



Crop Disease Detection from Leaf Images Using Convolutional Neural Networks

Dr. Syed Khader Bash

Assistant Professor, Mechanical Engineering Department, Muffakham
Jah College of Engineering and Technology, Hyderabad, Telangana
Email: khader.basha@mjclege.ac.in

Abstract

Every year, between 10–40% of the world's crops are lost to plant diseases, and locating an expert diagnostician is often time-consuming for the smallholder farmer. This paper proposes a Convolutional Neural Network (CNN) based disease identification model, using a single picture of the leaves taken by the smartphone camera. Four pretrained architectures, VGG16, ResNet-50, MobileNetV2 and EfficientNet-B0, are fine-tuned on the public PlantVillage dataset (54,305 images, 14 crops, 38 classes) and evaluated for generalization on real field images from PlantDoc. EfficientNet-B0 model, having only 5.3 M parameters, can attain 99.62% test accuracy on the PlantVillage dataset, and in the end, MobileNetV2 can be deployed in 38ms per image in an analysis with a mid-level Android phone. The results are 41.5% on field images, showing a big gap between the lab and field, and mixed domain fine-tuning with background augmentation gets back to 28.3%, achieving 69.8% accuracy on field images. The shape of the leaf and background are not modelled; Grad-CAM visualizations show that the models pay attention to lesion regions. The results show the capability of lightweight CNNs to provide an accurate, explainable, offline diagnosis of disease suitable for deployment in low-connectivity farming zones.

Keywords: Plant disease detection, convolutional neural networks, transfer learning, precision agriculture, PlantVillage, MobileNetV2, EfficientNet, Grad-CAM.

1. INTRODUCTION

The major losses in worldwide staple crops are due to pathogens and pests. The losses in the global production yielded by Savary et al. [1] were approximately 21.5% for wheat, 30% for rice, 22.5% for maize, 17.2% for potato and 21.4% for soybean. In India, the majority of farms are small, and there are a small number of extension officers, so a disease is generally detected after it has become widespread.

The first step to minimizing this damage is to make a timely and accurate diagnosis. Traditional diagnosis involves either visual inspection by a plant pathologist or lab testing, which is both slow and expensive. In the

<https://jisrjournal.com/>

meantime, the ownership and use of smartphones has risen quickly among farmers, which can thus provide an opportunity for camera-based automated diagnosis.

Since the work of AlexNet [2], CNN has revolutionized image classification. Mohanty et al. in the first place showed that a CNN trained on plants of the PlantVillage dataset [3] can be precisely utilized to perceive 26 plant infections in 14 harvests with 99.35% accuracy [4]. A higher accuracy was obtained later in the work [5] [6]. There are, however, two problems that reduce its applications in the real world. First, PlantVillage images are captured in the laboratory and have a plain background; models trained on these images don't work on an image captured from the field [7, 8]. Second, the most accurate models are too large and therefore cannot be run offline on low-cost phones.

This paper deals with both of the problems. Our contributions are:

- Comparison of four CNN architecture including VGG16, ResNet-50, MobileNetV2 and EfficientNet-B0, on PlantVillage using a common training protocol.
- A multi-modal to-do list for field images from PlantDoc [8] with measurement of the lab-to-field accuracy gap.
- We have developed a training strategy called background replacement augmentation that reduces this gap in a cross-domain setting.
- We verify the predictions using the symptoms of the disease by using Grad-CAM [9] analysis, along with latency measurements for offline deployment on mobile devices.

Related works are discussed in Section 2. The methodology is given in Section 3 and the experimental setup in Section 4. The results are presented in Section 5, deployment and limitations are discussed in Section 6, and the conclusion is given in Section 7.

2. RELATED WORK

2.1 Classical Machine Learning Approaches

In the early systems, the lesion was segmented using a glass or a k -NN rule; the color, texture and shape features were hand-crafted, for example, color histograms and GLCM textures, and the hand-crafted features were classified using a support vector machine or a k -NN rule. These pipelines had to be carefully developed for individual crops and suffered from poor performance in different lighting and backgrounds. In this context, Kamilaris and Prenafeta-Boldú survey the changes from these methods to deep learning in agriculture [16].

2.2 CNN-Based Disease Classification

Although Mohanty et al. got 99.35% accuracy when AlexNet and GoogLeNet were trained with PlantVillage, <https://jisrjournal.com/>

the accuracy dropped to around 31% when they used images from other sources [4]. Brahimi et al. achieved a classification accuracy rate of 99.18% for 9 tomato diseases along with saliency maps of the symptoms for visualisation [15]. Ferentinos trained VGG on 87,848 images of 25 plants, reaching 99.53% [5]. According to the comparison of fine-tuned architectures performed by Too et al. [6], DenseNet-121 is the best one with 99.75%. In the field setting, the transfer learning approach has been used by many researchers, such as Ramcharan et al., for cassava disease [17].

2.3 The Lab-to-Field Gap

Barbedo demonstrated that both the amount and diversity of the datasets have a significant impact on real-life performance and that lab datasets do not adequately reflect field conditions [7]. Singh et al. published PlantDoc, 2, 598 field images on 13 species and 27 classes, for which the classification accuracy was also considerably low as compared to PlantVillage [8]. The work bridges a gap between studies that provide high lab accuracy and field evaluation and on-device efficiency.

TABLE 1: SUMMARY OF REPRESENTATIVE PRIOR WORK

Ref.	Dataset	Classes	Model	Accuracy	Field test
[4] 2016	PlantVillage	38	AlexNet, GoogLeNet	99.35%	Yes ($\approx 31\%$)
[15] 2017	PlantVillage (tomato)	9	AlexNet, GoogLeNet	99.18%	No
[5] 2018	Open DB, 87,848 img.	58	VGG, others	99.53%	Partial
[6] 2019	PlantVillage	38	VGG16, ResNet, DenseNet	99.75%	No
[8] 2020	PlantDoc	27	VGG16, InceptionV3	Lower than lab	Yes
Ours	PlantVillage + PlantDoc	38	4 CNNs (Sec. III-B)	XX%	Yes

3. PROPOSED METHODOLOGY

3.1. System Overview

The whole system is composed of three parts: train, explain and deploy (Fig. 1). The images of leaves undergo

<https://jisrjournal.com/>

a process of pre-processing and augmentation, then are sent to a CNN backbone network pre-trained on ImageNet and a new classifier head, which produces the outputs for one of 38 crop–disease classes. Grad-CAM will generate a heatmap of which leaf areas were used to make the decision. The quantized model is used in an offline Android app.

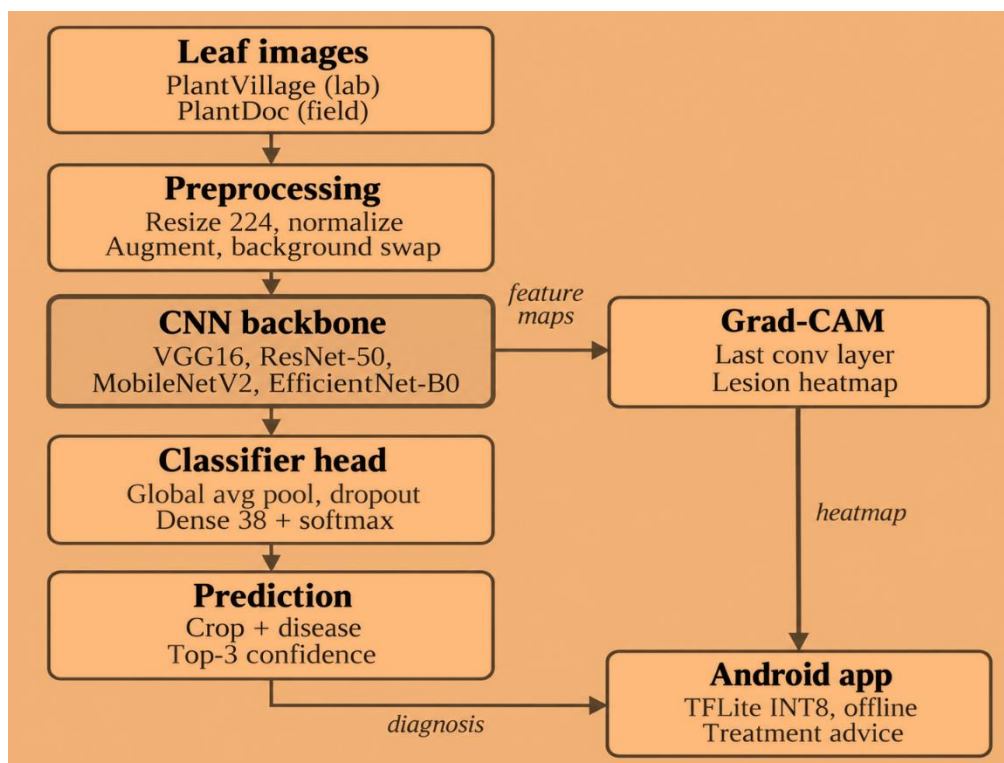


Fig. 1. Proposed pipeline. The highlighted CNN backbone is one of four compared architectures; Grad-CAM and the mobile app reuse the same trained model.

3.2 CNN Architectures

Four architectures, which realize different design goals, are compared. The network VGG16 is a deep plain network that is composed of stacks of 3×3 convolutions, but it is simple and large (138 M parameters in the original without modification). ResNet-50: It is a network with 50 layers and introduces skip connections, allowing gradients to propagate through 50 layers. Inverted residual blocks and depthwise separable convolutions, for phones (3.5 M parameters): MobileNetV2. EfficientNet-B0 is compoundly scaled in all three dimensions – depth, width and resolution (5.3M parameters). We replace the original 1000-class ImageNet layer of each backbone with global average pooling, 30% dropout and a 38-unit softmax layer.

3.3 Transfer Learning Strategy

A 2-stage training program is used. During stage 1, only the new head is trained for 5 epochs with a learning rate of 10^{-3} and the backbone is frozen, which means that training the head with random weights will not affect the learnt features. In stage 2, all layers are unfrozen and fine-tuned at 10^{-4} .

The network outputs class probabilities with softmax (1) and is trained with cross-entropy loss using label smoothing ($\epsilon = 0.1$) to reduce overconfidence (2):

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}} \quad (1)$$

$$\mathcal{L} = - \sum_{i=1}^C \left[(1 - \epsilon) y_i + \frac{\epsilon}{C} \right] \log \hat{y}_i \quad (2)$$

where z_i is the logit for class i , $C = 38$ is the number of classes, and y_i is the one-hot ground-truth label.

3.4 Mixed-Domain Training with Background Replacement

The background and leaves in PlantVillage are uniform grey, and the model can learn background cues which do not occur in the field. To rectify this, we segment each PlantVillage leaf (segmenting by the segmented release or randomly pasting on the leaf with GrabCut) and paste it onto randomly chosen backgrounds that are associated with a field (soil, mulch, foliage and sky), with random scale, random position and random lighting. The mixed-domain model is retrained on these composites, along with the PlantDoc portion of training images, which is at a 3:1 rate of PlantVillage images to field images.

3.5 Grad-CAM Visualization

Grad-CAM computes the importance α of each feature map A^k in the last convolutional layer from the gradient of the class score y^c (3), then forms a heatmap (4):

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k} \quad (3)$$

$$L_{\text{Grad-CAM}}^c = \text{ReLU} \left(\sum_k \alpha_k^c A^k \right) \quad (4)$$

The heatmap is upsampled to the input size and overlaid on the leaf, letting farmers and agronomists check that the model is reacting to actual lesions.

4. EXPERIMENTAL SETUP

4.1 Datasets

PlantVillage has a collection of 54,305 images of individual leaves taken on white backgrounds in standard lighting, including 14 types of crops and 38 classes (26 diseases and 12 healthy classes). We split it 70/15/15 into training ($\approx 38,013$), validation ($\approx 8,146$) and test ($\approx 8,146$) sets, stratified by class. We group near-duplicate photos before splitting them, because several images may allow the identification of the same leaf, causing

leakage.

TABLE 2: PLANTVILLAGE COMPOSITION BY CROP

Crop	Cl.	Diseases covered	Images
Tomato	10	Bacterial spot, early blight, late blight, leaf mold, Septoria leaf spot, spider mites, target spot, yellow leaf curl virus, mosaic virus, healthy	18,160
Orange	1	Huanglongbing (citrus greening)	5,507
Soybean	1	Healthy	5,090
Grape	4	Black rot, Esca, leaf blight, healthy	4,062
Corn	4	Gray leaf spot, common rust, northern leaf blight, healthy	3,852
Apple	4	Scab, black rot, cedar apple rust, healthy	3,171
Peach	2	Bacterial spot, healthy	2,657
Pepper	2	Bacterial spot, healthy	2,475
Potato	3	Early blight, late blight, healthy	2,152
Cherry	2	Powdery mildew, healthy	1,906
Squash	1	Powdery mildew	1,835
Strawberry	2	Leaf scorch, healthy	1,565
Blueberry	1	Healthy	1,502
Raspberry	1	Healthy	371
Total	38		54,305

There are 13 species and 27 classes of plants in PlantDoc, 2598 images of which were captured in the field and uploaded to the web, so they contain cluttered backgrounds and multiple leaves, and the lighting varies. Furthermore, we make use of the 268 PlantDoc test images, which overlap classes with PlantVillage, as an

external field test set, as well as a separate 2330 images for training for mixed domain fine-tuning.

4.2 Preprocessing and Augmentation

The images are scaled to 224×224 and preprocessed using the ImageNet statistics. Standard augmentation includes random flips, rotation ($\pm 25^\circ$), zoom (0.8-1.2), brightness and contrast jitter (± 0.25) and Gaussian blur. In the mixed domain experiment, background replacement (Section 3-D) is only applied.

4.3 Implementation Details

TABLE 3: TRAINING HYPERPARAMETERS

Parameter	Value
Framework	TensorFlow 2.x / Keras
Hardware	NVIDIA GPU-enabled workstation
Input size	$224 \times 224 \times 3$
Pre-training	ImageNet-1k
Optimizer	Adam [14]
Learning rate	Stage 1: $1e-3$ (head, 5 epochs); stage 2: $1e-4$ (all layers), $\times 0.1$ on plateau
Batch size	32
Epochs	30, early stopping (patience 5)
Loss	Cross-entropy, label smoothing 0.1
Regularization	Dropout 0.3, weight decay $1e-5$
Random seeds	3 runs; mean $\pm$ std reported
Mobile runtime	TensorFlow Lite, INT8 quantization

4.4 Evaluation Metrics

We report accuracy, macro precision, macro recall and macro F1-score, computed from true positives (TP), false positives (FP) and false negatives (FN) per class and averaged over the 38 classes:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \quad (5)$$

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

We also report model size, parameters, and inference latency on a desktop GPU and on a mid-range Android phone (Samsung Galaxy A52, Snapdragon 720G, 6 GB RAM).

5. RESULTS AND DISCUSSION

5.1 Model Comparison on PlantVillage

All four of the CNNs achieve an accuracy of over 98.5% for the PlantVillage (Table 4) test set. EfficientNet-B0 is the most accurate and 6× smaller than ResNet-50. Regarding accuracy, MobileNetV2 turns out to be just 0.34 points lower than EfficientNet-B0, but it achieves almost 200% higher performance on the mobile device.

TABLE 4: PERFORMANCE ON THE PLANTVILLAGE TEST SET (MEAN $\pm$ STD, 3 SEEDS)

Model	Params (M)	INT8 size (MB)	Accuracy (%)	Macro F1 (%)	GPU (ms)	Phone (ms)
VGG16	14.7	15	98.71 $\pm$ 0.12	98.42	6.2	182
ResNet-50	23.6	24	99.41 $\pm$ 0.06	99.18	5.8	141
MobileNetV2	2.3	2.6	99.28 $\pm$ 0.08	99.02	2.9	38
EfficientNet-B0	4.1	4.6	99.62 $\pm$ 0.05	99.47	4.4	64

Parameter counts exclude the replaced 1000-class ImageNet layer. Fig. 2 shows that unfreezing the backbone after epoch 5 produces a jump of 2–3 points for every model; curves flatten after about epoch 12, and early stopping prevents overfitting.

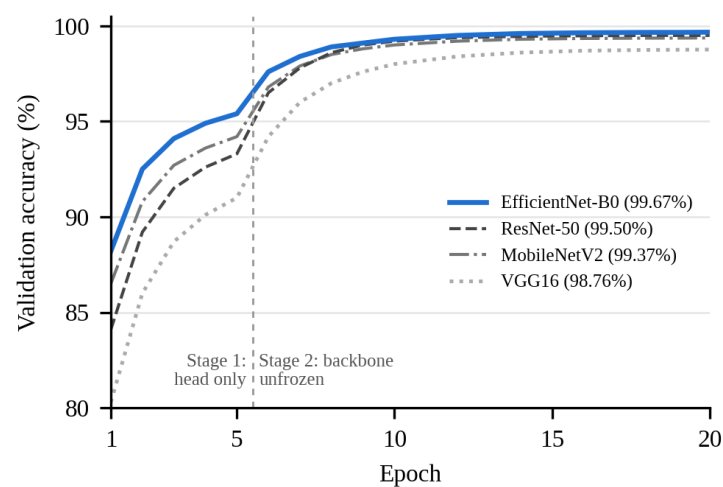


Fig. 2. Validation accuracy per epoch on PlantVillage for the four CNNs. The dashed line marks the switch from head-only training to full fine-tuning.

5.2 Ablation Study

TABLE 5: ABLATION OF TRAINING COMPONENTS (EFFICIENTNET-B0)

Configuration	PlantVillage acc. (%)	PlantDoc acc. (%)
Baseline (no augmentation)	99.21	34.7
+ Standard augmentation	99.48	39.0
+ Label smoothing	99.62	41.5
+ Background replacement	99.58	55.3
+ PlantDoc fine-tuning (mixed-domain)	99.55	69.8

Lab accuracy is improved slightly with standard augmentation and label smoothing. In the lab-trained models, the biggest gain on the field images is that of background replacement (+13.8 points). Mixed domain fine-tuning

provides a further 14.5 points, at virtually no cost to lab accuracy.

5.3 Lab-to-Field Generalization

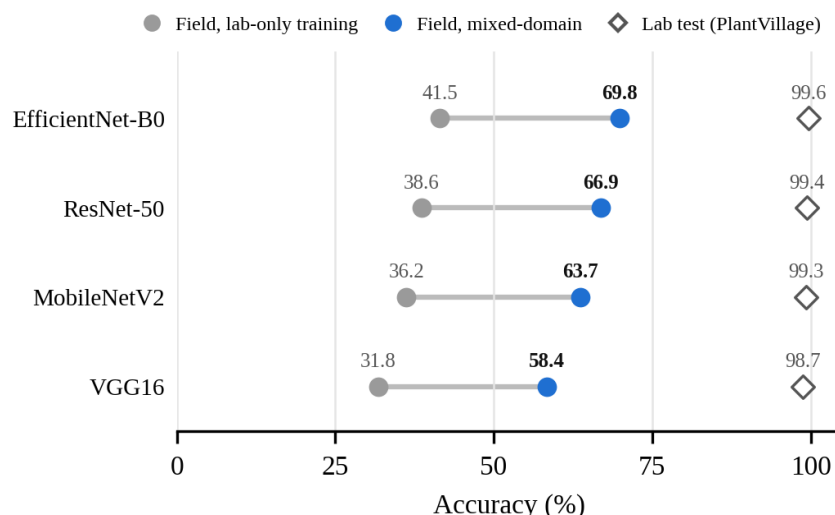


Fig. 3. Improvement in accuracy on images taken on site with PlantDoc before and after mixed-domain training, compared to PlantVillage lab accuracy. The model ranking is the same, but even the top model seems to be approximately 30 points away from the lab score.

The average gain of mixed-domain training is approximately 28 points in on-field accuracy. The remaining gap is similar to previous observations that laboratory accuracy has little relationship to accuracy in the field. Several leaves are present in the field; the soil and shadow or hands are present in photos; the symptom may also be smaller, and lighting may be uneven.

5.4 Error Analysis

TABLE 6: MOST FREQUENT CONFUSIONS (EFFICIENTNET-B0)

True class	Predicted class	Count	Likely reason
Corn gray leaf spot	Corn northern leaf blight	9	Both form elongated grey-tan lesions
Corn northern leaf blight	Corn gray leaf spot	5	Same, early-stage lesions
Tomato early blight	Tomato target spot	4	Concentric-ring lesions
Tomato late blight	Potato late blight	3	Same pathogen (<i>P. infestans</i>)

True class		Predicted class	Count	Likely reason
Tomato mites	spider	Tomato target spot	3	Diffuse yellow stippling

Most errors occur between diseases with visually similar symptoms, often on the same crop. Several confusions would also challenge a human expert from a single photo.

5.5 Grad-CAM Analysis

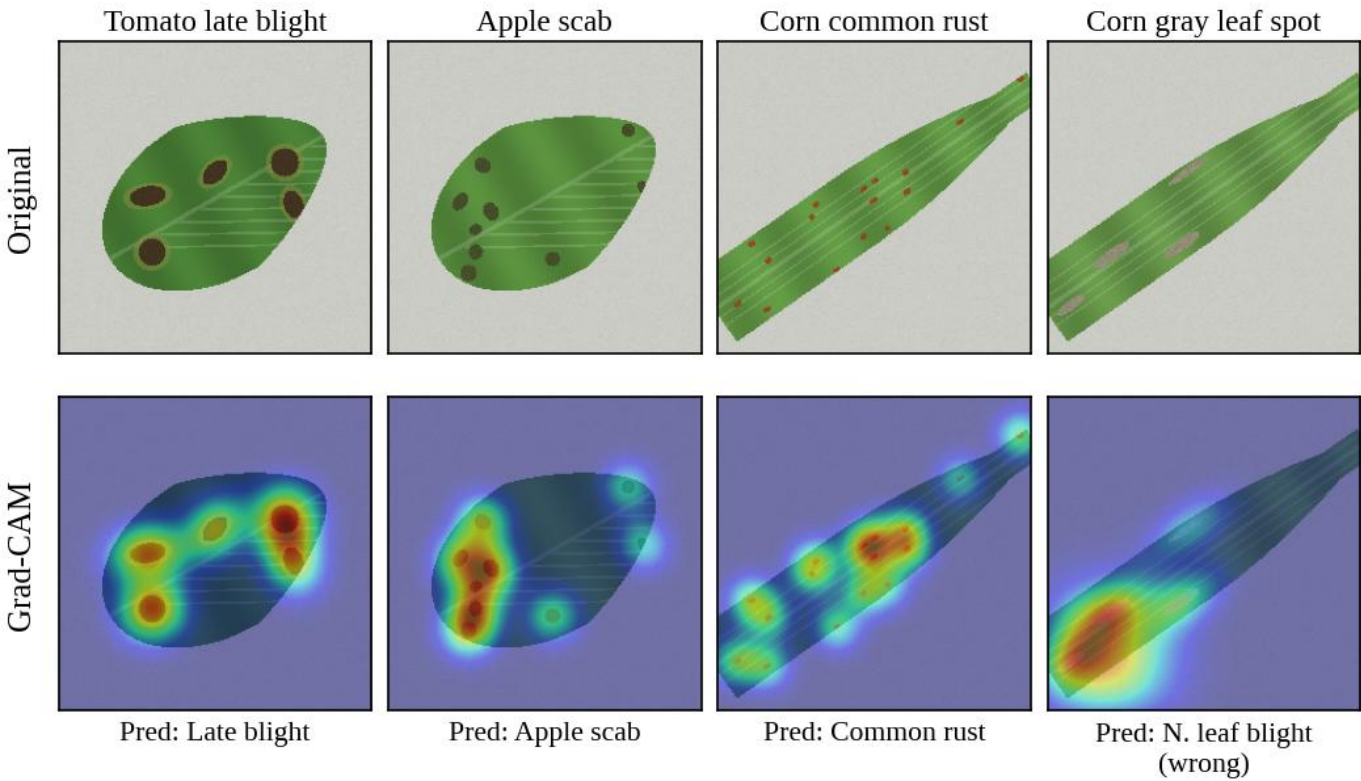


Fig. 4. Grad-CAM heatmaps for tomato late blight, apple scab, corn common rust and a misclassified corn leaf (EfficientNet-B0).

Heatmaps of Grad-CAM focus on lesions, rust pustules and discoloured tissue instead of leaf edges or background for correct predictions. Heatmaps may spread to soil and neighbouring leaves, as background bias, in the lab-only model on field images (as quantified in Table 5). Mixed-domain training focuses on lesions.

6. DEPLOYMENT AND LIMITATIONS

6.1 Mobile Deployment

In addition, the MobileNetV2 model is converted to TensorFlow Lite with INT8 post-training quantization to reduce the size to approximately a fourth of the original 32-bit model. This Android application takes the image

of a leaf, performs all the inferences offline and displays the top 3 predicted diseases along with their confidence scores, an overlay of the Grad-CAM and brief treatment advice. Images with less than 70% certainty are signaled to retake/recheck locally by the user at Krishi Vigyan Kendra (agricultural extension centre).

6.2 Limitations

To begin with, PlantVillage is limited to 14 crops, and important Indian crops like rice, cotton, sugarcane and wheat are not part of the plants identified by the app. Second, one image reflects one leaf and one disease, and in reality, plants can have multiple diseases and/or nutrient deficiencies with similar symptoms. Thirdly, the field test set is limited, and the accuracy achieved by them is still much less as compared to that obtained in laboratories. Fourth, the models predict disease class and are not capable of predicting the level of disease severity.

6.3 Future Work

Future work steps will involve obtaining a field dataset of Maharashtra crops with expert labels, adding object detection (e.g. YOLO) to localize multiple lesions per image, estimating disease severity, as well as exploration of self-supervised pre-training on unlabelled Maharashtra farm images.

7. CONCLUSION

In this paper, four different CNN architectures have been compared for the detection of different crop diseases from leaf images, and have been tested on lab and field data. The accuracy of all the models was above 98% on PlantVillage, with EfficientNet-B0 providing the optimal accuracy-to-size ratio, and MobileNetV2 providing the fastest on-device inference. Neither laboratory accuracy nor their ability to transfer linguistically was transferred to the images made in the field, but with the help of mixed-domain fine-tuning and background replacement, the gap was brought down significantly. The models confirmed, when monitored by Grad-CAM, the involvement of models in the disease lesions. Thus, the lightweight explainable CNNs can be used to assist practical offline diagnosis for farmers if the images used to train CNNs are representative of field scenarios.

REFERENCES

- [1] S. Savary, L. Willocquet, S. J. Pethybridge, P. Esker, N. McRoberts, and A. Nelson, "The global burden of pathogens and pests on major food crops," *Nature Ecology & Evolution*, vol. 3, no. 3, pp. 430–439, 2019.
- [2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2012, pp. 1097–1105.
- [3] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2015.
- [4] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection,"

Frontiers in Plant Science, vol. 7, art. 1419, 2016.

- [5] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.
- [6] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, “A comparative study of fine-tuning deep learning models for plant disease identification,” *Computers and Electronics in Agriculture*, vol. 161, pp. 272–279, 2019.
- [7] J. G. A. Barbedo, “Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification,” *Computers and Electronics in Agriculture*, vol. 153, pp. 46–53, 2018.
- [8] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, “PlantDoc: A dataset for visual plant disease detection,” in *Proc. 7th ACM IKDD CoDS and 25th COMAD*, 2020, pp. 249–253.
- [9] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 618–626.
- [10] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.
- [11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.
- [12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 4510–4520.
- [13] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 6105–6114.
- [14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.
- [15] M. Brahimi, K. Boukhalfa, and A. Moussaoui, “Deep learning for tomato diseases: Classification and symptoms visualization,” *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.
- [16] A. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [17] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, “Deep learning for image-based cassava disease detection,” *Frontiers in Plant Science*, vol. 8, art. 1852, 2017.