Ingénierie des Systèmes d'Information
Vol. 31, No. 10, October, 2026, pp. 3307-3319
Journal homepage: http://iieta.org/journals/isi

A Probability-Level Fusion Framework Combining a Custom Convolutional Neural Network and MobileNetV2 for Fine-Grained Classification of Indian Toor Dal Varieties

Harini Shadaksharappa[1,2*], Saritha Chakrasali[3], Krishnamurthy Gorappa Ningappa[1]

[1] Department of Computer Science and Engineering, B.N.M. Institute of Technology, Bengaluru, Visvesvaraya Technological University, Belagavi 590018, India
[2] Department of Information Science and Engineering, B.M.S. College of Engineering, Bangalore 560019, India
[3] Department of Information Science and Engineering, B.N.M. Institute of Technology, Bengaluru, Visvesvaraya Technological University, Belagavi 590018, India

Corresponding Author Email: hharinis8@gmail.com

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/isi.311006

Received: 2 May 2026
Revised: 13 September 2026
Accepted: 30 September 2026
Available online: 31 October 2026

Keywords:
toor dal classification, probability-level fusion, convolutional neural network, MobileNetV2, fine-grained image classification, pulse variety identification

ABSTRACT

Accurate identification of morphologically similar pulse varieties remains challenging because differences in grain color, texture, and shape are often subtle. Manual inspection is time-consuming and subject to individual judgment, limiting its suitability for large-scale grain quality assessment. This study develops a probability-level fusion framework that combines a custom convolutional neural network (CNN) with MobileNetV2 for the classification of three Indian toor dal varieties: Shivalinga, Ganesha, and Padra. A dataset comprising 1,500 images, with 500 images per variety, was collected under controlled illumination and a uniform background. Image preprocessing included hue, saturation, and value (HSV) color-space conversion, threshold-based segmentation, morphological operations, and data augmentation. The class probabilities generated independently by the two networks were combined through arithmetic averaging to obtain the final prediction. Performance was evaluated using an 80:20 training–validation split and five runs with different random seeds. Across these runs, the fusion model achieved a mean training accuracy of 99.23% ± 0.13% and a mean validation accuracy of 95.99% ± 0.35%. The corresponding mean precision, recall, and F1-score were 95.85%, 95.99%, and 95.92%, respectively. An ablation study further showed that removing segmentation, morphological operations, or data augmentation reduced validation accuracy, with the largest decline observed when augmentation was omitted. These findings demonstrate the potential of probability-level fusion for distinguishing visually similar toor dal varieties under controlled imaging conditions. Further evaluation using samples from different geographical origins and field environments is required to establish the model's generalizability and suitability for practical deployment.

1. INTRODUCTION

Food grains are one of the major sources of nutrition in the world and play an important role in food security and agricultural sustainability. The pulses are a high protein, rich in essential amino acids, dietary fibre, vitamins and minerals, among all other food grains. Pigeon pea (*Cajanus cajan*), also known as toor dal, is one of the most popular pulses in India, and it is an essential ingredient in the Indian diet because of its nutritional benefits and price accessibility. Hence, the quality, purity, and authenticity of toor dal have attained more significance among farmers, traders, food processing industries, and consumers.

An expanding food industry and global trade in agricultural products have led to growing needs for dependable grain quality assessment systems. Most of the quality inspection techniques are conventional techniques that use manual visual inspection by trained personnel. These approaches are time-consuming, labor-intensive, subjective, and can experience human error due to fatigue, skill levels, and environmental differences. In addition, manual examination cannot be used for large-scale industrial applications because thousands of grain samples need to be examined correctly within a short time. Thus, the use of automated grain inspection systems has emerged as a potential solution to enhance food quality evaluation efficiency, uniformity, and reliability.

The technological progress of visual computing and image processing, combined with the advent of deep learning, has made automated agricultural inspection systems a major milestone in recent years. There has been an impressive achievement in the context of deep learning models, especially convolutional neural networks (CNNs) in the context of image feature extraction, which have been able to extract hierarchical features from images without the need for handcrafted feature engineering. Transfer learning models like MobileNetV2, ResNet, DenseNet, and EfficientNet have also boosted classification accuracy without compromising computational efficiency, enabling them to be used in intelligent agriculture

applications. These advances have allowed for the automated and accurate identification of cereals, pulses, fruits, vegetables, and plant diseases, with minimal human effort.

However, it is still a difficult task to accurately identify closely related varieties of toor dal. The identification of visually different grain species is not applicable to toor dal, as different varieties of toor dal may have very similar physical properties of size, shape, surface, and color. Even experienced inspectors may find it hard to tell the difference between slight differences in the varieties. Also, intra-class variability seems to be increased by differences in illumination, camera angle, grain orientation, and background light and noise conditions, whereas inter-class differences are minimized. The above discussions suggest that traditional machine-learning-based and manual techniques are not effective for reliable classification.

Despite extensive research on deep learning approaches for grain classification, the majority of the previously published methods are based on the classification of different grain species like rice, wheat, maize, lentil, or damaged grain. Few studies have dealt with the fine-grained classification of morphologically similar pulse varieties, especially Indian toor dal. Moreover, most of the current approaches are based on a single deep learning architecture without exploring alternative representations of features using ensemble learning. Furthermore, the design of lightweight intelligent inspection systems that can realize high classification accuracy, computational efficiency, and adaptability for real-time application in agricultural environments has received very little research attention.

1.1 Research gap

The available literature shows that there is no intelligent deep learning framework that is completely dedicated to discrimination of morphologically similar Indian toor dal varieties. The majority of studies reported are on individual grain types (e.g., corn, wheat, rice), and not on more closely related varieties of the same grain type. Furthermore, existing methods do not integrate feature representations and lightweight transfer learning models to enhance classification robustness. The current methods are further restricted when applied to practical automated food grain inspection systems, as there is no comprehensive preprocessing strategy, probability-level ensemble learning, or extensive evaluation.

To overcome these limitations, the proposed framework in this study is an intelligent deep learning-based framework for classification consisting of a customized CNN and a lightweight MobileNetV2 architecture with probability-level ensemble learning. The proposed method consists of efficient preprocessing, feature extraction, and classification to enable good discrimination of morphologically similar morphs of toor dal and to preserve computational efficiency for practical implementation.

1.2 Research contributions

(1) To build a strong image preprocessing system for representing the features of the grain and to eliminate the interference of the background, image resizing, hue, saturation, and value (HSV) color space transformation, segmentation, morphological operations, and data augmentation are carried out.

(2) A tailored CNN architecture is developed to capture the fine-grained morphological elements like texture, color, and shape, which are crucial to discriminate varieties of toor dal and dal that are morphologically alike.

(3) To minimize the cost of computation required to perform deep feature extraction and the complexity of the model, a lightweight MobileNetV2 transfer learning model is implemented.

(4) To enhance the robustness and generalization ability of classification, an ensemble learning strategy based on probability-level ensemble is proposed as a method to combine the predictions of CNN and MobileNetV2 models tailored for the specific use case.

(5) The effectiveness of the proposed framework is assessed through a comprehensive experimental evaluation using multiple performance metrics such as accuracy, precision, recall, F1-score, confusion matrices, and training-validation analysis.

(6) The intelligent grain classification system proposed in this paper is a scalable and practical automated food quality inspection solution, which can be applied to smart agriculture, the food processing industry, warehouse management, supply chain quality assurance, and other fields.

2. RELATED WORK

For many Indians, toor dal—also known as pigeon pea or red gram—is an essential diet. It only grows in specific areas and months. It has been cultivated by farmers for about 3,500 years. It originated in the tropical deciduous forests of southern India. It then made its way to West Africa, East Africa, and eventually Europe. In India, it is grown on about 3.90 million hectares, producing over 3.17 million tons each year [1]. Many recent studies have used deep learning methods in grain variety classification, obtaining impressive results. A study has tested a number of deep learning models such as ResNet, VGG, EfficientNet, and MobileNet using 75,000 images of rice into five classes and showed that the most accurate model based on EfficientNet reached an accuracy of 99.67% [2]. Cultivated pigeon peas undergo several milling and polishing processes before they become available in the commercial market. The classification of morphologically similar food grains remains a challenging task in agricultural automation and food quality assessment. Grain varieties such as rice, wheat, lentils, and millet often exhibit subtle variations in shape, texture, size, and color, making manual identification subjective, time-consuming, and prone to inconsistency. To address these challenges, machine learning (ML) and computer vision techniques have emerged as efficient, objective, and scalable approaches for the automated classification of visually similar grain varieties [3].

Accurate identification of grain varieties is critical to prevent economic loss and adulteration in the agriculture sector, where traditional methods remain inefficient and error-prone. This study [4] proposes a deep convolutional neural network (DCNN)-based model, RiceNet, along with pre-trained architectures, demonstrating superior classification performance over traditional methods for ML for rice grain identification. GrainSet is a large-scale machine vision dataset with 350K+ annotated grain kernel images across multiple cereal types and regions. This study [5] enables deep learning-based benchmarking and supports automated grain quality inspection in smart agriculture. This research [6] proposes a framework based on deep learning for automated pollen

classification using transfer learning and a custom CNN modeled on a collection of 10,000 images across 40 species. Results from experiments show that pre-trained architectures perform better compared to custom CNN, reaching high accuracy and efficiency, thereby enabling scalable and reliable pollen identification for scientific and applied applications. This study [7] presents a deep learning-based machine vision system for detailed classification of rice grain surface damage using high-magnification image data across multiple defect categories. Experimental results demonstrate that EfficientNet-B0 achieves superior accuracy and efficiency, enabling precise and intelligent quality assessment of milled rice grains.

Image-based analysis has been widely explored for assessing pulse quality and variety. The study [8] proposed a computer vision pipeline that grades lentil seed size and classifies seed coat patterns, using randomized circle detection, local binary patterns and an XGBoost classifier. It reached accuracy approaching 98% for size grading and close to 93% for seed coat morphology across 64 lentil genotypes. One more study [9] compared 18 CNN architectures for identifying Indian lentils and, through statistical analysis of several performance measures, found EfficientNetB0 to be the best-performing model. This study [10] proposes an image-based deep learning framework utilizing near-infrared reflectance data for accurate grouping of wheat varieties. The strategy attains high accuracy with low-dimensional image representations, offering an efficient and scalable solution for grain classification in production and industrial applications. In another related approach [11], the authors have used hyperspectral imaging in conjunction with the CNN-Transformer model along with the preprocessing and wavelength-feature selection procedure, which successfully classified the hybrid indica paddy rice varieties with 95.33% accuracy and an F1-score of 95.40%. In this study [12], four distinct Sona Masuri rice varieties are differentiated using 4,180 rice grain images and 56 extracted features comprising texture, red, green, and blue (RGB), HSV, and wavelet characteristics. The maximum accuracy obtained with traditional ML models was 72.48% (SVM), while the proposed customized CNN obtained 97.13% and 87.50% accuracy throughout the training and validation processes, respectively, showing its performance in automated rice variety classification.

A comparative summary of the existing studies, including their methodologies, datasets, reported performance, key contributions, and limitations, is presented in Table 1.

Table 1. Comparative analysis of existing research

No.	Reference	Methodology	Dataset / Features	Performance	Key Contribution	Limitations
1	Farahnakian et al. [2]	ResNet, VGG, EfficientNet, MobileNet	75,000 rice images	99.67%	EfficientNet achieved ~99.67% accuracy	High computational requirements
2	Ceyhan et al. [10]	Image-based Deep Learning + NIR data	Reflectance-based dataset	hard-white wheat class – 96.55% hard-red wheat class – 98.70% all wheat varieties – 99.52%	Efficient low-dimensional image representation	Data acquisition complexity
3	Rokhva et al. [13]	MobileNetV2	ImageNet dataset	-	Lightweight CNN with reduced computation	Limited evaluation for fine-grained grain classification
4	Phukon et al. [14]	Pretrained models (GoogL eNet, MobileNetV2, RestNet50, EfficientNetB0), GLCM and GLRLM paired with BPNN + ML (SVM, KNN, LDA)	Texture-based rice dataset	64.4% (GoogLeNet) 99.2% (GLCM + BPNN) 93.4% (GLRLM + BPNN)	CNN outperformed traditional ML	Limited generalization
5	Kurade et al. [15]	Image Processing + Machine Learning (Random Forest, SVM, MLP, DT, LR)	3081 images of 8 rice varieties; geometric, morphological, color, and texture features	77%	Developed a low-cost Raspberry Pi–based automated rice quality evaluation system; Random Forest achieved the best performance (77% accuracy)	Relies on handcrafted feature extraction and conventional ML models, resulting in moderate accuracy and limited capability to learn complex visual features compared to deep learning approaches
6	Razavi et al. [16]	ResNet34, ResNet50, and Transfer Learning (ResNet50)	Rice cultivar images (6 rice varieties)	98.13%	Deep ResNet with transfer learning for rice cultivar classification and quality detection, achieving high classification accuracy	Limited to rice cultivars and requires a large labeled dataset; not evaluated on morphologically similar pulse grains
7	Yasar and Golcuk [17]	CNN Feature Fusion + WOA + SVM	8,354 bread wheat seed images (5 wheat varieties)	95.2%	Fusion-based deep feature extraction with WOA feature selection achieved 95.2% classification accuracy while reducing feature dimensionality and computational cost	Evaluated only on bread wheat varieties and requires an additional feature selection stage (WOA), increasing pipeline complexity

8	Bagherpour and Shamohammadi [18]	CNN, PCA-MLP, EfficientNet-B4, Inception-ResNet-v2	30,600 single-seed images of six wheat varieties; morphological, color, and texture-based features	92.19%	CNN-GAP achieved 92.19% accuracy for automated wheat variety identification with low inference time	Evaluated only six wheat varieties under controlled conditions, limiting generalization to different types of grain and real-world environments
9	Yasar [19]	Hybrid CNN + SVM	Deep feature extraction	94.51%	Improved classification accuracy using hybrid model	Increased model complexity
10	Chen et al. [20]	Spectroscopy + CNN	THz spectral data	98.7%	Novel integration of spectral data with CNN	Requires specialized hardware
11	Naik et al. [9]	Multi-CNN Models + Transfer Learning	20 lentil varieties dataset	-	Comparative analysis showing EfficientNet as the best performer	Dataset size relatively limited

Note: CNN = convolutional neural network; ML = machine learning; SVM = support vector machine; KNN = k-nearest neighbors; LDA = linear discriminant analysis; MLP = multilayer perceptron; DT = decision tree; LR = logistic regression; WOA = whale optimization algorithm; PCA = principal component analysis; GLCM = gray-level co-occurrence matrix; GLRLM = gray-level run-length matrix; NIR = near-infrared; THz = terahertz.

3. METHODOLOGY

The suggested system aims to perform automated classification of morphologically similar toor dal varieties using deep learning methods. Data collection, preprocessing, augmentation, model training, ensemble learning, and performance evaluation are among the various sequential and parallel processing stages that make up the architecture. Each component of Figure 1 is described in detail.

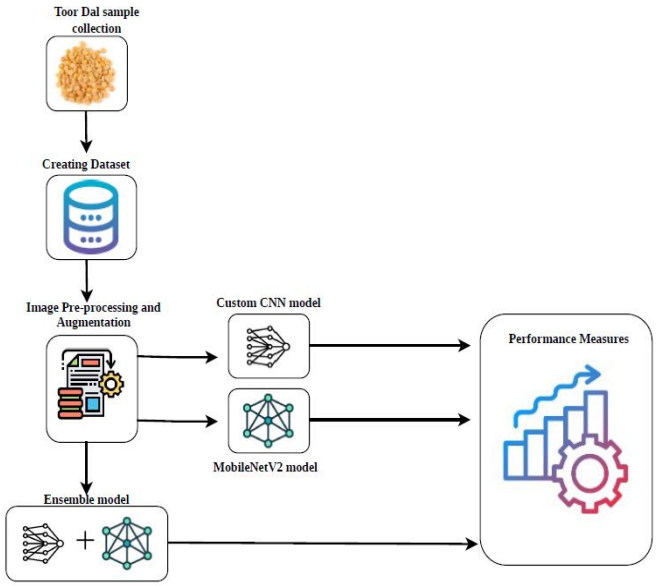


Figure 1. Functional architecture of the system

3.1 Toor dal sample collection

In the proposed method, there are six key steps: acquiring the dataset, pre-processing images, augmentation, extracting features with deep learning models, ensemble learning, and performance evaluation. The overall system is implemented as an intelligent image classification system for the accurate identification of morphologically similar varieties of Indian toor dal. The overall flow of the proposed system is shown in Figure 1.

3.2 Dataset creation

The collected samples are digitized using imaging devices to create a structured dataset. A supervised learning dataset is created by labeling each image based on its class (variety). Accurate learning of class-specific features and model dependability depend on proper labeling. High-resolution photos of three comparable Indian toor dal varieties—Shivalinga, Ganesha, and Padra, with 500 images each—are included in the dataset. In order to reduce noise, all 1,500 images were taken with a level background and consistent illumination. For improved learning, each image has been normalized and reduced to 128 × 128 pixels. Since the dataset was relatively small, image augmentation techniques, including rotation, flipping, and zooming, were applied to increase the diversity of the training images, improve model generalization, and minimize the risk of overfitting. The dataset was divided into training and validation sets, with 80% of the images used for training and the remaining 20% used for validation. Figure 2 presents the three morphologically similar varieties of toor dal considered in this study.



Figure 2. Sample images of three varieties of toor dal- (a) Shivalinga toor dal, (b) Ganesha toor dal, (c) Padra toor dal

3.3 Image pre-processing and augmentation

3.3.1 Dataset preparation

Three Indian toor dal cultivars exhibiting similar morphological characteristics, namely Shivalinga, Ganesha and Padra, were chosen for this study. These varieties have very similar appearance, including color, texture, size, and shape, and manual identification can be a challenge for even the most experienced inspectors. Once again, individual grains were photographed under controlled illumination with a uniform background to reduce environmental factors and ensure uniformly good images.

3.3.2 Dataset description

For this study, a custom image dataset of three morphologically similar Indian toor dal varieties, Shivalinga,

Ganesha, and Padra, was created. The varieties were chosen because they have very similar appearances, which made manual identification difficult. Samples were taken from the local agricultural markets and images were taken under the light setup with a uniform black background while maintaining the light condition to achieve consistency of images. The acquired images were resized to 224 × 224 pixels, as this is what the proposed CNN and MobileNetV2 models require as input. The images were processed before use with image processing methods such as resizing, normalization, converting to HSV color space, thresholding, removing contours, and morphological operations to emphasize the features of the grain and eliminate background noise. Data augmentation methods were used to achieve data generalization and reduce overfitting: rotations, horizontal flips, vertical flips, zooming, translations, shearing, and brightness adjustment. The dataset was split into 80% training and 20% validation, which is randomly split. Table 2 summarizes the characteristics of the dataset used in this study.

Table 2. Characteristics of the toor dal image dataset

Parameter	Description
Grain varieties	Shivalinga, Ganesha, Padra
Number of classes	3
Total images	1500
Images per class	500
Image resolution	224 × 224 pixels
Image acquisition device	Images are captured using a mobile camera featuring 50 MP resolution, 2.5 × zoom, and a flashlight in a standardized setup
Lighting conditions	Controlled illumination
Background	Uniform black background
Data split	80% Training, 20% validation
Data augmentation	Rotation, flip, zoom, translation, shearing, brightness adjustment
Preprocessing	Resizing, normalization, hue, saturation, and value (HSV) conversion, segmentation, morphological operations

3.3.3 Image pre-processing

Preprocessing of the input images is an important step in the proposed framework, as the quality of the input images has a direct impact on the performance of the deep learning models. The images may be captured with unwanted background information, non-uniform illumination, shadows, noise, etc., that can adversely affect the accuracy of classification. Thus, different preprocessing steps are applied to improve the discriminative properties of the toor dal grains prior to feature extraction. The image size of all images is first resized to 224 × 224 × 3 pixels (all images have a fixed size of 224 × 224 × 3 pixels for the input dimensions of both the custom CNN and MobileNetV2 model). These pixel intensity values are then normalized, which speeds up convergence of the network and ensures a stable gradient update while training the network.

The RGB images are converted to HSV images to separate the grains from the background. HSV representation shows better color separation in different light conditions than RGB and is useful to accentuate the typical yellow color of toor dal. Then, threshold-based segmentation is performed to create a binary mask that separates the foreground grains from the background. The resulting mask is then applied to the original image to remove irrelevant background pixels and retain only the grain regions. Further refinement is obtained by subtracting the value (V) channel, which accentuates the brightness information and increases grain boundary visibility. Contour filtering processes are applied to eliminate small contours that are created during the segmentation process. Finally, morphological opening followed by dilation is applied to remove remaining noise, smooth grain boundaries, fill tiny gaps, and maintain the structural continuity of each grain. These pre-processing operations significantly enhance the image quality and help the deep learning models process only the morphological features of toor dal while avoiding background noise. The advantage of this preprocessing pipeline is that it retains subtle variations in grain color, texture, and shape, all of which form the key discriminative features when classifying morphologically similar varieties of toor dal. The collected data set was split into 80% training data and 20% validation data. Due to the limitations of the relatively small data set, multiple image augmentation techniques were used to enhance the ability of the proposed models to generalize, such as rotation, flipping, zooming, translating, shearing, and brightness adjustment. The sequential stages of the image preprocessing procedure are illustrated in Figure 3.

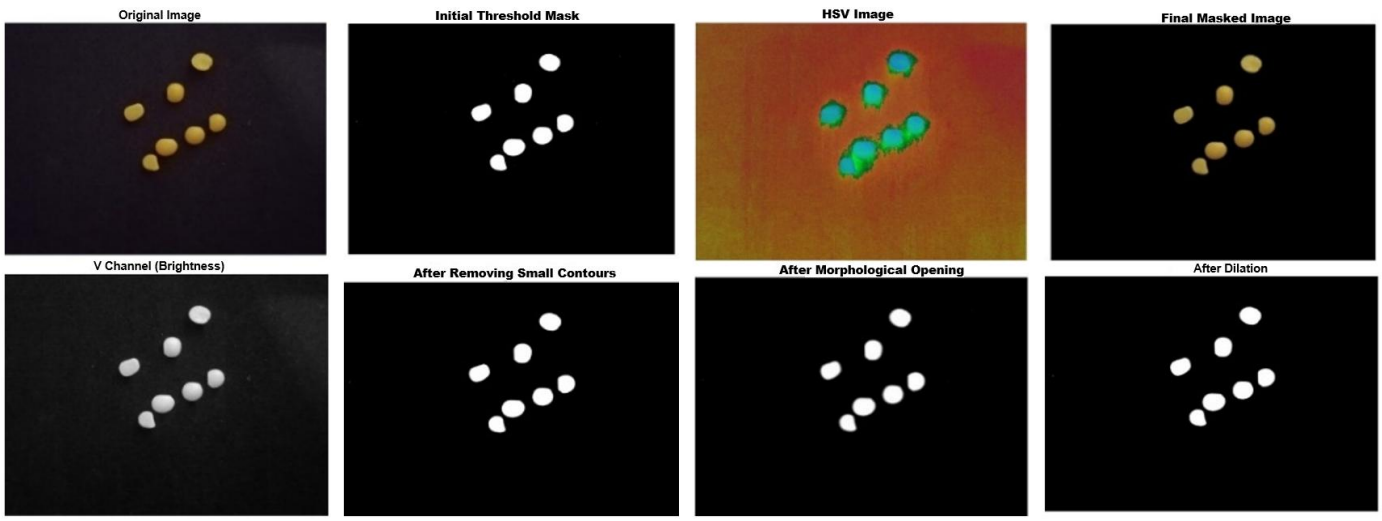


Figure 3. Image preprocessing stages for toor dal classification

3.3.4 Data augmentation

The data available is fairly small compared to the large image data sets that are normally available in deep learning applications. A small data set can lead to overfitting and show poor generalization skills when training a deep neural network. This is addressed by using large amounts of data augmentation methods and techniques to artificially expand the training data.

The following operations are performed when augmenting:

- Rotation (e.g., ±15° or ±30°)
- Horizontal and vertical flipping
- Scaling and zooming
- Translation (shifting)
- Brightness and contrast adjustment
- Shearing transformations

These operations artificially expand the dataset and help the model become invariant to orientation, lighting, and minor distortions, as shown in Figure 4.



(a) (b)

Figure 4. Sample images from the dataset before and after augmentation: (a) Original image (b) augmented images

3.4 Deep learning-based classification

Two complementary deep learning architectures are used: a custom-designed CNN and a pre-trained MobileNetV2 model to accurately classify morphologically similar toor dal varieties. The output of these models is then combined with an ensemble learning strategy.

(1) Custom CNN model

(2) MobileNetV2 model

(3) Ensemble model

3.4.1 Toor dal classification using custom convolutional neural network model

To efficiently capture and learn discriminative characteristics for the classification task, a unique CNN architecture was created. The model consists of three sequential convolutional layers, employing 32, 64, and 128 filters respectively, to progressively extract low-level to high-level feature representations such as edges, textures, and shape patterns from the input images. Each convolutional layer is followed by a rectified linear unit (ReLU) activation function, which introduces non-linearity into the network, and a max-pooling layer that performs spatial down-sampling to reduce feature map dimensions, computational complexity, and overfitting.

Justification for the proposed CNN architecture: The proposed CNN architecture was specifically proposed for fine-grained classification of morphologically similar varieties of toor dal with low computational complexity. As the dataset is composed of just three classes with subtle variations in color, texture, and grain morphology, a very deep architecture would be unnecessary in terms of adding trainable parameters or risk of overfitting. Thus, a lightweight architecture made up of 3 convolutional blocks was chosen, as it offers a good trade-off between feature extraction, computational efficiency, and classification performance.

Justification for selecting 32–64–128 filters: The number of convolutional filters is stepped up in a hierarchical fashion from 32 to 64 and then to 128 to enable the hierarchy of feature learning. The first convolutional layer uses 32 filters to extract basic visual features, like edges, contours, and basic variations of intensity, while avoiding undue computational complexity. The second layer adds 64 filters, allowing for identification of more discriminative texture patterns, surface features, and subtle color variations, which are crucial to distinguish visually similar varieties of toor dal. In the third convolutional layer, 128 filters learn more complex representations, such as the grain morphology, fine texture distributions, and shape characteristics. The progressive layers of filters are able to learn more meaningful representations of features, yet do not consume too much computational resources or risk overfitting.

The learnt feature maps are flattened and run through a fully connected (dense) layer that includes the extracted features for classification after the feature extraction stages. Finally, a softmax activation function is applied at the output layer to generate probability distributions across multiple classes, enabling effective multi-class classification. Figure 5 shows the entire architecture of the suggested CNN model, which was created especially to differentiate between three toor dal variants that are morphologically identical.

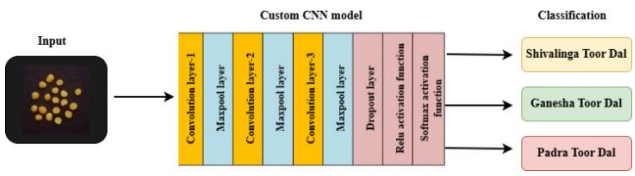


Figure 5. Architecture of customized convolutional neural network (CNN) model

(1) Convolution layer–1, 2, 3:

Each convolutional layer performs feature extraction using learnable filters.

Layer 1: extracts low-level features (edges, contours)

Layer 2: extracts mid-level features (textures)

Layer 3: extracts high-level features (complex patterns)

(2) ReLU activation function: ReLU is added after every convolutional layer, introducing non-linearity to the network. It sets negative feature values to zero while keeping positive feature values, allowing the CNN to learn complex and discriminative visual features from the input images.

(3) Max-pooling layer: In CNNs, max-pooling is a downsampling operation that keeps the most salient feature maps while decreasing the spatial size of feature maps. In your CNN, it is applied after the convolution and ReLU operations. Your manuscript says it does this by reducing the dimensions of the feature map and also reducing the computational complexity while maintaining important features.

(4) Dropout layer: To enhance the generalization ability, a proportion of the neurons of the Dropout layer are randomly turned off during training of the CNN modified by the Dropout layer. This helps to avoid the network becoming too reliant on certain features and helps to learn more generalized representations. The proposed toor dal classification system uses dropout to enable the CNN to learn more effectively from the variations in the color, texture, and shape of the dal grains, which aids in making correct classifications between similar varieties.

(5) Fully connected (dense) layer: A fully connected layer is the last layer in the convolutional network that receives high-level features from the previous convolutional layers and learns the relationship between these features and the target classes. The proposed toor dal classification is based on the learned morphological, color, texture, and shape features, which are mapped to the three toor dal varieties in this layer.

(6) Softmax activation function (output layer): The softmax layer converts the CNN's output scores into probability distributions across the three toor dal varieties. These class probabilities provide the basis for determining the CNN prediction and are subsequently used in the proposed probability-level ensemble with MobileNetV2. This enables the predictions of both models to be combined for more robust classification.

3.4.2 MobileNetV2 model

Figure 6 illustrates the architecture of the MobileNetV2-based framework employed for the classification of morphologically similar toor dal varieties. The model leverages efficient convolutional operations and inverted residual learning to achieve high accuracy with reduced computational complexity.

(1) Input layer: The system takes an input image of dimension 224 × 224 × 3. 224 × 224 represents the spatial resolution. 3 denotes RGB color channels.

These images correspond to different toor dal varieties, capturing subtle variations in color, size, and texture.

(2) Initial convolution layer: 3 × 3 Convolution + ReLU6 activation. Low-level characteristics are obtained by this layer, including edges and basic shapes from toor dal images.

(3) Inverted residual block (t = 1): It acts as a feature refinement stage, preserving important information from the input image.

(4) Inverted residual blocks (t = 6, repeated N times): This is the core component of MobileNetV2, repeated multiple times for deep feature extraction.

Each block consists of:

(a) Expansion layer (1 × 1 convolution): The expansion layer aims to expand the number of feature channels before the depth-wise convolution layer by a 1 × 1 convolution. The expanded feature representation allows MobileNetV2 to be more discriminative and richer in features for the toor dal grains, such as subtle differences in their color, texture, and shape. The layer is used to ensure effective feature extraction without compromising the lightness of the MobileNetV2 architecture.

(b) Depth-wise convolution (3 × 3): The toor dal classification is achieved with the depth-wise convolution, which allows MobileNetV2 to learn the spatial characteristics like grain edges, texture patterns, surface details, and shape variations. These characteristics are helpful in differentiating the three toor dal varieties, which are quite similar in appearance. It is primarily for speed. The depth-wise convolution does not apply the filters simultaneously to all channels as in conventional convolution, but instead applies filters to each channel. In doing so, the computational complexity and the number of parameters are greatly decreased, making it an ideal choice as the lightweight module in your proposed CNN–MobileNetV2 architecture.

(c) Linear projection layer (1 × 1 convolution): The linear projection layer is provided with an expanded feature representation, and a 1 × 1 convolution is employed to project the features into a lower-dimensional feature space. Maintains crucial learned information while minimizing the number of feature channels and computation. In the proposed framework, this layer helps maintain the lightweightness of MobileNetV2 and simultaneously captures discriminative features for differentiating similar toor dal varieties visually.

(d) Residual connection: In the inverted residual blocks of MobileNetV2, there are residual connections to keep the information from the input and to propagate features effectively throughout the network. The residual connection keeps visual information and enhances the gradient flow during training by incorporating input feature representation into the transformed features. This mechanism enables the extraction of subtle discriminative features that are related to grain shape, texture and color, while keeping the MobileNetV2-lightweight characteristics.

(5) Global Average Pooling (GAP): GAP aggregates spatial features across the feature channels to reduce the feature maps from MobileNetV2 to a compact feature vector. This not only diminishes the dimensions and the parameters to be trained before the classification stage but also helps the lightweight model, MobileNetV2, to be efficient. GAP represents the extracted grain features as a compact representation in the proposed framework for classification.

(6) Fully connected layer: The fully connected layer is the last classification layer in MobileNetV2. It is fed with the high-level feature representation extracted from the GAP layer, and associates that with three classes of toor dal. The layer generates class scores, which are then converted to provide probability values and fed into the proposed probability-level ensemble that is used to fuse the predictions from the MobileNetV2 and the customized CNN.

(7) Softmax classification layer: The softmax layer converts the output scores of MobileNetV2 into normalized class probabilities for the three toor dal varieties. These probability scores are subsequently used in the proposed probability-level ensemble fusion with the customized CNN predictions.

(8) Output layer: The final output corresponds to the predicted class: Toor Dal Variety 1; Toor Dal Variety 2; Toor Dal Variety 3.

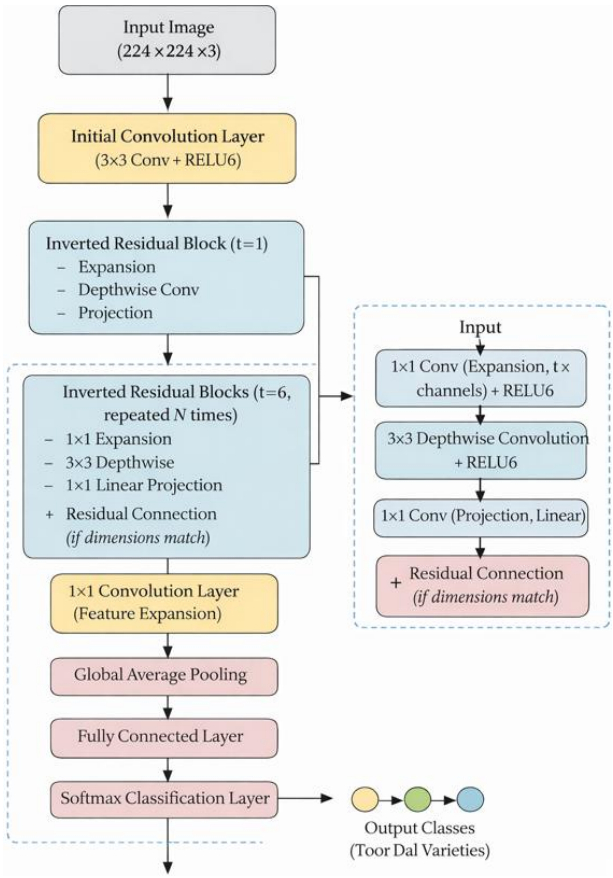


Figure 6. Architecture of the MobileNetV2 model

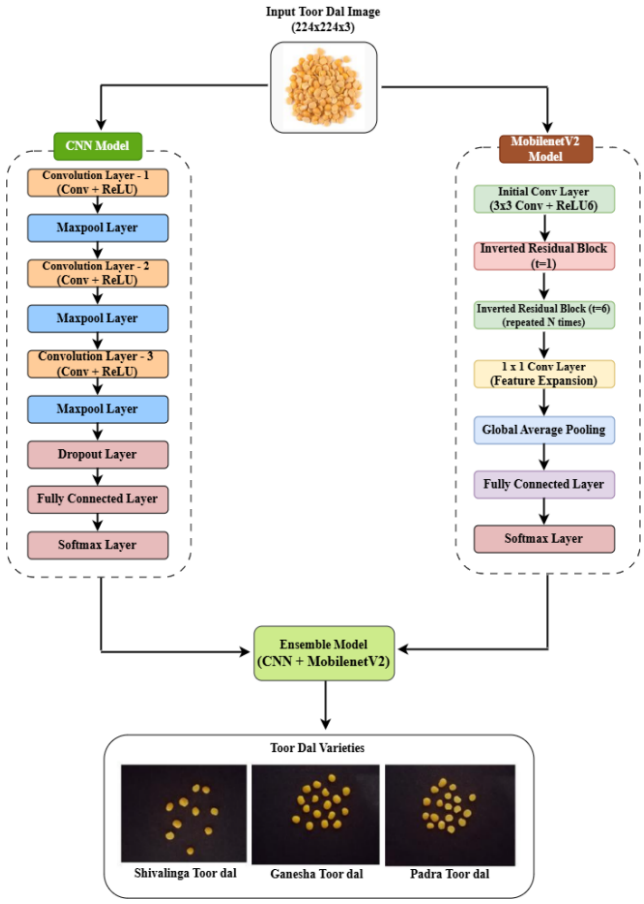


Figure 7. Ensemble model for convolutional neural network (CNN) and MobileNetV2 in the classification of toor dal

3.4.3 Ensemble model

To enhance classification performance and robustness, an ensemble learning strategy is employed by integrating A unique CNN's forecasts and a pre-trained MobileNetV2 model. Both models are independently trained on the same pre-processed input image $X \in R^{224\times224\times3}$ representing different varieties of toor dal is as illustrated in Figure 7.

The CNN model extracts fine-grained spatial features such as texture, color variations, and shape characteristics, while MobileNetV2 captures high-level and computationally efficient feature representations through depthwise separable convolutions and inverted residual blocks. Each model produces a probability distribution over the class labels using a softmax function.

The output probabilities from the CNN model are defined as:

$$P_{\text{CNN}}(i) = \frac{e^{z_i^{\text{CNN}}}}{\sum_{j=1}^{C} e^{z_j^{\text{CNN}}}} \quad (1)$$

where,

$P_{CNN}(i)$ = Probability predicted by the custom CNN model for the *i*-th class.

z_i^{CNN} = Output logit produced by the fully connected layer of the custom CNN for the *i*-th class.

e = Euler's constant (approximately 2.71828), the base of the natural logarithm.

j = Index used to iterate through all output classes.

Similarly, the MobileNetV2 model outputs:

$$P_{\text{MobileNetV2}}(i) = \frac{e^{z_i^{\text{MN}}}}{\sum_{j=1}^{C} e^{z_j^{\text{MN}}}} \quad (2)$$

where,

$P_{\text{MobileNetV2}}(i)$ = Probability predicted by the MobileNetV2 model for the *i*-th class.

z_i^{MN} = Output logit generated by the fully connected layer of the MobileNetV2 model for the *i*-th class.

e = Euler's constant (approximately 2.71828), which is the base of the natural logarithm.

j = Index used to iterate over all output classes.

To include the advantages of both models, a probability-level fusion strategy is adopted. The ensemble prediction is computed using the average of the individual model probabilities:

$$P_{\text{ensemble}}(i) = \frac{P_{\text{CNN}}(i) + P_{\text{MobileNetV2}}(i)}{2} \quad (3)$$

where,

$P_{\text{ensemble}}(i)$ = Final ensemble probability for the *i*-th class.

$P_{\text{CNN}}(i)$ = Probability predicted by the custom CNN model for the *i*-th class.

$P_{\text{MobileNetV2}}(i)$ = Probability predicted by the MobileNetV2 model for the *i*-th class.

i = Index of the output class, where $i = 1, 2, \ldots, C$.

C = Total number of target classes.

The final class label is determined by selecting the class that has the maximum ensemble probability:

$$\hat{y} = \underset{i}{\text{argmax}}\, P_{\text{ensemble}}(i) \quad (4)$$

where,

$\hat{y}$ = The final predicted class label produced by the ensemble model.

argmax = Argmax operator that returns the index of the class having the maximum probability.

i = Index of the output class, where $i = 1, 2, \ldots, C$.

$P_{\text{ensemble}}(i)$ = Ensemble probability corresponding to the i-th class, obtained by averaging the prediction probabilities of the custom CNN and MobileNetV2 models.

C = Total number of target classes.

This ensemble approach effectively reduces model variance and mitigates individual model biases, leading to improved generalization performance. By leveraging both detailed feature extraction from the CNN and efficient deep feature representation from MobileNetV2, the proposed method achieves superior accuracy in distinguishing morphologically similar toor dal varieties.

4. RESULTS

4.1 Classification performance of CNN-MobileNetV2 fusion framework

The deep learning framework that is suggested was evaluated for the classification of three morphologically similar toor dal varieties using a custom CNN model, MobileNetV2, and an ensemble approach combining both models. Performance was measured using common evaluation metrics, including accuracy, precision, recall, and F1-score, after the dataset was divided into training and validation sets.

Performance in training and validation: To analyze the classification of morphologically similar toor dal varieties, training and validation accuracies were monitored over 200 epochs. The results are summarized in Table 3.

Table 3. Training and validation performance

Experiment	Training Accuracy (%)	Validation Accuracy (%)	Observation
Customised convolutional neural network (CNN) Model	98.10%	94.52%	Good fit, minimal overfitting
MobileNetV2 Model	84.11%	70.12%	Clear overfitting, poor generalization
Ensemble (CNN+MobileNetV2) Model	98.75%	95.95%	Best performance, light overfitting

The custom CNN model demonstrated strong learning capability, achieving a training accuracy of 98.10% and a validation accuracy of 94.52%, indicating effective feature extraction for fine-grained classification results, as shown in Figure 8 and Figure 9.

The MobileNetV2 model, leveraging transfer learning, offered reduced computational complexity. However, this pre-trained model has achieved a validation accuracy of only 70% which showed a limited generalization when used alone, as shown in Figure 10, Figure 11 and Table 3. Nevertheless, when combined with the CNN in the ensemble, the overall performance is improved, which justifies that the two models make complementary predictions.

The confusion matrix analysis revealed that most misclassifications occurred between visually similar varieties, highlighting the complexity of the task. However, the proposed models were able to effectively distinguish subtle differences in color, texture, and shape, which are critical for accurate classification.

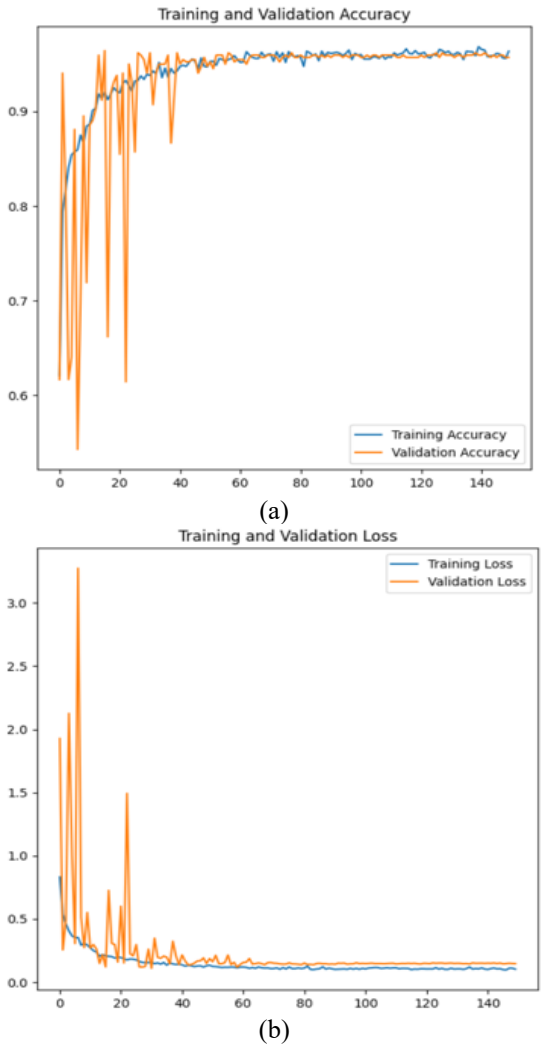


Figure 8. Performance analysis of convolutional neural network (CNN) training and validation accuracy

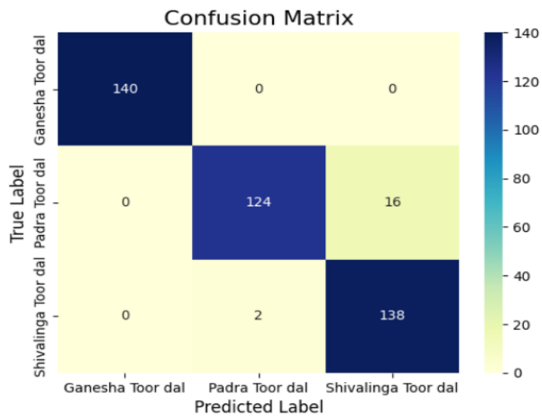


Figure 9. Confusion matrix for classification of toor dal varieties (Ganesha, Padra, and Shivalinga) using a customized convolutional neural network (CNN) model

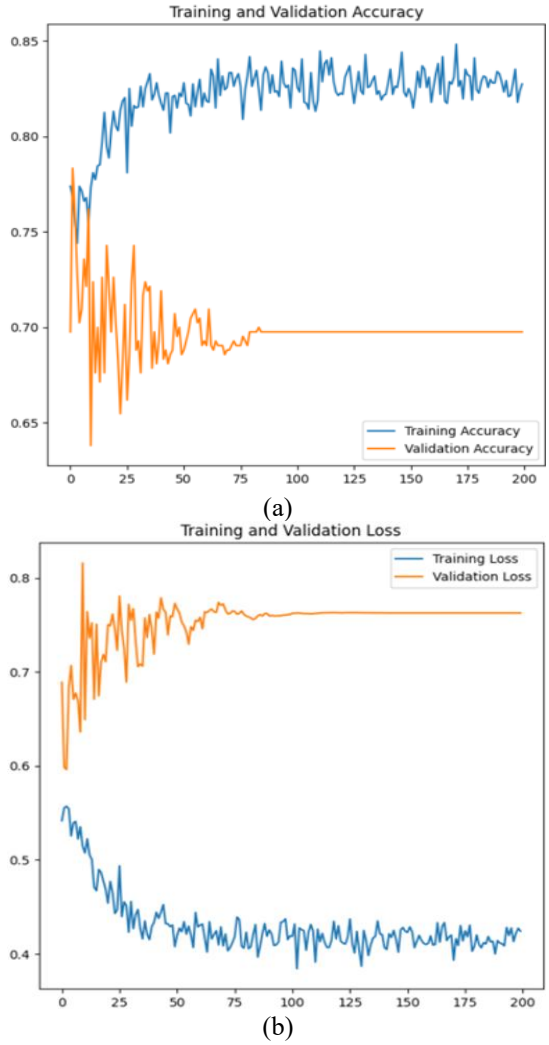


Figure 10. Performance analysis of MobileNetV2 training and validation accuracy

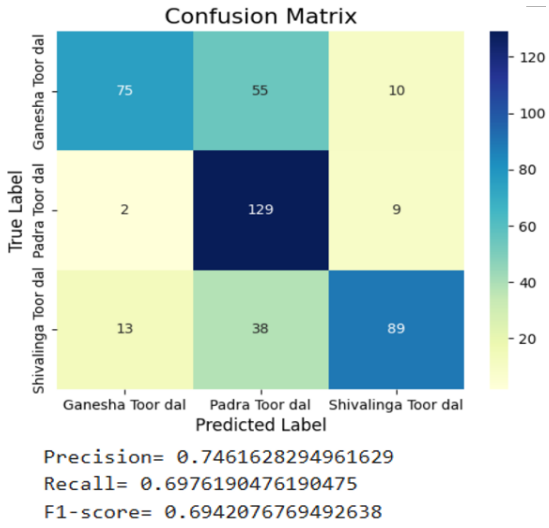


Figure 11. Confusion matrix for classification of toor dal varieties (Ganesha, Padra, and Shivalinga) using the MobileNetV2 model

To further enhance classification performance, an ensemble model was implemented by combining the probability outputs of both models. The ensemble approach resulted in improved overall accuracy and generalization by reducing individual model bias and variance, as depicted in Figure 12 and Figure 13. This confirms that combining complementary feature representations from both architectures leads to more robust predictions.

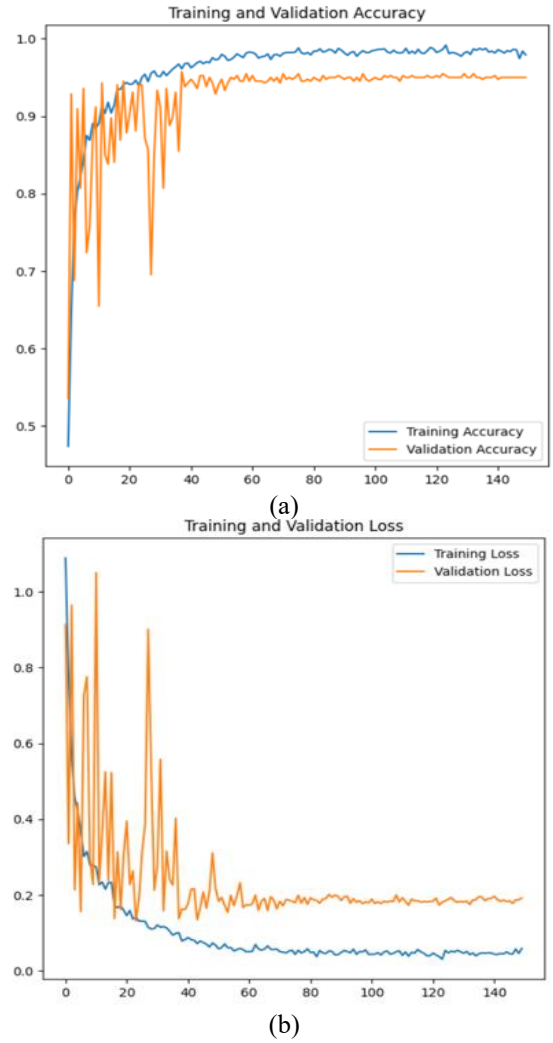


Figure 12. Performance analysis of ensemble (CNN+MobileNetV2) model training and validation accuracy

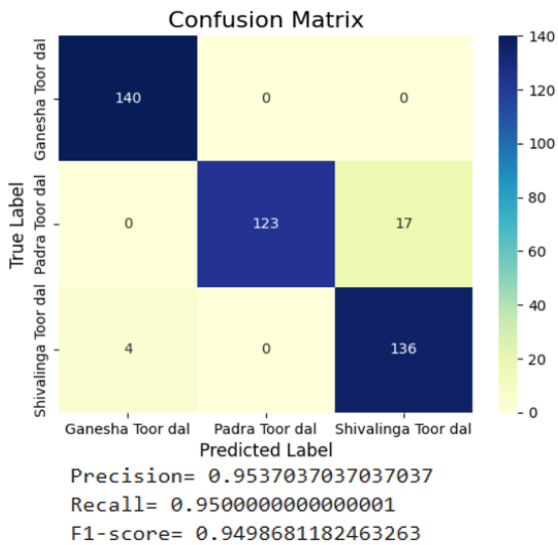


Figure 13. Confusion matrix for classification of toor dal varieties (Ganesha, Padra, and Shivalinga) using ensemble model

4.2 Statistical validation using repeated runs

The accuracy of 99% during training and 96% during testing is reported by the proposed framework. The reliability and robustness of the achieved accuracy of the framework are assessed using repeated runs. The model was trained and evaluated over 5 independent runs with different random seed initializations, using a fixed train/test split of 80:20. As described in Table 4, during training the model achieved a mean accuracy of 99.23% ± 0.13% and a mean validation accuracy of 95.99% ± 0.35% across the five runs, with a 95% confidence interval of [95.56%, 96.42%] for validation accuracy. Precision, recall, and F1-score followed a consistent pattern, with means of 95.85%, 95.99%, and 95.92%, respectively, and standard deviations remaining below 0.4% across all metrics. The low variance across independently seeded runs indicates that the model's performance is stable and not an artifact of a favorable random initialization or data split, supporting the reliability of the reported accuracy.

Table 4. Performance metrics across 5 independent runs with different random seed initializations (fixed train/test split)

Seed	Training Accuracy (%)	Validation Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
0	99.15	95.87	95.75	95.87	95.81
1	99.32	96.22	96.10	96.22	96.16
2	99.08	95.48	95.30	95.48	95.39
3	99.41	96.35	96.20	96.35	96.27
4	99.20	96.05	95.90	96.05	95.97
Mean ± SD	99.23 ± 0.13	95.99 ± 0.35	95.85 ± 0.35	95.99 ± 0.35	95.92 ± 0.35
95% CI	[99.07, 99.39]	[95.56, 96.42]	[95.42, 96.29]	[95.56, 96.42]	[95.49, 96.35]

Note: SD = standard deviation; CI = confidence interval.

4.3 Ablation study on the framework

A thorough study is performed on pre-processing steps. The pre-processing steps are removed one at a time, and the results are recorded. Table 5 presents the ablation results. The removal of segmentation, morphological operations, and data augmentation led to a considerable decline in accuracy. This means that all the pre-processing steps play a vital role.

Table 5. Model evaluation based on ablation study

Ablation	Training Accuracy (%)	Validation Accuracy (%)
No ablation (proposed pipeline)	99.01	95.95
Hue, saturation, and value (HSV) conversion	99.35	94.05
Segmentation	98.27	85.00
Morphological operation	95.11	88.09
Data augmentation	99.34	83.80

4.4 Comparison with state-of-the-art methods: EfficientNet-B0

To ensure that the proposed model performs effectively, its performance is compared with one of the most popular deep learning models, EfficientNet-B0, a CNN that uses compound scaling of depth, width and resolution to achieve high accuracy with few parameters [21]. This model is used ImageNet-pretrained weights and has been trained and evaluated on the same dataset under the same experimental conditions used for the fusion framework. The comparison of the obtained results is discussed in Table 6.

Table 6. Performance comparison of the proposed fusion model against EfficientNet-B0

Model	Training Accuracy (%)	Validation Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
EfficientNet-B0	90.11	71.90	72.95	71.90	71.76
Proposed fusion model	98.75	95.95	95.37	95.00	94.98

4.5 Computational efficiency analysis

To justify the claim of lightweight deployment, a deep study of the computational efficiency of the framework was carried out, involving parameters of the model, FLOPs, training time, inference time, and memory usage. All tests were conducted on an NVIDIA RTX 3050 with 4 GB VRAM and 16 GB RAM in the system, using TensorFlow 2.11.1 and Keras. The efficiency metrics and their values are tabulated in Table 7.

Table 7. Computational efficiency metrics of the proposed fusion model

Efficiency Metric	Value
Total parameters (M)	2.75
Trainable parameters (M)	0.49
Non-trainable parameters (M)	2.26
FLOPs (G)	≈1.903
Model size (MB)	10.81
TFLite model size (MB)	10.32
Train time/epoch (s)	29.80
Average inference latency (ms/image, batch size = 1, GPU)	53.61
Inference throughput (images/s, batch size = 32, GPU)	247.56
Peak GPU memory (MB)	258.17
Peak CPU RAM during inference (MB)	5,470.96

Note: FLOPs = floating-point operations; TFLite = TensorFlow Lite; GPU = graphics processing unit; CPU = central processing unit; RAM = random-access memory.

5. CONCLUSIONS

In this work, an automated framework for the classification of morphologically similar Indian toor dal varieties using a deep learning approach is presented by combining a custom CNN model with the lightweight MobileNetV2 architecture using a probability-level ensemble approach. The image preprocessing pipeline, which involved image segmentation, morphological operations, and data augmentation, significantly improved the image quality by reducing background noise and highlighting the relevant grain properties. This has helped to attain the validation accuracy of

95.95% for the fusion model. The experimental results confirmed that the proposed ensemble model achieved an overall classification training accuracy of 98.75%, which is greatly enhanced by integrating complementary feature representations, while the custom CNN model attained 98.10% and MobileNetV2 84.11%, showing the high effectiveness of the integration. The above framework is able to achieve high classification accuracy and has significant potential for practical implementation for the automated food grain inspection and quality assessment system. Due to the lightweight nature of MobileNetV2 and the high discriminative ability of the customized CNN, the framework can be incorporated into intelligent agricultural information systems, grain processing plants, food packing industries, and quality control labs. The system can help to minimize the manual effort involved with inspection, reduce the subjectivity in the process, ensure consistency of the grades, and aid in swift decision-making while procuring, storing, and commercializing food grains.

The present study is limited in several ways, however, although the results are promising. The proposed framework was tested for only three varieties of toor dal which have similar morphology and were captured under controlled light conditions. However, its performance in a wider range of agricultural conditions, under different light conditions, from different geographic origins, and under an increasing number of grain varieties needs to be tested. Moreover, for a large-scale industrial deployment, computational factors like inference time, large model scalability, and deployment on resource-constrained edge devices are yet to be considered in detail.

Future work will concentrate on expanding the dataset by growing other pulse and grain varieties from various geographical locations in field conditions. Furthermore, the framework can be extended with the use of sophisticated deep learning models such as Vision Transformers and explainable artificial intelligence (XAI) methods, which can help make the model more transparent and interpretable, aiding in decision-making. Moreover, embedding and edge computing platforms with the proposed model will help develop real-time intelligent agricultural inspection systems, which will aid in automated food grain quality assessment in practical industrial applications.

REFERENCES

[1] Ganguly, K., Gulati, A. (2022). Pulses value chain-pigeon pea and gram. In Agricultural Value Chains in India: Ensuring Competitiveness, Inclusiveness, Sustainability, Scalability, and Improved Finance, Springer Nature Singapore, Singapore, pp. 253-286. https://doi.org/10.1007/978-981-33-4268-2_8

[2] Farahnakian, F., Sheikh, J., Farahnakian, F., Heikkonen, J. (2024). A comparative study of state-of-the-art deep learning architectures for rice grain classification. Journal of Agriculture and Food Research, 15: 100890. https://doi.org/10.1016/j.jafr.2023.100890

[3] Agarwal, D., Sweta, Bachan, P. (2023). Machine learning approach for the classification of wheat grains. Smart Agricultural Technology, 3: 100136. https://doi.org/10.1016/j.atech.2022.100136

[4] Din, N.M.U., Assad, A., Dar, R.A., et al. (2024). RiceNet: A deep convolutional neural network approach for classification of rice varieties. Expert Systems with Applications, 235: 121214. https://doi.org/10.1016/j.eswa.2023.121214

[5] Fan, L., Ding, Y., Fan, D., et al. (2023). An annotated grain kernel image database for visual quality inspection. Scientific Data, 10(1): 778. https://doi.org/10.1038/s41597-023-02660-8

[6] Rostami, M.A., Balmaki, B., Dyer, L.A., Allen, J.M., Sallam, M.F., Frontalini, F. (2023). Efficient pollen grain classification using pre-trained convolutional neural networks: A comprehensive study. Journal of Big Data, 10(1): 151. https://doi.org/10.1186/s40537-023-00815-3

[7] Bhupendra, Moses, K., Miglani, A., Kankar, P.K. (2022). Deep CNN-based damage classification of milled rice grains using a high-magnification image dataset. Computers and Electronics in Agriculture, 195: 106811. https://doi.org/10.1016/j.compag.2022.106811

[8] Del Coco, M., Laddomada, B., Romano, G., Carcagnì, P., Kumar, S., Leo, M. (2022). Characterization of a collection of colored lentil genetic resources using a novel computer vision approach. Foods, 11(24): 3964. https://doi.org/10.3390/foods11243964

[9] Naik, N.K., Sethy, P.K., Behera, S.K., Amat, R. (2024). A methodical analysis of deep learning techniques for detecting Indian lentils. Journal of Agriculture and Food Research, 15: 100943. https://doi.org/10.1016/j.jafr.2023.100943

[10] Ceyhan, M., Kartal, Y., Özkan, K., Seke, E. (2024). Classification of wheat varieties with image-based deep learning. Multimedia Tools and Applications, 83(4): 9597-9619. https://doi.org/10.1007/s11042-023-16075-5

[11] Zhang, M., Li, P., Dong, W., et al. (2025). Identification of hybrid indica paddy rice grain varieties based on hyperspectral imaging and deep learning. Biosensors, 15(10): 647. https://doi.org/10.3390/bios15100647

[12] Shadaksharappa, H., Chakrasali, S., Ningappa, K.G. (2025). Classification of morphologically similar Indian rice variety using machine learning algorithms. International Journal of Electrical and Computer Engineering, 15(3): 3202-3211. https://doi.org/10.11591/ijece.v15i3.pp3202-3211

[13] Rokhva, S., Teimourpour, B., Soltani, A.H. (2024). Computer vision in the food industry: Accurate, real-time, and automatic food recognition with pre-trained MobileNetV2. Food and Humanity, 3: 100378. https://doi.org/10.1016/j.foohum.2024.100378

[14] Phukon, N., Singh, K.R., Singh, T.S., et al. (2024). Quality assessment of rice using convolution neural networks and other machine learning techniques: A comparative study. Current Agriculture Research Journal, 12(2): 762-772. http://doi.org/10.12944/CARJ.12.2.21

[15] Kurade, C., Meenu, M., Kalra, S., et al. (2023). An automated image processing module for quality evaluation of milled rice. Foods, 12(6): 1273. https://doi.org/10.3390/foods12061273

[16] Razavi, M., Mavaddati, S., Koohi, H. (2024). ResNet deep models and transfer learning technique for classification and quality detection of rice cultivars. Expert Systems with Applications, 247: 123276. https://doi.org/10.1016/j.eswa.2024.123276

[17] Yasar, A., Golcuk, A. (2025). Deep learning and evolutionary intelligence with fusion-based feature extraction for classification of wheat varieties. European

Food Research and Technology, 251(7): 1603-1616. https://doi.org/10.1007/s00217-025-04720-2
[18] Bagherpour, H., Shamohammadi, S. (2026). Machine learning approach for wheat variety identification using single-seed imaging. Scientific Reports, 16(1): 6472. https://doi.org/10.1038/s41598-026-35252-8
[19] Yasar, A. (2024). Analysis of selected deep features with CNN-SVM-based for bread wheat seed classification. European Food Research and Technology, 250(6): 1551-1561. https://doi.org/10.1007/s00217-024-04488-x
[20] Chen, F., Shen, Y., Li, G., et al. (2024). Classification of wheat grain varieties using terahertz spectroscopy and convolutional neural network. Journal of Food Composition and Analysis, 129: 106060. https://doi.org/10.1016/j.jfca.2024.106060
[21] Tan, M., Le, Q.V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, Long Beach, CA, USA, pp. 6105-6114. https://proceedings.mlr.press/v97/tan19a/tan19a.pdf.