Bachelor of Technology

In

Computer Science and Engineering

FRUIT RECOGNIZATION USING COLOUR DETECTION ANALYSIS

Janvi Kohli(191520010)
Shruti Verma (191520021)
Vidhi Gautam (191520025)

Under the Guidance of
Mr. Ashutosh Shankhdhar

Department of Computer Engineering & Applications
Institute of Engineering & Technology

GLA University

Mathura- 281406, INDIA
May, 2021

Department of Computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O.- Chaumuhan,
Mathura – 281406

Declaration

We hereby declare that the work which is being presented in the B.Tech. Project **"Colour detection Analysis "**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my own work carried under the supervision of **Mr. Ashutosh Shankhdhar, Assistant Professor, GLA University, Mathura.**
The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

1. Janvi Kohli(191520010)
2.Shruti Verma (191520021)
3.Vidhi Gautam (191520025)

Department of Computer Engineering and Applications
GLA University, Mathura
17 km. Stone NH#2, Mathura-Delhi Road, P.O.- Chaumuhan,
Mathura – 281406

CERTIFICATE

This is to certify that the project entitled **“Color Detection Analysis”** carried out in Mini Project is a bonafide work done by **JANVI KOHLI (191520010), SHRUTI VERMA (191520021) and VIDHI GAUTAM (191520025)**is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Supervisor:
Name of Supervisor : Mr. Ashutosh Shankhdhar
Date:

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Mini Project undertaken during B. Tech. third Year. This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.

Our heartiest thanks to **Dr. (Associate Prof). Rohit Agrawal,** Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.

We owe special debt of gratitude to **Mr. Shashi Shekhar**, Program Coordinator, Department of CEA, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.

We owe special debt of gratitude to **Mr. Ashutosh Shandkdhar,** Assistant Professor Department of CEA, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

1. **JANVI KOHLI (191520010)**
2. **SHRUTI VERMA (191520021)**
3. **VIDHI GAUTAM (191520025)**

ABSTRACT

How many times has it occurred to you that even after seeing, you don't remember the name of the color? There can be 16 million colors based on the different RGB color values but we only remember a few

In this project, we are going to build an interactive app that will detect the selected color from any image. To implement this we will need a labeled data of all the known colors then we will calculate which color resembles the most with the selected color value.

The computer vision strategies used to recognize a fruit rely on four basic features which characterize the object: intensity, color, shape and texture. This paper proposes an efficient fusion of color and texture features for fruit recognition. The recognition is done by the minimum distance classifier based upon the statistical and co-occurrence features derived from the Wavelet transformed sub- bands. Experimental results on a database of about 2635 fruits from 15 different classes confirm the effectiveness of the proposed approach.

Table of Contents

Chapter 1
INTRODUCTION

The computer vision strategies used to recognize a fruit rely on four basic features which characterize the object: intensity, color, shape and texture. This paper proposes an efficient fusion of color and texture features for fruit recognition. The recognition is done by the minimum distance classifier based upon the statistical and co-occurrence features derived from the Wavelet transformed sub- bands. Recognition system has emerged as a ‘grand challenge' for computer vision, with the longer term aim of being able to achieve near human levels of recognition for tens of thousands of categories under a wide variety of conditions. The fruit recognition system can be applied for educational purpose to enhanced learning, especially for small kids and Down syndrome patients, of fruits pattern recognition based on the fruit recognition result [1]. It can be used in grocery store which makes the customers label their purchases using automatic fruit recognition based on computer vision. A number of challenges had to be overcome to enable the system to perform automatic recognition of the kind of fruit or vegetable using the images from the camera. Many kind of fruits are subject to significant variation in color and texture, depending on how ripe they are. For example, Bananas range from being uniformly green, to yellow, to patchy and brown. Color and texture are the fundamental character of natural images, and plays an important role in visual perception. Color has been a great help in identifying objects for many years. It is often useful to simplify a monochrome problem by improving contrast or separation. The process of color classification involves extraction of useful information concerning the spectral properties of object surfaces and discovering the best match from a set of known descriptions or class models to implement the recognition task . Texture is one of the most active topics in machine intelligence and pattern analysis since the 1950s which tries to discriminate different patterns of images by extracting the dependency of intensity between pixels and their neighboring pixels [3] or by obtaining the variance of intensity across pixels . Recently, different features of color and texture are combined together for their applications in the food industry .

An unified approach that can combine many features and classifiers, where all features are simply concatenated and fed independently to each classification algorithm. The fusion approach is validated using a multi-class fruit-and-vegetable categorization task in a semi-controlled environment, such as a distribution center or the supermarket cashier. The results show that the solution is able to reduce the classification error in up to 15 percentage points with respect to the baseline

Chapter 2
SOFTWARE REQUIREMENT ANALYSIS

2.1 REQUIREMENT ANALYSIS

2.1.1 SOFTWARE COMPONENTS

- Jupyter Notebook
- Any Operating System.

2.1.2 HARDWARE COMPONENTS

- Personal computers with min 8 GB RAM

2.2 MODULES AND FUNCTIONALITIES

2.2.1 KNN Algorithm

The K-nearest neighbors (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both Classification and Regression Problems.

1. Load the data

2. Initialize K to your chosen number of neighbors

3. For each example in the data

3.1 Calculate the distance between the query example and the current example from the data.

3.2 Add the distance and the index of the example to an ordered collection

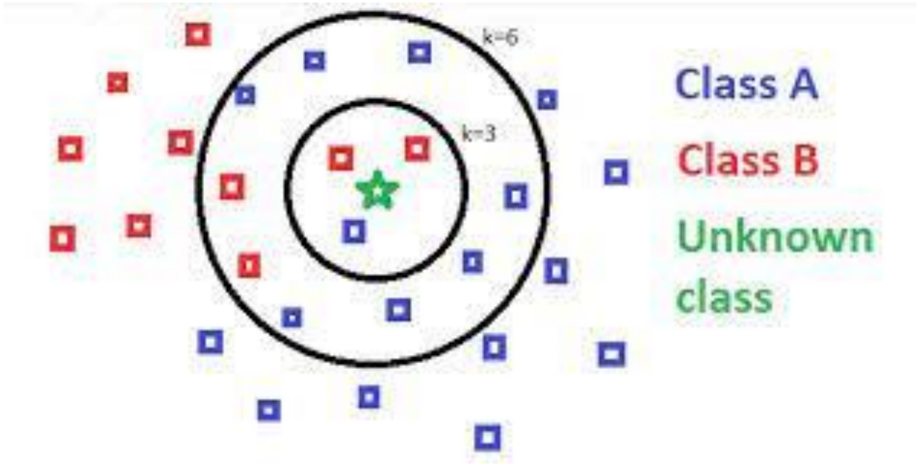
4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances

5. Pick the first K entries from the sorted collection

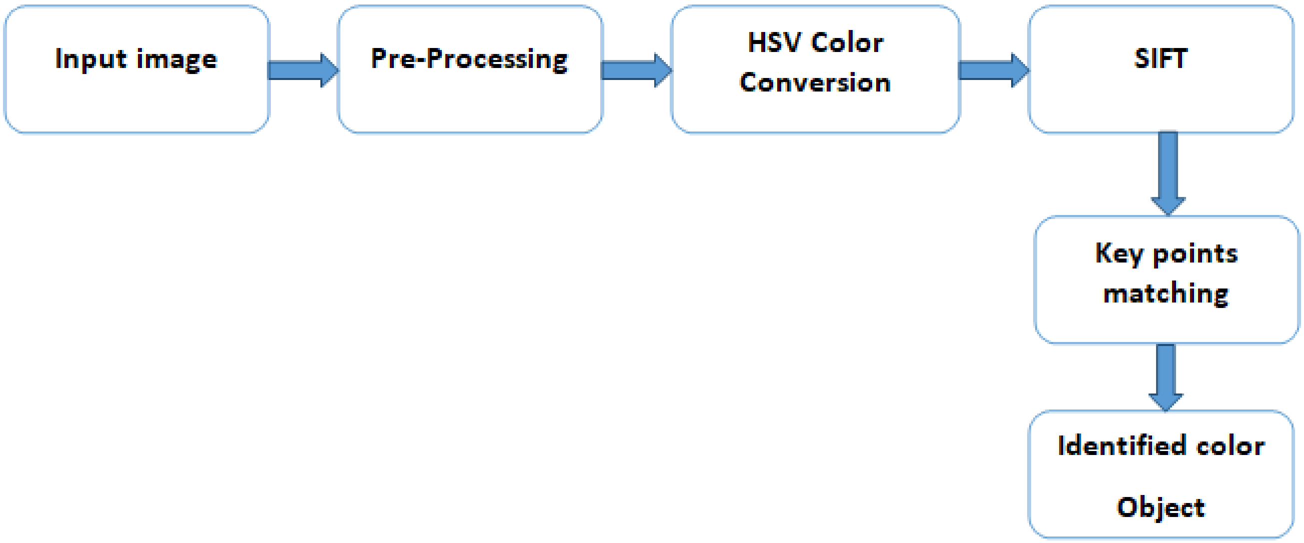
6. Get the labels of the selected K entries

7. If regression, return the mean of the K labels

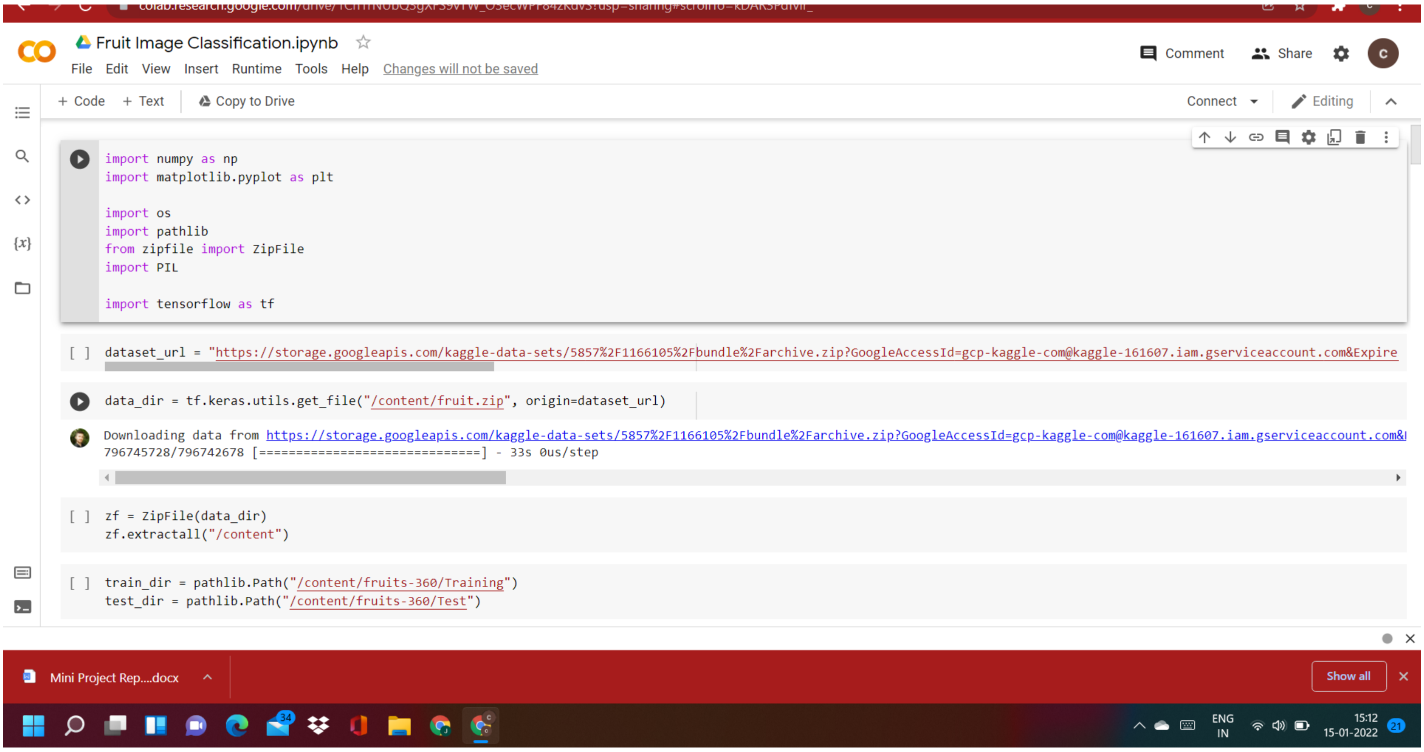
8. If classification, return the mode of the K labels



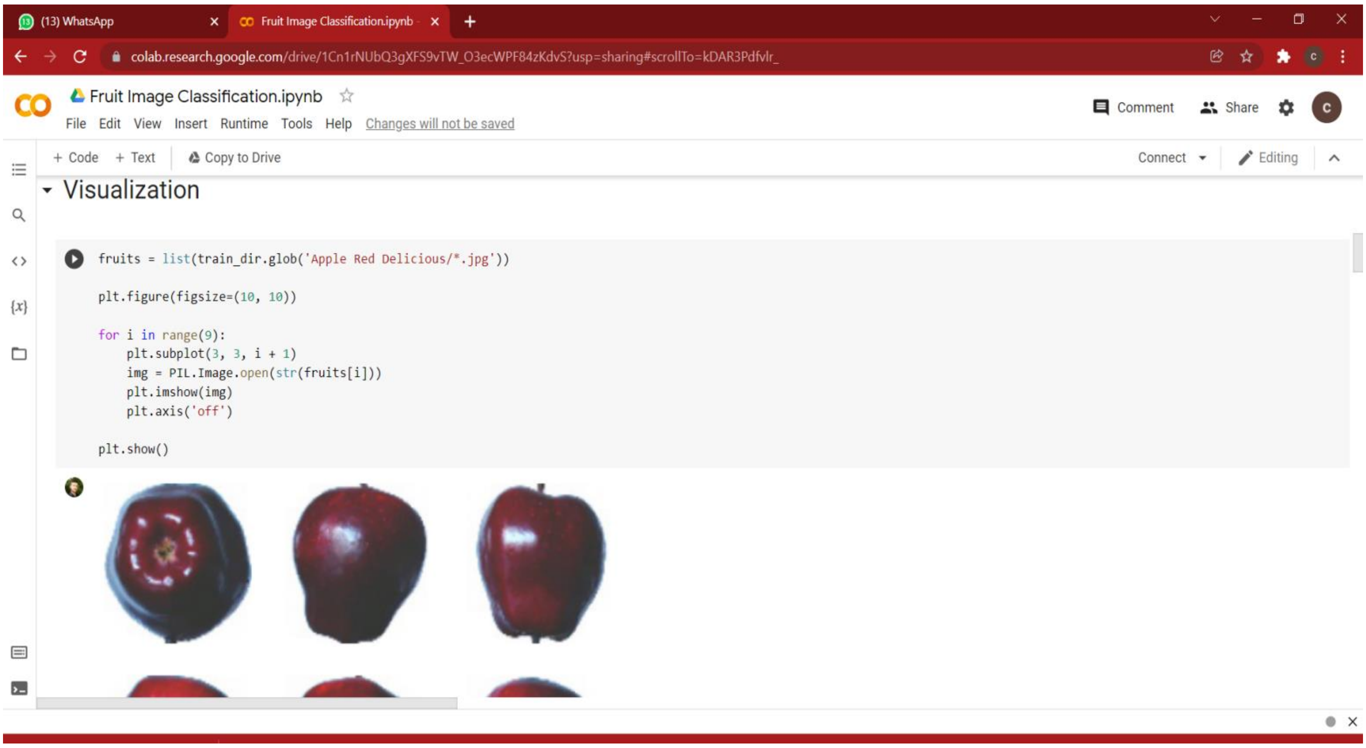
Chapter 3
SOFTWARE DESIGN



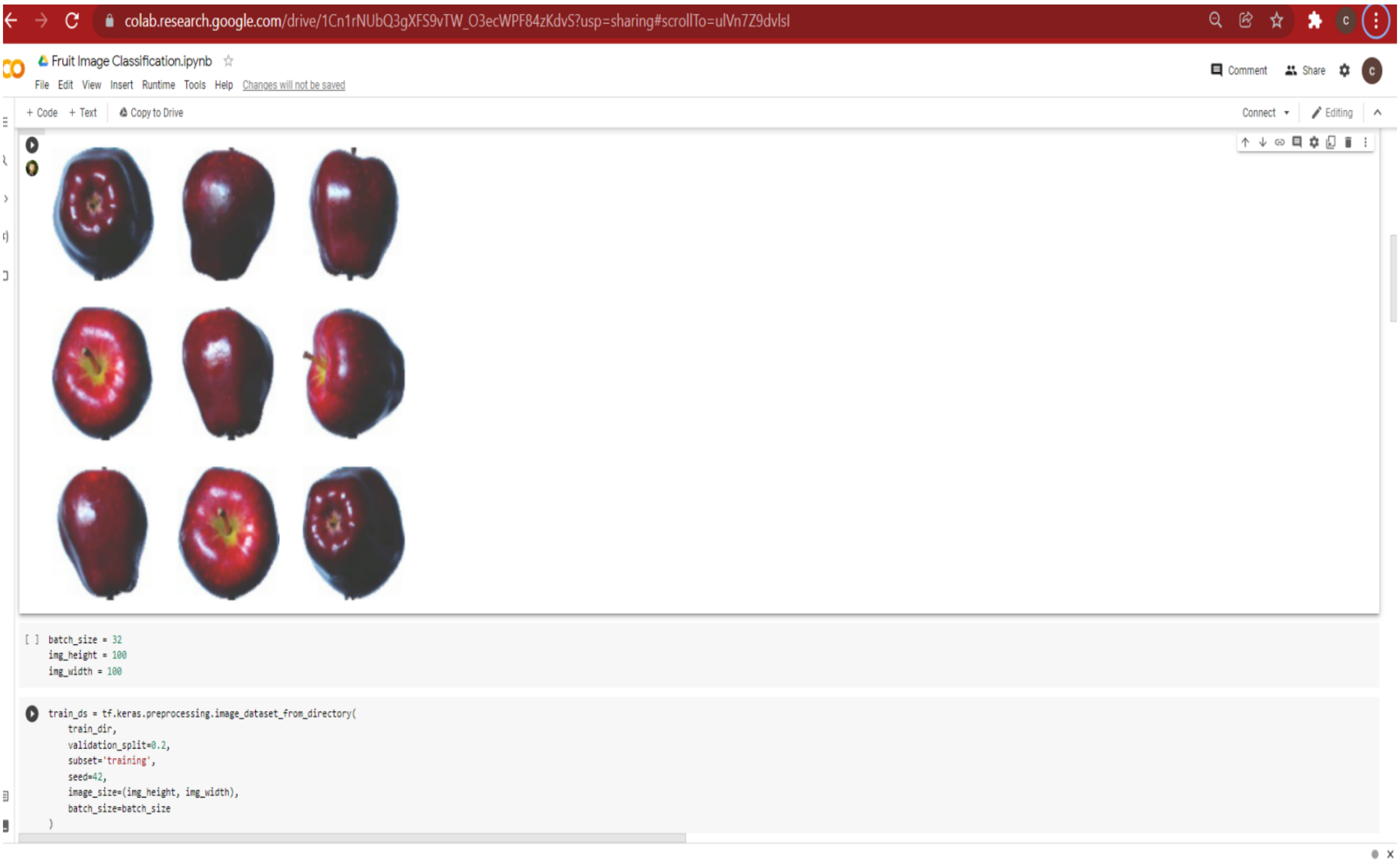
Chapter 4
IMPLEMENTATION



VISUALIZATION

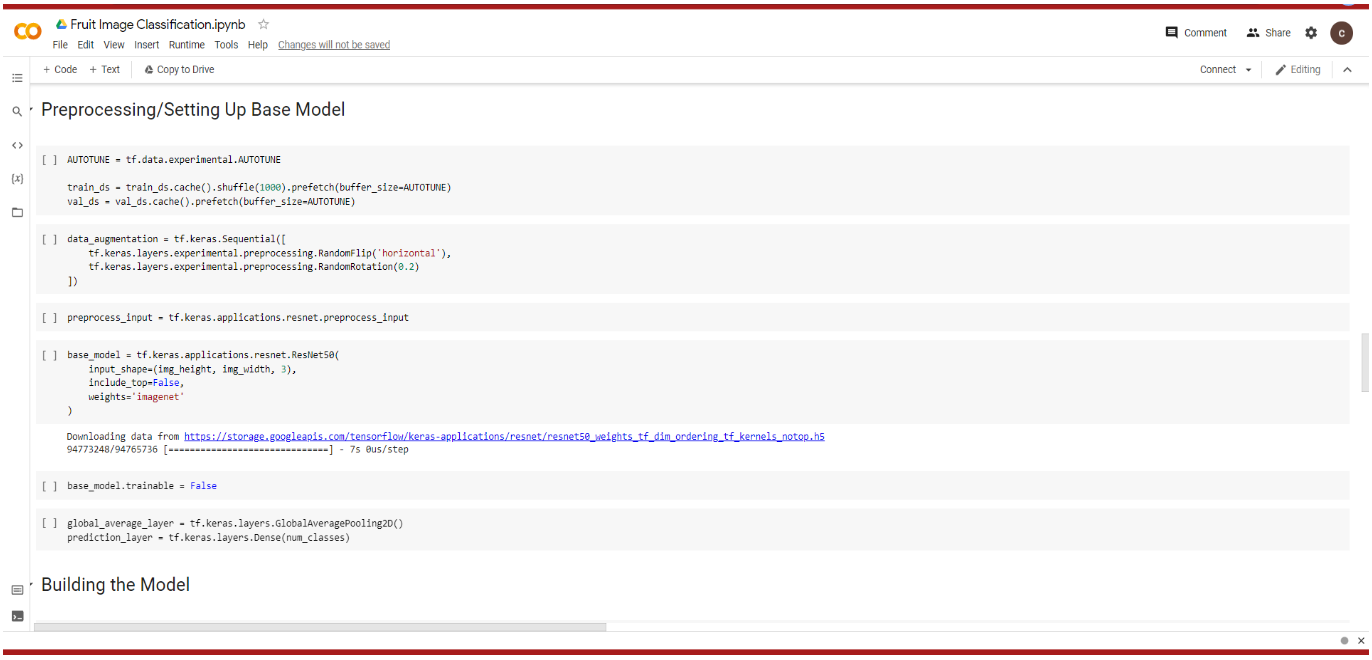


F

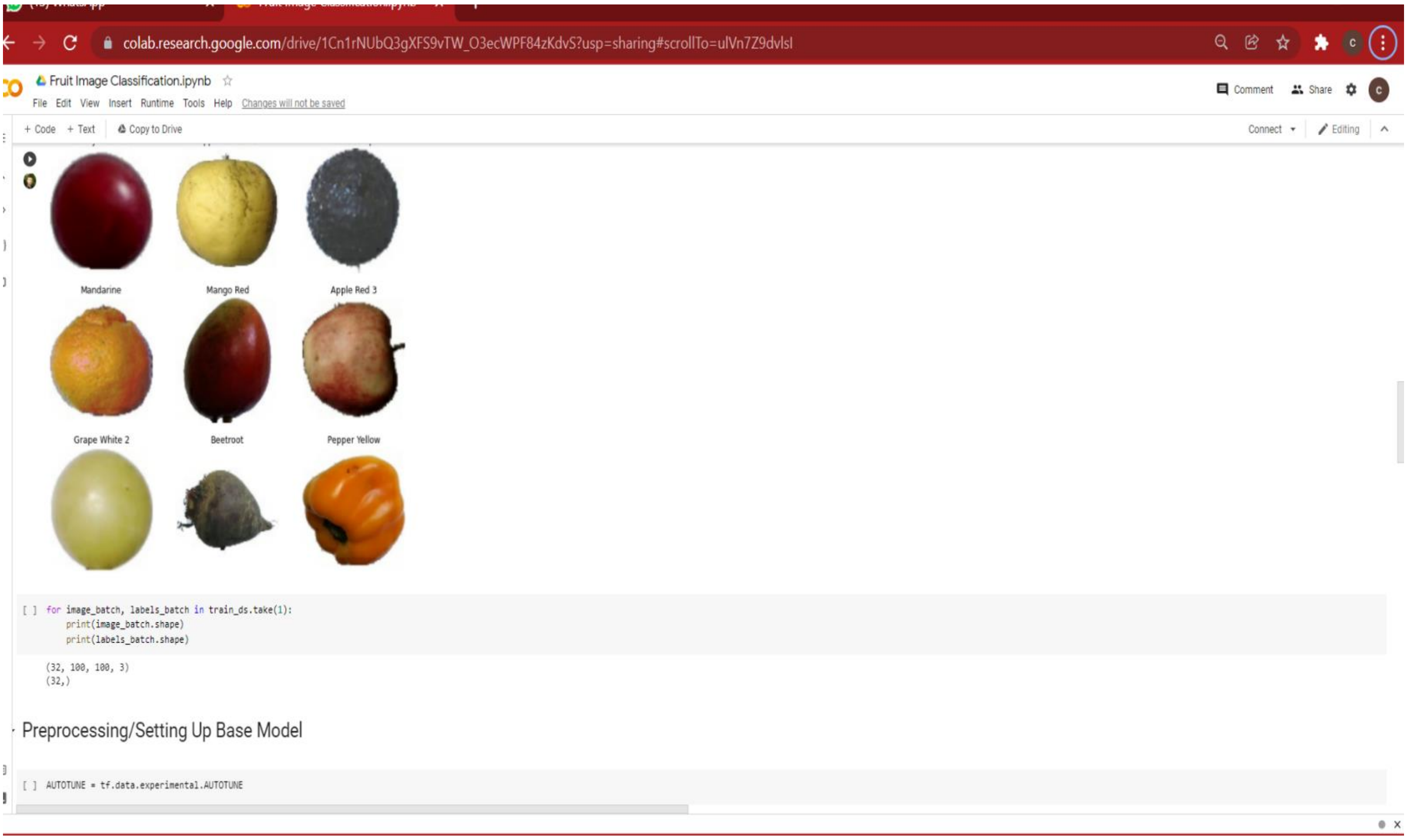

colab.research.google.com/drive/1Cn1rNUbQ3gXFS9vTW_O3ecWPF84zKdvS?usp=sharing#scrollTo=ulVn7Z9dvlsl
Fruit Image Classification.ipynb
File Edit View Insert Runtime Tools Help Changes will not be saved
Comment Share
+ Code + Text Copy to Drive
Connect Editing
batch_size = 32
img_height = 100
img_width = 100
train_ds = tf.keras.preprocessing.image_dataset_from_directory(
train_dir,
validation_split=0.2,
subset='training',
seed=42,
image_size=(img_height, img_width),
batch_size=batch_size
)

PREPROCESSING

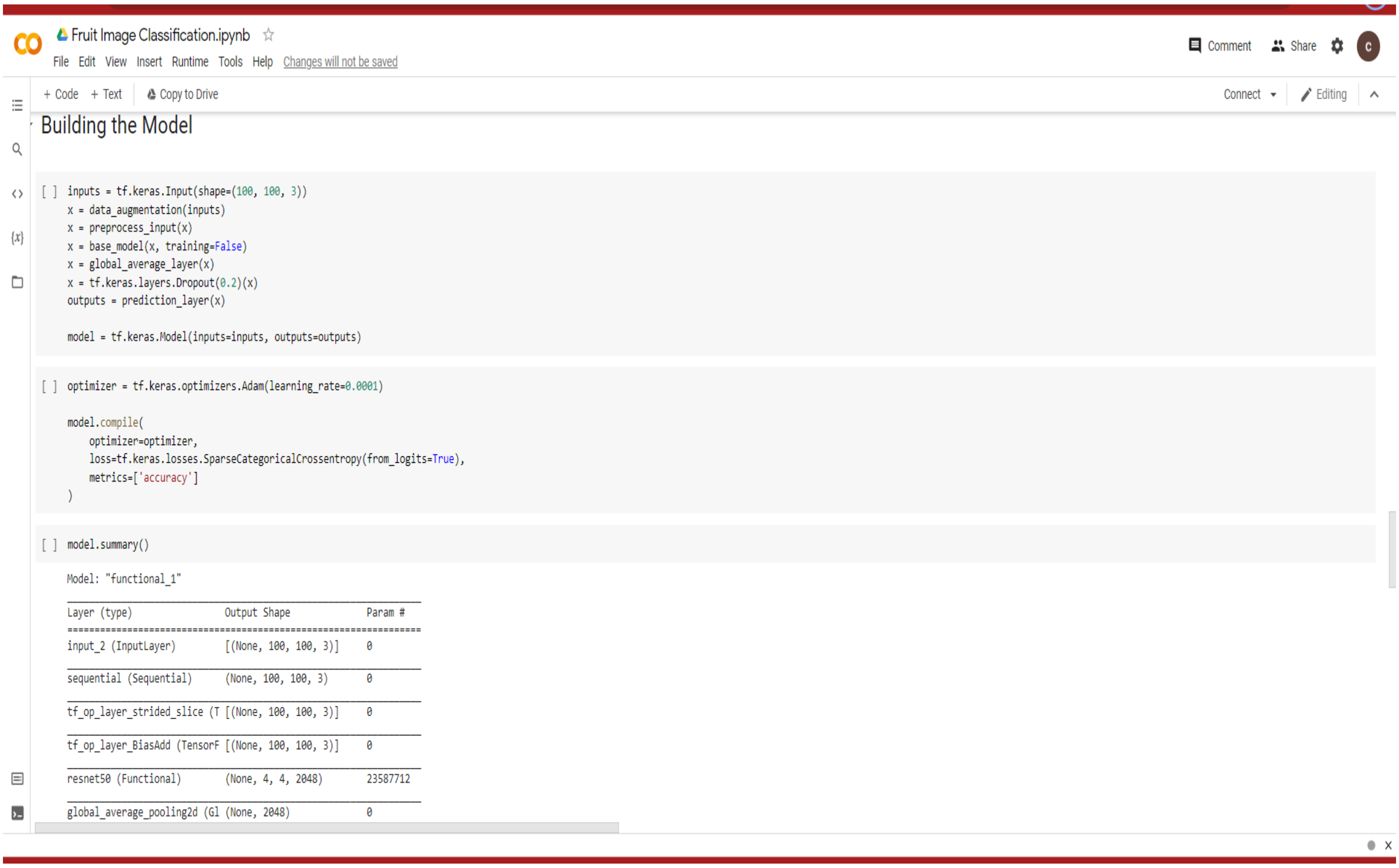
For a real application in a supermarket, it might be necessary to cope with illumination variations, sensor capturing artifacts, specular reflections, background clutter, shading, and shadows. Therefore, in order to reduce the scene complexity, it might be interesting to perform background subtraction and focus in the object's description. Background subtraction is a commonly used class of techniques for segmenting out objects of interest in a scene. The name "background subtraction" comes from the simple technique of subtracting the observed image from the estimated image and thresholding the result to generate the objects of interest. The best channel to perform the background subtraction is the S channel of HSV-stored images. This is understandable, given that the S channel is much less sensitive to lighting variations than any of the RGB color channels.



BUILDING A MODEL



BUILDING THE MODEL



Fruit Image Classification.ipynb

File Edit View Insert Runtime Tools Help Changes will not be saved

Comment Share

+ Code + Text Copy to Drive

Connect Editing

Building the Model

```
[ ] inputs = tf.keras.Input(shape=(100, 100, 3))
    x = data_augmentation(inputs)
    x = preprocess_input(x)
    x = base_model(x, training=False)
    x = global_average_layer(x)
    x = tf.keras.layers.Dropout(0.2)(x)
    outputs = prediction_layer(x)

    model = tf.keras.Model(inputs=inputs, outputs=outputs)
```

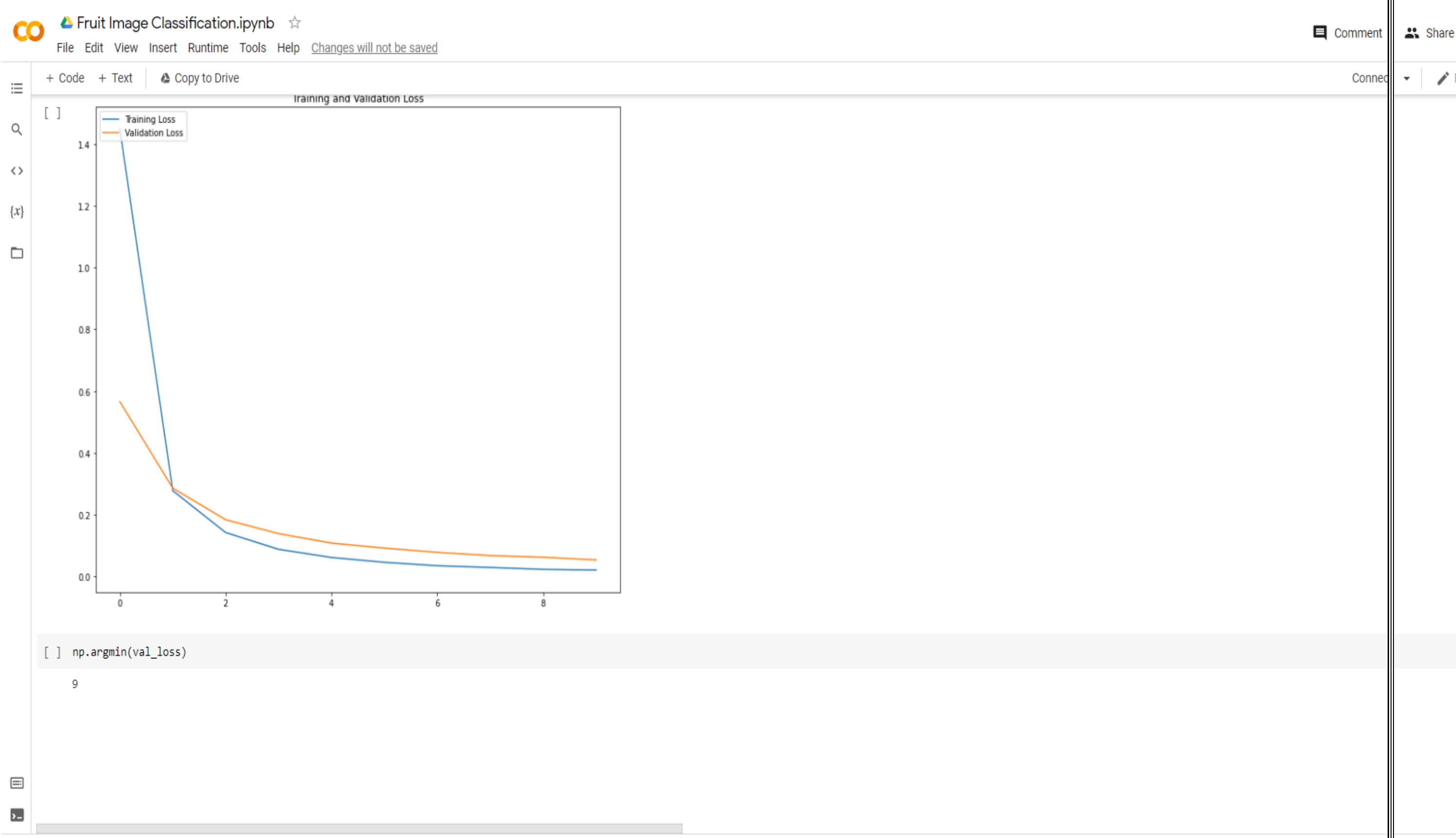
```
[ ] optimizer = tf.keras.optimizers.Adam(learning_rate=0.0001)

    model.compile(
        optimizer=optimizer,
        loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
        metrics=['accuracy']
    )
```

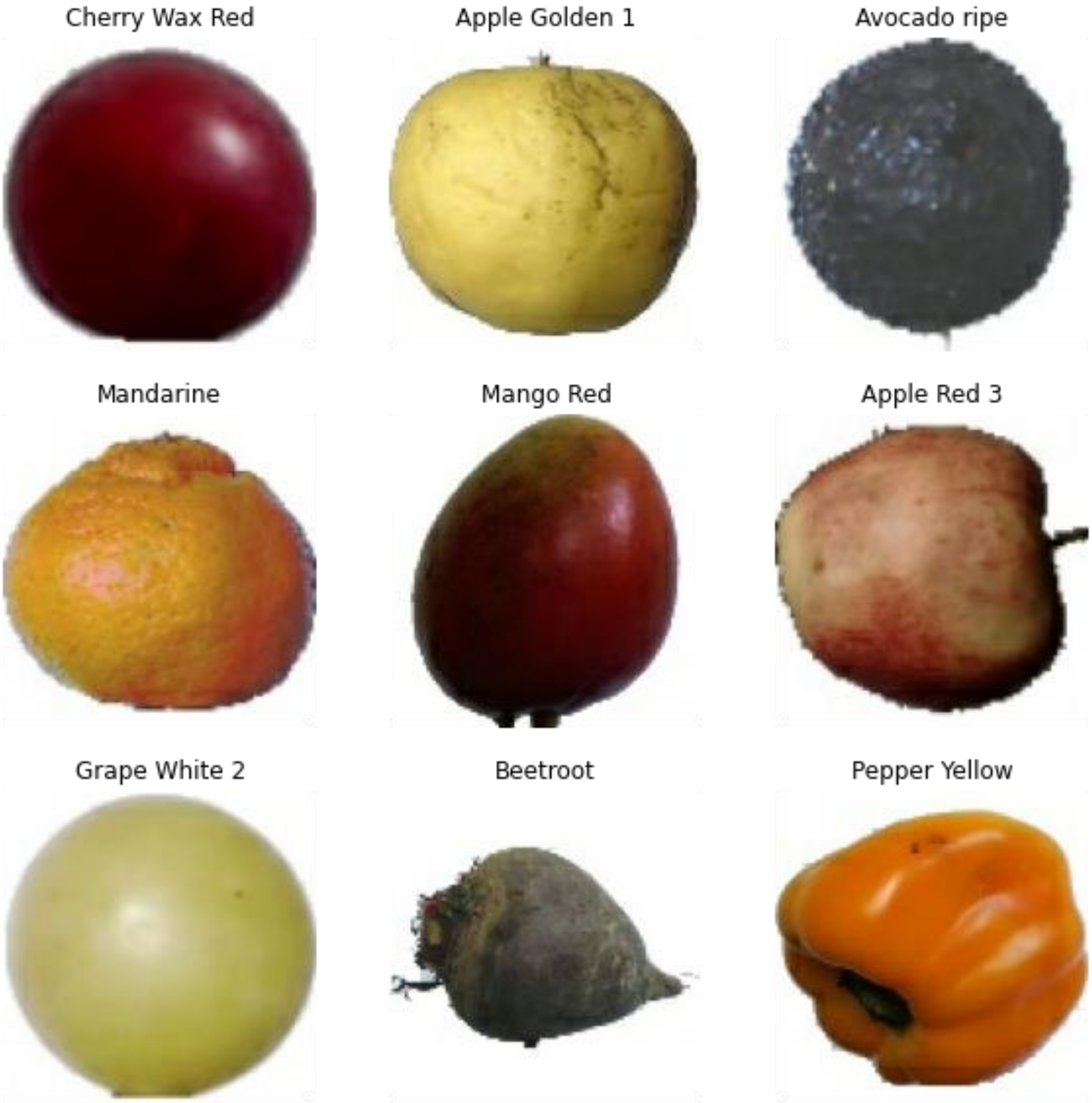
```
[ ] model.summary()

    Model: "functional_1"
    _________________________________________________________________
    Layer (type)                 Output Shape              Param #
    =================================================================
    input_2 (InputLayer)         [(None, 100, 100, 3)]     0
    _________________________________________________________________
    sequential (Sequential)      (None, 100, 100, 3)       0
    _________________________________________________________________
    tf_op_layer_strided_slice (T [(None, 100, 100, 3)]     0
    _________________________________________________________________
    tf_op_layer_BiasAdd (TensorF [(None, 100, 100, 3)]     0
    _________________________________________________________________
    resnet50 (Functional)        (None, 4, 4, 2048)        23587712
    _________________________________________________________________
    global_average_pooling2d (Gl (None, 2048)              0
```

OUTPUT



Fruit Image Classification.ipynb
File Edit View Insert Runtime Tools Help Changes will not be saved
Comment
Share
+ Code + Text
Copy to Drive
Training and Validation Loss
Training Loss
Validation Loss
np.argmin(val_loss)
9


Cherry Wax Red
Apple Golden 1
Avocado ripe
Mandarine
Mango Red
Apple Red 3
Grape White 2
Beetroot
Pepper Yellow

CONCLUSION

The use of computers to analyze images has many potential applications for automated agricultural tasks. But, the variability of the agricultural objects makes it very difficult to adapt the existing industrial algorithms to the agricultural domain. The proposed method can process, analyze and recognize fruits based on color and texture features. In order to improve the functionality and flexibility of the recognition system shape and size features can be combined together with color and texture features. Further, by increasing the number of images in the database the recognition rate can be increased. This algorithm can be used for smart self service scales.