Detection of AI Generated Fake Images

1st Mohammad Ammar Siddique
Department of Data Science
Faculty of Computing and Information Technology
University of the Punjab
Lahore, Pakistan
bsdsf22m041@pucit.edu.pk

2nd Muhammad Ali Raza
Department of Data Science
Faculty of Computing and Information Technology
University of the Punjab
Lahore, Pakistan
bsdsf22m011@pucit.edu.pk

3rd Ali Hamza
Department of Data Science
Faculty of Computing and Information Technology
University of the Punjab
Lahore, Pakistan
bsdsf22m032@pucit.edu.pk

4th Faisal Bukhari
Department of Data Science
Faculty of Computing and Information Technology
University of the Punjab
Lahore, Pakistan
faisal.bukhari@pucit.edu.pk

I. Introduction

In today's age, the advent of advanced generative models has made it possible to generate highly photorealistic images from simple textual prompts. [1]. AI softwares, such as, DALL-E 2, Stable Diffusion and Midjourney have made their way into digital and creative industries, fueling innovation and creativity. Although this has essentially started a new era in the field of digital creativity, it has also created significant challenges related to digital authenticity and security [2]. With the rise of text-to-image models, strong concerns over misinformation, deep fakes, and intellectual property infringement have increased. As a result, reliable and adaptable detection mechanisms have become the need of the hour [3].

The rapid evolution of generative techniques has significantly improved the realism of synthetically generated images [4]. These models make use of highly complex processes, such as adversarial training and iterative noise refinement, to generate highly realistic visual images [5]. As a result, conventional forensic methods that rely on predictable discrepancies, often fail at detecting these images. Traditional forensic approaches to detect synthetic images depended on static features extracted from digital images. Techniques such as Photo Response Non-Uniformity (PRNU) and Error Level Analysis (ELA) were initially very effective in identifying differences between camera-specific noise patterns and manipulated images [6]. As generative tools evolved, these traditional techniques became less effective and reliable, demonstrating the need for robust detection frameworks [7].

In response to these limitations, researchers have shifted to deep learning-based approaches that have been transformative [8]. Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) have proven much more effective in capturing the visual inconsistencies found in AI-generated images [6]. These models have achieved resounding success in distinguishing synthetic images from natural ones by employing a more sophisticated and complex approach. Convolutional Neural Networks (CNNs) provide a robust framework for AI image detection by eliminating the need of hand-crafted feature extraction [9]. Increasing the bar, the ViT model attains an even higher level of accuracy in distinguishing between real and synthetic images, compared to CNN based models [10]. However, despite their impressive performance, deep learning models often require substantial computational resources, posing challenges for practical deployment in resource-constrained environments [11].

A promising direction in detection research involves semantic-based techniques that leverage pre-trained vision language models such as CLIP (Contrastive Language-Image Pre-Training) [12]. Trained on vast datasets comprising hundreds of millions of image-text pairs, these models can identify high-level semantic mismatches between visual content and corresponding textual descriptions [7]. By focusing on broader contextual signals rather than relying solely on low-level artifacts, CLIP-based methods are capable of detecting synthetic images with minimal training data. This approach not only improves the detection accuracy especially in scenarios involving heavy post processing but also enhances the generalizability of detectors across diverse generative architectures [5].

Parallel to supervised deep learning approaches, unsupervised and zero-shot techniques have emerged as a compelling alternative for detecting AI-generated images without relying on extensive labeled synthetic datasets [13]. These methods typically model the statistical distributions of natural images and identify potential anomalies indicative of synthetic manipulation [2]. For example, analyzing coding cost gaps through lossless image compression can reveal intrinsic discrepancies that synthetic images introduce, even when conventional artifacts are absent. Such zero shot frameworks are particularly valuable in real world scenarios where new generative models

978-1-6654-6477-2/22/$31.00 ©2022 IEEE

rapidly appear, and acquiring comprehensive training data for every variant is impractical [14].

Complementing these semantic and unsupervised techniques are forensic approaches that exploit frequency-domain analysis and sensor noise patterns to unmask synthetic content. Methods that apply Discrete Fourier Transform (DFT) and azimuthal averaging have shown promise by converting images into power spectra that expose high frequency irregularities unique to AI-generated images [7]. Similarly, the extraction of sensor pattern noise through high pass filtering provides a reliable forensic fingerprint for natural images, enabling the accurate classification of computer generated graphics versus authentic photographs [14]. These techniques offer a resource light alternative to deep learning, with the added benefit of robustness to common image alterations such as compression and scaling [5].

Despite these advances, significant challenges persist in the detection of images generated by state-of-the-art text to image models [4]. The complex processes by which these models generate images often result in subtle artifacts ranging from anomalies in color normalization to inconsistencies in high frequency details that may not be readily captured by conventional detection methods [6]. Moreover, as these models become more sophisticated, the forensic fingerprints they leave behind become increasingly nuanced [5]. Recent research has begun to demystify these complexities, demonstrating that even highly realistic synthetic images can be attributed to specific generative models based on unique, model specific artifacts [3]. This insight paves the way for more targeted detection and attribution strategies that hold model creators responsible for potential misuse.

The wider consequences of robust fake image detection extend far beyond the technical world. In an era where digital misinformation can have far-reaching consequences, it can also influence public opinion on sociopolitical processes, potentially creating a public sentiment that can go against their greater good [5]. Henceforth, the development of adaptable and reliable detection systems is significant. Powerful detectors can play a key role in media forensics, helping to authenticate visual evidence and protecting intellectual property rights [15]. Furthermore, the deployment of these technologies in social media and news platforms could significantly alleviate the spread of disinformation, thereby preserving the integrity of digital communication networks. The integration of advanced detection mechanisms is therefore, not only a technical challenge but also a societal imperative [2].

II. Literature Review

The swift progress of generative models, including GANs and more recently, Diffusion Models, has made the production of hyper-realistic synthetic images more easier than ever. As a result, the demand for effective strategies to accurately identify AI-generated fake images has become more important than ever, particularly in fields such as journalism, digital forensics, and the moderation of social media content. Various methods have been developed to address this issue, mainly utilizing deep learning techniques. Early research in this field focused on detecting subtle visual artifacts produced by generative models. These artifacts encompass inconsistencies in texture, unnatural lighting, and distortions in facial features, especially around the eyes or in the background (Fig. 1). Nevertheless, with advancements in generative models like StyleGAN2 and DALL·E, these artifacts have become increasingly difficult for the human eye to distinguish, highlighting the need for automated and more effective detection methods.

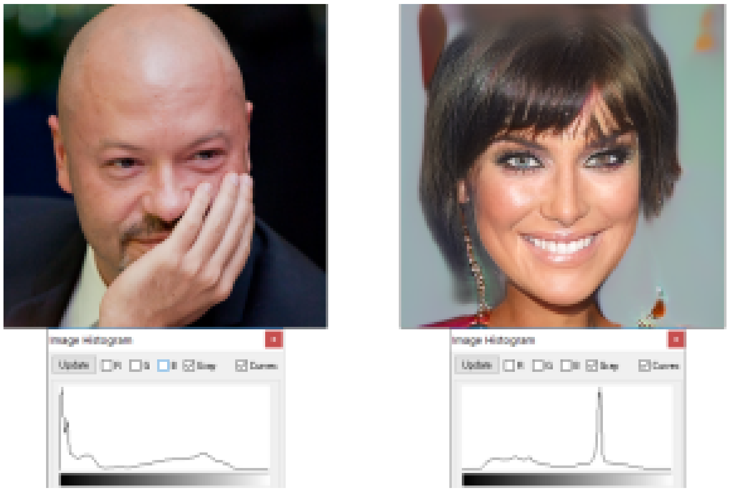

Fig. 1: An approach demonstrated in S. McCloskey and M. Albright, *Detecting gan-generated imagery using color cues,”* 2018. detects images via color mismatches in R/G/B channels

Convolutional Neural Networks (CNNs) are widely utilized because of their exceptional performance in image classification tasks. Models such as ResNet, Xception, and EfficientNet have been optimized to distinguish between authentic and AI-generated images with notable accuracy. A prevalent approach is to employ pre-trained models on extensive datasets and subsequently fine-tune them on specialized datasets like CIFAKE or DeepFakeDetection to better accommodate the subtleties of synthetic images.

Recent literature also explores the use of Vision Transformers (ViTs) and attention-based mechanisms to improve detection performance, particularly for higher resolution images where global context is more important. Additionally, some studies have proposed ensemble methods and hybrid models that combine CNNs and frequency-domain analysis to enhance robustness against adversarial examples or post-processing artifacts.

In general, the domain has progressed from detection methods based on artifacts to more comprehensive and data-centric strategies that utilize deep learning. Transfer learning and pre-trained models have been pivotal in attaining high accuracy even with a comparatively small amount of training data.

III. Methodology

A. Dataset

In order to effectively differentiate between authentic and AI-generated images, this research uses the CIFAKE dataset, a carefully curated benchmark introduced to support the advancement and assessment of deep learning-based fake image detection systems. The CIFAKE dataset consists of a total of 60,000 images, evenly split into 30,000 real images and 30,000 fake images. The real images are sourced directly from the CIFAR-10 dataset, while the fake images are artificially created through a combination of GAN-based and diffusion-based models, such as StyleGAN2 and Stable Diffusion.

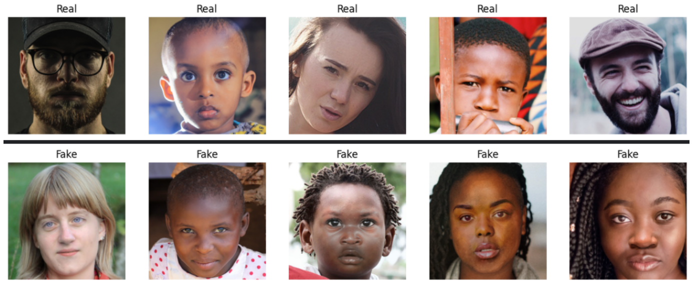

Fig. 2: Sample images for CIFAKE dataset.

Each image within the dataset measures 32×32 pixels and is evenly assigned among 10 categories. Notably, the synthetic images replicate this class distribution, maintaining class balance making the dataset highly appropriate for deep learning frameworks. Due to its balanced composition and the clear differentiation between synthetic and real images, CIFAKE provides a robust basis for training binary classifiers in the field of synthetic image forensics.

B. Model Selection

In order to evaluate the efficacy of transfer learning in identifying AI-generated images, five pre-trained convolutional neural network architectures were chosen. Among these are MobileNetV2, VGG16, InceptionV3, ResNet50, and EfficientNetB0. These models were selected due to their varied depth, number of parameters, and architectural designs, which facilitate a comparative analysis of both performance and computational efficiency.

MobileNetV2 is a streamlined and effective neural network tailored for mobile and embedded systems. Building upon MobileNetV1, it incorporates linear bottlenecks and inverted residual blocks, which reduce computational demands while improving accuracy [16]. The architecture is fundamentally based on depth-wise separable convolutions, which break down standard convolutions into point-wise and depth-wise operations, significantly lowering computational costs. Each block enhances the input channels, employs efficient depth-wise filters, and subsequently compresses the channels, resulting in a network that is both compact and powerful. With only 19 layers, MobileNetV2 is particularly suited for real-time inference on devices with limited processing capabilities [17].

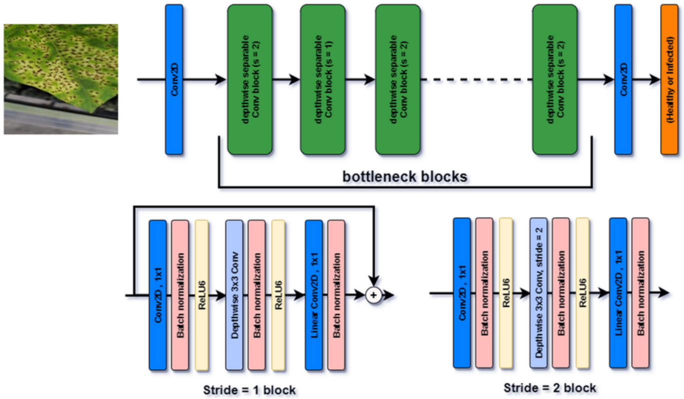

Fig. 3: MobileNetV2 architecture.

Another well-known deep convolutional neural network is VGG16 developed by the Visual Geometry Group at Oxford. It consists of 16 weight layers, including 13 convolutional layers that utilize small 3x3 filters and 3 fully connected layers. The use of uniform filter sizes simplifies the architecture while allowing it to capture intricate details. Nonlinearity is introduced through ReLU activation, and max-pooling layers reduce spatial dimensions to improve learning efficiency [18].

VGG16 is extensively utilized in AI-driven image detection and classification, often serving as a foundational model in object detection frameworks such as Faster R-CNN and SSD. Despite the emergence of more advanced models such as ResNet and MobileNetV2, which offer improved efficiency and speed, VGG16 remains popular due to its interpretability and reliable performance, although it is less favored in real-time or resource-constrained environments [19].

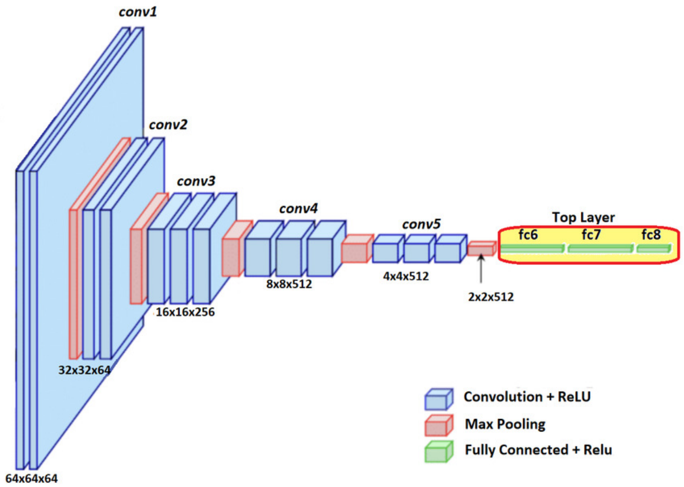

Fig. 4: VGG16 architecture.

InceptionV3 is a deep learning model developed by Google, widely utilized in AI image detection. It is designed for both accuracy and speed, employing distinctive multi-branch blocks that analyze images at different scales. These blocks simultaneously utilize filters of varying sizes (1×1, 3×3, and 5×5) to allow the model to detect both fine and broad features [20]. To reduce the size of the model and improve processing speed, it incorporates dimension reduction, batch normalization, and label smoothing, which contribute to training stability and accuracy. In contrast to VGG16 and other earlier models, InceptionV3 is remarkably efficient, requiring fewer parameters while delivering superior performance [21].

ResNet50 is an older and advanced deep learning architecture that enables the creation of much deeper networks by tackling the problem of vanishing gradients through the implementation of residual (skip) connections. These shortcuts allow the model to directly combine input and output, thus improving its learning abilities even at significant depths. The structure of ResNet50 comprises 50 layers, featuring bottleneck blocks that make use of 1×1, 3×3, and 1×1 convolutions to decrease the number of parameters while maintaining performance [22]. Furthermore, it includes batch normalization to guarantee stability throughout the training phase and uses global average pooling to mitigate the risk of overfitting. With approximately 25.6 million parameters, it is notably more efficient than previous models such as VGG16 [23].

EfficientNetB0 has improved model scaling by implementing compound scaling, which uniformly adjusts the depth, width, and resolution of a network using a single factor. Its architecture is composed of MBConv (Mobile Inverted Bottleneck) blocks and Squeeze-and-Excitation (SE) modules. These blocks increase the number of channels, perform lightweight depthwise convolutions, and modify channel significance through attention mechanisms, enabling the model to extract dense features with minimal computational cost [24]. Despite having only 5.3 million parameters, EfficientNetB0 achieves a top-1 ImageNet accuracy of 77.1 and outperforms larger models such as ResNet50 in terms of accuracy per computation.

C. Pre-processing and Model Training

In order to prepare the CIFAKE dataset for training, a series of pre-processing steps were uniformly implemented across all models to guarantee comparability and generalization. Given that CIFAKE images have a resolution of 32×32 pixels and pretrained models generally require larger inputs, all images were resized to 224×224 pixels. As a result, the pixel values were normalized to the range of [0, 1].

To mitigate the risk of overfitting and improve the model's capacity to generalize to previously unseen data, data augmentation techniques were utilized throughout the training process. These techniques encompassed random horizontal flipping, minor-angle rotations, zooming, and slight translations. Such transformations added variability to the input space while maintaining semantic integrity, which is particularly crucial for a small and low-resolution dataset such as CIFAKE.

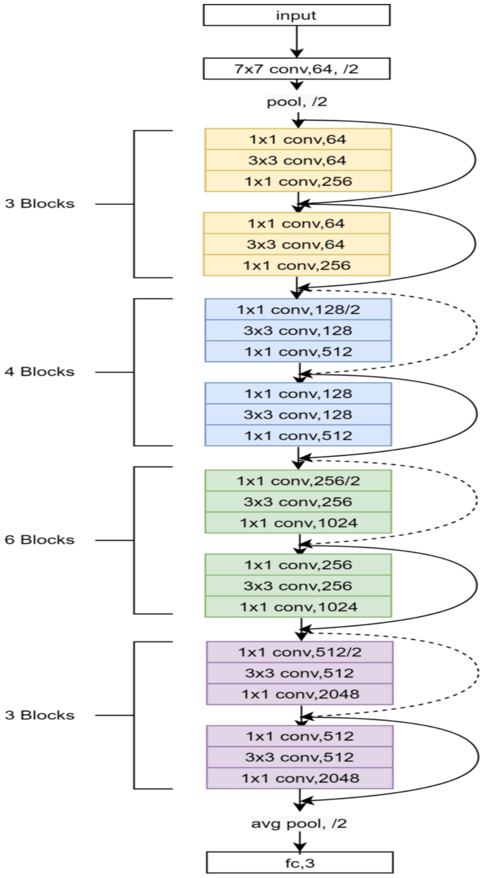

Fig. 5: ResNet50 architecture.

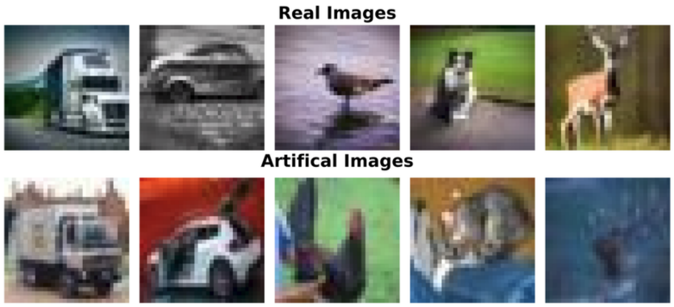

Fig. 6: Sample images after intial pre-processing.

The dataset was partitioned into an 80:20 ratio for training and validation purposes, using a stratified sampling technique to maintain the class distribution in both subsets. The classification task was established as binary, requiring the models to determine whether a given image was genuine or counterfeit. As a result, the binary cross-entropy loss function was selected, and optimization was performed using the Adam optimizer with a fixed learning rate of 0.0001. Each model was trained for a maximum of 10 epochs, with early stopping applied based on validation accuracy to avert overfitting. A batch size of 32 was used consistently, although modifications were made as necessary to meet the memory requirements of each model. All training and evaluation activities were conducted in a GPU-enabled environment using Kaggle notebooks. Each model used the same data pipeline and training parameters, ensuring a fair and controlled evaluation of classification performance.

IV. Results

The performance of five pretrained convolutional neural network architectures was evaluated on the CIFAKE dataset. These include MobileNetV2, VGG16, InceptionV3, ResNet50, and EfficientNetB0. Each model was fine-tuned using a consistent classification head and trained under identical conditions for fair comparison. The resulting training and validation accuracies are summarized in Table I.

Model	**Training Accuracy (%)**	**Validation Accuracy (%)**
MobileNetV2	73.2	53.9
VGG16	78.3	56.1
InceptionV3	74.4	53.4
ResNet50	70.7	56.8
EfficientNetB0	55.7	48.5

TABLE I: Model Training and Validation Accuracies

The figures indicate that VGG16 achieved the highest training accuracy of 78%, closely followed by InceptionV3 and MobileNetV2. However, both VGG16 and ResNet50 reached the same optimal validation accuracy of 56%, which indicates a superior level of generalization compared to the other models. Interestingly, despite its modern architecture, EfficientNetB0 did not perform as effectively, possibly due to its sensitivity to the limited CIFAKE dataset or its requirement for longer training durations to attain effective generalization. Furthermore, a notable train–validation gap is observed across all models, indicating a mild to moderate level of overfitting. This issue may be associated with the relatively small size of the CIFAKE dataset, the high degree of similarity between genuine and counterfeit images, or an insufficient application of regularization techniques (such as dropout or more robust augmentation). VGG16 and ResNet50 exhibited the most promising generalization, suggesting that deeper classical architectures with fixed lower layers may be more effective for synthetic-real binary datasets like CIFAKE.

In addition to the accuracy metrics, the loss curves were examined to assess the generalizability of each model. Although all models demonstrated satisfactory training performance, the loss curves for ResNet50 and EfficientNetB0 displayed evident signs of overfitting, marked by a consistent decline in training loss while the validation loss increased after several epochs. This suggests that these models memorized the training data, but struggled to generalize effectively to unseen fake or real images.

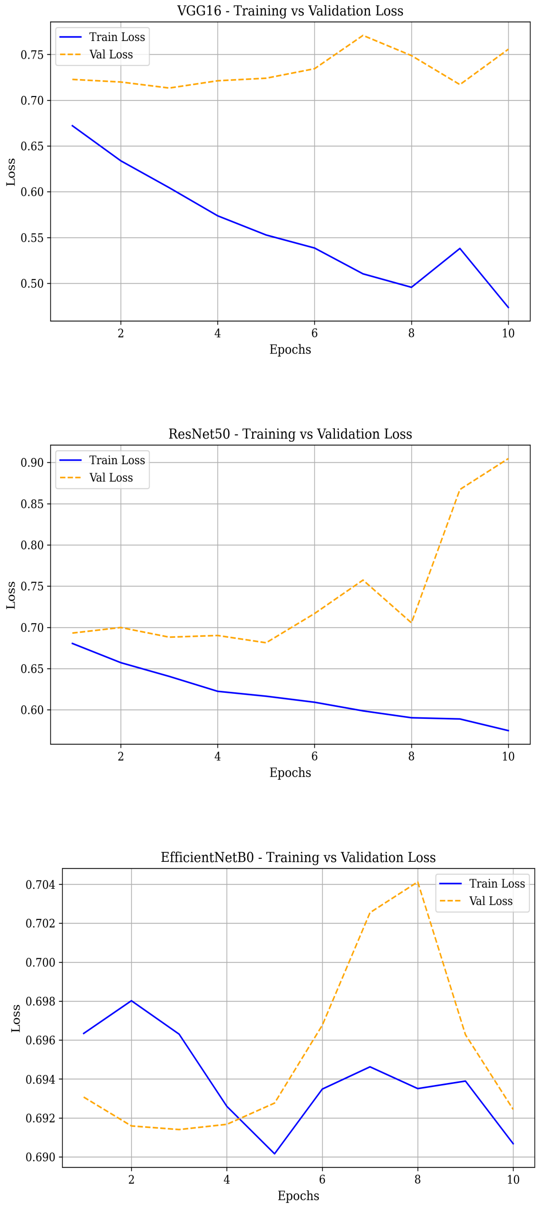

Fig. 7: Training and Validation Loss Curves for VGG16, ResNet50 and EfficientNetB0

In contrast, VGG16 not only achieved the highest training accuracy of 78% but also exhibited a relatively stable validation loss, indicating a superior generalization. Its loss curve showed a close alignment between training and validation losses with minimal divergence, further supporting its exceptional performance. The overfitting observed in ResNet50 and EfficientNetB0 may stem from their greater model capacity, which, when combined with a relatively small and visually subtle dataset like CIFAKE, results in inadequate generalization. These results highlight that simpler or moderately deep architectures, such as VGG16, can surpass more complex models in areas like synthetic image detection, where overparameterization heightens the risk of fitting to noise.

In addition to accuracy, the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) was utilized to evaluate each model's ability to distinguish between AI-generated and genuine images at different thresholds. The AUC results further validate the superiority of VGG16, which achieved the highest score of 0.57, indicating a modest but consistent discrimination capacity. ResNet50 closely followed with an AUC of 0.56. In contrast, MobileNetV2, InceptionV3 and EfficientNetB0 all hovered around 0.49 to 0.50, suggesting a performance similar to random guessing. These low AUC scores, especially when paired with relatively high training accuracies, reinforce the earlier observation of overfitting in these models. Overall, the AUC analysis aligns with the loss curves and validation accuracy, confirming that VGG16 demonstrated the most reliable generalization among the five architectures assessed.

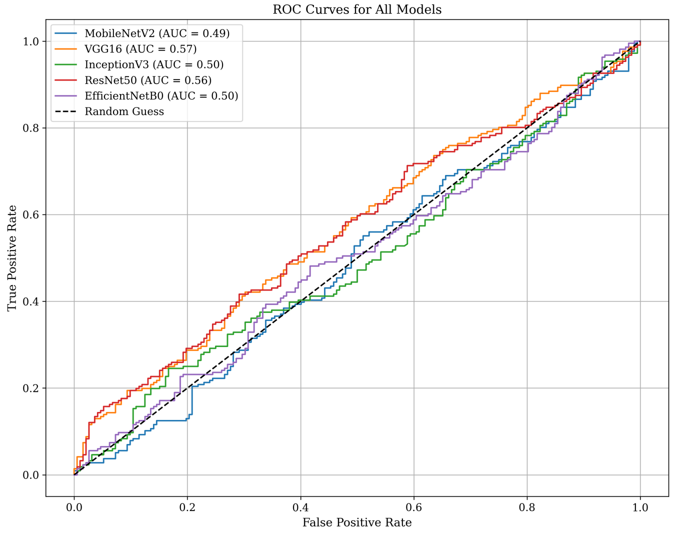

Fig. 8: ROC-AUC curves for all models.

V. Conclusion

This study examined the effectiveness of five pre-trained convolutional neural network architectures, that is, MobileNetV2, VGG16, InceptionV3, ResNet50, and EfficientNetB0, in detecting AI-generated fake images using the CIFAKE dataset. While all models exhibited satisfactory training performance, significant differences were observed in their generalization capabilities. VGG16 consistently outperformed the other models, achieving the highest validation accuracy of 56% and an AUC of 0.57, while also displaying stable loss curves that indicated minimal overfitting. In contrast, models such as ResNet50 and EfficientNetB0, despite their higher capacity, faced overfitting issues and demonstrated lower generalization. The results suggest that moderately deep architectures like VGG16 may offer a more advantageous balance between complexity and performance in situations with limited data, especially for identifying subtle artifacts in synthetic images.

References

[1] D. Cozzolino, G. Poggi, R. Corvi, M. Nießner, and L. Verdoliva, "Raising the bar of ai-generated image detection with clip," 2024.

[2] A. G. Moskowitz, T. Gaona, and J. Peterson, "Detecting ai-generated images via clip," 2024.

[3] Z. Sha, Z. Li, N. Yu, and Y. Zhang, "De-fake: Detection and attribution of fake images generated by text-to-image generation models," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, p. 3418–3432, Association for Computing Machinery, 2023.

[4] S. McCloskey and M. Albright, "Detecting gan-generated imagery using color cues," 2018.

[5] R. Corvi, D. Cozzolino, G. Zingarini, G. Poggi, K. Nagano, and L. Verdoliva, "On the detection of synthetic images generated by diffusion models," 2022.

[6] R. B. R. A. S. B. Misal Thakre, Satyam Kadu, "Detection of ai-generated images," *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 8, pp. 805–810, October 2024.

[7] R. Durall, M. Keuper, F.-J. Pfreundt, and J. Keuper, "Unmasking deepfakes with simple features," 2020.

[8] *Deep Learning vs. Traditional Computer Vision*. Springer International Publishing, 2020.

[9] E. R. S. de Rezende, G. C. S. Ruppert, A. Theophilo, and T. Carvalho, "Exposing computer generated images by using deep convolutional neural networks," 2017.

[10] N. Aldahoul and Y. Zaki, "Detecting ai-generated images using vision transformers: A robust approach for safeguarding visual media integrity," 01 2025.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[12] U. Ojha, Y. Li, and Y. J. Lee, "Towards universal fake image detectors that generalize across generative models," 2024.

[13] D. Cozzolino, G. Poggi, M. Nießner, and L. Verdoliva, "Zero-shot detection of ai-generated images," 2024.

[14] Y. Yao, W. Hu, W. Zhang, T. Wu, and Y.-Q. Shi, "Distinguishing computer-generated graphics from natural images based on sensor pattern noise and deep learning," *Sensors*, vol. 18, no. 4, 2018.

[15] R. Sabitha, A. Aruna, S. Karthik, and J. Shanthini, "Enhanced model for fake image detection (emfid) using convolutional neural networks with histogram and wavelet based feature extractions," *Pattern Recognition Letters*, vol. 152, pp. 195–201, 2021.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[17] K. Dong, C. Zhou, Y. Ruan, and Y. Li, "Mobilenetv2 model for image classification," in *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, pp. 476–480, 2020.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[19] D. Theckedath and R. Sedamkar, "Detecting affect states using vgg16, resnet50 and se-resnet50 networks," *SN Computer Science*, vol. 1, no. 2, p. 79, 2020.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015.

[21] S. R. Shah, S. Qadri, H. Bibi, S. M. W. Shah, M. I. Sharif, and F. Marinello, "Comparing inception v3, vgg 16, vgg 19, cnn, and resnet 50," *Agronomy*, vol. 13, no. 6, 2023.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[23] S. Mascarenhas and M. Agarwal, "A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification," in *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, vol. 1, pp. 96–99, 2021.

[24] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 2020.