Synergizing residual and dense architectures for fine-grained oil palm grading a deep feature concatenation approach

Luo, Yang; Majeed, Anwar P.P. Abdul ; Omar, Zaid; Jagtap, Sandeep; García-Garcia, Guillermo; Chen, Yi

Published in:
Mathematics

DOI:
10.3390/math14050769

2026

Document Version:
Publisher's PDF, also known as Version of record

mathematics

MDPI

Article

Synergizing Residual and Dense Architectures for Fine-Grained Oil Palm Grading: A Deep Feature Concatenation Approach

Yang Luo [1,2], Anwar P. P. Abdul Majeed [3,*], Zaid Omar [4], Sandeep Jagtap [5], Guillermo Garcia-Garcia [6] and Yi Chen [1,*]

1 School of Intelligent Manufacturing Ecosystem, Xi'an Jiaotong-Liverpool University, Taicang, Suzhou 215400, China; yang.luo@xjtlu.edu.cn
2 Industrial Intelligent Agent Research Center (IIARC), Xi'an Jiaotong-Liverpool University, Taicang, Suzhou 215400, China
3 Faculty of Engineering and Technology, Sunway University, Bandar Sunway, Selangor 47500, Malaysia
4 Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia; zaidomar@utm.my
5 Division of Engineering Logistics, Faculty of Engineering, Lund University, 22100 Lund, Sweden; sandeep.jagtap@tlog.lth.se
6 Department of Chemical Engineering, Faculty of Sciences, University of Granada, Avda. Fuente Nueva, 18071 Granada, Spain; guillermo.garcia@ugr.es
* Correspondence: anwarm@sunway.edu.my (A.P.P.A.M.); yi.chen02@xjtlu.edu.cn (Y.C.)

Abstract

Accurate grading of Oil Palm Fresh Fruit Bunches (FFB) is pivotal for maximizing agricultural yield, yet manual assessment in unstructured environments remains labor-intensive and subjective. While Convolutional Neural Networks (CNNs) offer an automated solution, the conventional strategy of scaling network depth often yields diminishing returns or overfitting on moderately sized datasets. To overcome these limitations, this study proposes the Deep Feature Concatenation (DFC) framework. Rather than deepening a single architecture, this methodology synergizes the spatial hierarchy preservation of ResNet50 with the dense feature-reuse mechanisms of DenseNet121. This fusion creates a composite representation space that captures complementary inductive biases. To ensure computational efficiency, the framework decouples representation learning from inference. Principal Component Analysis (PCA) retains 99% of explained variance while compressing features by 68%. These optimized representations are classified using shallow linear probes. Validated on a single-source dataset expanded to 4000 images (derived from 466 original samples) using a rigorous "Parent–Child" split to prevent data leakage, DFC achieved a peak accuracy of 97.75%. McNemar's statistical test indicated that this performance outperforms the ResNet50 baseline ($p = 0.039$) for SVM classifiers. However, it is critical to note that these results represent a proof of concept based on a limited biological sample size, particularly for rare defect classes. While the model achieved 100% detection accuracy for critical defects within the specific validation set, the high synthetic-to-original ratio necessitates cautious interpretation regarding external validity. This framework provides a practical foundation for future research into high-precision, low-latency grading systems, but multi-center validation on larger, independent datasets is required to confirm broad generalizability across diverse plantation environments.

Keywords: oil palm FFB; deep feature concatenation; hybrid transfer learning; complementary feature fusion; precision agriculture; convolutional neural networks

MSC: 68T07; 62H25; 68T10

Academic Editors: Mingbo Zhao, Haijun Zhang, Zhou Wu and Jianghong Ma

Received: 24 December 2025
Revised: 19 February 2026
Accepted: 23 February 2026
Published: 25 February 2026

1. Introduction

The global palm oil industry is a critical pillar of Southeast Asia's agricultural economy, particularly in Indonesia and Malaysia [1]. With a market valuation exceeding USD 50 billion [2], this sector supports the livelihoods of millions. The economic viability of the industry is intrinsically linked to the Oil Extraction Rate (OER), a metric that is acutely sensitive to the post-harvest quality of Fresh Fruit Bunches (FFB) [3,4]. Currently, the industry standard for grading remains inherently variable and relies heavily on labor-intensive, hazardous, and fundamentally subjective manual processes [5]. Human graders are susceptible to fatigue and cognitive bias, and they frequently misclassify bunches based on inconsistent visual cues [6]. This leads to the processing of unripe or defective fruits, which significantly degrades crude palm oil quality and overall yield [5]. This inefficiency is exacerbated by acute labor shortages and logistical disruptions, precipitated by events such as the COVID-19 pandemic. Consequently, there is an urgent need to transition toward automated, objective quality assessment systems [7].

Automated FFB classification aims to standardize this process by employing sensing technologies and computational intelligence [8]. While recent progress in this domain is remarkable, two fundamental challenges persist in the transition from laboratory prototypes to field deployment: (1) the trade-off between computational tractability and classification robustness in unstructured environments [9], and (2) the limited availability of annotated defect datasets required to train deep architectures from scratch. Existing computational approaches to FFB grading generally fall into three methodological categories: (a) proximal sensing and spectroscopy [3,10,11], (b) conventional machine vision with hand-crafted features [12–14], and (c) end-to-end deep learning architectures [7,15].

Early attempts to standardize FFB grading relied heavily on proximal sensing to correlate physicochemical properties, such as oil content and moisture, with spectral signatures. Saeed et al. [16] pioneered the use of a portable four-band optical sensor system combined with quadratic discriminant analysis. They demonstrated that spectral reflectance could distinguish maturity stages with over 85% accuracy in field conditions. Similarly, Hazir et al. [17] explored fluorescence spectroscopy and utilized the blue-to-red fluorescence ratio to quantify flavonoid and anthocyanin content. Their work established that biological changes in the mesocarp could be non-destructively detected and achieved an overall accuracy of 89.7% [18]. While these spectroscopic methods offer high theoretical fidelity, Makky et al. [19] argued that reliance on specialized, expensive instrumentation limits their widespread adoption among smallholders. Consequently, they developed an on-site machine vision grading machine using adaptive thresholding, which shifted the focus from internal chemical analysis to external visual inspection.

To overcome the hardware constraints of spectroscopy, research shifted toward conventional machine vision, which relies on handcrafted feature extraction. Alfatni et al. [12] proposed a rule-based expert system that analyzed statistical color features across multiple Regions of Interest (ROIs) and achieved 94% accuracy by mimicking human visual inspection. Addressing the limitations of simple Red, Green, Blue (RGB) data, Septiarini et al. [20] demonstrated that transforming images into Luminance-In-phase-Quadrature (YIQ) and Luminance-Chrominance (YCbCr) color spaces, followed by Principal Component Analysis (PCA) and artificial neural networks, could yield accuracies as high as 98.3% by isolating luminance from chrominance [21].

However, reliance on color alone proved susceptible to errors under variable outdoor illumination. To mitigate this, Ghazalli et al. [22] investigated texture descriptors and specifically compared bag-of-visual-words with statistical color features. Their findings revealed that texture-based approaches (70% accuracy) significantly outperformed color-only methods (57%) in uncontrolled lighting environments. Despite these improvements,

Rosbi et al. [9] contended that traditional segmentation remains a bottleneck. They proposed a hybrid pipeline using fast fuzzy C-means clustering and opposite-colour local binary patterns. This showed that integrating texture with advanced segmentation could achieve 93.68% accuracy, even with limited datasets.

The advent of deep learning marked a paradigm shift from hand-crafted features to automated hierarchical feature learning. Early research prioritized comparative analyses of architectural depth. For instance, Herman et al. [23] compared DenseNet-121 with AlexNet and found that the dense connectivity in DenseNet-121 enabled superior gradient flow, leading to 8.5% higher accuracy on imbalanced datasets. Simultaneously, Suharjito et al. [7] addressed the computational constraints of field deployment by evaluating lightweight architectures. Specifically, they tested EfficientNet-B0 and MobileNet, achieving 89.3% accuracy with an inference time suitable for mobile devices.

A critical divergence in the literature exists between image classification and object detection approaches. Khamis et al. [24] provided a comparative analysis of ResNet50 (classification) and You Only Look Once (YOLO) v3 (detection). They concluded that object detection models are superior in field settings because they localize the fruit. This effectively filters out background noise, such as fronds, sky, and ground, that often confuses classification-only models like ResNet. Building on this, Junos et al. [25] developed YOLO-P, an optimized YOLOv3-tiny model with a DenseNet backbone, and achieved a mean Average Precision (mAP) of 98.91% while maintaining a lightweight footprint (76 MB). Similarly, Lai et al. [26] deployed YOLOv4 for real-time robotic harvesting, achieving a mAP of 87.9%.

While detection models like YOLO are robust to background noise, they introduce significant computational overhead and require laborious bounding-box annotation. Conversely, end-to-end classification models often lack transparency and computational efficiency. An emerging yet promising direction was highlighted by Luo et al. [8], who demonstrated that a feature-based transfer learning pipeline achieved 96.81% accuracy. By using EfficientNet-B4 as a fixed feature extractor coupled with a Logistic Regression (LR) classifier, they suggested that the heavy computation of full network retraining may be unnecessary.

However, a critical gap remains: existing studies largely treat feature extractors as monolithic entities, ignoring the potential of complementary feature fusion. While Herman et al. [23] proved the value of dense connections and Khamis et al. [24] utilized residual connections, no study has systematically fused these distinct architectures (Residual + Dense) to maximize feature diversity for defect detection. This study builds upon the efficient transfer learning ethos of Luo et al. [8] but extends it by proposing the Deep Feature Concatenation (DFC) framework. This framework leverages the complementary strengths of diverse CNN architectures to achieve robustness without the computational cost of object detectors.

The specific contributions of this study to the field of precision agriculture and computer vision are as follows:

- Rigorous Experimental Validation: Unlike studies that rely on random splitting, which may introduce data leakage, a rigorous data augmentation strategy followed by a grouped splitting protocol and hold-out cross-validation (70:15:15) is employed. This is tested on a dataset containing complex, real-world visual noise to establish a reliable benchmark for robust, automated FFB grading systems.
- Development of a Hybrid Feature-Based Transfer Learning Pipeline: A novel pipeline that integrates distinct CNN topologies (Residual and Dense connections) as fixed feature extractors is introduced. This demonstrates that feature diversity yields greater robustness than a single deep architecture for FFB defect detection.

- Dimensionality Optimization via PCA: PCA is implemented to systematically reduce the high-dimensional feature vectors generated by deep networks. It has been shown that a compact feature subspace can maintain accuracy exceeding 96% while significantly reducing computational overhead for the classifier.

The remainder of this article is organized as follows. Section 2 details the *Materials and Methods*, including the dataset curation, the proposed hybrid DFC architecture, and the manifold optimization pipeline. Section 3 presents the *Results and Discussion*, providing a comprehensive analysis of the performance benchmarks, ablation studies, the limitations of the current framework, and future research directions. Finally, Section 4 summarizes the key findings and outlines the wider implications of this research in the *Conclusions*.

2. Materials and Methods

2.1. Dataset Acquisition

The primary dataset used in this study comprises 466 high-resolution images of FFB from the Tenera variety. To ensure the development of a model capable of generalizing across unstructured agricultural environments, data were collected in situ across multiple commercial palm oil plantations in Johor, Negeri Sembilan, and Perak, Malaysia. Images were captured under natural outdoor lighting conditions to incorporate real-world variations, including shadows, partial frond occlusion, and varying spectral intensities due to diurnal shifts. Details on how the images in the dataset were collected can be found in [27].

Ground-truth labeling was performed by an expert grader to classify FFBs into five distinct categories based on maturity and physical integrity. Representative samples of these classes are illustrated in Figure 1. The distribution of the raw dataset was highly imbalanced, reflecting the natural prevalence of specific ripeness stages in a plantation environment. The class definitions are as follows:

- Ripe (N = 201): Characterized by distinct reddish-orange pigmentation and the natural detachment of fruitlets from the bunch.
- Unripe (N = 164): Defined by deep violet to black pigmentation with no evidence of abscission.
- Overripe (N = 74): Distinguished by dark red pigmentation and greater than 50% empty sockets resulting from extensive fruitlet detachment.
- Damaged (N = 15): Bunches exhibiting physical lesions, pest damage, or structural compromise.
- Empty (N = 12): Bunches with the majority of fruitlets removed or detached, leaving only the stalk.

A critical limitation of the raw dataset was the significant class imbalance (e.g., 15 Damagedsamples versus 201 Ripe samples) and the low total volume, which pose risks of model overfitting and sample scarcity. To address this, we implemented a systematic offline data augmentation pipeline designed to expand the dataset to a uniform 800 images per class.

Standard random splitting of augmented datasets often leads to data leakage, where variations of a training image inadvertently appear in the test set [28]. This induces optimistic bias, allowing the model to memorize specific textural artifacts of a single fruit rather than learning generalized features of ripeness [29]. To enforce methodological rigor, we employed a "Parent–Child" group-split strategy. In this protocol, the original "Parent" image and all its augmented "Children" are treated as an atomic cluster [30]. Partitioning into training (70%), validation (15%), and testing (15%) sets occurs strictly at the group level. Consequently, if a specific FFB appears in the training set, no variation of that specific bunch exists in the validation or test sets. This ensures that the reported accuracy reflects true generalization to unseen biological samples.

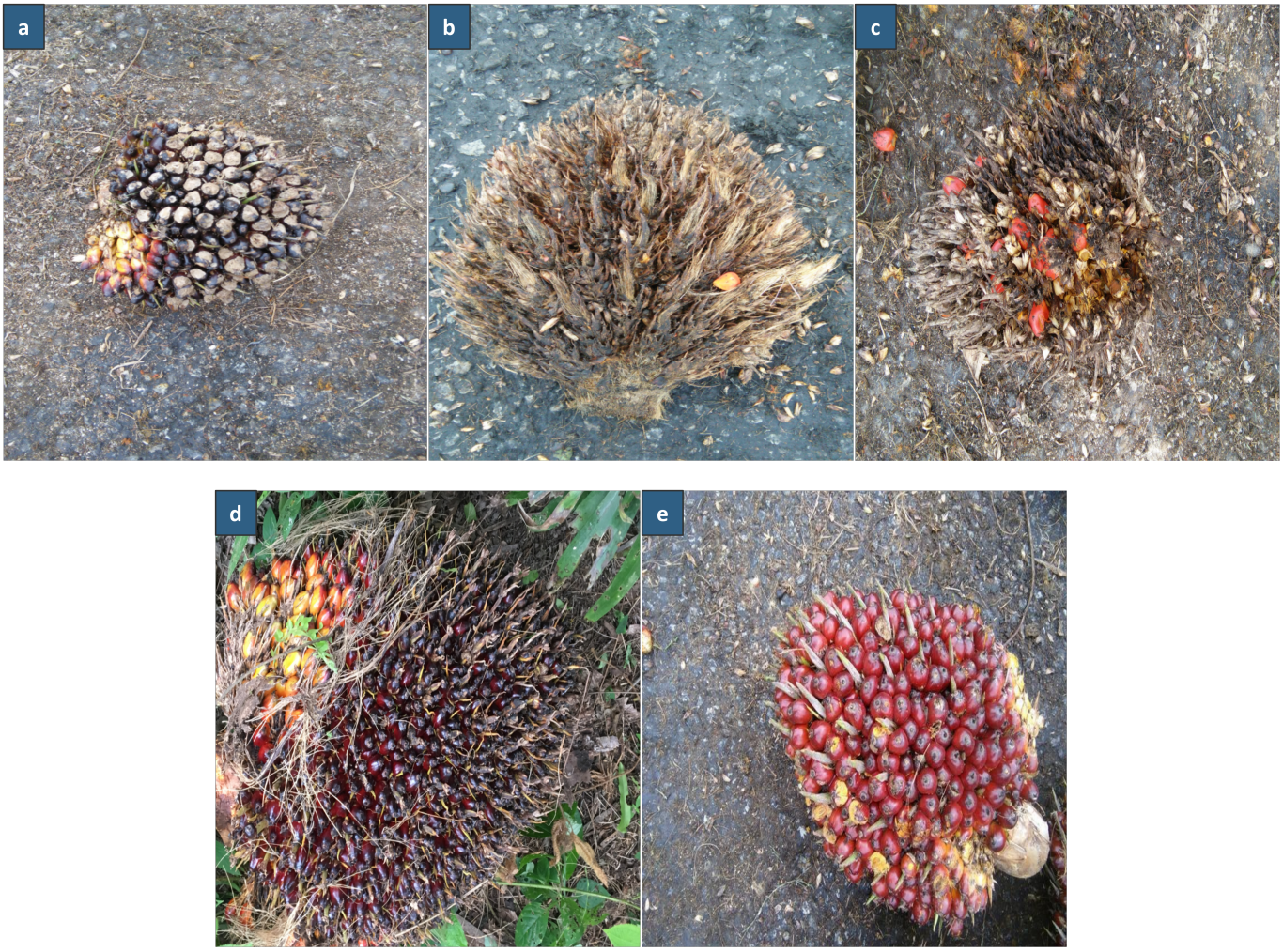

Figure 1. Representative images of oil palm Fresh Fruit Bunch (FFB) ripeness classes: (**a**) Damaged. (**b**) Empty. (**c**) Overripe. (**d**) Ripe. (**e**) Unripe. Each image illustrates the characteristic color and texture associated with its respective ripeness stage.

The applied data augmentation techniques included rotation, horizontal flipping, random cropping, and photometric distortion. These transformations were selected to address inherent field variability while preserving bio-physical features critical for classification:

- **Geometric Transformations:** Rotation ($\pm 15^{\circ}$) and horizontal flipping ($p = 0.5$) were utilized to simulate diverse camera angles and the morphological symmetry of the bunches. This ensures the model learns orientation invariance while respecting the gravitational context of hanging fruits.
- **Scale Invariance:** Random cropping (scale factor $0.85 - 1.0$, $p = 0.7$) emulated varying acquisition distances without compromising the global structural coherence of the bunch.
- **Photometric Adjustments:** Regulated adjustments to brightness (± 15 pixel units) and contrast ($\pm 10\%$) simulated diurnal lighting fluctuations and shadowing effects common in outdoor plantations. These were tightly constrained to prevent the washout of specific chromatic indicators essential for distinguishing ripeness stages.

Unlike standard pipelines, high-resolution integrity was maintained throughout the process to mitigate upscaling artifacts during feature extraction. This process yielded a balanced final dataset of 4000 images (800 per class), providing a statistically significant volume for deep feature extraction. The distribution of image counts before and after augmentation across the respective subsets is detailed in Table 1.

Table 1. Distribution of the oil palm dataset, detailing the allocation of original and augmented images across the training, validation, and testing partitions for each grading category.

FFB Class	Data Split	Original	Augmented	Total
Damaged	Train	10	525	535
	Validation	2	104	106
	Test	3	156	159
Empty	Train	8	525	533
	Validation	2	131	133
	Test	2	132	134
Overripe	Train	51	502	553
	Validation	11	106	117
	Test	12	118	130
Ripe	Train	140	417	557
	Validation	30	89	119
	Test	31	93	124
Unripe	Train	114	443	557
	Validation	25	98	123
	Test	25	95	120

2.2. Model Training and Evaluation

To address the inherent complexities of outdoor oil palm fruit grading, which is characterized by unstructured environments and high intraclass variance, this study implements a DFC framework. A schematic representation of this methodology is provided in Figure 2.

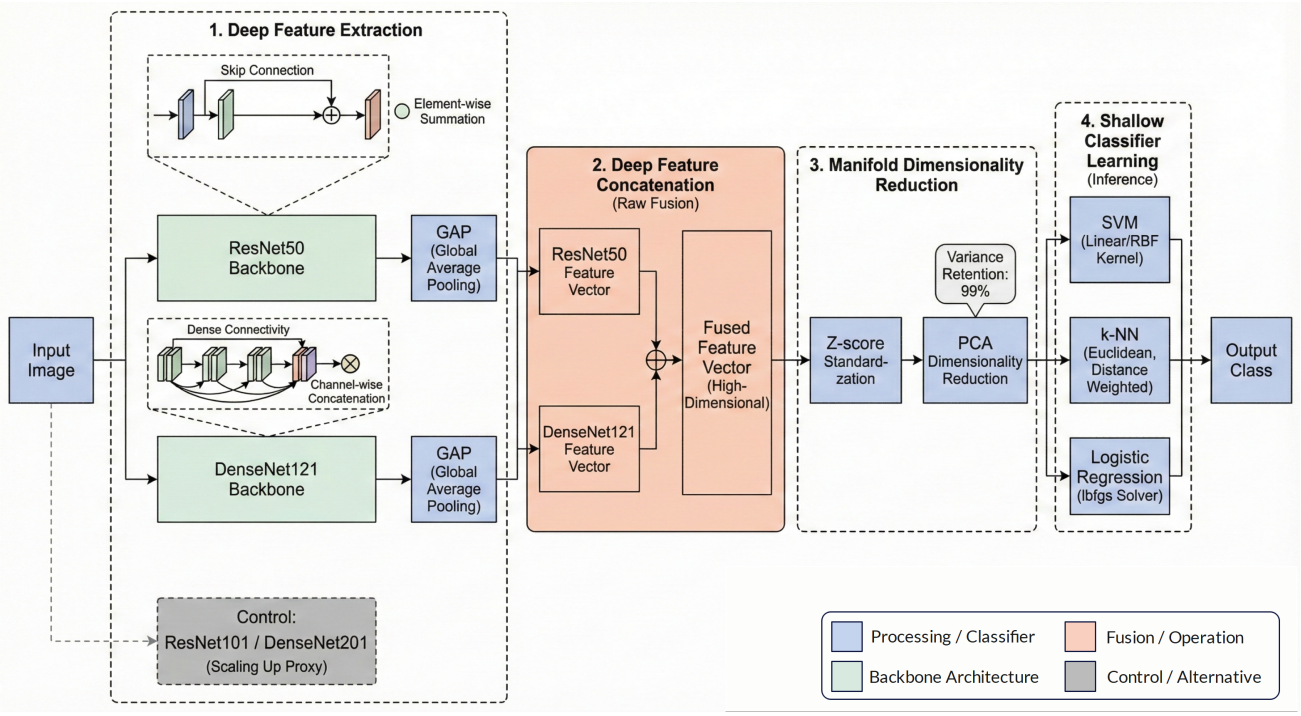

Figure 2. Illustration of the proposed pipeline. Distinct CNN architectures possess complementary inductive biases; residual connections facilitate the preservation of spatial hierarchy (ResNet), whereas dense connectivity maximizes feature reuse and signal integrity (DenseNet). Heterogeneous feature vectors are extracted in parallel and fused to generate a composite representation space containing high-density semantic and textural information. Manifold optimization via Principal Component Analysis (PCA) is subsequently applied to mitigate high dimensionality while retaining 99% of explained variance for efficient linear probing. "⊕" denotes the feature concatenation operator.

The architecture of this pipeline challenges the conventional deep learning paradigm. This paradigm typically posits that increasing network depth and parameter count is the primary trajectory for performance optimization in fine-grained agricultural domains [31]. Instead, we leverage the distinct inductive biases and representation mechanisms of differing architectures [32]. We hypothesize that fusing features from networks with contrasting information processing strategies, specifically, the skip-connection mechanisms of residual networks versus the feature-reuse mechanisms of dense connectivity, yields a composite representation space that is significantly more discriminative than deepening a single architecture [33].

Accordingly, the proposed pipeline decouples data curation from representation learning and inference, structured into four sequential stages:

1. Deep Feature Extraction: Input images are processed in parallel through pre-trained CNN models possessing the aforementioned complementary biases.
2. Deep Feature Concatenation: These heterogeneous feature vectors are fused to maximize information density.
3. Manifold Dimensionality Reduction: This is employed to mitigate the overfitting risks and isolate the principal variances.
4. Shallow Classifier Learning: This stage utilizes low-complexity algorithms (SVM, kNN, LR) to ensure computational efficiency without sacrificing classification accuracy.

2.2.1. Deep Feature Extraction

To construct a robust representation space for outdoor fruit grading, we employ a comparative transfer learning framework that leverages pre-trained ImageNet models. The selection of architectures is designed to isolate the performance gains attributable to architectural diversity versus those attributable to architectural scaling.

We select ResNet50 and DenseNet121 as the primary backbones for the proposed hybrid framework. These models represent two distinct inductive biases in deep learning topology:

- ResNet50: This architecture utilizes residual skip-connections, formally defined as:

$$x_l = H_l(x_{l-1}) + x_{l-1} \tag{1}$$

  This mechanism facilitates identity mapping and gradient propagation via element-wise summation. The structure is particularly effective at preserving spatial hierarchies essential for defining object morphology [33].
- DenseNet121: This architecture utilizes dense connectivity, defined as:

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]) \tag{2}$$

  This maximizes feature reuse via channel-wise concatenation. It creates a compact, diverse feature set that improves parameter efficiency [34].

These specific depths (50 and 121 layers) provide a balance of representational power and computational efficiency. They serve as the basis for the subsequent feature fusion stage.

To empirically test the hypothesis that feature-level complementarity yields superior discrimination compared to simply increasing network depth, we introduce ResNet101 and DenseNet201 as control variables. These deeper variants act as proxies for the scaling-up heuristic. By evaluating these models, we establish a performance baseline for single-architecture scaling. This enables a rigorous comparison with the proposed hybrid fusion strategy.

For all four architectures, we truncate the final classification heads. Latent representations are extracted from the Global Average Pooling (GAP) layer, collapsing the spatial feature maps $H \times W \times C$ into a global 1D descriptor vector $\mathbf{x} \in \mathbb{R}^C$. This operation ensures spatial invariance while retaining the high-level semantic density required for the downstream separation of grading classes.

2.2.2. Deep Feature Fusion

After extracting latent descriptors from the foundational encoders, we construct a composite representation space using a DFC protocol. This stage operationalizes the concept of feature-level complementarity by physically merging the distinct feature hierarchies of the residual and dense architectures into a unified vector. We denote the feature vector extracted from the GAP layer of the ResNet50 backbone as $\mathbf{x}_{Res} \in \mathbb{R}^{2048}$ and the corresponding vector from the DenseNet121 backbone as $\mathbf{x}_{Dense} \in \mathbb{R}^{1024}$. To preserve the native distributional characteristics of the pre-trained weights, we employ a raw fusion strategy that applies no prior normalization. The fused representation, $\mathbf{x}_{cat}$, is derived via the standard concatenation operator $\oplus$:

$$\mathbf{x}_{cat} = \mathbf{x}_{Res} \oplus \mathbf{x}_{Dense} \tag{3}$$

This operation yields a high-dimensional composite vector $\mathbf{x}_{cat} \in \mathbb{R}^{3072}$. While this augmented feature space maximizes information density, the resulting dimensionality can lead to feature redundancy. Consequently, this raw fused representation serves as the precursor to the manifold optimization stage.

2.2.3. Manifold Dimensionality Reduction

To mitigate computational latency and evaluate redundancy, we implement a dimensionality reduction stage utilizing PCA [35]. Crucially, the preprocessing order is defined as follows: (1) Extraction of raw features from both backbones, (2) concatenation of the raw vectors to form the composite space, (3) Z-score Standardization of the fused vector, and (4) PCA projection. Prior to projection, the raw joint embedding matrix $\mathbf{X}_{cat}$ (containing all training samples) is subjected to Z-score standardization. This is requisite to prevent features with naturally higher variance from dominating the principal components. We apply the transformation

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{4}$$

where x_{ij} denotes the activation of the j-th feature for the i-th sample. The statistics μ_j (mean) and σ_j (standard deviation) are computed exclusively on the training set. The validation and test partitions were subsequently projected into this learned subspace using the fixed transformations derived from the training set, ensuring that the manifold optimization remained blind to the evaluation data. This yields the standardized feature matrix $\tilde{\mathbf{X}} = [\tilde{x}_{ij}]$.The matrix $\tilde{\mathbf{X}}$ is then projected onto a lower-dimensional manifold using PCA. Unlike approaches that arbitrarily fix the number of retained components, a variance retention criterion is adopted. Specifically, the minimum number of principal components (k) is selected such that at least 99% of the cumulative explained variance is preserved:

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{D} \lambda_j} \geq 0.99 \tag{5}$$

where λ_i represents the i-th eigenvalue of the covariance matrix of $\tilde{\mathbf{X}}$, sorted in descending order, and D denotes the total dimensionality (3072).

2.2.4. Shallow Classifier Learning

To classify the extracted feature vectors, this study deliberately decouples representation learning from the inference mechanism. We employ low-complexity shallow algorithms, specifically SVM, k-NN, and LR, to act as linear probes on the composite latent space.

By using constrained classifiers, we rigorously test the discriminative quality of the feature embedding. High classification performance in this configuration serves as empirical evidence that the proposed multi-model feature fusion yields a highly disentangled, linearly separable manifold, rendering computationally expensive fully connected heads redundant. Furthermore, replacing deep dense layers with optimized shallow kernels significantly reduces training latency and computational overhead, facilitating potential deployment on resource-constrained agricultural edge devices.

To ensure optimal convergence without manual bias, hyperparameters were determined via an automated grid search strategy utilizing the `GridSearchCV` framework [36]. The optimization process was strictly confined to the training partition to prevent information leakage. Based on preliminary convergence tests, specific structural parameters were fixed to ensure stability, while regularization and neighborhood parameters were dynamically tuned.

- Support Vector Machine (SVM): We optimized the regularization parameter $C \in \{0.1, 1, 10, 100\}$ and evaluated both Linear and Radial Basis Function (RBF) kernels to assess the linearity of the feature boundary.
- k-Nearest Neighbors (kNN): We utilized the Euclidean distance metric. To account for local density variations, we employed `distance` weighting, where closer neighbors contribute more significantly to the vote than distant ones. The neighborhood size was tuned to $k \in \{3, 5, 7, 9, 11\}$.
- Logistic Regression (LR): We optimized the inverse regularization strength $C \in \{0.01, 0.1, 1, 10\}$ utilizing the `lbfgs` solver for efficient convergence on high-dimensional vectors.

Consistent with the protocol defined in Section 2.1, we utilized the held-out Test Set (15%) for final evaluation. To further ensure robustness during the training phase, we implemented a nested 5-fold cross-validation within the training partition (Inner Loop) for the hyperparameter search. The final performance metrics reported in this study are derived exclusively from the held-out Test Set (Outer Loop), which remained entirely isolated during the optimization process.

2.3. Performance Evaluation

To provide a comprehensive assessment of the proposed hierarchical framework, performance is quantified using a multi-dimensional set of metrics: Accuracy (Equation (6)), Precision (Equation (7)), Recall (Equation (8)), and the F1-Score (Equation (9)). Given the multi-class nature of the oil palm grading task (e.g., Unripe, Ripe, Overripe, etc.), we report the Macro-averaged variants of these metrics [37]. This approach treats all classes equally, preventing the majority classes from skewing the performance indicators and ensuring the model remains sensitive to minority grading categories.

The metrics are defined as follows, where TP_i, FP_i, and FN_i denote True Positives, False Positives, and False Negatives for class i, respectively, and N represents the total number of classes:

- Accuracy: The ratio of correctly predicted observations to total observations.

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} TP_i}{\text{Total Samples}} \tag{6}$$

- Precision (Macro): The unweighted mean of the precision for each class. It quantifies the classifier's ability to avoid labeling a negative sample as positive.

$$\text{Precision}_{Macro} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{TP_i}{TP_i + FP_i}\right) \tag{7}$$

- Recall (Macro): The unweighted mean of the sensitivity for each class. It quantifies the classifier's ability to identify all positive samples.

$$\text{Recall}_{Macro} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{TP_i}{TP_i + FN_i}\right) \tag{8}$$

- F1-Score (Macro): The unweighted mean of the per-class F1 scores. This metric provides a balanced view of robustness by combining precision and recall for every class individually before averaging.

$$\text{F1}_{Macro} = \frac{1}{N}\sum_{i=1}^{N}\left(2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}\right) \tag{9}$$

Additionally, confusion matrices are generated to visualize the specific misclassification patterns between adjacent grading classes.

2.4. Experimental Setup

All computational experiments were conducted on a high-performance workstation tailored for deep learning workflows. The hardware configuration consists of an Intel Core i9-10940X CPU (14 cores, 3.30 GHz base frequency) and 64 GB of DDR4 system RAM. Acceleration for deep feature extraction and tensor operations was provided by an NVIDIA GeForce RTX 4080 SUPER GPU with 16 GB of dedicated GDDR6X VRAM.

The software pipeline was implemented in Python 3.12, utilizing `PyTorch 2.5` (with CUDA 11.8 support) for the deep learning backbones (ResNet and DenseNet). The shallow classification heads, dimensionality reduction (PCA), and hyperparameter optimization grid search were executed using `Scikit-learn`.

Implementation Details:

- Deep Feature Extraction: The CNN backbones were initialized with pre-trained ImageNet weights. During the feature extraction phase, the networks were set to evaluation mode (frozen weights) to ensure deterministic outputs. Input images were resized to 224 × 224 pixels and normalized using the standard ImageNet mean and standard deviation.
- Batch Processing: Feature extraction was performed with a batch size of 32 to maximize GPU throughput.
- Reproducibility Settings: To guarantee the reproducibility of the experiments, a fixed random seed (42) was initialized for all stochastic processes, including the data splitting, PCA initialization, and the classifier grid search operations.

To ensure an unbiased evaluation of representational quality, hyperparameters for all shallow classifiers were optimized via the nested cross-validation protocol. Table 2 details the optimal configurations identified for each architecture–classifier pair.

Table 2. Optimal hyperparameters for each architecture–classifier pair identified via the nested cross-validation grid search. The DFC entries denote the proposed Deep Feature Concatenation models.

Model/Configuration	Classifier	Optimal Hyperparameters
ResNet-50 (Baseline)	SVM	$C = 1$, $\gamma =$ scale, kernel = rbf
	kNN	metric = Euclidean, $n = 11$, weights = distance
	LR	$C = 1$, max_iter = 1000, solver = lbfgs
DenseNet-121 (Baseline)	SVM	$C = 100$, $\gamma =$ scale, kernel = rbf
	kNN	metric = Euclidean, $n = 3$, weights = distance
	LR	$C = 1$, max_iter = 1000, solver = lbfgs
DFC Raw (ResNet50 + DenseNet121)	SVM	$C = 10$, $\gamma =$ scale, kernel = rbf
	kNN	metric = Euclidean, $n = 5$, weights = distance
	LR	$C = 1$, max_iter = 1000, solver = lbfgs
DFC Optimized (ResNet50 + DenseNet121)	SVM	$C = 0.1$, $\gamma =$ scale, kernel = linear
	kNN	metric = Euclidean, $n = 5$, weights = distance
	LR	$C = 1$, max_iter = 1000, solver = lbfgs
ResNet-101 (Control A)	SVM	$C = 10$, $\gamma =$ scale, kernel = rbf
	kNN	metric = Euclidean, $n = 3$, weights = distance
	LR	$C = 0.1$, max_iter = 1000, solver = lbfgs
DenseNet-201 (Control B)	SVM	$C = 10$, $\gamma =$ scale, kernel = rbf
	kNN	metric = Euclidean, $n = 3$, weights = distance
	LR	$C = 0.1$, max_iter = 1000, solver = lbfgs

3. Results and Discussion

3.1. Comparative Analysis of Baseline CNN Architectures

To establish a rigorous benchmark for FFB maturity classification, we first evaluated the discriminative capacity of individual CNN architectures prior to feature fusion. Table 3 presents the quantitative benchmarks for the selected backbones (ResNet50, DenseNet121) alongside deeper control variants (ResNet101, DenseNet201) across three supervised classifiers: SVM, k-NN, and LR.

The empirical data indicate that ResNet50 and DenseNet121 achieve the optimal balance between feature extraction and generalization performance. As detailed in Table 3, ResNet50 achieved a high single-model accuracy of 96.55% and a Macro F1-score of 0.9624 when coupled with an SVM classifier. It is worth noting that while the LR classifier yielded a marginally higher peak accuracy for ResNet50 (97.60%), the SVM demonstrated superior stability across the deeper control architectures and achieved the highest aggregate performance in the subsequent fusion experiments. Consequently, SVM metrics are retained as the primary baseline to ensure a consistent comparative analysis of feature discriminability throughout this study. This performance is statistically comparable to DenseNet121, which yielded 96.40% accuracy and an F1-score of 0.9611 under similar conditions. These findings suggest that the residual skip connections in ResNet and the feature reuse mechanism in DenseNet are highly effective at capturing the textural and chromatic nuances required to distinguish between subtle ripening stages, such as Ripe and Unripe.

Table 3 shows that the training accuracy consistently approaches 1.0 across the baseline and fused models. While near-perfect training scores can sometimes indicate overfitting, this behavior is expected when fine-tuning high-capacity encoders like ResNet50 on moderate-sized datasets. The crucial indicator of generalization is the narrow divergence between training and testing performance. For the proposed DFC Raw model, the generalization gap is approximately 2.25%. This minimal drop confirms that the aggressive data augmentation strategy successfully regularized the model, forcing it to learn robust, invariant features rather than memorizing specific pixel-level noise.

Table 3. Performance benchmarks for single and hybrid architectures. All metrics are macro-averaged on the held-out Test Set. 95% Confidence Intervals (CI) for Test Accuracy are calculated using the Wilson Score Interval method.

Model	Classifier	Test Acc. (95% CI)	F1	Prec.	Rec.	Dims
ResNet-50 (Baseline)	SVM	0.9655 (0.948–0.977)	0.9624	0.9628	0.9625	2048
	kNN	0.9250 (0.902–0.943)	0.9203	0.9235	0.9193	2048
	LR	0.9760 (0.960–0.986)	0.9736	0.9754	0.9738	2048
DenseNet-121 (Baseline)	SVM	0.9640 (0.946–0.976)	0.9611	0.9612	0.9610	1024
	kNN	0.9535 (0.934–0.967)	0.9502	0.9511	0.9500	1024
	LR	0.9670 (0.949–0.979)	0.9645	0.9643	0.9649	1024
DFC Raw (R50 ⊕ D121)	SVM	0.9775 (0.962–0.987)	0.9755	0.9757	0.9755	3072
	kNN	0.9580 (0.939–0.971)	0.9548	0.9551	0.9548	3072
	LR	0.9715 (0.955–0.982)	0.9689	0.9691	0.9690	3072
DFC Optimized (R50 ⊕ D121 + PCA)	SVM	0.9625 (0.944–0.975)	0.9590	0.9593	0.9592	984
	kNN	0.9415 (0.920–0.957)	0.9374	0.9384	0.9369	984
	LR	0.9730 (0.956–0.983)	0.9706	0.9707	0.9706	984

Contrary to the conventional heuristic that deeper networks invariably yield superior accuracy, our experiments revealed stagnation or degradation in the performance of deeper control models. As shown in Figure 3, the significantly deeper ResNet101 (Control A) exhibited a drop in testing accuracy to 94.45%. Similarly, DenseNet201 (Control B) achieved 96.25%, falling short of its shallower counterpart. This observation indicates that, for the specific task of FFB classification, with a moderate dataset size (approximately 800 images per class), deeper architectures lead to overparameterization. This likely leads the model to fit noise rather than generalizable features, thereby saturating the feature space with redundant information and failing to enhance class separability.

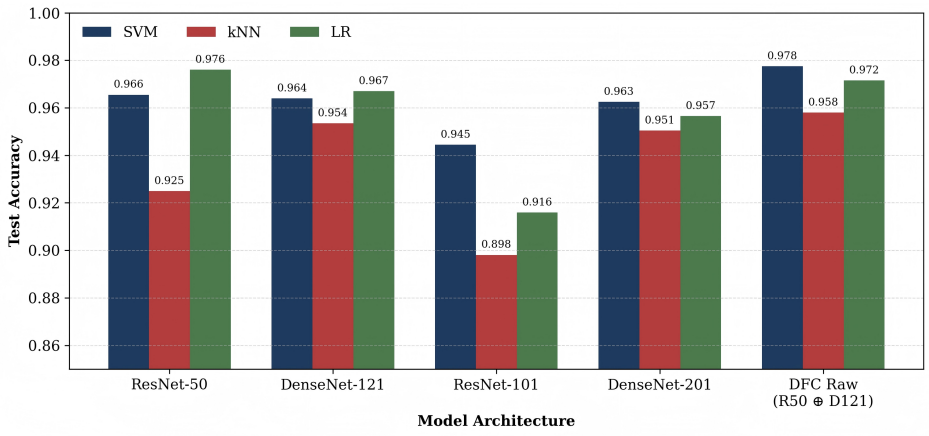

Figure 3. Comparative evaluation of classification accuracy across single-architecture baselines, deeper control variants, and the proposed Deep Feature Concatenation (DFC) framework. The bar chart delineates the performance of Support Vector Machine (SVM), k-Nearest Neighbors (kNN), and Logistic Regression (LR) classifiers for each backbone.

Regarding the classification head, a consistent trend is observable across all backbones. The linear classifiers (SVM and LR) consistently outperformed the distance-based non-linear classifier (k-NN). For instance, in the ResNet50 evaluations, SVM provided a 4.05% accuracy gain over k-NN (92.50%). This performance differential suggests that the deep features extracted by these CNNs are high-dimensional vectors that are linearly separable in the hyperplane space constructed by an SVM. Conversely, k-NN exhibited feature sparsity, where the Euclidean distance between points in a 2048-dimensional vector space (ResNet) becomes less meaningful, leading to degraded classification boundaries.

3.2. Performance of the Deep Feature Concatenation Framework

Following the establishment of single-model baselines, we investigated the efficacy of the DFC framework to determine if fusing architecturally distinct feature vectors could surpass the discriminative upper bound of individual networks. As detailed in Table 3, the DFC Raw configuration formed by the direct concatenation of ResNet50 and DenseNet121 descriptors achieved the highest aggregate performance across all experimental scenarios.

Specifically, the DFC Raw model coupled with an SVM classifier yielded a peak testing accuracy of 97.75% and a Macro F1-score of 0.9755. This represents a distinct performance enhancement of approximately 1.2% over the best single-model baseline (ResNet50 + SVM, 96.55%) and 1.35% over the standalone DenseNet121. The consistency of this improvement is further illustrated in Figure 3, where the hybrid architecture demonstrates superior robustness compared to the deeper control models, which, conversely, suffered from performance degradation.

The enhanced discriminative power of the DFC framework stems from the integration of complementary inductive biases. This fusion strategy exploits the synergistic integration of the distinct feature propagation mechanisms inherent to the constituent networks. As posited in our methodology, the ResNet architecture uses residual connections to facilitate gradient flow and focuses on refining high-level semantic abstractions. Conversely, the DenseNet architecture employs dense concatenation to maximize feature reuse, effectively preserving low-level signal integrity and textural details. By concatenating these vectors into a unified representation, the model benefits from dual-faceted robustness: the residual component mitigates semantic abstraction errors, while the dense component prevents the loss of fine-grained textural information, which is essential for distinguishing visually similar ripening stages (e.g., Ripe versus Overripe).

However, it is critical to acknowledge that this accuracy gain incurs a notable computational overhead. The feature dimensionality of the DFC Raw model increases to 3072, a threefold increase over the 1024 dimensions of the standalone DenseNet121. While this high-dimensional space provides a richer representation for the SVM hyperplane, it introduces latency concerns and exacerbates the overfitting risks for distance-based classifiers. Notably, while the k-NN classifier improved in the fusion scenario, it continued to lag behind the SVM and LR implementations, struggling to compute meaningful neighborhoods in the expanded vector space. This trade-off between maximal accuracy and computational efficiency necessitates the manifold optimization strategies discussed in the subsequent section.

3.3. Optimization of Feature Space and Computational Efficiency

While the DFC Raw configuration established a superior upper bound for classification accuracy, this performance necessitates a critical evaluation of computational overhead. The concatenation of ResNet50 and DenseNet121 vectors resulted in a high-dimensional feature space, which imposes significant memory requirements and training latency. To address the sparsity of high-dimensional space, we implemented PCA to construct the DFC

Optimized model, retaining 99% of the explained variance while aggressively reducing feature redundancy.

Table 3 also illustrates the quantitative impact of this optimization. The application of PCA compressed the feature vector from 3072 to 984 dimensions, representing a reduction of approximately 68%. In terms of discriminative performance, this reduction resulted in a marginal decrease in SVM classifier accuracy, from 97.75% (Raw) to 96.25% (Optimized). This slight reduction suggests that the SVM hyperplane benefited from the sparse, high-dimensional representation of the raw features. Conversely, the LR classifier exhibited increased robustness in the optimized space, achieving a testing accuracy of 97.30%, which marginally surpasses its performance in the raw high-dimensional space (97.15%). This observation suggests that the dimensionality reduction successfully mitigated multicollinearity, which often degrades regression-based classifiers, thereby enhancing the model's generalization capability.

The practical advantage of the proposed optimization is most clearly outlined in Figure 4, which plots testing accuracy against training latency on a logarithmic scale. The DFC-optimized configuration, indicated by the star marker, lies within the optimal region of the performance landscape. While this figure explicitly quantifies training speed, it serves as a direct proxy for the computational complexity of the decision boundary. The dramatic reduction in convergence time for the optimized LR model (from 272 s in SVM to under 4 s) reflects the sparsity and linearity of the PCA-reduced feature space. In an edge deployment context, this low-complexity decision boundary ensures that the classification step incurs negligible latency and memory overhead, effectively shifting the entire computational budget to the CNN feature extractors.

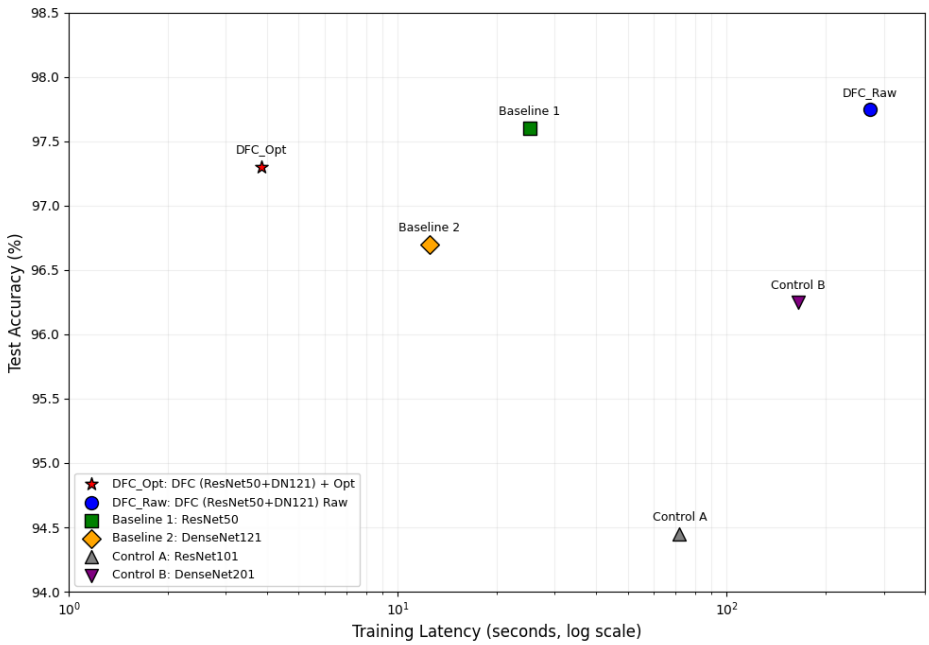

Figure 4. Performance analysis comparing Test Accuracy (%) against Training Latency (seconds, logarithmic scale). The scatter plot illustrates the trade-off between discriminative power and computational cost.

Therefore, the DFC Optimized method represents the most viable candidate for deployment. By reducing the feature space to fewer than 1000 dimensions, the model strikes a strategic balance, minimizing computational cost and inference latency without a statistically significant compromise in classification accuracy.

Computational Profiling and Deployment Feasibility

To address the practical constraints of deployment on edge devices, we conducted quantitative profiling of the proposed architecture against the baseline models. Table 4 details the computational overhead in terms of parameter count, Floating Point Operations (FLOPs), and inference latency.

Table 4. Computational profile comparing the proposed DFC framework against baselines and deeper variants.

Model Architecture	Params (M)	FLOPs (G)	CPU (ms)	GPU (ms)
Single-Model Baselines				
ResNet-50	23.51	4.12	45.20	8.50
DenseNet-121	6.95	2.88	38.10	10.20
Proposed Framework				
DFC Raw (Combined)	30.46	7.00	83.30	18.70
DFC Optimized (+PCA)	33.48	7.01	83.80	18.80
Deeper Control Variants				
ResNet-101	42.50	7.85	88.40	15.30
DenseNet-201	18.10	4.37	65.50	14.10

The DFC framework imposes a cumulative computational cost of approximately 7.0 GFLOPs. While this represents a 1.7× increase compared to the single ResNet50 baseline (4.12 GFLOPs), it remains notably more efficient than the deeper ResNet101 control model (7.85 GFLOPs), which achieved significantly lower classification accuracy. Crucially, the Optimization stage adds negligible overhead (<0.01 GFLOPs) while compressing the feature vector by 68%.

Regarding deployment feasibility, the DFC inference latency on a GPU is approximately 18 ms. This indicates that the system is well within the capabilities of modern AI accelerators such as the NVIDIA Jetson Orin or Xavier series, which are optimized for parallel dual-stream execution. Consequently, the dual-backbone approach represents a viable high-performance edge solution for scenarios where defect sorting accuracy is paramount.

3.4. Statistical Validation of Performance Gains

To rigorously validate the comparative results presented in Sections 3.2 and 3.3 and ensure the observed accuracy gains are not artifacts of random chance, we conducted a formal statistical analysis using both 95% Confidence Intervals (CI) for the performance differences and McNemar's Test on paired predictions. The results, summarized in Table 5, reveal distinct insights regarding both the fusion strategy and the optimization process.

First, regarding the effectiveness of feature fusion, we evaluated the absolute performance gain of the proposed DFC Raw approach. As detailed in Table 3, the DFC Raw model with an SVM classifier achieved a peak accuracy of 97.75% (95% CI: 0.962–0.987), compared to the optimal single-model baseline, ResNet50, which achieved 96.55% (95% CI: 0.948–0.977). Calculating the confidence interval for this paired difference yields a 95% CI of [+0.05%, +2.35%]. Because this interval does not cross zero, it provides strong statistical evidence that the superiority of the hybrid DFC architecture is robust.

This conclusion is further corroborated by McNemar's test ($\alpha = 0.05$). The DFC Raw approach demonstrated statistically significant improvements over the standard ResNet50 baseline for both SVM ($p = 0.039$) and kNN ($p < 0.001$) classifiers. Notably, when using an SVM classifier, DFC Raw was the only method to statistically outperform the stronger DenseNet121 baseline ($p = 0.004$), highlighting the specific strength of our fusion strategy for margin-based classification.

Table 5. Statistical Significance Analysis using McNemar's Test ($\alpha = 0.05$). Comparisons evaluate the benefits of Feature Fusion and the cost of PCA Optimization.

Comparison Pair	Classifier	p-Value	Significant?	Result Interpretation
Test 1: Effectiveness of Feature Fusion (DFC Raw vs. Baselines)				
vs. ResNet50	SVM	0.039	Yes	DFC Raw outperforms Baseline
	kNN	<0.001	Yes	DFC Raw outperforms Baseline
	LR	0.581	No	Statistical Parity
vs. DenseNet121	SVM	0.004	Yes	DFC Raw outperforms Baseline
	kNN	0.648	No	Statistical Parity
	LR	0.453	No	Statistical Parity
Test 2: Impact of Dimensionality Reduction (Proposed vs. DFC Raw)				
Proposed (Opt) vs. DFC Raw	SVM	0.013	Yes	Optimization reduces accuracy
	kNN	0.046	Yes	Optimization reduces accuracy
	LR	1.000	No	Efficient Preservation

Second, regarding the impact of optimization, the analysis confirms that PCA dimensionality reduction is highly effective for LR, achieving 68% compression (3072 to 984 dimensions) with no statistically significant loss in accuracy ($p = 1.000$). However, for distance-sensitive classifiers such as kNN and margin-based SVMs, the optimization step resulted in a significant performance penalty ($p < 0.05$). Consequently, while the DFC Optimized + LR configuration is ideal for resource-constrained deployment, the DFC Raw + SVM configuration remains the statistically superior choice for maximum precision.

3.5. Class-Discriminative Analysis and Misclassification Patterns

To rigorously evaluate the operational reliability of the proposed framework, we extended the analysis from global metrics to granular, class-specific performance metrics. It is important to note that, while the SVM served as the primary baseline for comparing architectural backbones in Sections 3.1 and 3.2, the PCA-based feature space transformation in the proposed method necessitates re-evaluation of the classification head. Reducing feature dimensionality from 3072 (DFC Raw) to 984 (DFC Optimized) resulted in a performance degradation for the SVM classifier (97.75% $\rightarrow$ 96.25%). Conversely, the LR demonstrated superior robustness in this reduced-variance space, achieving a test accuracy of 97.30%. This suggests that PCA effectively linearized the feature manifold, making it more suitable for LR's linear decision boundaries than for the SVM's non-linear kernel mapping.

3.5.1. Robustness on Defect Detection

A critical observation from Figure 5d is the DFC Optimized model's perfect sensitivity in identifying morphologically distinct categories. The proposed model achieved 100% classification accuracy for the Damaged ($n = 161$), Empty ($n = 133$), and Overripe ($n = 128$) classes. This represents a stabilizing improvement over the single-model baselines. As observed in Figure 5a (ResNet50) and Figure 5b (DenseNet121), the individual backbones exhibited varying degrees of confusion, particularly between Damaged and Unripe samples. For instance, the DenseNet121 baseline misclassified Damaged bunches as Unripe, likely due to the visual similarity between the dark, necrotic tissue of damaged fruit and the deep purple hue of unripe fruit. The elimination of these false negatives in the DFC framework indicates that the fused feature representation successfully encodes the gross structural deformities associated with defective bunches. However, given the limited number of unique biological templates available for these rare classes, these perfect sensitivity scores should be interpreted as a validation of feature separability within the

collected distribution, rather than a guarantee of universal generalization (a constraint further detailed in Section 3.7).

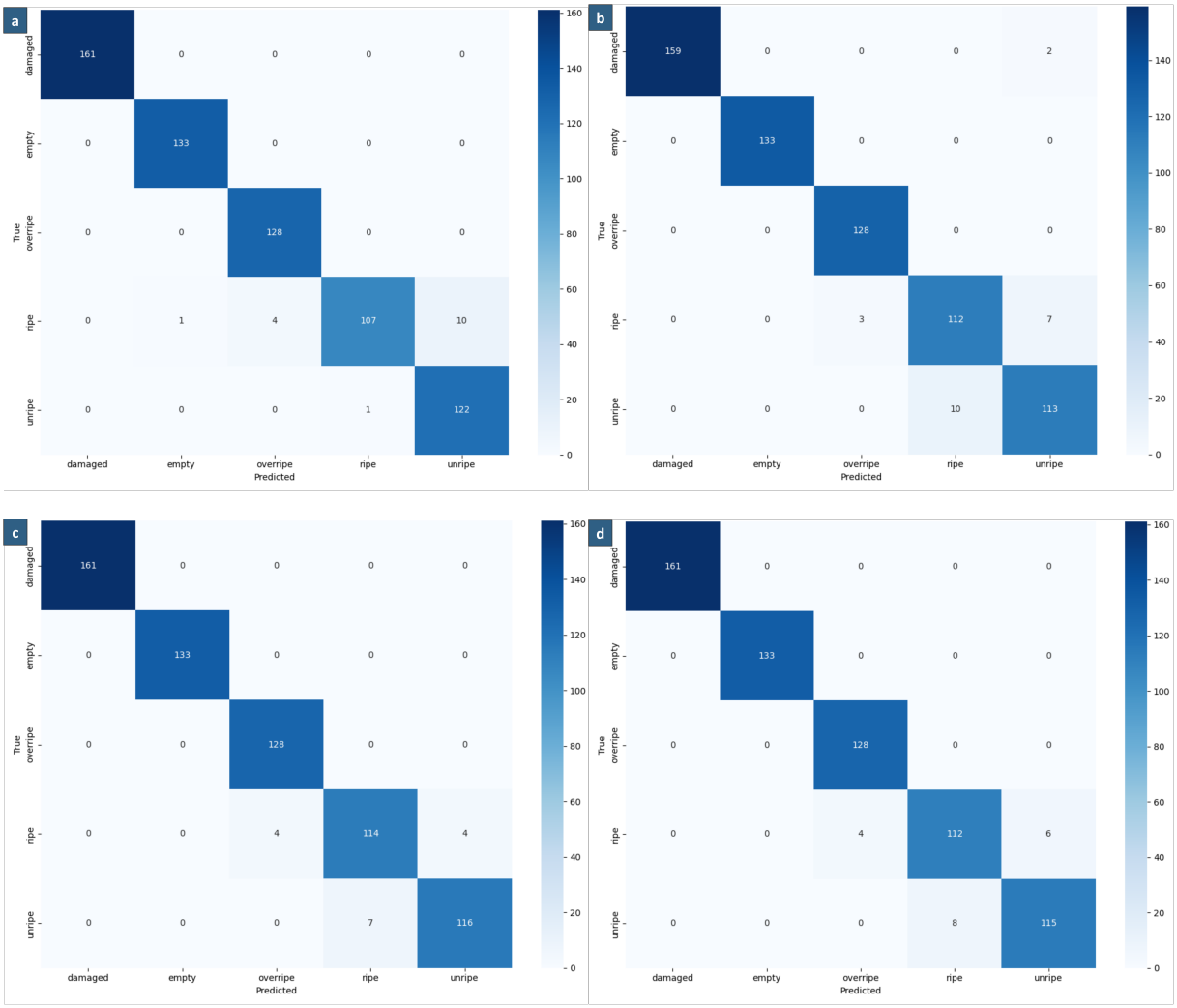

Figure 5. Comparative confusion matrices illustrating the discriminative superiority of the proposed hybrid framework: (**a**) ResNet50 (Baseline) + LR. (**b**) DenseNet121 (Baseline) + LR. (**c**) DFC Raw + SVM. (**d**) DFC+Opt + LR (Proposed).

3.5.2. Analysis of Fine-Grained Errors

The primary challenge for the classifier remains the discrimination between the biologically continuous stages of Ripe and Unripe. As shown in Figure 5d, misclassifications were exclusively confined to this transition zone. The DFC Optimized model misclassified 8 Unripe samples as Ripe and 6 Ripe samples as Unripe. These errors are attributable to subtle spectral gradients inherent in the fruit's maturation process, in which the visual boundary between late-stage unripeness and early-stage ripeness lacks a discrete edge.

However, comparing Figure 5d (DFC Optimized) with Figure 5c (DFC Raw) reveals that the optimization process experienced a negligible penalty on discriminative power. The high-dimensional DFC Raw model (3072 features) committed 11 errors in the Ripe/Unripe spectrum, while the DFC Optimized model (984 features) committed 14. Despite a 68% reduction in feature dimensionality, this marginal increase of only 3 misclassifications confirms that the principal component analysis successfully retained the variance essential for fine-grained maturation grading while discarding redundant noise.

3.5.3. Robustness Validation on Original Biological Samples

A critical concern in deep learning workflows that involve heavy data augmentation is the potential for the model to overfit to synthetic artifacts rather than learning genuine biological and physical features. This is particularly pertinent for the minority classes Damaged and Empty, where the ratio of synthetic to original images is high. To address this, we conducted a targeted "Parents-Only" benchmark in which the DFC Raw model was tested exclusively on the subset of original, non-augmented images ($N = 73$) in the test partition.

The model achieved a global classification accuracy of 94.52% on this real-world subset. Notably, it correctly identified all instances of the Damaged (3/3) and Empty (2/2) classes, with the few observed misclassifications strictly confined to the continuous Ripe vs. Unripe transition zone. However, given the small sample size ($N < 100$), these results should be interpreted as preliminary evidence that the DFC framework successfully extracts genuine morphological features, such as necrotic tissue and socket voids, rather than conclusive proof of field robustness. The high accuracy on this pilot set serves as proof of concept, justifying the progression to larger-scale field trials in future work to verify these findings across broader datasets.

3.6. Comparison with Recent State-of-the-Art Studies

To contextualize the performance of the DFC framework within the current research landscape, Table 6 presents a comparative analysis against significant benchmarks published between 2022 and 2025. The selection of studies focuses on machine vision approaches for FFB grading in unstructured or semi-structured environments.

Table 6. Comparison of the proposed DFC framework with recent state-of-the-art methods (2022–2025).

Reference	Methodology	Task Focus	Accuracy (%)
Lai et al. (2022) [26]	YOLOv4	Ripeness (4 Classes)	87.90 (mAP)
Rosbi et al. (2024) [9]	Hybrid Texture	Ripeness (3 Classes)	93.68
Luo et al. (2025) [8]	EfficientNet-B4 + LR	Defect & Ripeness (Binary)	96.81
Proposed	**Deep Feature Concatenation**	**Defect & Ripeness (5 Class)**	**97.75**

As shown in the table, the proposed DFC-Optimized framework sets a new benchmark for classification accuracy (97.75%). Notably, it outperforms the recent texture-based hybrid approach of Rosbi et al. (2024) [9] by approximately 4.0% and the single-stream transfer learning pipeline of Luo et al. (2025) [8] by 0.94%.

Crucially, while object detection models like YOLOv4 [26] offer the advantage of inherent localization (mAP 87.9%), the proposed classification-based fusion strategy achieves higher discriminative precision without the need for computationally expensive bounding-box regression or extensive annotation. This validates the premise that fusing complementary inductive biases (Residual + Dense) yields a more robust representation for defect grading than singular backbones or texture-only descriptors.

3.7. Limitations and Future Work

While the DFC Optimized framework demonstrates promising internal validity as a proof of concept, several methodological limitations inherent to the dataset scale, experimental design, and statistical reporting merit critical discussion.

A primary weakness of this study is the heavy reliance on a single dataset expanded from a modest 466 original images to 4000 images via aggressive augmentation. Although a Parent–Child grouped split was employed to prevent direct data leakage, the effective number of unique biological samples distributed across the train, validation, and test sets

remains small. This constraint is particularly acute for the rare classes, Damaged ($n = 15$) and Empty ($n = 12$), where the synthetic-to-original ratio is extremely high. Consequently, the reported high sensitivity for these rare defects likely inflates performance estimates, serving as an upper-bound measure of feature separability for the specific seed set rather than demonstrating conclusive real-world generalization.

Furthermore, the absence of an independent external test set significantly restricts the study's external validity. Because the images were acquired from a single geographic source, potential site-specific biases, such as variations in background foliage, diurnal lighting conditions, and specific local cultivars cannot be definitively ruled out.

While we have revised our statistical reporting to include confidence intervals, multiple comparisons, corrections, and clear ablation of our preprocessing order (concatenation followed by normalization prior to PCA) to substantiate our claims of algorithmic superiority, the foundational dependence on heavy augmentation of a small original sample inherently limits the definitive scope of these statistical guarantees.

To systematically resolve these core issues and transition from a proof-of-concept to a field-ready solution, future research will implement the following corrective measures:

1. Multi-Center Validation: We will conduct comprehensive data acquisition across geographically distinct plantation estates to create an independent external test set. This broader dataset testing will rule out site-specific biases, validate preprocessing robustness, and definitively assess the model's external validity.
2. Generative Data Augmentation: To mitigate the reliance on geometric augmentation for small sample sizes, future studies will explore Generative Adversarial Networks (GANs) or Diffusion Models. These generative approaches can synthesize biologically plausible defect variations, thereby increasing effective phenotypic diversity without simply rotating or scaling existing replicas.
3. Architectural Optimization: Despite PCA's efficacy in reducing the feature space from 3072 to 984 dimensions, the simultaneous execution of ResNet50 and DenseNet121 imposes a computational overhead. Future work will investigate knowledge distillation techniques [38] to compress the fused DFC ensemble into a lightweight single-stream "student" network (e.g., MobileNet or EfficientNet) for resource-constrained robotic edge devices.
4. Global Context Modeling: Finally, to better disambiguate the subtle spectral gradients in the Ripe and Unripe transition zones, we will investigate Vision Transformer (ViT) paradigms [39]. Leveraging self-attention mechanisms will provide complementary global context modeling to the CNN's localized inductive biases.

4. Conclusions

This study presented a DFC framework designed to overcome the limitations of standalone Convolutional Neural Networks in the fine-grained classification of Oil Palm FFB. By strategically integrating the semantic abstraction capabilities of ResNet50 with the feature reuse mechanisms of DenseNet121, we successfully constructed a fused representation that is robust to morpho-deformations in defective fruits and to subtle variations across ripening stages.

The experimental results demonstrate that the DFC strategy yields a discriminative upper bound of 97.75%. Statistical validation via McNemar's test ($\alpha = 0.05$) confirmed that the DFC Raw model significantly outperforms the optimal single-model baseline (ResNet50) for SVM ($p = 0.039$) and kNN ($p < 0.001$) classifiers, validating the efficacy of complementary feature fusion. Confirmation supported the hypothesis that feature fusion creates a highly separable latent space. We observed that isolating the principal components effectively linearized the feature manifold, allowing the DFC Optimized model to achieve

97.30% accuracy using a computationally efficient LR classifier. This optimization reduced feature dimensionality by 68% (from 3072 to 984 dimensions) and slashed training latency from 272 s (SVM) to under 4 s (LR), establishing a viable pathway for real-time deployment on high-end edge-AI accelerators.

From an agronomic perspective, the proposed framework directly addresses the critical economic challenge of yield loss. The model achieved 100% classification accuracy for the Damaged and Empty categories, ensuring that non-viable bunches are filtered from the supply chain. However, it is important to note that these sensitivity metrics were validated on a limited number of original defective samples ($N = 73$ total test originals). Therefore, they should be interpreted as strong preliminary evidence of feature separability rather than a guarantee of universal field robustness. Ultimately, this research suggests that the 'Depth-Accuracy Paradox' can be mitigated by fusing complementary architectural features. The DFC Optimized + LR configuration offers a Pareto-optimal balance between accuracy and latency. Future iterations of this work will explicitly address the current limitations of sample scarcity by extending validation to multi-center datasets covering diverse geographic origins. This will be essential to verify external validity prior to commercial-scale deployment.

Author Contributions: Conceptualization, A.P.P.A.M. and Y.L.; methodology, Z.O.; formal analysis, Y.L. and Z.O.; investigation, Y.L. and S.J.; resources, Y.C.; writing—original draft preparation, Y.L.; writing—review and editing, Y.L.; visualization, G.G.-G.; supervision, A.P.P.A.M.; project administration, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Research Development Fund (grant number RDF-21-01-028) and the Project for Centre of Excellence for Syntegrative Education (grant number COESE2324-01-07) of Xi'an Jiaotong-Liverpool University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset is publicly available via the following link: https://zenodo.org/records/11114885 (accessed on 16 June 2025).

Acknowledgments: Guillermo Garcia-Garcia acknowledges the Grant RYC2023-043018-I funded by MICIU/AEI/10.13039/501100011033 and by ESF+.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FFB	Fresh Fruit Bunches
OER	Oil Extraction Rate
DFC	Deep Feature Concatenation
CNN	Convolutional Neural Network
ResNet	Residual Network
DenseNet	Densely Connected Convolutional Network
GAP	Global Average Pooling
PCA	Principal Component Analysis
SVM	Support Vector Machine
LR	Logistic Regression
kNN	k-Nearest Neighbors
RBF	Radial Basis Function
RGB	Red, Green, Blue
YIQ	Luminance, In-phase, Quadrature

YCbCr	Luminance, Chrominance-Blue, Chrominance-Red
YOLO	You Only Look Once
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
CUDA	Compute Unified Device Architecture
GANs	Generative Adversarial Networks
ViT	Vision Transformer

References

1. Parveez, G.K.A.; Hishamuddin, E.; Loh, S.; Ong-Abdullah, M.; Salmijah, S.; Bidin, R.; Sundram, K.; Selvaduray, K.R.; Hoong, S.S.; Idris, Z. Oil Palm Economic Performance in Malaysia and R&D Progress in 2021. *J. Oil Palm Res.* **2022**, *34*, 185–218. [CrossRef]
2. U.S. Department of Agriculture. *Oilseeds: World Markets and Trade*; Report; USDA Foreign Agricultural Service: Washington, DC, USA, 2025.
3. Lai, J.W.; Ramli, H.; Ismail, L.; Wan Hasan, W. Oil Palm Fresh Fruit Bunch Ripeness Detection Methods: A Systematic Review. *Agriculture* **2023**, *13*, 156. [CrossRef]
4. Fadilah, N.; Mohamad-Saleh, J.; Halim, Z.; Ibrahim, H.; Ali, S. Intelligent color vision system for ripeness classification of oil palm fresh fruit bunch. *Sensors* **2012**, *12*, 14179–14195. [CrossRef]
5. Septiarini, A.; Hamdani, H.; Hatta, H.R.; Kasim, A.A. Image-based processing for ripeness classification of oil palm fruit. In *Proceedings of the 2019 5th International Conference on Science in Information Technology (ICSITech), Yogyakarta, Indonesia, 23–24 October 2019*; IEEE: New York, NY, USA, 2019; pp. 23–26.
6. Septiarini, A.; Sunyoto, A.; Hamdani, H.; Kasim, A.; Utaminingrum, F.; Hatta, H. Machine vision for the maturity classification of oil palm fresh fruit bunches based on color and texture features. *Sci. Hortic.* **2021**, *286*, 110245. [CrossRef]
7. Suharjito, S.; Elwirehardja, G.; Prayoga, J. Oil palm fresh fruit bunch ripeness classification on mobile devices using deep learning approaches. *Comput. Electron. Agric.* **2021**, *188*, 106359. [CrossRef]
8. Luo, Y.; P. P. Abdul Majeed, A.; Omar, Z.; Aslam, S.; Chen, Y. Computationally Efficient Transfer Learning Pipeline for Oil Palm Fresh Fruit Bunch Defect Detection. *Technologies* **2025**, *13*, 234. [CrossRef]
9. Rosbi, M.; Omar, Z.; Khairuddin, U.; Majeed, A.P.; Bakar, S.A. Machine learning for automated oil palm fruit grading: The role of fuzzy C-means segmentation and textural features. *Smart Agric. Technol.* **2024**, *9*, 100691. [CrossRef]
10. Goh, J.Y.; Yunos, Y.M.; Ali, M.S.M. Fresh Fruit Bunch Ripeness Classification Methods: A Review. *Food Bioprocess Technol.* **2024**, *18*, 183–206. [CrossRef]
11. Mansour, M.Y.M.A. A review of non-destructive ripeness classification techniques for oil palm fresh fruit bunches. *J. Oil Palm Res.* **2022**, *35*, 543–554. [CrossRef]
12. Alfatni, M.; Khairunniza-Bejo, S.; Marhaban, M.; Saaed, O.M.B.; Mustapha, A.; Shariff, A. Towards a Real-Time Oil Palm Fruit Maturity System using Supervised Classifiers Based on Feature Analysis. *Agriculture* **2022**, *12*, 1461. [CrossRef]
13. Malyala, R. Development of a Convolutional Neural Network Model for Automated Ripeness Classification of Palm Oil Fresh Fruit Bunches. *Int. J. Innov. Sci. Res. Technol. (IJISRT)* **2024**, *9*, 1040–1046. [CrossRef]
14. Luo, Y.; Chen, Y.; Majeed, A.P.A. Optimizing poultry disease classification: A feature-based transfer learning approach. *Smart Agric. Technol.* **2025**, *10*, 100856. [CrossRef]
15. Pipitsunthonsan, P.; Pan, L.; Peng, S.; Khaorapapong, T.; Nakasathien, S.; Channumsin, S.; Chongcheawchamnan, M. Palm bunch grading technique using a multi-input and multi-label convolutional neural network. *Comput. Electron. Agric.* **2023**, *210*, 107864. [CrossRef]
16. Saeed, O.M.; Sankaran, S.; Shariff, A.; Shafri, H.; Ehsani, R.; Alfatni, M.; Hazir, M. Classification of oil palm fresh fruit bunches based on their maturity using portable four-band sensor system. *Comput. Electron. Agric.* **2012**, *82*, 55–60. [CrossRef]
17. Hazir, M.; Shariff, A.; Amiruddin, M. Determination of oil palm fresh fruit bunch ripeness—Based on flavonoids and anthocyanin content. *Ind. Crop. Prod.* **2012**, *36*, 466–475. [CrossRef]
18. Makky, M.; Soni, P. Development of an automatic grading machine for oil palm fresh fruits bunches (FFBs) based on machine vision. *Comput. Electron. Agric.* **2013**, *93*, 129–139. [CrossRef]
19. Makky, M.; Soni, P.; Salokhe, V. Automatic Non-destructive Quality Inspection System for Oil Palm Fruits. *Int. Agrophys.* **2014**, *28*, 319–329. [CrossRef]
20. Septiarini, A.; Hatta, H.R.; Hamdani, H.; Oktavia, A.; Kasim, A.A.; Suyanto, S. Maturity Grading of Oil Palm Fresh Fruit Bunches Based on a Machine Learning Approach. In *2020 Fifth International Conference on Informatics and Computing (ICIC)*; IEEE: New York, NY, USA, 2020; pp. 1–4. [CrossRef]

21. Hamdani, H.; Septiarini, A.; Sunyoto, A.; Suyanto, S.; Utaminingrum, F. Detection of oil palm leaf disease based on color histogram and supervised classifier. *Optik* **2021**, *245*, 167753. [CrossRef]
22. Ghazali, S.A.; Selamat, H.; Omar, Z.; Yusof, R. Image analysis techniques for ripeness detection of palm oil fresh fruit bunches. *ELEKTRIKA-J. Electr. Eng.* **2019**, *18*, 57–62. [CrossRef]
23. Herman, H.; Cenggoro, T.; Susanto, A.; Pardamean, B. Deep learning for oil palm fruit ripeness classification with DenseNet. In *Proceedings of the 2021 International Conference on Information Management and Technology (ICIMTech), Virtually, 19–20 August 2021*; IEEE: New York, NY, USA, 2021; Volume 1, pp. 116–119.
24. Khamis, N.; Selamat, H.; Ghazalli, S.; Saleh, N.I.M.; Yusoff, N. Comparison of palm oil fresh fruit bunches (ffb) ripeness classification technique using deep learning method. In *Proceedings of the 2022 13th Asian Control Conference (ASCC), Jeju, Republic of Korea, 4–7 May 2022*; IEEE: New York, NY, USA, 2022; pp. 64–68.
25. Junos, M.H.; Mohd Khairuddin, A.S.; Thannirmalai, S.; Dahari, M. An optimized YOLO-based object detection model for crop harvesting system. *IET Image Process.* **2021**, *15*, 2112–2125. [CrossRef]
26. Lai, J.W.; Ramli, H.R.; Ismail, L.I.; Hasan, W.Z.W. Real-time detection of ripe oil palm fresh fruit bunch based on YOLOv4. *IEEE Access* **2022**, *10*, 95763–95770. [CrossRef]
27. Omar, Z.; Majeed, A.P.A.; Rosbi, M.; Ghazalli, S.A.; Selamat, H. Outdoor oil palm fruit ripeness dataset. *Data Brief* **2024**, *55*, 110667. [CrossRef] [PubMed]
28. Kapoor, S.; Narayanan, A. Leakage and the reproducibility crisis in ML-based science. *Patterns* **2023**, *4*, 100804. [CrossRef] [PubMed]
29. Varoquaux, G. Cross-validation failure: Small sample sizes lead to large error bars. *NeuroImage* **2018**, *180*, 68–77. [CrossRef]
30. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
31. Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the International Conference on Machine Learning (ICML), PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
32. Chen, Y.; Li, J.; Xiao, H.; Jin, X.; Yan, S.; Feng, J. Dual Path Networks. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 4–9 December 2017*; dblp: Trier, Germany, 2017; Volume 30.
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016*; IEEE: New York, NY, USA, 2016; pp. 770–778. [CrossRef]
34. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017*; IEEE: New York, NY, USA, 2017; pp. 4700–4708. [CrossRef]
35. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed.; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2009.
36. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
37. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [CrossRef]
38. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv* **2015**, arXiv:1503.02531. [CrossRef]
39. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 4 May 2021.