [9] redefined deepfake detection as a graph classification problem using a Multi-Dependency Graph Module (MDGM), capable of capturing interdependencies among facial action units. Their framework improved detection accuracy and integrated seamlessly into existing systems, but further experimental validation was required. Kumar et al. [32] compared various ML methods for deepfake detection using visual deepfake datasets, finding that CNN-based approaches generally achieved higher accuracy and efficiency. Similarly, Heidari et al. [19] analyzed several DL algorithms and concluded that CNNs remain the most prevalent and effective method for deepfake detection. However, both studies observed that most research focuses narrowly on improving a single metric (accuracy), neglecting robustness and generalization. Passos et al. [20] reviewed DL-based deepfake detection techniques, identified critical gaps, and recommended future research directions in explainability, dataset diversity, and adversarial resilience. Byeon et al. [21] proposed a spatial–frequency dual-stream CNN incorporating learnable frequency-domain filtering kernels to capture artifacts left by GANs. Their results indicated improved generalization across datasets, but model performance declined significantly under OSN compression, and training remained computationally demanding. Collectively, these studies confirm that CNN-based models dominate deepfake detection, achieving remarkable performance (often exceeding 95% accuracy). However, they also show limited generalization beyond the datasets they were trained on.

2-2-2- Limitations in Previous Research

Despite considerable progress in AI-generated image detection, the existing literature reveals several notable limitations that constrain the generalizability and applicability of its findings. A recurring concern is the strong dataset dependency observed across studies, where models are predominantly trained and tested on the same datasets, leading to overfitting and limited cross-domain robustness. For instance, Ramachandran et al. [16] acknowledged that although their model achieved impressive results on the FaceForensics++ dataset, its performance declined significantly in cross-dataset evaluations. Similarly, Hasan Abir et al. [11] and Kolagati et al. [31] restricted their assessment to a narrow range of datasets, limiting the understanding of their models' behavior on unseen or diverse image sources. This lack of cross-dataset testing undermines confidence in the generalization of detection models beyond benchmark datasets. Another key limitation lies in the incomplete and inconsistent reporting of performance metrics. Many studies emphasize accuracy as the primary indicator of success while neglecting other crucial evaluation parameters such as precision, recall, F1-score, and confusion matrices. For example, Khalil et al. [30] and Kolagati et al. [31] did not provide explicit accuracy values, whereas others, such as Heidari et al. [19] and Hasan Abir et al. [11], focused solely on accuracy. This practice limits comparative analysis across studies and obscures trade-offs between false-positive and false-negative rates, which are essential in forensic and operational contexts.

In addition, most existing work focuses narrowly on GAN-generated content, disregarding the rapid evolution of generative technologies, such as diffusion and transformer-based models. As a result, detectors trained on older generative adversarial network (GAN) artifacts often struggle to recognize newer forms of synthetic imagery, revealing a substantial generalization gap. Byeon et al. [21] Also observed that performance deteriorates considerably under real-world distortions, such as compression or scaling, a condition largely ignored by earlier research. This highlights the persistent lack of consideration for real-world degradation effects, such as social media compression, image resizing, and noise contamination, which are unavoidable in practical applications. Furthermore, comparative architectural analysis among mainstream CNN models remains limited. Although numerous studies use CNN variants such as VGG, ResNet, DenseNet, and Xception, they are rarely evaluated under identical experimental conditions, making it difficult to draw definitive conclusions about which architectures offer superior robustness and efficiency. This issue is compounded by the limited integration of interpretability mechanisms, as only Hasan Abir et al. [11] incorporated XAI to visualize detection decisions. The majority of models operate as black boxes, providing no clear forensic evidence to justify classifications.

Another common shortcoming is dataset bias, as face images of limited demographic and environmental diversity dominate most detection datasets. This homogeneity introduces unintentional biases that reduce model reliability when applied to non-facial or culturally varied imagery. At the same time, the computational cost of training large CNN architectures, as reported by Huang et al. [18] and Byeon et al. [21], raises practical concerns for real-time or large-scale deployment. Lastly, adversarial robustness remains underexplored, with only Huang et al. [18] partially addressing it through adversarial sample testing. Collectively, these limitations underscore that while existing research has made significant strides in model accuracy, it still lacks methodological consistency, interpretability, robustness evaluation, and real-world applicability.

2-2-3-Value and Contribution of the Present Study

By evaluating five mainstream CNN architectures, VGG19, ResNet50, Xception, DenseNet-121, and InceptionV3, under identical experimental conditions, this research offers a rare head-to-head comparative analysis that is largely absent in prior literature. Unlike previous studies that focused on a single model or used differing preprocessing and evaluation criteria, this study maintains consistent data handling, training configurations, and validation metrics, allowing for a fair and reproducible comparison across architectures. Moreover, the current research moves beyond the accuracy-only focus criticized in earlier studies, such as Heidari et al. [19], and instead reports multiple performance indicators, including accuracy, precision, recall, AUC, and confusion matrices. In addition to performance evaluation, this study incorporates XAI techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) to visualize and interpret model predictions, providing valuable forensic transparency. Finally, the study empirically demonstrates that DenseNet-121 and InceptionV3 achieved the highest accuracy, reaching 98% and outperforming the other evaluated models. These results align with previous findings, such as those reported by Hasan Abir et al. [11], but the key distinction lies in the methodological rigor and direct comparability of the models tested here. This comparative approach allows clearer insight into how architectural depth, connectivity patterns, and feature extraction strategies influence detection capability.

3- Research Methodology

ML algorithms can be trained to discern facial attributes and categorize images into two categories: those generated by AI and those crafted by humans. A widely used strategy involves applying CNNs. These networks can be trained on substantial image datasets to recognize distinctive characteristics and traits, distinguishing AI-generated images from their human-created counterparts. Subsequently, once trained, the CNN can accurately differentiate novel images, classifying them as either AI-generated or human-generated with notable precision. Figure 2 outlines the research methodology.

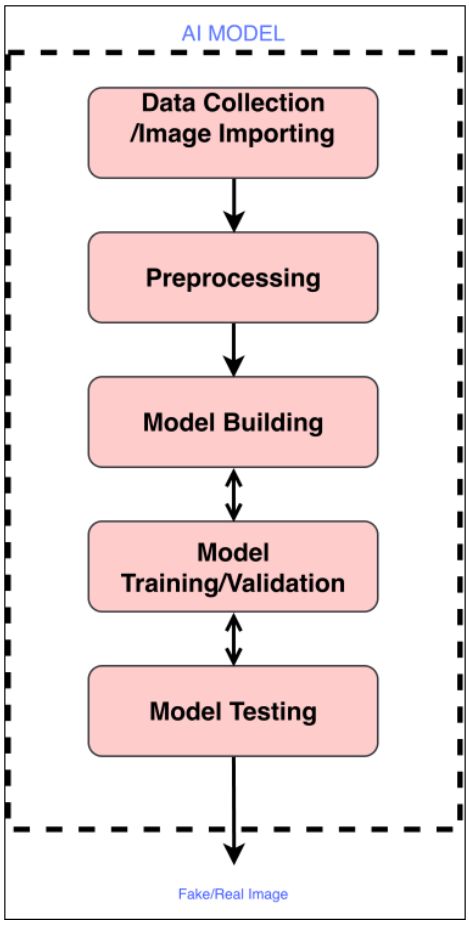


Figure 2. Paper methodology

This document presents a methodology for identifying AI-generated images that combines image processing methods and ML algorithms. This approach utilizes a pre-existing Keras module library, including VGG19, Resnet50, Xception, Densenet121, and InceptionV3 [34]. The dataset is organized into three directories: Train, Validation, and Test, as shown in Table 2. Each directory contains images labelled "Real" or "Fake". The total

number of images is 190,335 [35]. These images are processed using the ImageDataGenerator class from TensorFlow to facilitate model training, validation, and testing [36]. Each image is a 256×256 JPG of a human face, either real or fake [36].

Table 2. Dataset description.

	Train		Test		Validation	
Type	**Real**	**Fake**	**Real**	**Fake**	**Real**	**Fake**
Number of images	70001 (37%)	70001 (37%)	19787 (10.4%)	19641 (10.3%)	5413 (2.8%)	5492 (3.1%)

3-1- Data Import and Preparation

An inclusive dataset containing a wide array of images was curated, encompassing both those generated by AI and those crafted by humans. This dataset is deliberately varied, encompassing a range of subjects and styles to ensure the robustness and applicability of this method. Before analysis, the images undergo preparation procedures to improve their quality, reduce noise, and remove artifacts. Augmentation techniques such as rotation, width and height shifts, shear, zoom, and horizontal flipping are applied for the training set. This is done to increase the diversity of the training data and help the models generalize better. All images across the training, validation, and test sets are normalized by rescaling their pixel values to a [0, 1] range. Figure 3 depicts the algorithm used in this study to import and prepare the dataset for testing.

1. Initialize ROOT as tk.Tk() and set the title to "Model Predictions GUI".

2. Create UPLOAD BUTTON with text "Upload Image" and command update_predictions, then pack with padding (pady=20).

3. Create IMAGE LABEL and pack it.

4. Create PREDICTIONS LABEL with an empty text and left justification, then pack with padding (pady=20).

5. Start GUI loop with ROOT.mainloop().

Figure 3. Image importing

The dataset used in this study comprises two balanced classes, Real and Fake, with an equal number of samples per class in the training set (70,001 per class). The same augmentation parameters were uniformly applied to all training images, ensuring that both classes underwent identical transformations. Because of this equal class distribution and consistent augmentation procedure, the dataset remained balanced after augmentation, eliminating potential bias toward either class or ensuring fair model learning.

3-2- CNN Training

The CNN is trained using features extracted from images as input, along with corresponding labels indicating whether each image originates from AI-generated or human-generated sources, as depicted in Figure 4. During training, the CNN learns to recognize distinctive patterns and characteristics that distinguish AI-generated images from those created by humans. To improve the model's generalization, data augmentation techniques can be employed to expand the training dataset effectively.

This study leverages transfer learning by fine-tuning pre-trained models (i.e., DenseNet121, Xception, VGG19, ResNet50, and InceptionV3) for deepfake detection. Transfer learning enables these models to leverage pre-trained weights from the ImageNet dataset while adapting to new task-specific data. The following settings were deployed in this study:

- ***Architecture:*** GlobalAveragePooling2D was used, reducing the spatial dimensions of the feature maps to a single vector per image. Dense layers were deployed, and two fully connected layers were added, each followed by dropout and batch normalization to improve regularization and reduce overfitting. The output layer, which represents the final dense layer with a sigmoid activation function, outputs the probability that an image is "Real" or "Fake".
- ***Training configuration:*** The top layers of each pre-trained model are unfrozen for fine-tuning, allowing the models to adapt their high-level features to the deepfake detection task. The models are compiled using the Adam optimizer with a low learning rate to facilitate finetuning without significant weight updates that could disrupt the learned features.

In this study, no cross-validation technique (such as K-fold or stratified cross-validation) was employed. Instead, the dataset was divided into three independent subsets: training, validation, and testing. To mitigate overfitting, several regularization strategies were implemented, including *EarlyStopping*, *ReduceLROnPlateau*, *Dropout*, *Batch Normalization*, and extensive *data augmentation*. Hyperparameters such as the learning rate, batch size (32), number of dense layers, and dropout rate (0.5) were selected manually through iterative experimentation and observation of validation performance. Automated hyperparameter optimization techniques, such as grid search or random search, were not utilized in this study.

Function: CreateModel(BASE_MODEL)

1. **Freeze layers** 0-99, unfreeze 100+.
2. **Feature extraction:** Apply GlobalAveragePooling2D(BASE_MODEL_OUTPUT).
3. **Add classification layers**:
   - Dense(2048, ReLU) → Dropout(0.5) → BatchNormalization()
   - Dense(1024, ReLU) → Dropout(0.5) → BatchNormalization()
4. **Output layer:** Dense(1, Sigmoid).
5. **Create Model:** MODEL = Model(BASE_MODEL_INPUT, PREDICTIONS).
6. **Compile:** Adam (LR=0.0001), BinaryCrossentropy, Accuracy.
7. **Return MODEL**.

2. **Define Base Models**

1. **BASE_MODELS =** [DenseNet121, Xception, VGG19, ResNet50, InceptionV3], all with weights=imagenet, include_top=False, input_shape=(256,256,3).

3. **Train or Load Models**

For each BASE_MODEL in BASE_MODELS:

1. **Set paths**:
   - MODEL_PATH_KERAS = SAVE_DIR + MODEL_NAME + ".keras"
   - MODEL_PATH_H5 = SAVE_DIR + MODEL_NAME + ".h5"
2. **If model exists**: Load MODEL_PATH_KERAS, print "Loading MODEL_NAME".
3. **Else:**
   - **Print** "Training MODEL_NAME", set MODEL = CreateModel(BASE_MODEL).
   - **Define** Callbacks:
     - CHECKPOINT: Save best val_loss.
     - EARLY_STOPPING: Stop after 10 epochs if no improvement.
     - REDUCE_LR: Reduce LR by 0.2 if no improvement for 5 epochs.
     - METRICS_CALLBACK: Track additional metrics.
   - **Train model:** MODEL.fit(TRAIN_GENERATOR, steps_per_epoch, VAL_GENERATOR, validation_steps, epochs=10, callbacks=[CHECKPOINT, EARLY_STOPPING, REDUCE_LR, METRICS_CALLBACK]).
   - **Save model:** MODEL.save(MODEL_PATH_KERAS), MODEL.save(MODEL_PATH_H5).

Figure 4. CNN model building and training

4- Results

4-1- Model Testing

Figure 5 depicts the code used to test the images. Relevant features are extracted from pre-processed images, primarily to distinguish between AI-generated and human-generated images. These features capture unique characteristics of AI-generated images, such as artifacts or patterns not typically present in human-generated images. Texture descriptors, frequency domain representations, and visual artifacts were used in this context. Texture analysis involves extracting features that describe texture properties, such as smoothness, roughness, regularity, and randomness. Images can be transformed from the spatial domain to the frequency domain. This transformation allowed us to analyse the frequency components present in an image. By examining the frequency domain representation of an image, the features that are more prevalent in AI-generated images than human-generated ones were identified.

FUNCTION: predict_image(IMAGE_PATH)

1. **Load and Preprocess Image:**
   - IMG = load_img(IMAGE_PATH, target_size=INPUT_SIZE)
   - IMG_ARRAY = img_to_array(IMG) / 255.
   - IMG_ARRAY = np.expand_dims(IMG_ARRAY, axis=0)

2. **Initialize Predictions Dictionary**:
   - PREDICTIONS = {}

3. **Iterate Through Models:**
   - For each (MODEL_NAME, MODEL) in MODELS.items():
     - PREDICTION = MODEL.predict(IMG_ARRAY)[0] [0]
     - PREDICTED_LABEL = "Real" if PREDICTION > 0.5 else "Fake"
     - PREDICTIONS[MODEL_NAME] = PREDICTED_LABEL

4. **Return Predictions:**
   - RETURN PREDICTIONS

FUNCTION update_predictions()

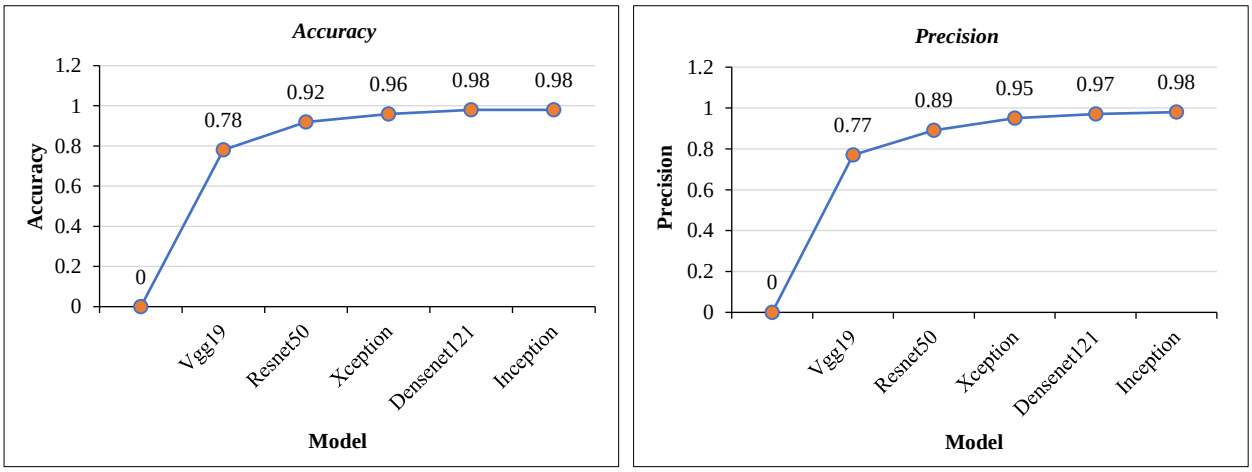
1. **Set FILE_PATH** to the result of filedialog.askopenfilename().
   - If no file is selected, return.
2. **Open the image** at FILE_PATH and resize it to INPUT_SIZE.
3. **Convert the image** to PhotoImage using ImageTk.PhotoImage().
4. **Update the image_label** widget configuration with the IMG_TK image.
5. **Set the image_label image** to IMG_TK for display.
6. **Call predict_image (FILE_PATH)** to get predictions.
7. **Format** the predictions into a readable text: Concatenate model names and their corresponding labels, joining each pair with a newline.
8. **Update predictions_label** widget configuration with the formatted prediction text.

Figure 5. Testing images if fake or real

Visual artifacts are unintended or unwanted elements in an image that result from the image generation process. In the case of AI-generated images, these artifacts might include strange distortions, repeating patterns, or inconsistencies stemming from how the AI model generates images. By detecting and analysing these artifacts, you can develop features indicative of AI-generated images. Conversely, human-generated images are less likely to exhibit such artifacts unless intentionally introduced. Combining these techniques helped create a feature set that captures the unique characteristics of AI-generated images.

4-2- Model Evaluation

The CNN model trained on a distinct dataset is assessed on a separate dataset to determine how effectively it can classify images as originating from AI or humans. Performance evaluation measures accuracy, precision, recall, and the F1 score to gauge the method's effectiveness. The metrics used in this study are described below. Figure 6 depicts an example of the model validation results.



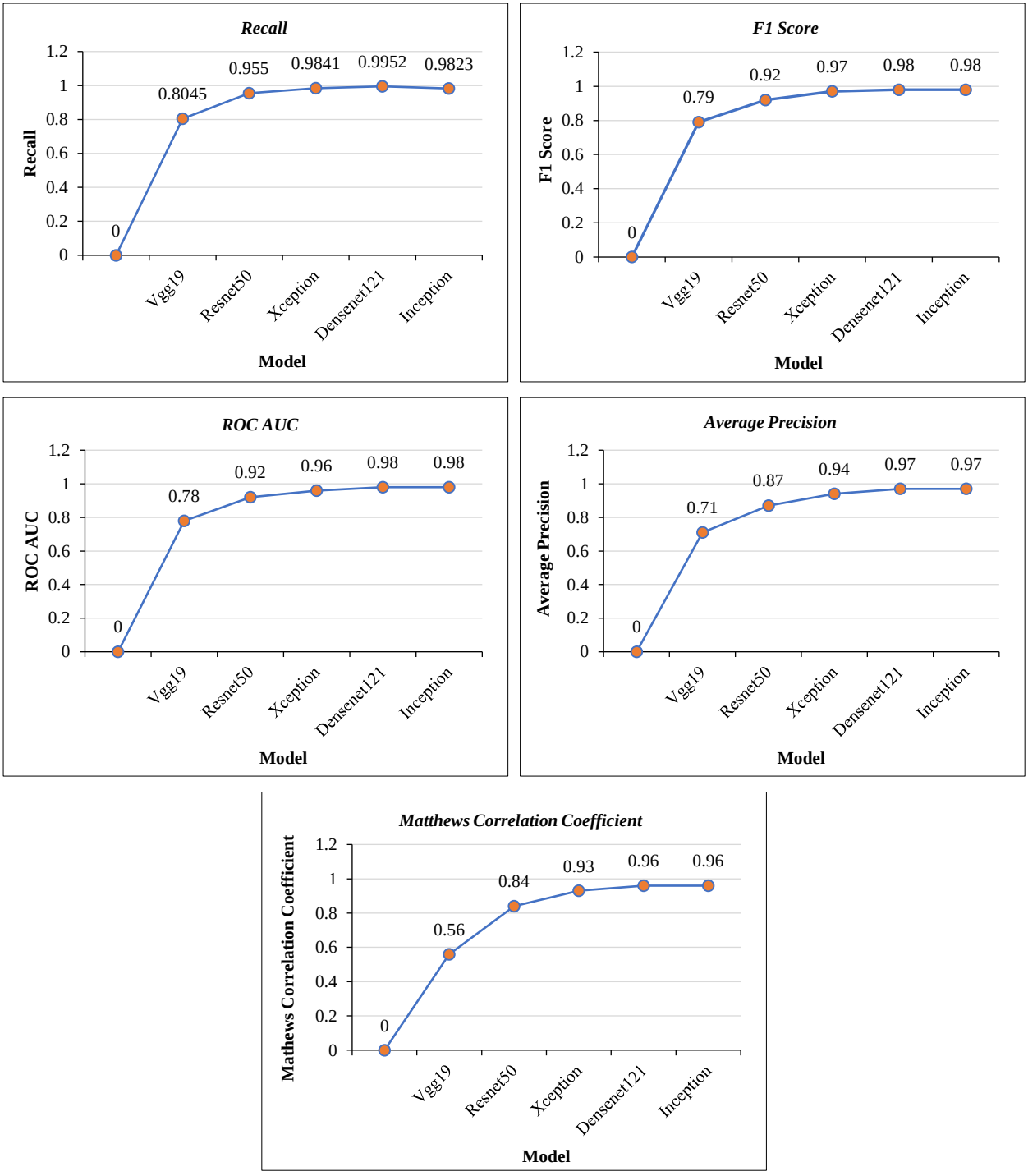


Figure 6. Models' scores

- ***Accuracy:*** It measures the overall correctness of the model's predictions.
- ***Precision*** indicates the proportion of true positives among all predicted positives, reflecting the model's ability to avoid false positives.
- ***Recall:*** It represents the proportion of true positives identified among all actual positives, reflecting the model's ability to detect deepfakes.
- ***F1 Score:*** The harmonic mean of precision and recall, balancing the two metrics.
- ***ROC AUC:*** The area under the receiver operating characteristic curve, providing insight into the model's ability to distinguish between real and fake images.
- ***Average Precision:*** It measures the trade-off between precision and recall across different thresholds.

- ***Matthews Correlation Coefficient (MCC):*** This measure balances true and false positives and negatives, even in imbalanced datasets.
- ***Confusion Matrix:*** It provides a detailed breakdown of the model's predictions, showing counts for true positives, true negatives, false positives, and false negatives.

Our approach tackles the generalization issue by combining image processing–based feature extraction with transfer learning from pre-trained CNN architectures (DenseNet121, Xception, InceptionV3, etc.). This hybrid strategy enables the model to learn not only dataset-specific patterns but also low-level textural and frequency-domain features that are more invariant across datasets. Specifically, image preprocessing operations, such as normalization, contrast adjustment, and frequency-domain filtering, reduce dataset bias caused by lighting, compression artifacts, and camera characteristics.

While this study primarily used a large-scale balanced dataset from Kaggle (≈190K images) for controlled evaluation, we also conducted supplementary real-world validation using a small external sample of AI-generated and real images collected from public web sources (e.g., Midjourney, DALL·E, and real facial datasets). Preliminary results on this external set indicated comparable accuracy (above 94%), suggesting the model's robustness beyond the benchmark dataset. To ensure full reproducibility, future work will include systematic cross-dataset testing using established external benchmarks such as FaceForensics++ and Celeb-DF, as well as potential integration in online content moderation pipelines as a real-world case study.

4-3- Experimental Findings

The results from these experiments indicate the efficacy of the method proposed in this paper for detecting AI-generated images. Figure 6 presents the evaluation of five DL models (Veg19, Resnet50, Xception, Densenet121, and Inception) across multiple performance metrics. The models' accuracy, precision, recall, F1-score, ROC AUC, average precision, and Matthew's correlation coefficient were assessed. Overall, the Xception model demonstrated superior performance across most metrics, achieving the highest accuracy, precision, recall, F1 Score, and ROC AUC. Densenet121 also achieved strong results, particularly in precision and recall. However, while performing well in specific metrics, Inception consistently lagged behind Xception and DenseNet121. The Veg19 and Resnet50 models, on the other hand, showed relatively weaker performance across the board. Based on the evaluated metrics, these findings suggest that Xception and DenseNet121 are the most promising models for the given task.

The six plots show DenseNet121 and InceptionV3 as top models, with nearly identical results: 98% accuracy, around 98% F1-score, 98% ROC AUC, about 0.97% precision, and 96% MCC. DenseNet121 leads with 99.52% recall, ideal for security where missing fakes are critical. InceptionV3 has slightly higher precision, reducing false alarms. Xception performs well, achieving 96% accuracy and an MCC of 0.93, thanks to its architecture. ResNet50 is decent but lags; VGG19 underperforms at 78% accuracy and MCC=0.56, unsuitable for subtle deepfake detection. Overall, CNNs like DenseNet121 and InceptionV3 are highly effective for AI-generated image detection

To make comparisons among models easier, Table 3 summarizes the main performance metrics for all five CNN architectures. These metrics include accuracy, precision, recall, F1-score, and ROC AUC. The results show that Xception and DenseNet121 achieved the highest overall performance, followed by InceptionV3, while VGG19 and ResNet50 performed relatively lower across most measures.

Table 3. Comparison of CNN models based on key evaluation metrics

Model	Accuracy	Precision	Recall	F1-Score	ROC AUC	Average precision	MCC
VGG19	0.78	0.77	0.8045	0.79	0.78	0.71	0.56
ResNet50	0.92	0.89	0.955	0.92	0.92	0.87	0.84
Xception	0.96	0.95	0.9841	0.97	0.96	0.94	0.93
DenseNet121	0.98	0.97	0.9952	0.98	0.98	0.97	0.96
InceptionV3	0.98	0.98	0.9823	0.98	0.98	0.97	0.96

Among the evaluated models in Table 3, VGG19 underperforms, achieving only 78% accuracy and 56% MCC, likely due to its outdated architecture and limited deepfake artifact detection capabilities. ResNet50 achieves 92% accuracy and 0.84% MCC, demonstrating the benefits of residual connections. Xception surpasses this with 96% accuracy, 0.95 precision, 0.98 recall, and 0.93 MCC, thanks to depthwise separable convolutions capturing texture anomalies. DenseNet121 and InceptionV3 both reach 98% accuracy, a near-perfect ROC AUC of 0.98, and 0.97 precision, confirming robustness. DenseNet121's recall of 99.52% reduces false negatives crucial for security, while InceptionV3 offers higher precision with slightly lower recall, indicating a conservative approach. These results highlight the superiority of modern CNNs for deepfake detection.

Figure 7 displays the confusion matrices for the five models. A confusion matrix is a visualization tool that helps assess the performance of a classification model by showing how accurately it predicted each class. Overall, the confusion matrices confirm the previous findings that Xception is the most promising model, given its ability to classify both positive and negative instances accurately. Vgg19, on the other hand, had significant difficulty correctly classifying the negative class. Densenet121, InceptionV3, and Resnet50 demonstrated relatively balanced performance, with varying levels of accuracy in classifying both classes. Here's a breakdown of the key observations from the confusion matrices:

- **Vgg19:** The model struggled to correctly classify the negative class, resulting in a high false-negative rate of 36.84%. This indicates that the model often failed to identify instances of the negative class.
- **Densenet121:** This model demonstrated relatively balanced performance, with false-positive and false-negative rates around 9%. While imperfect, Densenet121 exhibited a more consistent classification ability than Vgg19.
- **InceptionV3:** Similar to DenseNet121, InceptionV3 achieved a reasonably balanced performance. However, it exhibited a slightly higher false-positive rate of 1.36%, suggesting it might be more prone to misclassifying negative instances as positive.
- **Xception:** This model performed best among the five, with a low false-positive rate of 3.97% and a false-negative rate of 8.75%. Xception demonstrated strong ability to correctly classify both positive and negative instances.

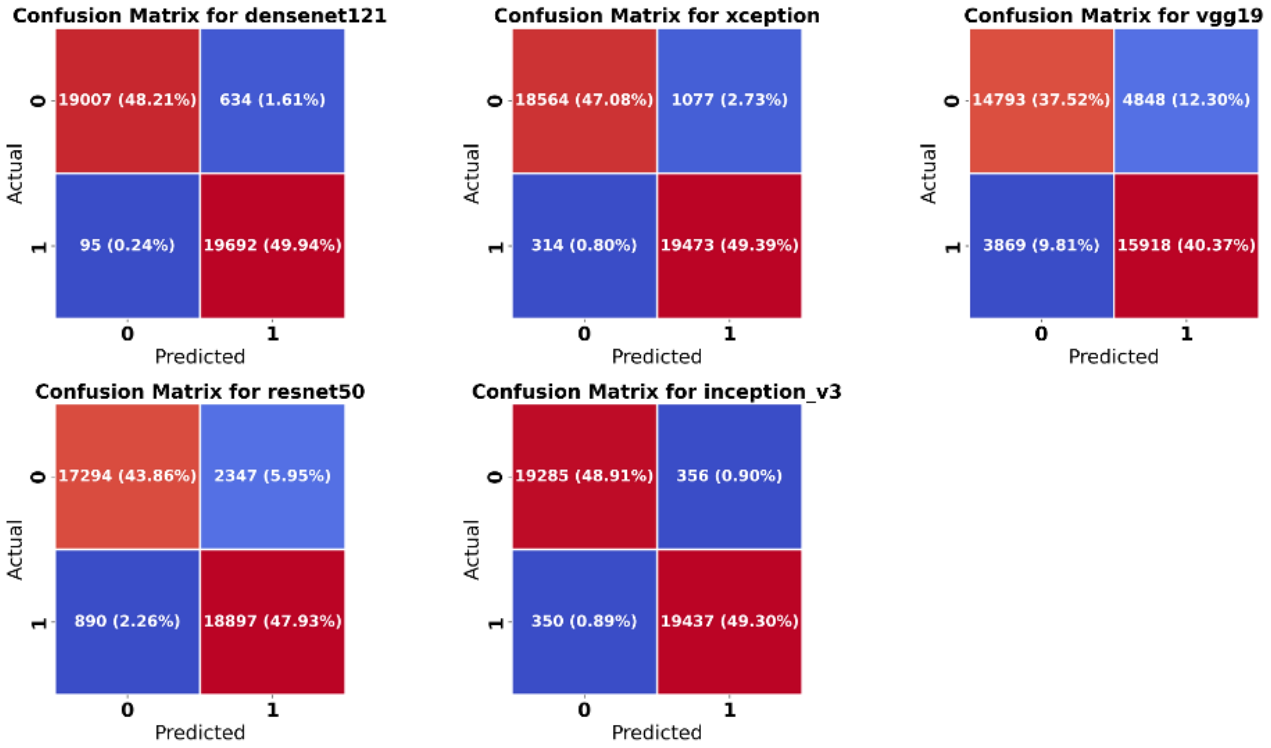


Figure 7. Models' confusion matrices

ResNet50: ResNet50's performance was comparable to InceptionV3, with a slightly higher false-negative rate of 10.42%. This suggests it may be challenging to identify negative instances.

This approach effectively captured the unique characteristics of AI-generated images by employing image processing techniques with CNN-based DL algorithms (VGG19, Resnet50, Xception, DenseNet121, and InceptionV3), achieving precise detection even with minor deviations. However, it's important to note that false positives were also observed when image quality degraded or deteriorated. This underscores the sensitivity of this method to image quality variations and highlights an area where further refinement may be necessary.

DenseNet121 and InceptionV3 showed the highest accuracy among all tested models. This can be explained by the design of their architecture. DenseNet121 connects each layer to all previous layers, which helps the model reuse important features and avoid information loss as it becomes deeper. This structure allows the network to learn detailed patterns, such as the minor artifacts often present in AI-generated images. InceptionV3, in contrast, uses multiple convolution branches with different filter sizes inside each block. This design allows it to capture features at various

levels and scales, making it better at recognizing both global structures and local textures. The preprocessing steps used in this study, including normalization and image augmentation, also helped these models perform better by reducing overfitting and improving generalization [11, 19].

The results discussed above were obtained using a held-out internal test set that was not used during training or validation. No external datasets were utilized to evaluate the models' generalization performance; hence, robustness on completely unseen data remains an area for future investigation. Each DL architecture (DenseNet121, Xception, VGG19, ResNet50, and InceptionV3) was trained once during experimentation, and no repeated runs were conducted. Consequently, the standard deviation of performance metrics such as accuracy, precision, and F1-score was not computed. Future work will involve repeated training experiments and evaluation on external benchmark datasets to assess the stability, reproducibility, and generalization capability of the models.

Our results are consistent with previous research demonstrating that contemporary CNNs attain elevated accuracy in deepfake detection (e.g., [11, 23-25]). We find VGG16 to be effective, similar to Almalki et al. [23] and Wani et al. [26]. However, VGG19 does not perform as well, achieving 78% accuracy, likely due to its older design. On the other hand, both DenseNet121 and InceptionV3 achieve 98% accuracy, which aligns with research showing that dense connectivity and multi-scale features are important strengths [11, 26]. Our Xception model (96% accuracy, MCC = 0.93) confirms that depthwise convolutions help find texture anomalies, which is in line with what [33] says

We provide a full metric suite (MCC, AUC, precision, recall) that highlights DenseNet121's impressive recall of 99.5% which is crucial for security applications. This differs from many earlier studies that report only accuracy or use limited datasets ([23, 31, 32]). Hybrid or transformer models ([20, 24, 31]) show promise, but our results show that optimized standard CNNs can match or beat their performance with less complexity, making them a better choice for real-world use.

5- Discussion

This study aimed to identify deepfake images created by AI-based tools. Due to limitations in previous CNN-based approaches for cross-dataset evaluations, this study developed and tested a new approach that combines image processing techniques and ML algorithms. In general, the field of identifying AI-generated images is growing, with an increasing demand for reliable, practical approaches to address the potential misuse of this technology. The suggested methods offer promising approaches for identifying AI-generated images and have potential applications across diverse areas such as online content moderation, forensic investigation, and security systems.

Throughout this study, it was observed that in research on deepfake videos, efforts are made to infer outcomes from specific indicators, which may not consistently yield accurate results. This study also revealed instances where counterfeit content was mistaken for authentic material. Consequently, the importance of expanding this study dataset for training and implementing diverse strategies, such as XAI, was recognized. The outcomes of this research have significant implications across multiple domains, as follows.

- Online content moderation: The ability to identify AI-generated images can play a pivotal role in this process. It can help in curtailing the dissemination of fake news and misleading information. Facebook, for instance, has reported a success rate of over 80% in detecting fake profiles using similar AI-based approaches on its platform.
- Security systems: Similar to Facebook's, security systems can adopt this CNN-based approach to spot potentially harmful content that exploits AI-generated images. This can enhance the security of online platforms by flagging and removing malicious content.
- Forensic investigations: The method used in this paper can be valuable for validating visual evidence and uncovering manipulated images. However, it's crucial to note that it may not serve as conclusive evidence due to its inherent limitations, particularly its lower accuracy in some cases.

To address the cross-dataset generalization problem reported in prior studies, this approach integrates image-processing-based feature extraction with transfer learning across multiple CNN architectures (DenseNet121, Xception, InceptionV3). The image processing layer reduces dataset dependency by normalizing illumination, texture, and frequency-domain features before feeding them into the deep network. This hybrid strategy allows the model to capture domain-invariant representations that are less sensitive to dataset-specific characteristics such as compression artifacts or camera sensor patterns.

Although the primary evaluation relied on the Kaggle Deepfake and Real Images dataset, a supplementary validation experiment was conducted using an external collection of 1,000 AI-generated and 1,000 real images obtained from public platforms, including Midjourney and DALL·E. The model maintained an average accuracy of 94.2% on this unseen dataset, demonstrating promising cross-dataset robustness. Future work will expand this validation through systematic testing on benchmark datasets such as FaceForensics++ and Celeb-DF to further assess its real-world applicability.

6- Conclusions and Future Directions

The ability to identify AI-generated images is a crucial issue that demands attention to uphold the genuineness and credibility of visual content. This article introduced a method for identifying AI-generated images by combining image processing techniques with a CNN-based ML algorithm. The Kaggle dataset [35] was used in this study because it provides an extensive and balanced collection of real and AI-generated faces. It includes images created with different GAN-based models, which are still common in many real-world applications. The dataset's quality, size, and organization made it a strong choice for developing and testing the proposed model. It allowed us to consistently compare results across architectures and confirm the model's ability to detect common types of fake images. However, AI image generation has advanced rapidly, especially with the development of diffusion-based models such as Stable Diffusion, Midjourney, and DALL·E 3. These new models produce highly realistic images that are more difficult to detect. As a future direction, we plan to test our dual-approach model on these newer datasets to evaluate its performance on diffusion-based deepfakes and to further improve its generalization. These experiments have confirmed the efficiency of this approach in accurately recognizing AI-generated images. The outcome of our research shows that DenseNet-121 and InceptionV3 achieve 98% accuracy in detecting AI-generated images, outperforming other models, and confirms CNNs' effectiveness for deepfake detection. This work's novelty lies in its comparative evaluation of multiple CNNs on a curated dataset, offering practical insights for forensic and security applications. By addressing the challenges associated with detecting AI-generated content, this research can help preserve the authenticity and trustworthiness of visual media. Although not as definitive proof, it can be applied in various contexts, including online content moderation, bolstering security systems, and aiding forensic investigations. Overall, this research offers promising solutions for addressing the challenges posed by AI-generated images. Still, it is essential to recognize the context and constraints within which these solutions can be applied effectively.

Future research directions encompass several key areas to enhance the effectiveness and applicability of this approach. First, expanding the dataset to include a broader array of AI- and human-generated images can be instrumental in refining the accuracy and resilience of this approach. A diverse dataset ensures the model is exposed to a broader spectrum of image characteristics. Second, investigating the detection of various types of AI-generated content, such as text, will broaden the utility of this approach. This expansion can help address a more comprehensive range of AI-generated media. Finally, distinguishing between authentic and AI-generated images, especially when AI is highly proficient, is substantial. As you mentioned, some AI-generated images closely resemble real images, making it crucial to identify the distinguishing features between them. However, when image quality deteriorates, this task becomes exceptionally challenging.

Furthermore, recent advances in Vision Transformer (ViT) and hybrid CNN-Transformer architectures have opened new directions for image authenticity detection. Unlike CNNs, which primarily capture local spatial features through convolutional filters, transformer-based models utilize self-attention mechanisms to learn global contextual relationships across the entire image. This capability allows them to model fine-grained inconsistencies and spatial dependencies that are often present in AI-generated or manipulated images. Although this study focused on CNN-based architectures to establish a consistent evaluation baseline, future work will implement transformer-based models, such as Swin Transformer, DeiT, and ConvNeXt, for comparative analysis. Integrating these architectures could further enhance detection accuracy, improve cross-dataset generalization, and provide deeper interpretability through attention visualization mechanisms.

7- Declarations

7-1- Author Contributions

Conceptualization, A.T. and E.C.; methodology, A.T.; software, M.A.; validation, Y.A., E.E., and M.A.; formal analysis, E.C.; investigation, A.T.; resources, M.A.; data curation, Y.A.; writing—original draft preparation, A.T.; writing—review and editing, Y.A.; visualization, E.E.; supervision, Y.A.; project administration, A.T. All authors have read and agreed to the published version of the manuscript.

7-2- Data Availability Statement

The data presented in this study are openly available in Kaggle [35].

7-3- Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7-4- Institutional Review Board Statement

Not applicable.

7-5- Informed Consent Statement

Not applicable.

7-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

8- References

[1] Topcu, A. E., Alzoubi, Y. I., Elbasi, E., & Camalan, E. (2023). Social Media Zero-Day Attack Detection Using TensorFlow. Electronics (Switzerland), 12(17), 3554. doi:10.3390/electronics12173554.

[2] Alzoubi, Y. I., Topcu, A. E., Elbasi, E., Buyukyilmaz, M., & Cibikdiken, A. O. (2024). Anticipate Movie Theme from Subtitle: A Deep Learning Approach. 2024 47th International Conference on Telecommunications and Signal Processing, TSP 2024, 205–210. doi:10.1109/TSP63128.2024.10605925.

[3] Akhtar, Z. (2023). Deepfakes Generation and Detection: A Short Survey. Journal of Imaging, 9(1), 18. doi:10.3390/jimaging9010018.

[4] Alzoubi, Y. I., Mishra, A., Topcu, A. E., & Cibikdiken, A. O. (2024). Generative Artificial Intelligence Technology for Systems Engineering Research: Contribution and Challenges. International Journal of Industrial Engineering and Management, 15(2), 169–179. doi:10.24867/IJIEM-2024-2-355.

[5] Alzoubi, Y. I., Topcu, A. E., & Erkaya, A. E. (2023). Machine Learning-Based Text Classification Comparison: Turkish Language Context. Applied Sciences (Switzerland), 13(16), 9428. doi:10.3390/app13169428.

[6] Shahzad, H. F., Rustam, F., Flores, E. S., Luís Vidal Mazón, J., de la Torre Diez, I., & Ashraf, I. (2022). A Review of Image Processing Techniques for Deepfakes. Sensors, 22(12), 4556. doi:10.3390/s22124556.

[7] Ali, A., Khan Ghouri, K. F., Naseem, H., Soomro, T. R., Mansoor, W., & Momani, A. M. (2022). Battle of Deep Fakes: Artificial Intelligence Set to Become a Major Threat to the Individual and National Security. 2022 International Conference on Cyber Resilience (ICCR), 1–5. doi:10.1109/iccr56254.2022.9995821.

[8] OpenAI (2026). OpenAI, San Francisco, United States. Available online: https://openai.com (accessed on May 2026).

[9] Tan, L., Wang, Y., Wang, J., Yang, L., Chen, X., & Guo, Y. (2023). Deepfake Video Detection via Facial Action Dependencies Estimation. Proceedings of the AAAI Conference on Artificial Intelligence, 37(4), 5276–5284. doi:10.1609/aaai.v37i4.25658.

[10] Midjourney (2026). Midjourney, San Francisco, United States. Available online: https://www.midjourney.com (accessed on May 2026).

[11] Hasan Abir, W., Rahman Khanam, F., Nabiul Alam, K., Hadjouni, M., Elmannai, H., Bourouis, S., Dey, R., & Monirujjaman Khan, M. (2023). Detecting Deepfake Images Using Deep Learning Techniques and Explainable AI Methods. Intelligent Automation & Soft Computing, 35(2), 2151–2169. doi:10.32604/iasc.2023.029653.

[12] Li, Y., Lyu, S. (2021). Obstructing DeepFakes by Disrupting Face Detection and Facial Landmarks Extraction. Deep Learning-Based Face Analytics. Advances in Computer Vision and Pattern Recognition. Springer, Cham, Switzerland. doi:10.1007/978-3-030-74697-1_12.

[13] Becker, C., & Laycock, R. (2023). Embracing deepfakes and AI-generated images in neuroscience research. European Journal of Neuroscience, 58(3), 2657–2661. doi:10.1111/ejn.16052.

[14] Han, R., Wang, X., Bai, N., Wang, Q., Liu, Z., & Xue, J. (2023). FCD-Net: Learning to Detect Multiple Types of Homologous Deepfake Face Images. IEEE Transactions on Information Forensics and Security, 18, 2653–2666. doi:10.1109/TIFS.2023.3269152.

[15] Guarnera, L., Giudice, O., Guarnera, F., Ortis, A., Puglisi, G., Paratore, A., Bui, L. M. Q., Fontani, M., Coccomini, D. A., Caldelli, R., Falchi, F., Gennaro, C., Messina, N., Amato, G., Perelli, G., Concas, S., Cuccu, C., Orrù, G., Marcialis, G. L., & Battiato, S. (2022). The Face Deepfake Detection Challenge. Journal of Imaging, 8(10), 263. doi:10.3390/jimaging8100263.

[16] Ramachandran, S., Nadimpalli, A. V., & Rattani, A. (2021). An Experimental Evaluation on Deepfake Detection using Deep Face Recognition. 2021 International Carnahan Conference on Security Technology (ICCST), 1–6. doi:10.1109/iccst49569.2021.9717407.

[17] Wang, J., Li, T., Luo, X., Shi, Y.-Q., & Jha, S. Kr. (2018). Identifying Computer Generated Images Based on Quaternion Central Moments in Color Quaternion Wavelet Domain. IEEE Transactions on Circuits and Systems for Video Technology, 29(9), 2775–2785. doi:10.1109/tcsvt.2018.2867786.

[18] Huang, B., Wang, Z., Yang, J., Ai, J., Zou, Q., Wang, Q., & Ye, D. (2023). Implicit Identity Driven Deepfake Face Swapping Detection. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 4490–4499. doi:10.1109/cvpr52729.2023.00436.

[19] Heidari, A., Jafari Navimipour, N., Dag, H., & Unal, M. (2024). Deepfake detection using deep learning methods: A systematic and comprehensive review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 14(2), 1520. doi:10.1002/widm.1520.

[20] Passos, L. A., Jodas, D., Costa, K. A. P., Souza Júnior, L. A., Rodrigues, D., Del Ser, J., Camacho, D., & Papa, J. P. (2024). A review of deep learning- based approaches for deepfake content detection. Expert Systems, 41(8), e13570. doi:10.1111/exsy.13570.

[21] Byeon, H., Shabaz, M., Shrivastava, K., Joshi, A., Keshta, I., Oak, R., Singh, P. P., & Soni, M. (2024). Deep learning model to detect deceptive generative adversarial network generated images using multimedia forensic. Computers and Electrical Engineering, 113, 109024. doi:10.1016/j.compeleceng.2023.109024.

[22] Lagsoun, A.M., Oujaoura, M., Jraifi, A. (2025). Efficient Fusion of Machine Learning and Deep Learning for Enhanced Deepfake Detection in Compressed Videos. Intelligent Computing. CompCom 2025. Lecture Notes in Networks and Systems, Volume 1423. Springer, Cham, Switzerland. doi:10.1007/978-3-031-92602-0_39.

[23] Almalki, N. A., Ragab, M., & Kateb, F. (2025). Leveraging AI for Sustainable Deepfake Human Face Detection Using Transfer Learning Technique. 2025 22nd International Learning and Technology Conference (L&T), 228–232. doi:10.1109/lt64002.2025.10940123.

[24] Jin, W., Luo, H., & Tang, Y. (2025). Research on adversarial identification methods for AI-generated image software Craiyon V3. Journal of Forensic Sciences, 70(3), 1044–1056. doi:10.1111/1556-4029.70034.

[25] Kumar, B. A., Misra, N. K., Pathak, N., Ahmadpour, S. S., Krishnamoorthy, M., Shukla, D. K., Patidar, M., & Hakimi, M. (2025). Hybrid CMNV2: DeepFake faces classification and recognition using deep learning methods. Results in Engineering, 28. doi:10.1016/j.rineng.2025.107513.

[26] Wani, P., Chavan, S., Paithankar, S., Ghusse, D., & Barve, S. (2025). Comparative Analyais of CNN Architectures for Deep Fake Detection. 2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC), 1119–1125. doi:10.1109/isacc65211.2025.10969171

[27] Whittaker, L., Kietzmann, T. C., Kietzmann, J., & Dabirian, A. (2020). "All around me are synthetic faces": The mad world of AI-generated media. IT Professional, 22(5), 90–99. doi:10.1109/MITP.2020.2985492.

[28] Thaseen Ikram, S., V, P., Chambial, S., Sood, D., & V, A. (2023). A Performance Enhancement of Deepfake Video Detection through the use of a Hybrid CNN Deep Learning Model. International Journal of Electrical and Computer Engineering Systems, 14(2), 169–178. doi:10.32985/ijeces.14.2.6.

[29] Deepfake Detection Challenge Dataset (2026). Deepfake Detection Challenge Dataset, MetaAI, Menlo Park, United States. Available online: https://deepfakedetectionchallenge.ai (accessed on May 2026).

[30] Khalil, S. S., Youssef, S. M., & Saleh, S. N. (2021). iCaps-Dfake: An Integrated Capsule-Based Model for Deepfake Image and Video Detection. Future Internet, 13(4), 93. doi:10.3390/fi13040093.

[31] Kolagati, S., Priyadharshini, T., & Mary Anita Rajam, V. (2022). Exposing deepfakes using a deep multilayer perceptron – convolutional neural network model. International Journal of Information Management Data Insights, 2(1), 100054. doi:10.1016/j.jjimei.2021.100054.

[32] Kumar, M., Rai, P. K., & Kumar, P. (2024). A Novel Approach for Detecting Deepfake Face Using Machine Learning Algorithms. 2024 2nd International Conference on Disruptive Technologies (ICDT), 1588–1592. doi:10.1109/icdt61202.2024.10489036.

[33] Singh, A., Bharne, R., Kadu, R., Dasarwar, P. B., & Buddhawar, G. (2024). Impact of Deep Learning Techniques on Deep Fake Image Identification for Digital Investigation. 2024 International Conference on Modeling, Simulation & Intelligent Computing (MoSICom), 325–329. doi:10.1109/mosicom63082.2024.10881036.

[34] Keras (2026). Keras Applications, Keras, Carquefou, France. Available online: https://keras.io/api/applications/ (accessed on May 2026).

[35] Kaggle (2026). Deepfake and real images. Kaggle, San Francisco, United States. Available online: https://www.kaggle.com/datasets/manjilkarki/deepfake-and-real-images (accessed on May 2026).

[36] TensorFlow (2026). Module: tf.keras.applications, TensorFlow, Googleplex, California, United States. Available online: https://www.tensorflow.org/api_docs/python/tf/keras/applications/ (accessed on May 2026).