CLASSIFICATION OF FRESH AND ROTTEN ORANGES USING CONVOLUTIONAL NEURAL NETWORKS

A
Project Report
Submitted in partial fulfillment of the Requirements for Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

R. ABHINAV REDDY (1602-17-737-062)

Y. SAI RACHANA (1602-17-737-098)

Y S S SRI DATTA (1602-17-737-107)

Under the guidance of

Dr. V. VENKATA KRISHNA

Professor

Department of Information Technology
Vasavi College of Engineering (Autonomous) (Affiliated to Osmania University) Ibrahimbagh, Hyderabad-31

2019-2020

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

I, **R.ABHINAV REDDY, Y. SAI RACHANA, Y S S SRI DATTA** bearing hall ticket number **1602-17-737-062, 1602-17-737-098, 1602-17-737-107** hereby declare that the project report entitled **"CLASSIFICATION OF FRESH AND ROTTEN ORANGES USING CONVOLUTIONAL NEURAL NETWORKS"** under the guidance of **Dr. V. VENKATA KRISHNA**, Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology.**

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

R. ABHINAV REDDY(1602-17-737-062)

Y. SAI RACHANA (1602-17-737-098)

Y S S SRI DATTA (1602-17-737-107)

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

BONAFIDE CERTIFICATE

This is to certify that the project entitled "CLASSIFICATION OF FRESH AND ROTTEN ORANGES USING CONVOLUTIONAL NEURAL NETWORKS" being submitted by R. ABHINAV REDDY, Y. SAI RACHANA , Y S S SRI DATTA bearing 1602-17- 737-062, 1602-17-737-098, 1602-17-737-107 in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by him/her under my guidance.

Dr. V. Venkata Krishna	**Dr. K. Ram Mohan Rao**
Professor	**HOD, IT**

ACKNOWLEDGEMENT

The satisfaction that accompanies that the successful completion of the project would not have been possible without the kind support and help of many individuals. We would like to extend my sincere thanks to all of them.

We would like to take the opportunity to express our humble gratitude to Dr. V. Venkata Krishna sir under whom we executed this project.

We would like to thank our coordinator Mr. Dharma Reddy sir who kept us motivated throughout to complete the project successfully. His ideas and insights helped us to make this project more operative.

We would also use this opportunity to thank our senior Veeramalla Akhil (1602-16-737- 004). We are grateful to his guidance, and constructive suggestions that helped us in the preparation of this project. His constant guidance and willingness to share his vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks. We would like to thank all faculty members and staff of the Department of Information Technology for their generous help in various ways for the completion of this project.

Finally, yet importantly, we would like to express our heartfelt thanks to our HOD Dr. K. Ram Mohan Rao Sir and classmates for their help and wishes for the successful completion of this project.

ABSTRACT

India is worldwide well known for exporting fruits, having a massive significance in the world. Global food security is essential for the durable production of fruits as well as for a remarkable reduction in pre and post-harvest waste. For sustainability of life and rural development, assessment of fruits in a non-destructive manner is required. The fruits which are available in the market must fulfill buyer needs. The quality is very important to increase sales and competitiveness in the market. Quality assessment of agricultural products is one of the most important factors in promoting their marketability and waste control management. The oranges classification process is performed by humans, it is not accurate and we often get the different results due to the difference in human perceptions. Generally, the reviewing of Orange fruit is done by the visual examination and by utilizing the size as a specific quality characteristic. For large volumes, we cannot assess by human graders. Picture preparing offers answer for computerized Orange Fruits estimate on solid, predictable and quantitative data which are separated from dealing with extensive volumes that may not be accomplished by utilizing the human graders. Machine learning is used in many fields but the most upgrading and preferred area in which this technology can be seen with value is agriculture. Machine Learning and Deep Neural Networks have made a significant footprint in agriculture area. Thus, we are using Convolutional Neural Networks to classify the oranges. This research contains a division of orange images based on fresh and rotten criteria, the images are of different kind based on the rotation of images. Then the classification of images is done based on CNN, binary cross-entropy loss function, along with accuracy is calculated and resultant graphs are illustrated with an accuracy of **78.57%**.

TABLE OF CONTENTS

LIST OF FIGURES:

LIST OF TABLES:

1 INTRODUCTION

India is the country where there is a higher rate of exports as well as imports in case of vegetables, fruits and it's a known fact. Among the fruits, our country ranks first in production of Bananas, Papayas and Mangoes. The people living in India are more dependent on an agricultural domain, statistics say that 61.5% are dependent on it. Despite the ranking of India in the area of fruits, fruits have defects, for example, apple has defects like rot, blotch, and scab.

The fruits must be of high quality for the customers to purchase them. The vendors or the distributors buy these fruits from farmers by sending their men/ workers to check the harvest. The manual graders can’t classify the fruits into healthy and rotten accurately as their might be a difference in human perceptions. Also, the work is too tedious if the fruits are in larger quantity. Thus, there is a need for automation of the work done by the human graders. One of the possible solution to the above problem is to use Machine Learning and Deep Neural Network models to detect and classify the fruits into fresh and rotten categories.

In this case, here we will discuss about orange fruit. Orange is a kind of citric fruit. The production of oranges per year is estimated to be 54.23 million tonnes and the highest producing country is Brazil which is 35.6 million tonnes. For the past half-century, the Brazilian orange story has increasingly been one of dominance. Presently, about one in three oranges produced worldwide is from the country. Although citrus trees have been cultivated in Brazil for centuries, it was only in the 1960s through the 1980s, when top producer Florida wrestled with repeated crop failures, that Brazil rose to its current top position. And while Brazil’s rise was on the back of another producing region’s failure, it now appears that the South American giant is finally facing the prospect of its own failure. Tree diseases, drought, and shifting consumer preferences now pose formidable threats to Brazilian citrus.

The classification of oranges is done based on Healthy ones and Defected ones (see Fig. 1) . The defects on oranges include insect damage, copper burn, phytotoxicity, etc.

Fig. 1. Types of Oranges- Fresh and Rotten

The proposed solution to the above mentioned problem is to classify orange fruit images into fresh and rotten categories using Convolutional Neural Networks. Using the CNN model, we can classify the oranges. So, the human graders task becomes much easier than before. The only task of them is to prepare quantitative, solid and predictable images of the oranges and feed them to the model to get the accurate as well as fast results. Thus, the assessment of fruits is done in a non-destructive manner leading to the increase in sales and competitiveness in the market.

2 RELATED WORKS

Ahmed.M.Abdelsalam and Mohammed.S.Sayed(2016) built a computer vision system that identifies the outdoor layered defects of orange fruits using a sensor called multi-spectral images. In this model[7], they have implemented an algorithm that captures orange fruit images by only a Near-Infra Red component. After taking the respective images, pre-processing steps on these images was done, which detects the seven different color components of orange which is a thresholding technique that was implemented. After all these above steps a process of voting is done on all the obtained seven different threshold color images for detecting the defected and defect-free orange fruits.

Hardik Patel ,Rashmin Prajapati and Milin Patel developed a model with a method of image preparation which is done in a computer, where the fruits are estimated on different factors like solid and quantitative data and mainly for avoiding human grading system. This method shows Bacteria spot defects and also they proposed a framework for reviewing. The factors like texture,color and size are required for the early appraisal of the fruit orange. The side view of oranges and some characteristics of the fruits are removed by identifying some sort of calculations. By observing these characteristics they figured out the reviewing process[6]. In this reviewing a very decent Orange quality and evaluating pf zones is done by considering the advantages and disadvantages, the classification of different features were discussed.

Yuwin Chen,Jinghua Wu and Mengtian Cui[5] proposed a method on automatic grading detection of orange which is based on computer vision. In this method, the orange fruit images are gathered and preprocessed, an image segmentation and edge detection were applied on the images of orange fruits and are segmented. Based on the segmentation of images, main features of oranges are extracted, they are fruit color, fruit size, orange surface defect and shape of the fruit, all of these features were learnt by using neural network which yields automatic detection of grade first oranges. The accuracy of grading obtained was 94.38% , accuracy of classification of the grade first is 100%. When it was compared with artificial mode, excellent real-time identification performancerate was observed.

M.Recce, J.Taylor, A.Piebe and G.Tropiano developed a novel system for orange grading into quality bands of three with respect to the characteristics of the surface[1].This method is the only non-automated processing operation in the family of citrus. This system also handles a huge variety of defect marking and surface coloration, shape(highly eccentric to spherical) and also the size of fruit which is from 55mm to 100mm. In order to distinguish the defects, the point of attachment of stem which is nothing but calyx must also be recognized. For identifying the radial color variation they implemented a neural network classifier in the area of rotation invariant transformations,which is also called as Zernike moments. This method requires a complex pattern recognition and also a high throughput which needs 5 to 10 oranges per second.Grading of these fruits is achieved by imaging the fruit from all orthogonal directions simultaneously by passing them through the chamber of inspection. The first stage contains the histograms of all the views of orange fruit which are analyzed by using classifiers of neural networks. From these views, the one which contains defect is further analyzed by utilizing five independent masks combined with a classifier of neural network. The expensive process of stem detection is applied to a slight part of the images. From all these steps results are presented with performance analysis.

Manali R. Satpute and Sumati M. Jagdale [2] proposed a system which is based on automatic inspection of fruit quality for tomato defect detection, grading and sorting of tomato. In this method as a primary step, they segmented the tomato-based on the algorithm called OTSU. After the segmentation of tomato, extracting of features like colour detection, size detection was done. Methods like Erosion and Dilation were used for size detection, these features where in turn utilized for size detection like large, medium and small. In order to extract colors like green, yellow and red, color detection was used for tomato sorting.

A model was proposed for the classification of citrus fruit using the parameter GLCM. They performed the conversion of RGB image into the gray scale image, extraction of features was implemented using the GLCM feature extraction[4]. The feature of GLCM is four types like Energy, Correlation, Contrast, and Homogeneity, all these features are used for feature extraction.A model was constructed for grading and sorting of agricultural product. Primarily RGB images are converted into scale GRAY image which is nothing but pre-processing. The secondary step is the extraction of feature, here in this method extraction is based on shape feature, based on fuzzy logic and Support vector machine classification of fruits was performed.

A model for processing images based on Dates Maturity Status and dates classification[3] was constructed by T. Najeeb and M. Safar. The primary step proposed was to resize the images i.e., pre-processing of input data. The second step was to perform segmentation within the threshold. Post Segmentation they have processed the image measurement labeling. The final step was, based on color detection and size detection through which extraction of a feature of the fruit was proposed.

A model for olive fruit detection of defects using an automatic method. The following procedure was proposed with firstly pre-processing the images from RGB to GRAY on the given olive fruit. Segmentation was done based on the threshold, following the feature extraction of the olive fruit[11]. They have implemented the SICA which is the Special Image Convolution Algorithm,(THMT) which is a Texture Homogeneity Measuring Technique in the proposed method.

3 PROPOSED WORK

A. USE CASES/ DELIVERABLES OF THE PROJECT:

- **Classification of Oranges** into fresh and rotten oranges using Convolutional Neural Network.
- Provide an **User Interface** for the CNN model.
- Perform **error analysis**/ analysis of the CNN model built to obtain higher accuracy.

B. UI PROTOTYPES/ SCREENSHOT:

The CNN model was built to classify the Oranges into Fresh and Rotten categories. The model was saved using model.save() command. Using the command, we can save the model as a Hierarchical Data Format (HDF) file with h5 extension. The UI is created using HTML, CSS, JavaScript, Python's library Flask. The UI code internally executes the entire model whenever we pass the orange images as input in UI. The below image(see Fig. 2) shows the UI prototype where

- Option to attach input (Oranges) images,
- Predict button, when pressed, passes the input images to the CNN model and runs the entire model. Thus, displays the output i.e., the given input orange image is Fresh/ Rotten.

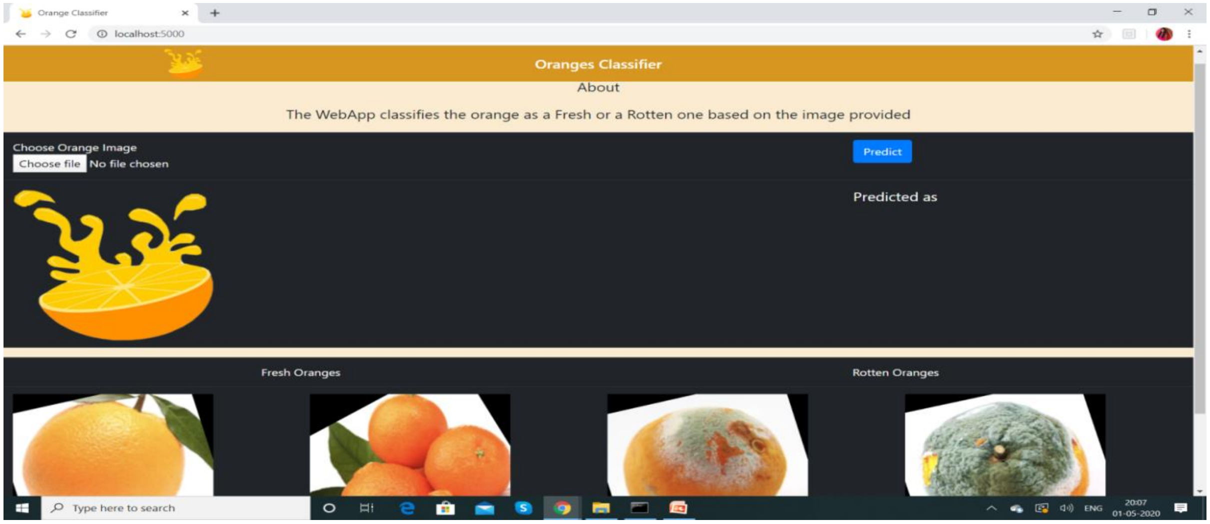


Fig 2. UI Prototype- Screenshot

C. **HARDWARE SPECIFICATIONS:**

This project was executed on Intel® Core™ i7-8550U CPU@1.8GHz, 1.99 GHz., Installed Memory (RAM) of 12GB, 64-bit Operating System, x64-based processor.

D. ARCHITECTURE AND TECHNOLOGY USED:

The **architecture** of the CNN model can be depicted (see Fig 3) as

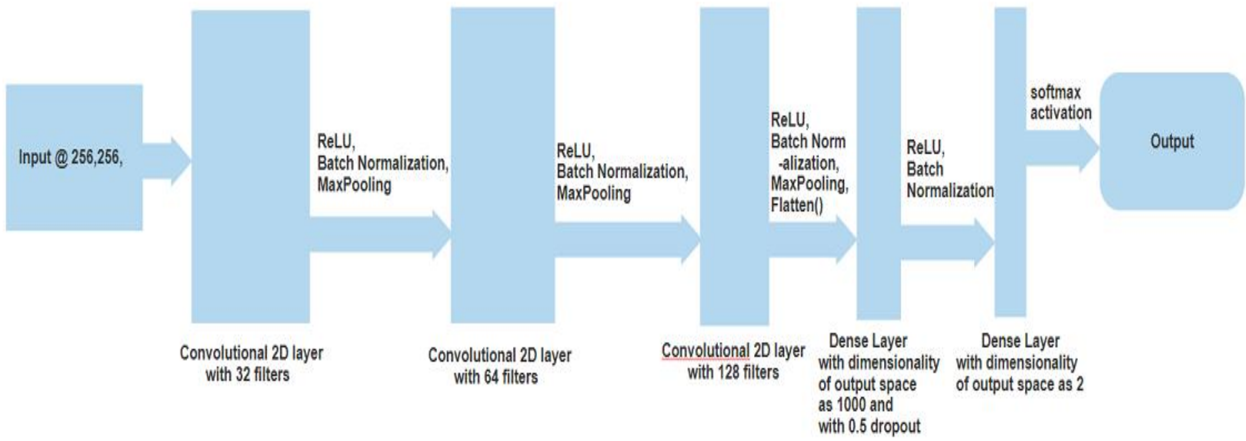


Fig 3. CNN Model Architecture

Various **technologies** are used to build the CNN model as well as User Interface .

For developing the *Model*,

- Python 3.7
- Tensorflow 1.14 version [The model was developed in Google **Colab** environment.]

For developing the *UI*,

- Python 3.7
- Flask
- HTML
- CSS
- JavaScript
- Bootstrap

E. DESIGN:

The entire flow of the project can be understood by the following diagram (see Fig. 4).

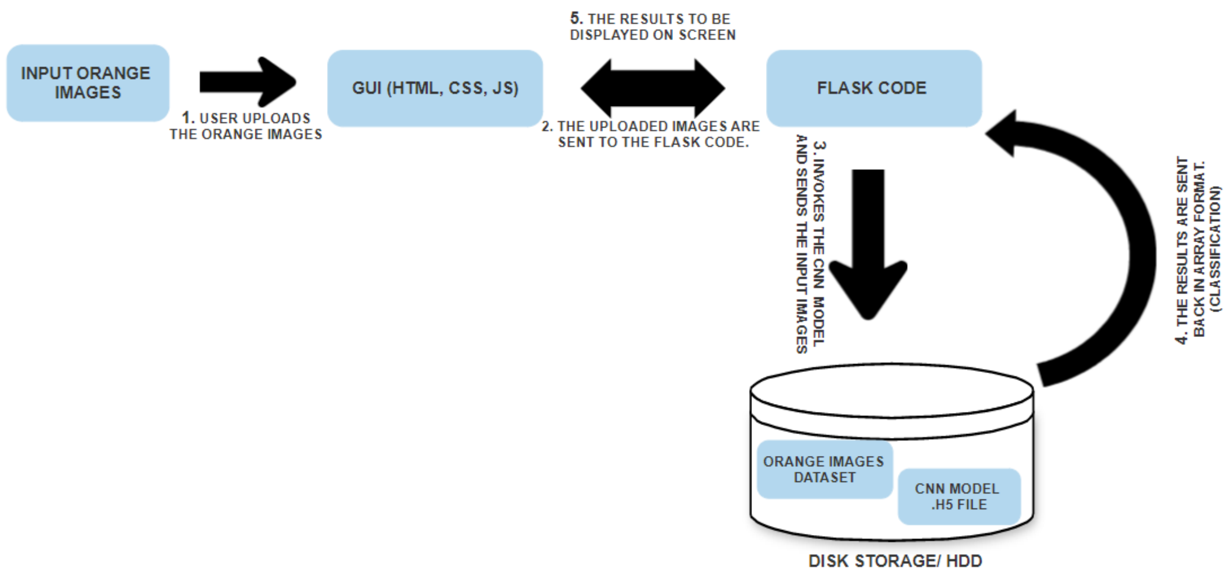


Fig 4. The flow of the project using various files and components.

The model flow can be observed as (see Fig. 5):

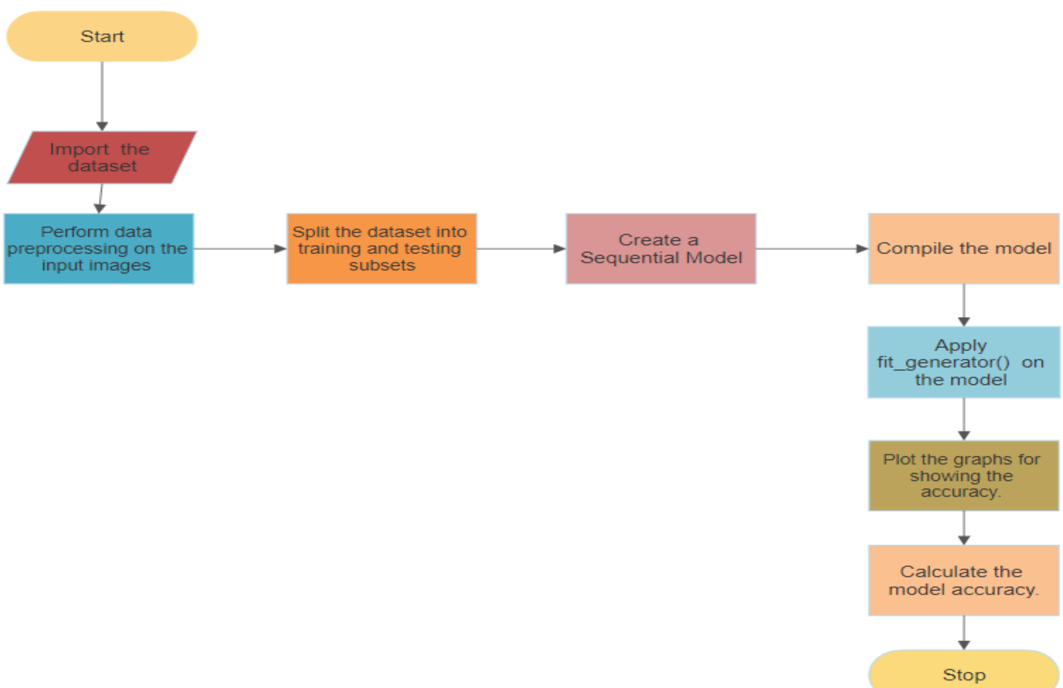


Fig 5. Flow Diagram to show the flow of the CNN model

F. IMPLEMENTATION:

- **Important Functions or Terms to understand about the CNN model:**

Data pre-processing is a technique of preparing the data by cleaning and organizing it. In the proposed method we have used several libraries/ modules like sklearn.preprocessing, keras.preprocessing to preprocess our data i.e, images. LabelBinarizer from sklearn.preprocessing which is used to convert multi-class labels to binary label i.e., belongs/doesn't belong to the class. The img_to_array method of Keras library is used to convert the image instances to NumPy array. ImageDataGenerator of keras.preprocessing.image is a great tool for data augmentation of our images and to generate batch samples for our model/ network. If the data used numbers as classes, then **to_categorical** method was used to transform these numbers to vector representation, making it suitable for our model.

The proposed model was built with the help of **Convolutional Neural Networks (CNN)**. Convolutional Neural Networks are used effectively in the areas of image recognition and classification. As the name suggests, it has a connection with biology or neuroscience. CNN's take the biological inspiration from the visual cortex. Image Classification deals with taking an image as an input and classifying it into a class or probability of classes that describe it as the best. This task is done using a series of convolutional layers by understanding the low-level features of the images like intensity, edges, curves, etc. The input image is passed through a set of convolutional, pooling and fully connected layers to get an output.

The **test_train_split** function of sklearn.model_selection is a method used to split the dataset into two disjoint subsets for training and testing purposes. The optimal test_size is 0.20 referring to the train-test split as 80%-20%. The model is built/trained using a training dataset while tested against the testing dataset.

The **loss function** used is binary cross-entropy loss, which is a default loss function to be used for binary classification. It is always intended to use with binary classification problems as the target values always belong to the set {0,1}.
The **Cross-Entropy Loss** is defined in Eq(1) as:

$$CE = -\sum_{i}^{c} ti \log(si) \tag{1}$$

Where t_i and s_i are the ground truth and the CNN score for each class $_i$ in C. As usually an activation function (Sigmoid / Softmax) is applied to the scores before the CE Loss computation, we write $f(s_i)$ to refer to the activations.
In a binary classification problem, where C′=2, the Cross-Entropy Loss can be defined Eq(2) as:

$$CE = -\sum_{i=1}^{C'=2} ti \log(si) = -t1(logs1) - (1-t1)\log(1-s1) \qquad (2)$$

Where it's assumed that there are two classes: C_1 and C_2. t_1 [0,1] and s_1 are the ground truth and the score for C_1 and $t_2=1-t_1$ and $s_2=1-s_1$ are the ground truth and the score for C2.

The **activation function** used is "**ReLU**" – Rectified Linear Unit. It is of the non-linear activation function. The non- linear activation functions always help the model to generalize or adapt to different data. The advantage of the ReLU activation function over other activation functions is that it doesn't activate all the neurons at the same time. The function and its derivative are monotonic in nature. ReLU is the default loss function for CNN as it allows the model to learn faster and perform better. Mathematically, ReLU is defined as: **y = max(0,x).** The Softmax function makes the outputs of each unit to be between 0 and 1. Also, it divides each output such that the total sum of the outputs is equal to 1. Mathematically, it is represented as in Eq(3)

$$\sigma(Z)j = e^{z}j\sigma(z_j) = \frac{e^{z}j}{\Sigma_{k=1}^{k} e^{zk}} \qquad (3)$$

The Optimization used here is **Adam optimization**. It is used to iteratively update the network weights in training data. It is used instead of the classical stochastic gradient descent algorithm. Adam's algorithm has an advantage of the Adaptive Gradient Algorithm and Root Mean Square Propagation. The Adaptive Gradient Algorithm maintains multiple learning rates per parameter to enhance the performance**. Root Mean Square Propagation** helps to maintain per-parameter learning rates that are adapted based on the average of recent magnitudes of the gradients for the weight.

- **ALGORITHM: (for *model*)**

  Step 1: Import the dataset which contains the images of fresh and rotten oranges.
  Step 2: Perform data preprocessing on the input images using the functions img_to_array, to_categorical, LabelBinarizer.
  Step 3: Using train_test_split function, split the dataset into training and testing subsets with test_size=0.20.
  Step 4: Using the ImageDataGenerator function, which augments the images and generates batch samples to the model.
  Step 5: Create a Sequential Model.
  Step 6: Compile the model.
  Step 7: Apply fit_generator using on the model and store it in history variable/object.
  Step 8: Assign the values into acc, val_acc, loss, val_loss from history dict based on keys and initialize epochs=range (1, len(acc),+1)
  Step 9: Plot the graphs
  Step 10: Calculate the model accuracy.

- **CODE SNIPPETS OF MODEL AND *UI-MODEL INTERCONNECTIVITY:***

  Code Snippets for the model: (see Fig. 6,7,8)

```
history = model.fit_generator(
    aug.flow(x_train, y_train, batch_size=BS),
    validation_data=(x_test, y_test),
    steps_per_epoch=len(x_train) // BS,
    epochs=EPOCHS, verbose=1
    )
```

Fig 6. Code snippet of training the CNN model

```
print("[INFO] Calculating model accuracy")
scores = model.evaluate(x_test, y_test)
print("Test Accuracy: {scores[1]*100}")
```

Fig 7. Code snippet of calculating the accuracy

-

```
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
# distribution
model.compile(loss="binary_crossentropy", optimizer=opt,metrics=["accuracy"])
```

Fig 8. Code snippet of Adam optimization method.

Code Snippet for UI- CNN model interconnectivity using Flask:

The HTML, CSS(using Bootstrap), JavaScript codes are written for the User Interface. Using Flask library of Python, the model is loaded and the input images from the user are given to the model. The model is executed for the given input images and the results are obtained. Thus, obtained results are displayed on the UI. The code which connects the UI to CNN model is provided below (see Fig. 9).

```
app = Flask(__name__)
IMG_FOLDER = os.path.join('.\static\img', 'upload')
app.config['UPLOAD_FOLDER'] = IMG_FOLDER


@app.route('/')
def home():
    # ..\\ ->outside the template folder
    return render_template('./index.html', orange_image="..\static\img\orange logo.png")


@app.route('/predictOrange', methods=['POST'])
def predictOrange():
    model = load_model('./model.h5')
    if request.method=='POST':
        if request.files:
            file = request.files["imgfile"]
            print(file)
            if file.filename == '':
                flash('No selected file')
                return redirect(request.url)
            if file:
                filename = secure_filename(file.filename)
                file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))
            filename = os.path.join(app.config['UPLOAD_FOLDER'], filename)
            print(file, filename)
            img = image.load_img(filename, target_size=(256, 256))
            data = image.img_to_array(img)
            data = data/255.0 #normalizing
            data = data.reshape(1, 256, 256, 3)
            #print(data)
            #print(data.shape)
            result = model.predict(data)
            #print(result)
            result = result[0]
            ans = 'Fresh Orange'
            if(result[1]==1):
                ans = 'Rotten Orange'



    return render_template('index.html', orange_image=filename, load_answer=ans)
if __name__ == "__main__":
    app.run(debug=True, host='0.0.0.0')
```

Fig 9. Code Snippet displaying interconnectivity between CNN model and the GUI

Code Snippets of UI: (see Fig. 10,11)

```html
<body>
 <div>
  <nav class="navbar navbar-expand-sm bg-secondary fixed-top" id="navBar">
    <div class="container">
      <ul class="navbar-nav">
        <li class="nav-item">
       <a class="navbar-brand" href="#">
         <img src="../static/img/orange logo.png" width="50" height="50" alt="IconNotAvailable">
       </a>
       <h1 class="text">Oranges Classifier</h1>
    </div>
  </nav>
 </div><br><br><br>
 <div class="container">
  <div><p class="centerIt">About</p></div>
  <p class="centerIt">The WebApp classifies the orange as a Fresh or a Rotten one based on the image provided</p>
 </div>
 <div class="form-group">
  <table class="table table-dark">
  <form action="{{ url_for('predictOrange')}}" method="POST" class="form-group" enctype="multipart/form-data">
    <tr>
     <td>
      Choose Orange Image
      <input type="file" class="form-control-file" name="imgfile" id="imgfile" onchange="loadFile(event)">
     </td>
     <td>
      <button type="submit" class="btn btn-primary">Predict</button>
     </td>
    </tr>
    <tr>
     <td>
      <img src="{{orange_image}}"  id='output' width=256 height=256>
     </td>
     <td>
      <p>Predicted as {{load_answer}}</p>
     </td>
    </tr>
  </form>
 </table>
 </div>
```

Fig 10. HTML Code snippet

```
<script>
var loadFile = function(event) {

   var output = document.getElementById('output');
   output.src = URL.createObjectURL(event.target.files[0]);
   output.onload = function() {
     URL.revokeObjectURL(output.src) // free memory
   }
 };
</script>
```

Fig 11. JavaScript Code snippet

- **GITHUB LINK:**

https://github.com/Yerram-Sai-Rachana/CNN_Orange_Classification

G. ERROR ANALYSIS

Using various values produced during model.fit_generator() like the accuracy of the training, accuracy of the validation, loss during training and loss during validation, we can plot various graphs to understand the model (see Fig. 12).

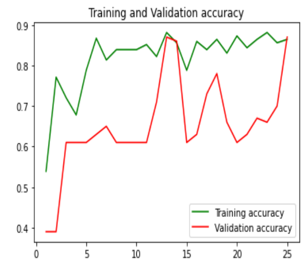
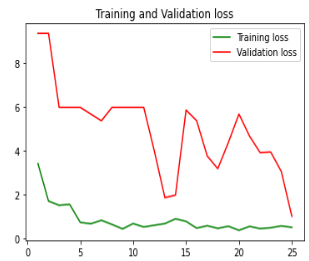
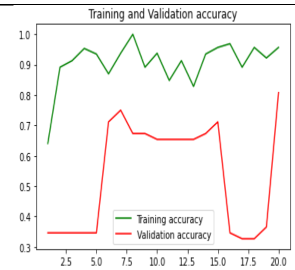
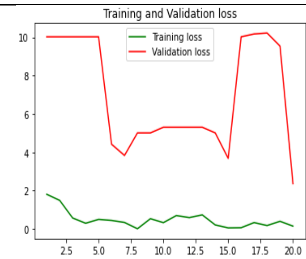
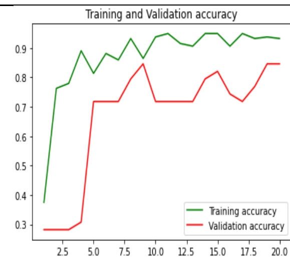
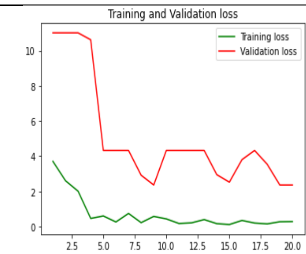
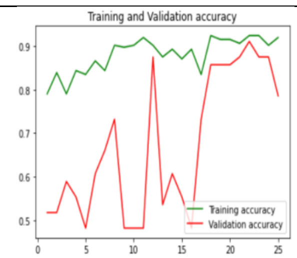
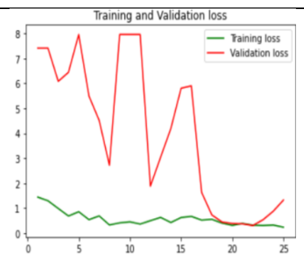
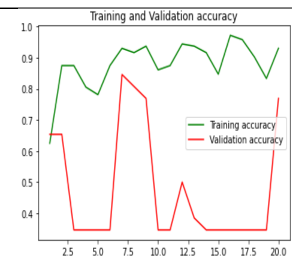
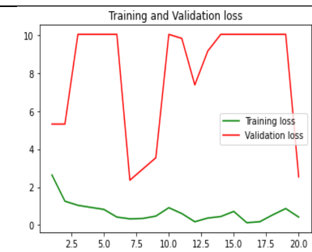
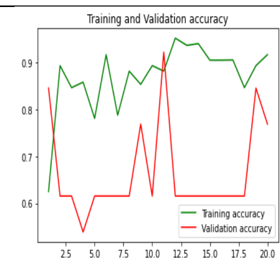
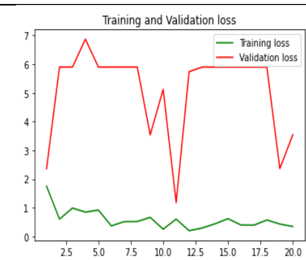
TRAIN -TEST SPLIT (in %)	NUMBER OF EPOCHS = 25		NUMBER OF EPOCHS = 20	
	Plot to show validation-accuracy	Plot to show validation-loss	Plot to show validation-accuracy	Plot to show validation-loss
60-40	 Accuracy= 87.00%		 Accuracy= 84.76%	
70-30	 Accuracy= 80.23%		 Accuracy= 84.61%	
80-20	 Accuracy= 78.57%		 Accuracy=76.92%	
90-10	 Accuracy= 86.42%		 Accuracy=84.42%	

Fig 12. obtained training and validation accuracies, training and validation loss plotted as graphs

As a part of analysis of the model, we can compare various accuracies obtained with different model architectures based on CNN, SVM for different fruits. (see Table 1)

Table 1: Comparison of accuracy obtained with different model architectures based on CNN, SVM for different fruits.

Type of the Fruit	Model Name	Methods Used for Classification	Accuracy
Banana[12]	CNN	3-convolutional layers, fully connected layer, ReLU (activation function), Max Pooling layer	80%
Grapes[13]	CNN	Color Feature Extraction(RGB, HSV), Morphological features such as shape	79.49%
Tomatoes[14]	ResNetv2 based CNN	Batch Normalization, ReLU (activation function)	87.27%
Tomatoes[14]	Convolutional Autoencoder	The encoder is used to generate internal representation of input images, The decoder converts those internal representations into outputs.	79.09%
Orange[6]	SVM Classifier	Gabor+LBP+GCH	61.29%
Orange[6]	SVM Classifier	Gabor+CLBP+LTP	64.52%
Orange[6]	SVM Classifier	ColorMoment+GLCM+Shape	67.74%

From the above table, we can observe that classification of Oranges using SVM resulted in the accuracy that falls in the range from 61.29% - 67.74%. Whereas, when we try to classify the oranges using CNN, we obtained the accuracy of 78.57%. As we change the ratio of train to test split, the accuracy might vary a bit, with extensive experimentation, we have noted that accuracy always lies above 70%.

4 RESULTS

The prime motive of the project was to develop the CNN model to classify the fruits to fresh and rotten categories by extracting the image features. The CNN model was successfully built and was classifying the Oranges into Fresh and Rotten Categories with an accuracy of 78.57%.

The following table gives the analysis of our proposed model (see Table 2).

Table 2: Summary of the results obtained

Model	Size of dataset	Train-Test Split	Accuracy in %	
			#epochs-20	#epochs-25
CNN with softmax classifier	Orange Fruit Dataset (rotten, healthy) 800 images	80-20	76.92	78.57

The model was compiled with two different number of epochs as 20 and 25. When compiled with train and test split of 80-20 we achieved a highest accuracy of 78.57%.

The following UI screenshot shows the corresponding output when we provide *fresh orange* image as input (see Fig. 13).

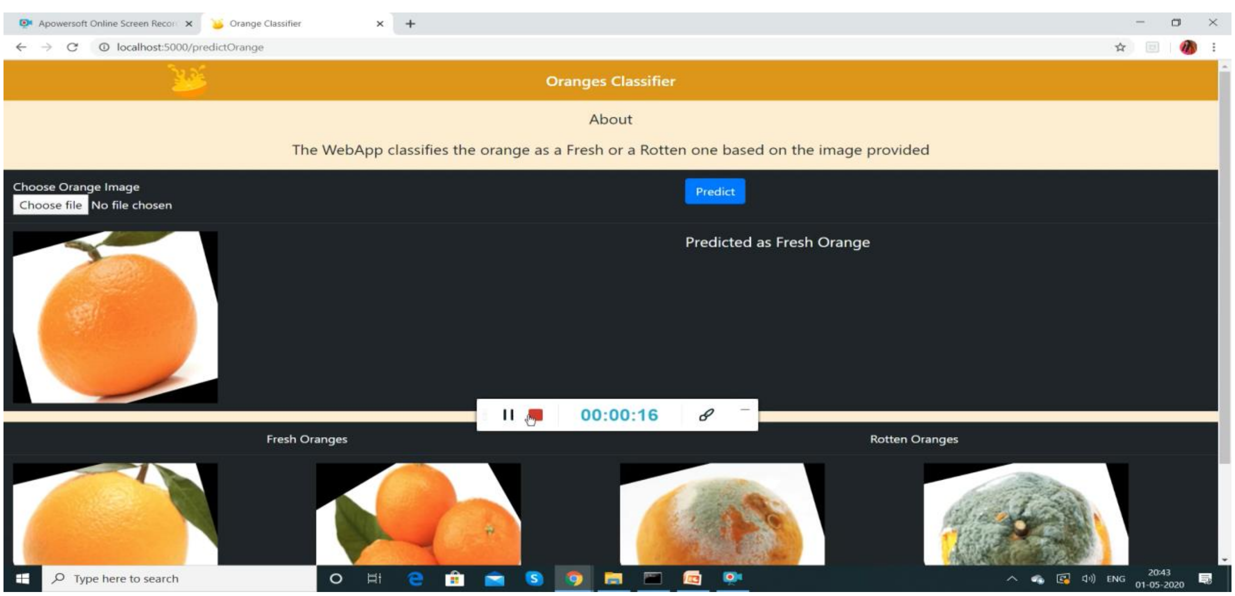


Fig 13. UI Screenshot showing Fresh Orange as output

The following UI screenshot shows the corresponding output when we provide *rotten orange* image as input (see Fig. 14).

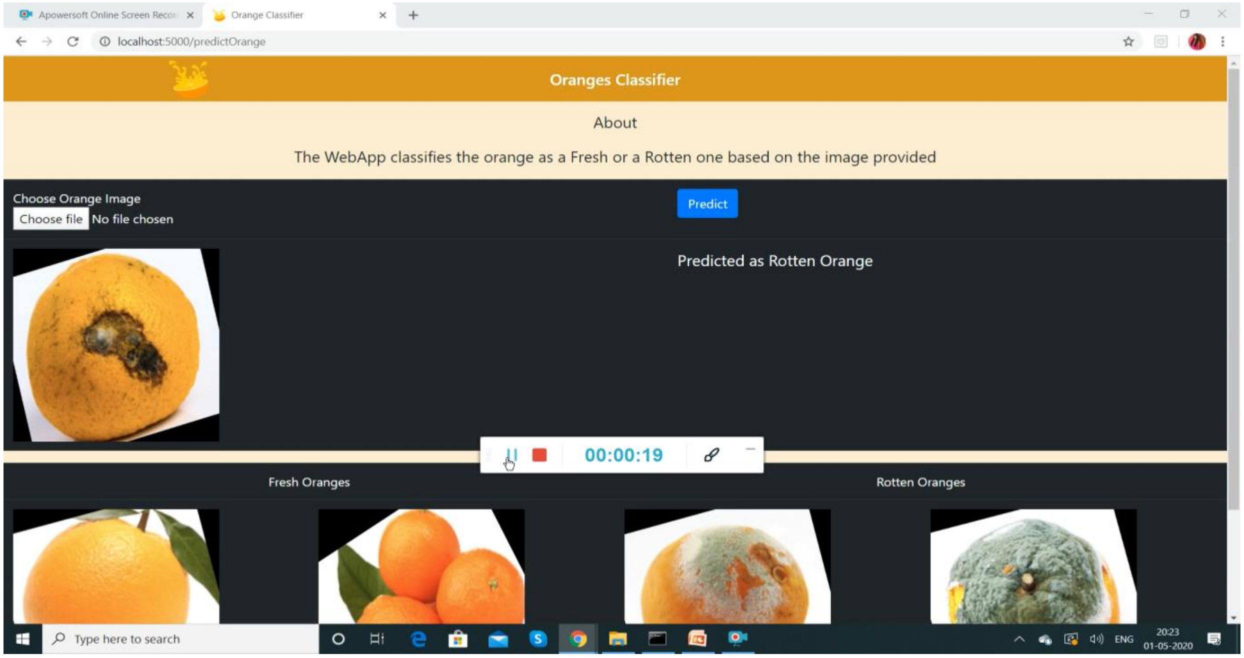


Fig 14. UI Screenshot showing Rotten Orange as output

5 CONCLUSION/ FUTURE SCOPE

When coming to the conclusion, we can see that there is more research work going on in the area of agriculture and this is a good sign of increasing the countries' economy. Integrating the new technologies and assessing the agricultural crops is a great way of moving forward in the sector. There are many things which we eat in a daily which are not assessed with automation and just assessed with human vision. Our method provides the solution to it.

In the proposed method, we have successfully classified the rotten oranges from the rest and robustly analyzed the same by specifying the accuracy of the work done. The methods which are used here are flexible and potent. The work can also be used to improvise the accuracy by combining with other classification methods. This method can also be applied for the automatic prediction by improvising a bit, which is a great use for the agricultural field to yield better crop in the current era. However, classifying and identifying the defects of the orange fruit by implementing this model in real-time can be done in the future.

The scalability of the model must be increased and the model must be made to give sustained and correct results. The accuracy of the model might be increased by using Spatial Transformer Network. So, the model with highest accuracy can be used widespread.

The model can be deployed in real-time in agricultural sector for reducing the workload of the human graders and improving the efficiency of the work done. The results obtained through model can be more accurate than compared to the work done by human graders. The future scope of this project is to be deployed in agricultural area to improve the sales, reduce the human workload, maximize the productivity.

6 REFERENCES

1. M. Recce, J. Taylor, A. Piebe and G. Tropiano, "High speed vision-based quality grading of oranges," Proceedings of International Workshop on Neural Networks for Identification, Control, Robotics and Signal/Image Processing, Venice, Italy, 1996, pp. 136-144, doi: 10.1109/NICRSP.1996.542754.
2. M. R. Satpute and S. M. Jagdale, "Automatic fruit quality inspection system," 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, 2016, pp. 1-4, doi: 10.1109/INVENTIVE.2016.7823207.
3. T. Najeeb and M. Safar, "Dates Maturity Status and Classification Using Image Processing," 2018 International Conference on Computing Sciences and Engineering (ICCSE), Kuwait City, 2018, pp. 1-6, doi: 10.1109/ICCSE1.2018.8374209.
4. C. Kumar, S. Chauhan, R. N. Alla and H. Mounicagurram, "Classifications of citrus fruit using image processing -GLCM parameters," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, 2015, pp. 1743-1747, doi: 10.1109/ICCSP.2015.7322820.
5. Y. Chen, J. Wu and M. Cui, "Automatic Classification and Detection of Oranges Based on Computer Vision," 2018 IEEE 4th International Conference on Computer and Communications (ICCC), Chengdu, China, 2018, pp. 1551-1556, doi: 10.1109/CompComm.2018.8780680.
6. H. Patel, R. Prajapati and M. Patel, "Detection of Quality in Orange Fruit Image using SVM Classifier," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2019, pp. 74-78, doi: 10.1109/ICOEI.2019.8862758.
7. A. M. Abdelsalam and M. S. Sayed, "Real-time defects detection system for orange citrus fruits using multi-spectral imaging," 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS), Abu Dhabi, 2016, pp. 1-4, doi: 10.1109/MWSCAS.2016.7869956.
8. B. J. Samajpati and S. D. Degadwala, "Hybrid approach for apple fruit diseases detection and classification using random forest classifier," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, 2016, pp. 1015-1019, doi: 10.1109/ICCSP.2016.7754302.
9. Naeem Sattar, Sheikh Ziauddin, Sajida Kalsoom, Ahmad R. Shahid, RafiUllah, Amir H. Dar,“An Orange Sorting Technique based on Size andExternalDefects”,April 2019,DOI: 10.1109/ICOEI.2019.8862758

10. Nur Badariah Ahmad Mustafa, Syed Khaleel Ahmed, Zaipatimah Ali, Wong Bing Yit, AidilAzwinZainulAbidin and ZainulAbidin Md Sharrif, "Agricultural produce Sorting and Grading using Support Vector Machines and Fuzzy Logic," 2009 IEEE International Conference on Signal and Image Processing Applications, Kuala Lumpur, 2009, pp. 391-396, doi: 10.1109/ICSIPA.2009.5478684.
11. Hussain Hassan, N.M., Nashat, A.A. New effective techniques for automatic detection and classification of external olive fruits defects based on image processing techniques. Multidim Syst Sign Process 30, 571–589 (2019). https://doi.org/10.1007/s11045-018-0573-5
12. Sri, M. Kusuma and Saikrishna, K. and Kumar, V. Vinay, Classification of Ripening of Banana Fruit Using Convolutional Neural Networks (February 21, 2020). Available at SSRN: https://ssrn.com/abstract=3558355 or http://dx.doi.org/10.2139/ssrn.3558355
13. K. Kangune, V. Kulkarni and P. Kosamkar, "Grapes Ripeness Estimation using Convolutional Neural network and Support Vector Machine," 2019 Global Conference for Advancement in Technology (GCAT), BANGALURU, India, 2019, pp. 1-5, doi: 10.1109/GCAT47503.2019.8978341.
14. Tran, Trung-Tin et al. "A Comparative Study of Deep CNN in Forecasting and Classifying the Macronutrient Deficiencies on Development of Tomato Plant." Applied Sciences 9 (2019): 1601.
15. http://apeda.gov.in/apedawebsite/six_head_product/FFV.htm