Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Leakage-audited cross-dataset evaluation reveals large generalization gaps in deep learning models for apple and maize disease recognition

Yogesh .

yogesh-aiml@dsu.edu.in

Dayananda Sagar University

P Sudir

Dayananda Sagar University

Princy Randhawa

Dayananda Sagar University

Muzameel Ahmed

Dayananda Sagar Academy of Technology and Management

B M Chandrakala

Dayananda Sagar College of Engineering

Article

Keywords: plant disease recognition, cross-dataset evaluation, domain shift, data leakage, deep learning, model evaluation

Posted Date: November 6th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11242039/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Leakage-audited cross-dataset evaluation reveals large generalization gaps in deep learning models for apple and maize disease recognition

Yogesh[1,*], Sudir P[1], Princy Randhawa[1], Muzameel Ahmed[2] & Chandrakala B M[3]

[1]Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning), School of Engineering, Dayananda Sagar University, Harohalli Campus, Bengaluru 562112, Karnataka, India. [2]Department of Information Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bengaluru, Karnataka, India. [3]Department of Information Science and Engineering, Dayananda Sagar College of Engineering, Bengaluru, Karnataka, India.

*Corresponding author: yogesh-aiml@dsu.edu.in

Abstract

Deep learning models for plant disease recognition often reach near-perfect accuracy on curated benchmark images, yet how well this performance carries over to independently collected images is rarely measured directly. We trained five image-classification networks (ResNet50, EfficientNet-B0, DenseNet121, ConvNeXt-Tiny and ViT-B/16) on PlantVillage apple and maize leaf images under one shared protocol and tested each trained model, without adaptation, on independently sourced images: 1,730 apple images from Plant Pathology 2020 and 371 web-sourced maize images. Before training, an exact-file audit removed seven duplicate PlantVillage images and found no exact or perceptual near-duplicates shared between source and external datasets. On held-out PlantVillage images, Macro-F1 ranged from 0.949 to 1.000. On external images it fell to 0.472–0.723 for apple and 0.209–0.375 for maize. Exact McNemar tests with Holm correction identified significant differences in per-image correctness for several model pairs. Because each architecture was trained once with one seed, architecture-specific comparisons are treated as exploratory. The results show that near-perfect benchmark scores should not be interpreted as evidence of field readiness; independent external testing and leakage auditing are essential.

Keywords: plant disease recognition; cross-dataset evaluation; domain shift; data leakage; deep learning; model evaluation

Introduction

Plant diseases reduce the yield and quality of staple and horticultural crops and remain a persistent threat to food security[1,2]. Early and accurate diagnosis supports timely management, but expert scouting is slow and difficult to scale. Image-based recognition with deep learning has therefore attracted sustained interest as a basis for low-cost decision support[3,4], and convolutional networks are now routinely applied to leaf and fruit images for disease and nutrient-disorder assessment[5].

Much of this progress has been measured on PlantVillage, a public collection of leaf images photographed under controlled conditions against uniform backgrounds[6]. Mohanty et al. trained a convolutional network on 54,306 PlantVillage images covering 14 crops and 26 diseases and reported 99.35% accuracy on a held-out test set[7]. Later studies reported similarly high accuracies with larger databases and newer architectures[8]. These results show what networks can learn from curated images, but a random hold-out drawn from the same collection shares its acquisition protocol, backgrounds and photographers with the training data. It therefore says little about images taken in orchards and fields, where illumination, background clutter, viewpoint, symptom stage and image quality vary widely[9,10].

Two problems make benchmark accuracy difficult to interpret. The first is dataset bias. Classifiers can exploit regularities that are specific to how a dataset was assembled rather than to the target concept[11,12]. For PlantVillage, Noyan showed that a model trained on only eight background pixels per image reached 49.0% accuracy, far above the 2.6% expected by chance, which indicates that non-leaf cues are correlated with disease labels[13]. The second problem is leakage. Exact or near-duplicate images that fall on both sides of an evaluation boundary inflate measured performance[14,15], and near-duplicate contamination has been documented even in standard computer-vision benchmarks[16]. When a model is tested on a second dataset, overlap between the two collections cannot be excluded without checking.

Independently collected datasets now make external testing possible. The Plant Pathology 2020 challenge released expert-annotated apple foliar disease images captured under variable illumination, angles, surfaces and noise[17], and PlantDoc assembled 2,598 images of 13 plant species from internet sources to provide non-laboratory imagery[18]. Cross-dataset studies have begun to report how performance changes outside the training distribution; for example, Mahapatra et al. reported 99.38% accuracy for single-dataset training and 82.93% in cross-dataset evaluation for corn and potato disease classification with an attention-based network[19]. However, cross-dataset results are seldom accompanied by an explicit overlap audit, a comparison of several architecture families under an identical protocol, uncertainty intervals and paired statistical tests on the same external images.

This study asks four questions. How much does performance fall when networks trained on PlantVillage are applied, without adaptation, to independently sourced apple and maize images? Can that fall be attributed to exact or near-duplicate overlap between datasets? Does the size of the fall depend on architecture, comparing four convolutional networks and one vision transformer? And are the differences between architectures on external images statistically supported once multiple comparisons are controlled? To answer them, we combined SHA-256 and perceptual-hash auditing with a frozen data split, a single training protocol for all models, bootstrap confidence intervals, and exact McNemar tests with Holm correction (Fig. 1). The Results are presented next, followed by the Discussion; full details of data handling, training and statistical analysis are given in the Methods.

Results

Dataset composition and leakage audit

The final analysis comprised two three-class tasks (Table 1). For apple, the source data were PlantVillage images of apple rust, apple scab and healthy leaves, and the external data were single-label images from Plant Pathology 2020. For maize, the source data were PlantVillage images of common rust, gray leaf spot and leaf blight, and the external data were 371 single-label maize images from a YOLO-formatted, PlantDoc-derived Kaggle distribution. Before duplicate removal the combined inventory contained 7,341 images.

SHA-256 hashing identified seven groups of byte-identical files. All seven groups lay within PlantVillage healthy apple images; none crossed a dataset boundary. Removing the redundant copies reduced the PlantVillage healthy apple class from 1,645 to 1,638 images and the full inventory to 7,334 images. Perceptual hashes were then compared across datasets within each harmonized crop–class pair. No pair of images from different datasets had a Hamming distance of 8 bits or less. Under these criteria, therefore, no exact or near-duplicate image was shared between any source and external set.

The deduplicated PlantVillage data were split once, stratified by class, into training, validation and internal-test partitions (70:15:15). This gave 1,780, 381 and 382 apple images and 1,883, 403 and 404 maize images, respectively. The same partitions were used for all five models, and no external image was used for training, early stopping or checkpoint selection.

Internal performance

All five networks performed almost perfectly on the internal PlantVillage test sets (Table 2; Supplementary Table S1). For apple, ResNet50, DenseNet121 and ViT-B/16 classified all 382 internal-test images correctly (Macro-F1 = 1.0000), and EfficientNet-B0 and ConvNeXt-Tiny

each reached a Macro-F1 of 0.9976. Maize internal Macro-F1 ranged from 0.9491 (DenseNet121) to 0.9671 (ViT-B/16), with internal Matthews correlation coefficients (MCC) between 0.9393 and 0.9609. Validation Macro-F1 at the selected checkpoint was 1.0000 for every apple model and between 0.9678 and 0.9800 for the maize models (Table 4).

External performance and generalization gap

External performance was much lower for every model and both crops (Fig. 2; Table 2). For apple, external Macro-F1 ranged from 0.4722 (EfficientNet-B0) to 0.7228 (ConvNeXt-Tiny), and external accuracy from 0.5231 to 0.7341. The generalization gap, defined as internal minus external Macro-F1 (equation (3)), ranged from 0.2748 for ConvNeXt-Tiny to 0.5253 for EfficientNet-B0.

The fall was larger for maize. External Macro-F1 ranged from 0.2092 (EfficientNet-B0) to 0.3746 (ConvNeXt-Tiny), and the corresponding gaps ranged from 0.5922 to 0.7556. External maize accuracy (0.2668–0.3989) was below the 0.5121 that would be obtained by always predicting the most frequent external class, leaf blight (190 of 371 images). External balanced accuracy for ResNet50 (0.3447), DenseNet121 (0.3453) and EfficientNet-B0 (0.3086) was close to or below the value of 1/3 expected from uninformative predictions over three classes, and the external MCC of these three models was −0.0185, −0.0028 and −0.0679, respectively. ConvNeXt-Tiny (MCC 0.1850) and ViT-B/16 (MCC 0.1048) retained weak positive agreement. For apple, by contrast, all models exceeded the majority-class accuracy of 0.3595 (rust, 622 of 1,730 images), and external MCC ranged from 0.3524 to 0.6318.

External macro-precision exceeded macro-recall for every model except EfficientNet-B0 on maize (Supplementary Table S1).

Uncertainty of external estimates

Percentile bootstrap intervals (2,000 resamples of the external images) are shown as error bars in Fig. 2 and listed in Table 2. Apple intervals were narrow (width 0.043–0.049) because the external set was large. The ConvNeXt-Tiny interval (0.7000–0.7434) did not overlap with those of any other apple model, and the ViT-B/16 interval (0.5830–0.6300) lay above those of ResNet50, EfficientNet-B0 and DenseNet121. Maize intervals were wider (0.068–0.099) owing to the smaller external set and overlapped extensively among models. In both crops, every external interval lay far below the corresponding internal Macro-F1. These intervals describe sampling uncertainty in the external test set for a fixed trained model; they do not include variation from retraining with different random seeds.

Paired comparison of models on external images

Because all models were tested on the same external images, pairs of models were compared with exact two-sided McNemar tests on per-image correctness, with Holm correction applied to the ten comparisons within each crop (Table 3). For apple, seven of ten comparisons remained significant after correction. ConvNeXt-Tiny differed from each of the other four models (Holm-adjusted $P \leq 1.27 \times 10^{-11}$), and ViT-B/16 differed from ResNet50, EfficientNet-B0 and DenseNet121 (adjusted $P \leq 1.84 \times 10^{-12}$). The three remaining convolutional networks did not differ from one another (ResNet50 vs EfficientNet-B0, adjusted $P = 0.0875$; ResNet50 vs DenseNet121 and EfficientNet-B0 vs DenseNet121, adjusted $P = 0.436$).

For maize, five of ten comparisons remained significant: ConvNeXt-Tiny vs ResNet50 (adjusted $P = 0.0148$), ConvNeXt-Tiny vs EfficientNet-B0 (adjusted $P = 3.48 \times 10^{-6}$), ConvNeXt-Tiny vs DenseNet121 (adjusted $P = 0.00105$), ViT-B/16 vs EfficientNet-B0 (adjusted $P = 3.39 \times 10^{-4}$) and ViT-B/16 vs DenseNet121 (adjusted $P = 0.0148$). ConvNeXt-Tiny and ViT-B/16 did not differ (adjusted $P = 0.735$), nor did ResNet50, EfficientNet-B0 and DenseNet121 among themselves (adjusted $P \geq 0.124$) or ResNet50 and ViT-B/16 (adjusted $P = 0.169$). The McNemar test addresses per-image correctness and is not a test of differences in Macro-F1.

Discussion

The central finding is a large and consistent gap between internal and external performance. Five networks from different design families reached internal Macro-F1 between 0.949 and 1.000 on PlantVillage, yet the same checkpoints reached 0.472–0.723 on independently collected apple images and 0.209–0.375 on web-sourced maize images. Internal scores were also nearly identical across architectures and could not separate them, whereas external scores spread over a range of 0.25 for apple. The ranking that matters for deployment was visible only on external data.

The leakage audit constrains how this gap can be explained. Exact duplicates were present, but only within PlantVillage, and they were removed before splitting. No byte-identical file and no perceptual near-duplicate at a Hamming distance of 8 bits or less was shared between source and external datasets. The degradation therefore cannot be attributed to images being reused across the evaluation boundary under these criteria. The audit has a defined scope. A 64-bit perceptual hash detects resized, recompressed or lightly edited copies, not different photographs of the same leaf or plant, and the cross-dataset search was restricted to images with the same harmonized label. The audit also did not search for near-duplicates within PlantVillage across the training and internal-test partitions; PlantVillage contains multiple photographs taken under the same protocol, and any such within-dataset similarity would make

the internal estimates optimistic. This possibility strengthens, rather than weakens, the conclusion that internal PlantVillage accuracy overstates transferable performance.

The more plausible explanation is a shift in acquisition conditions between source and target domains. PlantVillage images show detached or isolated leaves against uniform backgrounds[6], whereas Plant Pathology 2020 images were taken in orchards with variable illumination, angles, surfaces and noise[17], and PlantDoc-style images were collected from the internet with heterogeneous backgrounds, resolutions and framings[18]. Networks trained on curated images can exploit cues that are predictive within the curated collection but absent elsewhere[12], and PlantVillage backgrounds alone carry label information[13]. The present experiment shows the size of the consequence but does not isolate its cause; doing so would require controlled interventions such as background replacement or leaf segmentation.

The maize results showed the largest loss of external performance. Three convolutional networks produced external MCC values at or below zero, and all five models had external accuracies below the majority-class rate. Two factors specific to this task require caution. First, the external set is small (371 images) and imbalanced, which widens uncertainty intervals. Second, class semantics may not align exactly: PlantVillage labels Northern Leaf Blight specifically, whereas the external distribution uses the broader label “Corn leaf blight”. The final analysis therefore used the neutral harmonized label leaf blight rather than claiming biological equivalence. In addition, file names in the external distribution include terms such as “Southern_Corn_Leaf_Blight” and “corn_southern_rust”, and the 371 retained images were not screened for such indicators, so the external rust and blight classes may include aetiologically different diseases. The maize findings should therefore be interpreted as evidence of poor transfer under the operational label mapping rather than as a disease-specific diagnostic estimate.

In the single seeded training run reported here, the observed external Macro-F1 values differed across architectures, with ConvNeXt-Tiny yielding 0.7228 for apple and 0.3746 for maize and ViT-B/16 yielding 0.6071 and 0.3519, respectively. These observations should not be interpreted as an architecture ranking because each model was trained only once. The paired McNemar tests establish differences in correctness patterns for the particular trained checkpoints and test images, not superiority that would necessarily persist across retraining. Multi-seed training would be required to estimate training-induced variability and to support architecture-level ranking claims. The robust conclusion is narrower: every architecture experienced a substantial internal-to-external performance loss, and changing the architecture reduced but did not eliminate the generalization gap.

Cross-dataset studies of plant disease classification have reported declines when the test distribution departs from the training distribution[19]. Our study adds explicit evidence that such

a decline persists after an image-overlap audit, holds across five architecture families trained under one protocol, and is accompanied by interval estimates and paired tests computed on identical external images. Similar transfer issues are likely to affect other agricultural vision tasks that are trained on curated collections, including nutrient-disorder detection in fruit crops[20], and the same evaluation practices would apply there. Distribution shift is a general problem in machine learning, and methods for domain adaptation and domain generalization exist[21]. The present study deliberately applied none of them, so that the gap could be measured directly.

Several limitations apply. Only two crops and three classes per crop were studied, and multi-disease Plant Pathology images were excluded. The maize external set was small and its provenance is a third-party Kaggle redistribution with PlantDoc-style labels. The retained maize images were not screened at file-name level for disease names inconsistent with the operational class mapping. Each model was trained once with a single seed, so the reported bootstrap intervals quantify external test-set sampling uncertainty for a fixed checkpoint and do not capture training variability. Consequently, architecture-level ordering is exploratory; repeating training with at least three independent seeds would materially strengthen any comparative claim. No target-domain data were used for adaptation or calibration, and per-class performance and calibration were not analysed in the main text. Finally, perceptual hashing at a single threshold cannot rule out all forms of dependence between datasets.

These limitations point to concrete steps. Plant disease recognition studies should report performance on at least one independently collected test set alongside random-split results, audit exact and near-duplicate overlap before interpreting either, use class-balanced metrics such as Macro-F1, balanced accuracy and MCC instead of accuracy alone, give confidence intervals and paired tests, and repeat training over several seeds. Benchmark accuracy on curated images remains a useful development check, but it should not be presented as evidence that a model is ready for use in the field.

Methods

Study design

The study had four stages (Fig. 1): dataset harmonization and leakage auditing; training and internal evaluation on PlantVillage; external evaluation on independent datasets without any adaptation; and statistical analysis. All stages were run as a single scripted pipeline with a fixed random seed of 42. No human participants, animals or human-derived material were involved, and no ethical approval was required.

Datasets and class harmonization

Source domain. PlantVillage[6,7] was obtained from a Kaggle redistribution, and only the colour images were used; greyscale and segmented versions were excluded. Six folders were mapped to canonical labels: `Apple___Cedar_apple_rust` to apple rust, `Apple___Apple_scab` to apple scab, `Apple___healthy` to healthy, `Corn_(maize)___Common_rust_` to common rust, `Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot` to gray leaf spot and `Corn_(maize)___Northern_Leaf_Blight` to leaf blight. Before deduplication these classes contained 275, 630, 1,645, 1,192, 513 and 985 images, respectively.

External apple domain. Plant Pathology 2020 (FGVC7)[17] training images were labelled from the competition's annotation file. Images labelled healthy, scab or rust were mapped to healthy, apple scab and apple rust; the rust class corresponds to cedar apple rust. Images annotated as having multiple diseases were excluded to keep the label space identical to the source task, leaving 1,730 images (622 rust, 592 scab and 516 healthy).

External maize domain. The maize images came from a Kaggle distribution (andresmgs/plantdec) that provides PlantDoc-style class names[18] with YOLO bounding-box labels in train, validation and test folders. All three folders were pooled because none of these images was used for training. Three labels were mapped: "Corn rust leaf" to common rust, "Corn Gray leaf spot" to gray leaf spot and "Corn leaf blight" to leaf blight. An image was retained only if its label file contained exactly one of these three class identifiers and no identifier of any other class. This gave 371 images (116 common rust, 65 gray leaf spot and 190 leaf blight). The broader external term "Corn leaf blight" was harmonized with PlantVillage's Northern Leaf Blight under the neutral label leaf blight; this is a computational harmonization and does not establish biological equivalence.

Exact-duplicate audit

A SHA-256 digest[22] was computed over the raw bytes of every image file, giving a hash $H_i = \mathrm{SHA256}(I_i)$ for image I_i. Two files were treated as exact duplicates if $H_i = H_j$. Duplicate groups were recorded together with the datasets in which they occurred. Within PlantVillage, the first file in each group was kept and the other copies were removed before data splitting. A cross-dataset group would have been reported as leakage; none occurred.

Perceptual near-duplicate audit

For every retained image, a 64-bit DCT-based perceptual hash p_i was computed with the ImageHash library (default hash size). The distance between two images was the Hamming distance

$$d_H(p_i, p_j) = \sum_{k=1}^{64} \mathbb{1}\left[p_{ik} \neq p_{jk}\right], \qquad (1)$$

where $\mathbb{1}[\cdot]$ is the indicator function and p_{ik} is the kth bit of p_i. A pair of images from different datasets with the same harmonized crop and class was flagged as a candidate near-duplicate if $d_H \leq 8$. Pairs from the same dataset were not compared.

Data partitioning

For each crop, the deduplicated PlantVillage images were split with stratified random sampling (scikit-learn `train_test_split`, random state 42) into 70% training and 30% held-out data, and the held-out portion was then split equally, again stratified, into validation and internal-test sets. The partitions were saved and reused unchanged for all architectures. External images were used only for final evaluation.

Preprocessing and augmentation

All images were converted to RGB and resized to 224 × 224 pixels. During training, random horizontal flipping (probability 0.5), random rotation of up to ±10° and colour jitter (brightness, contrast and saturation factors of 0.1) were applied. Validation, internal-test and external images were only resized. All inputs were normalized with the ImageNet channel means (0.485, 0.456, 0.406) and standard deviations (0.229, 0.224, 0.225) for every architecture, which matches the preprocessing of the torchvision pretrained weights used.

Architectures and training

Five ImageNet-pretrained[23] architectures from torchvision were fine-tuned: ResNet50[24], EfficientNet-B0[25], DenseNet121[26], ConvNeXt-Tiny[27] and ViT-B/16[28]. In each, the final classification layer was replaced with a randomly initialized linear layer with three outputs, and all parameters were updated during training. Training used cross-entropy loss and the AdamW optimizer[29] with a learning rate of 1×10^{-4}, weight decay of 1×10^{-4} and batch size 16 for a maximum of ten epochs (Table 4). Validation Macro-F1 was computed after every epoch; training stopped when it had not improved for three consecutive epochs, and the weights from the epoch with the highest validation Macro-F1 were restored. Per-epoch training loss and validation Macro-F1 for every run are listed in Supplementary Table S2. The random seed was reset to 42 before each model–crop run. No class weighting, resampling or learning-rate schedule was used, and no hyperparameter was tuned on external data.

Evaluation metrics

For class k of $K = 3$ classes, with true positives TP_k, false positives FP_k and false negatives FN_k, precision $P_k = TP_k/(TP_k + FP_k)$, recall $R_k = TP_k/(TP_k + FN_k)$ and $F1_k = 2P_kR_k/(P_k + R_k)$. Macro-F1 was the primary metric because it weights each class equally regardless of prevalence:

$$\text{Macro-F1} = \frac{1}{K}\sum_{k=1}^{K} F\,1_k. \qquad (2)$$

Balanced accuracy, the unweighted mean of per-class recall $\frac{1}{K}\sum_k R_k$, was also reported[30]; for the multiclass case it equals macro-recall. Accuracy and the multiclass Matthews correlation coefficient, as implemented in scikit-learn, were also computed[31]. For model m, the generalization gap was defined as

$$G_m = \text{Macro-F1}_{\text{internal},m} - \text{Macro-F1}_{\text{external},m}. \qquad (3)$$

Majority-class accuracy and the chance-level balanced accuracy of 1/3 are given in the Results only as reference values and were derived from the class counts.

Bootstrap confidence intervals

For each model and crop, the N external images were resampled with replacement $B = 2{,}000$ times (NumPy generator, seed 42), and Macro-F1 was recomputed for each resample from the fixed per-image predictions. The 2.5th and 97.5th percentiles of the bootstrap distribution formed the 95% percentile confidence interval[32]. Intervals were computed for external Macro-F1 only.

Paired statistical comparison

For each pair of models A and B evaluated on the same n external images, let b be the number of images classified correctly by A and incorrectly by B, and c the reverse. Under the null hypothesis that both models have the same probability of being correct, $b \sim \text{Binomial}(b + c, 0.5)$ conditional on $b + c$. The exact two-sided McNemar test[33,34] was computed as a two-sided binomial test of $\min(b, c)$ with SciPy's `binomtest`[35]:

$$P = \min\left\{1,\ 2\sum_{i=0}^{\min(b,c)} \binom{b+c}{i} 0.5^{\,b+c}\right\}. \qquad (4)$$

The exact form was chosen because it makes no large-sample approximation. The ten pairwise comparisons within each crop were corrected with the Holm step-down procedure[36] (statsmodels `multipletests`[37]) at a family-wise α of 0.05. The tests are two-sided. The P

values in Table 3 were recomputed from the reported discordant counts with the same test; they reproduce the unadjusted and Holm-adjusted values from the pipeline and are given in full precision where the pipeline output had been rounded to zero.

Software and computing environment

Experiments were run in a Kaggle notebook with a CUDA-enabled GPU using Python, PyTorch[38], torchvision, scikit-learn[39], SciPy[35], statsmodels[37], ImageHash, NumPy, pandas and Matplotlib. The pretrained weight files were resnet50-11ad3fa6, efficientnet_b0_rwightman-7f5810bc, densenet121-a639ec97, convnext_tiny-983f1562 and vit_b_16-c867db91.

Use of large language models

Generative artificial intelligence tools (OpenAI ChatGPT and Anthropic Claude) were used to assist with manuscript organization, language editing, code review and numerical consistency checking. The authors designed and executed all analyses, verified the data, code, statistical results, references and final text, and retain full responsibility for the accuracy, originality and integrity of the work. No generative-AI output was treated as experimental evidence or substituted for author judgement.

Data availability

All datasets analysed are publicly available. PlantVillage (colour images) was obtained from https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset, Plant Pathology 2020 from https://www.kaggle.com/c/plant-pathology-2020-fgvc7, and the external maize images from https://www.kaggle.com/datasets/andresmgs/plantdec. The image inventories with SHA-256 and perceptual hashes, duplicate-audit outputs, frozen split files, per-image external predictions for all models, metric tables, bootstrap and McNemar results, and confusion matrices generated in this study are available from the corresponding author on reasonable request.

Code availability

The code used for dataset harmonization, duplicate auditing, training, evaluation and statistical testing is available from the corresponding author on reasonable request.

References

1. Strange, R. N. & Scott, P. R. Plant disease: a threat to global food security. *Annu. Rev. Phytopathol.* **43**, 83–116 (2005). https://doi.org/10.1146/annurev.phyto.43.113004.133839

2. Savary, S. et al. The global burden of pathogens and pests on major food crops. *Nat. Ecol. Evol.* **3**, 430–439 (2019). https://doi.org/10.1038/s41559-018-0793-y

3. Kamilaris, A. & Prenafeta-Boldú, F. X. Deep learning in agriculture: a survey. *Comput. Electron. Agric.* **147**, 70–90 (2018). https://doi.org/10.1016/j.compag.2018.02.016

4. Liu, J. & Wang, X. Plant diseases and pests detection based on deep learning: a review. *Plant Methods* **17**, 22 (2021). https://doi.org/10.1186/s13007-021-00722-9

5. Kumar, Y., Dubey, A. K., Arora, R. R. & Rocha, A. Multiclass classification of nutrients deficiency of apple using deep neural network. *Neural Comput. Appl.* **34**, 8411–8422 (2022). https://doi.org/10.1007/s00521-020-05310-x

6. Hughes, D. P. & Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. Preprint at https://arxiv.org/abs/1511.08060 (2015).

7. Mohanty, S. P., Hughes, D. P. & Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **7**, 1419 (2016). https://doi.org/10.3389/fpls.2016.01419

8. Ferentinos, K. P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **145**, 311–318 (2018). https://doi.org/10.1016/j.compag.2018.01.009

9. Barbedo, J. G. A. Factors influencing the use of deep learning for plant disease recognition. *Biosyst. Eng.* **172**, 84–91 (2018). https://doi.org/10.1016/j.biosystemseng.2018.05.013

10. Barbedo, J. G. A. Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification. *Comput. Electron. Agric.* **153**, 46–53 (2018). https://doi.org/10.1016/j.compag.2018.08.013

11. Torralba, A. & Efros, A. A. Unbiased look at dataset bias. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 1521–1528 (2011). https://doi.org/10.1109/CVPR.2011.5995347

12. Geirhos, R. et al. Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2**, 665–673 (2020). https://doi.org/10.1038/s42256-020-00257-z

13. Noyan, M. A. Uncovering bias in the PlantVillage dataset. Preprint at https://arxiv.org/abs/2206.04374 (2022).

14. Kaufman, S., Rosset, S., Perlich, C. & Stitelman, O. Leakage in data mining. *ACM Trans. Knowl. Discov. Data* **6**, 1–21 (2012). https://doi.org/10.1145/2382577.2382579

15. Kapoor, S. & Narayanan, A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns* **4**, 100804 (2023). https://doi.org/10.1016/j.patter.2023.100804

16. Barz, B. & Denzler, J. Do we train on test data? Purging CIFAR of near-duplicates. *J. Imaging* **6**, 41 (2020). https://doi.org/10.3390/jimaging6060041

17. Thapa, R., Zhang, K., Snavely, N., Belongie, S. & Khan, A. The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. *Appl. Plant Sci.* **8**, e11390 (2020). https://doi.org/10.1002/aps3.11390

18. Singh, D. et al. PlantDoc: a dataset for visual plant disease detection. In *Proc. 7th ACM IKDD CoDS and 25th COMAD* 249–253 (2020). https://doi.org/10.1145/3371158.3371196

19. Mahapatra, P., Panda, M., Dash, S. K. & Sahu, U. K. Advancing plant disease classification using an attention-based CNN for intra-dataset and cross-dataset training. *Sci. Rep.* **16**, 10925 (2026). https://doi.org/10.1038/s41598-026-45464-7

20. Yogesh, Dubey, A. K. & Rocha, A. A non-invasive approach for calcium deficiency detection in pears using machine learning. *Neural Comput. Appl.* **37**, 7609–7618 (2025). https://doi.org/10.1007/s00521-023-08444-w

21. Zhou, K., Liu, Z., Qiao, Y., Xiang, T. & Loy, C. C. Domain generalization: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 4396–4415 (2023). https://doi.org/10.1109/TPAMI.2022.3195549

22. National Institute of Standards and Technology. *Secure Hash Standard (SHS)*. FIPS PUB 180-4 (U.S. Department of Commerce, 2015). https://doi.org/10.6028/NIST.FIPS.180-4

23. Deng, J. et al. ImageNet: a large-scale hierarchical image database. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848

24. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90

25. Tan, M. & Le, Q. V. EfficientNet: rethinking model scaling for convolutional neural networks. In *Proc. 36th International Conference on Machine Learning*, PMLR **97**, 6105–6114 (2019).

26. Huang, G., Liu, Z., van der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 2261–2269 (2017). https://doi.org/10.1109/CVPR.2017.243

27. Liu, Z. et al. A ConvNet for the 2020s. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 11966–11976 (2022). https://doi.org/10.1109/CVPR52688.2022.01167

28. Dosovitskiy, A. et al. An image is worth 16x16 words: transformers for image recognition at scale. In *International Conference on Learning Representations* (2021).

29. Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations* (2019).

30. Brodersen, K. H., Ong, C. S., Stephan, K. E. & Buhmann, J. M. The balanced accuracy and its posterior distribution. In *Proc. 20th International Conference on Pattern Recognition* 3121–3124 (2010). https://doi.org/10.1109/ICPR.2010.764

31. Chicco, D. & Jurman, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics* **21**, 6 (2020). https://doi.org/10.1186/s12864-019-6413-7

32. Efron, B. Bootstrap methods: another look at the jackknife. *Ann. Stat.* **7**, 1–26 (1979). https://doi.org/10.1214/aos/1176344552

33. McNemar, Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* **12**, 153–157 (1947). https://doi.org/10.1007/BF02295996

34. Dietterich, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput.* **10**, 1895–1923 (1998). https://doi.org/10.1162/089976698300017197

35. Virtanen, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020). https://doi.org/10.1038/s41592-019-0686-2

36. Holm, S. A simple sequentially rejective multiple test procedure. *Scand. J. Stat.* **6**, 65–70 (1979).

37. Seabold, S. & Perktold, J. Statsmodels: econometric and statistical modeling with Python. In *Proc. 9th Python in Science Conference* 92–96 (2010). https://doi.org/10.25080/Majora-92bf1922-011

38. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* **32** (2019).

39. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

Author contributions

Y. conceived the study, developed the methodology and software, performed the formal analysis and visualization, and wrote the original draft. S.P., P.R., M.A. and C.B.M. reviewed and edited the manuscript. All authors reviewed and approved the final manuscript.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Additional information

Competing interests: The authors declare no competing interests.

Supplementary information accompanies this paper.

Figures

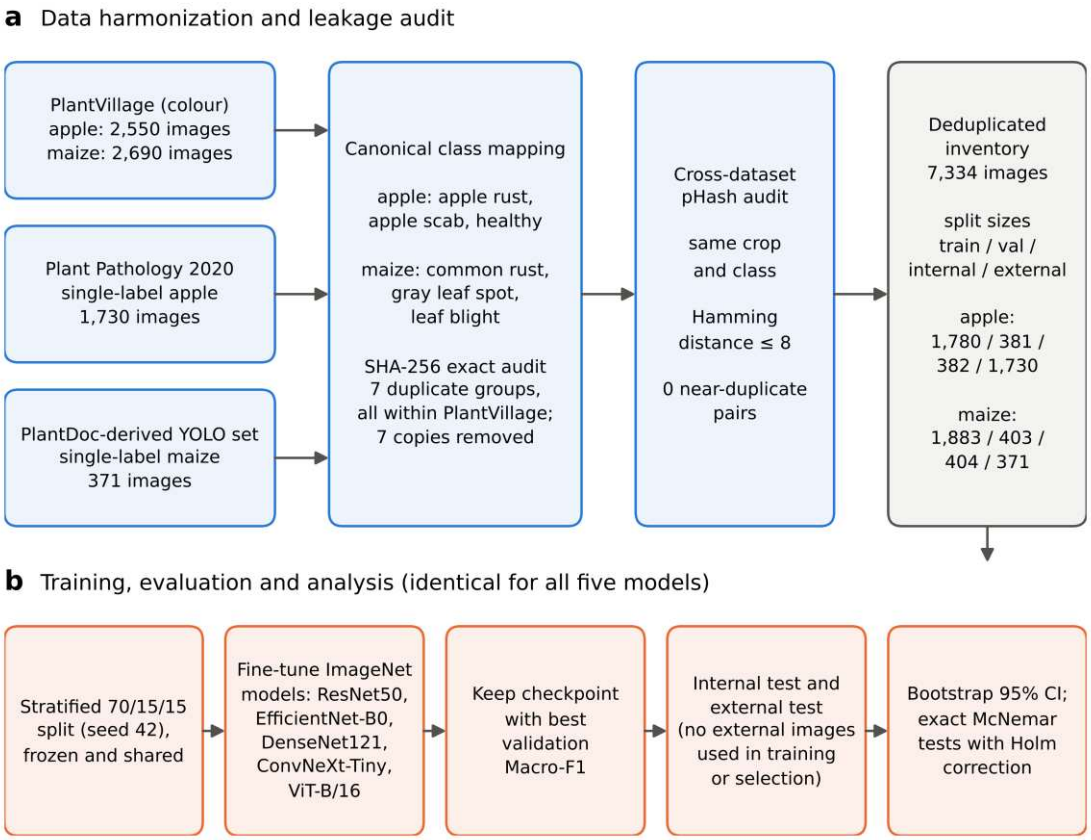


Figure 1. Leakage-audited cross-dataset evaluation workflow. a, Source (PlantVillage) and external (Plant Pathology 2020 for apple; PlantDoc-derived set for maize) images were mapped to three canonical classes per crop, audited for byte-identical files with SHA-256 and for cross-dataset near-duplicates with a 64-bit perceptual hash (Hamming distance ≤ 8 within the same crop and class). Seven duplicate groups, all within PlantVillage, were removed, leaving 7,334 images. **b**, A single stratified 70:15:15 split of PlantVillage (seed 42) was shared by all five ImageNet-pretrained models. The checkpoint with the best validation Macro-F1 was evaluated on the internal test set and, without adaptation, on the external set, followed by bootstrap confidence intervals and exact McNemar tests with Holm correction. Split sizes are given as training / validation / internal test / external test.

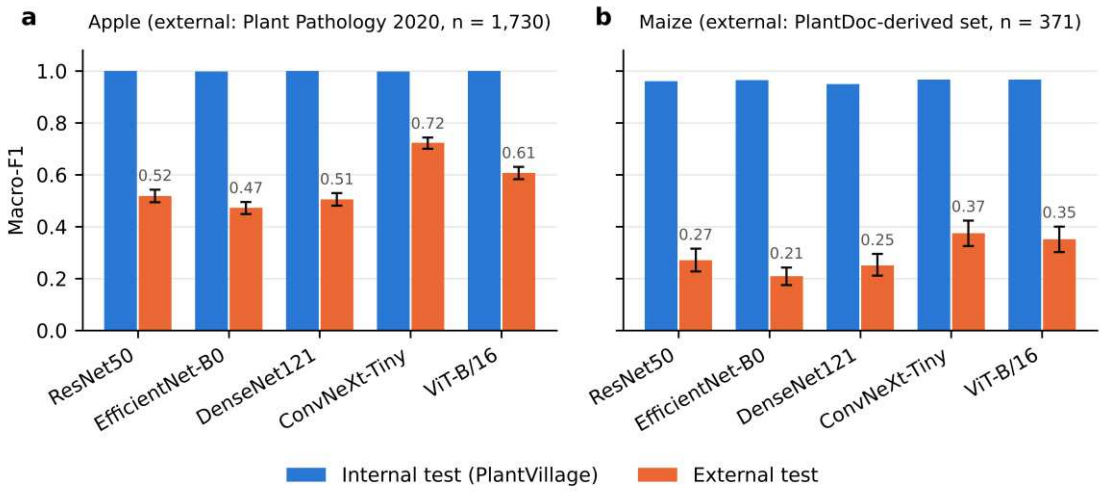


Figure 2. Internal versus external Macro-F1 for five architectures. a, Apple; external test set Plant Pathology 2020 (n = 1,730 images); internal test set n = 382. **b**, Maize; external test set PlantDoc-derived images (n = 371); internal test set n = 404. Blue bars show Macro-F1 on the held-out PlantVillage internal test set; orange bars show Macro-F1 on the external set, with the value printed above each bar. Error bars on external bars are 95% percentile bootstrap confidence intervals from 2,000 resamples of the external images for a fixed trained model; they do not include variation from retraining. Internal values are single point estimates. The vertical axis starts at zero.

Tables

Table 1. Composition of the source and external datasets after duplicate removal. Counts are numbers of images. PlantVillage healthy apple images were reduced from 1,645 to 1,638 after removal of seven byte-identical copies; no other class changed. Plant Pathology 2020 images labelled as having multiple diseases were excluded. The external maize set contains only images whose bounding-box labels included exactly one of the three target maize classes and no other class.

Dataset (role)	Crop	Class	Images
PlantVillage (source)	Apple	Apple rust	275
PlantVillage (source)	Apple	Apple scab	630
PlantVillage (source)	Apple	Healthy	1,638
Plant Pathology 2020 (external)	Apple	Apple rust	622
Plant Pathology 2020 (external)	Apple	Apple scab	592
Plant Pathology 2020 (external)	Apple	Healthy	516
PlantVillage (source)	Maize	Common rust	1,192
PlantVillage (source)	Maize	Gray leaf spot	513
PlantVillage (source)	Maize	Leaf blight	985
PlantDoc-derived set (external)	Maize	Common rust	116
PlantDoc-derived set (external)	Maize	Gray leaf spot	65
PlantDoc-derived set (external)	Maize	Leaf blight	190
Source split, apple		Train / validation / internal test	1,780 / 381 / 382

Dataset (role)	Crop	Class	Images
Source split, maize		Train / validation / internal test	1,883 / 403 / 404

Table 2. Internal and external performance of the five architectures. Internal metrics are from the held-out PlantVillage test set (apple n = 382; maize n = 404); external metrics are from Plant Pathology 2020 (apple, n = 1,730) and the PlantDoc-derived set (maize, n = 371). The 95% confidence interval (CI) for external Macro-F1 is a percentile bootstrap interval from 2,000 resamples of the external images. Gap = internal Macro-F1 − external Macro-F1 (equation (3)); gaps were computed from unrounded values and can differ by 0.0001 from the difference of the rounded entries. Int., internal; Ext., external; Acc., accuracy; BA, balanced accuracy (equal to macro-recall); MCC, Matthews correlation coefficient. Full internal and external metrics, including macro-precision, are given in Supplementary Table S1.

Crop	Model	Int. Macro-F1	Int. MCC	Ext. Acc.	Ext. BA	Ext. Macro-F1 (95% CI)	Ext. MCC	Gap
Apple	ResNet50	1.0000	1.0000	0.5474	0.5404	0.5180 (0.4935–0.5427)	0.4047	0.4820
Apple	EfficientNet-B0	0.9976	0.9949	0.5231	0.5277	0.4722 (0.4483–0.4950)	0.3524	0.5253
Apple	DenseNet121	1.0000	1.0000	0.5358	0.5309	0.5050 (0.4807–0.5290)	0.3982	0.4950
Apple	ConvNeXt-Tiny	0.9976	0.9949	0.7341	0.7416	0.7228 (0.7000–0.7434)	0.6318	0.2748
Apple	ViT-B/16	1.0000	1.0000	0.6364	0.6204	0.6071 (0.5830–0.6300)	0.4836	0.3929
Maize	ResNet50	0.9603	0.9531	0.3154	0.3447	0.2702 (0.2269–0.3149)	−0.0185	0.6901
Maize	EfficientNet-B0	0.9648	0.9576	0.2668	0.3086	0.2092 (0.1751–0.2427)	−0.0679	0.7556
Maize	DenseNet121	0.9491	0.9393	0.2938	0.3453	0.2504 (0.2106–0.2944)	−0.0028	0.6987
Maize	ConvNeXt-Tiny	0.9669	0.9609	0.3989	0.4440	0.3746 (0.3248–0.4232)	0.1850	0.5922
Maize	ViT-B/16	0.9671	0.9609	0.3666	0.4060	0.3519 (0.3017–0.4003)	0.1048	0.6151

Table 3. Pairwise exact McNemar tests on external predictions with Holm correction. For each pair, b is the number of external images classified correctly by model A and incorrectly by model B, and c the reverse; b + c is the number of discordant images (external test sets: apple n = 1,730; maize n = 371). P values are from the exact two-sided McNemar (binomial) test; Holm correction was applied separately to the ten comparisons within each crop, at a family-wise α = 0.05. Holm P, Holm-adjusted P; Sig., significant after Holm correction.

Crop	Model A	Model B	b	c	b + c	P	Holm P	Sig.
Apple	ResNet50	EfficientNet-B0	198	156	354	0.0292	0.0875	No
Apple	ResNet50	DenseNet121	129	109	238	0.218	0.436	No
Apple	ResNet50	ConvNeXt-Tiny	75	398	473	3.39×10^{-54}	2.71×10^{-53}	Yes
Apple	ResNet50	ViT-B/16	149	303	452	3.68×10^{-13}	1.84×10^{-12}	Yes
Apple	EfficientNet-B0	DenseNet121	152	174	326	0.245	0.436	No
Apple	EfficientNet-B0	ConvNeXt-Tiny	62	427	489	4.46×10^{-68}	4.46×10^{-67}	Yes
Apple	EfficientNet-B0	ViT-B/16	195	391	586	4.23×10^{-16}	2.54×10^{-15}	Yes
Apple	DenseNet121	ConvNeXt-Tiny	59	402	461	8.44×10^{-64}	7.59×10^{-63}	Yes
Apple	DenseNet121	ViT-B/16	136	310	446	1.05×10^{-16}	7.35×10^{-16}	Yes
Apple	ConvNeXt-Tiny	ViT-B/16	379	210	589	3.18×10^{-12}	1.27×10^{-11}	Yes
Maize	ResNet50	EfficientNet-B0	38	20	58	0.0247	0.124	No
Maize	ResNet50	DenseNet121	29	21	50	0.322	0.735	No
Maize	ResNet50	ConvNeXt-Tiny	34	65	99	0.00239	0.0148	Yes
Maize	ResNet50	ViT-B/16	30	49	79	0.0422	0.169	No
Maize	EfficientNet-B0	DenseNet121	25	35	60	0.245	0.735	No
Maize	EfficientNet-B0	ConvNeXt-Tiny	22	71	93	3.48×10^{-7}	3.48×10^{-6}	Yes
Maize	EfficientNet-B0	ViT-B/16	21	58	79	3.76×10^{-5}	3.39×10^{-4}	Yes
Maize	DenseNet121	ConvNeXt-Tiny	31	70	101	1.31×10^{-4}	0.00105	Yes

Crop	Model A	Model B	b	c	b + c	P	Holm P	Sig.
Maize	DenseNet121	ViT-B/16	23	50	73	0.00212	0.0148	Yes
Maize	ConvNeXt-Tiny	ViT-B/16	52	40	92	0.251	0.735	No

Table 4. Training configuration and convergence. All models were trained on the same frozen PlantVillage training partitions with identical settings. Epochs run includes the epochs completed before early stopping; the selected epoch is the epoch with the highest validation Macro-F1, whose weights were restored for evaluation.

Setting	Value
Initialization	ImageNet-pretrained torchvision weights; new three-class linear head; all layers fine-tuned
Input size and normalization	224 × 224 pixels; ImageNet mean (0.485, 0.456, 0.406) and s.d. (0.229, 0.224, 0.225)
Training augmentation	Horizontal flip (p = 0.5), rotation ±10°, colour jitter (brightness, contrast, saturation 0.1)
Loss and optimizer	Cross-entropy; AdamW, learning rate 1×10^{-4}, weight decay 1×10^{-4}
Batch size; maximum epochs	16; 10
Early stopping and checkpoint	Patience of 3 epochs on validation Macro-F1; best-validation weights restored
Random seed	42 (data split, weight initialization of new head, data order, bootstrap)

Crop	Model	Epochs run	Selected epoch	Validation Macro-F1 at selected epoch
Apple	ResNet50	9	6	1.0000
Apple	EfficientNet-B0	7	4	1.0000
Apple	DenseNet121	7	4	1.0000
Apple	ConvNeXt-Tiny	5	2	1.0000
Apple	ViT-B/16	7	4	1.0000
Maize	ResNet50	5	2	0.9800
Maize	EfficientNet-B0	10	10	0.9771
Maize	DenseNet121	5	2	0.9678
Maize	ConvNeXt-Tiny	7	4	0.9770
Maize	ViT-B/16	10	8	0.9768

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInformation.docx