Food detection & Calorie estimation using images

Group - Code Wars

Abhitej Thonupunoori
Manoj Kumar Madireddy
Aravind Kumar Gardas

Introduction

The main aim of the project is to build a food recognition model that predicts the food captured using the camera and also estimate its calorie, to help patients and dietitians in managing daily food intake.

Technology & Libraries

- Deep Learning – CNN
- Libraries used –
    1. Numpy, Pandas for prepocessing the data
    2. Open-CV, to read the image
    3. Keras, for training the data & visualizing the data
- Interface – Flask, Gradio

Data set

We have downloaded data from Kaggle,

https://www.kaggle.com/datasets/aelchimminut/fruits262

We have classified 6 food types from the data set, each category has 400 images uploaded to respective folders in google drive, which is mounted to google colab for project implementation.

Categories: Apple, Banana, Cherry, Pineapple, Lemon, Tomato

320 images from each category are used for training the model & 80 images from each category are used for testing the model.

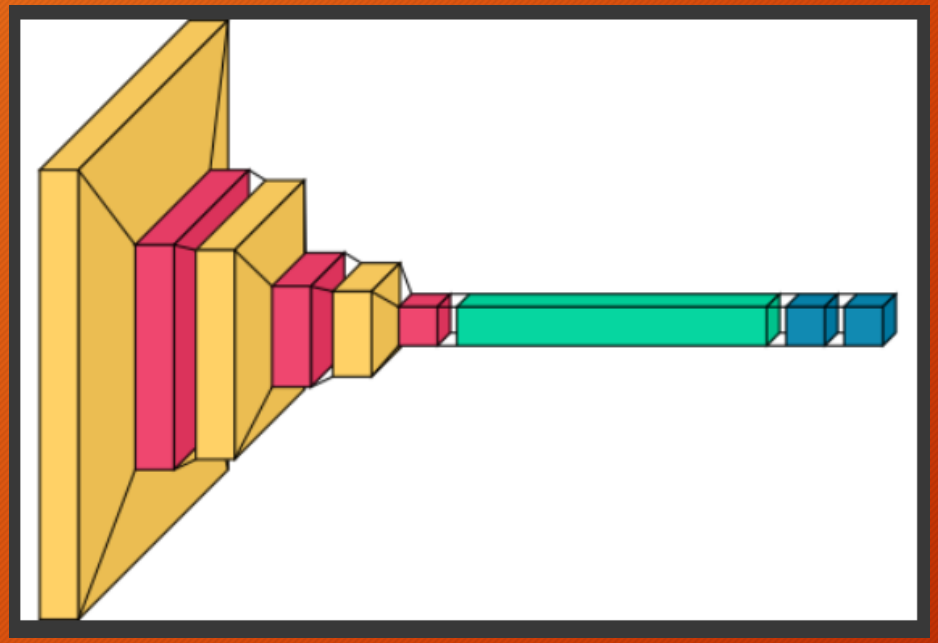
Data pre-processing

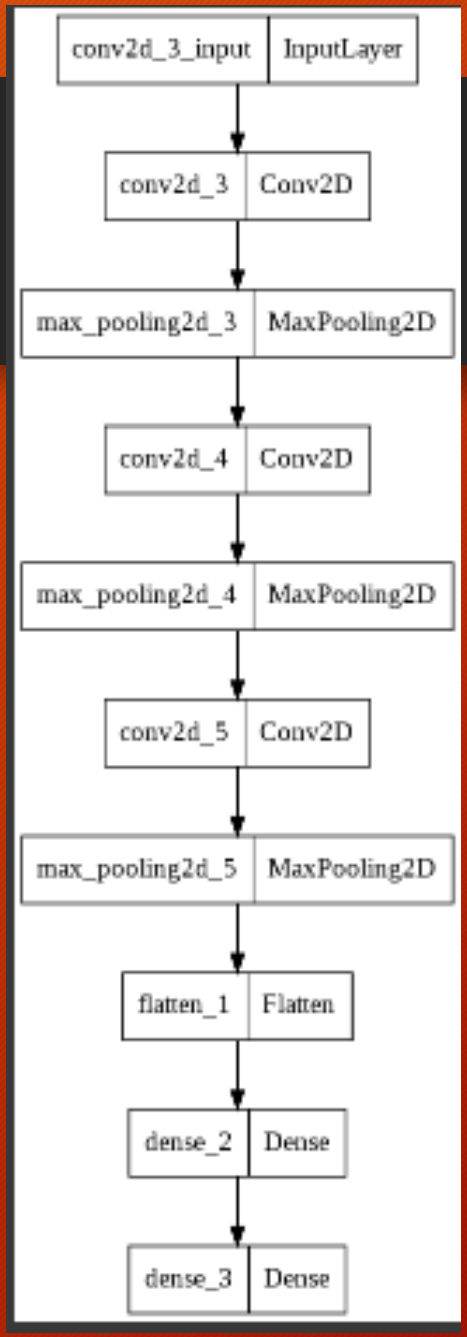
- Converting the data types
- Image reshaping
- Image scaling between values 0 and 1
- Transforming the target data to binary class matrix

Model summary

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_3 (Conv2D)           (None, 58, 58, 32)        2432

 max_pooling2d_3 (MaxPooling  (None, 29, 29, 32)       0
 2D)

 conv2d_4 (Conv2D)           (None, 27, 27, 64)        18496

 max_pooling2d_4 (MaxPooling  (None, 13, 13, 64)       0
 2D)

 conv2d_5 (Conv2D)           (None, 11, 11, 64)        36928

 max_pooling2d_5 (MaxPooling  (None, 5, 5, 64)         0
 2D)

 flatten_1 (Flatten)         (None, 1600)              0

 dense_2 (Dense)             (None, 128)               204928

 dense_3 (Dense)             (None, 6)                 774

=================================================================
Total params: 263,558
Trainable params: 263,558
Non-trainable params: 0
_________________________________________________________________
None
```

Model flow diagram

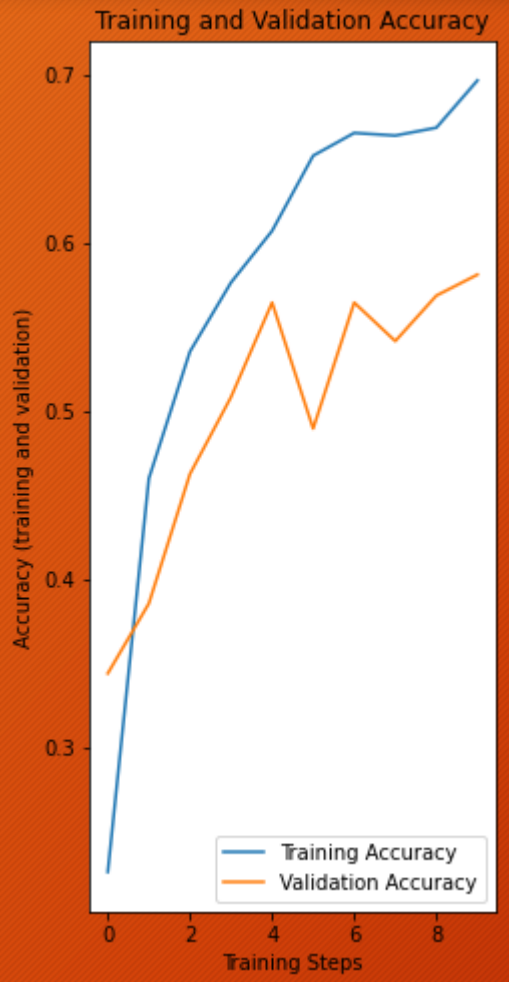


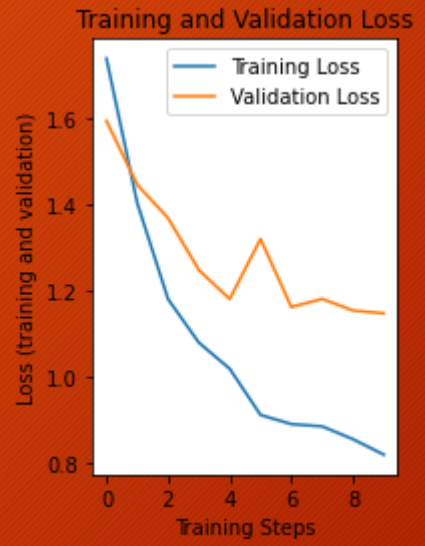


Model performance

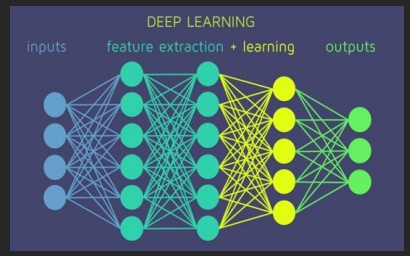
```
Epoch 1/10
15/15 [==============================] - 13s 822ms/step - loss: 1.7382 - accuracy: 0.2255 - val_loss: 1.5936 - val_accuracy: 0.3438
Epoch 2/10
15/15 [==============================] - 12s 820ms/step - loss: 1.4024 - accuracy: 0.4599 - val_loss: 1.4452 - val_accuracy: 0.3854
Epoch 3/10
15/15 [==============================] - 12s 805ms/step - loss: 1.1796 - accuracy: 0.5354 - val_loss: 1.3678 - val_accuracy: 0.4625
Epoch 4/10
15/15 [==============================] - 12s 809ms/step - loss: 1.0785 - accuracy: 0.5766 - val_loss: 1.2473 - val_accuracy: 0.5083
Epoch 5/10
15/15 [==============================] - 12s 804ms/step - loss: 1.0180 - accuracy: 0.6073 - val_loss: 1.1799 - val_accuracy: 0.5646
Epoch 6/10
15/15 [==============================] - 12s 807ms/step - loss: 0.9108 - accuracy: 0.6521 - val_loss: 1.3194 - val_accuracy: 0.4896
Epoch 7/10
15/15 [==============================] - 12s 807ms/step - loss: 0.8895 - accuracy: 0.6656 - val_loss: 1.1613 - val_accuracy: 0.5646
Epoch 8/10
15/15 [==============================] - 12s 804ms/step - loss: 0.8842 - accuracy: 0.6641 - val_loss: 1.1802 - val_accuracy: 0.5417
Epoch 9/10
15/15 [==============================] - 12s 822ms/step - loss: 0.8544 - accuracy: 0.6687 - val_loss: 1.1532 - val_accuracy: 0.5688
Epoch 10/10
15/15 [==============================] - 12s 820ms/step - loss: 0.8186 - accuracy: 0.6969 - val_loss: 1.1467 - val_accuracy: 0.5813
```

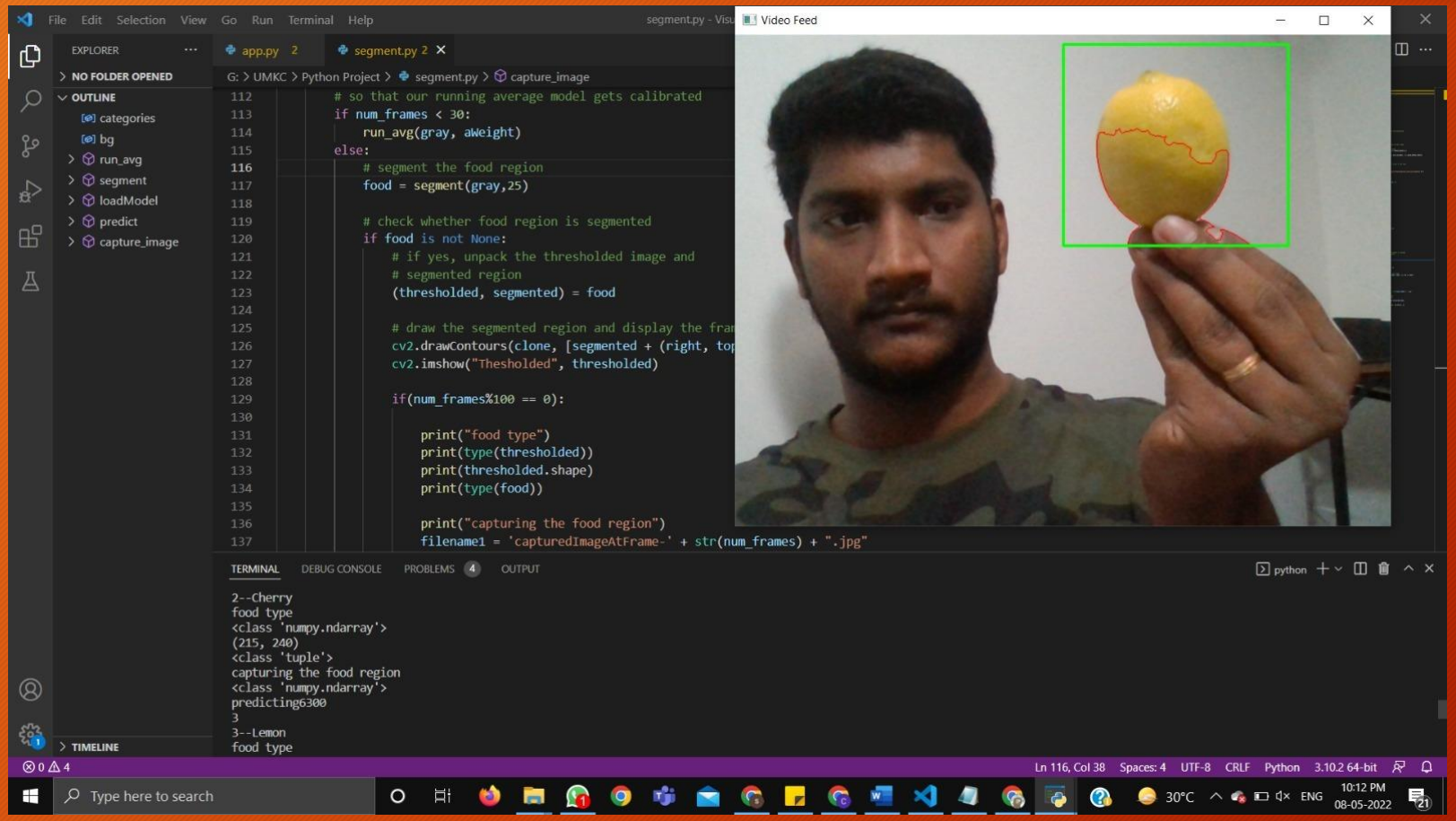
Accuracy & Loss



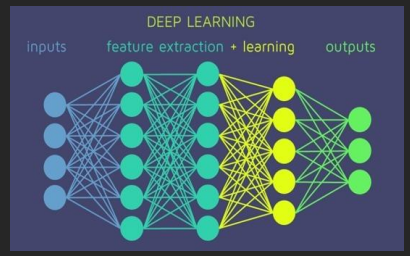


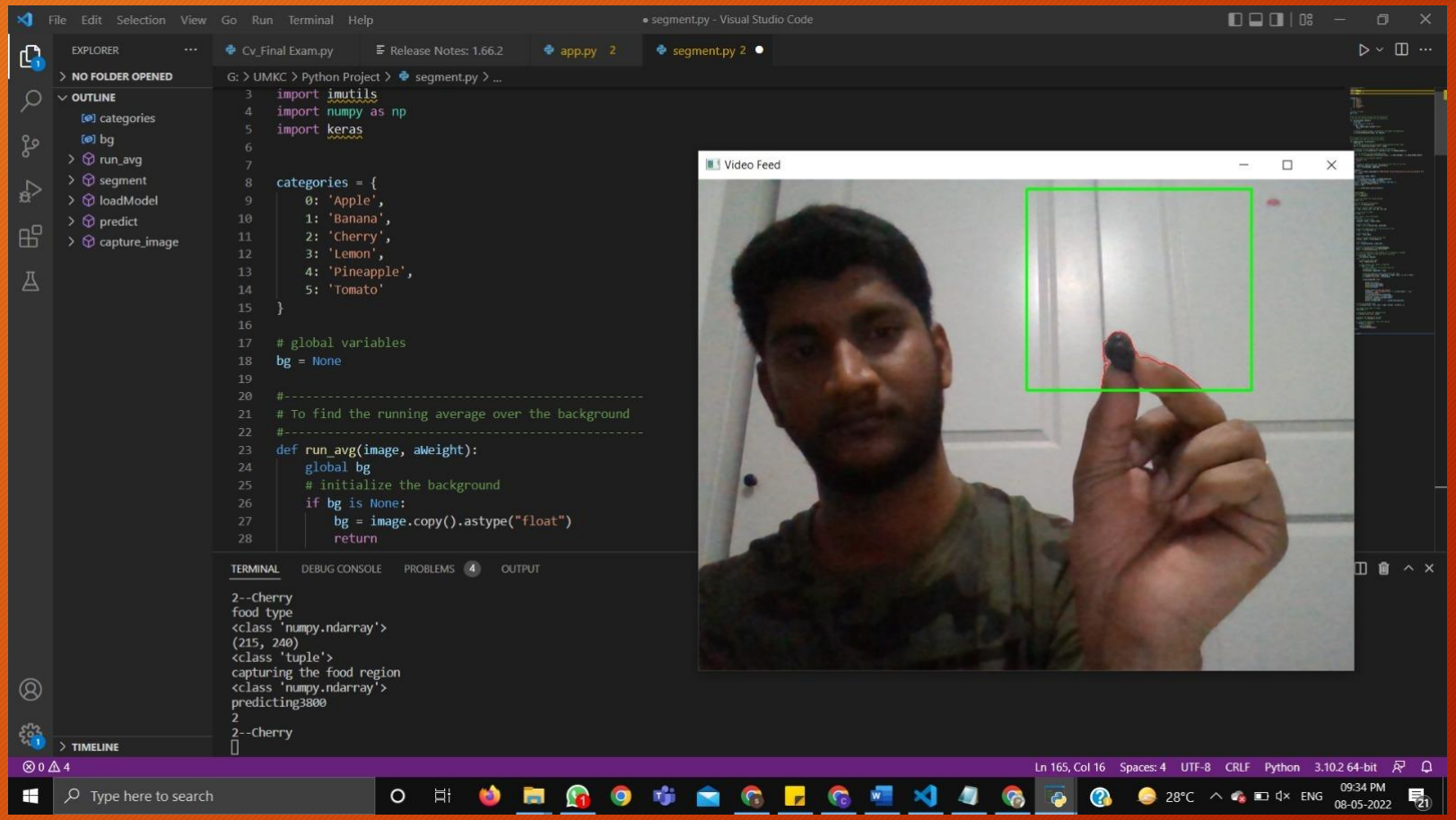
Predictions





Predictions





Interface

Colab notebook detected. To show errors in colab notebook, set `debug=True` in `launch()`
Running on public URL: https://55319.gradio.app

This share link expires in 72 hours. For free permanent hosting, check out Spaces (https://huggingface.co/spaces)

INP

OUTPUT 0.1s

Pineapple - 452 cal

Pineapple - 452 cal	100%
Apple - 95 cal	0%
Banana - 105 cal	0%

Flag

Clear

Submit

Future Scope

Now that we are able to predict the food given by the end user using our application, we can extend the functionality so to predicting multiple types of food in a single image, calculate quantity, derive nutrition tables etc.

Thank you!! Any Questions???