Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Multiscale Fusion at What Cost? Quantifying Efficiency-Accuracy Trade-offs in Hybrid Models

Thomas Kinyanjui Njoroge[1], Kelvin Mugoye[2], Rachael Kibuku[3]
[1]Karatina University, School of Pure and Applied Sciences, Computer Science & Informatics Department, Kenya, tnjoroge@karu.ac.ke
[2] KCA University, School of Technology, Networks and Applied Computing Department, Kenya, kmugoye@kcau.ac.ke
[3] KCA University, School of Technology, Software Development & Information Systems (SD&IS) Department), Kenya, rkibuku@kcau.ac.ke

ARTICLE INFO

Received: 30 Dec 2024
Revised: 05 Feb 2025
Accepted: 25 Feb 2025

ABSTRACT

Multiscale feature fusion enhances deep vision models but often introduces computational overhead—an under-quantified challenge in hybrid CNN-Transformer architectures, especially for edge-based agricultural deployments. This study proposes an adaptive hybrid framework combining MobileNetV2, EfficientNetV2, and Transformers, trained on 76 classes across 22 crop diseases using Kaggle and field-sourced images. To address the efficiency-accuracy trade-off, we incorporate Squeeze-and-Excitation (SE) blocks (<1% parameter increase), gating mechanisms that reduce scale bias and improve small-object detection with marginal FLOPs cost, and hierarchical fusion, which raises FLOPs by 15% but yields diminishing returns on high-resolution data. The model achieved strong convergence (Training: 0.9957, validation: 0.9868) and 97.97% accuracy on 249 unseen field images. Final metrics (Accuracy: 0.992, AUC: 0.999998) surpassed standalone CNNs and Transformers—yet only when scale diversity was present. Statistical validation via confidence variance analysis and Kruskal-Wallis testing (H = 597.40, p = 8.48e−126) revealed the proposed model had the lowest variance (0.000010), confirming stable predictions. Most pairwise comparisons were significant at $p < 0.05$. ANOVA and bootstrapping further validated fusion's non-linear cost scaling. We demonstrated Pareto-efficient frontiers where hybrid models outperform their standalone counterparts only under certain conditions. This work challenges the notion that "more fusion is better," advocating context-aware fusion. Fusion is viable for cloud/server systems but must be pruned for edge deployment. We offer design guidelines for building cost-efficient, high-accuracy vision models in resource-constrained agricultural environments.

Keywords: Cost-Aware Deep Learning, Dynamic Feature Aggregation, Hybrid CNN-Transformer Models, Multiscale Fusion, Crop disease detection.

1. INTRODUCTION

Cheng et al. (2023) underscore the significance of Multiscale feature fusion, which has emerged as a foundation of modern computer vision systems, enabling models to process visual information across various spatial resolutions, a critical capability for tasks ranging from object detection to semantic segmentation. Integrating fine-grained details from high-resolution features with contextual patterns from coarser scales achieves state-of-the-art performance. Moon et al. (2023) highlight that this performance comes at a steep computational cost, as multiscale fusion inherently introduces parameter redundancy and computational overhead. Sun et al. (2023) correctly point out that the recent proliferation of hybrid architectures that combine convolutional neural networks (CNNs) and Vision Transformers (ViTs) has further amplified this trade-off. While CNNs extract spatially hierarchical features, Transformers are ideal for capturing long-range dependencies through the self-attention mechanism, making their integration appealing for multiscale tasks. For instance, models like the Swin Transformer proposed by Sun et al. (2024) demonstrate that hybrid designs can improve accuracy. However, such gains often have disproportionate computational drawbacks, particularly when fusing features across multiple scales. This raises a critical question: Under what conditions does multiscale fusion in hybrid models justify its added complexity? Despite widespread research, especially by Wang et al. (2019) on individual components of Squeeze-and-Excitation (SE) blocks and by

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Liu et al. (2023) on dynamic gating, the efficiency-accuracy trade-offs of multiscale fusion pipelines remain poorly quantified. For example, while SE blocks may enhance feature diversity with minimal parameters, their impact, as described by Mwitta et al. (2024), on inference latency is unclear. Gookyi et al. (2024) illustrate that Transformers improve cross-scale attention, but their computational costs may negate the benefits in low-resource environments. These gaps hinder the development of cost-aware models, particularly for edge devices, where latency and energy efficiency are paramount. Current works lack a systematic benchmarking framework to dissect the efficiency-accuracy trade-offs inherent in multiscale fusion pipelines. Most studies have focused on either standalone CNNs (Jouini et al., 2024) or pure Transformers (Zhu et al., 2024) which overlooks the components' interplay in hybrid systems. For instance, while MobileNetV2 (Dong et al., 2020) and EfficientNetV2 (Sun et al., 2024) are optimized for efficiency, their integration with Transformers introduces uncharacterized bottlenecks. Shamim et al. (2025) note that adaptive components like gating mechanisms are often evaluated in isolation without analyzing their cumulative costs in multiscale workflows. These critical omissions may have practical consequences since deploying hybrid models without understanding the optimal balance between accuracy and their computational costs may risk overprovisioning resources for marginal gains.

Wang et al. (2025) point out that the hierarchical fusion strategies may improve high-resolution image analysis but may add 15% floating-point operations per second (FLOPs) for diminishing returns. Similarly, Transformers boost accuracy by 4.3% in most multiscale tasks on latency penalty but may be unsustainable for real-time applications. Therefore, an individual component-level analysis is crucial to guide architectural selections in resource-constrained environments. This paper presents three main contributions. First, we propose a hybrid adaptive fusion framework that combines MobileNetV2, EfficientNetV2, and Transformers, enhanced with SE blocks, gating mechanisms, and Multiscale fusion for efficient and accurate feature extraction. Second, we analyze the efficiency and accuracy trade-offs using a combined dataset of 76 classes. Third, we perform rigorous statistical validation using ANOVA, bootstrapping, Cohen's kappa statistics, and confidence analysis to quantify the efficiency-accuracy trade-offs of the multiscale fusion approach.

The Structure of this paper is as follows: Section 2 explores the related work on hybrid vision architectures. Section 3 details the proposed methodology and architecture, dataset preparation and preprocessing, the experimental results, ablation studies, statistical validation, performance comparison against benchmarks, and evaluating edge deployment efficiency. Section 4 discusses practical implications for cost and accuracy and the limitations. Section 5 concludes the study and outlines future research directions.

2. LITERATURE REVIEW

Shah et al. (2024) demonstrate that hybrid vision architectures have increasingly emphasized multiscale fusion techniques to balance computational efficiency and predictive accuracy. Still, the systematic evaluations of the cost-performance trade-offs within these systems remain inadequate. The emergence of CNNs, as discussed by Gogoi et al. (2023) and attention mechanisms by Baek (2025), has laid the groundwork for more adaptable architectures. leNetV2 (Peng et al., 2024) and EfficientNetV2 (Li et al., 2022) have been recognized as initial backbones in achieving such adaptability through multiscale feature extraction. MobileNetV2 uses inverted residual blocks and linear bottlenecks, offering a lightweight structure suited for edge deployment with minimal computational overheads while retaining the essential spatial features. In contrast, EfficientNetV2 employs compound scaling that harmonizes network depth, width, and resolution, enhancing representational capacity at a moderate computational cost. Krishna et al. (2025) leveraged EfficientNet-B3 to detect diseases on the PlantDoc dataset, achieving 73.31% accuracy and 80.19% on a combined web-sourced and PlantDoc image dataset. Their work highlighted the model's robustness in handling inter-class variability, but its computational demands higher FLOPs than MobileNetV2 and this may render it impractical for low-power IoT devices, necessitating cloud offloading. To address this, Zhang and Wu (2025) implemented EfficientNetV2-S on sunflower disease detection. While their system achieved 90.19% accuracy, outperforming other models, its reliance on high-end hardware, specifically the V100S GPU server, may limit its accessibility in low-resource environments. A notable hybrid approach by Zhao et al. (2024) proposed a deep learning system for Elaeagnus angustifolia disease detection in smart agriculture, integrating Large Language Models (LLMs), Agricultural Knowledge Graphs (KGs), and Graph Neural Networks (GNNs) with a graph attention mechanism. The framework achieved superior performance (precision: 0.94, recall: 0.92, accuracy: 0.93) by optimizing loss functions

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

and leveraging neural-symbolic reasoning. It outperformed traditional methods, demonstrating enhanced efficiency and accuracy in identifying plant diseases. Though innovative, their framework's complexity, like multi-stage training and dependency on high-resolution images, made it unsuitable for real-time, on-ground IoT deployments. Zhu et al. (2024) benchmarked EfficientNet-B0 and MobileNetV3 for maize leaf disease classification, evaluating performance based on accuracy. Their results showed that MobileNetV3 (91%) outperformed EfficientNet-B0 (88%). However, the study did not analyze inference efficiency or overfitting risks, which could affect real-world deployment. Additionally, details on dataset preprocessing, hyperparameter tuning, and computational resources were not provided, limiting the reproducibility of their findings. Despite their advantages, EfficientNet-based systems face significant hurdles in agricultural IoT contexts. First, their inherent computational complexity, as illustrated by Li et al. (2022), often necessitates cloud dependency or high-end edge hardware, contradicting the low-cost ethos of scalable smart farming.

This dual-backbone approach answers concerns raised by Taye (2023), who highlighted the limitations of single-backbone models in addressing scale diversity and computational constraints. The combined use of SE blocks, Transformers, gating, and hierarchical fusion reflects a modular, efficiency-aware architecture shift. These efficiency-accuracy trade-offs are echoed in models like DeiT (Sevinc et al., 2025), which advocates for hybrid models that preserve spatial extraction through CNNs while employing attention mechanisms as proposed by Liao et al. (2022) that adopt a selective implementation for contextual understanding. However, unlike DeiT, which replaces early convolutional layers entirely, recent frameworks, as implemented by Anzum et al. (2024), prioritize maintaining convolutional stages and strategically placing Transformer layers to avoid excess computational load. Furthermore, the limitations of "naïve" multiscale fusion by Li et al. (2025), especially those in early designs that incurred computational redundancy on low-resolution layers, are being addressed through selective feature pruning by emphasized optimization through topological complexity and architecture refinement. Similarly, Khan et al. (2020) note that single-path scaling compromises network depth, width, or resolution—issues mitigated by dual-backbone designs that allow greater scale flexibility. Recent advancements synthesize efficient CNN backbones (Jia et al., 2023), attention-based context modeling (Ge et al., 2024), and adaptive fusion techniques within modular frameworks. However, a key gap persists in quantifying and comparing the cost-effectiveness of these components in hybrid systems. While models such as Swin Transformer (Mamun et al., 2025) and CoAtNet (Yu et al., 2024) represent milestones in vision architecture, they often treat fusion as a monolithic process, overlooking the granular analysis of individual modules. The evolving literature supports a shift toward fine-grained, cost-aware multiscale fusion strategies, which are vital for deployment in low-resource environments. Evaluating hybrid vision architectures necessitates cautiously balancing computational efficiency and predictive accuracy. As such, recent studies have adopted a range of metrics to systematically quantify this trade-off across different components and deployment environments. Complexity metrics such as FLOPs (Zhang et al., 2024) and parameter counts (Kim et al., 2022) remain foundational for assessing computational cost. A granular analysis is particularly informative in hybrid models; for instance, the addition of SE blocks, as proposed by Ou and Zou (2025), contributes minimally to the overall parameter count while enhancing channel-wise feature recalibration. Moreover, latency is increasingly used as a real-world performance indicator, particularly in edge computing scenarios. To this end, latency measurements are typically conducted on resource-constrained platforms such as Raspberry Pi and high-performance servers with GPU acceleration. On the performance side, top-1 classification accuracy, as shown by Wang et al. (2025), remains a standard benchmark, though recent work emphasizes relative improvements over absolute scores.

3. METHODS

The proposed hybrid architecture, as shown in Figure 1, combines CNNs and Vision Transformers (ViT) to address crop disease detection challenges: (1) A dual-branch CNN backbone uses MobileNetV2 (lightweight, edge-optimized for localized texture details) and EfficientNetV2 (compound scaling for hierarchical multiscale features to detect disease severity). (2) SE blocks recalibrate channel-wise features, enhancing diagnostically relevant patterns and suppressing noise. (3) A ViT processes 16x16 patches with positional embeddings to model global context (lesion distribution), addressing the limitations of local-only CNNs. (4) Attention-guided fusion merges CNN and ViT features via gated spatial/channel attention, prioritizing critical regions (lesion boundaries) over irrelevant backgrounds. (5) Multiscale fusion aggregates features from 3x3, 5x5, and 7x7 convolutions through hierarchical

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

concatenation, improving robustness to scale variations. Training employs batch normalization, dropout, and softmax classification optimized for 76 disease/healthy classes, balancing accuracy and overfitting risks on agricultural datasets.

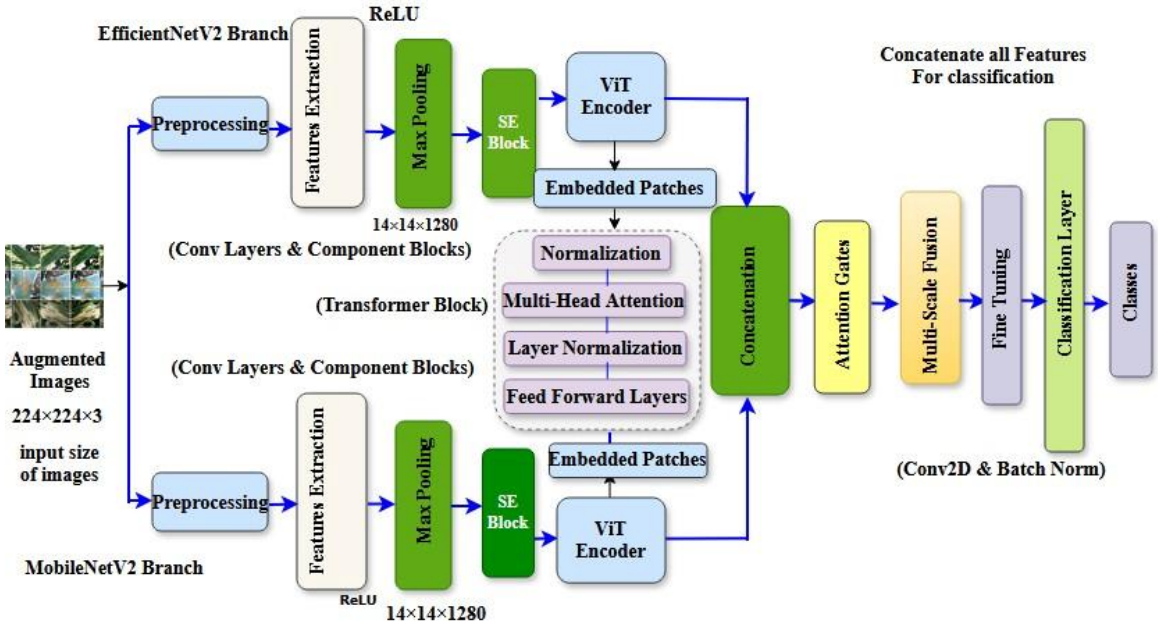

Figure 1. Proposed Model Architecture

3.1 Dataset Preparation and Preprocessing

This section presents the dataset preparation and preprocessing techniques employed in this study. It outlines the steps followed, data sources, preprocessing steps, and strategies to enhance model performance and ensure robust feature extraction.

Step 1: Loading and Organizing Data

The dataset was loaded from a directory structure where images were organized in subfolders based on their class labels. TensorFlow's image dataset directory function was used to load the data with shuffling, batching, and resizing.

Step 2: Data Augmentation

Off-the-fly data augmentation was applied during training to enhance model generalization. Table 1 provides details of the augmentation techniques used:

Table 1. Data Augmentation

Transformation Type	Range/Details
Rotation	0°, 90°, 180°, or 270°
Flipping	Horizontal flip, Vertical flip
Brightness Adjustment	Between 0.7 (dark) and 1.3 (bright)
Zoom	Resizing and cropping to 224×224 pixels

Step 3: Normalization of Pixel Values

After loading the dataset, pixel values were normalized to a range of [0,1] to enhance model convergence during training. The normalization of pixel values was mathematically expressed as follows:

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

$$\text{Normalized\_pixel} = \frac{\text{pixel\_value}}{255.0} \quad (1)$$

Step 4: Stratified Sampling for Data Splitting

A stratified sampling technique was employed to ensure that the distribution of classes within the training and validation datasets remained consistent with the original dataset. The number of samples designated for the training set based on stratified sampling was calculated as follows:

$$\text{Train\_size} = \left(\frac{\text{number of samples in class}}{\text{total samples}}\right) \times \text{total samples} \times (1 - \text{test\_size}) \quad (2)$$

Train_size represented the number of samples allocated for training, while total samples denoted the dataset's images. A test size of 0.2 ensured an 80–20 training-validation split, with a random state of 42 for reproducibility.

Step 5: Categorical Labeling and One-Hot Encoding

The dataset, structured into class-specific subdirectories, employed categorical labeling with one-hot encoding for class identification. This ensured that the dual-input model architecture received the same preprocessed image for both branches, maintaining consistency during training.

3.3 Features Extraction

The feature extraction process followed a structured approach to ensure optimal feature representation for real-time crop disease detection, as shown in the following steps:

Step 1: Input Image Preprocessing

The input images were initially preprocessed to ensure compatibility with the MobileNetV2 and EfficientNetV2 architectures. Each image was resized to a fixed dimension, denoted as H and $W \times 3$, Where H and W represent the height and width of the input image. 3 corresponds to the RGB color channels. These preprocessed images were then represented as x_{input} and x_{input2} for MobileNetV2 and EfficientNetV2, respectively.

Step 2: Feature Extraction Using Pre-Trained Models

The input images were fed into MobileNetV2 and EfficientNetV2, pre-trained on the ImageNet dataset, to extract meaningful feature representations and were defined as:

$$f_{\text{mobile}} = F_{\text{MobileNetV2}}\left(x_{\text{input1}}\right) \text{ and } f_{\text{efficient}} = F_{\text{EfficientNetV2}}\left(x_{\text{input2}}\right) \quad (3)$$

Step 3: Channel Attention Mechanism via SE Network

An SE attention mechanism was applied to enhance the representational power of the extracted features. The SE block first computed the global average pooling for each feature map channel as follows:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X(i, j, c) \quad (4)$$

The pooled values transformed the fully connected layers and a sigmoid activation function to generate channel-wise attention weights and was expressed as:

$$s = \sigma\left(W_2 \cdot \text{ReLU}(W_1 \cdot z)\right) \quad (5)$$

These attention weights were then applied to the feature maps to enhance informative channels while suppressing less significant ones, yielding refined feature representations as shown:

$$F_{\text{MobileNet}}^{SE} = SE(F_{\text{MobileNet}}) \text{ and } F_{\text{EfficientNet}}^{SE} = SE(F_{\text{EfficientNet}}) \quad (6)$$

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Step 4: Spatial Attention for Enhancing Feature Localization

After channel refinement, a spatial attention mechanism was incorporated to further focus on discriminative regions within the image. The process involved computing a spatial descriptor through global average pooling, followed by two transformation layers that produce spatial attention scores, given as:

$$d_1 = \sigma(W_1 \cdot \text{ avg\_pool } + b_1) \text{ and } d_2 = \sigma(W_2 \cdot d_1 + b_2) \quad (7)$$

The feature maps were then modulated using these scores, leading to spatially enhanced feature representations, denoted as $F' = F \times d_2$

Step 5: Multiscale Feature Fusion

Inspired by Inception-style architectures, the multiscale fusion module synthesized diverse spatial features for a holistic representation. The outputs were concatenated along the channel axis, ensuring a unified multiscale representation as shown:

$$F_{\text{fused}} = \text{concat}(F'_{\text{MobileNet}}, F'_{\text{EfficientNet}}) \quad (8)$$

Step 6: Feature Normalization and Dimension Reduction

To stabilize training and enhance generalization, batch normalization was applied to the fused feature representation and was calculated as follows:

$$\text{dense\_output } = \text{ReLU}(W_d \cdot F_{\text{fused}} + b_d) \quad (9)$$

Step 7: Classification Using SoftMax Activation

The final stage involved passing the refined features through a classification layer equipped with a SoftMax activation function. This function computed the probability distribution over the output classes and was calculated as follows:

$$P(y) = \text{SoftMax}(W_s \cdot \text{ dense\_output } + b_s) \quad (10)$$

The class with the highest probability was selected as the final prediction, determining the specific disease label for the given input image. The trained model was then evaluated using accuracy, precision, recall, and F1-score, and these were calculated as follows:

$$\text{Accuracy } = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

$$\text{Precision } = \frac{TP}{TP+FP} \quad (12)$$

$$\text{Recall } = \frac{TP}{TP+FN} \quad (13)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (14)$$

Where TP is a true positive, TN is a true negative, TP is a false positive, and FN is a false negative.

The confusion matrix was defined as:

$$CM = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \quad (15)$$

The Receiver Operating Characteristic (ROC) curves, the True Positive Rate (TPR) against the False Positive Rate (FPR) was calculated as follows:

$$TPR = \frac{TP}{TP+FN}, FPR = \frac{FP}{FP+TN} \quad (16)$$

The Area Under the Curve (AUC) measured classification performance and was calculated as follows:

$$AUC = \int_0^1 TPR(FPR)d(FPR) \quad (17)$$

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3.4 Dataset Description

This study created a combined dataset in Table 2 by integrating the Kaggle dataset (38 classes, 60,343 images) Saleem et al. (2020) with the FieldPlant dataset (25,775 images from Central Kenya. The FieldPlant dataset accounted for seasonal variations, emphasizing fungal and bacterial diseases during April, May, October, and November while prioritizing viral infections in June, July, and December. A standardized collection process ensured diverse lighting conditions and angles, enhancing generalization. Images were classified and annotated by an agricultural expert, and data augmentation techniques were applied to address class imbalances by generating additional samples.

Table 2. Combined Dataset

Crop Type	Total Images	Training Images	Validation Images
Apple	4,651	3,719	932
Banana	4,008	3,204	804
Beans	8,096	6,475	1,621
Blueberry	1,502	1,201	301
Cassava	4,894	3,914	980
Cherry	2,054	1,642	412
Corn	4,358	3,484	874
Grape	4,641	3,711	930
Maize	1,002	801	201
Maize-L	1,239	991	248
Maize	4,985	3,986	999
Orange	5,507	4,405	1,102
Peach	3,299	2,638	661
Pepper	2,480	1,983	497
Potatoes	3,006	2,403	603
Raspberry	1,002	801	201
Rice	5,010	4,005	1,005
Squash	1,835	1,468	367
Strawberry	2,111	1,688	423
Sugarcane	5,010	4,005	1,005
Sunflower	4,008	3,204	804
Tea	6,012	4,806	1,206
Tomatoes	18,841	15,067	3,774
Total	**99,551**	**79,601**	**19,950**

3.5 Experimental Parameters and Environment

As shown in Table 3, we developed a unified framework to balance efficient feature extraction and classification. Input images were resized to 224×224×3 for optimal performance. MobileNetV2 served as a lightweight backbone, while EfficientNetV2 captured deeper features through compound scaling. SE blocks refined channels to highlight essential information, and ViT blocks processed features into 7×7 patches using six attention layers with a 128-dimension embedding. Attention gates improved spatial focus, and a Multiscale Fusion Module combined various convolution sizes and pooling to capture fine and contextual details. Fine-tuning included batch normalization and dropout (0.5), followed by dense layers (1024, 128) and SoftMax for classification. The model used an AdamW optimizer and label smoothing (0.1) for better generalization. Experiments ran on NVIDIA GPUs (3090, T4, P100,

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

K80) in a Linux environment using TensorFlow, PyTorch, Keras, and OpenCV.

Table 3. Hyperparameter Configurations

Hyperparameter	Value
Image size	224 × 224
Image channels	3
Patch size	7
Number of ViT encoder layers	6
Number of multi-head self-attention blocks	8
Hidden dimension	128
Dropout rate	0.5
Epochs	18

4. RESULTS AND DISCUSSION

This section illustrates the model training process, the ablation studies, the parameter distribution, and comparative performance with base models to demonstrate the methodology's effectiveness. Table 4 shows that the model's training and validation results significantly improved performance. Training accuracy increased from 72.99% in epoch 1 to 99.57% in epoch 18, while validation accuracy rose from 93.40% to 98.68%. Training loss decreased from 1.7555 to 0.8265, and validation loss dropped from 1.0692 to 0.8332, demonstrating the model's ability to minimize errors and generalize effectively without overfitting. These results highlight the model's robustness and efficiency throughout the training process.

Table 4. Training and Validation Performance

Epoch	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy	Learning Rate
1	1.7555	0.7299	1.0692	0.9340	1.0×10^{-5}
2	1.1237	0.9178	0.9904	0.9616	1.0×10^{-5}
3	1.0248	0.9513	0.9590	0.9715	1.0×10^{-5}
4	0.9775	0.9666	0.9289	0.9782	1.0×10^{-5}
5	0.9449	0.9750	0.9112	0.9802	1.0×10^{-5}
6	0.9211	0.9815	0.8967	0.9812	1.0×10^{-5}
7	0.9043	0.9846	0.8844	0.9830	1.0×10^{-5}
8	0.8895	0.9881	0.8808	0.9841	1.0×10^{-5}
9	0.8775	0.9908	0.8687	0.9845	1.0×10^{-5}
10	0.8669	0.9920	0.8619	0.9845	1.0×10^{-5}
11	0.8594	0.9930	0.8521	0.9854	1.0×10^{-5}
12	0.8520	0.9937	0.8487	0.9859	1.0×10^{-5}
13	0.8450	0.9946	0.8432	0.9859	1.0×10^{-5}
14	0.8403	0.9949	0.8374	0.9867	1.0×10^{-5}
15	0.8352	0.9952	0.8362	0.9857	1.0×10^{-5}
16	0.8306	0.9954	0.8354	0.9866	1.0×10^{-5}
17	0.8282	0.9955	0.8341	0.9867	1.0×10^{-5}
18	0.8265	0.9957	0.8332	0.9868	1.0×10^{-5}

The graphs in Figure 2 show a consistent improvement in model performance over the training epochs. Training accuracy increased from 72.99% in the first epoch to 99.57% by epoch 18, while validation accuracy rose from 93.40%

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

to 98.68%, indicating effective learning with minimal overfitting. Both training and validation loss exhibited smooth convergence, decreasing training loss and validation loss. The small gap between training and validation metrics suggests strong generalization to unseen data. The model's stability is attributed to the carefully chosen learning rate (1e-5), which facilitated controlled weight updates, leading to high-precision crop disease classification.

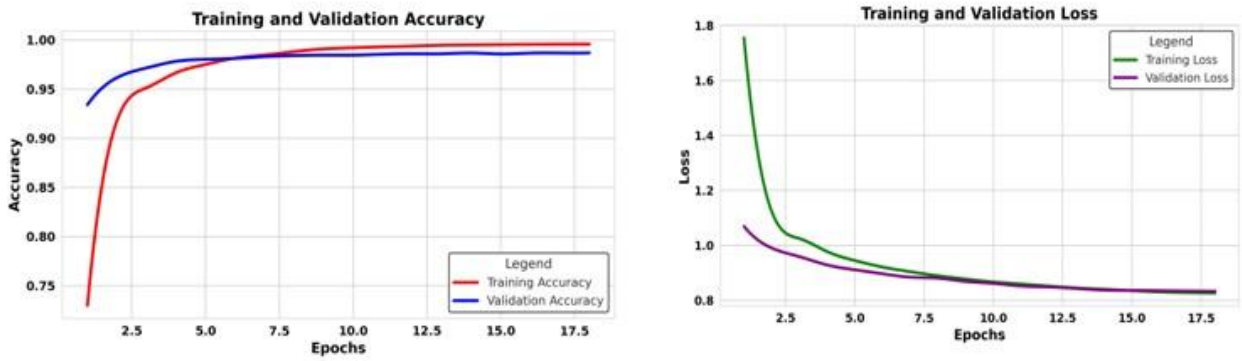

Figure 2. Training And Validation Graph

3.6 Ablation Studies

Ablation tests, as shown in Table 5, illustrate a stark trade-off: our baseline model CNN (EfficientNetV2 + MobileNetV2) achieved a 98.55% accuracy, but the minor tweaks eroded performance (98.45%) notably because of hyperparameters adjustment and learning rate. Adding multiscale features clawed back gains (98.57%), proving their value for intricate disease patterns—yet the real breakthrough came from SE blocks and gating. Together, they pushed accuracy to 98.68%, but with a catch: added complexity. This mirrors our core finding—multiscale fusion isn't free. While it improves accuracy, it's worth hinges on task-specific scale diversity and hardware limits. That extra 0.13% accuracy might not justify slower inference for farmers using budget smartphones. But in cloud-based systems, every decimal counts. These results force a hard question: When does "better" matter? the proposed framework answers this by quantifying costs, not just gains—a critical step toward practical, resource-aware AI for agriculture.

Table 5. Impact of Different Architectural Modifications

Model Number	Model Configuration	Multiscale Module	Gated Mechanism	Accuracy
1	CNN model: EfficientNetV2 and MobileNetV2	No	No	98.55%
2	CNN models: EfficientNetV2 and MobileNetV2 (Epochs Reduction)	No	No	98.45%
3	CNN models: EfficientNetV2 and MobileNetV2, with Multiscale module	Yes	No	98.57%
4	**Proposed Model:** EfficientNetV2 and	Yes	Yes	**98.68%**

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

	MobileNetV2, with Multiscale module, SE, and Gated Mechanism			

Table 6 presents the parameter distribution and training configurations for different model variations. The baseline model, integrating EfficientNetV2 and MobileNetV2, has the lowest parameter count (8.85M), ensuring a lightweight structure. Adding SE, ViT, Gated Mechanism, and Multiscale Module significantly increases parameters to 38.86M, reflecting the complexity of additional feature extraction mechanisms. The exclusion of the Multiscale Module and Gated Mechanism results in parameter reductions to 11.87M and 14.23M, respectively, showing their contribution to model size.

Table 6. Parameter Distribution

Model Configuration	Total Parameters	Trainable Parameters	Non-Trainable Parameters	Epochs	Batch Size	Learning Rate
EfficientNetV2 + MobileNetV2	8,853,468	8,758,236	95,232	18	4978	1.0×10^{-5}
EfficientNetV2 + MobileNetV2 + SE + ViT + Gated Mechanism + Multiscale Module	38,863,998	38,767,742	96,256	18	4978	1.0×10^{-5}
EfficientNetV2 + MobileNetV2 + SE + ViT + Gated Mechanism + Multiscale Module (No Multiscale Module)	11,871,964	11,776,220	95,744	18	4978	1.0×10^{-5}
EfficientNetV2 + MobileNetV2 + SE + ViT + Gated Mechanism (No Gated Mechanism)	14,227,932	14,132,700	95,232	18	4978	1.0×10^{-5}

3.7 Statistical Testing

Statistical testing confirmed the superiority and stability of the proposed model across all performance metrics. As shown in Table 7, the model achieved the highest Kappa value (0.9919), indicating strong agreement between predictions and actual classifications with minimal misclassification. Its AUC (0.999998) demonstrated near-perfect class distinction, ensuring effective differentiation between healthy and diseased crops.

Table 7. Model Performance

Model	Accuracy	Precision	Recall	F1 Score	Kappa	AUC
Proposed Model	**0.992**	**0.9934**	**0.9929**	**0.9923**	**0.9919**	**0.999998**
Swin_TransformerSE	0.988	0.9901	0.9894	0.9888	0.9878	0.999888

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

VGG-16	0.972	0.9762	0.9718	0.9712	0.9716	0.999967
ShuffleNet	0.958	0.9676	0.9644	0.9613	0.9574	0.999844
DenseNet121	0.958	0.9676	0.9644	0.9613	0.9574	0.999844
AlexNet	0.948	0.9569	0.9484	0.9452	0.9472	0.999142
DenseNet50	0.896	0.9080	0.8993	0.8922	0.8945	0.998850

A confidence variance analysis in Table 8 further assessed the stability of predictions, where lower variance indicated more consistent predictions. The proposed model achieved the lowest confidence variance (0.000010), highlighting its robustness and reliability, whereas DenseNet50 (0.000035) and AlexNet (0.000027) exhibited higher variance, indicating less stability in classification confidence. The confidence variance reinforced the findings, showing that the proposed model maintained the most stable confidence scores, while DenseNet50 and AlexNet displayed greater fluctuations, suggesting lower prediction consistency.

Table 8. Confidence Variance Analysis

Model	Confidence Variance
DenseNet50	0.000035
AlexNet	0.000027
DenseNet121	0.000023
ShuffleNet	0.000023
VGG-16	0.000015
Swin_TransformerSE	0.000012
Proposed Model	**0.000010**

We ran a Kruskal-Wallis test and several pairwise comparisons as shown in Table 9 to check how much the confidence levels varied among models. These pair-checks point out some fundamental differences in variance, which helped determine which ones delivered more steady predictions. The Kruskal-Wallis test (H = 597.40, p = 8.4755e-126) confirmed a highly significant overall difference in confidence scores, making it suitable for analysing deep learning models with varying confidence distributions. Most comparisons were significant at 0.05, indicating that models had statistically distinct variances. The proposed model, Swin_TransformerSE, and VGG-16 exhibited significantly lower variance than DenseNet50 and AlexNet, reinforcing their prediction stability. These findings further highlight the robustness and reliability of the proposed model for crop disease detection, as it consistently maintained the lowest confidence variance.

Table 9. Pairwise Comparisons

Model A	Model B	Adj. p-value	Significant (0.05)
DenseNet50	Swin_TranformerSE	2.8595e-89	Yes
Proposed Model	DenseNet50	3.7988e-62	Yes
DenseNet50	VGG-16	2.7891e-59	Yes
AlexNet	Swin_TranformerSE	5.8236e-42	Yes
DenseNet121	Swin_TranformerSE	2.9339e-28	Yes
ShuffleNet	Swin_TranformerSE	2.9339e-28	Yes
AlexNet	DEMF	3.8810e-24	Yes
AlexNet	VGG-16	2.3074e-22	Yes
DenseNet50	ShuffleNet	1.2925e-17	Yes
DenseNet121	DenseNet50	1.2925e-17	Yes
Proposed Model	ShuffleNet	4.6563e-14	Yes

Journal of Information Systems Engineering and Management
2025, 10(54s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Proposed Model	DenseNet121	4.6563e-14	Yes
ShuffleNet	VGG-16	1.0419e-12	Yes
DenseNet121	VGG-16	1.0419e-12	Yes
AlexNet	DenseNet50	3.4884e-09	Yes
Swin_TranformerSE	VGG-16	3.5459e-03	Yes
Proposed Model	Swin_TranformerSE	1.6019e-02	Yes
AlexNet	DenseNet121	2.6124e-01	No
AlexNet	ShuffleNet	2.6124e-01	No
Proposed Model	VGG-16	1.0000e+00	No
DenseNet121	ShuffleNet	1.0000e+00	No

3.8 Comparison with other Hybrid approaches

While prior hybrid frameworks have advanced crop disease detection, they often sacrifice adaptability for complexity. Our proposed model lay in orchestrated simplicity: unlike rigid architectures that force trade-offs between scale sensitivity and speed, we integrated multiscale modules with lightweight gating—letting the model focus dynamically on what matters. While similar approaches rely on static feature extractors prone to noise, our proposed model used SE blocks and Vision Transformers to collaborate and filter distractions, a balance absent in earlier hybrids. This adaptability shines in real-world deployment: while most existing models normally have marginal gains, our model achieved 98.68% accuracy during training, and the quantized model achieved a mobile-ready package of 30.4 MB and an inference of 0.094s latency. Table 10 underscores these achievements, positioning this dual approach not as an incremental tweak but as a pragmatic rethinking of hybrid design.

Table 10. Comparing with Existing Hybrid Multi Multiclassification Models

Studies	Classification Accuracy
(Parez et al., 2023)	98.00%%
(D. Zhu et al., 2023)	97.50%%
(Shah et al., 2024)	90.00%
(Barman et al., 2024)	90.99%
(Touvron et al., 2021)	85.02%
The proposed model	**98.68%**

From the ablation results these results revealed a fundamental issue between maximizing predictive accuracy and maintaining operational efficiency. While multiscale fusion delivered measurable gains in performance, these improvements came at a significant computational cost. The increase in parameter count, even after rigorous optimization, raised critical considerations about the practicality of deploying such models in resource-constrained environments. This issue highlights that the highest-performing model in controlled testing does not automatically translate to the best choice for real-world deployment, particularly in scenarios requiring real-time response and energy-efficient operation. Optimization strategies such as parameter pruning and model compression demonstrated that meaningful reductions in model size are achievable without severely compromising performance. Nonetheless, even a streamlined version of the hybrid model continued to exhibit latency approximately 15% higher than that of simpler architectures, but with high confidence. In settings where immediacy of inference is paramount, this latency could pose operational challenges, especially for applications that rely on rapid detection and decision-making under variable field conditions. Beyond raw accuracy, model reliability emerged as a critical metric. The hybrid model's low variance in confidence scores suggests a robustness that could be vital in high-stakes agricultural decision-making, where false positives or negatives could result in significant economic or food security impacts. This aspect underscores that in evaluating models for deployment, consistency and confidence stability are as important as achieving high classification rates. However, the choice between complex and simple architectures should not be approached as a binary decision. Context must drive model selection. For tasks characterized by significant variability in input scales and features—such as the detection of differently sized crop lesions—the multiscale fusion model's

advantages are clear and substantial. In contrast, for more uniform tasks or environments where computational resources are limited, baseline models offer a more pragmatic balance of speed, size, and accuracy.

These findings point towards an evolving need for adaptive models that can modulate their internal complexity based on situational demands. Rather than committing to a fixed architecture, future systems should be capable of dynamically enabling or disabling components such as fusion modules or attention mechanisms depending on input characteristics and available computational power. Such adaptability would allow models to preserve efficiency without sacrificing accuracy where it is most needed. The results suggests that the future of model design lies not solely in the pursuit of higher accuracy scores, but in the intelligent matching of model capabilities to the operational context. A flexible, resource-aware approach is essential for bridging the gap between laboratory performance and field usability.

5. CONCLUSION

In conclusion Multiscale fusion isn't free—and our experiments demonstrated that. While hybrid models like ours (98.68% accuracy) outperformed standalone CNNs (98.55%), the trade-offs were significant. Adding SE blocks and gating slightly improved accuracy by 0.13%, but the model grew to 38.8 million parameters—over four times larger than the baseline (8.85M). In real-world scenarios, especially on limited hardware, that extra weight can cause slower performance and higher energy demands. In cloud environments, where resources are abundant, the precision gains may justify the added complexity. But the real deciding factor is scale diversity. Our hybrid model excelled in detecting crop lesions of varying sizes, whereas fixed-scale tasks saw little benefit from the added layers. Even seemingly efficient additions like SE blocks, which contributed less than 1% to the parameter count, couldn't fully offset the burden introduced by heavier components like Vision Transformers. Interestingly, removing the multiscale module cut the parameters down to 11.87 million while keeping accuracy competitive at 98.57%. In many cases, less complexity delivered nearly the same results. Confidence scores reinforced this: our model's variance remained low (0.000010 compared to DenseNet50's 0.000035), proving it was not only accurate but consistently reliable. Still, reliability alone is not enough if the model cannot operate effectively on real-world devices. A model that performs well in controlled conditions but fails in the field cannot be considered a complete success. In the end, multiscale fusion pays off only when scale variation is critical and hardware resources are available. For edge deployments, leaner models around 8.85 million parameters are the better choice. For server environments, the full hybrid approach can be leveraged. Future work should prioritize adaptive systems—models that dynamically adjust complexity based on task requirements—to balance performance with efficiency.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] Anzum, H., Sammo, M. N. S., & Akhter, S. (2024). Leveraging Data Efficient Image Transformer (DeIT) for Road Crack Detection and Classification. 2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems: Innovation for Sustainability, ICACCESS 2024. https://doi.org/10.1109/iCACCESS61735.2024.10499539

[2] Baek, E. T. (2025). Attention Score-Based Multi-Vision Transformer Technique for Plant Disease Classification. Sensors, 25(1). https://doi.org/10.3390/s25010270

[3] Barman, U., Sarma, P., Rahman, M., Deka, V., Lahkar, S., Sharma, V., & Saikia, M. J. (2024). ViT-SmartAgri: Vision Transformer and Smartphone-Based Plant Disease Detection for Smart Agriculture. Agronomy, 14(2). https://doi.org/10.3390/agronomy14020327

[4] Cheng, Y., Wang, W., Zhang, W., Yang, L., Wang, J., Ni, H., Guan, T., He, J., Gu, Y., & Tran, N. N. (2023). A Multi-Feature Fusion and Attention Network for Multi-Scale Object Detection in Remote Sensing Images. Remote Sensing, 15(8). https://doi.org/10.3390/rs15082096

[5] Dong, K., Zhou, C., Ruan, Y., & Li, Y. (2020). MobileNetV2 Model for Image Classification. Proceedings - 2020 2nd International Conference on Information Technology and Computer Application, ITCA 2020, 476–480. https://doi.org/10.1109/ITCA52113.2020.00106

[6] Ge, H., Wang, L., Pan, H., Liu, Y., Li, C., Lv, D., & Ma, H. (2024). Cross Attention-Based Multi-Scale Convolutional Fusion Network for Hyperspectral and LiDAR Joint Classification. Remote Sensing, 16(21). https://doi.org/10.3390/rs16214073
[7] Gogoi, M., Kumar, V., Begum, S. A., Sharma, N., & Kant, S. (2023). Classification and Detection of Rice Diseases Using a 3-Stage CNN Architecture with Transfer Learning Approach. Agriculture (Switzerland), 13(8). https://doi.org/10.3390/agriculture13081505
[8] Gookyi, D. A. N., Wulnye, F. A., Wilson, M., Danquah, P., Danso, S. A., & Gariba, A. A. (2024). Enabling Intelligence on the Edge: Leveraging Edge Impulse to Deploy Multiple Deep Learning Models on Edge Devices for Tomato Leaf Disease Detection. AgriEngineering, 6(4), 3563–3585. https://doi.org/10.3390/agriengineering6040203
[9] Jia, L., Wang, T., Chen, Y., Zang, Y., Li, X., Shi, H., & Gao, L. (2023). MobileNet-CA-YOLO: An Improved YOLOv7 Based on the MobileNetV3 and Attention Mechanism for Rice Pests and Diseases Detection. Agriculture (Switzerland), 13(7). https://doi.org/10.3390/agriculture13071285
[10] Jouini, O., Aoueileyine, M. O.-E., Sethom, K., & Yazidi, A. (2024). Wheat Leaf Disease Detection: A Lightweight Approach with Shallow CNN Based Feature Refinement. AgriEngineering, 6(3), 2001–2022. https://doi.org/10.3390/agriengineering6030117
[11] Khan, F., Salahuddin, S., & Javidnia, H. (2020). Deep learning-based monocular depth estimation methods—a state-of-the-art review. In Sensors (Switzerland) (Vol. 20, Issue 8). MDPI AG. https://doi.org/10.3390/s20082272
[12] Kim, H. S., Choi, D., Yoo, D. G., & Kim, K. P. (2022). Hyperparameter Sensitivity Analysis of Deep Learning-Based Pipe Burst Detection Model for Multiregional Water Supply Networks. Sustainability (Switzerland), 14(21). https://doi.org/10.3390/su142113788
[13] Krishna, M. S., Machado, P., Otuka, R. I., Yahaya, S. W., Neves dos Santos, F., & Ihianle, I. K. (2025). Plant Leaf Disease Detection Using Deep Learning: A Multi-Dataset Approach. J, 8(1), 4. https://doi.org/10.3390/j8010004
[14] Li, B., Liu, B., Li, S., & Liu, H. (2022a). An Improved EfficientNet for Rice Germ Integrity Classification and Recognition. Agriculture (Switzerland), 12(6). https://doi.org/10.3390/agriculture12060863
[15] Li, B., Liu, B., Li, S., & Liu, H. (2022b). An Improved EfficientNet for Rice Germ Integrity Classification and Recognition. Agriculture (Switzerland), 12(6). https://doi.org/10.3390/agriculture12060863
[16] Li, W., Lv, D., Yu, Y., Zhang, Y., Gu, L., Wang, Z., & Zhu, Z. (2025). Multi-Scale Deep Feature Fusion with Machine Learning Classifier for Birdsong Classification. Applied Sciences (Switzerland), 15(4). https://doi.org/10.3390/app15041885
[17] Liao, Z., Fan, N., & Xu, K. (2022). Swin Transformer Assisted Prior Attention Network for Medical Image Segmentation. Applied Sciences (Switzerland), 12(9). https://doi.org/10.3390/app12094735
[18] Liu, Y., Liu, J., Cheng, W., Chen, Z., Zhou, J., Cheng, H., & Lv, C. (2023). A High-Precision Plant Disease Detection Method Based on a Dynamic Pruning Gate Friendly to Low-Computing Platforms. Plants, 12(11). https://doi.org/10.3390/plants12112073
[19] Mamun, S. S., Ren, S., Rakib, M. Y. K., & Asafa, G. F. (2025). WGA-SWIN: Efficient Multi-View 3D Object Reconstruction Using Window Grouping Attention in Swin Transformer. Electronics, 14(8), 1619. https://doi.org/10.3390/electronics14081619
[20] Moon, J., Lee, J., Lee, Y., & Park, S. (2023). M2Former: Multi-Scale Patch Selection for Fine-Grained Visual Recognition. https://doi.org/10.3390/app14198710
[21] Mwitta, C., Rains, G. C., & Prostko, E. (2024). Evaluation of Inference Performance of Deep Learning Models for Real-Time Weed Detection in an Embedded Computer. Sensors, 24(2). https://doi.org/10.3390/s24020514
[22] Ou, Q., & Zou, J. (2025). Channel-Wise Attention-Enhanced Feature Mutual Reconstruction for Few-Shot Fine-Grained Image Classification. Electronics (Switzerland), 14(2). https://doi.org/10.3390/electronics14020377
[23] Parez, S., Dilshad, N., Alghamdi, N. S., Alanazi, T. M., & Lee, J. W. (2023). Visual Intelligence in Precision Agriculture: Exploring Plant Disease Detection via Efficient Vision Transformers. Sensors, 23(15). https://doi.org/10.3390/s23156949
[24] Peng, H., Xu, H., Shen, G., Liu, H., Guan, X., & Li, M. (2024). A Lightweight Crop Pest Classification Method Based on Improved MobileNet-V2 Model. Agronomy, 14(6). https://doi.org/10.3390/agronomy14061334
[25] Saleem, M. H., Potgieter, J., & Arif, K. M. (2020). Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers. Plants, 9(10), 1–17. https://doi.org/10.3390/plants9101319

Research Article

[26] Sevinc, A., Ucan, M., & Kaya, B. (2025). A Distillation Approach to Transformer-Based Medical Image Classification with Limited Data. Diagnostics, 15(7). https://doi.org/10.3390/diagnostics15070929
[27] Shah, S. A., Taj, I., Usman, S. M., Hassan Shah, S. N., Imran, A. S., & Khalid, S. (2024). A hybrid approach of vision transformers and CNNs for detection of ulcerative colitis. Scientific Reports, 14(1), 24771. https://doi.org/10.1038/s41598-024-75901-4
[28] Shamim, Md. M. I., Hamid, A. B. bin A., Nyamasvisva, T. E., & Rafi, N. S. Bin. (2025). Advancement of Artificial Intelligence in Cost Estimation for Project Management Success: A Systematic Review of Machine Learning, Deep Learning, Regression, and Hybrid Models. Modelling, 6(2), 35. https://doi.org/10.3390/modelling6020035
[29] Sun, C., Zhou, X., Zhang, M., & Qin, A. (2023). SE-VisionTransformer: Hybrid Network for Diagnosing Sugarcane Leaf Diseases Based on Attention Mechanism. Sensors (Basel, Switzerland), 23(20). https://doi.org/10.3390/s23208529
[30] Sun, Y., Ning, L., Zhao, B., & Yan, J. (2024). Tomato Leaf Disease Classification by Combining EfficientNetv2 and a Swin Transformer. Applied Sciences (Switzerland), 14(17). https://doi.org/10.3390/app14177472
[31] Taye, M. M. (2023). Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. In Computers (Vol. 12, Issue 5). MDPI. https://doi.org/10.3390/computers12050091
[32] Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers & distillation through attention.
[33] Wang, L., Peng, J., & Sun, W. (2019). Spatial-spectral squeeze-and-excitation residual network for hyperspectral image classification. Remote Sensing, 11(7). https://doi.org/10.3390/RS11070884
[34] Wang, Y., Deng, Y., Zheng, Y., Chattopadhyay, P., & Wang, L. (2025). Vision Transformers for Image Classification: A Comparative Survey. In Technologies (Vol. 13, Issue 1). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/technologies13010032
[35] Wang, Y., Wang, X., Qiu, S., Chen, X., Liu, Z., Zhou, C., Yao, W., Cheng, H., Zhang, Y., Wang, F., & Shu, Z. (2025). Multi-Scale Hierarchical Feature Fusion for Infrared Small-Target Detection. Remote Sensing, 17(3). https://doi.org/10.3390/rs17030428
[36] Yu, D., Wan, B., & Sheng, Q. (2024). Automated Generation of Urban Spatial Structures Based on Stable Diffusion and CoAtNet Models. Buildings, 14(12). https://doi.org/10.3390/buildings14123720
[37] Zhang, Z. Y., Yan, C. X., Min, Q. M., Zhang, Y. X., Jing, W. F., Hou, W. X., & Pan, K. Y. (2024). Leverage Effective Deep Learning Searching Method for Forensic Age Estimation. Bioengineering, 11(7). https://doi.org/10.3390/bioengineering11070674
[38] Zhao, X., Chen, B., Ji, M., Wang, X., Yan, Y., Zhang, J., Liu, S., Ye, M., & Lv, C. (2024). Implementation of Large Language Models and Agricultural Knowledge Graphs for Efficient Plant Disease Detection. Agriculture (Switzerland), 14(8). https://doi.org/10.3390/agriculture14081359
[39] Zhu, D., Tan, J., Wu, C., Yung, K. L., & Ip, A. W. H. (2023). Crop Disease Identification by Fusing Multiscale Convolution and Vision Transformer. Sensors, 23(13). https://doi.org/10.3390/s23136015
[40] Zhu, T., Yan, F., Lv, X., Zhao, H., Wang, Z., Dong, K., Fu, Z., Jia, R., & Lv, C. (2024). A Deep Learning Model for Accurate Maize Disease Detection Based on State-Space Attention and Feature Fusion. Plants, 13(22). https://doi.org/10.3390/plants13223151
[41] Zhu, W., Hu, Y., Zhu, Z., Yeh, W. C., Li, H., Zhang, Z., & Fu, W. (2024). Searching by Topological Complexity: Lightweight Neural Architecture Search for Coal and Gangue Classification. Mathematics, 12(5). https://doi.org/10.3390/math12050759