**PLANT PALETTE**

**A Fruit and Vegetable Prediction using**

**Deep Neural Networks**

**A MINI-PROJECT REPORT**

*Submitted by*

**SREENA R 2116210701256**

**SOWMIYA S 2116210701255**

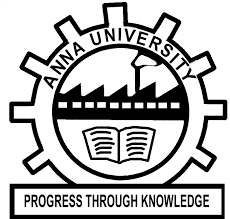
***In partial fulfillment of the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**RAJALAKSHMI ENGINEERING COLLEGE**

**AUTONOMOUS, CHENNAI**

**MAY 2024**

# **RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI**

**BONAFIDE CERTIFICATE**

Certified that this Report titled “**PlantPalette - A Fruit and Vegetable Prediction using Deep Neural Networks**” is the bonafide work of **“ Sreena R (210701256), Sowmiya S (210701255) ”** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Dr. M. RakeshKumar B.Tech., M.E.,

Assistant Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai- 602 105

Submitted to Project Viva-Voce Examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Internal Examiner External Examiner**

**ABSTRACT**

In recent years, deep learning techniques, particularly Convolutional Neural Networks (CNNs), have authenticated exceptional capabilities in various fields, including image recognition, natural language processing, and medical diagnosis. In this study, we propose a CNN-based approach for predicting fruits and vegetables. Leveraging CNN's hierarchical feature learning, the model achieves accurate classification of distinct produce from input images. Preprocessed datasets ensure regularity, while the architecture includes convolutional and max-pooling layers for feature extraction. Dropout regularization prevents overfitting, and batch normalization accelerates convergence. Experimental results demonstrate superior accuracy compared to traditional methods, promising applications in agricultural automation and food quality inspection.

# **ACKNOWLEDGMENT**

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our principal **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. KUMAR M.E., PhD, Director Computing and Information Science , and Head Of Department of Computer Science and Engineering** and our project coordinator **Dr. RAKESH KUMAR M, M.E., Ph.D.,** for his encouragement and guiding us throughout the project towards successful completion of this project and to our parents, friends, all faculty members and supporting staffs for their direct and indirect involvement in successful completion of the project for their encouragement and support.

## **SREENA R**

## **SOWMIYA S**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT** | iii |
|  | **LIST OF FIGURES** | vi |
|  | **LIST OF TABLES** | vii |
| **1.** | **INTRODUCTION** | 5 |
|  | 1.1 GENERAL | 5 |
|  | 1.2 OBJECTIVE | 5 |
|  | 1.3 EXISTING SYSTEM | 5 |
|  | 1.4 PROPOSED SYSTEM | 6 |
| **2.** | **LITERATURE SURVEY** | 7 |
| **3.** | **SYSTEM DESIGN** | 11 |
|  | 3.1 DEVELOPMENT ENVIRONMENT | 11 |
|  | 3.1.1 HARDWARE SPECIFICATIONS | 11 |
|  | 3.1.2 SOFTWARE SPECIFICATIONS | 11 |
|  | 3.2 SYSTEM DESIGN | 12 |
|  | 3.2.1 ARCHITECTURE DIAGRAM | 1 |
|  |  |  |
| **4.** | **PROJECT DESCRIPTION** | 15 |
|  | 4.1 MODULE DESCRIPTION | 15 |
|  | 4.1.1 DATA PRE-PROCESSING | 15 |
|  | 4.1.2 TRAINING SET | 15 |
|  | 4.1.3 TRAINING MODEL | 16 |
|  | 4.1.4 HYPERPARAMETER TURING | 16 |
| **5.** | **IMPLEMENTATION AND RESULTS** | 17 |
|  | 5.1 IMPLEMENTATION | 17 |
|  | 5.2 OUTPUT SCREENSHOTS | 17 |
|  | 5.3 RESULT ANALYSIS | 19 |
| **6.** | **CONCLUSION AND FUTURE ENHANCEMENTS** | 20 |
|  | **REFERENCES** | 21 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **TITLE** | **PAGE NO** |
| 3.1 | ARCHITECTURE DIAGRAM | 12 |
| 5.1 | PLANT PALETTE USER INTERFACE-1 | 18 |
| 5.2 | PLANT PALETTE USER INTERFACE-2 | 18 |
| 5.3 | TRAINING MODEL ACCURACY | 19 |

5.4 VALIDATION MODEL ACCURACY 19

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE NO** | **TITLE** | **PAGE NO** |
| 3.1 | HARDWARE SPECIFICATIONS | 11 |
| 3.2 | SOFTWARE SPECIFICATIONS | 12 |

**CHAPTER 1**

**INTRODUCTION**

**1.1 GENERAL**

The study addresses a Convolutional Neural Network (CNN) approach for predicting fruits and vegetables. The proposed model leverages the hierarchical feature learning ability of CNNs to spontaneously extract discriminative features from raw input images, enabling accurate classification of different fruits and vegetables. The dataset is composed of images of various fruits and vegetables collected from diverse sources. These images are preprocessed to ensure uniformity in size, color, and strong matches.

**1.2 OBJECTIVE**

The objective of this study is to develop a CNN-based approach for accurately predicting fruits and vegetables from images. The focus is on leveraging the strengths of deep learning techniques, such as hierarchical feature extraction, regularization, and optimization algorithms, to achieve high classification accuracy. The study also aims to explore the impact of various architectural components and hyperparameters on the model's performance through comprehensive experiments and ablation studies.

**1.3 EXISTING SYSTEM**

Traditional machine learning methods for fruit and vegetable classification often rely on manually engineered features and are limited in their ability to capture complex patterns in the data. These methods may include approaches such as decision trees, support vector machines, and k-nearest neighbors, which can struggle with large and diverse datasets. In contrast, deep learning models, particularly CNNs, have demonstrated superior performance by automatically learning features from raw input data, leading to improved classification accuracy and robustness.

**1.4 PROPOSED SYSTEM**

The proposed CNN model consists of multiple convolutional layers followed by max-pooling layers to extract spatial features and reduce dimensionality. Dropout regularization prevents overfitting, and batch normalization accelerates convergence during training. Learning rate scheduling and adaptive methods like AdamW ensure optimal training progress. To handle imbalanced classes, focal loss or class-weighted loss functions are used. Attention mechanisms and capsule networks help the model focus on relevant image regions, improving interpretability and performance. The softmax activation function enhances prediction confidence and accuracy, with temperature scaling providing finer control over probability distributions. These techniques make the CNN-based system highly accurate and versatile, suitable for applications in agricultural automation, food quality inspection, and dietary analysis.

**CHAPTER 2**

**LITERATURE SURVEY**

Several studies have explored fruit detection as an image segmentation problem, aiming to distinguish fruit from the background. Wang et al. focused on apple detection for yield prediction. Their system leveraged color and the unique specular reflection patterns of apples. Additional information, like average apple size, was used to refine detections and separate regions potentially containing multiple apples. They also employed a heuristic that only accepted mostly round detections.exclamation Bac et al. [12] proposed a segmentation approach for sweet peppers using a six-band multispectral camera and various features, including raw data, normalized difference indices, and texture features. Their experiments in a controlled environment yielded promising segmentation results, although they acknowledged limitations in building a reliable obstacle map.exclamation This section delves into existing fruit classification schemes, highlighting their shortcomings and the ongoing challenge of fruit classification.

While [8] employed random forests for fruit name recognition based on shape and color features, [9] utilized a 13-layer CNN with data augmentation for image-based fruit classification.Their approach included image restoration, gamma correction, and noise injection, comparing max pooling and average pooling, but lacked results on imperfect images during classification. Date fruit classification based on color, size, and texture features using SVM was explored in [10]. Their findings demonstrated SVM's superiority over neural networks, random forests, and decision trees in terms of accuracy. Machine learning oriented classification incorporating wavelet entropy, principal component analysis, feed-forward neural networks, and biogeography-based optimization for fruit classification was presented in [3]. K-fold cross-validation was employed for statistical analysis, with extracted features and measured accuracy demonstrating the effectiveness of their proposed methods.

In 2009, Woo Chaw Seng, Seyed Hadi Mirisaee[7] proposed a Fruit Recognition System. The system was applied on fifty image samples. The mean color values of these images are computed. The fruit area and perimeter are chosen as features to distinguish one fruit from another. Their system mainly consisted of five main processing modules, which are, fruit input selection module, fruit color computing module, fruit shape computing module, fruit size computing module, and fruit classification or recognition module. It used the KNN algorithm for classification and recognition of the input fruit.

Besides color, texture, and edge properties, many different methods are used in fruit and vegetable classification. For example, scholars use gas sensors, near-infrared, and high performance liquid chromatography devices to scan the fruit [15,16,17]. Fei-Fei et al. [18] introduced prior knowledge into the estimation of the distribution, thus reducing the number of training examples to around ten images while preserving a good recognition rate. Even with this improvement, the problem of exponential growth with the number of parts persists, which makes it impractical for the problem presented in this paper, which requires speed for on-line operation.

Another related application is Deep Fruit Detection for robotic harvesting in orchards. That research employs Faster R-CNN and compares the performance against other architectures such as VGG and ZFNet. They also explore the number of training images, transfer learning and data augmentation. They study three fruits: apple, mango and almond; with RGB images generated by themselves [2].

[13] automatically recognize fruit from multiple images.Counting numbers of fruits on a plant with image analysis method. Fruit counting from multiple views may arise the problem of ambiguity and counting of fruits via images may not define the exact numbers.The Aforementioned classification schemes have the follow-ing shortcomings:

1. The classifier may not be robust due to ambiguity in the fruitimages or may have identical shape, size or color features.

2. Some classification systems are appropriate to recognize fruits and vegetables based on distinct features.

To overcome above mentioned shortcomings, this paper proposed a scheme for classification of fruit images using deep learning applications. Deep learning models achieve excellent accuracy in various approaches to image classifi-cation and recognition methods. For these reasons, we were encouraged to use deep learning for fruit image classification.This proposed research work investigates three different deep learning models. CNN is employed to develop dis-criminative features. CNNs with deep structure have gained tremendous success in classification of text, human detection etc. using Convolution layer, nonlinear layer and pooling layer[14]. CNN is employed in proposed research work to extract optimal features for classification of fruits.

Authors [6] proposed an automatic mango sorting and grading model using a DL technique, where eight types of harvested mango features such as size, shape, color, and texture were considered. Image rotation, image translation, image zooming, image sharing, and image horizontal flip data augmentation methods are used. The article aimed to classify the papaya fruit according to its maturity level, whether it was ripe, partially ripe, or unripe [19]. Extensive DL techniques were used to identify the papaya fruit images. The trained model achieved 100% accuracy on the test dataset, explaining the feasibility of the proposed approach.

Deep learning-based fruit classification methods are gaining traction in the post-harvesting stage and fruit industry. Fan et al. [4] proposed a method for sorting apples into normal and defective categories. Their dataset comprised 300 Fuji apples with various defects. However, a common limitation in many studies [20, 21, 22, 23, 24, 25, 26] is the use of a single fruit species under controlled lighting conditions, potentially affecting the generalizability of their conclusions. Additionally, most existing datasets are limited in the number of fruit types and lack vegetable varieties. This study addresses these limitations by employing a comprehensive fruit and vegetable database encompassing various species. Furthermore, while prior research primarily focused on fruit classification, this work investigates the quality evaluation and sorting of vegetables as well. The application of deep neural networks has significantly improved object classification and detection performance, as demonstrated in several studies, including [5, 1, 27].

**CHAPTER 3**

**SYSTEM DESIGN**

**3.1 DEVELOPMENT ENVIRONMENT**

**3.1.1 HARDWARE SPECIFICATIONS**

| **COMPONENTS** | **SPECIFICATIONS** |
| --- | --- |
| PROCESSOR | Intel Core i5 |
| RAM | 8 GB RAM |
| HARD DISK | 512 GB |
| PROCESSOR SPEED | Minimum 1.1 GHz |

**Table 3.1 HARDWARE SPECIFICATIONS**

**3.1.2 SOFTWARE SPECIFICATIONS**

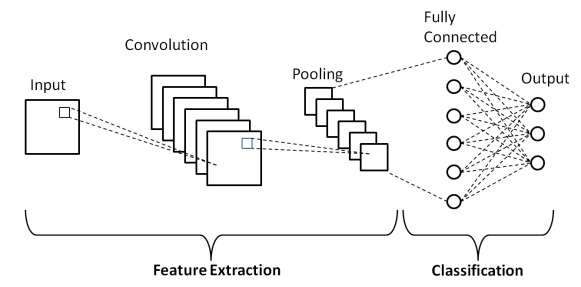
The software specifications in order to execute the project has been listed down in the below table. The requirements in terms of the software that needs to be preinstalled and the languages needed to develop the project has been listed out below.

| **COMPONENTS** | **SPECIFICATIONS** |
| --- | --- |
| FRONT END | HTML, CSS, Bootstrap, JavaScript |
| BACK END | Python, Django |
| FRAMEWORKS | Tensor Flow, Keras, Streamlit |
| SOFTWARES USED | Visual Studio, Jupyter Notebook |

**Table 3.2 SOFTWARE SPECIFICATIONS**

**3.2 SYSTEM DESIGN**

**3.2.1 ARCHITECTURE DIAGRAM**

****

**Fig 3.1 ARCHITECTURE DIAGRAM**

**PRE PROCESSING:**

The first step requires gathering a diverse dataset of images containing various fruits and vegetables from different sources. It's pivotal to ensure that the dataset encloses a wide range of classes, variations in appearance, and environmental conditions to increase the model's generalization capability. Once collected, the images undergo preprocessing to standardize attributes such as size, color, and orientation. Techniques like resizing, normalization, and augmentation are applied to ensure uniformity and increase the dataset's diversity, which aids in training a more robust model.

**TRAINING PROCESS:**

The training process comprises 36 classes containing a variety of fruits and vegetables. The dataset is split into three categories as Training data, Validation data and Testing data of equal ratio to ensure sufficient data for model training and to facilitate robust validation. After splitting the data, select a suitable CNN architecture for the task of fruit and vegetable prediction.

Import the libraries and frameworks required for the project. In this project, we used the Tensorflow framework. Tensorflow is an open source framework used to develop models for various tasks, including natural language processing, image recognition, handwriting recognition, and different computational based simulations such as partial differential equations. Since we need to train the images by considering their features, we are building a neural network consisting of the features that were common in the images. So we are using Keras which is a high-level neural network library that runs on the top of Tensorflow.

After loading the training set and validation set, create a CNN model. CNN(Convolutional Neural Network)is a type of artificial neural network used primarily for image recognition and processing, due to its ability to recognize patterns in images. Build the layers of CNN such as Conv2d, MaxPool2d and Dense layer. Conv2D creates a convolution kernel that is convolved with the layer input over a single spatial (or temporal) dimension to produce a tensor of outputs. In image processing a kernel is a convolution matrix which can be used for blurring, sharpening, embossing, edge detection, and more by doing a convolution between a kernel and an image. The MaxPool2D layer takes the important points from the Conv2D layer. The Flatten layer converts the 2D array into a one dimensional array.

The Dropout Layer is a regularization technique used in CNN to help prevent overfitting. The next layer is the Dense Layer which takes the input from the previous layers and produces the output. Dense layer is a fully, deeply connected neural network layer. It uses the activation function named Softmax converts a vector of values to a probability distribution. These values range between 0 to 1 and sums up to 1. Each vector is handled independently.

After building the layers, fit the training data and validation data. Fit the model for 30 epochs. Epoch is the number of iterations used to train the model for one cycle. In an epoch, we use all of the data exactly once. Then save the trained model for further testing.

Preprocess the test dataset using Keras API. Then load the trained model into the testing file. Create a function that takes the input image and converts it into an array for further predicting the desired result.

**CHAPTER 4**

**PROJECT DESCRIPTION**

**4.1 MODULE DESCRIPTION**

**4.1.1 DATA PRE-PROCESSING:**

The first step requires gathering a diverse dataset of images containing various fruits and vegetables from different sources. It's pivotal to ensure that the dataset encloses a wide range of classes, variations in appearance, and environmental conditions to increase the model's generalization capability. Once collected, the images undergo preprocessing to standardize attributes such as size, color, and orientation. Techniques like resizing, normalization, and augmentation are applied to ensure uniformity and increase the dataset's diversity, which aids in training a more robust model.

**4.1.2 TRAINING SET:**

The designed CNN model is then trained using the preprocessed dataset. During training, the model learns to map the input images to their corresponding fruit or vegetable classes by adjusting its parameters through optimization algorithms like stochastic gradient descent (SGD) or Adam. The training process requires iteratively feeding batches of images into the network, computing the loss between the predicted and actual labels, and updating the model's parameters to minimize this loss. Techniques such as learning rate scheduling and early stopping are employed to optimize convergence and prevent overfitting.

**4.1.3 TRAINING MODEL:**

The training process for the proposed CNN model begins with splitting the dataset into training, validation, and test sets, followed by preprocessing steps such as resizing, normalization, and augmentation to enhance diversity and generalization. The CNN is constructed with convolutional layers, ReLU or Leaky ReLU activations, and max-pooling layers for effective feature extraction. Dropout regularization and batch normalization are applied to prevent overfitting and accelerate convergence. Advanced optimization algorithms like Adam or RMSprop, along with learning rate scheduling, ensure efficient parameter updates and optimal training progress. Focal loss or class-weighted loss functions address class imbalances, improving classification accuracy for underrepresented categories. The model's performance is robustly evaluated using k-fold cross-validation, ensuring reliable and generalizable results.

**4.1.4 HYPERPARAMETER TUNING:**

Hyperparameter tuning plays a crucial role in optimizing the performance of the CNN model. Parameters such as learning rate, dropout rate, batch size, and optimizer configurations are thoroughly adjusted and authenticated using techniques like grid search or random search. The goal is to identify the optimal combination of hyperparameters that maximizes the model's accuracy and generalization ability on validation datasets.

**CHAPTER 5**

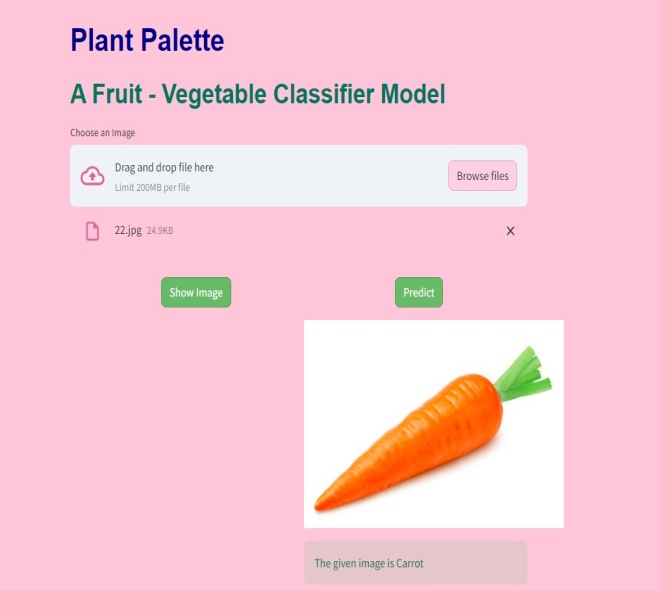
**IMPLEMENTATION**

**5.1 IMPLEMENTATION**

The specifics of implementation must go hand in hand when designing the CNN architecture that is particularly suited for fruit and vegetable prediction. The architecture commonly incorporates several Convolution layers together with Multiple pooling layers to enhance features extraction from the input images hierarchically. At the same time, the basic components like activation functions, dropout layers for the regularization of the overfitting problems, and batch normalization are incorporated into the model at specific points to improve the outcome and minimize overfitting. Some of the parametric settings include learning rate, batch size, and optimizers which are critical in enhanced convergence and improved model performance.

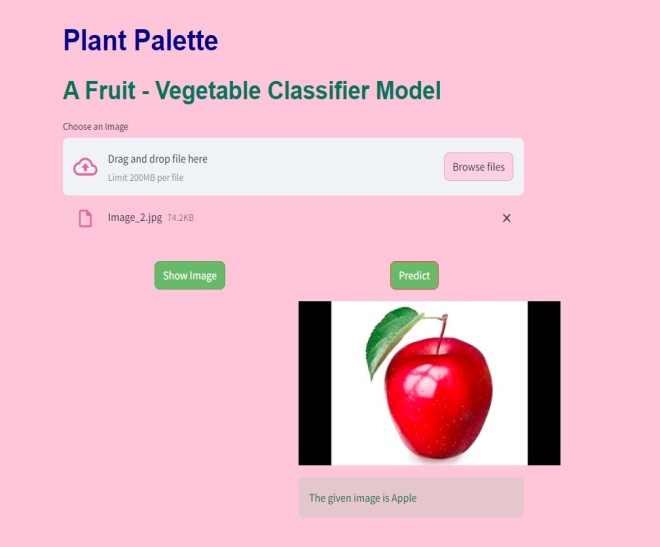
**5.2 OUTPUT SCREENSHOTS**

The trained model is tested using the user interface made of a framework named Streamlit which is an open-source Python framework for building Machine Learning and Deep Learning web applications. Here is the output screenshots of the tested model:



**Fig 5.2 PLANT PALETTE USER INTERFACE -1**

The image of a vegetable is given and it produces the corresponding output as shown in the Figure 5.2.



**Fig 5.3 PLANT PALETTE USER INTERFACE-2**

The image of a fruit is given and it produces the corresponding output as shown in the Figure 5.3.

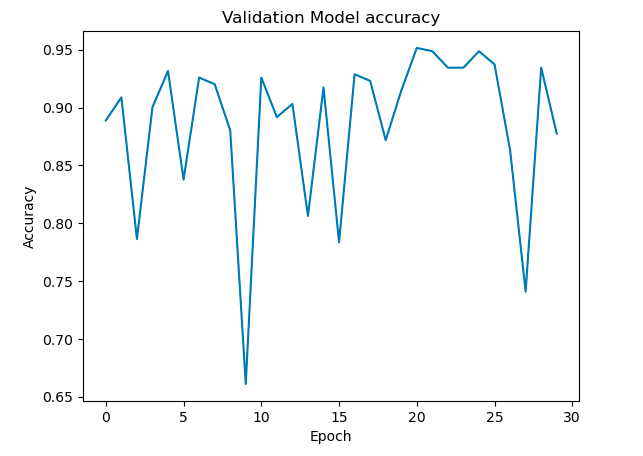
**5.3 RESULTS**

The trained model achieved 89.4% accuracy while the validation model achieved 87.7% accuracy. Here is the visualization of the trained model in each epoch:



**Fig 5.4 TRAINING MODEL ACCURACY**

The validation model is visualized as follows:



**Fig 5.5 VALIDATION MODEL ACCURACY**

**CHAPTER 6**

**CONCLUSION AND FUTURE ENHANCEMENTS**

**6.1 CONCLUSION**

We studied 27 different papers and proposed that deep learning techniques can be effectively applied to classify fruits and vegetables. Through our research, we concluded that Convolutional Neural Networks (CNNs) are the most suitable algorithm for this task. CNNs play a crucial role in various fields such as the food industry, agriculture, and other industries where predicting the quality of fruits and vegetables is essential. Our comprehensive training and validation procedures for the proposed models yielded impressive accuracy rates of 87.57% for the training data and 89.78% for the validation data. These results underscore the effectiveness of our approach in accurately distinguishing between different types of fruits and vegetables.

**6.2 FUTURE ENHANCEMENTS**

* Extend the application by adding more classes of fruits and vegetables beyond the current 36.
* Explore the potential of utilizing our dataset for other deep learning methods beyond CNNs.
* Investigate deep learning techniques for fruit maturity classification.
* Develop models for fruit and vegetable quality assessment.
* Examine alternative algorithms and techniques to uncover new insights.
* Refine predictive models to address dynamic challenges in agricultural automation.
* Enhance applications in food quality inspection through advanced deep learning methodologies.

**REFERENCES**

1. Russakovsky, Olga, et al. "Imagenet large scale visual recognition challenge." International journal of computer vision 115 (2015): 211-252.
2. Bargoti, Suchet, and James Underwood. "Deep fruit detection in orchards." 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017.
3. Wang, Shuihua, et al. "Fruit classification by wavelet-entropy and feedforward neural network trained by fitness-scaled chaotic ABC and biogeography-based optimization." Entropy 17.8 (2015): 5711-5728.
4. Fan, Shuxiang, et al. "Online detection of defective apples using a computer vision system combined with deep learning methods." Journal of Food Engineering 286 (2020): 110102.
5. Russakovsky, Olga, et al. "Imagenet large scale visual recognition challenge." International journal of computer vision 115 (2015): 211-252.
6. Rizwan Iqbal, Hafiz Muhammad, and Ayesha Hakim. "Classification and grading of harvested mangoes using convolutional neural network." International Journal of Fruit Science 22.1 (2022): 95-109.
7. Seng, Woo Chaw, and Seyed Hadi Mirisaee. "A new method for fruits recognition system." 2009 International conference on electrical engineering and informatics. Vol. 1. IEEE, 2009.
8. Zawbaa, Hossam M., et al. "Automatic fruit classification using random forest algorithm." 2014 14th international conference on hybrid intelligent systems. IEEE, 2014.
9. Zhang, Yu-Dong, et al. "Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation." Multimedia Tools and Applications 78 (2019): 3613-3632.
10. Abi Sen, Adnan Ahmed, et al. "A classification algorithm for date fruits." 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE, 2020.
11. Wang, Qi, et al. "Automated crop yield estimation for apple orchards." Experimental robotics: The 13th international symposium on experimental robotics. Springer international publishing, 2013.
12. Bac, C. W., Jochen Hemming, and E.J.Van Henten. "Robust pixel based classification of obstacles for robotic harvesting of sweet-pepper." Computers and electronics in agriculture 96 (2013): 148-162.
13. Song, Y., et al. "Automatic fruit recognition and counting from multiple images." Biosystems Engineering 118 (2014): 203-215.
14. Zhang, Yu-Dong, et al. "Image based fruit category classification by 13 layer deep convolutional neural network and data augmentation." Multimedia Tools and Applications 78 (2019): 3613-3632.
15. Pardo-Mates, Naiara, et al. "Characterization, classification and authentication of fruit-based extracts by means of HPLC-UV chromatographic fingerprints, polyphenolic profiles and chemometric methods".Food chemistry 221 (2017): 29-38.
16. Shao, Wenhao, et al. "Rapid classification of Chinese quince (Chaenomeles speciosa Nakai) fruit provenance by near-infrared spectroscopy and multivariate calibration." Analytical and Bioanalytical Chemistry 409 (2017): 115-120.
17. Ciptohadijoyo, S., et al. "Electronic nose based on a partition column integrated with a gas sensor for fruit identification and classification." Computers and Electronics in Agriculture 121 (2016): 429-435.
18. Fei-Fei, Li, Robert Fergus, and Pietro Perona. "One-shot learning of object categories." IEEE transactions on pattern analysis and machine intelligence 28.4 (2006): 594-611.
19. Al-Masawabe, Marah M., et al. "Papaya Maturity Classifications using Deep Convolutional Neural Networks." (2021).
20. Hao, Jinglei, Yongqiang Zhao, and Qunnie Peng. "A Specular Highlight Removal Algorithm for Quality Inspection of Fresh Fruits." Remote Sensing 14.13 (2022): 3215.
21. Ananthanarayana, Tejaswini, Raymond Ptucha, and Sean C. Kelly. "Deep learning based fruit freshness classification and detection with CMOS image sensors and edge processors." Electronic Imaging 32 (2020): 1-7.
22. Chen, Ming-Chih, Yin-Ting Cheng, and Chun-Yu Liu. "Implementation of a Fruit Quality Classification Application Using an Artificial Intelligence Algorithm." Sensors & Materials 34 (2022).
23. Ni, Jiangong, et al. "Monitoring the change process of banana freshness by GoogLeNet." IEEE Access 8 (2020): 228369-228376.
24. Fan, Shuxiang, et al. "Online detection of defective apples using computer vision combined with deep learning methods." Journal of Food Engineering 286 (2020): 110102.
25. Bhargava, Anuja, and Atul Bansal. "Classification and grading of multiple varieties of apple fruit." Food Analytical Methods 14.7 (2021): 1359-1368.
26. Palakodati, Sai Sudha Sonali, et al. "Fresh and Rotten Fruits Classification Using CNN and Transfer Learning." Rev. d'Intelligence Artif. 34.5 (2020): 617-622.
27. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).