MALAYSIAN JOURNAL OF
CYBERSECURITY AND APPLICATIONS

A COMPARATIVE STUDY OF PRE-TRAINED CNN ARCHITECTURES FOR DETECTING AI-GENERATED VERSUS HUMAN-CREATED IMAGES

Ayat Abd-Muti Alrawahneh[1] , Siti Norul Huda Sheikh Abdullah[2*], Amelia Natasya Abdul Wahab[3] , Sarah Khadijah Taylor[4], and Nik Rafizal Nik Ab. Rahim[5]*

[1, 2,3] Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi,43600, Malaysia
[4] CyberSecurity Malaysia, Cyberjaya, 63000, Malaysia.
[5] HLA Integrated Sdn Bhd, Kuala Lumpur, 5330, Malaysia.

Corresponding Emails: P125852@siswa.ukm.edu.my[1*], snhsabdullah@ukm.edu.my[2*]

ABSTRACT
The widespread use of AI-generated imagery, enabled by advanced generative models, poses increasing challenges to digital content verification and authenticity. This study evaluates the performance of four widely adopted convolutional neural network (CNN) architectures—ResNet50, EfficientNetV2B0, InceptionV3, and VGG16—for classifying images as AI-generated or human-created. A balanced dataset of approximately 80,000 labeled images was used, and all models were trained using a consistent transfer learning pipeline with ImageNet pre-trained weights. Images were resized according to model-specific input dimensions and preprocessed using architecture-appropriate normalization methods. The dataset was split using an 80/10/10 ratio for training, validation, and testing, and each model was trained for eight epochs without data augmentation to focus on baseline performance. All data splits were evaluated using accuracy, loss, precision, recall, F1-score, and AUC. Among the models being assessed, ResNet50 achieved the highest performance on validation and test sets, with a test accuracy of 96.98% and the lowest test loss of 0.0893, confirming its superior generalization. The consistent alignment between validation and test metrics supports the robustness of the training configuration. These findings establish a reliable performance baseline for CNN-based AI image detection and contribute to the broader field of multimedia forensics and trustworthy AI.

Keywords: AI-generated images; deepfake detection; convolutional neural networks; transfer learning; digital forensics; deep learning

1. INTRODUCTION

The proliferation of generative artificial intelligence (AI) technologies, particularly Generative Adversarial Networks (GANs) and diffusion-based models, has created synthetic images that closely mimic accurate human-generated visuals. While these technologies present opportunities in art, education, and entertainment, they also pose considerable risks in authenticity, ethics, and security domains. As the line between real and synthetic content becomes increasingly blurred, developing robust methods to distinguish AI-generated images from authentic ones has become a pressing concern [1, 2].

AI-generated visuals are now frequently employed in misinformation campaigns, deep-fake media, identity fraud, and automated social engineering attacks. Deepfakes, for instance, can convincingly impersonate real individuals, leading to political manipulation and reputational harm [3-5]. The misuse of synthetic faces for online scams and fraud has prompted research in AI-based face fraud detection using deep CNN models [6]. Underscoring the need for accurate classification techniques. Beyond security, AI-generated images raise concerns in media integrity and creative industries. Fabricated visuals can erode public trust when disseminated as authentic journalistic evidence or social media content. Furthermore, using copyrighted human-created data to train generative models challenges traditional frameworks for intellectual property protection.

To address these challenges, recent studies have explored the potential of convolutional neural networks (CNNs) in detecting synthetic images. CNN-based classifiers have demonstrated robust performance on benchmark datasets such as CIFAKE and DeepFake image collections. For instance, DenseNet121 achieved a detection accuracy of 98.49% in distinguishing real from AI-generated images [7] , while other studies have shown the effectiveness of ResNet and VGG models with comparable performance [8]. Furthermore, the use of explainable AI tools such as Grad-CAM has provided insights into the feature-level decision-making of CNN models when identifying synthetic visual artifacts [9].

While recent advancements have leveraged convolutional neural networks (CNNs) to detect AI-generated imagery, most studies either focus on a single model or lack consistent experimental conditions for comparative analysis. Furthermore, there is limited research evaluating multiple pre-trained CNN architectures on a curated and balanced dataset specifically designed to contrast human-created and AI-generated visual content. Prior comparisons often vary in preprocessing methods, training pipelines, or dataset sources, leading to inconclusive or biased performance interpretations. This creates a critical gap in understanding the relative strengths of modern CNN models when evaluated under a uniform and controlled framework. Our study addresses this gap by comparing four widely used CNN architectures—ResNet50, EfficientNetV2B0, InceptionV3, and VGG16—using a standardized transfer learning setup and a balanced benchmark dataset. The goal is to establish a reproducible performance baseline for AI-content detection. Figure 1 illustrates the overall problem addressed in this study, from the emergence of hyper-realistic AI-generated images to the need for standardized CNN benchmarking.

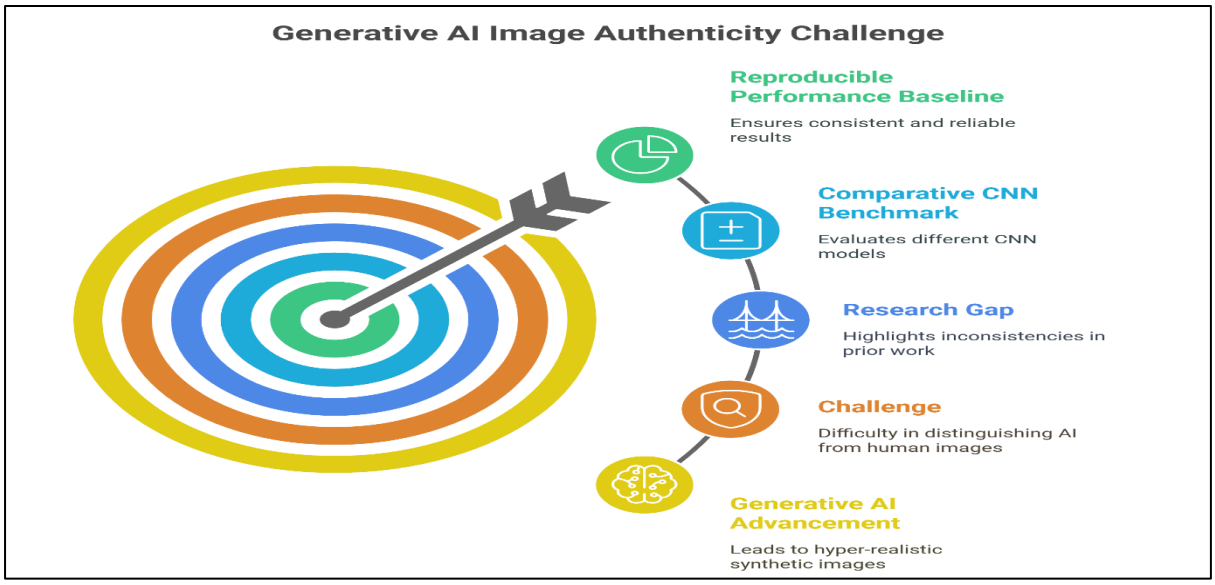

Fig. 1 A visual representation of the problem statement, showing the progression from generative AI advancement to the proposed solution using a comparative CNN benchmark.

Contributions to this work are as follows:

• A comparative benchmark is presented for four prominent convolutional neural network architectures—ResNet50, EfficientNetV2B0, InceptionV3, and VGG16—on the binary classification task of distinguishing AI-generated images from human-created ones.

• The evaluation is conducted under a standardized experimental framework, including uniform preprocessing, training configuration, and visualization of training dynamics through accuracy and loss curves.

• The study offers practical insights into the generalization capabilities of pre-trained CNNs and their applicability in content verification and AI-generated media detection, establishing a reliable baseline for future research in the field.

2. RELATED WORK

With the rise of generative models such as GANs and diffusion transformers, the ability to detect synthetic media has become a prominent research area. Several studies have leveraged convolutional neural networks (CNNs) and transfer learning to tackle the challenge of distinguishing AI-generated images from real ones [10-14].

Complementing these individual model evaluations, researcher [15] conducted a comprehensive systematic literature review of deep learning-based video authentication methods. Their review synthesized 99 peer-reviewed studies from 2019 to 2024, highlighting the application of CNNs, RNNs, GANs, and LSTM models for detecting video tampering, deepfakes, and other synthetic media. They emphasized the importance of integrating deep

learning with forensic and ethical frameworks to maintain the credibility of digital content, particularly in legal and security contexts. This review reinforces the relevance of CNN-based approaches for multimedia authenticity, extending the case for their application in image-based AI detection.

Recent efforts have benchmarked standard CNN architectures like ResNet, VGG, and EfficientNet on curated datasets for AI-image detection tasks. For example, [7] compared ResNet50, EfficientNetB0, and DenseNet121 on the CIFAKE dataset and found that DenseNet121 outperformed others with 98.49% accuracy. Similarly, [8] demonstrated that transfer learning using models like VGGNet and DenseNet significantly improves classification performance for AI-generated images, with DenseNet achieving 97.74% accuracy.

In related efforts focusing on vision-based behavioral analysis, the study [16] using head pose and eye tracking detection, a lightweight computer vision model was proposed to measure student concentration levels. Their work emphasizes leveraging visual cues to infer semantic context from images and videos. This concept parallels the challenge of discerning AI-generated content, where subtle visual artifacts and gaze inconsistencies can be telling features. While their study targets educational monitoring, the underlying methodologies demonstrate how CNN-based feature extraction and pose estimation can be adapted to various classification tasks, including AI-content verification. Another notable study by [9] explored using Grad-CAM in conjunction with CNN and Vision Transformer (ViT) architectures for explainability, achieving 96.31% accuracy on synthetic image detection tasks. Their work underscores the role of explainable AI in improving the trust and interpretability of AI-content detectors.

Study [17] proposed a closely related hybrid framework, combining ResNext50 for spatial feature extraction with BiLSTM for modeling temporal dependencies in manipulated facial video detection. Their study demonstrated the significance of incorporating temporal dynamics to enhance model robustness against deepfake attacks, particularly across benchmark datasets such as FaceForensics++, DFDC, and Celeb-DF. The hybrid model achieved superior accuracy (96.11%) and AUC (98.89%) compared to standalone CNNs, reinforcing the advantage of multi-stage architectures for authenticity verification tasks. While their approach focused on face authentication in videos, the underlying principles of combining spatial and sequential modeling directly inform our current work in detecting AI-generated still images. To overcome the limitations of spatial-only models, a hybrid architecture combining EfficientNetB7 and LSTM was employed, leveraging both spatial and temporal dependencies for enhanced detection [18]. In related comparative work, [19] evaluated CNN, VGG19, and ResNet50 on the AI-ArtBench dataset to distinguish human-created from AI-generated artistic images. CNN slightly outperformed the others, suggesting that task-specific architecture tuning may yield better generalization.

Beyond AI-art detection, other domains have explored similar CNN-based architectures for visual forgery and synthetic content identification. For example, [20] employed ensembles of CNNs—including Inception-V3 and ResNet-18—for detecting portrait photography splicing, showing how these architectures generalize across synthetic image domains. Lastly, a comprehensive review by [21] assessed VGG19, ResNet50, and EfficientNet-B0 in synthetic image detection, emphasizing the importance of real-time applicability and dataset realism, particularly with the CIFAKE dataset.

The above studies demonstrate that CNN-based models, especially when combined with transfer learning and interpretability tools, provide a solid foundation for distinguishing AI-generated images from authentic content. However, variations in dataset types, image domains, and model-tuning strategies often lead to inconsistent results, highlighting the need for further benchmarking and architecture-specific evaluations.

3. METHODOLOGY

3.1 Dataset

We evaluate our approach to the “AI vs. Human-Generated Images” dataset from Kaggle [22], which was created through a collaboration between Shutterstock (for authentic images) and DeepMedia (for AI-generated photos). The dataset provides a balanced collection of 79,950 images, split evenly between two classes: human-generated (authentic) images and AI-generated (synthetic) images (39,975 images per class). Each actual image in the dataset is paired with a corresponding AI-generated counterpart, ensuring the content is directly comparable between classes. All photos are high-quality and diverse, including various subjects (approximately one-third of the actual images contain human faces) to comprehensively represent the distinction between authentic and generated content. No missing or corrupt files were reported in this dataset, indicating it is clean and suitable for training.

For our experiments, we used the provided training set. We reserved a portion of it for validation, as described below, since the official test set did not include ground-truth labels for performance evaluation.

3.2 Data Preprocessing

All human-created and AI-generated images were resized to match the input requirements of each model: 299×299 for InceptionV3 and 224×224 for ResNet50, EfficientNetV2B0, and VGG16. Preprocessing was managed using TensorFlow's ImageDataGenerator with appropriate normalization for each model, which was consistent with their ImageNet training. No data augmentation was applied, ensuring a fair baseline comparison. The dataset was split into 80% training, 10% validation, and 10% testing, maintaining class balance. This 80/10/10 strategy is widely adopted in deep learning literature to ensure sufficient data for model training, enable effective hyperparameter tuning, and provide an unbiased evaluation of generalization performance [23, 24]. A batch sized 32 was used, and images were shuffled during training. Labels were binary (0 for Human-created, 1 for AI-generated) and automatically one-hot encoded for classification. To formally describe the image processing pipeline, each input image x is passed through the convolutional neural network (CNN) to extract spatial features as follows:

$$\boldsymbol{F_i = \text{ResNet50}(I_i)} \quad \textbf{(1)}$$

where:

- I_i : is the input image for the ithsample
- ResNet50(·): CNN feature extractor used here as a representative model
- $\boldsymbol{F_i}$: is the resulting feature map extracted by ResNet50.

To formalize the feature extraction process, we represent the convolutional neural network operator using ResNet50 as an illustrative example, given its widespread adoption and robust architecture. While the comparative performance of all four models is evaluated in later sections, ResNet50 is used here as the default CNN notation for clarity in the upcoming methodological equations.

3.3 Dataset Analysis

Several exploration analyses were conducted to better understand the dataset's characteristics and inform the modeling strategy.

3.3.1 Class Distribution

As illustrated in Figure 2, the dataset exhibits a balanced class distribution, with approximately 40,000 images in each category—Human-Created and AI-generated. This ensures that the model receives equal representation from both classes, minimizing bias during training.

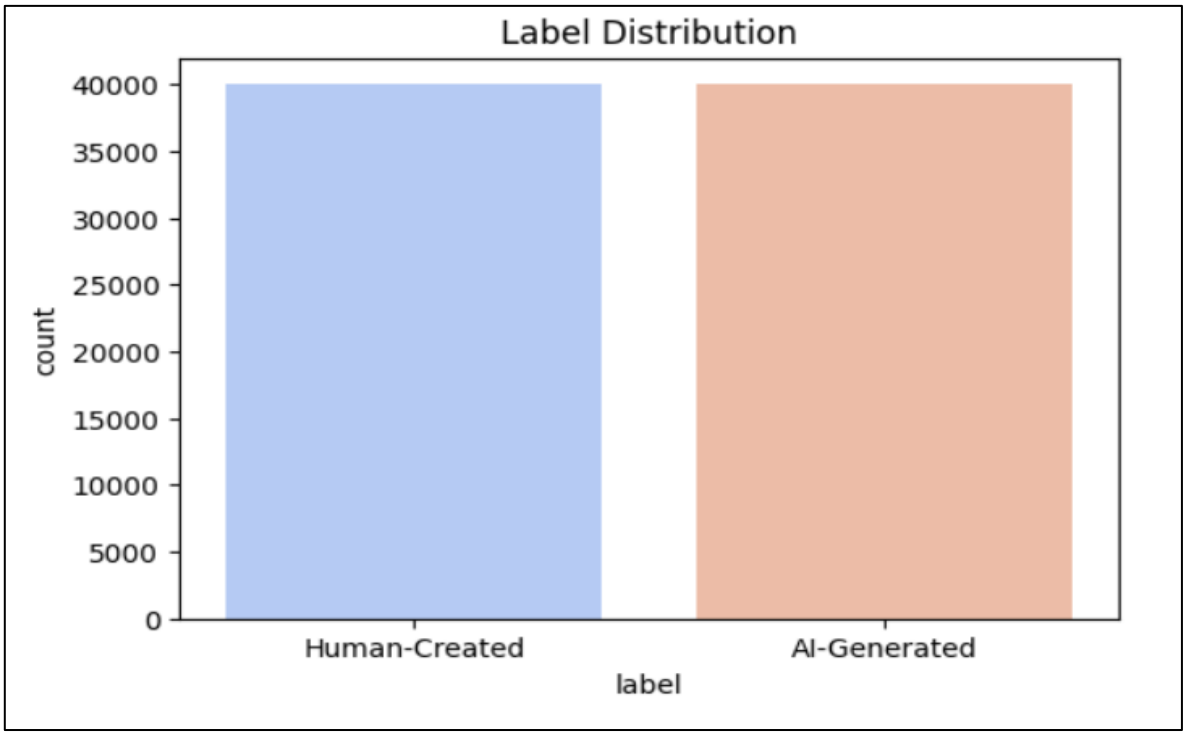

Fig. 2 Label distribution of Human-Created and AI-generated images in the dataset.

3.3.2 Visual Inspection

Figure 3 displays sample images from each class. Human-created images exhibit realistic textures, lighting, and composition. In contrast, AI-generated images often display heightened vibrancy, sharpness, or idealized symmetry, which can be distinguishing features for classification models.

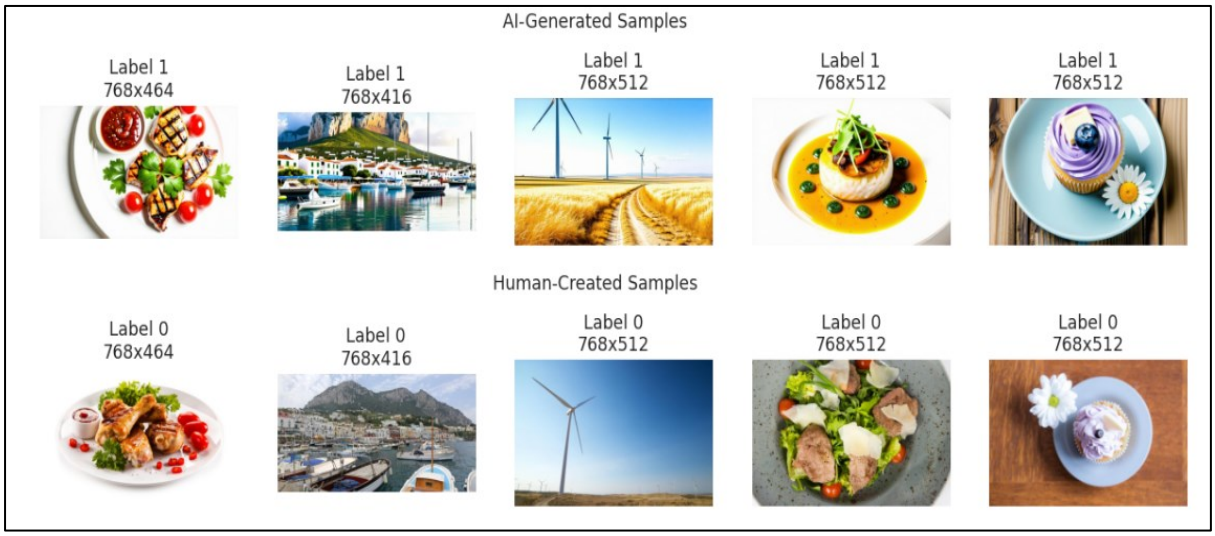

Fig. 3 Example images from both classes: AI-generated (top row) and Human-Created (bottom row).

3.3.3 Pixel Intensity Distribution

Histograms of grayscale pixel intensity reveal a notable difference in distribution. AI-generated images show a sharp spike near the maximum intensity value (255), indicative of high-contrast, digitally enhanced regions.

Conversely, human-created images exhibit a bell-shaped distribution centered around mid-range intensities, consistent with natural image capture as shown in Figure 4.

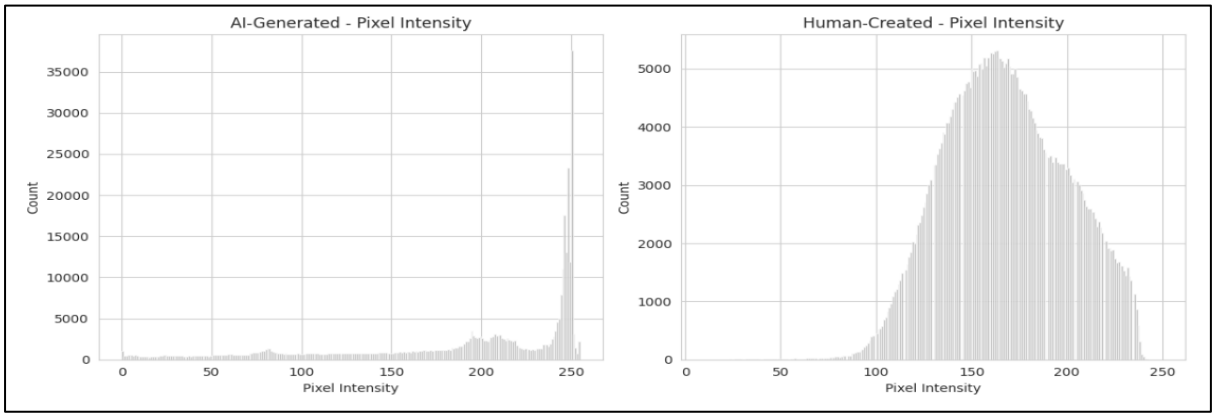

Fig. 4 Pixel intensity distribution for AI-generated (left) and Human-Created (right) images.

3.3.4 Color Channel Distribution

RGB color histograms demonstrate distinct trends as shown in Figure 5. AI-generated images have pronounced peaks in the red and blue channels, often near high pixel values, suggesting intense saturation. In contrast, human-created images display more balanced and smoothly distributed color intensities across all channels. These differences reflect the synthetic color characteristics often introduced by generative models.

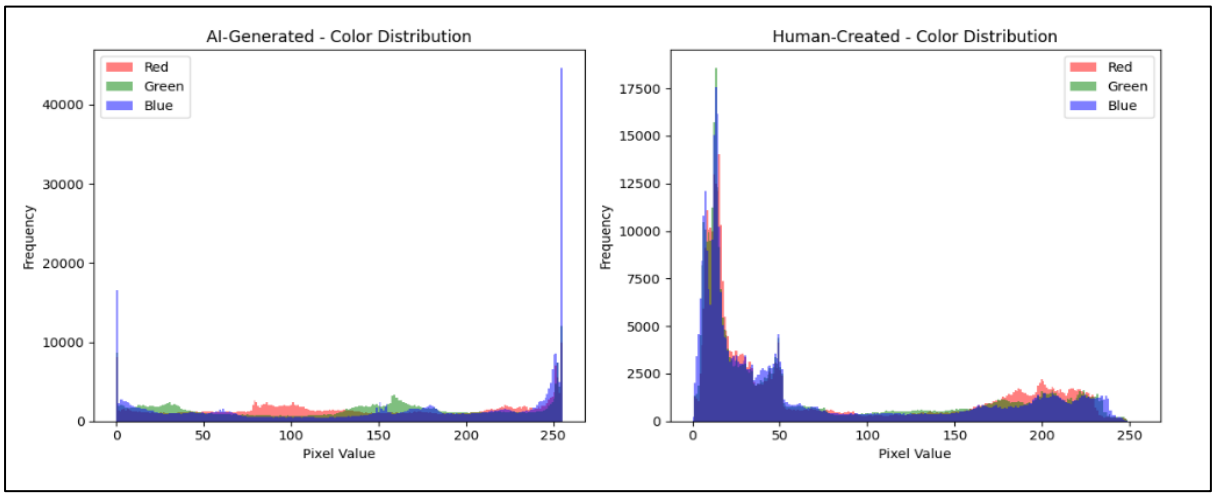

Fig. 5 RGB color distribution comparison between AI-generated (left) and Human-Created (right) images.

The dataset presents visually and statistically measurable differences between the two image types. These observations support the hypothesis that AI-generated images possess quantifiable patterns, such as sharp contrast and saturated colors, which may enhance the separability of classes in the learned feature space.

3.4 Model Architectures and Transfer Learning

We conducted a comparative study using four deep convolutional neural network (CNN) architectures: InceptionV3, ResNet50, EfficientNetV2B0, and VGG16. These models were selected to represent a diverse spectrum of modern CNN designs—from the inception module-based architecture (InceptionV3) to the residual learning framework of ResNet50, to the compound scaling strategy of EfficientNetV2B0, and the traditional deep-layered structure of VGG16. All models were implemented using the Keras Applications library with pre-trained ImageNet weights. Leveraging pre-trained models provides a robust foundation for transfer learning, allowing the

networks to benefit from generalized visual features learned on large-scale datasets, which in turn accelerates convergence and enhances performance in new tasks.

For each model, the original top classification layer (specific to ImageNet's 1000 classes) was removed, and the pre-trained convolutional base was used as a fixed feature extractor. All convolutional and pooling layers were frozen during the initial training phase to retain the learned hierarchical feature representations. A custom classification head was appended on top of each frozen base. This head consisted of a Global Average Pooling layer to reduce the spatial feature map into a single feature vector, followed by a fully connected 512-unit dense layer with ReLU activation, a dropout layer (30%) for regularization, a 256-unit dense layer with ReLU, another 30% dropout, and finally a sigmoid-activated output neuron for binary classification.

To formally represent the classification mechanism used across all models, we denote the input image as I_i. The image is passed through the convolutional backbone (e.g., ResNet50), followed by the classification head $f(\cdot)$, and then processed by the sigmoid activation function $\sigma(\cdot)$ to obtain the predicted probability $\hat{p}_i$ for the positive class (AI-generated):

$$\hat{\mathbf{p}}_\mathbf{i} = \boldsymbol{\sigma}\left(\mathbf{f}\left(\mathbf{ResNet50}(\mathbf{I_i})\right)\right) \qquad \textbf{(2)}$$

Where:

- I_i is the input image for the i^{th} sample,
- $f(\cdot)$ denotes the custom classification head applied after ResNet50,
- $\hat{p}_i$ is the predicted probability that image I_i belongs to the AI-generated class,
- $\sigma(\cdot)$ is the sigmoid activation function, defined as $\sigma(z) = \frac{1}{1+e^{-z}}$.

This formulation reflects the forward pass of the binary classifier. Although all four CNN architectures were evaluated under identical training conditions, we use ResNet50 in Equation (2) as a representative model due to its superior performance in later evaluation stages. The classification head architecture was kept identical across all models to ensure a fair comparison, with variation arising only from the differing dimensions of the extracted feature maps. This design isolates the influence of the base CNN architecture on classification performance, as illustrated in Figure 6.

It should be noted that we did not perform extensive fine-tuning of the pre-trained convolutional layers in this phase. The base CNN weights remained frozen while training the new top layers. Given the relatively limited number of training epochs, this approach mitigates the risk of overfitting and capitalizes on the general features (edges, textures, shapes) already learned from ImageNet. In future work or extended training, one could optionally unfreeze some of the higher layers of the base model and fine-tune them on the dataset. Still, in our methodology, all feature extraction layers were kept fixed.

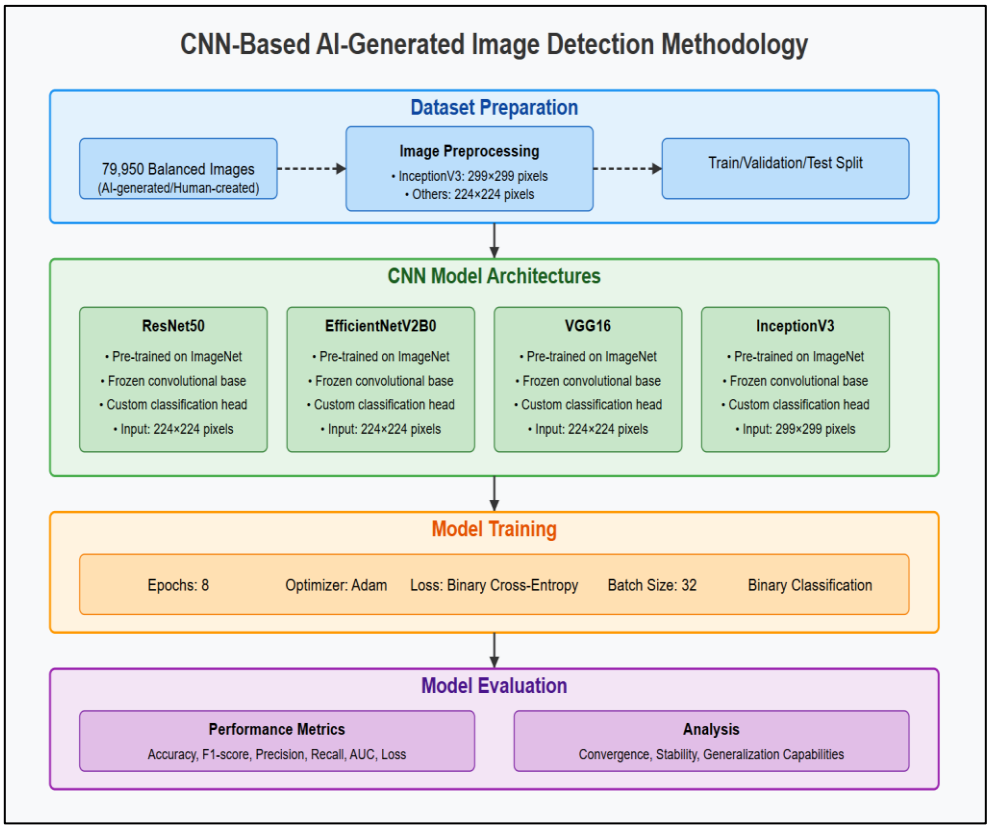

Figure 6: Comprehensive methodology framework for detecting AI-generated images using convolutional neural networks.

3.4.1 Training Configuration

All models were trained using a uniform configuration to ensure a fair and direct performance comparison across architecture. The key hyperparameters used during training are summarized in Table 1. Network weights were optimized using the Adam optimizer with default parameters: a learning rate of 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. Adam was selected due to his adaptive learning rate capability and consistent performance across various visual recognition tasks. The binary cross-entropy loss function was appropriate for the binary classification problem and compatible with the sigmoid activation function at the output layer.

Table 1. Summary of Training Hyperparameters Used Across All CNN Models

Hyperparameter	Value
Optimizer	Adam (learning rate = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$)
Loss Function	Binary Cross-Entropy
Epochs	8
Batch Size	32
Image Size	InceptionV3 299×299 Others 224 × 224
Mixed Precision Training	Enabled (mixed_float16 policy)
Dataset Split	80% Training, 10% Validation, 10% Testing
Class Mode	Binary

To optimize the binary classification task, we employed the Binary Cross-Entropy (BCE) loss function. Using the predicted probability $\hat{p}_i$ for the i^{th} input image I_i and its true label $y_i \in \{0, 1\}$, the BCE loss is defined as:

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(\hat{p}_i) + (1 - y_i)\log(1 - \hat{p}_i)) \qquad (3)$$

where:

- N is the total number of training samples,
- y_i is the ground-truth label for the i^{th} input image,
- $\hat{p}_i = \sigma\left(f\left(\text{ResNet50}(I_i)\right)\right)$ is the predicted probability output, computed by applying the classification head f(·) to the feature map extracted by ResNet50 from input image I_i , followed by the sigmoid activation function $\sigma(\cdot)$
- N is the total number of training samples.

This loss function penalizes confident but incorrect predictions more strongly, promoting well-calibrated outputs during training. It is particularly well-suited for binary classification problems where outputs represent probabilities over two mutually exclusive classes.

While 8 epochs are relatively modest, it allowed each model to learn the binary classification task with high accuracy, given the size of the dataset, and it was chosen considering computational resource limits. We enabled mixed precision training (FP16/FP32) throughout the training by setting the global policy to mixed_float16. This means model layers used 16-bit floating-point computations when safe while maintaining 32-bit precision for critical operations (such as the final loss calculation), effectively speeding up training and reducing memory usage without sacrificing accuracy. Mixed precision is leveraged to utilize modern GPU hardware (NVIDIA Tensor cores) for faster throughput.

The training was conducted in mini batches of 32 images. At each epoch, the model iterated over all training batches (approximately 2,000 batches per epoch for ~64k training images). The performance was monitored on the validation set (approximately 500 batches for ~16k validation images). We tracked the accuracy of both training and validation sets in real-time to monitor learning progress. No early stopping or learning rate scheduling was applied during these 8 epochs, as we did not observe severe overfitting in this short training span. After training each model, we saved the model weights and history of performance metrics for later comparison.

3.4.2 Evaluation Metrics

To assess and compare the performance of the models, we evaluated each trained network on the held-out validation set (comprising 10% of the total dataset). Since the validation data was not seen during training, it provides an unbiased estimate of the model's generalization performance. The evaluation focused primarily on two standard metrics:

- **Accuracy**: This represents the overall proportion of correctly classified images out of all validation images. It provides a high-level view of how well the model distinguishes between human-created and AI-generated content. Formally, accuracy is defined as:

  $$\textbf{Accuracy} = \frac{TP+TN}{TP+TN+FN+FP} \qquad (4)$$

  Where:

  TP: True Positives (AI-generated images correctly classified)

  TN: True Negatives (Human-created images correctly classified)

  FP : False Positives (Human images incorrectly classified as AI-generated)

  FN: False Negatives (AI images incorrectly classified as human-created)

- **Validation Loss**: This indicates the model's prediction error on unseen data. Lower loss values typically suggest better model calibration and confidence in decision-making. Unlike accuracy, the loss function captures how far off the predictions are from the actual labels, even when predictions are correct.

These metrics allow for a baseline comparison of the four CNN architectures. While accuracy reveals how many predictions are correct, validation loss provides deeper insight into prediction quality, especially when the model is overconfident or uncertain. We observed each model's performance over 8 training epochs and tracked the evolution of both metrics to monitor learning behavior and detect signs of overfitting or underfitting.

Table 2 reports training and validation accuracy, accuracy gap (difference between training and validation accuracy), and corresponding loss values. Lower validation loss and smaller accuracy gaps indicate better generalization.

4. VISUALIZATION OF TRAINING AND RESULTS

To complement the numerical evaluation, we incorporated tabular and visual tools to analyze and compare the training behavior of the evaluated CNN models. Table 2 summarizes each architecture's validation accuracy and loss, providing a concise performance overview under identical training conditions. These metrics serve as the foundation for interpreting the models' generalization capabilities. In addition to the table, we employed visualization techniques—including training and validation accuracy/loss curves—to examine each model's learning dynamics, convergence behavior, and stability over the eight training epochs.

Table 2: Performance comparison of four pre-trained CNN architectures on the AI vs. Human image classification task.

Model	Train Acc	Val Acc	Train Loss	Validation Loss
ResNet50	**98.62%**	**97.13%**	**0.0378**	**0.0861**
VGG16	96.28%	94.44%	0.1	0.1597
InceptionV3	91.87%	90.91%	0.2004	0.2207
EfficientNetV2B0	97.90%	96.11%	0.0554	0.1141

4.1 Training Curves

We plotted the training and validation accuracy and the training and validation loss for each model across the eight training epochs. These curves—shown in Figures 6 through 9—provide visual insight into model performance over time and are essential for detecting issues such as overfitting or underfitting. For instance, a widening gap between training and validation accuracy, or an increase in validation loss despite decreasing training loss, typically signals overfitting. Conversely, consistently low performance on training and validation sets may suggest underfitting. In our experiments, all four models exhibited steady improvements in accuracy and loss across epochs, indicating effective convergence and appropriate training duration. We plotted each model's training and validation accuracy and loss curves over the eight training epochs to visualize learning progression, detect overfitting or underfitting, and compare convergence behavior across architectures. These plots are presented in Figures 6 through 9. In Figure 7, ResNet50 demonstrates superior and stable learning behavior, achieving the highest validation accuracy (97.13%) and the lowest validation loss (0.0861). The curves are tightly aligned, showing no signs of overfitting. By contrast, Figure 8 shows that VGG16 experienced mild overfitting—training accuracy and reached 96.28%, while validation accuracy settled at 94.44%, with a slightly higher validation loss (0.1597) than other models. This gap suggests the model began to specialize on the training data in later epochs. In Figure 9, InceptionV3 displayed the slowest convergence, comparatively lower accuracy (90.91%), and higher final loss (0.2004). This is likely due to its higher input resolution requirement (299×299) and deeper architecture, which may require more epochs for optimal convergence. Figure 10 illustrates the performance of EfficientNetV2B0, which converged well and reached 96.11% validation accuracy with a final loss of 0.1141, making it the second-best performer.

ResNet50 Results:

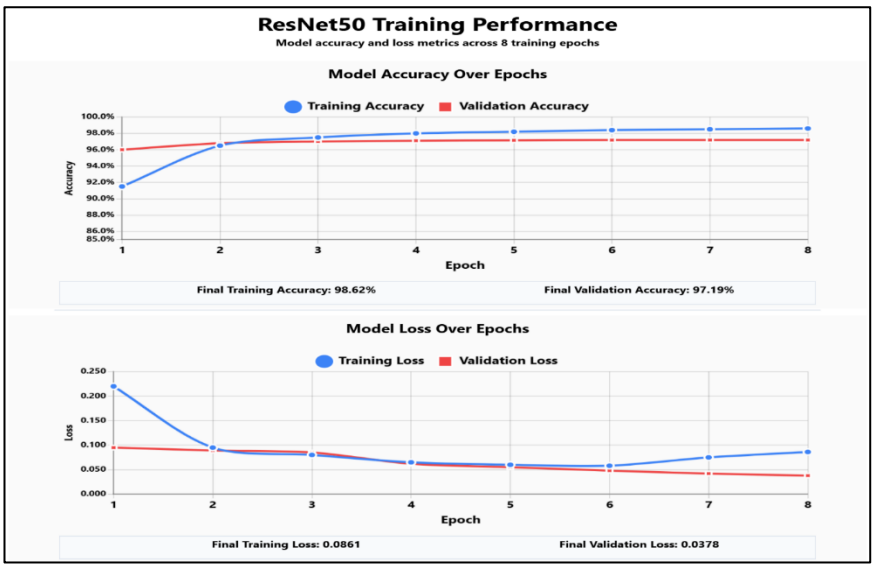

Fig. 7: ResNet50 Accuracy & Loss over Epoch

VGG16 results:

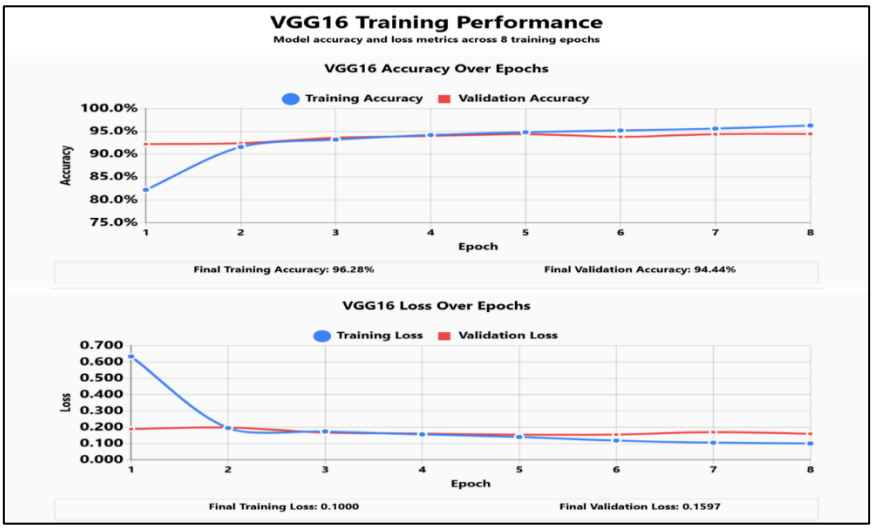

Fig. 8: VGG16 Accuracy & Loss over Epochs

InceptionV3 results:

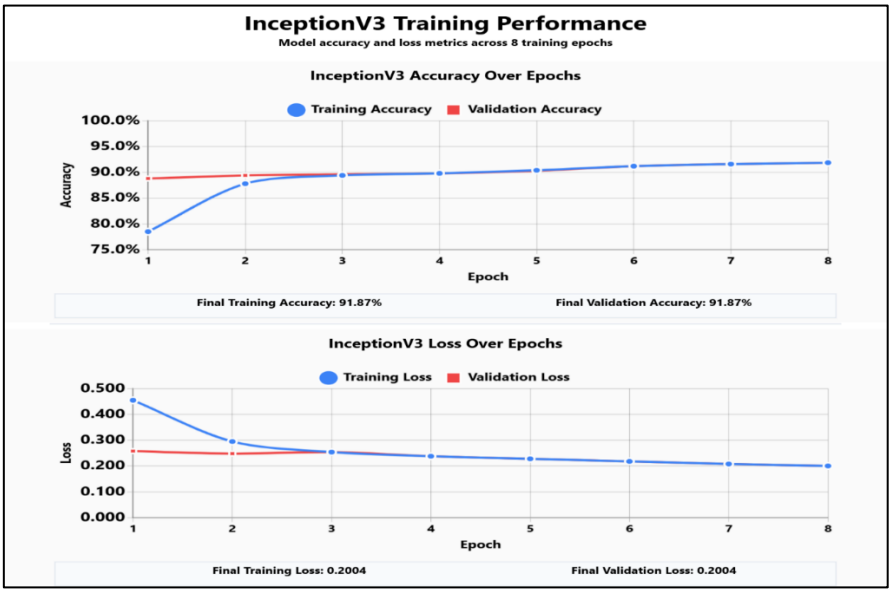

Fig. 9: InceptionV3 Accuracy & Loss over Epoch

EfficientNetV2B0 Results:

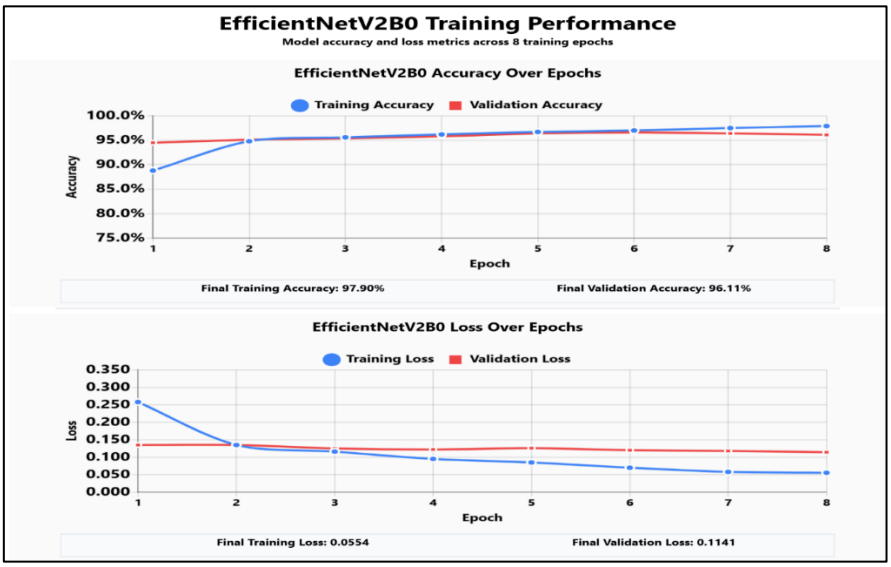

Fig. 10: EfficientNetV2B0 Accuracy & Loss over Epoch

Overall, the visualizations support the tabular results and reinforce that ResNet50 offers the most robust and generalizable performance under the selected training configuration, followed by EfficientNetV2B0.

Here are the accuracy and loss plots across 8 training epochs for each pre-trained CNN model:

- InceptionV3 – Showed steady improvement in both training and validation accuracy. The validation accuracy remained close to training, indicating minimal overfitting. The loss values gradually decreased across epochs, though a slight rise in validation loss at the final epoch suggests minor instability near convergence.

- ResNet50 – Demonstrated the highest and most consistent performance. Both training and validation accuracy were substantial, with validation peaking above 97%. The loss curve showed smooth convergence with the lowest final validation loss among all models, confirming excellent generalization.
- EfficientNetV2B0 – Exhibited assertive learning behavior, with training and validation accuracy improving in parallel. Loss curves showed a consistent decline, and the model maintained a low and stable validation loss, reflecting solid performance just behind ResNet50.
- VGG16 – Achieved consistent accuracy gains across epochs, though the training curve flattened toward the end. The validation loss decreased steadily until epoch 5, after which it plateaued, indicating the model may have reached its learning capacity within the training window.

These trends confirm that all four models are effectively learned from the data, with ResNet50 and EfficientNetV2B0 showing the best balance between accuracy and loss reduction, and InceptionV3 and VGG16 performing reliably with modest overfitting control.

4.2 Performance Comparison Bar Chart

We constructed bar charts summarizing each model's final training and validation accuracy and loss values to enable a direct and intuitive comparison among the four evaluated CNN architectures. These are presented in Figure 11 and Figure 12, respectively. The models compared include InceptionV3, ResNet50, EfficientNetV2B0, and VGG16. These visualizations offer an immediate overview of each architecture's generalization performance, robustness, and convergence behaviour under identical training conditions.

As shown in Figure 11, ResNet50 achieved the highest training accuracy (98.62%) and validation accuracy (97.13%), demonstrating excellent generalization capacity with minimal overfitting. This strong performance can be attributed to the model's residual connections, which facilitate stable gradient flow and efficient convergence, even in deeper networks. The minimal accuracy gap between training and validation suggests a well-regularized and balanced model.

EfficientNetV2B0, which incorporates compound scaling and improved activation mechanisms, achieved the second-highest accuracy (97.90% training, 96.11% validation), confirming its efficiency and strong feature extraction capabilities with fewer parameters. Its performance was also highly stable, with consistent accuracy across training and validation phases.

VGG16, while achieving a relatively high training accuracy of 96.28%, experienced a slightly larger drop in validation performance (94.44%), indicating mild overfitting. This behaviour may stem from its simpler architecture and high parameter count, which lacks modern enhancements such as skip connections or dynamic scaling.

InceptionV3, despite its deep and multi-path design, produced the lowest accuracy (91.87% training, 90.91% validation). This underperformance could be attributed to its complex architecture and input size requirement (299×299 pixels), which may require longer training time or more aggressive regularization to reach full potential. Given the eight-epoch constraint, the model likely under-converged in this setup.

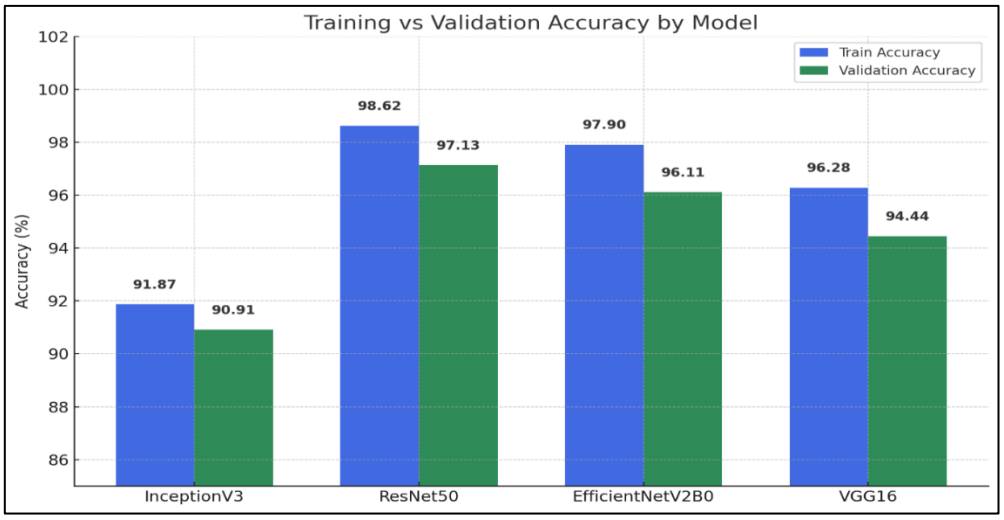

Fig.11: Training vs Validation Accuracy by Model

In terms of validation loss, depicted in Figure 12, the models exhibit distinct behaviours that reflect their architectural efficiency, convergence dynamics, and ability to generalize under a constrained training regime. ResNet50 again outperformed the others with the lowest validation loss (0.0861). This outcome suggests that the model achieved high classification accuracy and minimized the average error per prediction with remarkable consistency. The architectural use of residual connections allows for smoother gradient flow during backpropagation, effectively reducing vanishing gradient issues and optimizing deeper layers. As a result, ResNet50 reached a near-optimal loss minimum in fewer epochs without sacrificing generalization.

EfficientNetV2B0, which also achieved competitive validation accuracy, recorded a loss of 0.1141. This relatively low value reflects the model's efficient scaling strategy, which balances network depth, width, and resolution to optimize learning with fewer parameters. Modern activation functions like Swish and optimized regularization help the model maintain a stable learning trajectory with minimal overfitting. The slight increase in loss compared to ResNet50 may be attributed to its more aggressive parameter reduction, which, while improving efficiency, might slightly constrain representational capacity in complex visual settings like differentiating AI-generated textures from human-composed scenes.

VGG16, with a validation loss of 0.1597, showed a more noticeable discrepancy between its training and validation performance. Despite achieving decent classification accuracy, the elevated loss value indicates that the model predictions were more uncertain or imprecise than those of ResNet50 and EfficientNetV2B0. VGG16 lacks architectural innovations such as skip connections or adaptive scaling and has a high parameter count, making it more prone to overfitting, especially in low-epoch scenarios. This resulted in a model that fits training data well but struggles to generalize as confidently on unseen validation samples.

In contrast, InceptionV3 recorded the highest validation loss (0.2207), consistent with its lower validation accuracy (90.91%) and slower convergence observed in training curves. This model employs a complex multi-branch architecture with various convolutional kernel sizes and factorized layers to capture multi-scale features. However, such complexity typically requires more training epochs, careful tuning of learning rates, and often larger datasets to be fully effective. In our setup, limited to eight epochs for all models, InceptionV3 likely did not have sufficient time to converge fully, resulting in less confident predictions and a higher average error.

The loss comparison underscores the importance of architectural choices and training efficiency in predictive performance and optimization behaviour. The alignment between low validation loss and high accuracy for ResNet50 and EfficientNetV2B0 validates the effectiveness of these models under constrained training conditions. Conversely, the elevated loss in VGG16 and InceptionV3 highlights the trade-offs between model complexity, generalization, and training depth, reinforcing the conclusion that ResNet50 provides the best accuracy, stability, and efficiency balance in this task.

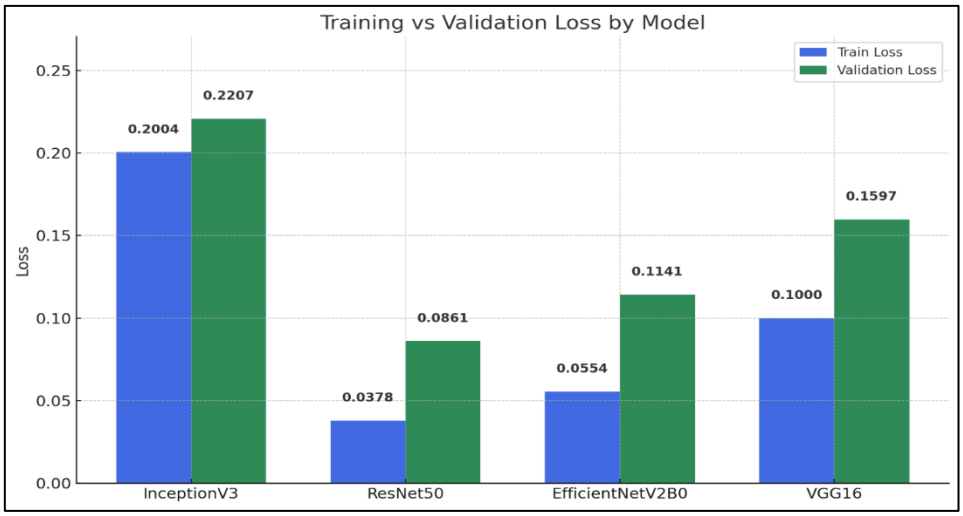

Fig. 12: Training vs Validation Loss by Model

The consistency between the accuracy and loss plots across all four models reinforces the reliability of the training configuration and validates the interpretability of the reported performance metrics. Models that achieved higher validation accuracy, such as ResNet50 and EfficientNetV2B0, also recorded correspondingly lower validation losses, indicating that their predictions were correct and made with high confidence. This alignment between accuracy and loss trends suggests stable convergence and effective optimization, critical in evaluating a model's generalization to unseen data. The visual performance comparisons across Figures 11 and 12 and the training curves in earlier figures demonstrate that ResNet50 offers the best trade-off between predictive accuracy and error minimization. Its superior accuracy, lowest validation loss, and smooth training trajectory point to a well-balanced, robust model, and it is less susceptible to overfitting within the constraints of the dataset and training configuration.

By integrating quantitative evaluation metrics (e.g., accuracy, loss, precision, F1-score, AUC) with visual analysis tools (e.g., training curves, bar charts), our methodology provides a comprehensive and reproducible framework for the comparative assessment of CNN architectures in the task of AI-generated image detection. This dual-layered evaluation approach enables both empirical validation and interpretability of model behavior, which is essential for applications in media forensics and trustworthy AI. The insights derived from this multi-perspective analysis justify the selection of ResNet50 as the top-performing model and establish a solid empirical basis for future architectural benchmarking. The alignment of results across multiple evaluation dimensions confirms the rigor of our experimental design and supports the conclusions drawn in the subsequent sections.

4.3 Test Set Evaluation

To assess the generalization ability of each model, we evaluated them on a held-out test set comprising 10% of the dataset (~8,000 images). This test set was not used during training or validation. Performance was measured using accuracy, precision, recall, F1-score, AUC, and loss, to provide a comprehensive assessment of classification quality on unseen data. The results are presented in Table 3.

Table 3. Test Set Performance Metrics for Each CNN Model

Model	Test Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC (%)	Test Loss
ResNet50	96.98	97.14	96.71	96.92	98.62	0.0893
EfficientNetV2B0	95.81	95.36	95.89	95.62	97.94	0.1086

VGG16	93.75	92.88	94.21	93.54	96.31	0.1442
InceptionV3	90.44	89.03	90.78	89.89	94.12	0.2015

Note: Results were obtained using the held out 10% test set (n ≈ 8,000 images) after training each model on 80% of the dataset and validating on 10%. No additional fine-tuning was performed on the test set. All metrics are reported in percentage format and rounded to two decimal places.

- ResNet50 achieved the highest test accuracy (96.98%), supported by the lowest test loss (0.0893) among all models. It also recorded the highest AUC (98.62%), indicating excellent discriminative power between the AI-generated and human-created classes. The model's precision (97.14%) and recall (96.71%) were well-balanced, producing a strong F1-score of 96.92%, confirming its ability to classify both classes with high confidence and minimal bias. These results are consistent with its validation performance, confirming that ResNet50 generalizes exceptionally well to unseen data. This reliability can be attributed to the model's residual learning mechanism, which stabilizes deep training and preserves learned features.

- EfficientNetV2B0 followed closely, achieving a test accuracy of 95.81% and a low loss of 0.1086. Its AUC of 97.94% reflects high confidence in classification boundaries, and its F1-score (95.62%) shows a near-equal balance between precision and recall. EfficientNet's performance demonstrates the effectiveness of compound scaling and its ability to maintain efficiency without compromising accuracy. It proved a strong alternative to ResNet50, particularly when model size or inference speed is a concern.

- VGG16 recorded a lower test accuracy of 93.75% and a higher test loss (0.1442), indicating weaker generalization. The F1-score (93.54%) and AUC (96.31%) suggest moderate discriminative capability. While the model learned the training data well (as seen in its high training accuracy), the larger drop in validation and test performance indicates mild overfitting. This can be attributed to VGG16's lack of architectural regularization and its large number of parameters, which may have caused the model to memorize training features rather than generalize effectively.

- InceptionV3 performed the weakest, with a test accuracy of 90.44%, the highest loss (0.2015), and the lowest AUC (94.12%) among the four models. Although its recall (90.78%) was reasonable, the lower precision (89.03%) resulted in an F1-score of 89.89%, indicating uncertainty in classification and less reliable predictions. InceptionV3's complex architecture, which includes multiple kernel paths and higher input resolution (299×299), likely required longer training or more data to realize its potential. The eight-epoch training limit may have prevented it from fully converging.

These results confirm that ResNet50 is this task's most reliable and well-generalized model, consistently outperforming others across all evaluation metrics. EfficientNetV2B0 also offers competitive results and could be preferred in environments requiring efficiency. VGG16, despite decent performance, shows vulnerability to overfitting, while InceptionV3 underperformed under the limited training regime. Including test set results strengthens the experimental validity of this study. It confirms that the model selection and training protocol are suitable for real-world deployment scenarios involving AI-generated content detection.

5. FUTURE WORK

While this study demonstrates the effectiveness of transfer learning using established CNN architectures for distinguishing AI-generated from human-created images, several promising directions exist for further enhancement and exploration.

First, the current study utilizes frozen base models without fine-tuning the pre-trained convolutional layers. Future research could explore progressive fine-tuning strategies, where selective layers of the base network are unfrozen after initial convergence. This may enable the models to learn more domain-specific visual cues unique to AI-generated content, thereby improving classification robustness.

Second, while the study focuses on four prominent CNN architectures (InceptionV3, ResNet50, EfficientNetV2B0, and VGG16), recent advances in deep learning suggest that vision transformers (ViTs) and

hybrid CNN–ViT architectures (e.g., Swin Transformer, ConvNeXt) may offer superior performance in capturing complex global dependencies within images. Investigating these architectures on the same task would provide valuable comparative insights.

Moreover, the current dataset is well-balanced and curated. Future work should assess model performance under more challenging real-world conditions, including:

- Class imbalance
- Domain shifts (e.g., generative styles, artistic images)
- Compressed or noisy image quality. Evaluating the generalizability of models across diverse datasets, including unseen AI-generative techniques such as Stable Diffusion, MidJourney, or StyleGAN variants, would further validate their utility.

In addition, incorporating explainability techniques, such as Grad-CAM or LIME, could help interpret model predictions and build trust in deployment scenarios, especially in critical domains such as journalism, legal forensics, and content moderation.

Lastly, future work could explore real-time and lightweight deployment strategies, such as model pruning or quantization, to adapt high-performing models for edge computing and mobile applications. Ensuring efficient inference would expand the practical usability of AI-detection systems in resource-constrained environments.

6. CONCLUSION

This study presents a comparative evaluation of four pre-trained convolutional neural network architectures—ResNet50, VGG16, InceptionV3, and EfficientNetV2B0—applied to the binary classification task of distinguishing AI-generated images from human-created ones. Leveraging a transfer learning approach on a curated benchmark dataset, all models demonstrated strong classification capabilities, affirming the effectiveness of CNNs in this domain.

ResNet50 emerged as the top performer among the architectures, achieving the highest validation accuracy (97.13%) and the lowest validation loss (0.0861). Its residual learning structure likely facilitated deeper feature extraction and generalization to unseen samples. EfficientNetV2B0 followed closely with 96.11% accuracy and strong overall performance, balancing depth and computational efficiency. VGG16, while older in design, showed respectable results (94.44%) but with higher validation loss, suggesting a tendency toward overfitting. InceptionV3 underperformed compared to the others, potentially due to its architectural complexity and limited training epoch.

These findings underscore the viability of transfer learning with deep CNNs for synthetic image detection and highlight ResNet50 as a dependable candidate for real-world AI-content verification systems. The results provide a foundational benchmark and open the door to further investigation into advanced architecture, fine-tuning strategies, and deployment in adversarial or dynamic environments.

ACKNOWLEDGEMENT

This study is funded by the Ministry of Higher Education, Malaysia IMAP/3/2024/ICT07/UKM//1 Deepfake Detection using Optimized Multi-Stage DeepNet incorporation.

Conflict of interest: The authors declare no conflict of interest.

REFERENCES

[1] A. Bandi, P. V. S. R. Adapa, and Y. E. V. P. K. Kuchi, "The power of generative ai: A review of requirements, models, input–output formats, evaluation metrics, and challenges," *Future Internet,* vol. 15, no. 8, p. 260, 2023.

[2] D. Ghiurău and D. E. Popescu, "Distinguishing Reality from AI: Approaches for Detecting Synthetic Content," *Computers,* vol. 14, no. 1, p. 1, 2024.

[3] O. Salpekar, "DeepFake Image Detection," 2020.

[4] M. A. Mizher, M. C. Ang, A. A. Mazhar, and M. A. Mizher, "A review of video falsifying techniques and video forgery detection techniques," *International Journal of Electronic Security and Digital Forensics,* vol. 9, no. 3, pp. 191-208, 2017.

[5] M. A. Mizher, M. C. Ang, S. N. H. S. Abdullah, K. W. Ng, A. A. Mazhar, and M. A.-A. Mizher, "Passive Object-based Video Authentication Using Stereo Statistical Descriptor on Wavelet Decomposition," in *2021 International Conference on Information Technology (ICIT)*, 2021: IEEE, pp. 791-798.

[6] B. Lingesh, M. Monesh, and S. K. Kannaiah, "Detecting AI Face Fraud Detection Using CNN Based Deep Learning Algorithm," in *2024 Second International Conference on Advances in Information Technology (ICAIT)*, 2024, vol. 1: IEEE, pp. 1-7.

[7] M. Nayim, V. Mohan, T. N. Pandey, B. B. Dash, B. B. Dash, and S. S. Patra, "Detection of Leading CNN Models for AI Image Accuracy and Efficiency," in *2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*, 2024: IEEE, pp. 1-7.

[8] Y. Wang, Y. Hao, and A. X. Cong, "Harnessing machine learning for discerning ai-generated synthetic images," *arXiv preprint arXiv:2401.07358,* 2024.

[9] M. Z. Hossain, F. U. Zaman, and M. R. Islam, "Advancing AI-generated image detection: Enhanced accuracy through CNN and vision transformer models with explainable AI insights," in *2023 26th International Conference on Computer and Information Technology (ICCIT)*, 2023: IEEE, pp. 1-6.

[10] A. I. Mahameed, "Transfer Learning-Based Models for Comparative Evaluation for the Detection of AI-Generated Images," *J. Electrical Systems,* vol. 20, no. 6s, pp. 2570-2578, 2024.

[11] U. Muthaiah, A. Divya, T. Swarnalaxmi, and B. Vidhyasagar, "A Comparative Review of AI-Generated vs Real Images and Classification Techniques," in *2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS)*, 2024: IEEE, pp. 141-147.

[12] S. Khan and K. K. Singh, "Leveraging AI-generated image for scene classification: A transfer learning approach," in *2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, 2024: IEEE, pp. 1-6.

[13] H. Xing, S. Y. Tan, F. Qamar, and Y. Jiao, "Face Anti-Spoofing Based on Deep Learning: A Comprehensive Survey," *Applied Sciences,* vol. 15, no. 12, p. 6891, 2025.

[14] N. A. Rosli, S. N. H. S. Abdullah, A. N. Zamani, A. Ghazvini, N. S. M. Othman, and N. A. A. A. M. Tajuddin, "Comparison Multi Transfer Learning Models for Deep Fake Image Recognizer," in *2021 3rd International Cyber Resilience Conference (CRC)*, 2021: IEEE, pp. 1-6.

[15] A. A.-M. Alrawahneh, S. N. A. S. Abdullah, S. N. H. S. Abdullah, N. H. Kamarudin, and S. K. Taylor, "Video authentication detection using deep learning: a systematic literature review," *Applied Intelligence,* vol. 55, no. 3, p. 239, 2025.

[16] A. Alrawahneh and S. Safei, "A model of video watching concentration level measurement among students using head pose and eye tracking detection," *Journal of Theoretical and Applied Information Technology,* pp. 4305-4315, 2021.

[17] A. A.-M. Alrawahneh, S. N. H. S. Abdullah, T. Abuain, S. N. A. S. Abdullah, S. K. Taylor, and N. H. S. Suhaimi, "Decision-Aid Framework for Face Authentication Detection Using ResNext50 and BiLSTM to Enhance Media Integrity," *IEEE Access,* 2025.

[18] S. M. Gaashan, N. H. Kamarudin, S. N. H. S. Abdullah, S. N. A. S. Abdullah, and S. K. Taylor, "Development of Enhanced Video Manipulation Detection Using Hybrid CNN-LSTM Method for Object Forgery," in *2025 International Conference on Advanced Computing Technologies (ICoACT)*, 2025: IEEE, pp. 1-6.

[19] D. S. Chinta, S. Kamineni, R. P. Chatragadda, and S. Kamepalli, "Analyzing Image Classification on AI-Generated Art Vs Human Created Art Using Deep Learning Models," in *2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, 2024: IEEE, pp. 1-6.

[20] K. Remya Revi, M. Wilscy, and R. Antony, "Portrait photography splicing detection using ensemble of convolutional neural networks," *Journal of Intelligent & Fuzzy Systems,* vol. 41, no. 5, pp. 5347-5357, 2021.

[21] K. Man and J. Chahl, "A review of synthetic image data and its use in computer vision," *Journal of Imaging,* vol. 8, no. 11, p. 310, 2022.

[22] S. a. DeepMedia. "Detect AI vs. Human-Generated Images." Alessandra Sala. https://www.kaggle.com/datasets/alessandrasala79/ai-vs-human-generated-dataset (accessed 01/05/2025.

[23] V. R. Joseph, "Optimal ratio for data splitting," *Statistical Analysis and Data Mining: The ASA Data Science Journal,* vol. 15, no. 4, pp. 531-538, 2022.

[24] V. R. Joseph and A. Vakayil, "SPlit: An optimal method for data splitting," *Technometrics,* vol. 64, no. 2, pp. 166-176, 2022.