Name: Bhuwan Basnet
Student ID: 1830167

Task : Design a fruit classification system to identify the fruit name shown in the image.

Table of Contents

Abstract

This report provides a comprehensive analysis of a fruit classification system which helps the local farmer to sort their fruit labels on the boxes. The system utilizes the Convolutional Neural Network (CNN), a powerful deep learning technique to classify different types of fruits. This dataset consist of various images which are categorized into 14 distinct fruit classes. This data is further inspect, preprocess and validate to enhance the model reliability. We have observed the various fruits images and their dataset which is further evaluated to identify the fruits according to their pattern recognition. The CNN was carefully implemented, trained and evaluated using various metrices such as accuracy and confusion metrix to enhace the performance of the system. Since, CNN demonstrate the good accuracy compare to other models, our primary goal is to obtain the fruits name from fruit classification system and the CNN model gives accuracy 90.1%.

Introduction

In today's digital age, the image classification and recognition is crucial and rapidly increasing in various sector such as food industry, agriculture where large number of images can be analysed and processed. Because of its great ability to learn and recognize the patterns, this system is widely used to identify the different types of fruits based on their images that helps to improve quality control and inventory management by accurately identifying different types of fruits from image. This system reduce the time-consuming and prone errors that we faced in traditional methods. CNN are well-suited in this image classification and recognition system because of their ability to automatically learn and extract features from the raw image data. This methods have practical applications in areas such as automated grocery store, dietary tracking, and agriculture monitoring etc.

Problem Statement

This project goal is to develop and identify the name of fruits from the 14 distinct fruit classes using CNN model. The challenge is this project lies in creating a robust model that can generalize well to new, unseen data and ensuring high accuracy and reliabilty in practical applications.

Objectives

The objectives of this system is to help the farmer and to identify the fruits name so that it will aid farmer to stack similar products in the same pallet which helps to save time. This process can be done following these steps.

1. **Data Preparation :** Load, preprocess and augment the dataset so that it will be fit for training a CNN.
2. **Model Design :** Develop a CNN architecture that effectively captures the features which is necessary for accurate fruit classification.
3. **Training and Validation :** K-Fold Cross Validation is used to validate the model and prevent from overfitting.
4. **Evaluation:** The model performance is evaluated by metrices such as accuracy and confusion matrix that helps to improve accuracy in identifying different fruits.
5. **Visualization:** The model performance can be visualized by plotting learning curves and a confusion matrix.

Dataset Description

The dataset used in this project is classified into two folders: "train" and "test". Where the train folder contain 14 subfolder, one for each fruit category and same applies to test folder. The training data was further split into training and cross-validation sets using K-Fold Cross Validation. All the image is uniformly resized to 100*100 pixels and normalized to ensure consistency in the data and simplifies processing for the model.

Methodology

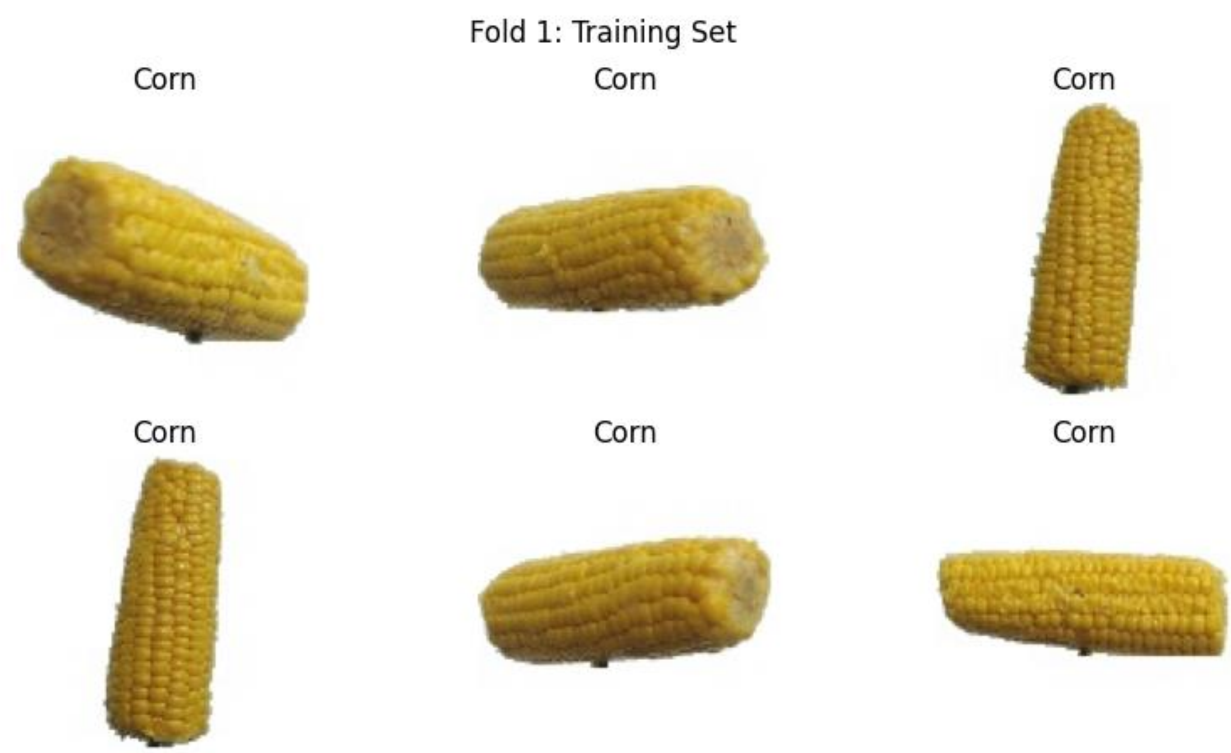
The methodology for fruit classification outlines the specific methods and techniques that will be used to identify various types of fruits from image. It involves several steps :

1. **Data Loading and Preprocessing**

All the raw data was organized into two main folder of ‘train’ and ‘test’, each contain subfolder having 14 fruit categories and that dataset is loaded using Python libraries such as ‘os’, and ‘PIL’. After that the fruits image is resized to a standard size of 100*100 pixels to maintain the consistency. Furthermore, the pixel value were also normalized in range of 0 to 1 so that it will be more stable and faster for training. Also, the labels were encoded using encoding techniques to transform the categorical labels into numerical format during data processing.

2. **K-Fold Cross Validation**

K-Fold Cross Validation is a technique used in machine learning to evaluate the performance and generalize the ability of the model. This technique helps to ensure the model’s performance and reliability by assessing on different subsets of the data. For eg, the dataset is divided into k subset or fold and model is trained and evaluated k times, each time using a different fold as a validation set and remaining fold as the training set. This process helps to minimize overfitting, hyperparameter tuning and ensure model performs well on different subsets of data.



3. **CNN Model Design**

This architecture is designed with different layers given below :

a. Convolutional Layers : These are the core building blocks of a CNN which contains a set of filters (or kernel) which is designed to extract the meaningful features from visual data such as images and videos. It relies on the working components such as input data, filters and feature maps. The three convolutional layers: convolution operation, activation functions and pooling, with increasing filter sizes (36, 64, 128) with ReLU activation functions helps to capture features at different levels of abstraction.

b. Pooling Layers : This layers also known as downsampling, helps to reduce the spatial dimension of inputs, by decreasing the number of parameters also retains the most important information. Two types of pooling are employed : max pooling and average pooling.

c. Flattening Layer : After convolutional and pooling layers, the data is transformed from a 2D feature map into a 1D vector suitable for fully connected layers.

d. Fully Connected Layers : This layers aims to provide the global connectivity between all neurons in the layer. These layers take the flattened data from the previous step and perform the final classification task. Its main task is to combine and transform these high-level features into the final output.

```
Size of train_labels: 4819
Epoch 1/10
151/151 [==============================] - 90s 590ms/step - loss: 2.5601 - accuracy: 0.7607 - val_loss:
Epoch 2/10
151/151 [==============================] - 92s 609ms/step - loss: 0.1542 - accuracy: 0.9494 - val_loss:
Epoch 3/10
151/151 [==============================] - 92s 611ms/step - loss: 0.0955 - accuracy: 0.9693 - val_loss:
Epoch 4/10
151/151 [==============================] - 93s 619ms/step - loss: 0.0656 - accuracy: 0.9813 - val_loss:
Epoch 5/10
151/151 [==============================] - 89s 590ms/step - loss: 0.0889 - accuracy: 0.9755 - val_loss:
Epoch 6/10
151/151 [==============================] - 87s 575ms/step - loss: 0.0559 - accuracy: 0.9846 - val_loss:
Epoch 7/10
151/151 [==============================] - 90s 595ms/step - loss: 0.0596 - accuracy: 0.9853 - val_loss:
Epoch 8/10
151/151 [==============================] - 92s 609ms/step - loss: 0.0484 - accuracy: 0.9844 - val_loss:
Epoch 9/10
151/151 [==============================] - 91s 606ms/step - loss: 0.1305 - accuracy: 0.9658 - val_loss:
Epoch 10/10
151/151 [==============================] - 86s 570ms/step - loss: 0.0493 - accuracy: 0.9849 - val_loss:
```

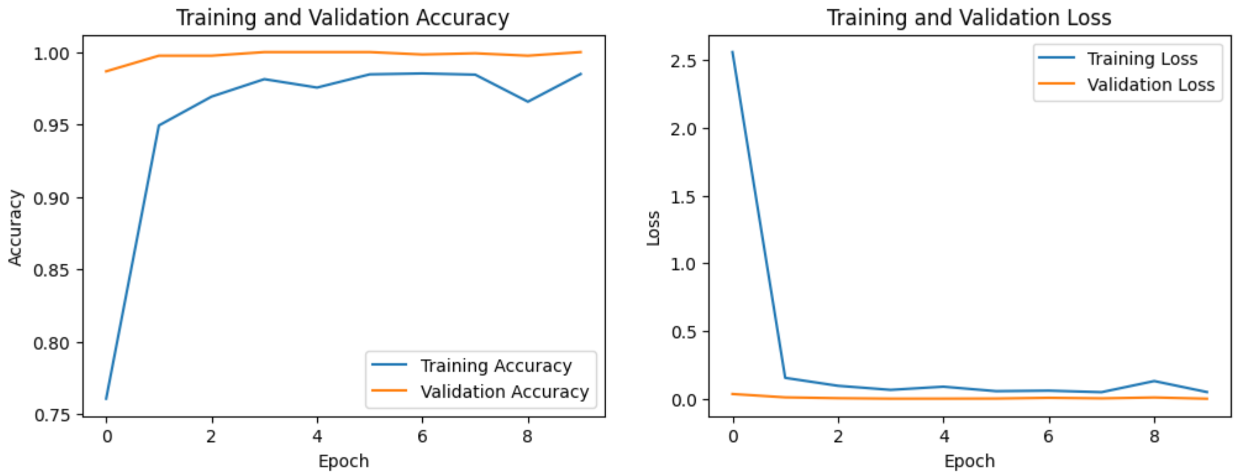
4. Model Training and Evaluation

This model utilizes the Adam optimizer and categorical cross-entropy loss function. It was trained for a set number of epochs, with training and validation accuracy monitored to adjust and fine-tune the model.

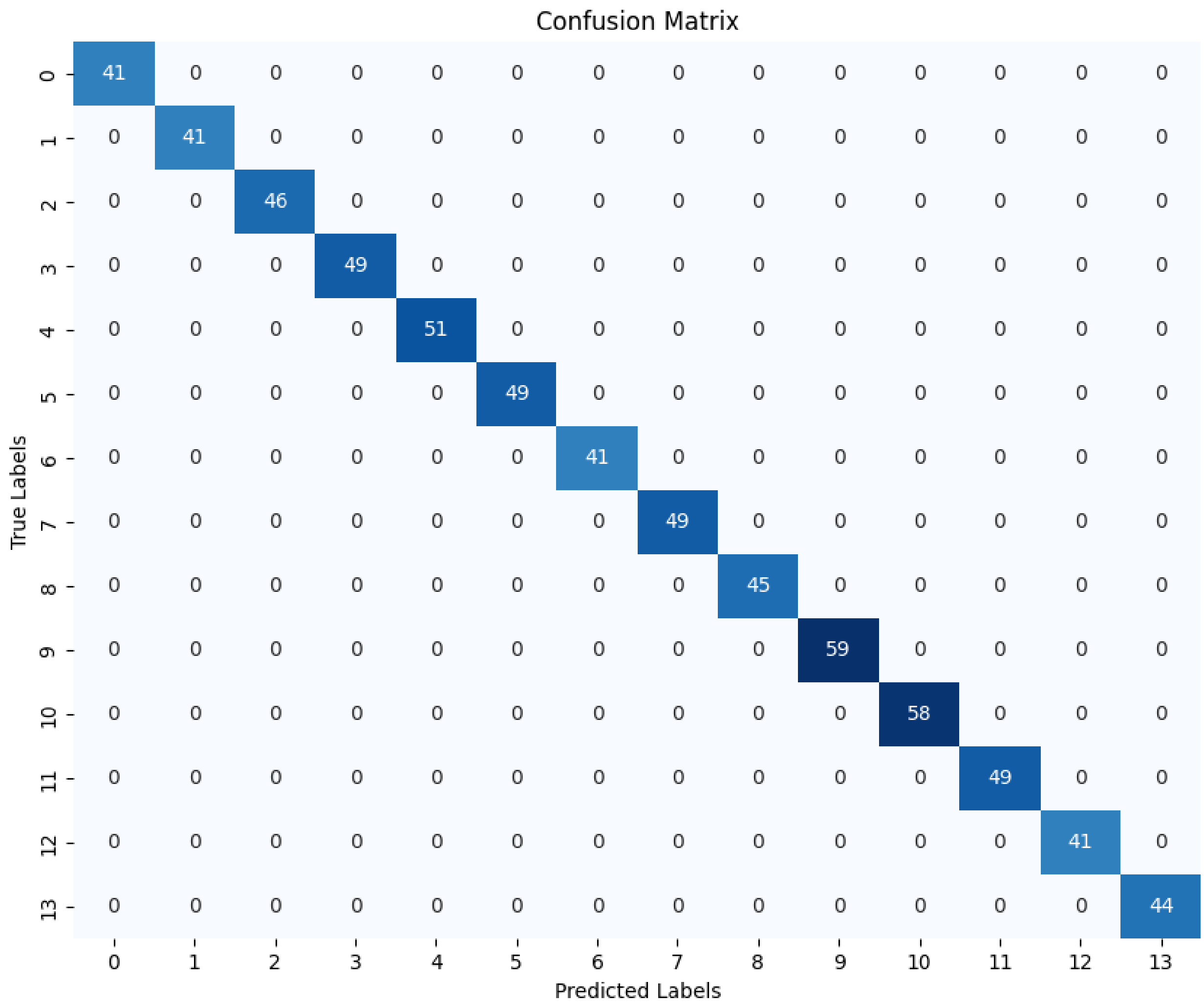
The model’s performance was evaluated on the test set accuracy as the primary metric.

Performance Metrics :

a. Accuracy: It is the proportion of correctly classified images out of the total number of images. For eg, if model correctly classifies 90 out of 100 fruit images, its accuracy would be around 90%.



b.Confusion Matrix : A table that provides a detailed breakdown of the model's correct and incorrect classifications for each fruit class.



5. **Visualization**
Afterwards, various learning curves were plotted to visualize the training and validation accuracy and loss over epochs. These curves provide insights into the model's learning process and helped in diagnosing issues such as overfitting or underfitting. This helps to identify how well the model performs on the training data as it learns.

6. **Hyper-parameter Tuning**

It is the process of finding the best setting for a machine learning model to improve its performance. This step involves trying different values for the model's hyperparameters, like the learning rate or number of layers, to see which combination works best. This helps the model learn more effectively from the data. For example, in our model we adjust the model using hyperparameters such as number of filters, dropout rate, to make the model perform best on the validation set, ensuring it can accurately classify various fruits.

Conclusion

In this way, we developed a CNN based fruit classification system that achieve robust performance across various fruits types from the images using a convolutional neural network architecture. The model, trained using K-Fold Cross Validation, helps to achieve high accuracy reaching more than 90% on the test data. Additionally, the implementation of cofusion matrix provides insights into the model's performance that helps to understand the strengths and potential areas of improvement. By utilizing different techniques such as K-Fold Cross Validation, convolutional and pooling layers, we ensure model's robustness, reliabilty and performance to new data using relevant metrices such as accracy and confusion matrix. In this way we helps farmers to stack similar products in the same pallet in less amount of time. Overall, the system highlights the positive outcome of the fruit classification by accurately identifying fruit types from images.

References:

1. Umer, A. (2023). Understanding Convolutional Neural Networks □: A Beginner's Journey into the Architecture 🚀. [online] CodeX. Available at: https://medium.com/codex/understanding-convolutional-neural-networks-a-beginners-journey-into-the-architecture-aab30dface10.

2. Joseph, J.L., Kumar, V.A. and Mathew, S.P. (2021). Fruit Classification Using Deep Learning. Lecture Notes in Electrical Engineering, pp.807–817. doi:https://doi.org/10.1007/978-981-16-0749-3_62.

3. Momeny, M., Jahanbakhshi, A., Jafarnezhad, K. and Zhang, Y.-D. (2020). Accurate classification of cherry fruit using deep CNN based on hybrid pooling approach. Postharvest Biology and Technology, 166, p.111204. doi:https://doi.org/10.1016/j.postharvbio.2020.111204.

4. Vijayalakshmi, M. and Peter, V.J. (2020). CNN based approach for identifying banana species from fruits. International Journal of Information Technology, 13(1), pp.27–32. doi:https://doi.org/10.1007/s41870-020-00554-1.

5. Ilmi, H. (2023). HillidatulIlmi/EDA-and-Deep-Learning-CNN-Fruit-Classification-Predictions. [online] GitHub. Available at: https://github.com/HillidatulIlmi/EDA-and-Deep-Learning-CNN-Fruit-Classification-Predictions [Accessed 26 May 2024].

6. kaggle.com. (n.d.). Fruit classification with a Simple CNN. [online] Available at: https://www.kaggle.com/code/mauriciofigueiredo/fruit-classification-with-a-simple-cnn.