**CHAPTER 5:** White whole cashew classification using hybrid models

* 1. **Introduction:**

Convolutional Neural Networks (CNNs) have emerged as a powerful tool in image classification, setting a high standard in various large-scale identification tasks. These networks utilize deep learning algorithms to analyze and classify images with remarkable accuracy, a quality that has positioned CNNs as a preferred method in the field of machine learning. Currently, cashew grading process relies heavily on manual sorting, which is both time-consuming and labor-intensive. To address this problem study explores the use of hybrid CNN models for cashew classification, combining CNN architectures such as VGG16, Inception-V3, and ResNet-50 with various machine learning classifiers, including Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN). This study highlights the potential of CNN-based hybrid models in optimizing cashew grading, paving the way for broader applications of automated grading systems across different industries.

* 1. **Image Acquisition and Pre-processing:**

The dataset for this research focuses on five distinct grades of cashew kernels, each labeled according to industry standards: W180, W210, W300, W400, and W500. Each category contains 800 number of images The dataset thus contains a total of 4000 high-quality images, providing a robust foundation for training convolutional neural networks (CNN) in classifying cashew kernels accurately. To ensure consistency, the images were systematically captured using an iPhone 13 Pro. The camera was positioned 6 cm away from each cashew kernel on a black background, enhancing contrast and emphasizing each cashew's distinct characteristics. By maintaining this fixed distance and a controlled background, the dataset preserves uniform visual standards, making it suitable for machine learning applications. Each image is formatted in JPEG, with a resolution of 224 x 224 pixels and a 24-bit color depth, which captures fine details and colors essential for differentiating between grades. The collection process spanned approximately 3 to 5 weeks, with meticulous attention given to environmental factors such as lighting, camera distance, background color, brightness, and capture anglehomogeneous dataset that would effectively highlight the natural variations among the different cashew kernel grades.

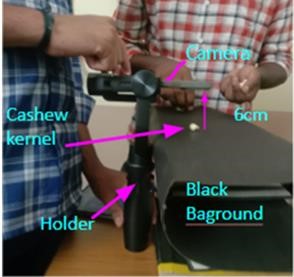
Upon capturing, the images underwent a series of pre-processing steps designed to standardize them for the CNN model. These steps included resizing to ensure all images were uniform in dimensions, noise reduction to remove any unwanted visual artifacts, and background removal to focus solely on the cashew kernels. This standardization step was crucial in creating a dataset that would not only maintain consistency but also reduce the computational load on the model during training.

Fig. 5.1: Image acquisition of white whole cashews

To increase the dataset's diversity and improve the CNN model's ability to generalize across real-world cases, several image augmentation techniques were applied. Augmentation techniques included random rotations, scaling, cropping, and horizontal and vertical flipping. These transformations created multiple variations of each image, expanding the dataset with new examples that simulate different viewing angles and perspectives. This augmentation step helps prevent overfitting, enabling the model to learn robust features from a broader set of visual patterns and effectively distinguishing between cashew grades. Once pre-processed and augmented, the dataset was used to train deep convolutional neural network (DCNN) models. These models were then combined with additional classifiers, creating hybrid architectures aimed at precise classification. The hybrid models took advantage of the feature-extraction capabilities of CNNs, while the additional classifiers refined the predictions, boosting accuracy and robustness across cashew types. By combining these components, the research aims to achieve a sophisticated classification system that can distinguish subtle differences between cashew kernel grades and classify each sample accurately.

This structured and detailed approach to data collection, pre-processing, augmentation, and model training lays the foundation for an efficient classification pipeline. It ensures the resulting model can not only identify specific grades of cashew kernels with high accuracy but also adapt to variations, making it a valuable tool for applications in quality control and grading in the cashew industry.

* 1. **Implementation of CNN model for cashew classification**

Convolutional neural networks (CNNs) were implemented for cashew nut classification, focusing on the use of high-performing models such as VGG16, InceptionV3, ResNet50. Each of these CNN models were designed to automatically learn and extract complex patterns and distinguishing features from cashew images, enabling precise classification based on visual characteristics.

* + 1. **VGG16**

The VGG16 model, known for its deep layers, is highly proficient in capturing multi-scale features essential for image classification. In its initial layers, VGG16 identifies low-level visual attributes, such as edges, textures, and corners, as described by convolutional filters that detect these characteristics at a small receptive field. As the network deepens, the layers begin to interpret more complex patterns, which are essential in recognizing detailed object components, shapes, and intricate textures. In applications such as cashew grading, VGG16 provides detailed feature extraction capabilities, capturing both fine-grained and large-scale characteristics. These may include minor surface defects, like cracks and spots (fc) or broader structural differences, such as shape and size (fs). Mathematically, if f(x) represents the function of the VGG16 model that applies convolutional layers to the input image x, then:

where Wi and bi ​ represent the weights and biases in each layer i for L total layers in VGG16.

In the context of cashew classification, the model's depth enables it to learn key discriminative features. As each convolutional layer contributes to refining the features, the objective function of the VGG16 model can be expressed as a minimization problem, minimizing the loss between predicted and actual grades: where y represents the true label distribution, the predicted distribution, and K the number of cashew grades. This approach enhances VGG16's ability to identify class-specific textures, shapes, and sizes of cashew kernels with high precision.

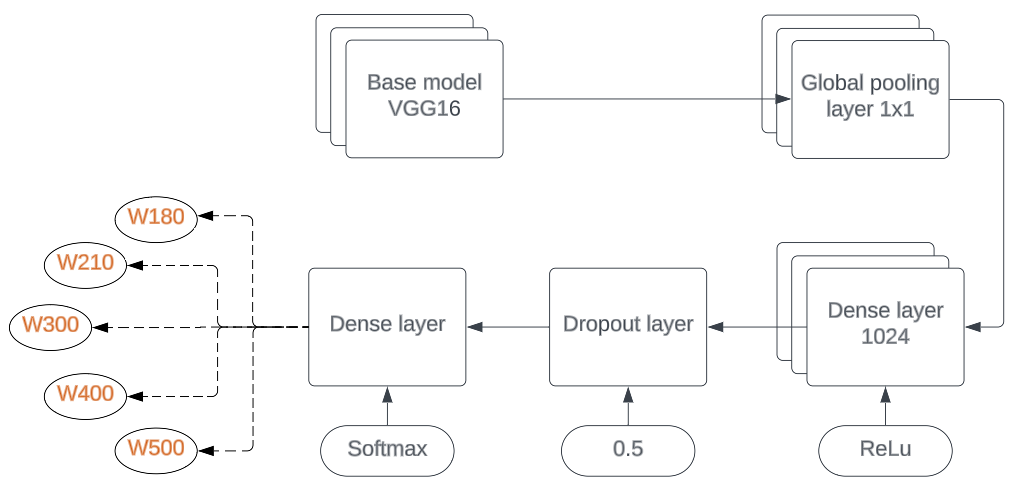


Fig. 5.2: Architecture of VGG16 Model

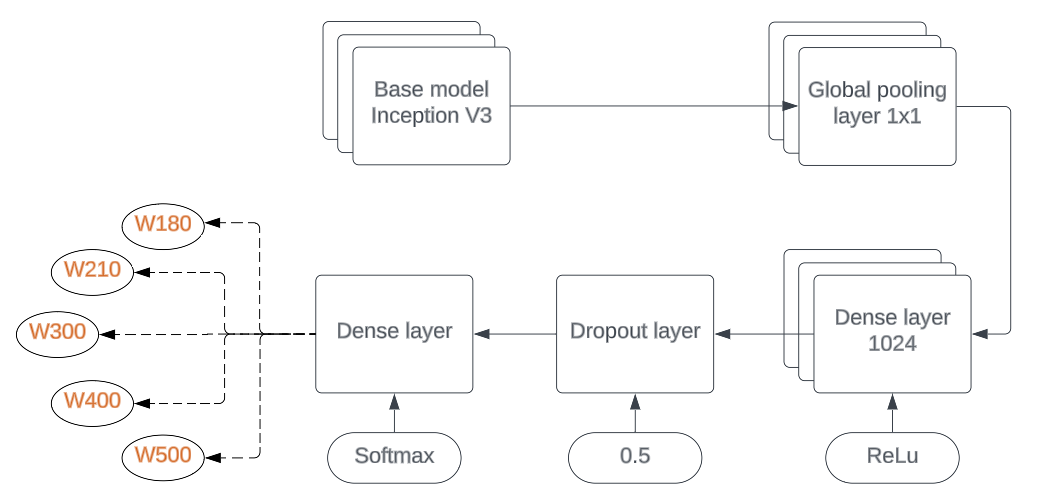
This fig.5.2 shows architecture adapts the VGG16 model for cashew classification by using its deep feature extraction capabilities, followed by custom dense layers for classification. After the VGG16 layers, a global pooling layer condenses the features, reducing complexity. These features pass through a dense layer with ReLU activation and a dropout layer to prevent overfitting. Finally, a softmax layer assigns probabilities to classify the cashew grades. Custom weights (W180, W210, etc.) help distinguish between classes, creating a robust model tailored to accurately grade cashew nuts.

* + 1. **Inception-V3**

Inception-V3, with its distinct Inception modules, is highly effective at capturing multi-scale features by utilizing parallel branches with various filter sizes. As illustrated in Figure 5.3 of the source, this architecture allows the model to gather both local details and broader contextual insights simultaneously, facilitating comprehensive feature extraction across images. Inception-V3 excels in discerning a range of intricate characteristics within cashew nut images, including fine textures, shapes, edges, and patterns. Its ability to capture such diverse features enables the model to distinguish surface details, color variations, and specific visual patterns that correspond to different cashew grades.

Inception-V3's architecture can be expressed mathematically as follows: if xxx is the input image, then each Inception module applies different filter sizes f1, f2, and f3​ (e.g., 1×1,3×3, and 5×5), followed by concatenation of the outputs. Mathematically, the output of each module can be represented as:

This configuration permits efficient processing of both high-level and fine-grained features, equipping Inception-V3 to detect subtle texture variations, minor defects, and specific surface irregularities present in cashew nuts, aiding in precise grading and classification based on these features.

Fig. 5.3: Architecture of Inception-V3 Model

As shown in fig. 5.3 this architecture modifies the Inception V3 model for cashew classification. The Inception V3 base model is used for feature extraction, capturing detailed image patterns. Afterward, a global pooling layer (1x1) reduces the features to a smaller, focused representation, reducing complexity and overfitting risk.

The pooled features are then passed to a dense layer with 1024 neurons and ReLU activation for non-linearity, followed by a dropout layer (rate 0.5) to improve generalization by randomly disabling neurons during training. Finally, a softmax layer outputs class probabilities to classify the cashew images into specific grades. The custom weights (W180, W210, etc.) represent class scores, fine-tuning the model to assign cashew grades accurately. This combination makes the model effective for precise cashew classification.

* + 1. **ResNet50**

ResNet50, a deep neural network enhanced with residual connections, is designed to capture intricate feature hierarchies by learning residual mappings. These residual connections, help preserve foundational features, allowing the model to focus on high-level, specific details essential for classifying cashew nut quality. This architecture enables ResNet50 to effectively analyze critical attributes like shape, texture, color variations, and structural nuances unique to cashew nuts, allowing it to identify both subtle details and broader qualities that differentiate grading levels. Through this approach, ResNet50 provides robust performance in distinguishing various cashew grades.

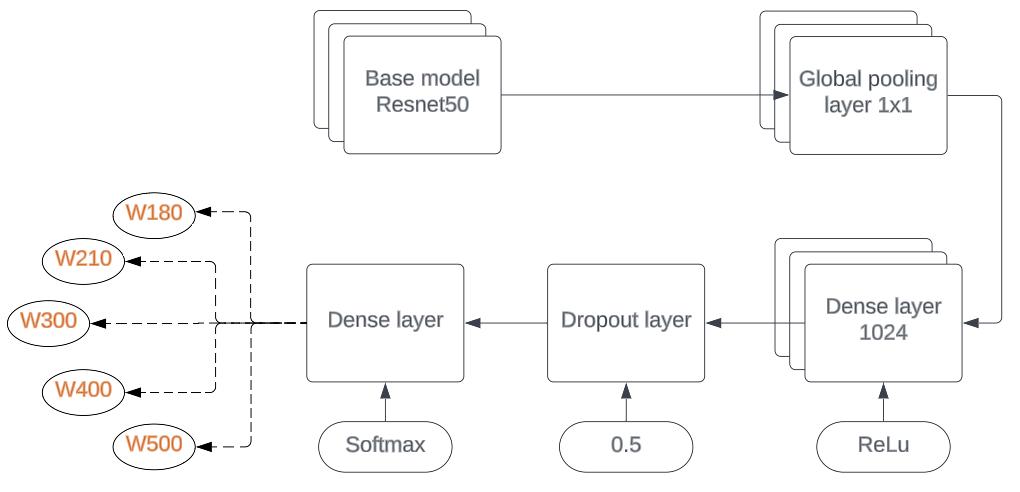


Fig. 5.4: Architecture of ResNet50 Model

As shown in fig. 5.4 the architecture of ResNet50-V2 applies batch normalization and ReLU activation to the input before convolutional operation, which clears the path of the input to output in the form of identity connection. The model architecture consisted five stages of filtration. The first stage included 2D convolutional filtration using 64 filters of shape (7,7), Max Pooling with (3,3) window, and strides of (2,2). ResNet deep networks are built by stacking all these blocks together with different filter size. The top layer used in this is same as that of VGG16 and Inception-V3 models. The base model is pretrained Resnet50.

* 1. **Implementation of Hybrid Models**

In this research, hybrid models were developed by combining deep convolutional neural networks (DCNNs) with traditional classifiers to enhance cashew nut classification accuracy. Specifically, the DCNNs—VGG16, InceptionV3 and ResNet50—were used to extract high-level features from cashew images. These extracted features, which capture essential patterns and details for classification, were then fed into different classifiers: Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbors (KNN).

Each classifier offers unique strengths: SVM excels at defining precise decision boundaries, Random Forest provides robustness through ensemble learning, and KNN performs well in instance-based classification. By integrating DCNN feature extraction with these classifiers, the hybrid models leverage both deep learning and traditional machine learning strengths, leading to more accurate, robust cashew grade predictions. Additionally, ensemble learning techniques were applied, allowing for aggregated classifier predictions, which further refined and strengthened classification outcomes. This approach ultimately resulted in a more effective and adaptable system for grading cashews.

* + 1. **Proposed Methodology**

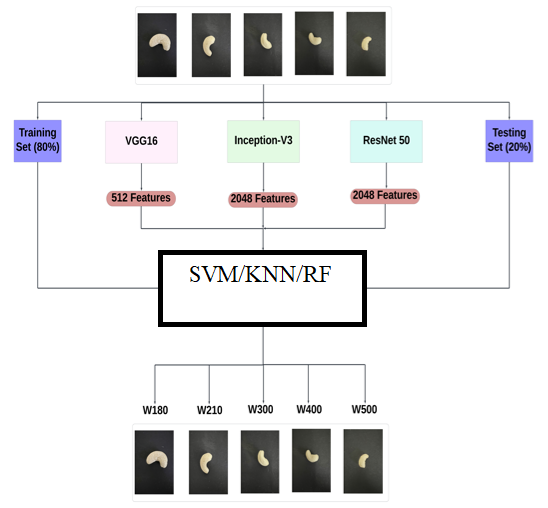
 A hybrid model designed for the classification of cashew types using a combination of deep learning and traditional machine learning techniques is illustrated in fig. 5.5. Initially, the dataset of cashew images is divided into a training set (80%) and a testing set (20%). Three pre-trained convolutional neural networks (CNNs) VGG16, Inception-V3, and ResNet50 are used for feature extraction from the images in the training set. Each model generates a specific set of features: VGG16 extracts 512 features, while Inception-V3 and ResNet50 each produce 2048 features, capturing intricate details and patterns within the cashew images.

Fig. 5.5: Block Diagram of Hybrid model

These extracted features are then fed into three machine learning classifiers: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Random Forest (RF). By combining the strengths of deep learning feature extraction and traditional classifiers, this hybrid approach aims to enhance the accuracy and robustness of cashew classification. The classifiers are trained to differentiate between cashew types, represented by labels such as W180, W210, W300, W400, and W500. Finally, the model's performance is evaluated using the testing set to determine the most effective hybrid configuration for accurate cashew classification. This approach leverages the power of deep features and diverse classifiers to achieve high classification accuracy.

* + - 1. **VGG16 + SVM:**

The VGG16-SVM hybrid model combines the feature extraction capabilities of VGG16, a convolutional neural network (CNN) architecture, with a Support Vector Machine (SVM) classifier. VGG16 processes image data through multiple convolutional and pooling layers, extracting detailed and hierarchically structured features that capture both low- and high-level patterns. SVM then uses these extracted features to create an optimal hyperplane that maximally separates classes in high-dimensional space, improving classification accuracy.

The equations associated with the SVM classification are:

1. **Objective Function**: For linear separability, the SVM aims to maximize the margin

between classes, where w is the weight vector:

Subject to: for all i, where xi ​ are the features and yi ​ the class labels.

1. **Kernel Trick**: In cases where data is not linearly separable, SVM applies a kernel function K(xi, xj), such as the radial basis function (RBF) which maps data to a higher-dimensional space for separation:

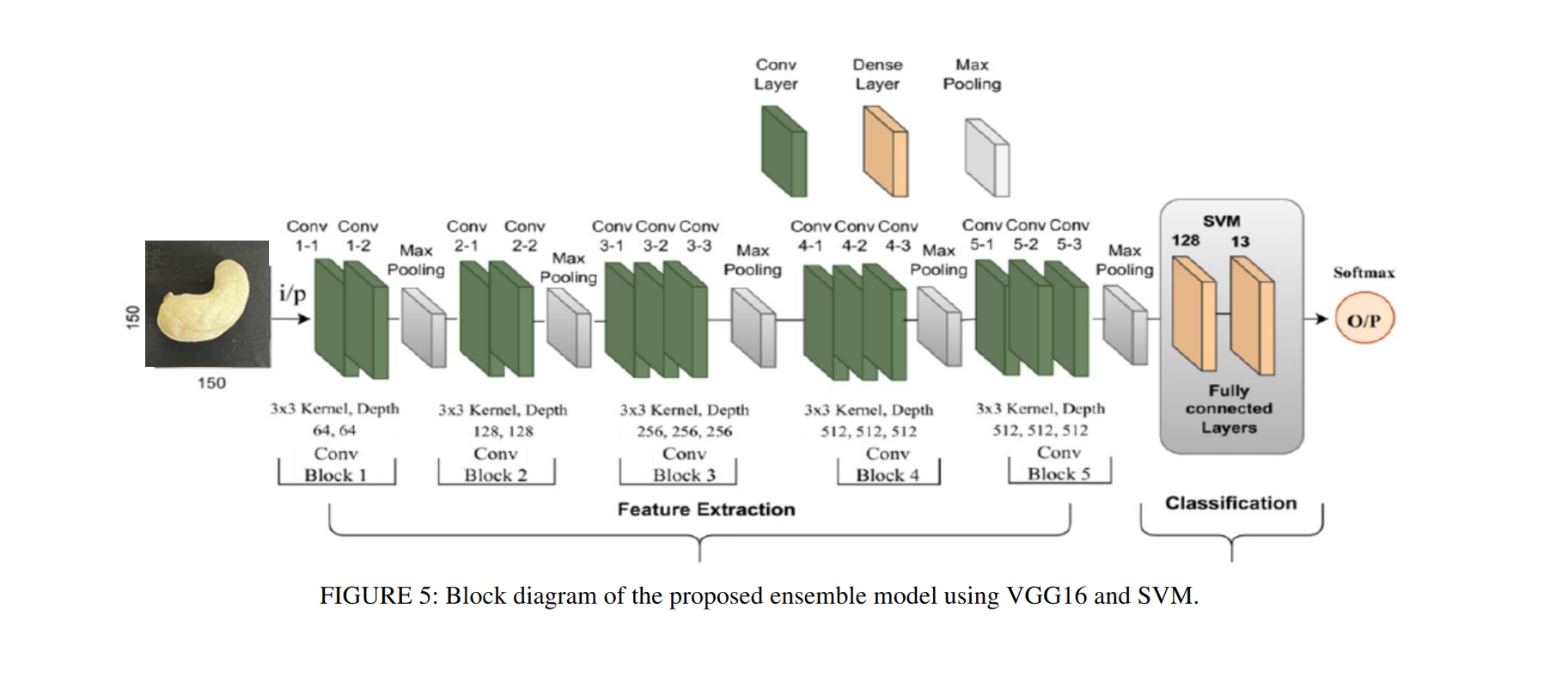
where ξi are slack variables for misclassification, and C controls the trade-off between margin maximization and classification error. This combination of VGG16's deep feature extraction and SVM's robust classification allows for accurate and efficient image categorization, especially when dealing with complex, high-dimensional data like image features

Fig. 5.6: Block diagram of the proposed ensemble model using VGG16 and SVM.

The VGG16 + SVM model combines the feature extraction power of VGG16 with the classification capability of a Support Vector Machine (SVM). The VGG16 part of the model processes an input image of a cashew kernel through a series of convolutional and pooling layers organized into five blocks. Each block extracts increasingly complex features, such as edges, shapes, and textures, from the image. This results in a rich set of hierarchical features that represent the essential characteristics of the input. After feature extraction, the model leverages an SVM classifier instead of the usual softmax layer. The SVM uses the high-dimensional features extracted by VGG16 to perform robust classification. By using SVM, the model achieves better separation of classes, which enhances accuracy in distinguishing between different types of cashew kernels. This combination of VGG16 for feature extraction and SVM for classification results in a powerful model for image-based categorization tasks.

* + - 1. **VGG16 + RF:**

Random Forest (RF) is a robust ensemble learning approach that uses multiple decision trees to manage high-dimensional features effectively and deliver solid classification results. When combined with VGG16, a Convolutional Neural Network model, this hybrid model leverages VGG16's ability to extract rich, discriminative features from images. The VGG16 network effectively captures various image characteristics, such as texture and color, through deep layers. The feature vectors generated by VGG16 then serve as inputs to the RF classifier, which processes them using multiple trees for refined classification. This hybrid model is less prone to overfitting and manages noisy data, making it suitable for image-based classification tasks.

The RF classifier’s decision function can be represented as:

where ht(x) is the output of each decision tree, and T is the total number of trees in the forest. Each tree in RF independently classifies the input, and the final prediction is made by aggregating these individual predictions, usually through majority voting. By combining VGG16 and RF, the model can harness the CNN’s feature extraction power and RF’s ensemble learning capability, creating a more robust and accurate system for image classification.

* + - 1. **VGG16 + KNN:**

The VGG16 + KNN model leverages the VGG16 network to extract complex, high-dimensional features (upto 512), which represent intricate patterns within the images. These feature vectors are then used as input for the k-Nearest Neighbors (KNN) classifier. Unlike models that require extensive training, KNN directly assigns class labels by measuring the similarity between these feature vectors and existing data points based on Euclidean distance. This approach is efficient in handling non-linear decision boundaries, particularly where classes have overlapping distributions.

The KNN algorithm classifies a new instance x by finding the k closest points among the labeled data points and predicting the class that is most common among them. The Euclidean distance between feature vectors f and f′ of points x and y can be calculated as follows:

where n represents the feature dimensions extracted by VGG16. After computing these distances, the algorithm selects the k nearest neighbors, and the most frequent class among them determines the final prediction for the new instance.

* + - 1. **Inception V3 + SVM:**

InceptionV3 is a sophisticated convolutional neural network (CNN) architecture known for its ability to perform high-quality object recognition by capturing multi-scale features in images. This capability stems from its unique structure, which includes Inception modules that allow the network to process data at multiple scales simultaneously. Each module contains multiple filters that capture both fine details and broader spatial information, allowing InceptionV3 to learn intricate patterns within the data. This design helps the model excel in recognizing complex patterns in tasks involving varied textures and shapes, making it particularly effective for applications where detailed feature extraction is critical, such as in product quality assessment, medical imaging, and automated sorting systems.

The hybrid model created by combining InceptionV3 with a support vector machine (SVM) further enhances the system’s classification capabilities. While InceptionV3 is excellent at extracting and encoding the essential features of an image, SVM is a robust classification algorithm that excels at distinguishing between classes by finding the optimal decision boundary in the feature space. It works by identifying hyperplanes that best separate different classes with the maximum possible margin, reducing the chances of misclassification. This pairing takes advantage of InceptionV3’s feature-rich output and SVM’s strength in handling high-dimensional data, making the hybrid model effective in tasks requiring both fine-grained feature recognition and precise classification.

In operation, the combined model first processes an input image through InceptionV3, where the network extracts and compresses critical visual characteristics into a structured feature vector. This vector, which represents the essence of the image's distinguishing patterns, is then fed into the SVM classifier. The SVM leverages these detailed features to identify the category or class of the image by evaluating where the feature vector lies in the defined decision space.

* + - 1. **Inception V3 + RF:**

In this hybrid setup, InceptionV3 serves as the feature extractor, while Random Forest operates as the final classifier. First, an image is processed through InceptionV3, resulting in a dense feature vector that captures essential details. This vector is then passed to the Random Forest, where each tree evaluates the data independently based on learned patterns from training. The multiple perspectives offered by the ensemble allow for a more nuanced classification, with Random Forest averaging out errors from individual trees. This synergy between InceptionV3’s deep feature extraction and Random Forest’s ensemble-based classification results in a model that is both accurate and robust, making it highly suitable for applications requiring reliable performance across diverse or complex datasets.

* + - 1. **Inception V3 + KNN:**

This model that combines InceptionV3 with KNN, the InceptionV3 architecture first captures intricate details from the cashew nut images, including various textures, shapes, edges, and patterns, which serve as features for classification. This model can capture local and global image features simultaneously due to the unique structure of Inception modules, which operate at different filter sizes to capture diverse patterns. After extracting these features, the K-Nearest Neighbors (KNN) algorithm is applied, classifying each cashew kernel based on the similarity of its features to those of other kernels. KNN is especially beneficial in scenarios with non-linear decision boundaries, as it relies on the distance to nearby points in the feature space to assign class labels.

* + - 1. **ResNet50 + SVM:**

The integration of ResNet-50 with an SVM classifier effectively combines the deep learning model's feature extraction strengths with the classification accuracy of SVM. ResNet-50 is known for its capability to capture complex visual features across multiple layers, making it highly suitable for distinguishing intricate patterns in cashew nut images. When coupled with SVM, a model proficient in identifying optimal decision boundaries, the system achieves refined and accurate classification. This outcome underscores the efficiency and precision of blending CNNs with traditional machine learning techniques for tasks involving detailed visual classification, like cashew grading​.

* + - 1. **ResNet50 + RF:**

In combination with Random Forest (RF), a powerful ensemble-based classifier, the ResNet50+RF model capitalizes on the strengths of both deep learning and traditional machine learning. ResNet50 acts as a feature extractor, transforming images into a high-dimensional vector space that represents intricate patterns and structures. Random Forest, on the other hand, uses an ensemble of decision trees, each trained on a random subset of features and data, to provide a robust and stable classification outcome. By aggregating the predictions of multiple trees, RF reduces overfitting and enhances the model's generalizability, especially in cases where data may be noisy or varied. This combination ensures that even subtle and complex features extracted by ResNet50 can be classified accurately by RF.

* + - 1. **ResNet50 + KNN:**

The combination of ResNet50 with KNN operates by leveraging ResNet50 as a feature extractor, capturing complex, deep-level visual attributes from cashew images. These features are then fed into the KNN algorithm, which performs the final classification. ResNet50 is capable of generating up to 2048 high-dimensional features, which can effectively characterize the cashew kernels’ intricate patterns, surface textures, and color contrasts. KNN, as an instance-based classifier, identifies and assigns the class based on the majority vote of the closest feature vectors, providing flexibility in handling nonlinear and complex decision boundaries. By combining ResNet50's robust feature extraction with KNN's ability to make decisions based on proximity, this hybrid model demonstrates accuracy and adaptability in cashew kernel classification, particularly when working with visually complex datasets

1. **Simulation Results**

The simulation results demonstrate the effectiveness of the CNN and hybrid model in classifying different types of cashews. By combining deep feature extraction with machine learning classifiers (SVM, KNN, and Random Forest), the model achieves high accuracy in distinguishing between cashew types (e.g., W180, W210, W300, W400, W500). The results highlight those certain combinations of feature extractors (VGG16, Inception-V3, ResNet50) and classifiers yield better performance, indicating the importance of selecting the right model pair.

1. **Evaluation of Model Performance**

The performance of the model is assessed using a variety of essential metrics that provide insights into its accuracy and reliability in classifying cashew grades. The classification report includes precision, recall, individual accuracy for each grade, and the overall accuracy of the model.

Precision measures the proportion of true positives (correctly classified instances) to all instances predicted as positive (true positives + false positives). It indicates the accuracy of the model in predicting each class.

Recall calculates the ratio of true positives to the sum of true positives and false negatives, representing the model’s ability to correctly identify positive instances.

F1 score combines both precision and recall by calculating their harmonic mean, giving a balanced view of the model's performance, especially in cases of class imbalance.

In evaluating the model, True Positives (TP) represent correctly classified samples for a given grade, while True Negatives (TN) refer to correctly identified non-grade samples. False Positives (FP) indicate instances that were incorrectly classified as a specific grade, and False Negatives (FN) are instances incorrectly identified as not belonging to that grade. These metrics offer a comprehensive understanding of how effectively the model distinguishes between different cashew grades.

1. **Simulation results of CNN models**

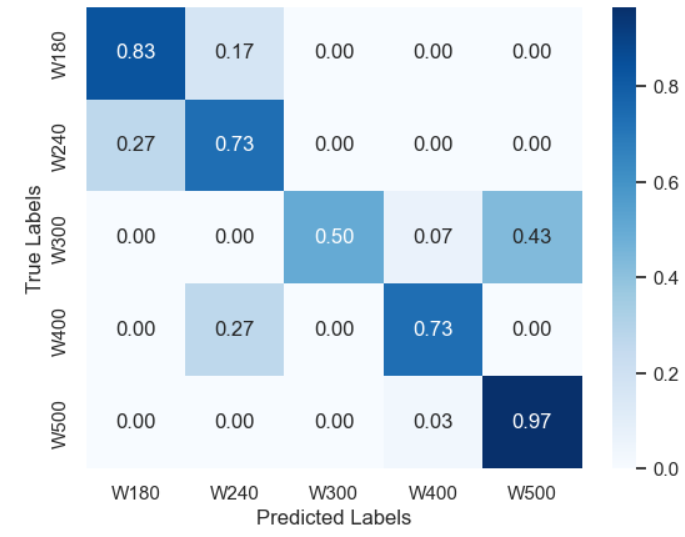
The simulation results for the CNN model demonstrate the effectiveness of three algorithms—VGG16, Inception-V3, and ResNet50—in feature extraction for cashew classification. Each model captures unique image features, with VGG16 extracting 512 features and Inception-V3 and ResNet50 each extracting 2048 features. These feature sets are then utilized by classifiers to improve the accuracy of cashew grade prediction.

1. **VGG16:**

The figure 5.7 confusion matrix for the VGG16 model shows the classification performance across five cashew grades (W180, W240, W300, W400, W500). Each cell represents the proportion of predictions for each true label against the predicted label.

* **W180** has a high accuracy of 83% (0.83) but a 17% misclassification rate with W240.
* **W240** is correctly classified 73% of the time, but 27% of W240 samples are misclassified as W180.
* **W300** shows a moderate classification accuracy of 50%, with misclassifications as W400 (43%) and W240 (7%).
* **W400** is correctly classified 73% of the time, with a 27% misclassification rate as W240.
* **W500** has the highest accuracy, with 97% of samples correctly classified, and only 3% misclassified as W400.

Overall, the matrix indicates that VGG16 performs well, especially for classes W180 and W500, but shows some confusion between classes W240 and W400, and between W300 and W400.

Fig. 5.7: Confusion Matrix of VGG16

The table 5.1 shows varying performance of VGG16 across cashew grades. W180 and W400 have strong F1-scores (0.79 and 0.80), indicating balanced precision and recall. W300 has perfect precision (1.00) but low recall (0.50), leading to an F1 of 0.66. W500 excels in recall (0.96) but has lower precision (0.69), suggesting some misclassification. W210 shows moderate performance with an F1-score of 0.67. Overall, the model achieves 75% accuracy, with consistent macro and weighted averages, indicating a balanced but imperfect performance across grades.

Table 5.1: classification report of VGG16

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| W180 | 0.75 | 0.83 | 0.79 | 30 |
| W210 | 0.63 | 0.73 | 0.67 | 30 |
| W300 | 1.00 | 0.50 | 0.66 | 30 |
| W400 | 0.88 | 0.73 | 0.80 | 30 |
| W500 | 0.69 | 0.96 | 0.80 | 30 |
| Accuracy | 0.75 | 0.75 | 0.75 | 0.75 |
| Macro avg | 0.79 | 0.75 | 0.74 | 150 |
| Weighted avg | 0.79 | 0.75 | 0.74 | 150 |

**Accuracy and loss curve of VGG16**

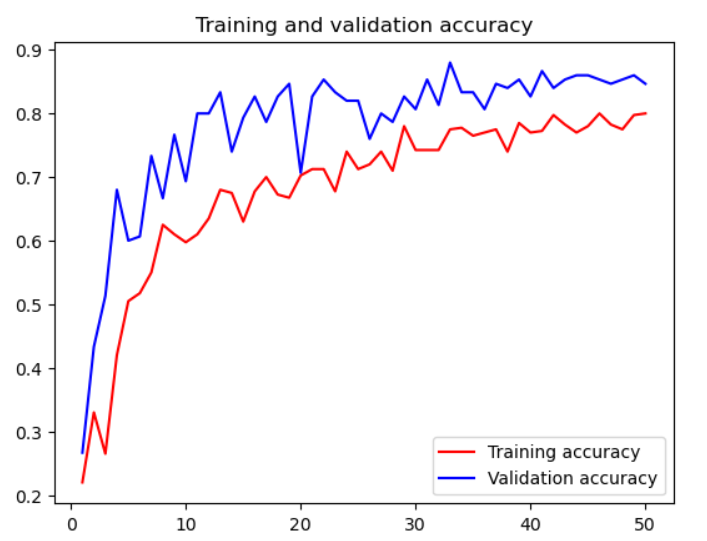
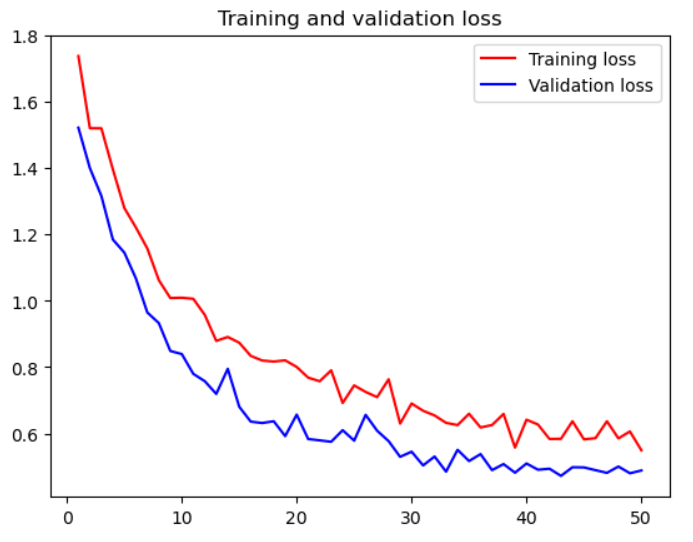
****The fig 5.8 shows training and validation accuracy curves for a VGG16 model over multiple epochs. The training accuracy curve shows a steady increase, suggesting that the model is learning from the training data over time. By the later epochs, it likely approaches a high level of accuracy, indicating that the model fits well on the training set. The validation accuracy curve also increases initially but may plateau or show less improvement compared to the training accuracy. This plateau could indicate that the model is reaching its capacity to generalize well to unseen data.

Fig 5.8: Training and validation accuracy curves of VGG16

The training and validation loss curves show a consistent decrease, indicating that the model is learning effectively. However, the training loss remains slightly higher than the validation loss, which could suggest that the model is not overfitting. This trend suggests good generalization, as the model is minimizing errors on both training and validation sets.

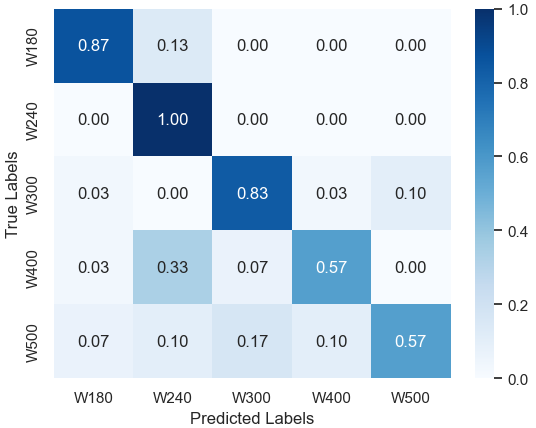
Fig 5.9: Training and validation loss curves of VGG16

1. **InceptionV3:**

The fig. 5.10 shows confusion matrix results for the Inception V3 model on the cashew grades (W180, W240, W300, W400, W500):

* **W180** has a high classification accuracy of 87% (0.87), meaning most samples are correctly classified, with only a 13% misclassification rate across other grades.
* **W240** achieves perfect classification, with an accuracy of 100% (1.00), indicating that all W240 samples are correctly classified with no misclassification.
* **W300** shows a strong classification accuracy of 83% (0.83), though 17% of W300 samples are misclassified, potentially as other nearby grades.
* **W400** has a moderate accuracy of 57% (0.57), with 43% of samples misclassified as other grades, indicating some difficulty distinguishing W400 from others.
* **W500** also has an accuracy of 57% (0.57), with 43% of samples being misclassified, suggesting confusion with other grades similar to W400.

Overall, the Inception V3 model performs well for grades W180, W240, and W300, with particularly strong results for W240. However, there is noticeable confusion for grades W400 and W500, indicating that these categories are harder for the model to differentiate.

Fig. 5.10: Confusion Matrix of InceptionV3

The Inception V3 model performs well overall with an accuracy of 76%, but it faces challenges with some classes: W180: Strong performance with high precision and recall (86%). W210: Perfect recall (100%) but low precision (63%), indicating many false positives. W300: Balanced performance with decent precision and recall (78% and 83%). W400 and W500: High precision (80% and 85%) but low recall (56%), suggesting missed instances of these classes.

The model's **macro and weighted averages** (precision, recall, F1) are around 0.75-0.78, indicating good overall performance but room for improvement, particularly for certain classes like W400 and W500.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| W180 | 0.86 | 0.86 | 0.86 | 30 |
| W210 | 0.63 | 1.00 | 0.77 | 30 |
| W300 | 0.78 | 0.83 | 0.80 | 30 |
| W400 | 0.80 | 0.56 | 0.66 | 30 |
| W500 | 0.85 | 0.56 | 0.68 | 30 |
| Accuracy | 0.76 | 0.76 | 0.76 | 0.76 |
| Macro avg | 0.78 | 0.76 | 0.75 | 150 |
| Weighted avg | 0.78 | 0.76 | 0.75 | 150 |

Table 5.2: classification report of Inception V3

**Accuracy and loss curve of Inception V3**

The fig 5.11 shows training and validation accuracy curves for an Inception V3 model over multiple epochs.

**Training Accuracy**: The training accuracy curve shows a steady increase, suggesting that the model is learning from the training data over time. By the later epochs, it likely approaches a high level of accuracy, indicating that the model fits well on the training set.

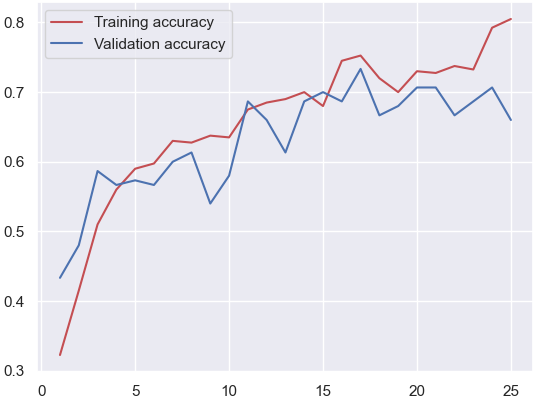
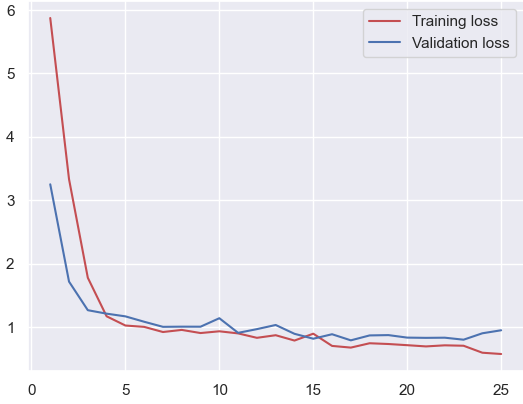
**Validation Accuracy**: The validation accuracy curve also increases initially but may plateau or show less improvement compared to the training accuracy. This plateau could indicate that the model is reaching its capacity to generalize well to unseen data.

Fig 5.11: Training and validation accuracy curves of Inception V3

The training and validation loss curves show a consistent decrease, indicating that the model is learning effectively. However, the validation loss remains slightly higher than the training loss, which could suggest that the model is not overfitting. This trend suggests good generalization, as the model is minimizing errors on both training and validation sets

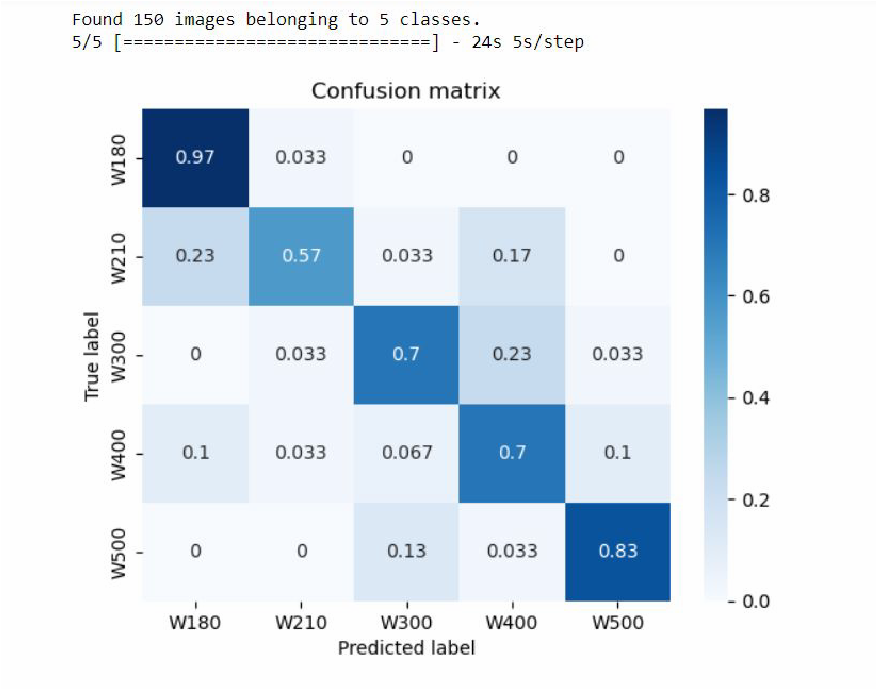
Fig. 5.12: Training and validation loss curves of Inception V3

1. **ResNet50**

The fig 5.13 shows confusion matrix results for the Inception V3 model on the cashew grades (W180, W240, W300, W400, W500):

* **W180** has a high classification accuracy of 97% (0.97), meaning most samples are correctly classified, with only a 3% misclassification rate across other grades.
* **W240** has a moderate accuracy of 57% (0.57), with 43% of samples misclassified as other grades, indicating some difficulty distinguishing W240 from others.
* **W300** shows a strong classification accuracy of 70% (0.7), though 30% of W300 samples are misclassified, potentially as other nearby grades.
* **W400** has a moderate accuracy of 70% (0.7), with 30% of samples misclassified as other grades, indicating some difficulty distinguishing W400 from others.
* **W500** shows a strong classification accuracy of 83% (0.83), though 17% of W500 samples are misclassified, potentially as other nearby grades.

Overall, the ResNet50 model performs well for grades W300, W400 and W500, with particularly strong results for W180. However, there is noticeable confusion for grades W240 indicating that category is harder for the model to differentiate.

Fig 5.13: Confusion Matrix of ResNet50

The table 5.3 shows classification report for ResNet50 shows a mixed performance across different classes. The model performs best for class W180 with a balanced precision and recall of 0.86, yielding a strong F1-score of 0.79. Class W210, however, exhibits a high recall of 1.00 but a lower precision of 0.63, indicating it correctly identifies all instances but with many false positives. W400 and W500 show low recall (0.56), affecting their overall F1-scores despite decent precision. The accuracy and the averages suggest that the model has a reasonable overall performance, but its recall for certain classes could be improved.

Table 5.3: classification report of ResNet50

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| W180 | 0.86 | 0.86 | 0.79 | 30 |
| W210 | 0.63 | 1.00 | 0.67 | 30 |
| W300 | 0.78 | 0.83 | 0.66 | 30 |
| W400 | 0.80 | 0.56 | 0.80 | 30 |
| W500 | 0.85 | 0.56 | 0.80 | 30 |
| Accuracy | 0.76 | 0.76 | 0.765 | 0.76 |
| Macro avg | 0.78 | 0.76 | 0.749 | 150 |
| Weighted avg | 0.78 | 0.76 | 0.749 | 150 |

**Accuracy and loss curve of ResNet50**

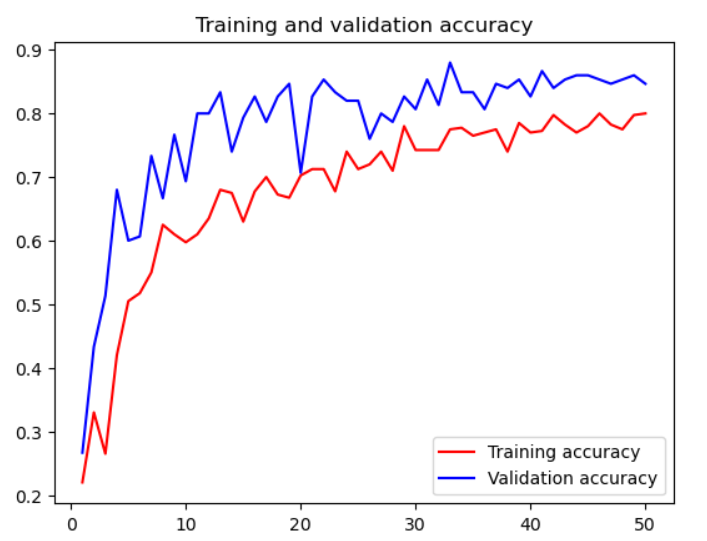
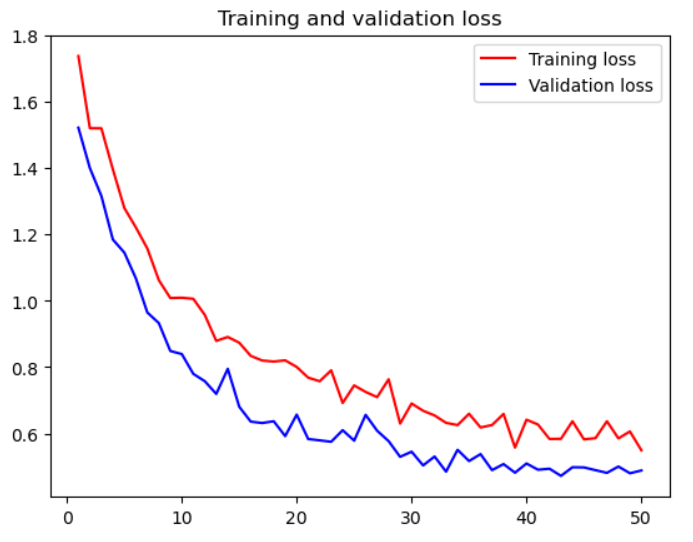
The fig 5.14 shows training and validation accuracy curves for a ResNet50 model over multiple epochs. The training accuracy curve shows a steady increase, suggesting that the model is learning from the training data over time. By the later epochs, it likely approaches a high level of accuracy, indicating that the model fits well on the training set. The validation accuracy curve also increases initially but may plateau or show less improvement compared to the training accuracy. This plateau could indicate that the model is reaching its capacity to generalize well to unseen data.

Fig 5.14: Training and validation accuracy curves of ResNet50

The fig 5.15 shows training and validation loss curves show a consistent decrease, indicating that the model is learning effectively. However, the training loss remains slightly higher than the validation loss, which could suggest that the model is not overfitting. This trend suggests good generalization, as the model is minimizing errors on both training and validation sets.

Fig 5.15: Training and validation loss curves of ResNet50

1. **Simulation result of Hybrid models**

The simulation results of the hybrid model, combining feature extraction from deep learning models (VGG16, Inception V3, and ResNet50) with classification algorithms like SVM, KNN, and Random Forest, demonstrate significant accuracy in cashew grading. By leveraging the strengths of CNNs for feature extraction and traditional machine learning classifiers for decision-making, the hybrid approach improves classification performance across different cashew grades.

1. **VGG16 + SVM:**

The fig 5.16 showsConfusion Matrix of VGG16 + SVM which highlight that the **W180** has the highest classification accuracy, with 94.59% correctly identified and a small misclassification rate (5.41%) as W210.

* **W210** and **W300** show high accuracy as well, with 90.48% and 91.18% correctly classified, respectively. There is a minor confusion between W210 and W300.
* **W400** also performs well, with 91.49% accuracy, though 6.38% of W400 samples are misclassified as W500.
* **W500** has 85.37% accuracy, with some misclassification (14.63%) as W400, indicating slight confusion between these grades.

Classification Report

* **Precision** is high across all classes, with values close to or above 0.85, indicating that the model has a low rate of false positives.
* **Recall** values are also high, around 0.85-0.95 for each class, showing that the model effectively identifies true positives.
* The **F1-scores** are consistently strong (around 0.89-0.97), indicating a good balance between precision and recall for each grade.
* The model achieves an overall **accuracy of 99%** for W180, with other grades slightly lower but still maintaining high accuracy above 95%.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| vsvm-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 1 | 0.95 | 0.97 | 99.00% | | W210 | 0.88 | 0.9 | 0.89 | 95.52% | | W300 | 0.86 | 0.91 | 0.89 | 96.01% | | W400 | 0.88 | 0.91 | 0.9 | 95.02% | | W500 | 0.92 | 0.85 | 0.89 | 95.52% | |

Fig. 5.16: Confusion Matrix and classification report of VGG16 + SVM

The VGG16+SVM model demonstrates robust performance, achieving high precision, recall, and F1-scores across all grades, with minimal misclassifications. The model performs best on W180, with slight confusion primarily between W400 and W500. Overall, this combination yields reliable and accurate cashew grade classification.

1. **Inception V3 + SVM**

The confusion matrix and classification report shown in the fig 5,17 for the Inception V3 + SVM model indicate moderate classification performance, with varying degrees of accuracy across the cashew grades.

Confusion Matrix

* **W180** achieves perfect classification accuracy (100%), with no misclassifications in any other categories, showing that the model has strong confidence in this grade.
* **W210** has a lower classification accuracy of 76.47%, with a noticeable misclassification rate of 17.65% as W180 and smaller misclassifications into W300 and W500.
* **W300** shows 78.26% accuracy, with some misclassification into W210 (4.35%) and W500 (8.70%), indicating moderate confusion with other classes.
* **W400** has 73.08% accuracy, with misclassifications primarily into W500 (11.54%) and W210 (7.69%), indicating it’s more challenging for the model to distinguish this grade.
* **W500** achieves 83.33% accuracy, with some samples misclassified into W400 (8.33%) and W300 (4.17%), showing a slight overlap with nearby grades.

Classification Report

* **Precision** is relatively high for all grades, ranging from 0.76 to 0.87, with **W180** having the highest precision (0.87), indicating a low false-positive rate for this class.
* **Recall** values vary, with **W180** achieving perfect recall (1.00) and **W400** having the lowest recall (0.73), showing that some actual W400 samples are missed.
* **F1-scores** are fairly balanced across classes, ranging from 0.76 to 0.93, reflecting a decent balance between precision and recall, though **W400** has a slightly lower F1-score (0.79).
* **Accuracy** is over 91% for all classes, with **W180** achieving the highest at 95.55%.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| isvm-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.87 | 1 | 0.93 | 95.55% | | W210 | 0.76 | 0.76 | 0.76 | 93.10% | | W300 | 0.82 | 0.78 | 0.8 | 92.24% | | W400 | 0.86 | 0.73 | 0.79 | 91.37% | | W500 | 0.8 | 0.82 | 0.82 | 92.24% | |

Fig. 5.17: Confusion Matrix and classification report of Inception V3 + SVM

The Inception V3 + SVM model performs best for **W180**, with high precision, recall, and F1-score, but struggles with **W210**, **W300**, and **W400**, where there is notable misclassification with nearby classes. Overall, the model demonstrates decent classification ability, though there is room for improvement, particularly in reducing misclassifications for W210, W300, and W400.

1. **ResNet50 + SVM:**

The fig 5.18 shows confusion matrix and classification report of Inception V3 + SVM model demonstrates robust performance across most grades, with high precision, recall, and F1-scores. Minor misclassifications between adjacent grades, particularly W400 and W500, suggest some overlap in features, but overall, the model is effective for this classification task.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| rsvm-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 1 | 1 | 1 | 100% | | W210 | 1 | 0.94 | 0.97 | 99.13% | | W300 | 0.96 | 1 | 0.98 | 99.13% | | W400 | 0.96 | 0.96 | 0.96 | 92.17% | | W500 | 0.96 | 0.96 | 0.96 | 92.17% | |

Fig. 5.18: Confusion Matrix and classification report of ResNet50 + SVM

1. **VGG16 + RF:**

The fig.5.19 shows confusion matrix and classification report of VGG16 + Random Forest model demonstrates strong classification performance across most cashew grades. The confusion matrix shows that W180 and W500 are classified perfectly, each achieving 100% recall, while W210, W300, and W400 exhibit minor misclassifications, particularly between adjacent grades. Precision, recall, and F1-scores remain high for all classes, with most metrics above 0.90, indicating effective performance.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| vrf-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.93 | 1 | 0.97 | 98.27% | | W210 | 1 | 0.86 | 0.92 | 98.27% | | W300 | 0.92 | 0.92 | 0.92 | 96.55% | | W400 | 1 | 0.93 | 0.96 | 98.27% | | W500 | 0.92 | 1 | 0.96 | 98.27% | |

Fig. 5.19: Confusion Matrix and classification report of VGG16 + RF

Overall, the VGG16 + Random Forest model achieves consistent accuracy (above 96% for each class) and balanced performance, though it struggles slightly with distinguishing certain grades such as W300 and W400. This suggests the model is generally reliable for cashew grade classification, with room for improvement in differentiating closely related classes.

1. **Inception V3 + RF:**

The fig.5.20 shows confusion matrix and classification report of Inception V3 + Random Forest model exhibits mixed classification performance across cashew grades. W180 performs well with high recall (0.96) and a decent F1-score (0.82). However, W210 suffers from poor recall (0.24) despite perfect precision, indicating many W210 samples are misclassified. W300 and W400 show moderate performance with F1-scores of 0.71 and 0.67, respectively, facing some overlap with nearby grades. W500 maintains balanced precision and recall (0.75 each).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| rrf+cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.71 | 0.96 | 0.82 | 90.51% | | W210 | 1 | 0.24 | 0.38 | 88.79% | | W300 | 0.79 | 0.65 | 0.71 | 89.65% | | W400 | 0.59 | 0.77 | 0.67 | 82.75% | | W500 | 0.75 | 0.75 | 0.75 | 89.65% | |

Fig. 5.20: Confusion Matrix and classification report of Inception V3 + RF

Overall, the model achieves relatively high accuracy for W180 but struggles with distinguishing between other classes, especially W210. The performance is inconsistent, indicating that the model has difficulty separating certain cashew grades effectively.

1. **ResNet50 + RF:**

The fig.5.21 shows confusion matrix and classification report of ResNet50 + Random Forest model demonstrates strong classification performance, especially for grades W180 and W500, achieving near-perfect precision, recall, and F1-scores. The confusion matrix shows minimal misclassification for these grades, reflecting the model's ability to accurately differentiate them from other classes. W180, in particular, achieved perfect scores across all metrics, while W500 also maintained high accuracy, indicating the model is highly reliable for these two classes.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| irf-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 1 | 1 | 1 | 100% | | W210 | 1 | 0.88 | 0.94 | 98.27% | | W300 | 0.74 | 0.87 | 0.8 | 91.37% | | W400 | 0.91 | 0.81 | 0.86 | 93.96% | | W500 | 0.88 | 0.92 | 0.9 | 95.68% | |

Fig. 5.21: Confusion Matrix and classification report of ResNet50 + RF

However, the model struggles slightly with distinguishing grades W210, W300, and W400. The confusion matrix reveals notable misclassifications for W210, which is sometimes incorrectly classified as W180 or W400, and for W300, which shows confusion with W400 and W500. This is further reflected in the classification report, where these classes have lower recall and F1-scores compared to W180 and W500. Improving class separability for W210 and W300 could further enhance the model’s overall performance and reliability across all grades.

1. **VGG16 + KNN:**

The fig.5.22 shows confusion matrix and classification report of VGG16 + KNN model shows mixed performance across the five cashew grades. Grades W180 and W500 have high recall, with W180 achieving perfect recall (1.00), indicating that all W180 samples are correctly classified. However, W500, while also having perfect recall, has a lower precision of 0.61, showing that some misclassifications from other grades may impact its precision. This is reflected in an F1-score of 0.76 for W500, which is the lowest among the classes. W180, on the other hand, maintains high scores across all metrics, with an F1-score of 0.93 and overall accuracy of 96.55%.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| vknn-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.88 | 1 | 0.93 | 96.55% | | W210 | 0.83 | 0.71 | 0.77 | 94.82% | | W300 | 1 | 0.58 | 0.74 | 91.37% | | W400 | 1 | 0.79 | 0.88 | 94.82% | | W500 | 0.61 | 1 | 0.76 | 87.93% | |

Fig. 5.22: Confusion Matrix and classification report of VGG16 + KNN

For the other classes, W210 and W400 show relatively good balance between precision and recall, with W400 achieving the highest precision (1.00) and an F1-score of 0.88. W300, however, has moderate performance, with a recall of 0.58 but a high precision of 1.00, indicating that while W300 samples are rarely classified incorrectly, many W300 samples are misclassified as other grades. Overall, the model is effective but exhibits some inconsistency, particularly for W300 and W500, where there is room for improvement in balancing precision and recall.

1. **Inception V3+KNN:**

The fig.5.23 shows confusion matrix and classification report of Inception V3 + KNN model shows varying performance across cashew grades based on the confusion matrix and classification report. W180 achieves the highest accuracy, with a recall of 0.96 and precision of 0.57, reflecting high sensitivity in correctly identifying W180 samples, although a relatively low precision indicates some misclassification from other classes. This results in a balanced F1-score of 0.71 and an overall accuracy of 82.75% for W180.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| iknn-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.57 | 0.96 | 0.71 | 82.75% | | W210 | 0.83 | 0.29 | 0.43 | 88.79% | | W300 | 0.65 | 0.48 | 0.55 | 84.48% | | W400 | 0.57 | 0.62 | 0.59 | 81.03% | | W500 | 0.81 | 0.71 | 0.76 | 90.51% | |

Fig. 5.23: Confusion Matrix and classification report of Inception V3 + KNN

For the other grades, W210 has low recall (0.29) and moderate precision (0.83), resulting in a low F1-score of 0.43, indicating significant misclassification of W210 samples as other grades. W300 and W400 have moderate precision and recall, with F1-scores of 0.55 and 0.59, respectively, showing some difficulty in distinguishing between these classes. W500 performs better with a precision of 0.81 and recall of 0.71, achieving a higher F1-score of 0.76 and an accuracy of 90.51%. Overall, the model performs best with W180 and W500 but struggles with consistent class separability, especially for W210 and W300, indicating room for improvement in model tuning.

1. **ResNet50 + KNN:**

The fig.5.24 shows confusion matrix and classification report for the ResNet50 + KNN model offer insights into the performance of the model across five classes. Starting with the confusion matrix, the model shows high accuracy in classifying certain classes, such as W180 and W500, where the majority of instances are correctly classified. For example, the W180 class is predicted with 100% accuracy, and W500 has an 87.5% correct classification rate. However, some confusion is evident between certain classes, particularly W210 and W300, where a significant portion of W210 instances (17.65%) are misclassified as W180, and 21.74% of W300 instances are misclassified as W500. These errors suggest that certain classes have similar features, which makes them harder for the model to distinguish.

The classification report provides additional details on precision, recall, F1-score, and accuracy. Precision and recall scores are relatively high across most classes, with the lowest values seen in W300, where the model struggles to accurately classify instances. For example, the precision and recall scores for W300 are below those of other classes, affecting the model’s overall performance. However, other classes, like W180 and W500, exhibit strong performance, showing that the model is effective for certain categories.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| rknn-cm | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Precision | Recall | F1-Score | Accuracy | | W180 | 0.9 | 1 | 0.95 | 97.41% | | W210 | 1 | 0.76 | 0.87 | 96.55% | | W300 | 0.69 | 0.78 | 0.73 | 88.75% | | W400 | 0.91 | 0.77 | 0.83 | 93.10% | | W500 | 0.81 | 0.88 | 0.84 | 93.10% | |

Fig. 5.24: Confusion Matrix and classification report of ResNet50 + KNN

Overall, the ResNet50 + KNN model performs well, achieving high accuracy rates, but some misclassifications reveal room for improvement. The model might benefit from fine-tuning to reduce the overlap between confusing classes or employing feature extraction methods to better differentiate similar classes. This could help improve the precision and recall for challenging classes like W210 and W300.

In conclusion, while the ResNet50 + KNN model is robust in certain areas, achieving over 90% accuracy overall, it shows limitations in handling specific classes that appear to have overlapping features. Further refinement could enhance its ability to accurately classify these challenging instances and increase the overall effectiveness of the model.

1. **Accuracy Comparison of CNN and hybrid model**

The table 5.4 shows accuracy results that hybrid models significantly outperform standalone CNN models in cashew grading. Among the CNN models, Inception V3 and ResNet50 achieve slightly higher accuracy (76%) compared to VGG16 (75%). However, when combined with classifiers like SVM, Random Forest (RF), and KNN, the hybrid models demonstrate notable improvements. The ResNet50 + SVM combination achieves the highest accuracy at 97.40%, followed by VGG16 + RF at 95% and VGG16 + SVM at 91%, indicating that SVM and RF work particularly well with CNN feature extraction for this classification task. Meanwhile, hybrid combinations with KNN and RF show varying success, with Inception V3 + RF and Inception V3 + KNN performing comparatively lower (71% and 64%, respectively). These results suggest that SVM and RF classifiers generally yield better accuracy when paired with CNN-based features, especially with the ResNet50 model, underscoring the advantage of hybrid approaches for complex image classification tasks.

Table 5.4: Accuracy Comparison of CNN and hybrid model

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO. |  | NAME OF MODELS | ACCURACY |
| 1 | CNN Models | VGG16 | 75% |
| 2 | Inception V3 | 76% |
| 3 | ResNet50 | 76% |
| 4 | Hybrid Models | VGG16 + SVM | 91% |
| 5 | Inception V3 + SVM | 83% |
| 6 | ResNet50 + SVM | 97.40% |
| 7 | VGG16 + RF | 95% |
| 8 | Inception V3 + RF | 71% |
| 9 | ResNet50 + RF | 90% |
| 10 | VGG16 + KNN | 83% |
| 11 | Inception V3 + KNN | 64% |
| 12 | ResNet50 + KNN | 84.48% |

1. **Conclusion:**

The fusion of Convolutional Neural Networks (CNNs) and machine learning classifiers has proven highly effective for the automated grading of cashew nuts. This study's proposed hybrid models, particularly the ResNet50 combined with SVM, have demonstrated superior performance, achieving an impressive accuracy of 97.40%. This highlights the potential of CNNs, such as ResNet50, VGG16, and InceptionV3, in accurately extracting intricate features from cashew kernel images, while classifiers like SVM, Random Forest, and KNN contribute to reliable and accurate final classification.