# CALORIE COACH – FOOD CALORIE COUNTER

R.Shylaja[1], P.Akhila[2] , S.Meghana[3] , M.Sowmya[4] , D.Lasya[5]

[1] R. Shylaja, Assistant Professor, Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women

[2] P. Akhila, Student, Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women

[3] S. Meghana, Student, Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women

[4] M. Sowmya, Student, Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women

[5] D. Lasya, Student, Department of Electronics and Communication Engineering, BVRIT HYDERABAD College of Engineering for Women

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract** - This Project presents the design and development of a food recognition UI application which can also display the estimated calorie/s of the food itself. It is intended for people who would like to monitor their diet through food calorie intake measurement (i.e. user's daily calorie intake record). It is equipped with a food database consisting of typical fruits and vegetables commonly consumed. The result shows 82.86 % accuracy for the all the food categories. The algorithm being used in this project is Mask- Regional Convolutional Neural Network (Mask R1CNN) for the recognition process to be properly achieved. The forementioned database supported by TensorFlow is implemented in the project, which is an open-source software library for Machine Intelligence.

**Keywords—**Food reorganization, UI Application, Mask Regional-Convolutional neural network, Tensor Flow

## 1.INTRODUCTION

Calorie is a measuring unit which is defined as the amount of heat energy needed to raise the temperature of one gram of water by one degree. The process of providing or obtaining the food necessary for health and growth is called Nutrition. This unit is commonly used to measure the overall amount of energy in any food portion that consists of the main food components of Carbohydrate, Protein and Fat. Calories are a must for the body, as they are generating energy. But it is said that an excess of anything is bad and the same applies to the intake of calories too. If there is an excess of calories in our body, it gets stored in the form of fats, thus making us overweight. There are many chances for obese people to face a serious health problem like hypertension, heart attack, diabetes, obesity, hypertension, high cholesterol etc. So, the main cause for obesity is imbalance of the amount of food intake and energy consumed by the individual since it is necessary to have healthy meal. This system reviews the different systems which had taken the food images to measure the calorie and nutritional level in the food sample. As such, this system is used to measure the number of calories consumed in a meal would be of great help not only to patients and dietitians in the treatment of obesity also to the calorie conscious person. In our proposed system, we use colour, size, shape, and texture features for object detection. Later the calorie of the food is detected and displayed.but also to the calorie conscious person. In our proposed system, we use colour, size, shape, and texture features for object detection. Later the calorie of the food is detected and displayed. heart attack, diabetes, obesity, hypertension, high cholesterol etc. So, the main cause for obesity is imbalance of the amount of food intake and energy consumed by the individual since it is necessary to have healthy meal. This system reviews the different systems which had taken the food images to measure the calorie and nutritional level in the food sample. As such, this system is used to measure the number of calories consumed in a meal would be of great help not only to patients and dietitians in the treatment of obesity, but also to the calorie conscious person. In our proposed system, we use colour, size, shape, and texture features for object detection. Later the calorie of the food is detected and displayed.

## 2. LITERATURE SURVEY

### 2.1. Mask R-CNN[5]

Mask R-CNN is a state –of–the- art-framework for Image Segmentation tasks. There are two types of Image Segmentation. They are

1. Instance Segmentation
2. Sematic Segmentation

**Sematic Segmentation**

Semantic segmentation refers to the process of linking each pixel in an image to a class label. These labels could include a person, car, flower, piece of furniture, etc., just to mention a few. We can think of semantic segmentation as image classification at a pixel level.

**Instance Segmentation**

Instance segmentation assigns a label to each pixel of the image. It is used for tasks such as counting the number of objects.

Instance segmentation requires

- Object detection of all objects in an image. Here the goal is to classify individual objects and localize each object instance using a bounding box
- Segmenting each instance. Here the goal is to classify each pixel into a fixed set of categories without differentiating object instances.

Mask R-CNN is a Convolutional Neural Network (CNN) and state of the art in terms of image segmentation. This variant of deep neural network detects objects in frequently an image and generates a high-quality segmentation mask for each instance.

**Steps to implement Mask R-CNN**

Step 1: Clone the repository. First, we will clone the Mask R- CNN repository which has the architecture for Mask R-CNN

Step 2: Install the dependencies.

Step 3: Train and test the model.

Step 4: Predicting for our image.

RCNN is an object detection architecture is one of the famous object detection architectures that uses convolution neural networks like YOLO (You Look Only Once) and SSD (Single Shot Detector). R-CNN detection system consists of three modules. The first generates category-independent region proposals. These proposals identify the set of candidate detections present in an image. The second module is a deep convolutional neural network that extracts a feature vector from each region. The third module is a set of class-specific classifier i.e. linear SVMs. R-CNN does what we might intuitively do as well – propose a bunch of boxes in the image and see if any of them correspond to an object. R-CNN creates these bounding boxes, or region proposals, using a process called Selective. At a high level, Selective Search (shown in Fig:1 below) looks at the image through windows of different sizes, and for each size tries to group adjacent pixels by texture, color, or intensity to identify objects.

## 2.2. Improving the Bounding Boxes

After founding the object in the box, we can tighten the box to fit the object to its true dimension. This is the final step of R-CNN. R-CNN runs a simple linear regression on the region proposal to generate bounding box coordinates to get the result. The inputs and outputs of this regression model are: Inputs: sub-regions of the image corresponding to objects.

## 2.3. R-CNN implementation

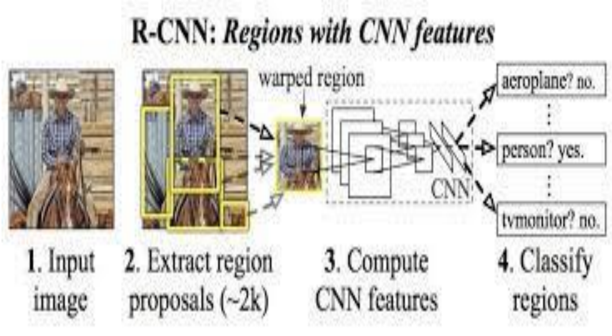


Fig-1: Architecture of R-CNN

RNN Architecture has 3 parts

**Part 1**

Convolution layers In this layers we train filters to extract the appropriate features the image, for example let's say that we are going to train those filters to extract the appropriate features for a human face, then those filters are going to learn through training shapes and colors that only exist in the human face. Convolution networks are generally composed of Convolution layers, pooling layers and a last component which is the fully connected or another extended thing that will be used for an appropriate task like classification or detection. We compute convolution by sliding filter all along our input image and the result is a two-dimension matrix called feature map. Pooling consists of decreasing quantity of features in the features map by eliminating pixels with low values. And the last thing is using the fully connected layer to classify those features.

**Part 2**

Region Proposal Network (RPN) RPN is small neural network sliding on the last feature map of the convolution layers and predict whether there is an object or not and predict the bounding box of those objects.

**Part 3**

Classes and Bounding Boxes prediction Now we use another Fully connected neural networks that takes as an input the regions proposed by the RPN and predict object class (classification) and Bounding boxes (Regression).

**Training**

To train this architecture, we use SGD to optimize convolution layers filters, RPN weights and the last fully connected layer weights.

## 3.Working

The steps to be implemented for building the proposed solution are as follows:

Dataset Loading

↓

Splitting into Training and Testing data

↓

Mask R-CNN Model

↓

Training and Testing the model

↓

Validating the Results

Fig-2:. Mask R-CNN Model Creation

Step1: Firstly, the required Mask R-CNN libraries need to be imported, Then the dataset, which contains the images of variety food items i.e., fruits, vegetables etc., is downloaded.

**Dataset preparation[2]**

We have used the publicly available Kaggle dataset in our project. The dataset is in the form of CSV file which containsthe calories of various food items. The CSV file contain three columns. First column is food, which the various food item names. Second column is serving, which contains quantity of food. Third column is calories, which contains the calorie of each food item.

| A | B | C |
|---|---|---|
| Pumpkin | 1 pumpkin (196 g) | 51 |
| Radishes | 1 radish (4.5 g) | 1 |
| Red Cabbage | 1 leaf (22 g) | 7 |
| Rutabaga | 1 rutabaga (386 g) | 147 |
| Shallots | 1 shallot (25 g) | 18 |
| Spinach | 1 bunch (340 g) | 78 |
| Squash | 1 squash (196 g) | 88 |
| Sweet Potato | 1 potato (130 g) | 112 |
| Tomato | 1 tomato (111 g) | 20 |
| Turnip Greens | 1 turnip green (170 g) | 34 |
| Turnips | 1 turnip (122 g) | 34 |
| Wasabi | 1 root (169 g) | 184 |
| Winter Squash | 1 squash (431 g) | 147 |
| Zucchini | 1 zucchini (196 g) | 33 |
| Acai | 1 oz. (28.35 g) | 20 |
| Apple | 1 apple (182 g) | 95 |
| Applesauce | 1 cup (246 g) | 167 |
| Apricot | 1 apricot (35 g) | 17 |
| Avocado | 1 avocado (200 g) | 320 |
| Banana | 1 banana (125 g) | 111 |

Fig-3: Input Dataset Sample1

Step2: Then the fruit dataset is divided into training and testing dataset. After dividing, that various functions are written for trainGenerator to train the model and testGenerator for testing the model with testing dataset. Latermodel displays the image along with its probable prediction and save the test features for further use.

Steps3: To design the Mask R-CNN through which we will transfer our features to train the model and then test it using the test features in this step, which is the most important part of the entire process. To construct Mask R- CNN, we used a combination of many different functions, which we will go over one by one.

1. Sequential () - A sequential model is simply a linear stack of layers that builds up from the input layer to the output layer.
2. Conv2D () - rtn.add (Conv2D ()) - The convolution operation is performed by this 2D Convolutional sheet. This layer generates a tensor of outputs by convolving the layer input with a convolution kernel.
3. rtn.add (BatchNormalization ()) - It performs batch normalization on inputs to the next layer so that the inputs are in each scale, such as 0 to 1, rather than being strewn around.
4. rtn.add (MaxPooling2D ()) - This function performs the data pooling process. With MaxPooling, the stride is usually set so that there is no overlap between regions.
5. rtn.add (Dropout ()) - Dropout is a training method in which randomly selected neurons are ignored. They disappear at random. Overfitting is reduced as a result.
6. rtn.add (Flatten ()) - This just flattens the input from ND to 1D and does not affect the batch size.
7. rtn.add (Dense ()) - According to the Keras documentation, this layer oversees producing the final label for the image being processed during testing.

After the model. summary () is executed.

Step4: First, the model is built using the RMSprop optimizer and categorical cross entropy as the loss function. For confirmation, accuracy is used as a measure. Following that, we fit the model with the epochs (50 in this case), and validation data that is obtained earlier by splitting the training data. The trained model can be tested with the help of testing images in the dataset.

In case of real time prediction scenario:

Input Url and Number of items

↓

Feature Extraction

↓

CNN Model

↓

Object Detection

↓

Reading Calorie Dataset

↓

Printing Calories

Fig-4: Real Time Calorie Prediction

Step5: In real time prediction scenario, we import our model and use Tkinter as GUI tool. While executing the model, the model asks image URL and number of items. Click upload and solve icon.[4]

Step6: Then model calls the various functions like LoadInput (), ImageDetection (), Calorieofeachitem () for loading image in local system, detecting the object, and reading calorie dataset to display calories for detected object.[6][3]

## 4. RESULTS

**A. Asking to enter the input image URL with number items in the image.**

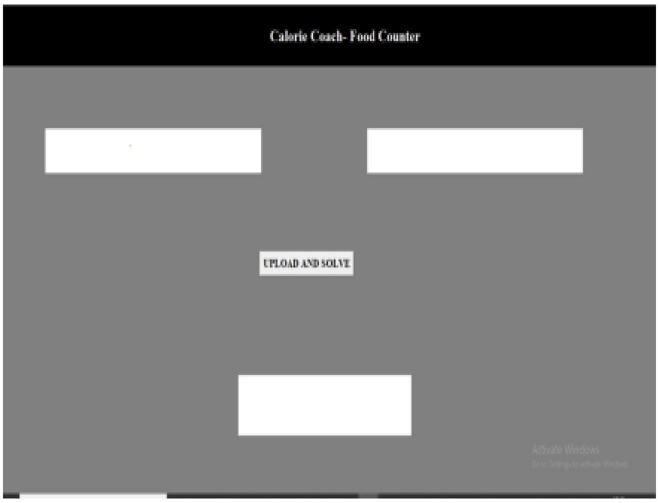


Fig-5:. Output1

**B. After entering details. Click Upload and solve.**

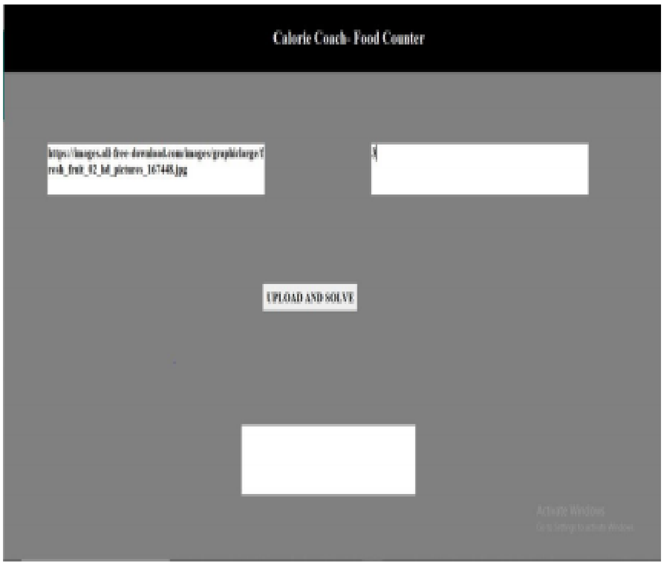


Fig-6:. Output2

**C. Loading the image in local disk and viewing the uploaded image in photos.**

Fig-7: Output3

**D. Calorie is displayed.**

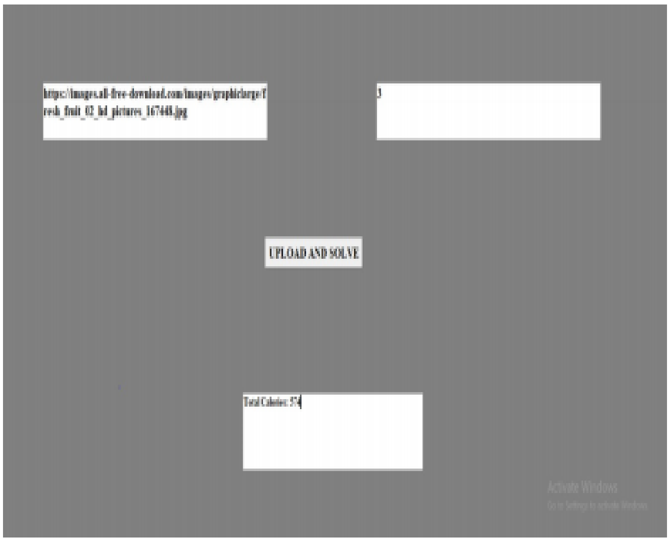


Fig-8: Output4

### E. Accuracy and loss function of our model

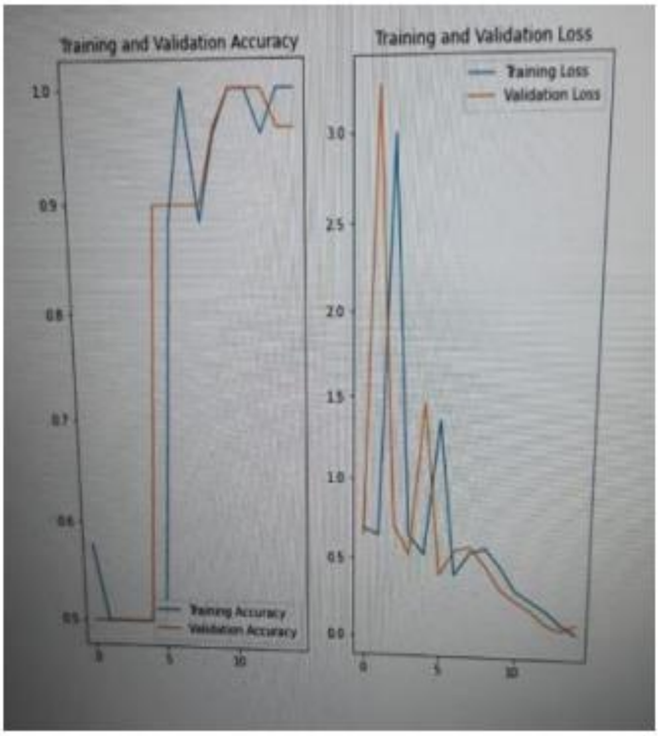


Fig-9: Output5

These are the plots of accuracy and loss function. Loss function helps to measure how far the estimated value is from true value. Accuracy is defined as percentage of correct predictions for the test data. Accuracy is chosen as a performance metric. The training accuracy of the model is approximately 99 percentage and the validation accuracy is around 93 percentage.

## 5. CONCLUSIONS

In this project, the implementation of food recognition and predicting the calories are based solely on their images. The proposed solution has a measurement method that estimates the number of calories from a food's image by measuring the area of the food portions from the image and using tables to measure the amount of calorie in the food. And calorie is shown in results with approximate value. It focused on identifying food items in an image by using image segmentation, and calorie measurement based on food portion and nutritional tables. To achieve this,a pipelined approach that predicts the type and size of the food item in the image, then displays the number of calories in the food item. All the prediction tasks were performed using supervised machine learning, which was based on a carefully annotated dataset of various food images.
This results indicated reasonable accuracy of the method in area measurement. The project UI is simple and easy to use as the actual platform is clear and easy to understand. Thus, the project changes the future of calorie prediction by implementing supervised machine learning and giving real time experience to the user. This project also aiddieticians/medical professionals for the treatment of obese or overweight people, although normal people can also benefit from this system by controlling more closely their daily eating without worrying about overeating and weight gain. Hence this system is very important in the field of biomedical.

In Future work & development, this system can be implemented using a hardware model for the calorie and nutrition measurement along with mass. By using a MATLAB and hardware interfacing controller for measuring the mass with high megapixel camera and precision sensor to take liquid food such as milk, sauce, tea, juices etc. Also, more work is needed for supporting mixed or even liquid food. Advance system can be designed to use any kind of plates having a different color for capturing a photo instead of white only. Also, extending the dataset to include different food cuisines from all over the world and add more diverse images with different settings such as the backgrounds or serving surfaces and study the effect of such factors on the prediction performance. Finally, proposed system can extend to handle the more realistic scenario where the user provides an image of a meal rather than just one individual food item as assumed here.

## REFERENCES

[1] Simonyan K., Zisserman A. Two-stream convolutional networks for action recognition in videos; Proceedings of the Advances in Neural Information Processing Systems; Montréal, QC, Canada. 8– 13 December 2014; pp. 568–576. [Google Scholar]

[2] https://ieeexplore.ieee.org/document/7007901

[3] https://ieeexplore.ieee.org/document/6748066

[4]https://docs.python.org/3/library/tk.html

[5] https://towardsdatascience.com/object-detection-using-mask-r-cnn-on-a-custom-dataset-4f79ab692f6d

[6] https://www.activestate.com/resources/quick-reads/what-is-pandas-in-python-everything-you-need-to-know/