

# A Comparative Analysis of Lightweight and Non-Lightweight Deep Learning Models for Agricultural Pest Identification

Chenglin Luo

College of Artificial Intelligence, South China Normal University, Foshan, Guangdong, 528225, China

20222005080@m.scnu.edu.com

**Abstract.** A key component of effective pest control is pest recognition, which involves the exact identification of pests and the prediction of outbreaks through the analysis of data from statistics, mathematics, and biology. More effective identification techniques are required as traditional field surveys, despite their depth, are labor-intensive and have a limited reach. This skill is crucial for the early detection of pests and diseases, enabling prompt actions to prevent their spread. With the advancement of artificial intelligence, approaches like Convolutional Neural Networks (CNNs) can handle vast amounts of visual data gathered by drones, cellphones, and other devices for large-scale agricultural applications, allowing for automatic, real-time monitoring. However, the bulkiness of standard CNN models makes them challenging to deploy on small mobile devices. To achieve effective and precise pest detection in a mobile environment, this study analyzes various lightweight and non-lightweight models for pest recognition. Results indicate that lightweight models show potential for mobile deployment with reasonable accuracy. Furthermore, with additional training and optimization, lightweight models can be effectively utilized for mobile pest identification, offering a promising direction for integrating advanced pest prediction technologies into practical agricultural applications.

**Keywords:** Pest recognition; convolutional neural networks; lightweight model.

## 1. Introduction

In order to precisely identify pests and anticipate pest outbreaks, pest prediction involves evaluating relevant data from disciplines including biology, mathematics, and statistics. For farmers, efficient pest management is essential. By employing timely and accurate pest predictions, farmers can efficiently control agricultural pests and reduce the adverse effects of pesticide use on the environment and public health [1,2]. The conventional approach to gathering pest information is conducting traditional field surveys [3]. Although field surveys are limited in scope and labor-intensive when conducted on a broad scale, they can yield detailed information regarding pests [4,5]. It is true that precise pest identification and classification are essential for pest prediction. The need for a quick and accurate way to identify pests is increasing.

Technological innovations, including machine learning and artificial intelligence, are being investigated to improve the precision and speed of pest identification. These tools have the potential to revolutionize how people monitor and control agricultural pests because they can evaluate data and images more quickly and effectively than conventional techniques [5]. Deep learning models, particularly convolutional neural networks (CNNs), have demonstrated great accuracy in image recognition tasks, and as a result, deep learning offers considerable advantages in photo recognition and pest identification. By using multi-level feature representations, these models can automatically extract features from photos and enhance identification accuracy. Because of this, CNNs may be able to identify diseases and pests at an early stage, allowing for early diagnosis and control as well as preventing the spread of diseases and pests. Large-scale agricultural production requires the ability to process massive amounts of visual data, which deep learning models can handle. Large-scale and automated pest and disease monitoring can be achieved by collecting field photos using drones, cellphones, and other devices, then analyzing the images in real time using deep learning models. Typical models, however, are bulky and unsuitable for use on small mobile devices. Numerous CNN models have been created and refined in an effort to achieve lightweight recognition [6,7]. This

research investigates the feasibility of implementing CNNs on mobile devices through an analysis of multiple models, both lightweight and non-lightweight.

## 2. Method

### 2.1. Dataset

Strict duplicate checks are carried out on the original dataset, which is based on Kaggle's farm pest dataset. These checks use picture hashing and feature matching techniques to identify and eliminate any duplicate images, guaranteeing that every image in the dataset is unique. Images with evident pest networks were gathered from public image libraries and other trustworthy sources and incorporated to the collection in order to boost diversity. These kinds of photos can help train models more accurately to recognize the pests' habitats. made sure the pest species in the images are varied by choosing a few with notable background color changes (such as varied leaf colors, dirt backdrops, etc.). Some images with pest hues that closely resemble the surrounding environment were chosen in order to improve the data's realistic scene portrayal. These examples were used to train the model to recognize objects better in complicated backdrops and to find pests hidden in corners or on photo edges. This improved the model's performance in identifying targets in edge areas. Lastly, make sure that every newly submitted image has accurate label information by carefully classifying and hand labeling each one. The author has added some shots of pest networks that are more pronounced and some pictures of insects with backgrounds that are noticeably different from one another. In the end, a dataset of 27648 images was produced, including a range of pests and background conditions to guarantee their representativeness and diversity and to facilitate efficient training and testing of farm pest identification models.

### 2.2. Models

#### 2.2.1 Non-lightweight Models

He Kaiming and colleagues were victorious in the 2015 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) with Residual Network (ResNet) [8]. This network addresses the vanishing or exploding gradient problem and deterioration, two issues brought on by excessive stacking in conventional convolutional neural networks. Regarding the first difficulty, the preparation of the data and the use of Batch Normalization (BN) layers in the network are suggested solutions to gradient vanishing or exploding problems. For the second issue, a residual structure was proposed as a solution to the degradation problem.

A common ResNet model is ResNet50 [8]. It employs the BottleNeck residual structure, which uses layers of convolutional filters (1x1, 3x3, and 1x1), BN, and Rectified Linear Units (ReLU) in that order. The first 1x1 convolutional layer performs dimensionality reduction, whereas the second 1x1 convolutional layer performs dimension expansion. ResNet50 utilizes two distinct bottleneck topologies. The first one adds convolutional layers to the constant mapping branch, increasing the number of input and output feature map channels for residual blocks compared to the second one. This helps to resolve the gradient vanishing issue and enables the training of deeper networks. The model can more easily extract complex features from the input due to these residual connections, which facilitates smoother cross-layer information flow.

Sharing the same core ideas as its predecessor, ResNet50V2, but with increased depth and complexity, the ResNet152V2 architecture marks a significant step towards more sophisticated and complex deep learning networks [8]. Thanks to the additional layers and functionality in this architecture, this work can extract more advanced features from the farm bug photos, including subtle variances and intricate patterns.

#### 2.2.2 Lightweight Models

Google introduced the concept of MobileNet in 2017 [9]. It is based on the VGG model and utilizes depthwise separable convolutions instead of the standard convolutional layers found in VGG,

resulting in a significant reduction in both the number of parameters and computational load while maintaining minimal loss in accuracy [10]. A typical example of MobileNet is MobileNetV1, which features an accurate and lightweight structure.

As an improved version of MobileNetV1, MobileNetV2 provided new insights into how to find the ideal backbone network for the task of classifying farm insects. Due to its efficiency-focused design, MobileNetV2 is an excellent option for situations where processing power is limited. With its simplified architecture, depthwise separable convolution, and emphasis on efficiency and agility, it allows the author to drastically reduce parameters and computational complexity without sacrificing speed. Thanks to its lightweight design, implementing CNNs on mobile devices is now feasible, aligning with the vision.

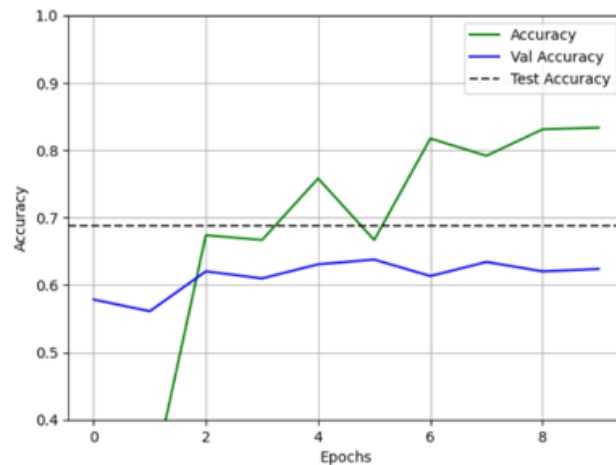
Researchers at the Google Research Institute carefully examined how the breadth, depth, and input image processing rate of network models affected performance. In 2019, they developed the lightweight, high-recognition-rate EfficientNet model [11]. This model is constructed by stacking several Mobile Inverted Residual Conv (MBConv) modules on top of one another. For each MBConv, dimensionality enhancement of the input residual block feature map is first conducted using a 1x1 convolution, followed by the extraction of additional image features using depthwise separable convolution. Secondly, the channels of the feature map are given varying weights using the Squeeze-and-Excitation (SE) attention mechanism. After that, the dimensionality of the input feature map is reduced using a 1x1 convolution. To produce the final output, a Dropout layer is applied to the reduced-dimensional feature map. Additionally, a shortcut operation is carried out (adding the input feature map to the output feature map) when the number of channels in the feature map of the input residual block matches the number of channels in the feature map of the output residual block and the stride is 1. The EfficientNet family comes in several versions. This study selects the EfficientNet-B0 with the lowest complexity as the base model for pest classification and recognition tasks after carefully considering the requirements of performance, complexity, and computational resources.

### 3. Result and Discussion

In this study, the model used an Adam optimizer with RGB color depth ranging from 10 to 100 epochs, a validation set size of 10%, a batch size of 32, a learning rate of 0.001, and an ImageDataGenerator to perform transformations such as rotation, translation, cropping, scaling, and horizontal flipping on the images to generate new enhanced images. The evaluation metrics for this research include the following: Accuracy, Quantified, Loss, AUC-ROC, Precision, Recall, and F1 score. Table 1 demonstrates the comparison among different models and Fig. 1 displays the accuracy curves during training.

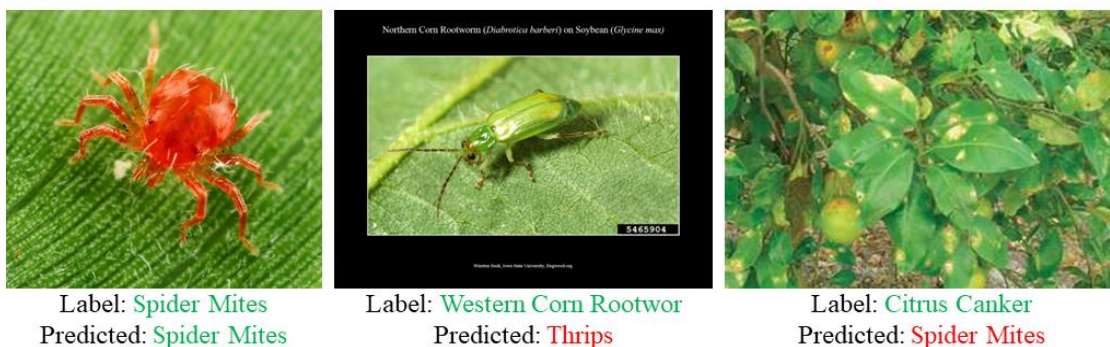
**Table 1.** Performance comparison among different models.

Model	Accuracy	Precision	F1 Score	Recall	AUC-ROC
MobilenetV1	0.69	0.72	0.69	0.69	0.91
MobilenetV2	0.49	0.64	0.45	0.49	0.90
EfficientNet	0.68	0.74	0.69	0.68	0.96
ResNet50	0.71	0.75	0.73	0.71	0.97
ResNet152V2	0.70	0.77	0.74	0.72	0.98



**Fig. 1** Validation accuracy during training using MobileNetV2 (Figure Credits: Original)

The aforementioned results make it clear that complex feature interactions in images are usually better captured by large models with intricate internal parameters and structures than by tiny models. This explains why ResNet models function better despite being heavier than other models. However, lightweight models can achieve similar or even better performance by adjusting the hyperparameters. This demonstrates unequivocally that smaller models' performance may be adjusted to equal that of larger units. Furthermore, as demonstrated in Fig. 2 below, all models have attained a high degree of generalization ability, allowing them to correctly learn and apply features and identify the majority of photos. The experiment still has certain drawbacks, though. The training dataset is prone to overfitting because of its small size. Additionally, due to constraints on training time and computing power, the hyperparameter combinations this work has selected may not accurately represent the optimal parameters of each model. This could cause the model's ultimate performance to deviate significantly. Furthermore, the model performance comparison lacks a statistical significance analysis, and further investigation is required to pinpoint the precise mechanism via which hyperparameters enhance performance. Furthermore, Fig. 2 demonstrates how poorly the model recognizes fuzzy photographs and images where it is impossible to discern between the surroundings and pests. More data could be required to train and enhance the model so that it can more accurately capture some subtle elements. When all things are taken into account, the model can be lightweight and used on mobile devices; more training and model optimization will be necessary for this.



**Fig. 2** Visualization of representative examples (Figure Credits: Original)

#### 4. Conclusion

The results of this study highlight how large models, like ResNet, can capture nuanced feature interactions in images due to their complex structures and parameters, which results in better performance. However, lightweight models can achieve similar or even better results by fine-tuning their hyperparameters, demonstrating that fine-tuning can close the performance gap between larger

and smaller models. All models demonstrated excellent generalization skills, correctly learning and applying characteristics in a variety of contexts.

Despite these achievements, the study has several drawbacks, most notably the small size of the training dataset, which raises the possibility of overfitting. Furthermore, the limited range of hyperparameter combinations examined, due to training time and computational resource limitations, might not be the optimal settings for every model, which could impact performance results. More scrutiny is required, as evidenced by the lack of a statistical significance analysis in the model comparisons.

In addition, the models had difficulty with blurry or hard-to-distinguish images of settings and pests, suggesting that more varied and substantial training data would be needed to improve the model's ability to pick up on minor details. Overall, even though lightweight models can be successfully used on mobile devices, more model optimization, more data, and intensive training will be needed to achieve the best results.

## References

- [1] Liu Chengkang, Zhai Zhiqiang, Zhang Ruoyu, et al. Field pest monitoring and forecasting system for pest control. *Frontiers in Plant Science*, 2022, 13: 990965.
- [2] Li Rui, Wang Rujing, Zhang Jie, et al. An effective data augmentation strategy for CNN-based pest localization and recognition in the field. *IEEE access*, 2019, 7: 160274-160283.
- [3] Ebrahimi Ma, Khoshtaghaza Mohammad Hadi, Minaei Saeid, et al. Vision-based pest detection based on SVM classification method. *Computers and Electronics in Agriculture*, 2017, 137: 52-58.
- [4] Dong Yingying, Xu Fang, Liu Linyi, et al. Monitoring and forecasting for disease and pest in crop based on WebGIS system. *International Conference on Agro-Geoinformatics*. 2019: 1-5.
- [5] Albattah Waleed, Masood Momina, Javed Ali, et al. Custom CornerNet: a drone-based improved deep learning technique for large-scale multiclass pest localization and classification. *Complex & Intelligent Systems*, 2023, 9(2): 1299-1316.
- [6] Chen Ching-Ju, Chen Ling-Wei, Yang Chun-Hao, et al. Improving CNN-based pest recognition with a Post-hoc Explanation of XAI. 2021: 1-28.
- [7] Li Wenyong, Zheng Tengfei, Yang Zhankui, et al. Classification and detection of insects from field images using deep learning for smart pest management: A systematic review. *Ecological Informatics*, 2021, 66: 101460.
- [8] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016: 770-778.
- [9] Howard, Andrew. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.
- [10] Chollet François. Xception: Deep learning with depthwise separable convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017: 1251-1258.
- [11] Tan Mingxing. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.