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.

A Controlled Multi-dataset Evaluation of Custom CNNs, Pretrained Feature Extractors, and Transfer Learning

Tanzeem Maliat

tanzeem-2019517834@cs.du.ac.bd

University Of Dhaka

Asfin Jannat Shamsi

asfinjannat.shamsi23@gmail.com

University Of Dhaka

Research Article

Keywords: Convolutional Neural Networks, Transfer Learning, Feature Extraction, Image Classification, Deep Learning

Posted Date: January 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8544413/v1

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

Additional Declarations: The authors declare no competing interests.

A Controlled Multi-dataset Evaluation of Custom CNNs, Pretrained Feature Extractors, and Transfer Learning

Asfin Jannat Shamsi[1*†] and Tanzeem Maliat[1*†]

[1*]Dept of Computer Science and Engineering, University of Dhaka, Dhaka, 1200, Bangladesh.

*Corresponding author(s). E-mail(s): asfinjannat.shamsi23@gmail.com; maliattanzeem@gmail.com;
†These authors contributed equally to this work.

Abstract

Convolutional neural networks (CNNs) are commonly trained using one of three paradigms: training from scratch, reuse of pretrained representations, or transfer learning via fine-tuning. While each strategy is widely adopted, their relative effectiveness depends strongly on dataset characteristics, computational constraints, and deployment requirements. This paper presents a controlled and reproducible comparison of these three training paradigms across five real-world image classification datasets spanning infrastructure inspection, agricultural disease recognition, and object-centric classification. All models are trained and evaluated under identical data splits, preprocessing pipelines, optimization settings, and evaluation metrics. Performance is reported using macro-averaged accuracy, precision, recall, and F1-score, alongside training time and model complexity. Rather than proposing new network architectures, this work emphasizes controlled evaluation and reproducibility to support evidence-based selection of CNN training strategies across diverse application domains.

Keywords: Convolutional Neural Networks, Transfer Learning, Feature Extraction, Image Classification, Deep Learning

1 Introduction

Deep convolutional neural networks (CNNs) have become the dominant paradigm for image classification, achieving state-of-the-art performance across a wide range of benchmark datasets and real-world applications. Advancements in network depth, architectural design, and large-scale supervised pretraining have enabled CNNs to learn rich hierarchical feature representations that generalize well to diverse visual tasks. As a result, modern pre-trained models such as VGG and ResNet are widely adopted in both academic research and applied systems.

However, in practical deployments, model selection is rarely driven by classification accuracy alone. Factors such as training time, computational complexity, memory footprint, and ease of adaptation to new datasets often play an equally critical role, particularly in resource-constrained environments or rapid prototyping scenarios. Consequently, practitioners typically adopt one of three common training strategies: (i) designing and training a task-specific CNN from scratch, (ii) using a pre-trained CNN as a fixed feature extractor, or (iii) applying transfer learning by fine-tuning a pre-trained model on the target dataset.

Although all three approaches are widely used, their relative strengths and trade-offs can vary significantly depending on the characteristics of datasets, model capacity, and the available computational resources. Existing studies often focus on a single dataset or a single training paradigm, making it difficult to draw consistent conclusions regarding their comparative effectiveness under controlled conditions. Moreover, experimental settings such as data splits, optimization parameters, and hardware configurations are not always aligned, limiting the reproducibility and fairness of such comparisons.

In this work, we present a systematic and controlled empirical evaluation of these three CNN training paradigms across five distinct image classification datasets. A custom CNN architecture, two widely used pre-trained models, and their transfer-learning counterparts are evaluated under a unified experimental protocol with consistent data splits, hyperparameters, and evaluation metrics. Performance is assessed not only in terms of classification accuracy, precision, recall, and F1-score, but also with respect to training time and model size, providing a comprehensive view of performance–efficiency trade-offs.

The primary objective of this study is to present a controlled and reproducible empirical comparison of widely used CNN training paradigms, with the goal of informing practical model selection decisions under real-world performance and resource constraints.

2 Related works

Convolutional neural networks have been extensively studied for image classification, with early deep architectures demonstrating the effectiveness of hierarchical feature learning over hand-crafted representations. The success of large-scale CNNs was popularized by AlexNet, which significantly advanced image classification performance on the ImageNet benchmark and established deep CNNs as the dominant approach in computer vision tasks [1]. Since then, numerous custom CNN architectures have been

proposed and trained from scratch for domain-specific applications, particularly where model simplicity, interpretability, or computational efficiency are prioritized.

With the availability of large-scale labeled datasets such as ImageNet, pre-trained CNN models have become widely used as generic feature extractors. Architectures such as VGG and ResNet have demonstrated strong transferability of learned representations across diverse visual domains [2, 3]. Several studies have shown that features extracted from intermediate and deep layers of these models can be effectively reused for downstream tasks without additional fine-tuning, making pre-trained CNNs a practical baseline in scenarios with limited training data or computational resources [4].

Transfer learning through fine-tuning pre-trained CNNs has further improved adaptability and task-specific performance. By retaining low-level feature representations while updating higher-level layers, fine-tuning allows models to balance generalization and specialization. Previous work has consistently reported that transfer learning outperforms both training-from-scratch approaches and fixed feature extraction, especially when target datasets are relatively small [5]. However, this performance gain often comes at the cost of increased training time and higher computational and memory requirements.

Despite extensive research on CNN architectures and transfer learning strategies, systematic comparisons across multiple datasets under consistent experimental conditions remain limited. Existing studies typically focus on a single dataset, a specific application domain, or a single training paradigm, making it difficult to generalize conclusions regarding performance–efficiency trade-offs.

In contrast, this work evaluates custom CNNs, pre-trained models, and transfer learning approaches using five image datasets sourced from independent arXiv studies, all reimplemented under a unified experimental protocol. This enables a fair and reproducible comparative analysis that highlights both predictive performance and computational efficiency across diverse datasets.

3 Datasets

In this study, we evaluate all models on five image classification datasets sourced from prior research works. These datasets cover diverse visual domains, including urban scene analysis, sidewalk encroachment, agricultural object recognition, and infrastructure inspection. All datasets were reimplemented under a unified preprocessing and partitioning protocol to ensure fair comparison across models.

3.1 Unauthorized Vehicle Detection Dataset

The Unauthorized Vehicle Detection dataset was introduced to facilitate automatic detection of auto-rickshaws and other unauthorized vehicles in traffic images relevant to smart city applications. The dataset consists of 1,730 annotated images captured under various traffic conditions and includes ground truth labels for object detection and classification tasks [6].

3.2 Road Damage and Manhole Detection Dataset

This dataset was created for urban infrastructure monitoring and contains annotated images representing road surface conditions and manhole presence. Polygonal annotations allow precise labeling of damage and manhole regions. The dataset comprises over one thousand images collected from urban environments, with multi-class labels corresponding to Broken, Not Broken, or Manhole categories [7].

3.3 FootpathVision Dataset

FootpathVision is a comprehensive dataset designed for sidewalk encroachment detection in dense urban environments. It contains 1,238 images captured in Dhaka, Bangladesh, with COCO-style bounding box annotations for both binary (encroached vs clear) and multi-class encroachment categories. Baseline performance results reported in the original study demonstrate the dataset's utility for both binary and multi-class classification tasks [8].

3.4 MangoImageBD Dataset

MangoImageBD is a large agricultural image dataset comprising 28,515 images of fifteen common mango (Mangifera indica) varieties collected from fruit markets across multiple districts. Images were captured under controlled conditions and augmented using standard image transformations to improve model robustness. This dataset enables fine-grained classification of mango varieties based on visual appearance [9].

3.5 PaddyVarietyBD Dataset

PaddyVarietyBD consists of 14,000 microscopic RGB images of paddy kernels representing various rice varieties. The dataset was assembled to support automated classification and analysis of agricultural crop species, particularly in relation to varietal identification and phenotypic diversity research [10].

Table 1 Summary of datasets used in this study.

Dataset	#Images	Classes	Description
Unauthorized Vehicle Detection	~1,730	2	Traffic vehicle classification
Road Damage & Manhole	> 1,000	2	Urban infrastructure condition classes
FootpathVision	1,238	2	Sidewalk encroachment detection
MangoImageBD	28,515	15	Mango variety classification
PaddyVarietyBD	14,000	35	Paddy variety classification

4 Model Architectures

This study evaluates three categories of convolutional neural networks (CNNs) on multiple image classification datasets: (i) a custom CNN trained from scratch, (ii) pre-trained CNNs used as fixed feature extractors, and (iii) transfer learning models obtained by fine-tuning pre-trained networks. All models were trained using identical input preprocessing and evaluation protocols to ensure a fair comparison.

4.1 Custom CNN

The custom CNN is a lightweight, task-specific architecture designed for general-purpose image classification. It consists of four convolutional blocks, each with a convolutional layer followed by batch normalization and ReLU activation. Max-pooling layers reduce spatial dimensions after selected blocks, and a dropout layer is applied prior to the fully connected layers to mitigate overfitting. The feature maps are flattened and passed through a fully connected layer with 512 units, followed by the output layer corresponding to the number of target classes. The network contains all trainable parameters and is optimized from scratch on each dataset.

4.2 Pre-trained CNNs (Fixed Feature Extractors)

Two widely used architectures, ResNet-18 and VGG-16, pre-trained on ImageNet, are employed as fixed feature extractors. For these models, all convolutional layers are frozen to retain learned representations, while only the final classifier layer is replaced to match the number of target classes. This approach evaluates the transferability of general-purpose ImageNet features without additional adaptation to the target dataset. Fixed-feature models require fewer training epochs and computational resources due to the frozen backbone.

4.3 Transfer Learning Models (Fine-tuned)

The same ResNet-18 and VGG-16 architectures are also fine-tuned for transfer learning. In this setting, higher-level layers (e.g., the last residual block in ResNet-18 and the final convolutional layers in VGG-16) and the classifier are retrained, while early layers remain frozen. This allows adaptation of high-level semantic features to the target domain, typically improving classification performance at the expense of additional computation. Fine-tuning is especially beneficial for datasets with domain-specific characteristics or small sample sizes.

4.4 Implementation Notes

All models receive images resized to 224×224 pixels and normalized with ImageNet mean and standard deviation. Training uses AdamW optimization with task-specific learning rates, and the checkpoint achieving the highest validation macro F1-score is selected for evaluation. Model complexity is reported in terms of total parameters and trainable parameters, providing a measure of computational footprint and learning capacity.

5 Experimental Setup

5.1 Dataset Splits and Preprocessing

All experiments use a fixed train/validation/test split. Images are resized to 224 × 224 and normalized using ImageNet statistics to ensure consistent input distributions across models and datasets.

5.2 Training Protocol

Models are trained with consistent hyperparameters, with epoch budgets determined by convergence behavior. Early stopping is applied based on the highest validation Macro F1. Wall-clock time is recorded for each training run.

5.3 Evaluation Metrics

Performance is evaluated using Macro Accuracy, Precision, Recall, and F1-score. Macro-averaging assigns equal weight to each class, which is particularly suitable for datasets exhibiting class imbalance. Model complexity is reported using total and trainable parameters.

5.4 Model Selection and Strategy Rationale

For each model, the checkpoint achieving the highest validation Macro F1 is selected. Custom CNNs are trained from scratch, pre-trained models are used as fixed feature extractors, and transfer learning models are fine-tuned to adapt higher-level features to the target dataset. Fixed feature extractors train faster but may under-adapt to domain-specific patterns; fine-tuning generally improves performance at higher computational cost.

5.5 Analysis Considerations

Macro-averaged metrics are used as the primary performance criterion. Confusion matrices and per-class metrics are examined to identify weak classes and guide potential improvements, such as class-specific augmentation or additional data collection. Efficiency metrics, including F1-per-minute and trainable capacity, are also reported to balance predictive performance with computational cost.

6 Results Analysis

This section reports dataset-wise experimental results for five image classification datasets: MangoImageBD, Road Damage and Manhole detection, FootpathVision, PaddyVarietyBD, and Unauthorized Vehicle Detection. Across all datasets, we evaluate five model regimes: (i) a Custom CNN trained from scratch, (ii) ResNet18 and VGG16 as fixed feature extractors (frozen backbone + trainable classifier head), and (iii) ResNet18 and VGG16 with transfer learning via fine-tuning. Model selection is based primarily on macro-averaged F1 score (Macro F1) to ensure that minority classes

are not overshadowed by majority classes and to maintain fairness across classes in multi-class settings.

Unless stated otherwise, hyperparameter selection (learning rate, dropout for Custom CNN; fine-tuning settings for pretrained models) is performed using the validation set only. The test set is used strictly for final reporting.

6.1 MangoImageBD Dataset

MangoImageBD is a multi-class fine-grained classification problem where inter-class differences can be subtle (texture, speckling patterns, shape cues, and color distribution). Such datasets generally benefit from pretrained representations because the backbone already encodes rich mid-level features (edges, textures, object parts), allowing the model to adapt with less labeled data than training from scratch.

Table 2 Comparison of CNN models on MangoImageBD dataset

Rank	Model	Accuracy	Precision	Recall	F1	Train_T(min)	#Params
1	ResNet18_transfer_learning	0.9493	0.9597	0.9493	0.9525	6.39	11.184
2	VGG16_transfer_learning	0.9422	0.9687	0.9422	0.9499	7.47	134.322
3	CustomCNN_from_scratch	0.8552	0.8973	0.8552	0.8678	31.39	411.439
4	ResNet18_fixed_feature	0.8120	0.9082	0.8120	0.8436	6.22	11.184
5	VGG16_fixed_feature	0.8544	0.8577	0.8544	0.8367	7.16	134.322

The fine-tuned ResNet18 achieves the best Macro F1, indicating that allowing partial backbone adaptation provides a stronger domain fit than training only the classification head. Despite VGG16 having far more parameters, ResNet18 achieves slightly higher Macro F1, suggesting that the residual architecture provides better generalization under limited data and subtle inter-class variance.

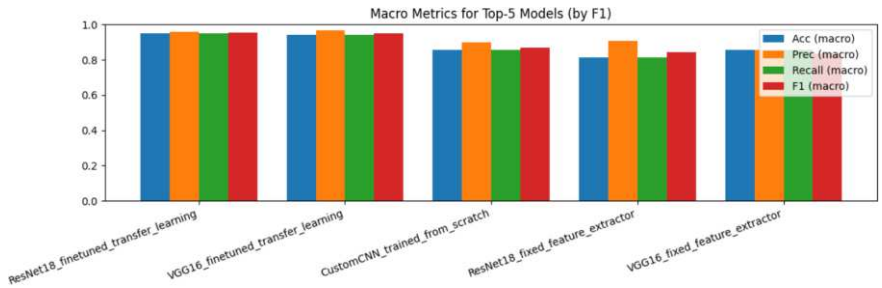


Fig. 1 MangoImageBD performance comparison plot (Macro F1 / Accuracy / efficiency)

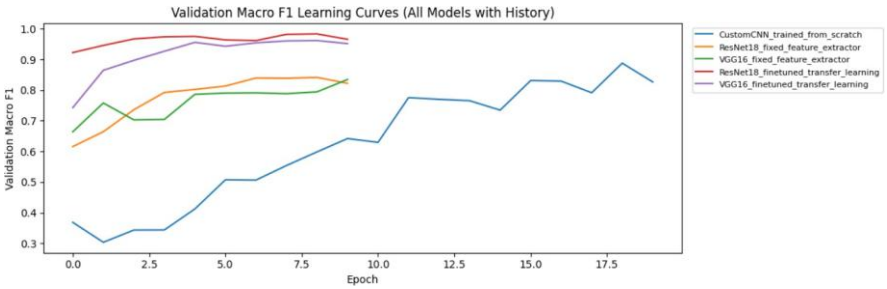


Fig. 2 MangoImageBD F1 Learning Curves

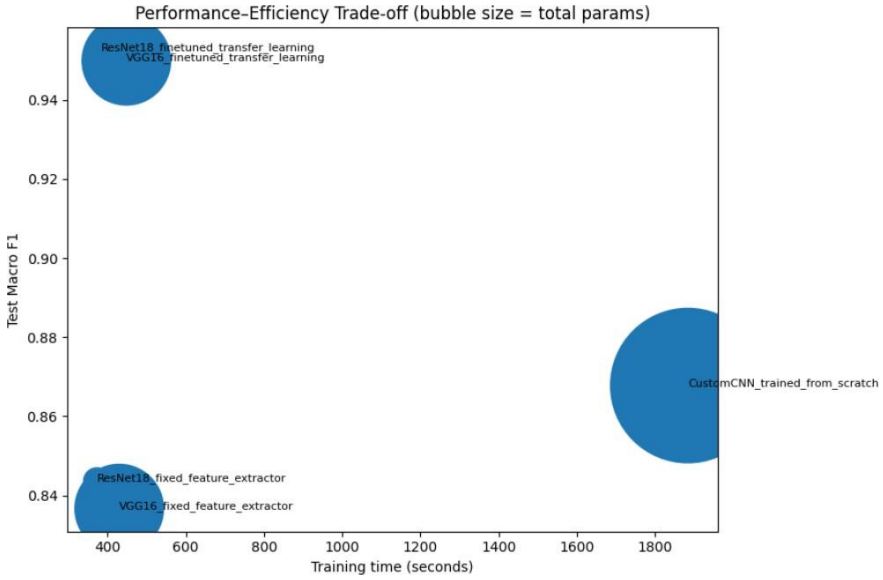


Fig. 3 MangoImageBD Performance Efficiency Trade-off

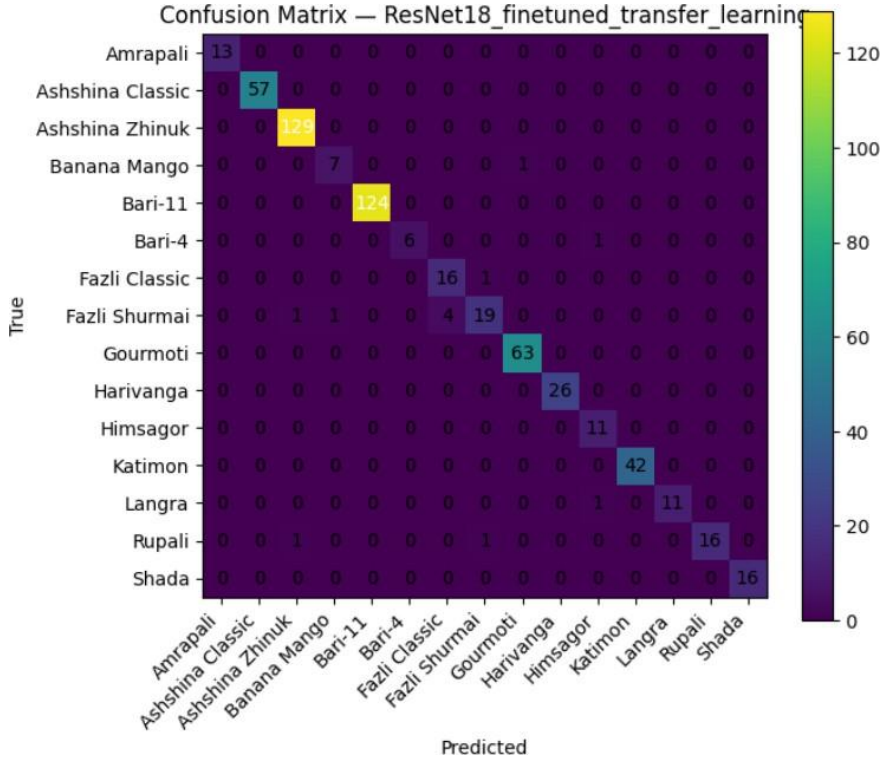


Fig. 4 MangoImageBD confusion Matrix (Best Model)

6.2 Road Damage and Manhole Detection Dataset

Road Damage and Manhole Detection Dataset is a binary classification task distinguishing *Damaged Roads* vs *Good Roads*. The dataset is relatively small (Train/-Val/Test = 314/68/68), which creates a high risk of variance and overfitting. In such regimes, transfer learning typically dominates because pretrained backbones reduce sample complexity.

Table 3 Comparison of CNN models on the Road Damage and Manhole Detection dataset

Model	Accuracy	Precision	Recall	F1	Train ⌶(min)	#Params
VGG16 finetuned transfer learning	1.0000	1.0000	1.0000	1.0000	10.49	134.269
ResNet18 finetuned transfer learning	1.0000	1.0000	1.0000	1.0000	10.35	11.178
ResNet18 fixed feature extractor	0.9737	0.9900	0.9737	0.9814	10.76	11.178
CustomCNN trained from scratch	0.9533	0.9396	0.9533	0.9461	22.74	411.433
VGG16 fixed feature extractor	0.9329	0.8982	0.9329	0.9127	10.79	134.269

A perfect test score (Macro F1 = 1.0) is achieved by both fine-tuned backbones. While this is the strongest possible result, it must be interpreted carefully given the small test size (68 images). A plausible explanation is that the two classes may be highly separable with consistent visual signals (surface cracks, texture discontinuity, irregular edges), allowing pretrained models to fully generalize. However, a second explanation is that the test set may not contain the most challenging borderline cases; therefore, the result should be discussed as *excellent but potentially optimistic* and ideally validated under repeated splits or cross-validation.

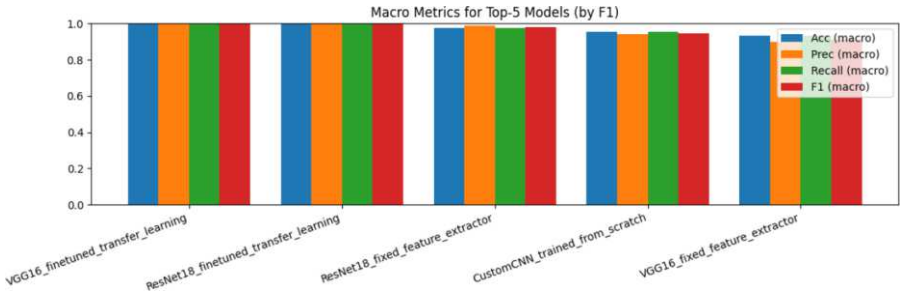


Fig. 5 RoadDamage performance comparison plot (Macro F1 / Accuracy / efficiency)

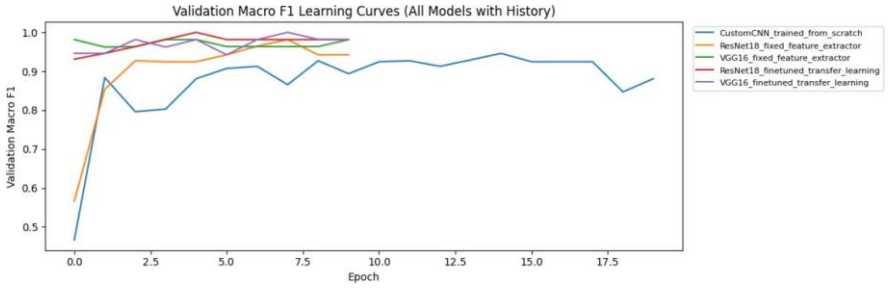


Fig. 6 RoadDamage F1 Learning Curves

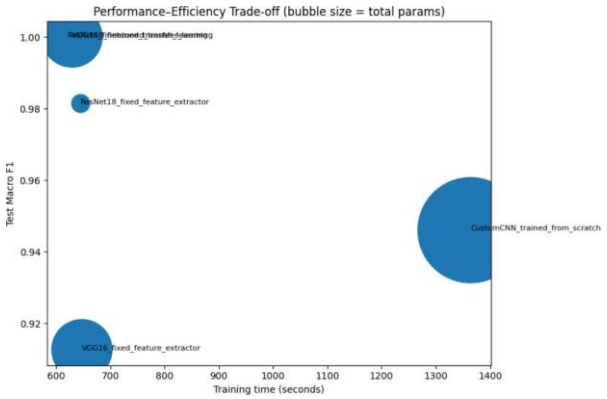


Fig. 7 RoadDamage Performance Efficiency Trade-off

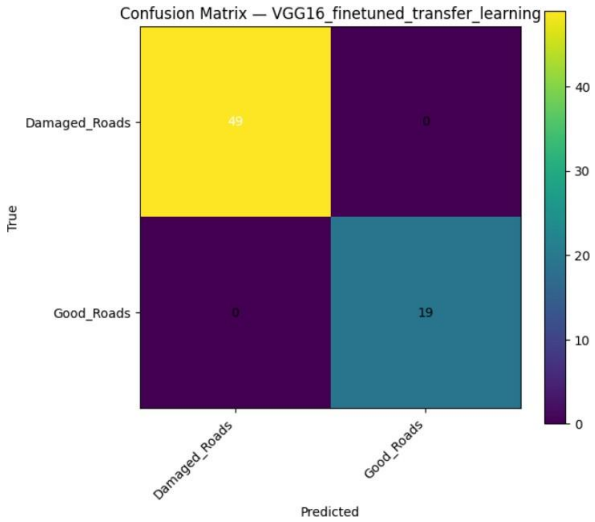


Fig. 8 RoadDamage Confusion matrix (Best Model)

6.3 FootpathVision Dataset

FootpathVision is a binary classification task where higher-level semantic cues (blocked walking space, presence of stalls/objects, crowd and scene layout) are strong discriminators. As a result, even fixed feature extractors perform well, because pretrained backbones already represent objects and scenes effectively.

Table 4 Comparison of CNN models on the FootpathVision dataset

Model	Accuracy	Precision	Recall	F1	Train ⌶(min)	#Params
ResNet18_finetuned_transfer_learning	0.88695	0.88848	0.88695	0.88769	26.96	11.178
VGG16 finetuned ⊥transfer ⌋earning	0.88316	0.89193	0.88316	0.88673	27.68	134.269
ResNet18 _fixed _feature _extractor	0.85576	0.87259	0.85576	0.86156	27.13	11.178
VGG16 fixed _feature _extractor	0.85420	0.84668	0.85420	0.84857	27.79	134.269
CustomCNN trained from _scratch	0.82870	0.82292	0.82870	0.82479	56.58	411.433

Compared to RoadDamage and Paddy, the performance gap between fixed extractor and fine-tuning is smaller, which supports the hypothesis that generic pretrained features already capture most discriminative patterns for this dataset, and fine-tuning only provides incremental improvements.

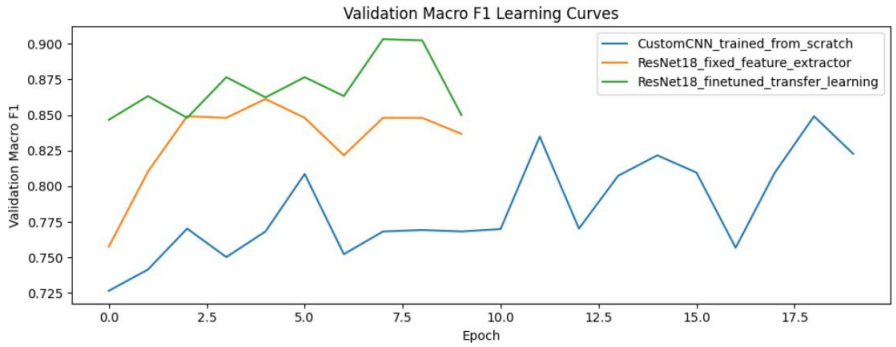


Fig. 9 FootpathVision F1 Learning Curves

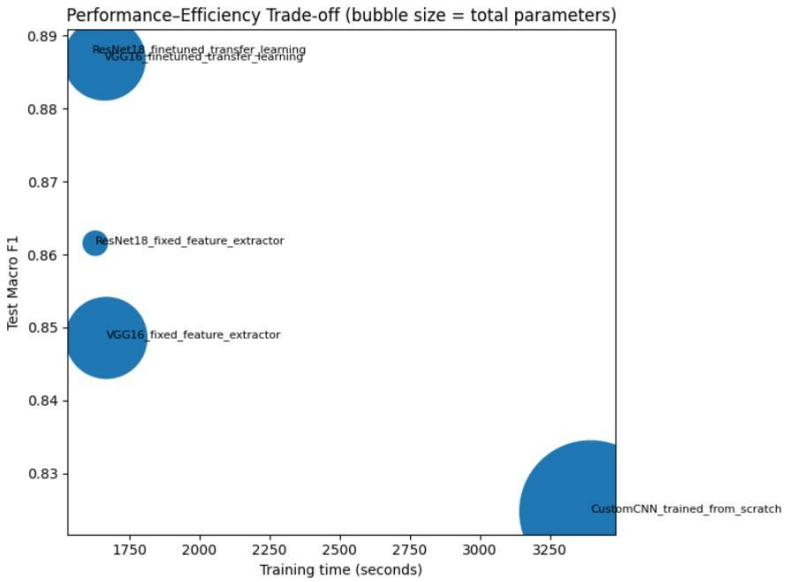


Fig. 10 FootpathVision Performance Efficiency Trade-off

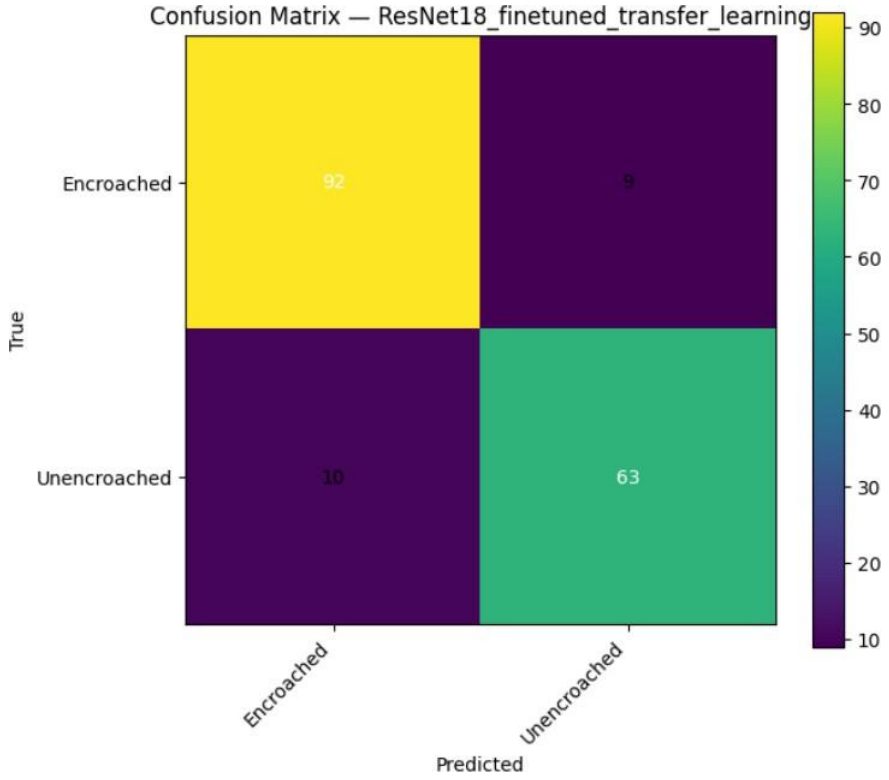


Fig. 11 FootpathVision confusion Matrix

6.4 PaddyVarietyBD Dataset

PaddyVarietyBD is a 35-class fine-grained classification dataset with high inter-class similarity. This setting amplifies the importance of strong feature hierarchies and domain adaptation. Training from scratch generally struggles because the model must learn both generic visual primitives and fine-grained discriminators from limited domain-specific samples.

Table 5 Comparison of CNN models on the PaddyVarietyBD dataset

Model	Accuracy	Precision	Recall	F1	Train Time (min)	#Params
ResNet18_finetuned_transfer_learning	0.8800	0.8875	0.8800	0.8796	19.48	11.194
VGG16_finetuned_transfer_learning	0.8571	0.8755	0.8571	0.8575	21.64	134.404
CustomCNN_trained_from_scratch	0.6471	0.6923	0.6471	0.6287	39.78	0.398
ResNet18_fixed_feature_extractor	0.5914	0.6460	0.5914	0.5906	19.16	11.194
VGG16_fixed_feature_extractor	0.5150	0.5728	0.5150	0.5035	21.20	134.404

The jump from scratch (Macro F1 ≈ 0.63) to fine-tuned ResNet18 (Macro F1 ≈ 0.88) is substantial and indicates that pretrained transfer is essential for high-class-count fine-grained tasks. Additionally, fixed extractors underperform because they cannot sufficiently adapt to dataset-specific cues (grain texture patterns, lighting conditions, subtle phenotype differences), while fine-tuning enables representation alignment.

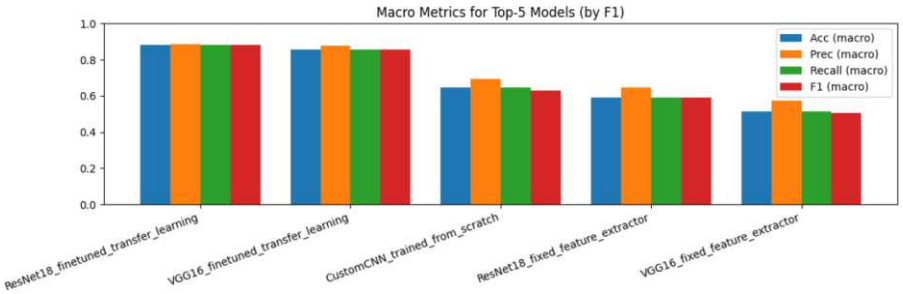


Fig. 12 PaddyVarietyBD performance comparison plot (Macro F1 / Accuracy / efficiency)

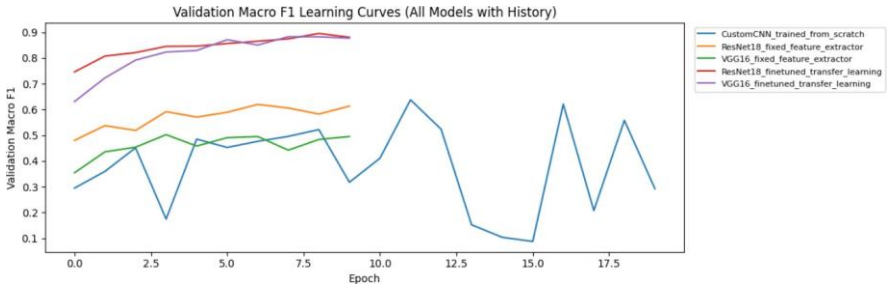


Fig. 13 PaddyVarietyBD F1 Learning Curves

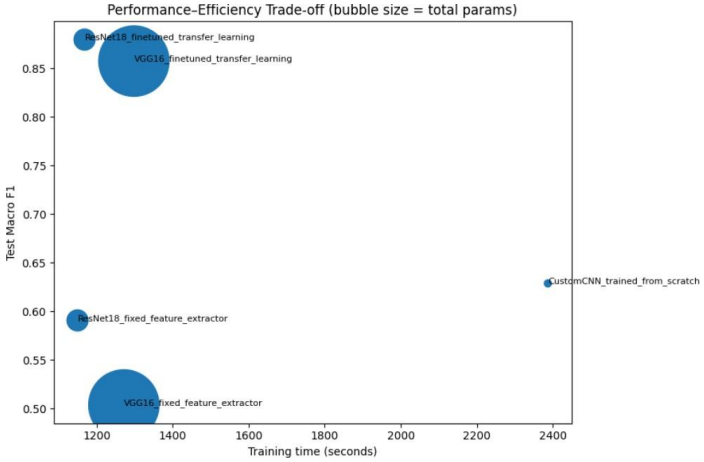


Fig. 14 PaddyVarietyBD Performance Efficiency Trade-off

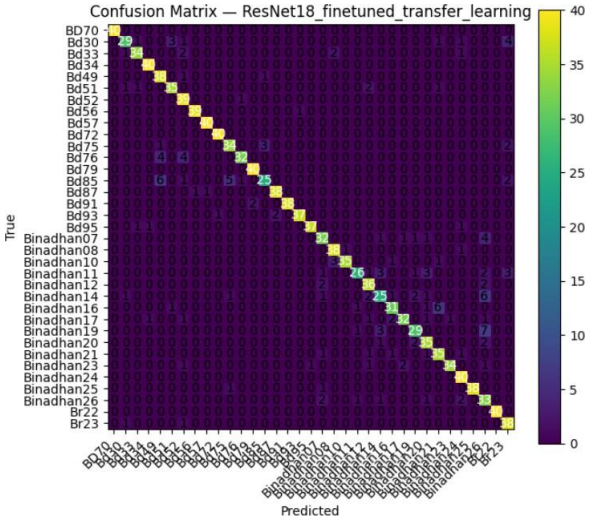


Fig. 15 PaddyVarietyBD Confusion matrix (Best Model)

6.5 Unauthorized Vehicle Detection Dataset

The Unauthorized Vehicle Detection Dataset is binary with crop-sampling applied (increasing training sample count via multiple crops per image). While this boosts effective data size, it also introduces spatial correlation between samples, requiring careful generalization.

Table 6 Comparison of CNN models on the Unauthorized Vehicle Detection Dataset

Model	Accuracy	Precision	Recall	F1	Train ⌶(min)	#Params
VGG16 finetuned transfer learning	0.8382	0.8335	0.8382	0.8358	34.30	134.269
ResNet18 finetuned transfer learning	0.8304	0.8172	0.8304	0.8233	34.74	11.178
ResNet18 fixed feature extractor	0.7814	0.7668	0.7814	0.7734	33.59	11.178
VGG16 fixed feature extractor	0.7327	0.7234	0.7327	0.7277	37.43	134.269
CustomCNN trained from scratch	0.7134	0.6976	0.7134	0.7041	52.14	411.433

Vehicle results show that fine-tuning yields measurable gains, and VGG16 slightly outperforms ResNet18. A reasonable interpretation is that VGG16's dense convolutional stacking can emphasize local texture and edge responses that may be highly predictive in cropped vehicle imagery.

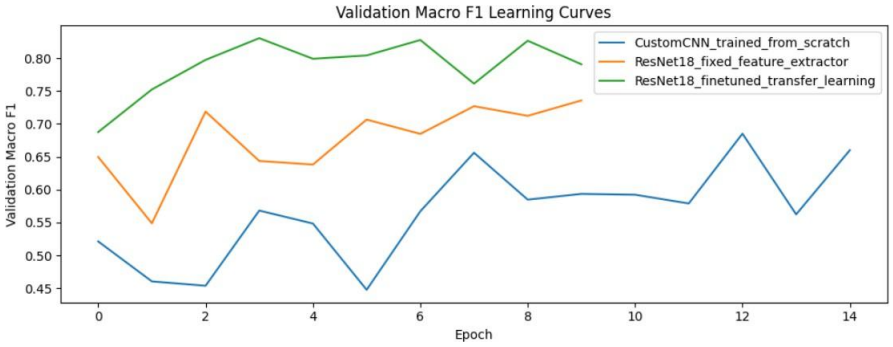


Fig. 16 Vehicle Detection F1 Learning Curves

7 Discussion

7.1 Custom CNN Hyperparameter Search: Learning Rate and Dropout

For the Custom CNN, we tuned two high-impact hyperparameters: learning rate (η) and dropout probability (p). This search is necessary because training from scratch is significantly more sensitive to optimization settings than pretrained transfer learning.

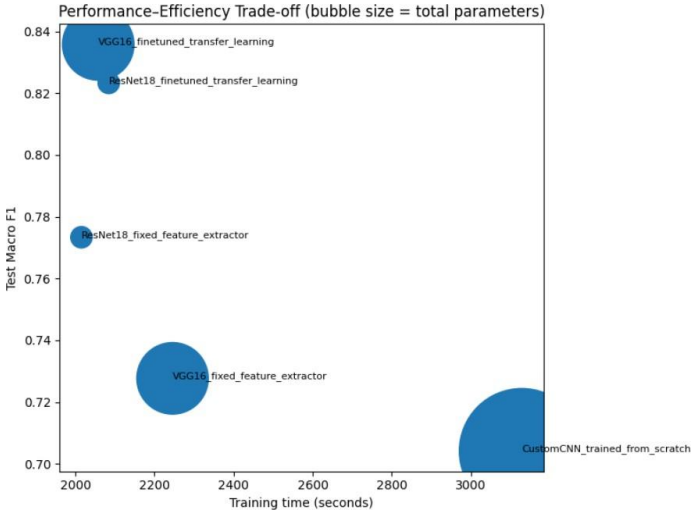


Fig. 17 Vehicle Detection Performance Efficiency Trade-off

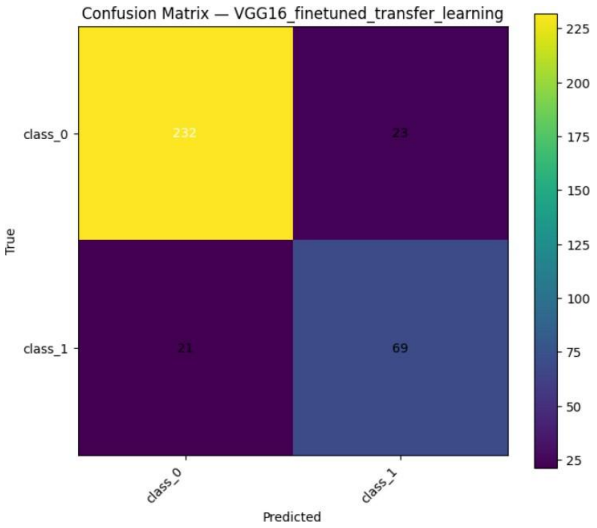


Fig. 18 Vehicle Detection Confusion matrix (Best Model)

Across datasets, a higher learning rate (e.g., η = 0.001) typically accelerates loss reduction early but can destabilize convergence, especially when dataset is small or highly fine-grained. A lower learning rate (e.g., η = 0.0005) tends to improve stability but can underfit if training budget is limited.

Dropout acts as the principal regularizer by preventing co-adaptation of feature detectors. In our experiments, larger dropout (e.g., p = 0.5) was generally beneficial in reducing overfitting, particularly for multi-class (Paddy) and smaller datasets (Road-Damage), where memorization is likely. For example, in RoadDamage validation logs,

the configuration ($\eta = 0.0005$, $p = 0.5$) reached the highest peak validation Macro F1 (≈ 0.9461), indicating strong generalization under regularization. Similarly, for Footpath the best Custom CNN configuration was also found to be stable under dropout $p = 0.5$.

7.2 Why Pretrained CNN Models Were Trained for Only 10 Epochs

Pretrained backbones converge quickly because they start from weights already optimized on large-scale datasets, providing a strong initialization. In practice, validation performance frequently saturates within a small number of epochs. Training beyond this point tends to yield diminishing returns and can even reduce validation Macro F1 due to overfitting, especially when the downstream dataset is small (RoadDamage) or when only the classifier head is trainable (fixed feature extractor).

Therefore, a 10-epoch training budget was chosen as a principled compromise: (i) enough time for the classifier head (and unfrozen blocks in fine-tuning) to adapt, (ii) short enough to reduce overfitting risk, and (iii) computationally efficient across five datasets and multiple model variants.

7.3 Dataset-wise Interpretation of Transfer Learning Benefits

The magnitude of improvement from pretrained transfer learning is dataset dependent. For RoadDamage (small binary), pretrained fine-tuning achieved perfect test performance; while impressive, it also highlights that for very small datasets, results must be interpreted cautiously and ideally revalidated under repeated splits. For Footpath and Vehicle (binary), gains exist but are moderate because pretrained features already capture most semantic content. For Paddy (35-class), pretrained fine-tuning is transformative, because the dataset requires fine-grained discrimination across many similar classes.

7.4 Limitations

This study has several limitations that should be considered when interpreting the results. First, for smaller datasets such as Road Damage and Manhole Detection, performance was evaluated using a single fixed train–validation–test split, which may lead to optimistic estimates; repeated random splits or cross-validation would provide stronger statistical confidence. Second, the analysis focuses on two widely used pretrained backbones (ResNet-18 and VGG-16), and conclusions may not directly generalize to newer architectures with different capacity or inductive biases. Third, efficiency comparisons are based on wall-clock training time measured under a specific hardware configuration and may vary across platforms. Finally, the study is limited to image classification tasks; extending the evaluation to detection or segmentation would offer additional insights into the generality of the observed trends.

8 Conclusion

This study systematically evaluated five CNN regimes across five heterogeneous image datasets: MangoImageBD, RoadDamage, FootpathVision, PaddyVarietyBD, and Unauthorized Vehicle Detection. Across datasets, fine-tuned pretrained models consistently achieved the strongest generalization and the highest Macro F1, confirming that transfer learning is the most effective strategy for practical vision tasks under limited labeled data. Fixed feature extractors performed competitively in binary, semantic tasks but underperformed in fine-grained multi-class settings where domain adaptation is necessary. Custom CNN models trained from scratch were generally weaker, highlighting the sample-efficiency advantage of pretrained representations. Future work should extend these findings with repeated random splits or cross-validation for small datasets (notably RoadDamage), and incorporate robustness analysis under lighting/viewpoint shifts to better reflect deployment conditions.

References

[1] Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems (2012)

[2] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations (2015)

[3] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)

[4] Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? Advances in Neural Information Processing Systems (2014)

[5] Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering (2010)

[6] Sukanto, S.D., Roy, D., Shakil, F., Singha, N., Asik, A., Joarder, A., Fuad, M.M.N., Ibrahim, M.: Detecting unauthorized vehicles using deep learning for smart cities: A case study on bangladesh. arXiv preprint arXiv:2510.26154 (2025)

[7] Hossen, R., Mistry, D., Rahman, M., Hridoy, W.A.S.A.R., Saha, S., Ibrahim, M.: Road damage and manhole detection using deep learning for smart cities: A polygonal annotation approach. arXiv preprint arXiv:2510.03797 (2025)

[8] Lubaina, A., Pahlowan, M.E.U., Munni, T.I., Ibrahim, M.: Footpathvision: A comprehensive image dataset and deep learning baselines for footpath encroachment detection. Technical report, SSRN (2025). Available at SSRN: https://papers.ssrn.com/sol3/papers.cfm?abstract id=5577201

[9] Ferdaus, H., Prito, R.H., Ahmed, M., Islam, M.M., *et al.*: Mangoimagebd: An extensive mango image dataset for identification and classification of various mango varieties in bangladesh. Data in Brief **62**, 111908 (2025)

[10] Tahsin, M., Ibrahim, M., Nafisa, A.T., *et al.*: Paddyvarietybd: Classifying paddy variations of bangladesh with a novel image dataset. Data in Brief **60**, 111514 (2025)