Evaluating CNN Architectures and Transfer Learning for Plant Recognition

Zhaokun Wang
Institute of Computational Linguistics
Heidelberg University, Heidelberg, Germany
zhaokun.wang@stud.uni-heidelberg.de

Abstract

This project report presents a study of convolutional neural networks (CNNs) for plant species recognition, focusing on the interaction between model architecture, dataset scale, and transfer learning strategies. On the small-scale Plant Seedlings dataset, a pretrained ResNet-18 reaches 97.7% accuracy, while ConvNeXt-Tiny shows limited convergence when trained from scratch. In contrast, on the larger and more complex LeafSnap dataset, the pretrained ConvNeXt-Tiny attains a higher F1-score (0.979 vs. 0.965 for ResNet-18), reflecting its greater capacity. Experiments also demonstrate that pretraining on a large, in-domain dataset yields a stronger initialization for fine-tuning on smaller tasks than generic ImageNet pretraining. These results show that the optimal architecture depends on dataset scale and complexity, and that in-domain transfer learning can provide clear advantages over standard ImageNet pretraining for specialized tasks.[1]

1 Introduction

Automated plant identification is a cornerstone of modern ecological monitoring and precision agriculture. Deep learning, particularly Convolutional Neural Networks (CNNs), has become the state-of-the-art approach (Kumar et al., 2012). However, deploying a robust system requires overcoming two primary challenges: the task's inherent nature as a fine-grained visual classification (FGVC) problem and the domain shift between controlled and in-the-wild imagery.

This paper investigates these challenges through a series of structured experiments to understand how model architecture, dataset properties, and pretraining strategies interact. We aim to answer three key research questions:

1. How does the performance of different architectures scale with dataset size and complexity?

2. Can pretraining on a large, in-domain plant dataset provide a better starting point for fine-tuning on a smaller, related task than generic ImageNet pretraining?

3. What is the impact of leveraging a massive but potentially noisy in-domain dataset (iNaturalist) for pretraining?

By addressing these questions, we provide practical guidance for selecting optimal models and training paradigms for plant recognition systems.

2 Experiment 1: Architecture Benchmarking on a Small-Scale Dataset

2.1 Dataset and Preprocessing

Our initial benchmark utilized the Plant Seedlings dataset (Giselsson et al., 2017), a collection of 4,750 images across 12 seedling species. Its controlled environment provides an ideal testbed for evaluating the core learning capacity of different models. We employed a 90%/10% stratified split. To prevent overfitting, we implemented a strong augmentation pipeline featuring 'RandAugment' (Cubuk et al., 2020) and 'RandomErasing' (Zhong et al., 2020), which encourages models to learn robust and generalizable features.

2.2 Methodology and Training Strategies

We compared three architectures representing different complexity levels: our custom lightweight **CustomCNN-S** (1.3 M parameters), the canonical **ResNet-18** (11.2 M parameters), and the modern **ConvNeXt-Tiny** (27.8 M parameters).

A unified, high-performance training framework was used for fair comparison:

[1] Codes are publicly available at https://github.com/BufferHund/PlantRecognition_SemesterProject

- **Optimizer:** AdamW paired with the **OneCycle learning rate policy** (Smith, 2018) to promote faster convergence and locate better minima.
- **Regularization: Mixup** (Zhang et al., 2017) and **CutMix** (Yun et al., 2019) were used to create synthetic training samples, pushing the model to learn more invariant features.
- **Progressive Resizing:** Models were first trained on 224×224 images and then briefly fine-tuned on a higher resolution (288×288) to refine feature extraction.

2.3 Results and Discussion

The results, summarized in Table 1, highlight the critical importance of transfer learning in data-constrained settings.

Table 1: Performance on the Plant Seedlings validation set. Pretrained models significantly outperform those trained from scratch.

Training Regime	Model	Params (M)	Acc. (%)	F1
From Scratch	CustomCNN-S	1.3	88.2	0.861
	ResNet-18	11.2	96.4	0.961
	ConvNeXt-Tiny	27.8	59.2	0.496
ImageNet Pretrained	ResNet-18	11.2	**97.7**	**0.974**
	ConvNeXt-Tiny	27.8	96.6	0.961

The pretrained **ResNet-18** emerged as the top performer. Most notably, the from-scratch ConvNeXt-Tiny model failed to converge effectively (Figure 1). This result vividly demonstrates the "data hunger" of modern architectures with weaker inductive biases. Without the strong structural priors of ResNet or the guidance of pretraining, ConvNeXt's vast parameter space could not be optimized on the limited data. This finding suggests that for smaller, specialized datasets, a classic pretrained architecture is the most reliable choice.

A detailed model-by-model analysis, including confusion matrices and PCA feature space visualizations is provided in the Appendix.

3 Experiment 2: Generalization and Fine-Grained Analysis on a Large-Scale Dataset

To test our initial conclusions under more demanding conditions, we evaluated the pretrained models on the larger and more complex LeafSnap dataset, probing their generalization on a challenging FGVC task.

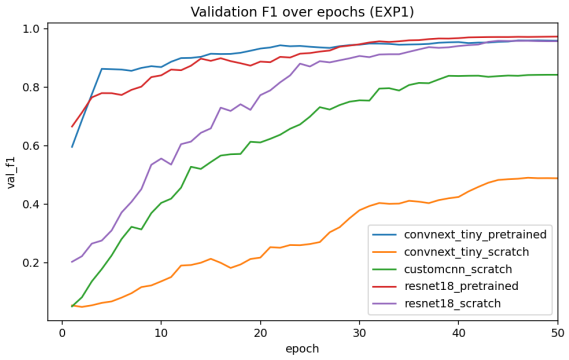

Figure 1: Validation F1-score on the Seedlings dataset. The plot highlights the robust convergence of ResNet-18 and the training failure of from-scratch ConvNeXt-Tiny.

3.1 Dataset and Domain Adaptation

We curated a subset of the LeafSnap dataset (Kumar et al., 2012) containing the top 50 species (with $\geq$80 images each). To test domain generalization, we merged images from both "lab" (clean background) and "field" (natural background) settings. An 85%/15% stratified split was used. To address the natural class imbalance, we employed a 'WeightedRandomSampler' during training to ensure minority classes were not overlooked.

3.2 Methodology

Both ImageNet-pretrained models (ResNet-18 and ConvNeXt-Tiny) were fine-tuned for 20 epochs using the AdamW optimizer with a learning rate of 2×10^{-4}. Augmentations included 'RandAugment' and 'ColorJitter' to suit the dataset's diversity.

3.3 Results: Model Capacity Matters at Scale

In a reversal of the findings from Experiment 1, the higher-capacity **ConvNeXt-Tiny** delivered superior performance on the larger LeafSnap dataset (Table 2).

Table 2: Performance on the LeafSnap validation set. The higher-capacity ConvNeXt-Tiny outperforms ResNet-18.

Model	Val. Acc. (%)	Val. F1
ResNet-18	96.8	0.965
ConvNeXt-Tiny	**97.8**	**0.979**

The convergence plot (Figure 2) reinforces this, showing that ConvNeXt-Tiny not only converged faster but also maintained a consistent performance advantage, indicating its ability to better leverage the richer dataset.

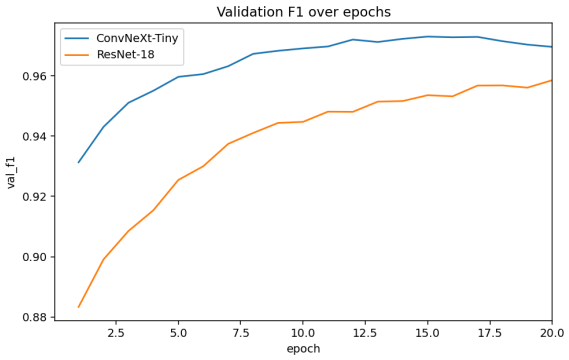

Figure 2: Validation F1-score on LeafSnap. ConvNeXt-Tiny demonstrates faster convergence and achieves a higher final performance.

3.4 Per-Class Analysis

A granular analysis of per-class F1-scores reveals the architectural trade-offs. ResNet-18's most significant advantage was on *prunus sargentii* (+0.23 F1), a species with distinct, serrated leaf margins that may align well with the localized convolutional filters of ResNet (Figure 3).

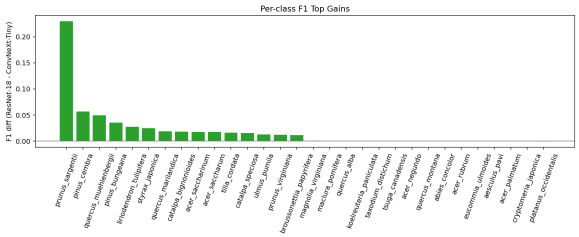

Figure 3: Top 10 species where ResNet-18's F1-score was higher. The advantage is most dramatic for *prunus sargentii*.

Conversely, ConvNeXt-Tiny excelled on species known to be visually similar, such as *ostrya virginiana* and *carpinus caroliniana* (Figure 4). This suggests that ConvNeXt's larger effective receptive fields are superior at capturing the subtle, holistic patterns required to differentiate these challenging FGVC cases.

4 Experiment 3: In-Domain vs. Generic Transfer Learning

4.1 Motivation and Setup

This experiment investigates whether pretraining on a large, **in-domain** dataset (LeafSnap) offers a better starting point for a smaller target task (Plant Seedlings) than generic ImageNet pretraining. We pretrained ConvNeXt-Tiny on LeafSnap

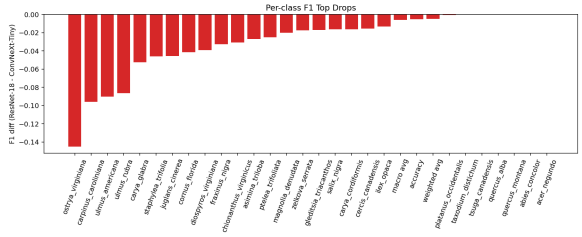

Figure 4: Top 10 species where ConvNeXt-Tiny's F1-score was higher, highlighting its strength in fine-grained differentiation.

and then applied two transfer strategies to the Plant Seedlings dataset:

- **Full Fine-Tuning (Full-FT):** All model parameters were updated on the target dataset for 30 epochs.
- **Linear Probing + Gradual Unfreezing (LP+Unfreeze):** Only the classification head was trained for 5 epochs, after which the backbone layers were progressively unfrozen over 15 epochs.

4.2 Results and Discussion

The results in Table 3 and Figure 5 clearly show that Full-FT is the superior strategy. It quickly reached a high F1-score and achieved a final performance that surpasses all previous results on the Plant Seedlings dataset.

Table 3: Experiment 3: Transfer from LeafSnap to Plant Seedlings.

Strategy	Best Val Acc	Best Macro-F1
Full Fine-Tuning	**0.979**	**0.978**
LP + Unfreeze	0.918	0.910

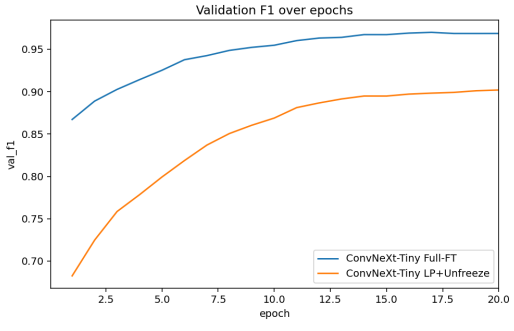

Figure 5: Validation F1 for Full-FT vs. LP+Unfreeze. Full fine-tuning demonstrates significantly better and faster convergence.

This experiment demonstrates that while in-domain pretraining provides a powerful feature representation, its full potential is only unlocked when the entire model is allowed to adapt to the target data distribution via full fine-tuning.

5 Experiment 4: Impact of Large-Scale, Noisy Pretraining

5.1 Experimental Setup

To evaluate the impact of even larger-scale pretraining, we trained ConvNeXt-Tiny on a massive subset of the iNaturalist dataset containing only *Plantae* entries. We then transferred this model to the Plant Seedlings dataset using two strategies: (1) full fine-tuning (**EXP4 Full-FT**) and (2) linear probing with gradual unfreezing (**EXP4 LP+Unfreeze**). Their performance was compared against the two baselines from Experiment 3: LeafSnap-pretrained Full-FT (**EXP3 Full-FT**) and LP+Unfreeze (**EXP3 LP+Unfreeze**).

5.2 Results and Discussion

As shown in Table 4 and Figure 6, the iNaturalist pretraining provided a strong initialization. The **EXP4 Full-FT** model converged rapidly, achieving a final macro F1-score of **0.965** and validation accuracy of **0.968**, only slightly below the best-performing model EXP3 Full-FT (0.978) but clearly surpassing both LP+Unfreeze baselines. The **EXP4 LP+Unfreeze** variant achieved **0.938** macro F1, representing a gain over EXP3 LP+Unfreeze (0.910) but still trailing behind both full fine-tuning runs.

Table 4: Experiment 4: Transfer from iNaturalist to Plant Seedlings.

Strategy	Best Val Acc	Best Macro-F1
Full Fine-Tuning	**0.968**	**0.965**
LP + Unfreeze	0.945	0.938

Per-class analysis (Figure 7) shows that while EXP3 Full-FT retains a slight edge on high-support classes such as *Maize*, the performance gap across most species is minimal. These results reinforce two conclusions: (1) full fine-tuning consistently outperforms LP+Unfreeze regardless of pretraining source, and (2) large-scale, diverse pretraining can yield near state-of-the-art performance even compared to smaller but in-domain pretraining, provided that the entire model is fine-tuned.

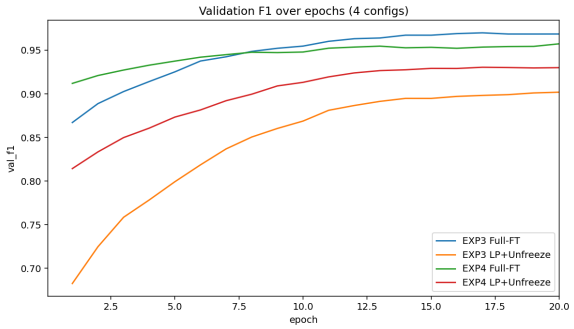

Figure 6: Validation F1 over epochs for all four transfer strategies. iNaturalist pretraining (EXP4) leads to rapid convergence and strong final performance.

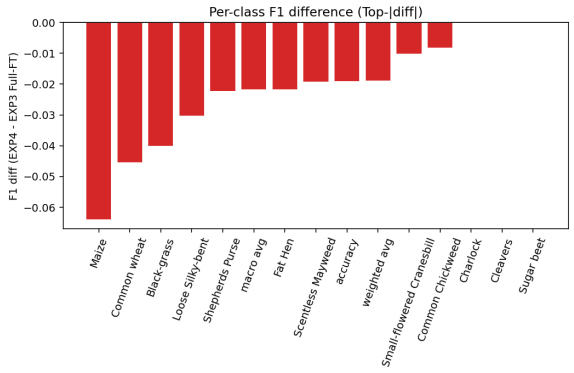

Figure 7: Per-class F1 difference (EXP4 Transfer minus EXP3 Full-FT). Performance is comparable across most classes.

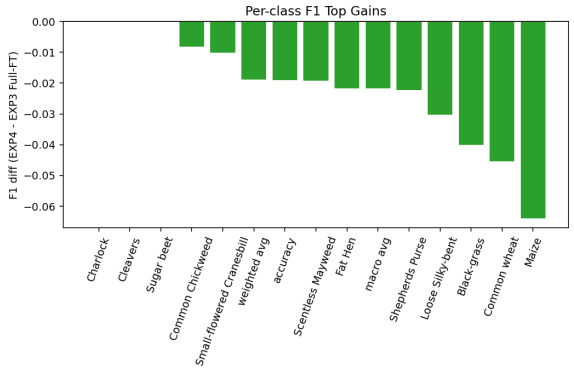

Figure 8: Top per-class F1 gains from EXP4 relative to EXP3 Full-FT.

6 Overall Discussion

Our multi-stage study provides an integrated view of the trade-offs involved in model selection for plant recognition.

Experiment 1 showed that on a small, clean dataset, the strong inductive bias of a classic architecture such as ResNet-18 acted as an effective regularizer, yielding higher accuracy than a higher-capacity ConvNeXt-Tiny, which was more prone to overfitting. **Experiment 2** revealed a different trend: on a larger and more varied dataset, ConvNeXt-Tiny's greater capacity offered an advantage, enabling it to learn finer-grained features and outperform ResNet-18.

Experiments 3 and 4 extended this analysis to pretraining strategies. The results indicate that pretraining on large, in-domain datasets (LeafSnap, iNaturalist) generally provides a stronger starting point than generic ImageNet pretraining, leading to faster convergence and improved performance. This benefit was most evident when the entire model was fine-tuned, allowing the learned features to align more closely with the target task.

Taken together, these findings support a central conclusion: there is no universally "best" architecture or pretraining method. The optimal choice depends on balancing model capacity with the scale, diversity, and domain similarity of the available data.

7 Conclusion and Future Work

This work provides practical guidance for aligning model architectures with dataset characteristics in plant species recognition.

- For **smaller, controlled datasets** (under roughly 10k images), a pretrained **ResNet-18** achieved a consistent balance of performance and training stability.
- For **larger, fine-grained tasks** (over roughly 20k images with many similar classes), the higher capacity of a modern architecture such as **ConvNeXt-Tiny** delivered stronger performance.

Across these scenarios, our results highlight the central role of transfer learning in specialized vision domains. While ImageNet remains a strong generic baseline, pretraining on large, in-domain datasets produced additional gains, particularly when full fine-tuning was applied.

Future work will include developing ensemble methods that integrate ResNet-18 and ConvNeXt-Tiny according to their respective strengths, and evaluating specialized FGVC techniques such as attention-based part localization or metric learning to improve performance on the most challenging species.

References

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703.

Thomas Mosgaard Giselsson, Rasmus Nyholm Jørgensen, Peter Kryger Jensen, Mads Dyrmann, and Henrik Skov Midtiby. 2017. A public image database for benchmark of plant seedling classification algorithms. *arXiv preprint arXiv:1711.05458*.

Neeraj Kumar, Peter N Belhumeur, Arijit Biswas, David W Jacobs, W John Kress, Ida C Lopez, and João VB Soares. 2012. Leafsnap: A computer vision system for automatic plant species identification. In *European conference on computer vision*, pages 502–516. Springer.

Leslie N Smith. 2018. A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008.

Appendix

A Confusion Matrices (Row-Normalized)

The row-normalized confusion matrices provide a granular view of per-class classification performance. The plots for the pretrained ResNet-18 and ConvNeXt-Tiny models exhibit strong diagonal dominance, indicating high and consistent accuracy across all classes. In contrast, the models trained from scratch, particularly the custom CNN, show significant off-diagonal noise. This pattern visually confirms the quantitative results from our experiments, highlighting the superior discriminative ability conferred by transfer learning.

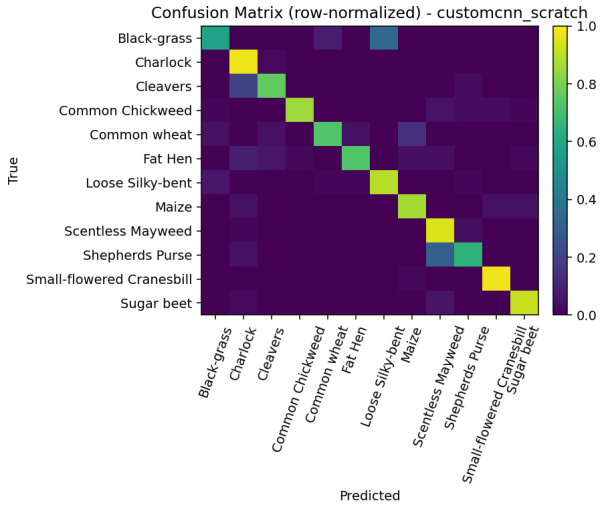

Figure 9: Confusion Matrix (Row-Normalized) – **customcnn_scratch**.

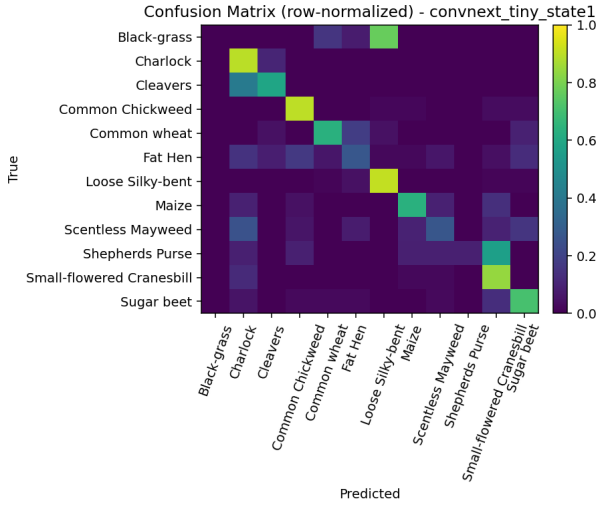

Figure 10: Confusion Matrix (Row-Normalized) – **convnext_tiny_scratch**.

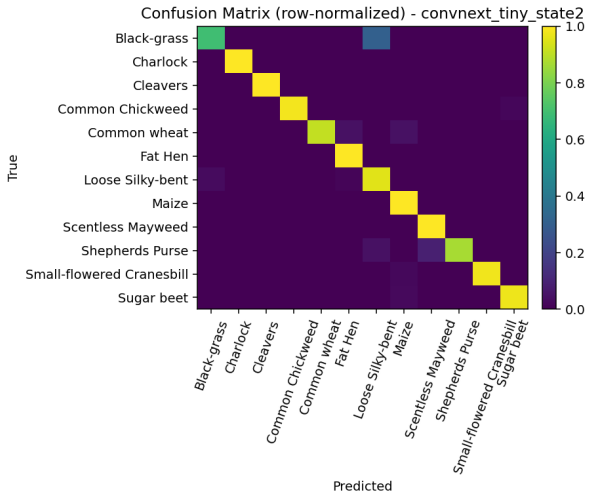

Figure 11: Confusion Matrix (Row-Normalized) – **convnext_tiny_pretrained**.

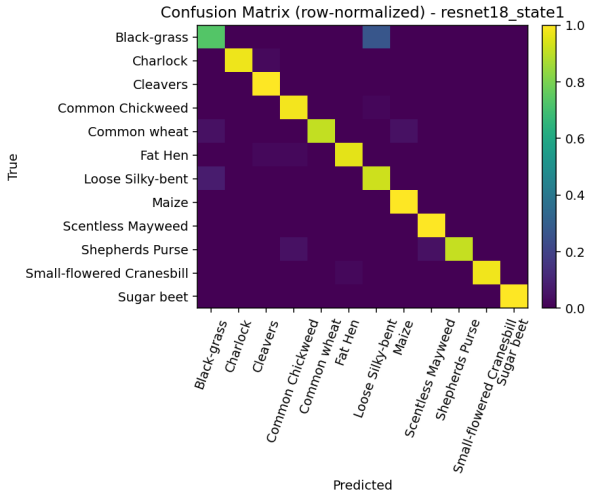

Figure 12: Confusion Matrix (Row-Normalized) – **resnet18_scratch**.

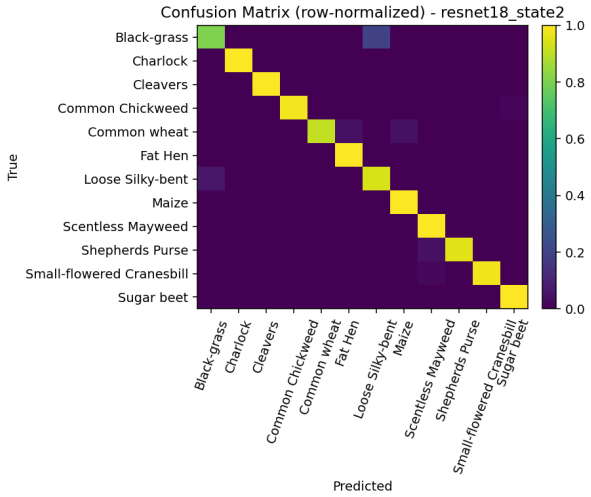

Figure 13: Confusion Matrix (Row-Normalized) – **resnet18_pretrained**.

B PCA of Validation Features

To visualize the quality of the learned feature representations, we project the validation embeddings into a two-dimensional space using PCA. The resulting plots clearly demonstrate that the pretrained models learn a highly structured and separable feature space, evidenced by the tight, distinct intra-class clusters. Conversely, the feature space of the models trained from scratch is largely overlapping and poorly defined, indicating a limited capacity to extract discriminative features from the training data.

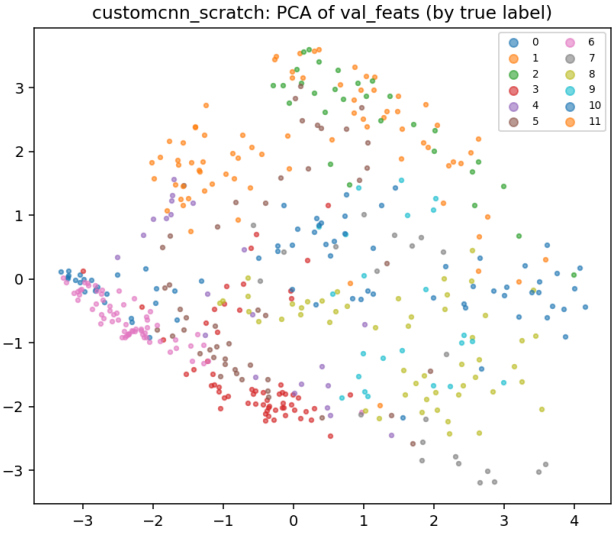

Figure 14: PCA Feature Space – **customcnn_scratch**.

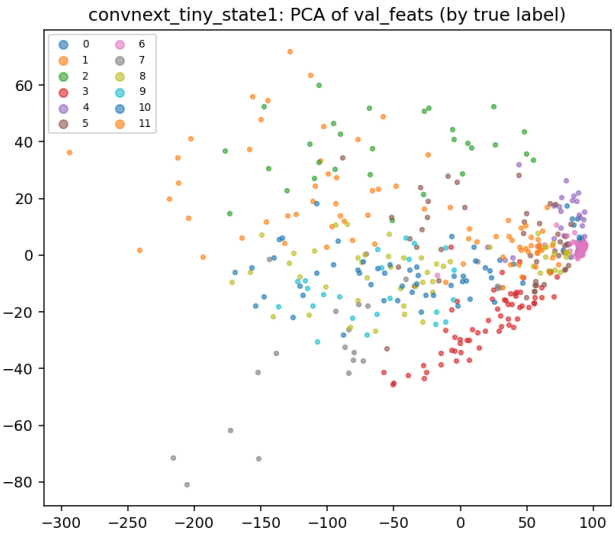

Figure 15: PCA – **convnext_tiny_scratch**.

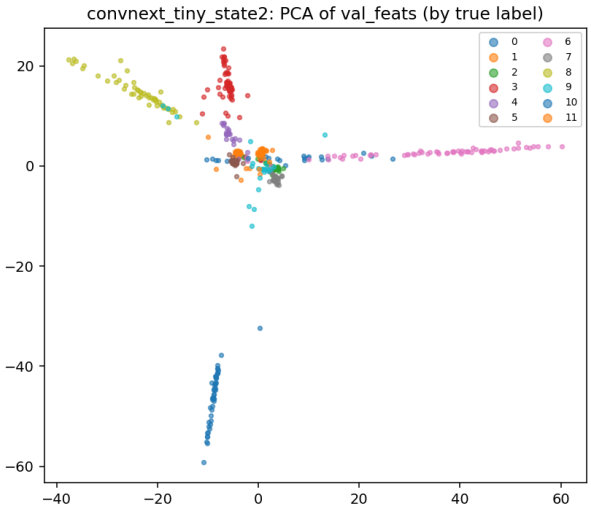

Figure 16: PCA – **convnext_tiny_pretrained**.

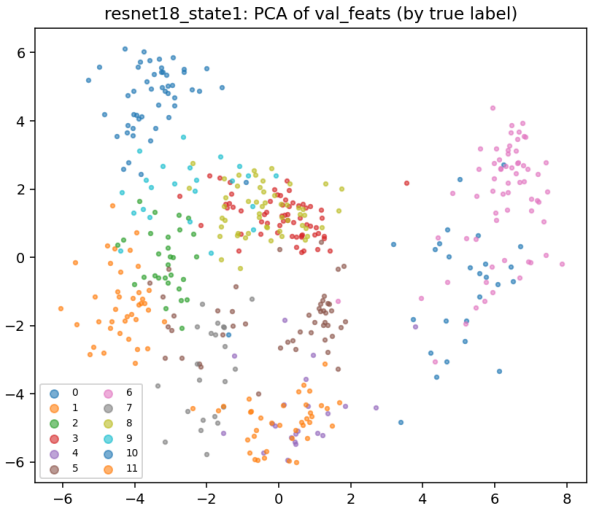

Figure 17: PCA Feature Space – **resnet18_scratch**.

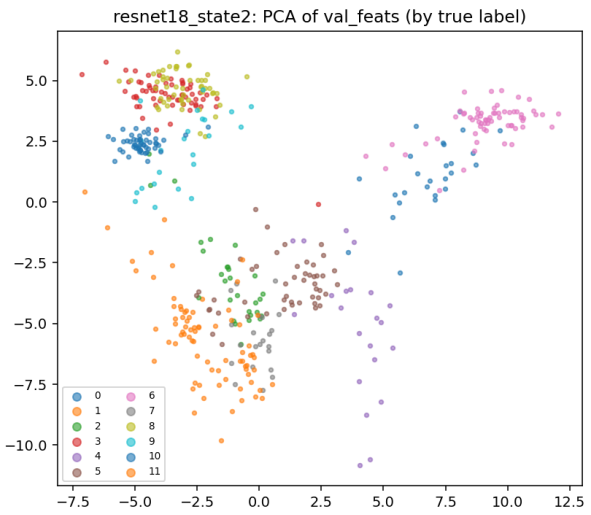

Figure 18: PCA Feature Space – **resnet18_pretrained**.

C Class Activation Map (CAM) Visualization Across Models

Class Activation Maps (CAMs) offer insight into the models' decision-making process by highlighting the image regions most influential for a given prediction. The visualizations reveal that the pretrained architectures learn to focus their attention precisely on salient plant structures, such as leaf margins and stems. In sharp contrast, the models trained from scratch often produce diffuse activations that incorrectly emphasize background elements, explaining their lower performance and demonstrating the value of pretraining for learning a robust semantic focus.

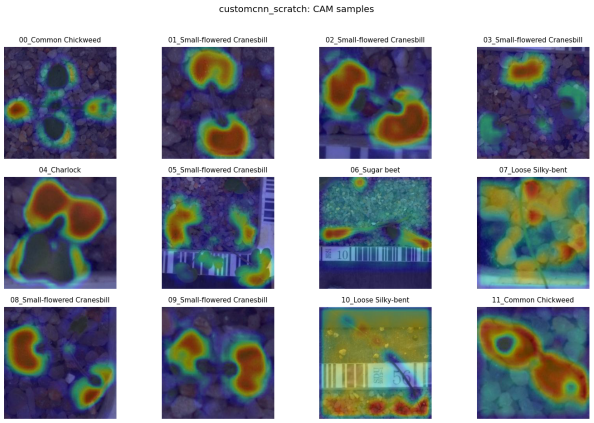

Figure 19: CAMs – **customcnn_scratch**.

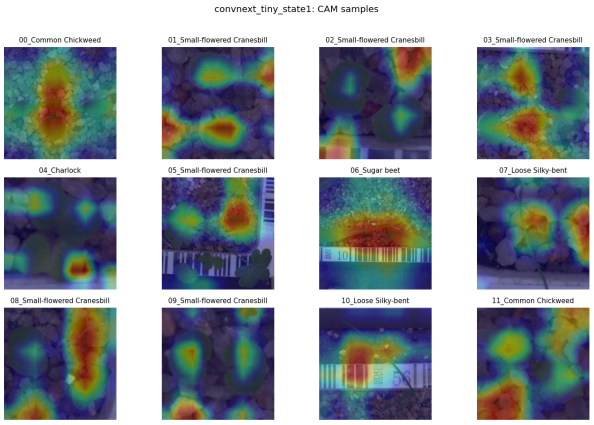

Figure 20: CAMs – **convnext_tiny_scratch**.

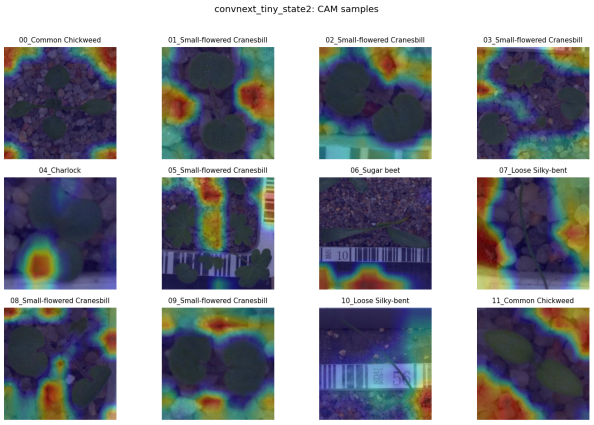

Figure 21: CAMs – **convnext_tiny_pretrained**.

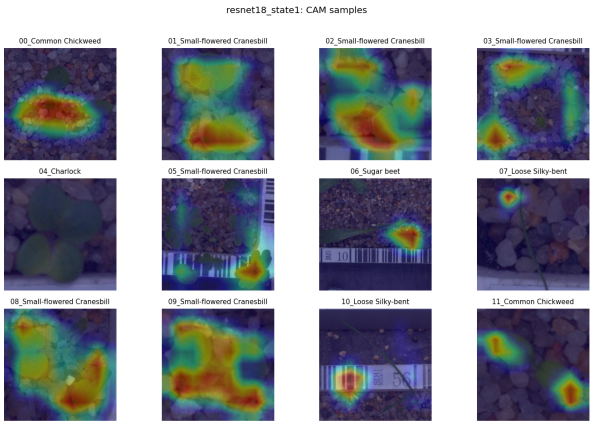

Figure 22: Representative CAMs – **resnet18_scratch**.

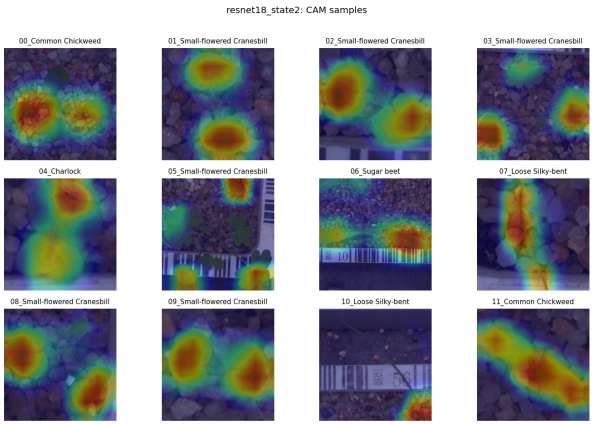

Figure 23: CAMs – **resnet18_pretrained**.

D Use of AI-Based Tools

This appendix documents the use of artificial intelligence (AI)-based tools in the preparation of this academic work.

List of Steps Involving AI-Based Tools

- **DeepSeek**: I consulted DeepSeek models to learn more formal organization of the conclusion and appendix chapter. The suggested frameworks were adapted and rewritten entirely in my own words. I also referred to DeepSeek during debugging to understand and resolve specific error messages.
- **QuillBot**: QuillBot was used sparingly to rephrase sentences for improved readability and flow. All suggestions were manually reviewed and edited to ensure alignment with my original intent and academic style.
- **DeepL and Youdao Translation**: DeepL and Youdao Translation assisted in translating a small number of technical terms and short phrases from Chinese to English to clarify meaning during drafting. These translations were verified and incorporated into my own text.