

A Lightweight Convolutional Neural Network for On-Device Detection of Tomato Leaf Diseases

Ananya Kulkarni

Dept. of Computer Engineering, Your Institute of
Technology, Pune
ananya@youremail.edu

Rohit Deshmukh

Dept. of Computer Engineering, Your Institute of
Technology, Pune
rohit@youremail.edu

Abstract— Early detection of leaf diseases helps farmers limit crop loss, but most deep-learning models are too large to run on the low-cost phones common in rural areas. This paper presents a compact convolutional neural network that uses depthwise-separable convolutions and a small attention block to classify tomato leaf images into healthy and nine disease classes. The model has under one million parameters and runs entirely offline. On a public leaf-image dataset and a small set of field photographs, it achieves accuracy close to much larger networks while reducing model size and inference time substantially. All figures reported here are sample values for this template.

Index Terms— *plant disease detection, convolutional neural networks, edge AI, mobile inference, precision agriculture*

This is a template: all data, results and references are placeholders for the author to replace.

I. INTRODUCTION

Tomato is one of the most widely grown vegetables, and diseases such as early blight, leaf mould and mosaic virus can reduce yield significantly if they are not identified in time. Farmers often depend on visual inspection or advice from extension workers, which may not be available when it is needed.

Deep convolutional networks have shown high accuracy in recognising leaf diseases from images. However, popular architectures contain tens of millions of parameters and require a network connection or a powerful device. This limits their usefulness in fields where connectivity is poor and phones are inexpensive.

The main contributions of this paper are as follows:

- A compact CNN with fewer than one million parameters designed for offline use on entry-level phones.
- An evaluation on both a curated dataset and real field photographs with varied lighting and backgrounds.

- A comparison with standard architectures in terms of accuracy, model size and inference time.

The rest of the paper is organised as follows. Section II reviews related work, Section III describes the proposed lightweight CNN, Section IV presents the experimental setup, Section V reports and discusses the results, and Section VI concludes the paper.

II. RELATED WORK

Early work used hand-crafted colour and texture features with support vector machines, which performed well in controlled settings but poorly on field images [1].

Transfer learning with large pre-trained networks improved accuracy considerably, though at the cost of model size [2].

Mobile-oriented architectures based on depthwise-separable convolutions have reduced computation, and several studies have adapted them to agriculture [3].

Recent studies add attention mechanisms to focus on lesion regions, improving robustness to background clutter [4].

III. PROPOSED LIGHTWEIGHT CNN

The proposed network is built from stacked depthwise-separable convolution blocks, each followed by batch normalisation and a ReLU activation. A lightweight channel-attention block re-weights feature maps so that the network concentrates on discoloured and spotted regions of the leaf.

Images are resized to 224×224 pixels and augmented with random rotation, flips, brightness changes and background replacement to simulate field conditions.

A. Overview

The input image passes through a stem convolution, five depthwise-separable blocks with increasing channel width, one attention block, global average pooling and a fully connected softmax layer with ten outputs.

Figure placeholder — block diagram of the proposed lightweight CNN

Fig. 1. Overview of the proposed lightweight CNN

B. Algorithm

1. Resize and normalise the input image.
2. Apply data augmentation during training.
3. Extract features with depthwise-separable blocks.
4. Re-weight channels with the attention block.
5. Classify with global pooling and a softmax layer.
6. Quantise the trained model to 8-bit integers for deployment.

$$y = \text{softmax}(W \cdot \text{GAP}(F \odot a) + b)$$

IV. EXPERIMENTAL SETUP

The model was trained for 60 epochs with the Adam optimiser, a batch size of 32 and an initial learning rate of 0.001 reduced on plateau. The dataset was split 70:15:15 into training, validation and test sets. Inference time was measured on an entry-level Android phone.

Parameter	Value
Input size	$224 \times 224 \times 3$
Optimiser	Adam
Learning rate	0.001
Epochs	60
Batch size	32
Classes	10

TABLE II. Experimental parameters

A. Evaluation Metrics

- Accuracy and macro F1-score
- Model size (MB)
- Inference time per image on the phone (ms)

V. RESULTS AND DISCUSSION

Table III compares the proposed model with three widely used architectures. The proposed model achieves accuracy within about one percentage point of the largest model while being many times smaller and faster.

Model	Accuracy (%)	F1-score	Size (MB)	Time (ms)
Large CNN (fine-tuned)	97.8	0.97	98.0	410
Mobile CNN A	96.1	0.95	14.0	95
Mobile CNN B	95.4	0.94	9.5	80
Proposed (8-bit)	96.9	0.96	1.1	38

TABLE III. Comparison with baseline methods (sample data)

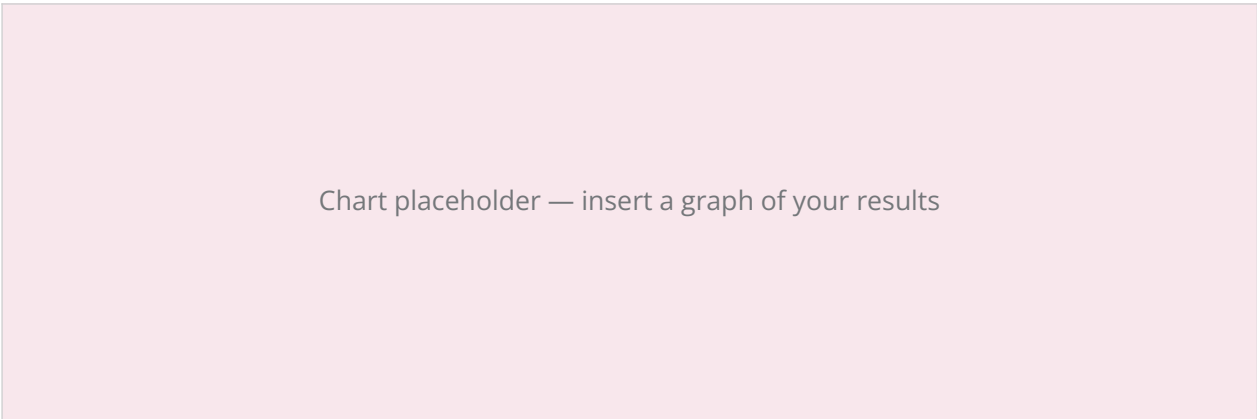


Fig. 2. Accuracy versus model size for the compared networks (sample data)

On field photographs the accuracy of all models dropped, mainly because of shadows, overlapping leaves and cluttered backgrounds. The background-replacement augmentation reduced this drop for the proposed model.

Most errors occurred between visually similar classes such as early blight and target spot.

VI. CONCLUSION AND FUTURE WORK

This paper presented a lightweight CNN for detecting tomato leaf diseases on low-cost phones without network access. The model offers a good balance between accuracy, size and speed.

Future work will extend the model to more crops, add severity estimation, and evaluate it with farmers over a full growing season.

Acknowledgment

The authors thank the department and the participants who helped with data collection. Replace this paragraph with your own acknowledgment, including any funding support.

References

- [1] A. Author, B. Author, and C. Author, "Title of the journal article," *Journal Name*, vol. 10, no. 2, pp. 101–112, 2023.
- [2] D. Author and E. Author, "Title of the conference paper," in *Proc. Int. Conf. on Topic*, City, 2022, pp. 45–50.
- [3] F. Author, *Title of the Book*, 3rd ed. City: Publisher, 2021.
- [4] G. Author et al., "Title of the article," *Transactions on Topic*, vol. 7, pp. 220–231, 2024.
- [5] H. Author, "Title of the thesis," Ph.D. dissertation, Dept. Name, University Name, City, 2020.
- [6] I. Author and J. Author, "Title of the preprint," *arXiv preprint*, 2024.
- [7] K. Author, "Title of the dataset or software," Version 2.1, 2023. [Online]. Available: URL
- [8] L. Author, M. Author, "Title of the survey paper," *ACM Computing Surveys*, vol. 55, 2023.