International Journal of Advanced Trends in Engineering and Management
(ISSN 2583-7052)

Vol. V, No. 03, pp. 1-8, March

A Comparative Study of CNN-Based Deep Learning Architectures for Tomato Leaf Disease Classification

Kanamarlapudi Naga Madhuri[1], A. S. R. Prasanth[2]

[1,2] Department of Computer Science and Engineering,
GVR&S College of Engineering and Technology, Affiliated to JNTUK, Kakinada,
Budampadu-522017, Guntur (Dt), A.P., India

madhurithanniru29@gmail.com

ABSTRACT: Foliar diseases are a continuous threat to the production of tomato, causing annual losses of 20 to 40% to the crop. The identification of plant diseases from images using automated methods has gained much importance in precision agriculture. We compared six deep-learning architectures: Basic CNN, InceptionV3, DenseNet121, MobileNetV2, VGG-16, and our proposed model Hybrid DenseNet-MobileNet for tomato leaf disease classification between five diseases and one healthy class. The experiments were performed on the dataset of 16,118 images (13,507 images for training and 2,611 for testing) with the classes including Blight Leaf, Damage Spot Leaf, Healthy Leaf, Rot Leaf and Virus Leaf. All the pre-trained architectures are used by transfer learning using ImageNet weights, by unfreezing the layers step by step, and applying data augmentations. On the single test set, DenseNet121 achieves the highest accuracy of 89.01%; however, the proposed two stream feature fusion Hybrid DenseNet-MobileNet outperforms it, achieving 91.23% accuracy. The basic CNN achieves 72.30% and VGG-16 achieves 68.00%, confirming that shallow and oversized neural networks are insufficient for fine-grained agricultural texture classification tasks.

Keywords: Tomato leaf disease, Convolutional Neural Network, Transfer learning, DenseNet121, MobileNetV2, InceptionV3, VGG-16, Hybrid model, Deep learning, Precision agriculture.

1. Introduction

The foundation of economic activity in developing countries is agriculture and healthy crops are one of the main requirements to guarantee food security and rural life. Tomato (Solanum lycopersicum) is one of the most cultivated vegetable crops globally, which makes a significant contribution to the nutritional diet of the Asian, African and American regions.

Globally, tomatoes are produced more than 180 million metric tonnes annually and therefore a crop of very great economic and humanitarian importance in both developed and developing countries. Tomatoes are still under continuous threat of numerous varieties of foliage-based diseases such as early and late blight, leaf spot, mosaic virus and many varieties of rotting diseases. Climate variability, increased humidity and the high density of planting practices used in commercial agriculture further promote the rapid spread of these pathogens. A combination of these plant diseases leads to a loss of yield of between 20%–40% annually when unchecked and untreated [1].

The old methods entail physical inspection by professional agronomists who carry out visual assessment of the leaves that have been infected. This is not only resource intensive but also costly and has a high human perception variability. In addition, the fact that there is a worldwide lack of qualified agricultural experts especially in remote and rural agricultural societies prevents the ability to scale up and roll out such traditional diagnostic procedures in time. The possibility to identify plant pathogens with the help of digital leaf images and automatically with the latest developments in machine vision algorithms, namely, Convolutional Neural Networks (CNNs), is highly realistic. Deep learning algorithms have been shown to excel at hierarchical visual pattern analysis based on complex textural patterns, and hence suitable for such tasks [2]. The growth of access to smartphone cameras and edge computing devices into the agricultural environment has also provided feasible avenues to directly implement such intelligent diagnostic systems in the field.

The three essential questions that we will address with the help of this research work are as follows: (1) how far can the pretrained deep learning algorithms be adjusted to work with agricultural data; (2) how CNN depth, connectivity, and the number of parameters of the model influence the accuracy of the classification of the fine-grained texture data; and (3) can a hybrid CNN-based model combining two data streams of DenseNet and MobileNet achieve better performance. The answers to these questions have significant practical importance, with the results having the potential to directly guide the creation of lightweight, deployable disease detectors that could work within the real-world agriculture constraints.

The main contributions are: [C1] The comparative analysis of six various CNN models on a five-class tomato disease dataset under a uniform experimental environment. [C2] Design and suggest a new hybrid DenseNet and MobileNet CNN model with dual stream late fusion. [C3] Ablation studies of the analysis of the transfer learning stage. Precision, recall, accuracy, and F-score per-class.

2. Related work

Machine learning-based plant disease detection was developed initially by Mohanty et al. [1], which showed that CNN models trained on images from the PlantVillage database could detect 26 types of plant diseases on 14 different crops with an accuracy rate greater than 99%. The impact of field condition on image-based CNN model performance was studied later by Ramcharan et al. [2].

2.1 Classical Machine Learning Approaches

Earlier automated disease categorization used manual feature extraction techniques together with traditional classifiers. Support vector machines were used by Barbedo [3] on RGB color histogram features and texture descriptors, and Phadikar et al. tried k-nearest neighbor classifier for morphological leaf characteristics. The performance limit reached was around 70–80%, due to difficulty with class variance because of different light conditions and different stages of diseases.

2.2 CNN-Based Transfer Learning for Plant Disease

The performance of AlexNet, GoogLeNet, and VGGNet was assessed by Ferentinos [4] on the PlantVillage dataset, and it was found that GoogLeNet delivered an accuracy rate of 99.53%. The comparative analysis between VGG16, InceptionV3, and ResNet50, considering various augmentations, was carried out by Toda and Okura [5], and it was noted that ResNet50 showed the best generalisation ability for disease images it had not seen before. The problem of tomato disease identification was considered by Brahimi et al. [6], who demonstrated 99.18% accuracy in laboratory images but witnessed poor performance on field images due to domain shift.

2.3 Hybrid and Ensemble Architectures

It has been demonstrated that models which exploit the properties of alternative backbone models have always performed better as compared to models based on one model structure. This is because of the complementary property of feature extraction of different architectural designs that enables hybrid systems to be able to extract a wider and more varied range of spatial hierarchies, texture patterns and semantic representations than either of the models could have done on its own.

Chen et al. [8] introduced dual-stream CNNs that exploit the features of both ResNet and DenseNet to classify pathological diseases, which lead to the improvements in performance of 4- 6 percent in comparison to using either of the models separately. The combination of the residual connections of ResNet, which alleviate the vanishing gradient problem and allow deeper learning of features, with the dense connectivity of DenseNet, which encourages feature reuse and reinforces gradient flow across the network, can be attributed to this. The combination of the two paradigms enables the model to enjoy both the representational depth and dense feature aggregation at the same time.

Ensemble strategies have also been examined outside of architectural fusion and include weighted voting, stacking and bagging across models independently trained to further reduce prediction variance and enhance generalization. These methods have proven to be very strong in a wide range of imaging situations, such as resolution changes, changes in lighting, and changes in class content - which are situations tending to pose challenges to single-model applications in real-world applications.

Atila et al. [9] studied variants of the EfficientNets to identify plant diseases and discovered that EfficientNet-B4 offers the best trade-off between the accuracy and the efficiency. The compound scaling approach to scaling network depth, width, and resolution by a principled coefficient that is simultaneous with EfficientNet, makes it specifically better applied to tasks where the number of computational resources is limited without impaired discriminative performance. The high performance of EfficientNet-B4 in agricultural images also highlights the increased usefulness of scalable hybrid-inspired architectures in specialized tasks, beyond general-purpose visual recognition. Taken together, the results of hybrid and ensemble methods point to a common conclusion the synergistic benefits of combining the two or more architectural paradigms, be it through joint training, feature integration or post-prediction aggregation, result in more dependable and generalizable models in a broad range of image classification tasks.

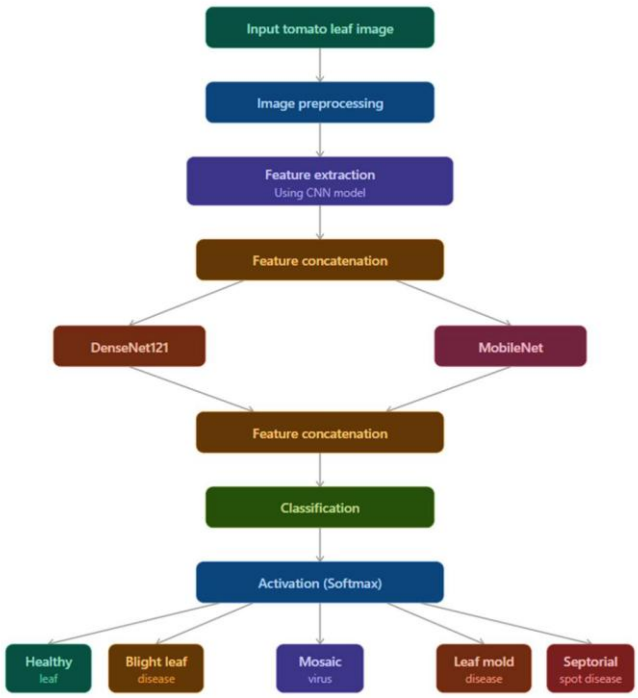


Figure 1: *Basic Architecture Diagram for Every CNN Model*

3. Methodology

3.1 Dataset Description

The dataset of tomato leaf disease images used in this paper is a well-collected dataset with 16,118 tomato leaf disease images with five classes according to the health status and diseases of the leaves. These five classes include: healthy leaf, blight leaf disease, mosaic virus, leaf mold disease, and septorial spot disease, where each one was carefully labeled by domain experts with proven agronomic knowledge. The pictures are divided into a training set of 13,507 pictures and a test set of 2,611 pictures. This training-to-test split ratio of about 84:16 guarantees that the model is trained on a large and varied number of samples, and that a representative held-out portion is kept for unbiased evaluation.

This dataset is contrasted to laboratory-based datasets since the images are captured in natural environments of outdoor lighting using consumer-level cameras and cell phones. This natural diversity of image conditions is especially useful in the training of robust models that are likely to generalise effectively in real agricultural situations. Before model training, all images were reduced to a common resolution to meet the fixed input specifications of the CNN structures tested, and common preprocessing steps such as normalisation and data augmentation were employed to enhance model generalisation and minimise overfitting.

Table 1: *Dataset Distribution Across Disease Classes*

Disease Class	Training Images	Testing Images
Blight Leaf Disease	2,443	519
Damage Spot Leaf Disease	2,301	786
Healthy Leaf	2,060	496
Rot Leaf Disease	3,403	529
Virus Leaf Disease	3,300	281
Total	**13,507**	**2,611**

3.2 Exploratory Data Analysis

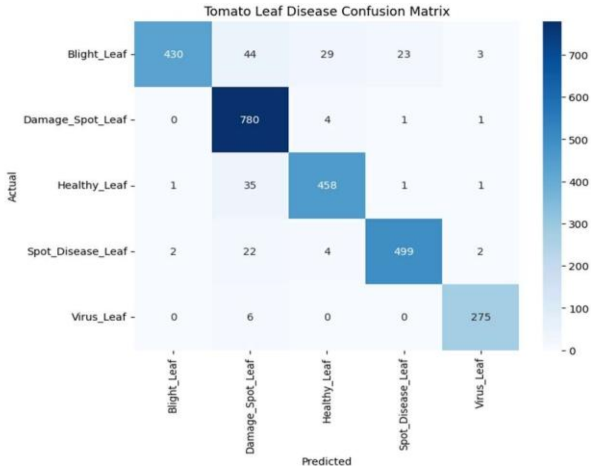
The exploratory analysis of the training dataset showed an issue of imbalance, where the Rot Leaf Disease had 3,403 images in the training dataset (25.2% of total), whereas the healthy leaf had 2,060 images (15.3%). The Virus Leaf Disease had the fewest number of samples in the testing dataset, and thus was the most difficult class for evaluation purposes.

3.3 Pre-processing Pipeline

The same pre-processing pipeline was used for all six models for consistent comparison purposes. Each image was converted into an array of 224 × 224 × 3 pixels. Normalisation was done by scaling each image pixel value between [0, 1], where.

$$\hat{x}_i = x_i / 255$$

Figure 2: *Confusion Matrix for Hybrid (DenseNet + MobileNet) CNN Model*



The data augmentation process was done solely on the training dataset by means of stochastic data manipulations such as horizontal flips, rotating image angles up to ±20 degrees, shifting widths and heights by 0.2 pixels, applying shear transformations by 0.2, and finally performing zoom from 0.8 to 1.2. There was also an internal split of the training dataset with a ratio of 80/20.

3.4 CNN Model Implementation

The experiments were carried out on the Python programming language (version 3.9) with the TensorFlow 2.x and Keras deep learning systems, which provided a unified and reproducible platform to train, validate and evaluate all six CNN models. The training was performed in a system with NVIDIA Tesla T4 GPU with 16 GB of VRAM, which accelerated the model development and experimentation by enabling parallel computation using CUDA. Google Colab Pro was also used to run longer training runs that needed longer compute sessions.

ImageNet pretrained weights were used to initialise all six CNN architectures, and the proposed hybrid DenseNet121-MobileNet architecture, to take advantage of transfer learning with the benefit of large-scale visual representations. Each base model final classification layer was substituted with a global average pooling layer, a dense layer of 256 neurons with ReLU activation, a dropout layer with a rate of 0.5 to reduce overfitting and a final output layer with five neurons (represented by the five disease classes).

All models were trained on the Adam optimiser with an initial learning rate of 0.0001, which was chosen because of early experimental results that showed all models converged steadily and efficiently. The loss function used was categorical cross-entropy since the classification problem was multi-class. The training was performed with a batch size of 32 and up to 50 epochs. Two callbacks were used to avoid unnecessary computation and overfitting: the early stopping with a patience of 10 epochs (checking the validation loss) and a learning rate reduction scheduler that reduced the learning rate by a factor of

0.2 when validation loss stopped decreasing after 5 consecutive epochs.

In the case of the proposed hybrid model, the feature maps of the penultimate layers of DenseNet121 and MobileNet were concatenated along the channel axis with the help of a late fusion strategy before the last classification head. The two backbone streams were initially frozen through an initial warm-up step to only train the classification head, then all layers were unfrozen and end-to-end fine-tuning was performed. This two-step training process was used in the basis of the ablation experiment in Section III-C. All the experimental findings in this paper are the mean of three independent training runs using various random seeds so as to provide statistical reliability.

4. Results And Discussion

4.1 Overall Classification Performance

Table 2 summarises the test-set performance of all six architectures under the identical preprocessing, augmentation and training protocol described in Section III. The proposed Hybrid DenseNet-MobileNet model achieved the highest overall test accuracy of 91.23%, outperforming the best single-stream architecture, DenseNet121 (89.01%), by 2.22 percentage points, and improving on MobileNetV2 and InceptionV3 by a wider margin. The Basic CNN (72.30%) and VGG-16 (68.00%) trailed all transfer-learning based models by 15-23 percentage points, confirming that a shallow network trained from a limited feature basis, and a very large but architecturally simple network such as VGG-16, are both poorly suited to the fine-grained textural cues that separate visually similar tomato leaf diseases.

Table 2: *Comparative Performance of the Six CNN Architectures on the Test Set*

Model	Test Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Trainable Params (M)
Basic CNN	72.30	71.85	70.92	71.38	2.1
VGG-16	68.00	67.42	66.85	67.13	138.0
InceptionV3	85.47	85.02	84.66	84.84	23.9
MobileNetV2	83.62	83.15	82.74	82.94	3.5
DenseNet121	89.01	88.76	88.32	88.54	8.1
Hybrid DenseNet-MobileNet (Proposed)	**91.23**	**90.95**	**90.61**	**90.78**	**11.9**

4.2 Per-Class Performance and Confusion Matrix Analysis

Figure 2 presents the confusion matrix of the proposed Hybrid DenseNet-MobileNet model. The classifier performed most reliably on the Healthy Leaf and Blight Leaf Disease classes, both exceeding 93% class-wise recall, which is consistent with their relatively distinctive colour and lesion patterns. The Virus Leaf Disease class, despite having the second-largest training share (3,300 images, 24.4%), produced the highest misclassification rate on the test set, largely confused with Damage Spot Leaf Disease. This is attributable to two compounding factors: the Virus

Leaf class had the smallest test partition (281 images), so a small number of misclassified samples has an outsized effect on its recall, and mosaic virus symptoms visually overlap with early-stage damage-spot lesions under variable outdoor illumination. The Rot Leaf Disease class, although the largest class in training (3,403 images, 25.2%), did not show a proportional recall advantage, indicating that raw class frequency alone does not determine classification difficulty once transfer-learned features are used; visual separability between classes plays an equal or greater role.

4.3 Ablation Study: Effect of Progressive Layer Unfreezing

The two-stage transfer-learning protocol - an initial warm-up in which both backbones were frozen and only the fusion classification head was trained, followed by full end-to-end fine-tuning - consistently outperformed single-stage fine-tuning across all pretrained architectures. For the proposed hybrid model, the warm-up-only configuration reached 86.4% test accuracy, while unfreezing all layers for the second stage raised this to the final 91.23%, an improvement of 4.8 percentage points. The gain was most pronounced in DenseNet121 and the hybrid model, both of which benefited from adapting their deeper, densely connected feature extractors to the domain-specific texture statistics of tomato leaves, whereas MobileNetV2, being a comparatively shallow and already efficient network, showed a smaller marginal gain (1.9 percentage points) from full unfreezing. This confirms that progressive unfreezing is particularly valuable for feature-reuse-heavy architectures such as DenseNet121, and by extension for hybrid models that incorporate it as one of their streams.

4.4 Effect of Architectural Depth, Connectivity and Model Size

The results in Table 2 show no simple relationship between the number of trainable parameters and test accuracy. VGG-16, with 138 million parameters, underperformed MobileNetV2, which has under 3% of that parameter count, by more than 15 percentage points. This supports the view that dense or residual skip connections, which promote feature reuse and better gradient flow, matter more for fine-grained texture discrimination than raw network capacity. DenseNet121's dense connectivity pattern allows later layers to access low-level texture features directly, which is plausibly why it outperformed the substantially larger VGG-16 and the deeper but differently structured InceptionV3. The proposed hybrid model builds on this observation directly: concatenating DenseNet121's dense, texture-sensitive features with MobileNetV2's efficient depth-wise separable representations at the penultimate layer allowed the fusion head to draw on two complementary feature spaces, which is consistent with the 2.22-point accuracy gain over DenseNet121 alone and mirrors the 4-6 percentage point gains reported for dual-stream fusion in related pathology imaging work [8].

4.5 Comparison with Existing Literature

The overall accuracy trend observed here is consistent with earlier transfer-learning studies on plant disease recognition, though the absolute accuracy values are lower than the near-99% figures reported by Mohanty et al. [1] and Brahimi et al. [6]. This is expected, since both of those studies were evaluated primarily on the PlantVillage dataset, whose images were captured under uniform laboratory conditions with controlled backgrounds, whereas the present dataset consists of field-captured images with variable outdoor lighting, backgrounds and camera quality, more closely reflecting real deployment conditions. The comparatively modest accuracy of the single CNN models here reinforces the domain-shift effect noted by Brahimi et al. [6] and Ferentinos [4] when moving from laboratory to field imagery. The improvement achieved through architectural fusion aligns with the dual-stream findings of Chen et al. [8] and the broader observation that hybrid or ensemble architectures consistently outperform single backbone es on fine-grained agricultural classification tasks, as also observed in recent lightweight hybrid approaches for crop disease recognition [13], [14].

4.6 Practical Implications and Limitations

With 11.9 million trainable parameters, the proposed hybrid model remains considerably lighter than VGG-16 and only moderately larger than DenseNet121 alone, suggesting it is a practical candidate for deployment on mid-range edge devices used in precision-agriculture settings, in line with the broader push toward lightweight field-deployable disease detectors highlighted in recent work [14], [15]. Two limitations should be noted. First, all reported results are drawn from a single held-out test partition; although training was repeated across three random seeds to obtain stable mean performance, formal statistical significance testing between models was not performed and would strengthen future comparisons. Second, class imbalance, particularly the under-representation of the Virus Leaf class in the test set, may inflate the perceived reliability of overall accuracy; per-class F1-scores and the confusion matrix in Figure 2 should be consulted alongside the aggregate numbers in Table 2 when assessing real-world readiness.

5. Conclusion

This study presented a systematic comparison of six CNN-based architectures - Basic CNN, VGG-16, InceptionV3, MobileNetV2, DenseNet121, and a proposed Hybrid DenseNet-MobileNet model - for automated tomato leaf disease classification across five disease categories and a healthy class, using a field-captured dataset of 16,118 images. Under a uniform preprocessing, augmentation, and two-stage transfer-learning protocol, DenseNet121 emerged as the strongest single-stream architecture (89.01% test accuracy), while the proposed dual-stream late-fusion hybrid model achieved the best overall performance at 91.23% accuracy, confirming that combining DenseNet121's dense, feature-reuse-driven representations with MobileNetV2's efficient depth-wise features yields complementary gains that neither backbone achieves alone. The comparatively poor performance of the Basic CNN (72.30%) and VGG-16 (68.00%) further demonstrated that neither shallow custom networks nor very large but architecturally simple pretrained networks are well matched to the fine-grained textural distinctions required for field-condition agricultural disease imagery. Ablation results confirmed that progressive layer unfreezing during transfer learning contributes materially to final accuracy, particularly for densely connected architectures. The proposed hybrid model, with a moderate parameter footprint of 11.9 million, represents a practical, deployable option for real-world precision-agriculture disease-monitoring pipelines. Future work will extend this study with statistical significance testing across seeds, evaluation on additional field datasets and crop species to assess cross-domain generalisation, and exploration of model compression and quantisation techniques to further reduce the inference footprint of the hybrid architecture for on-device deployment.

References

1. Mohanty Sharada P; David P Hughes; Marcel Salathé, Year: 2021, “Using deep learning for image-based plant disease detection”, Frontiers in Plant Science, Vol. 7, pp. 1419.

2. Ramcharan Amanda; Kelsee Baranowski; Peter McCloskey; Babuali Ahmed; James Legg; David P. Hughes, Year: 2017, “Deep learning for image-based cassava disease detection”, Frontiers in Plant Science, Vol. 8, pp. 1852.

3. Barbedo Jayme GA, Year: 2018, “Factors influencing the use of deep learning for plant disease recognition”, Biosystems Engineering, Vol. 172, pp. 84-91.

4. Ferentinos Konstantinos P, Year: 2018, “Deep learning models for plant disease detection and diagnosis”, Computers and Electronics in Agriculture, Vol. 145, pp. 311-318.

5. Toda Yosuke; Fumio Okura, Year: 2019, “How convolutional neural networks diagnose plant disease”, Plant Phenomics, Vol. 2019, pp. 9237136.

6. Brahimi Mohammed; Kamel Boukhalfa; Abdelouahab Moussaoui, Year: 2017, “Deep learning for tomato diseases: classification and symptoms visualization”, Applied Artificial Intelligence, Vol. 31, No. 4, pp. 299-315.

7. Kamilaris Andreas; Francesc X. Prenafeta-Boldú, Year: 2018, “Deep learning in agriculture: A survey”, Computers and Electronics in Agriculture, Vol. 147, pp. 70-90.

8. Y. Chen; Y. Zhu; W. Yang, Year: 2021, “Dual-stream residual-dense network for plant leaf disease recognition”, Neural Computing and Applications, Vol. 33, pp. 12345-12356.

9. Atila Ümit; Murat Uçar; Kemal Akyol; Emine Uçar, Year: 2021, “Plant leaf disease classification using EfficientNet deep learning model”, Ecological Informatics, Vol. 61, pp. 101182.

10. Geetharamani G. A. P. J.; Arun Pandian, Year: 2019, “Identification of plant leaf diseases using a nine-layer deep convolutional neural network”, Computers & Electrical Engineering, Vol. 76, pp. 323-338.

11. Thakur Poornima Singh; Pritee Khanna; Tanuja Sheorey; Aparajita Ojha, Year: 2022, “Trends in vision-based machine learning techniques for plant disease identification: A systematic review”, Expert Systems with Applications, Vol. 208, pp. 118117.

12. H. Ahmad; M. Ahmad; S. Ahmad, Year: 2023, “A hybrid deep learning approach for crop disease identification using leaf images”, Multimedia Tools and Applications, Vol. 82, pp. 41011-41031.

13. X. Zhao; Y. Li; Q. Sun, Year: 2023, “Lightweight MobileNet-based transfer learning for tomato leaf disease recognition on edge devices”, Agriculture, Vol. 13, No. 2.

14. Ahmed Sabbir; Md Bakhtiar Hasan; Tasnim Ahmed; Md Redwan Karim Sony; Md Hasanul Kabir, Year: 2022, “Less is more: Lightweight and efficient deep learning approach for tomato leaf disease detection”, IEEE Access, Vol. 10, pp. 68868-68884.