Comparative Analysis of Custom and Pre-Trained Convolutional Neural Networks (CNNs) for Object Recognition on the Cifar-10 Dataset

[1]Oluwadamilare (Asabia) Joseph Omoniyi; [2]Omotosho Olawale Jacob; [3]Ajaegbu Chigozirim; [4]Raymond Osi Alenoghena*; [5]Japinye Oluwaseun Abayomi; [6]Fatai Oguntade Aliu

[1,2,3]Department of Computer Science, Babcock University, Ilishan-Ogun, Nigeria

[4]Department of Economics, Caleb University, Imota Lagos, Nigeria

[5]Banking Supervision Department, Central Bank of Nigeria, Lagos, Nigeria

[6]Department of Business Administration, Trinity University, Yaba Lagos, Nigeria

***Corresponding Author**

DOI: https://doi.org/10.51244/IJRSI.2026.1306000253

Received: 14 June 2026; Accepted: 20 June 2026; Published: 03 July 2026

ABSTRACT

Convolutional Neural Networks (CNNs) have significantly changed image classification over the years by allowing computers to learn features directly from raw pixel data. However, deciding between building a customised model and using a pre-trained one can be a difficult task, especially when working with small datasets. In this study, we compare a custom CNN with three pre-trained models—VGG16, ResNet50, and MobileNetV2—on the CIFAR-10 dataset, which comprises 60,000 colour images (32×32 pixels) across 10 categories. We measured model performance using accuracy, precision, recall, F1-score, and training time. The results show that pre-trained models performed much better than the customised model. ResNet50 had the highest accuracy at 92.4%. However, MobileNetV2 gave the best mix of speed (1,800 seconds to train) and accuracy (90.2%). The custom CNN reached 82.3% accuracy, used less memory, and did not need image resizing. These results offer clear benchmarks for choosing models in the face of limited resources. They also demonstrate that transfer learning can achieve strong performance, while showing that custom CNNs remain useful for learning and simple tasks.

Keywords: Convolutional Neural Networks; Transfer Learning; CIFAR-10; Object Recognition; Image Classification; Deep Learning

JEL Classification: C45, C55, C63, O33

INTRODUCTION

Object recognition is the process of identifying and classifying objects in digital images. It is a key challenge in computer vision and is used in areas such as self-driving cars, medical imaging, facial recognition, and industrial quality control (Goodfellow et al., 2016). Older methods that rely on handcrafted features, such as Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG), have important limitations when dealing with complex, real-world images because they cannot represent features as effectively (LeCun et al., 2015; Oluwadamilare et al., 2025).

Deep learning, especially Convolutional Neural Networks (CNNs), has changed image classification by allowing computers to automatically learn features from raw pixel data. When AlexNet won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), it demonstrated that deep CNNs could outperform older computer vision methods (Krizhevsky et al., 2012). Since then, newer models such as VGGNet (Simonyan & Zisserman, 2014), ResNet (He et al., 2016), Inception (Szegedy et al., 2015), and MobileNet (Sandler et al., 2018) have further improved performance.

The CIFAR-10 dataset (Krizhevsky, 2009) is a common benchmark for testing image classification models because it has 10 balanced classes, 6,000 images per class, and is moderately complex. This makes it a good choice for comparing different CNN architectures. Even though CNNs work well, researchers still have to decide whether to build custom CNN models from scratch or use pre-trained models with transfer learning. Custom models offer greater flexibility and usually require less computing power, but they require substantial training data and careful tuning. Pre-trained models use features learned from large datasets like ImageNet, which can lead to higher accuracy and faster training. However, they can be more demanding to run and might not work as well on smaller datasets (Pandit & Kumar, 2020; Vinay & Balasubramanian, 2023; Alenoghena et al., 2026).

This study aims to address this gap by comparing a custom CNN model with three popular pre-trained models (VGG16, ResNet50, MobileNetV2) on the CIFAR-10 dataset using standard performance measures. The main goals are to: (1) build and train a custom CNN for CIFAR-10; (2) use transfer learning with the three pre-trained models; (3) evaluate all models based on accuracy, precision, recall, F1-score, and training time; and (4) find out which model works best for different situations.

RELATED WORKS

Convolutional Neural Networks

CNNs are specialized neural network architectures designed for processing grid-like data such as images. The fundamental operations include convolutional layers that extract hierarchical features using learnable filters, pooling layers that reduce spatial dimensions while preserving salient information, activation functions (predominantly ReLU) that introduce non-linearity, and fully connected layers that perform final classification (LeCun et al., 2015; Goodfellow et al., 2016). The inductive biases of local connectivity, weight sharing, and spatial subsampling enable CNNs to achieve translation invariance and parameter efficiency compared to fully connected networks.

Transfer Learning

Transfer learning enables knowledge acquired from one task to be leveraged for a related task. In the context of CNNs, this typically involves initializing a model with weights pre-trained on a large dataset (e.g., ImageNet), removing the original classification head, and fine-tuning the network on the target dataset (Prathivi, 2020; Adewale et al., 2025). This approach is particularly valuable when the target dataset is relatively small, as it mitigates overfitting and accelerates convergence.

CIFAR-10 Dataset

The CIFAR-10 dataset from the Canadian Institute for Advanced Research contains 60,000 colour images sized 32×32 pixels, split evenly among ten classes: aeroplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck (Krizhevsky, 2009). There are 50,000 images for training and 10,000 for testing. While its moderate size and resolution make it manageable for computation, the dataset remains challenging to classify due to intra-class variability and low image quality.

Prior Comparative Studies

Many researchers have tested convolutional neural networks (CNNs) on the CIFAR-10 dataset. Ghafouri (2024) reported over 85% accuracy using custom CNNs with batch normalisation and data augmentation. Pandit and Kumar (2020) reached 88% accuracy with optimised Keras models. Prathivi (2020) showed that VGG16 with transfer learning achieved 89% accuracy. Wang et al. (2017) achieved over 90% accuracy using residual attention networks. Vinay and Balasubramanian (2023) compared CNNs to cybernetic methods and found that CNNs worked better for classification.

However, most studies focus on either custom CNNs or pre-trained models, and there are few direct comparisons using the same evaluation methods.

Also, most studies do not mention how long training takes or what computing resources are needed, even though these are important for real-world use. This study aims to fill these gaps by comparing four architectures on several performance measures.

METHODOLOGY

Experimental Design

This study used an experimental research design. The independent variable was the model architecture, which included custom CNN, VGG16, ResNet50, and MobileNetV2. The selected dependent variables were performance metrics, including accuracy, precision, recall, F1-score, and training time. All experiments were run on the same hardware and software to make sure the results could be compared fairly.

Hardware and Software

The experiments ran on a system with an Intel Core i7 processor, 16GB of RAM, and an NVIDIA GPU with 8GB of VRAM that supports CUDA. The software setup was carefully selected to include a cocktail of Python 3.9, TensorFlow 2.10 with Keras, NumPy, Matplotlib, Seaborn, and Scikit-learn. Development and testing were done using Jupyter Notebook and Google Colab.

Dataset and Preprocessing

The CIFAThe CIFAR-10 dataset was loaded using the Keras datasets module (keras.datasets.cifar10). The preprocessing steps were carefully ordered to systematically follow the sequence: pixel values were normalized from [0, 255] to [0, 1]; images were resized from 32×32 to 224×224 for pre-trained models to match input size requirements; data augmentation included random horizontal flips, random rotations up to 15 degrees, and a zoom range of 0.2, all applied to the training set to help prevent overfitting; finally, class labels were one-hot encoded.

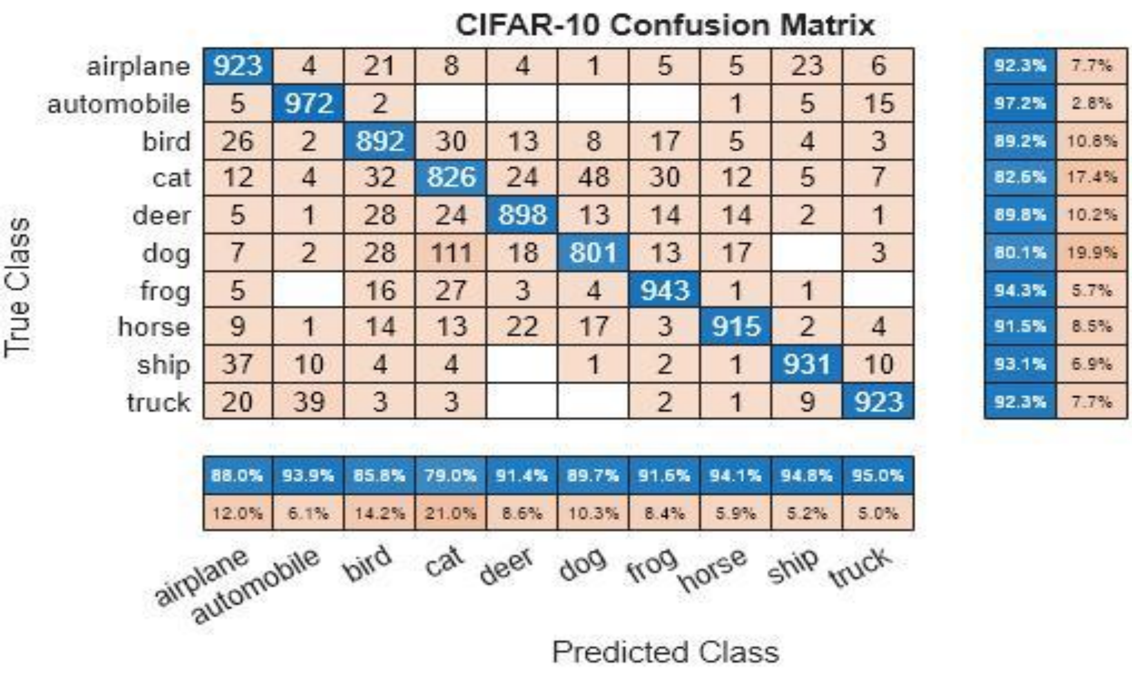

Figure 3.1: CIFAR-10 Confusion Matrix

Model Architectures

Custom CNN is a sequential architecture design that comprises, three convolutional layers with ReLU activation (32, 64, and 128 filters, each 3×3), each followed by 2×2 max pooling, a flattening layer, a dense layer (128 units, ReLU), dropout (0.5) for regularisation, and a softmax output layer (10 units). Pretrained models, VGG16, ResNet50, and MobileNetV2, were loaded with ImageNet weights via Keras. The original classification heads were removed, and a global average pooling layer was added, followed by a dense layer (128 units, ReLU), dropout (0.5), and a softmax output layer (10 units). Early layers were frozen, while later layers were fine-tuned.

Training Protocol

All models were trained using identical hyperparameters: batch size of 64, a maximum of 50 epochs with early stopping (patience of 5 epochs on validation loss). The training also took into account Adam optimiser (learning rate = 0.001), categorical cross-entropy loss function, and 20% validation split of training data.

Evaluation Metrics

Performance was evaluated using:

Accuracy: (TP + TN) / (TP + TN + FP + FN)

Precision: TP / (TP + FP)

Recall: TP / (TP + FN)

F1-Score: 2 × (Precision × Recall) / (Precision + Recall)

Training Time: Measured in seconds

Where TP = true positives, TN = true negatives, FP = false positives, FN = false negatives.

RESULTS

Overall Performance Comparison

Table 4.1 presents the comprehensive performance comparison of all four models. ResNet50 achieved the highest accuracy (92.4%), precision (91.8%), recall (91.5%), and F1-score (91.6%). MobileNetV2 achieved 90.2% accuracy with the shortest training time (1,800 seconds). The custom CNN achieved 82.3% accuracy with moderate training time (2,400 seconds).

Table 4.1: Performance Comparison of CNN Models on CIFAR-10

Model	Accuracy	Precision	Recall	F1-Score	Training Time (sec)
Custom CNN	82.3%	81.7%	80.9%	81.3%	2,400
VGG16	89.1%	88.5%	88.2%	88.3%	3,600
ResNet50	92.4%	91.8%	91.5%	91.6%	3,900
MobileNetV2	90.2%	89.7%	89.3%	89.5%	1,800

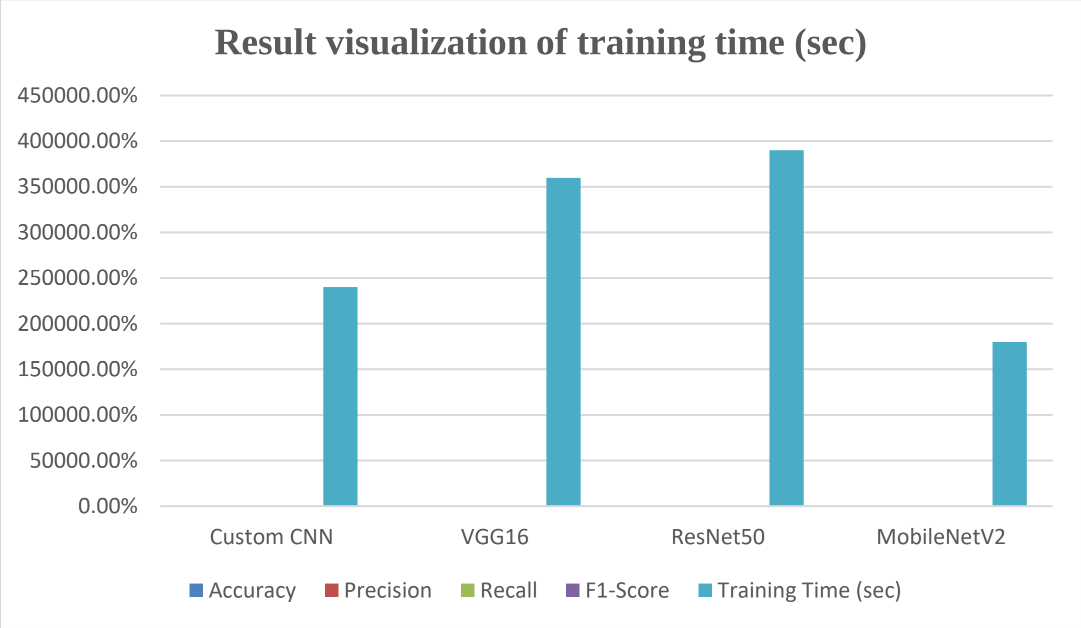

Figure 4.1 Result visualization of training time (sec)

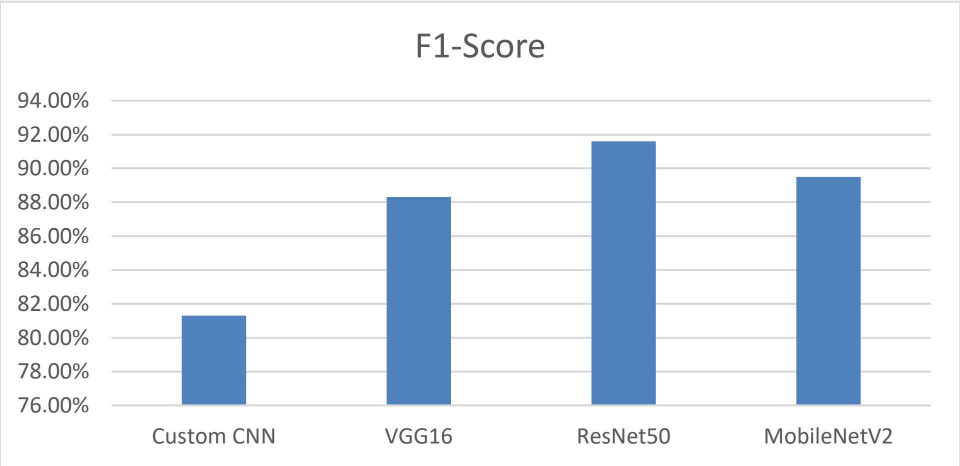

Figure 4.2: F1-Score Result

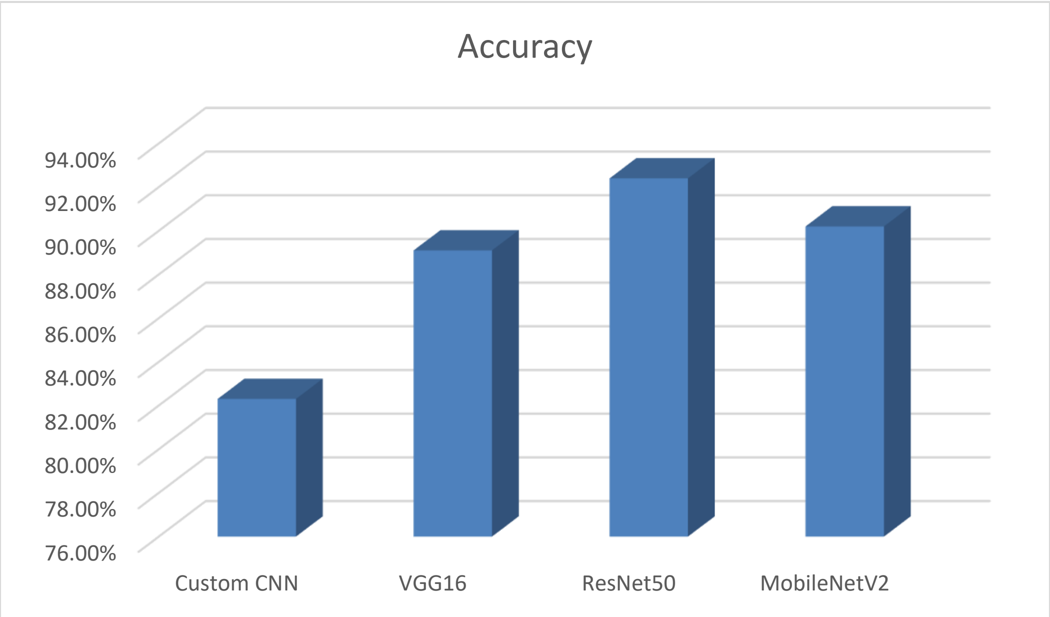

Figure 4.3: Accuracy Result

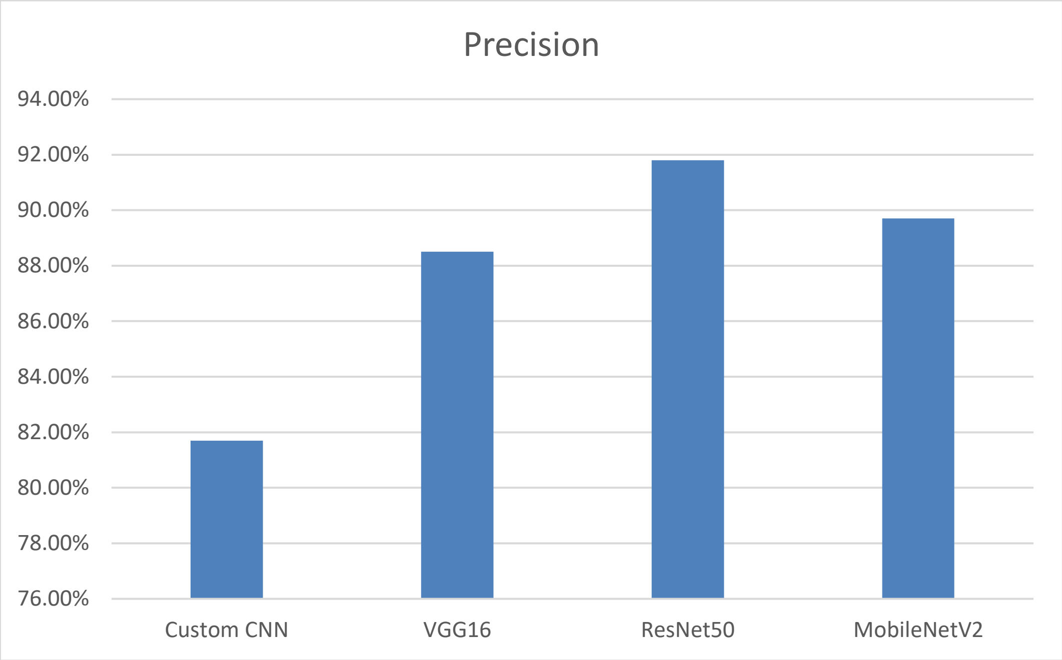

Figure 4.4: Accuracy Result

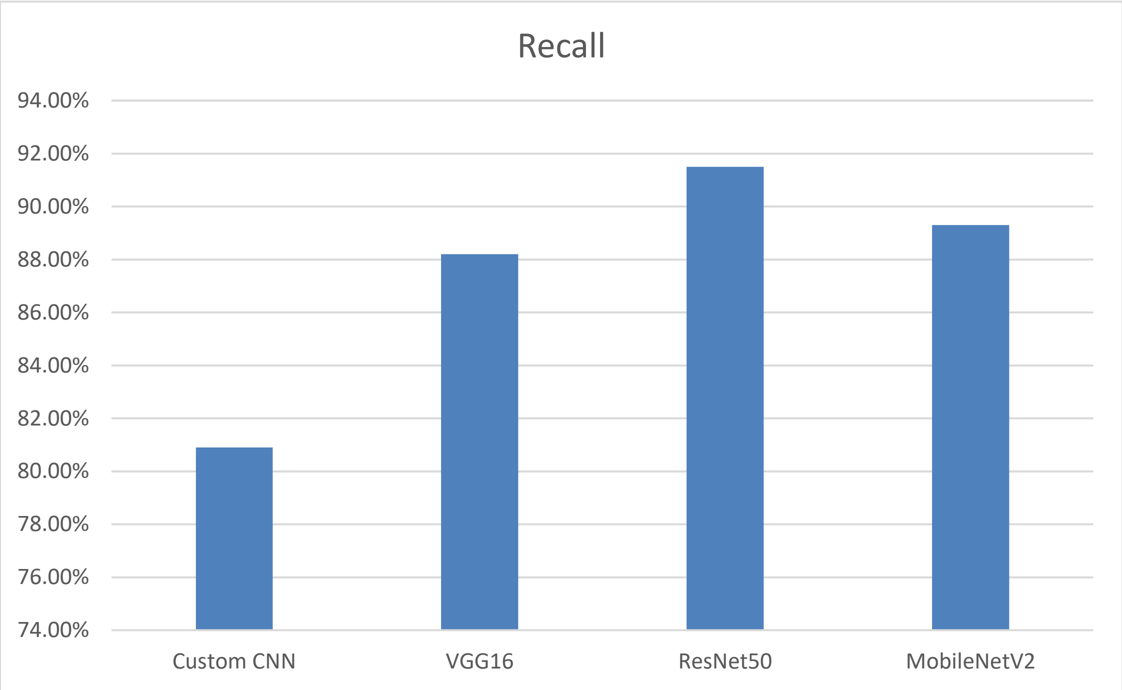

Figure 4.5: Recall Result

Training Convergence Analysis

Pre-trained models demonstrated superior convergence characteristics, achieving >85% accuracy by epoch 10. ResNet50 exhibited the smoothest convergence trajectory, attributable to its residual connections that mitigate vanishing gradient problems (He et al., 2016). The custom CNN reached 50% accuracy by epoch 10 and plateaued at approximately 82% after epoch 30, thus, requiring more extensive training to achieve convergence.

Computational Efficiency

MobileNetV2 was the most efficient model, finishing training in 1,800 seconds. This made it about 50% faster than VGG16, which took 3,600 seconds, and 54% faster than ResNet50, which took 3,900 seconds. Despite the speed, its accuracy stayed within 2.2 percentage points of the best model. The main reason for this efficiency is its use of depth-wise separable convolutions, which greatly lower the number of parameters and computations needed (Sandler et al., 2018).

DISCUSSION

Performance Advantages of Pre-trained Models

The 7 to 10 percentage point accuracy gap between pre-trained models and the custom CNN shows how important transfer learning is for small and medium datasets. Features learned from ImageNet, which has over 14 million images in 22,000 classes, create a strong and general feature set that works well for CIFAR-10 classification. These results support the main findings of this study. This pattern also matches earlier research showing that even when target images are very different, basic features like edges, textures, and color blobs are still useful (Prathivi, 2020; Ghafouri, 2024).

ResNet50: Optimal for Accuracy

ResNet50 achieved the highest accuracy at 92.4% thanks to its residual learning design, which solves the common problem of performance loss in deep networks. Its skip connections help gradients flow better during training and let the network learn by skipping layers when needed (He et al., 2016). This design is especially helpful for CIFAR-10, where the network needs to learn detailed features across many object types.

MobileNetV2: Optimal Efficiency-Accuracy Trade-off

MobileNetV2 offers both high accuracy (90.2%) and fast training (1,800 seconds), making it a good option for devices with limited resources, like edge devices, mobile phones, or real-time systems. Its design uses inverted residuals, linear bottlenecks, and depthwise separable convolutions, which reduce the number of parameters by about 10 times compared to standard networks while maintaining strong performance (Sandler et al., 2018; Oluwadamilare et al., 2025).

Custom CNN: Continued Relevance

The custom CNN is still useful for certain tasks, even though its accuracy is lower. It works directly with 32×32 CIFAR-10 images, so there is no need to resize and risk losing information. Its low memory usage and simple design make it well-suited for teaching, quick experiments, or when pre-trained models are unavailable due to licensing or hardware constraints (Vinay & Balasubramanian, 2023; Japinye et al., 2025). With improvements such as batch normalisation, more layers, or skip connections, custom models can approach the performance of simpler pre-trained models.

Practical Recommendations

Based on these findings, we propose the following guidelines for practitioners:

1. **For high accuracy applications:** ResNet50 with transfer learning should be deployed when computational resources permit.

2. **For edge or mobile deployment:** MobileNetV2 should be prioritize in order to balance accuracy and efficiency.

3. **For resource constrained environments:** Accept a moderate accuracy trade-offs while also considering custom CNNs with aggressive data augmentation,.

4. **As a general best practice:** Always incorporate data augmentation regardless of architecture choice to improve generalization.

CONCLUSION AND FUTURE WORK

This research study compares custom and pre-trained convolutional neural network (CNN) models for object recognition on the CIFAR-10 dataset. The results show the model from pre-trained experimentation performs much better than custom ones in both accuracy and learning speed. ResNet50 achieves the best accuracy of 92.4%, while MobileNetV2 balances accuracy and training time with 90.2% and 1,800 seconds, respectively. Custom CNNs remain useful for limited-resource situations or educational purposes, achieving 82.3% accuracy while using less memory.

These results provide clear benchmarks for selecting the right and most effective model while highlighting the weaknesses of transfer learning in relation to small image classification tasks. Future work in this area of study should compare these models on larger datasets such as CIFAR-100 and ImageNet subsets, explore transformer-based models such as Vision Transformer (ViT), investigate hybrid CNN-attention methods, test more pre-trained models like EfficientNet and DenseNet, and evaluate real-time deployment on devices such as the Raspberry Pi or NVIDIA Jetson Nano.

REFERENCES

1. Adewale, S.A., Alenoghena, R. O., Japinye, A. O., Imohi, G. D., & Oluwadamilare, J. O. (2025). The Impact of Foreign Direct Investment in Africa's Economic Growth: The Mediating Role of Financial Development, *International Journal of Research in Social Science and Humanities (IJRSS), 6* (12), 1–21. https://doi.org/10.47505/IJRSS.2025.12.1.
2. Alenoghena, R. O., Adebisi, S. D., Osunlana, K. I., & Ifarajimi, G. D (2026). Assessing the role of globalization and technology adoption in the economic growth of African countries. *Quality & Quantity*, 1-21. https://doi.org/10.1007/s11135-026-02624-1
3. Ghafouri, S. (2024). *Enhancing image classification accuracy using convolutional neural networks on CIFAR-10 dataset*. University of Victoria. https://hdl.handle.net/1828/20380
4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
5. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
6. Japinye, A. O., Opara, K. O., Alenoghena, R. O., Imohi, G. D., & Oguntade, A. F. (2025). Exploring the Contributions of Human Capital Development to Food Security in African Countries: The Mediating Influence of Technology Adoption. *International Journal of Innovative Science and Research Technology*, 10(8), 1082-1094.
7. Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images*. University of Toronto.
8. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 1097–1105.
9. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
10. Oluwadamilare, J. O., Alenoghena, R. O., Mariam, J., Sadiq, F. O. A., & Osunlana, K. I. (2025). Intelligent and Secure Digital Technologies: Artificial Intelligence, Internet of Things, and Information Management for Industrial Transformation. *International Journal of Humanities Social Science and Management (IJHSSM), 5*(6), 188-192.
11. Oluwadamilare, J. O., Alenoghena, R. O., Ajaegbu, N. G., Oguntade, F. A., Osunlana,, K. I., & Ajisafe, H. B. (2026). Project Charter Development and Its Influence on Software Project Success. *International Journal of Humanities Social Science and Management (IJHSSM), 6*(2), 403-409,
12. Pandit, S., & Kumar, S. (2020). Improvement in convolutional neural network for CIFAR-10 dataset image classification. *International Journal of Computer Applications*, 176(37), 25–29. https://doi.org/10.5120/ijca2020920489
13. Prathivi, R. (2020). The optimization of the convolutional neural network transfer learning model for CIFAR-10 image classification. *Jurnal RESTI*, 4(4). https://doi.org/10.29207/resti.v4i4.2131
14. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Conference on Computer Vision and Pattern*

Recognition, 4510–4520.

15. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
16. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.
17. Vinay, S. B., & Balasubramanian, S. (2023). A comparative study of convolutional neural networks and cybernetic approaches on CIFAR-10 dataset. *International Journal of Machine Learning and Cybernetics*, 1(1), 1–12. https://doi.org/10.17605/OSF.IO/QY32B
18. Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., & Tang, X. (2017). Residual attention network for image classification. *arXiv preprint arXiv:1704.06904*.

SUPPLEMENTARY MATERIAL

Data and Code Availability: Complete source code, trained models, and supplementary results are available from the corresponding author upon reasonable request.

Conflict of Interest Statement: The authors declare no conflicts of interest.

Acknowledgments: The authors thank the Computer Science Department of Babcock University, Ilishan for providing computational resources and research support.