Intelligent Nutrition Management System

ABSTRACT

Maintaining a food journal using traditional ways through manual logging is tedious, ineffective, and inefficient. With ever-growing diet consciousness among the masses and the emergence of fad-diets in the nutrition industry by the day, there is a growing need for a smart way to track your daily macro and calorie intake. This can be achieved by merely clicking a photograph of the image using our smartphones which can then be fed into a trained model which makes predictions, of not only the food items and ingredients on plate but also estimates its macro profile.
This project aims to achieve the same by using a deep learning model, Mask RCNN (Regions with Convolutional Neural Networks) and image processing techniques. I have made use of the coco dataset, which comes with pretrained weights. Hence the model is only capable of detecting apples, bananas, oranges, cakes, pizzas and carrots. Custom dataset training requires annotating a lot of test and validation data manually and a lot of time and resources to get trained. However, with proper training, the approach should work just fine on any food item.

INTRODUCTION

Despite recent advancements in medicine, the number of people affected by chronic diseases is still significantly large. This rate is primarily due to their unhealthy lifestyles and irregular eating patterns. Some of the more notable chronic diseases include obesity, hypertension, blood sugar, cardiovascular diseases, and different kinds of cancers. In order to control obesity and related chronic diseases, there is a pressing need to assess accurately the energy and nutrient intake of individuals in their daily lives. Traditionally, a dietary assessment is conducted using self-report in which individuals report their consumed foods and portion sizes. Although this method is standard and has been utilized for decades, numerous studies have indicated that it is inaccurate up to as much as 30% and biased. In addition, self-report does not work well in children. With the development of smartphones and wearable devices, dietary assessment can be performed without fully depending on individuals' memory and willingness to report their own intake.

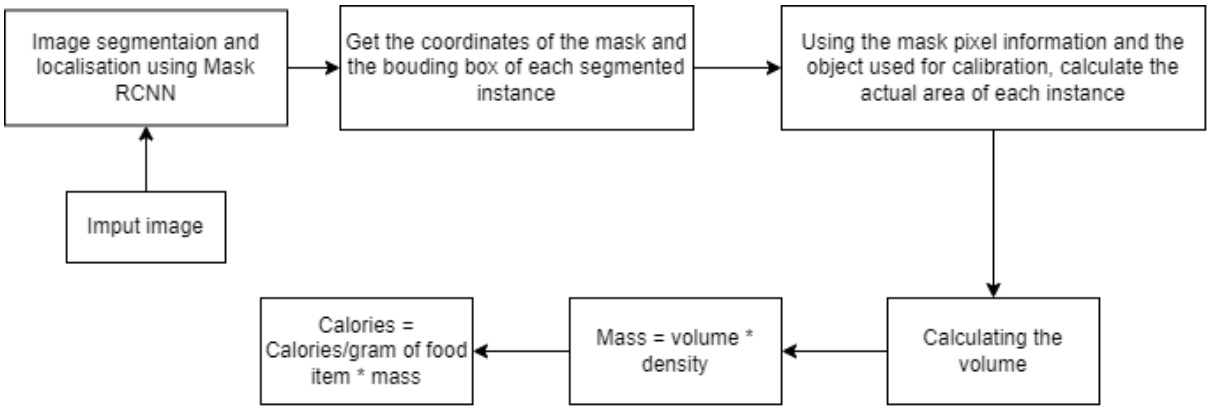
Several methods exist where manual food's volume is estimated using its image captures using a smartphone. Some methods involve capturing a single image, while multiple views are needed to determine accurate volume in other techniques. The food volume estimation process involves the following two steps 1) multiple images or a single image from a mobile camera is needed 2) computation of food volume from 3d construction or calibration object. Regardless of other volume estimation tasks, food volume estimation is a complex task with many specific challenges. Many foods have variations in shape and appearance due to shape and eating conditions.

Several such systems already exist like eButton, a wearable camera for objective and passive dietary assessment. Online platforms like HealthifyMe and FitGenie are also working in the same direction. Other such applications are FoodLog, Im2Calories, FoodCam, DietLens, Food Tracker, MyDietCam. All these apps use a variety of featyre extraction methods from SIFT, HoG, CNN, DCNN etc and classification methods like Adaboost classifier, Support Vector Machines, Linear SVM, Random Forest Classifier and the list goes on.

However, automatic food recognition using a smartphone camera in the real world is considered a multi-dimensional problem. Unlike other image classification problems, food recognition is a complex task that involves several challenges. There is no spatial layout information that it can exploits like, in the case of the human body, there is a spatial relationship between body parts. Similarly, the non-rigid structure of the food and intra-source variations make it even more complicated to classify food items correctly as preparation methods and cooking styles vary from region to region. Moreover, inter-class ambiguity is also a source of potential recognition problems as different food items may look very similar. Moreover, in many dishes, some ingredients are concealed from view that can limit the performance of food ingredient classification models. In addition to this, image quality from the smartphone camera is dependent on different types of cameras, lighting conditions, and orientations. As a result, the poor performance of food recognition models is highly susceptible to image distortions.

This project is my attempt to replicate the same function, nutrition assessment from image of food items. My proposed system runs on a web application where the user uploads an image of the meal. Mask RCNN model has been employed in this application. It uses pretrained COCO dataset to detect the food item present in the image.

PROPOSED METHODOLOGY



1. **Image Acquisition**: An image of the meal should be taken using the user's smartphone. The image must have a spoon inside it. This will be used for size calibration.

2. **Mask RCNN**: The image is then passed through the Mask RCNN model. I have used the COCO (Common Objects in Context) dataset for identification of our food items. Since the dataset has only 10 food items that it is trained on, apple, banana, oranges, sandwich, donut, hot dogs, carrot, cakes, and broccoli, it will not be able to identify other food items. Custom training of dataset is possible, but I have avoided that due to time constraints and the labour of manually annotating a large dataset, which is practically impossible.

3. **Area Calculation**: After applying the mask RCNN model, we get segmented instances of all the objects that the model could recognize. We use the masks of each of these images and calculate their actual area. This is done by finding the area covered by the spoon and

dividing it by the actual area of the spoon. This factor can then be multiplied to the area of all the images to yield correct areas.

4. **Volume Estimation**: After image recognition with localisation, we have their area. We have the approximate shape of each food item stored in a dictionary. We refer this dictionary for volume approximation making use of the area of the masks and the dimensions of the bounding box.

5. **Mass calculation**: In the same dictionary we have, the density and the macro information of each of these food items. Mass can be calculated by multiplying the obtained volume with the item's density, which can then be used to calculate their macro content.

MASK RCNN

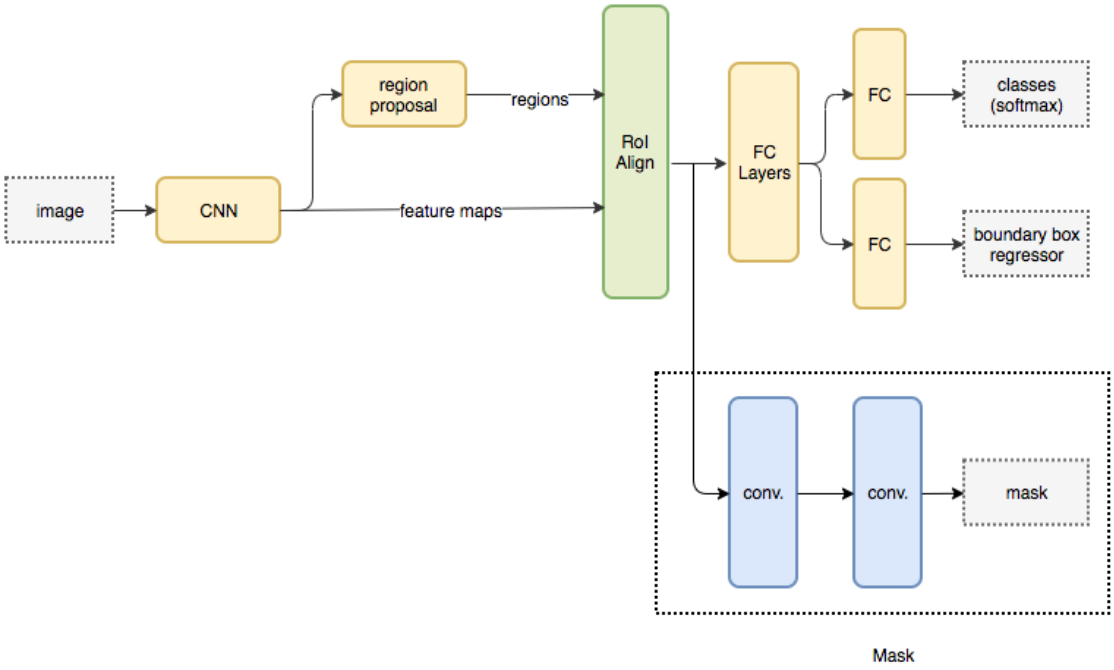
To identify what's on the plate, we need to instance-segment the given food image into the possible food categories. Instance Segmentation classifies individual pixel in the given picture into possible classes ie. foods in our case. Given the problem of instance segmentation, the architecture of Mask R-CNN would be a matching solution. Mask R-CNN takes an image and spits out three outputs, masks of the identified items, bounding boxes and classes for each mask detected. Masks are the binary coded single-channel matrices of the size of the input image which denote the boundaries of the identified object.

Mask R-CNN is developed based on Faster R-CNN, which is a region-based Convolutional Neural Network. A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm that can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and can differentiate one type of image from the other. To have object detection, we need to know the class of the object and also the bounding box size and location. Conventionally, for each image, there is a sliding window to search every position within the image as below. It is a simple solution. However, different objects or even the same kind of objects can have different aspect ratios and sizes depending on the object size and distance from the camera. And different image sizes also affect the effective window size. This process will be extremely slow if we use deep learning CNN for image classification at each location.

To bypass the problem of selecting a huge number of regions, Ross Girshick et al. proposed a method where we use selective search to extract just 2000 regions from the image and he called them region proposals. The architecture is called R-CNN.

The same author of the previous paper (R-CNN) solved some of the drawbacks of R-CNN to build a faster object detection algorithm and it was called Fast R-CNN. The approach is similar to the R-CNN algorithm. But, instead of feeding the region proposals to the CNN, we feed the input image to the CNN to generate a convolutional feature map.

Mask R-CNN extends the header to 3 branches compared to just 2 branches in Faster R-CNN, one additional branch of mask identification is added to the Faster R-CNN architecture. A mask image is simply an image where some of the pixel intensity values are zero, and others are non-zero, which determines the boundings of an object. Apart from that, Mask R-CNN uses ROI align which utilizes bilinear interpolation for Region Of Interest (ROI) compared to floor division used in Faster R-CNN which hugely misplaced masks at outputs but served sufficient accuracy for bounding box prediction.



Mask

CALORIE DETECTION AND ACCURACY OF THE MODEL

The code was run in google colab environment. The link to the source code is attached in the report.

We test our algorithm by feeding it the following input image. It has two bananas and a spoon that will serve as our calibration object.

The model returned the following output:

The model has segmented the spoon, the dining table and both the banans as a single object.

It returned the following estimate of the calorie of the food:

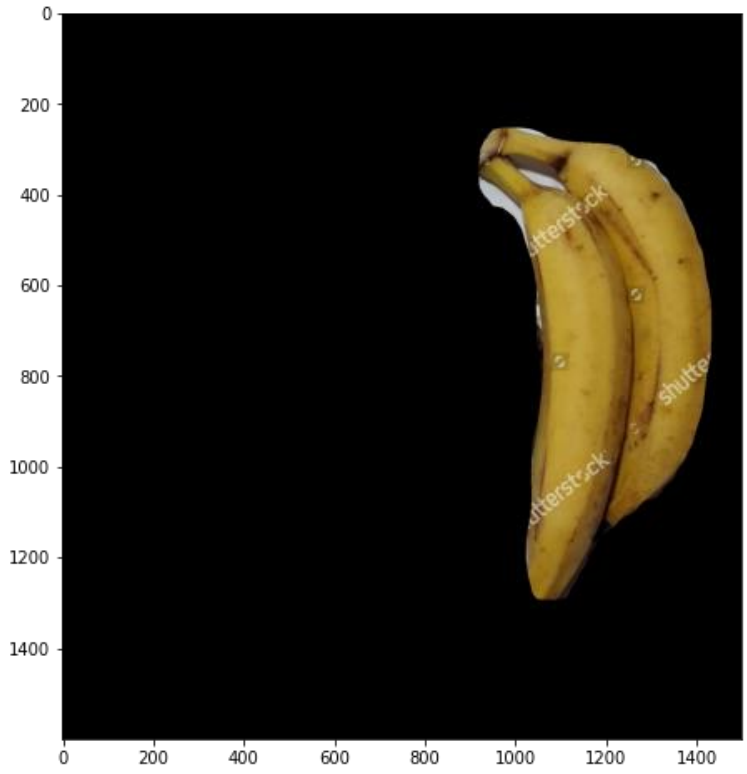
```
Detected one  banana of mass  201.39999999999998 with 221.54 calories
```

The average weight of a long banana is around 150g. Our model seems to have done a decent job in estimating the calorie of the food item.

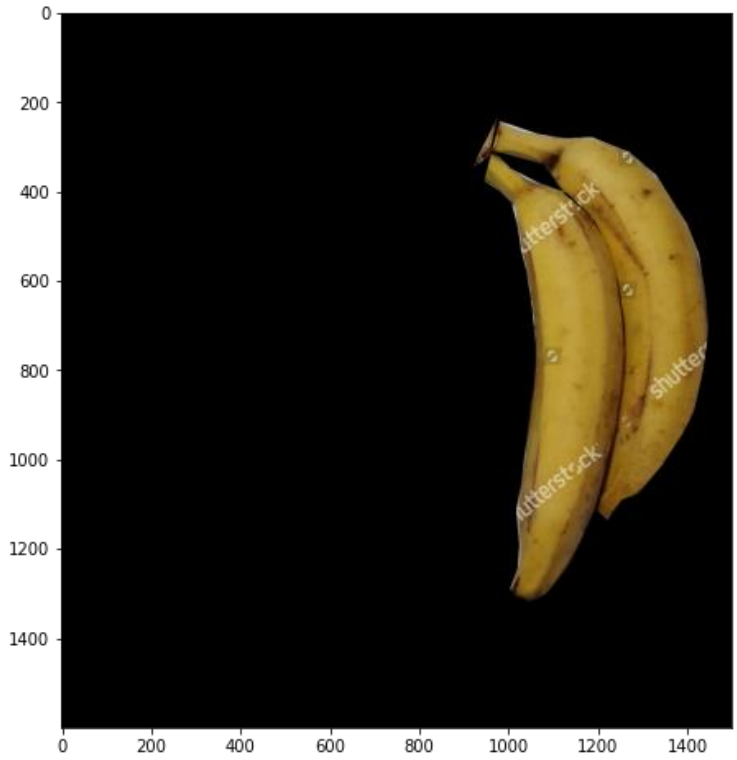
To test the accuracy of our model, we used the IOU metric that is the most popular parameter in image recognition algorithms. It stands for Intersection over union and if the IOU value is above 0.5, it is considered as a good prediction.

```
0.9170260275752886
```

This was the IOU score for this image. We have taken the IOU of the obtained masks and the actual area the bananas cover (annotations).



The image area under the segmented mask.



Actual region of the image

Average IOU scores of different objects detected by the algorithm are shown in the table:

Banana	0.9456
Apple	0.7800
Carrot	0.8299
Orange	0.9812

Link for the google colab notebook: https://colab.research.google.com/drive/1CzS7d8laioPyoBbhSMS7-BA938NRuOJe?usp=sharing

CONCLUSION

For this food recognition and object detection, we saw that the model does a pretty good job if the calibration object is present in the image otherwise in the absence of which it uses the set approximate factor to calculate the volume which often gives a value which is farther from the actual value.
The model has a good IOU score for almost all the objects it has been trained on.
The model will not be able to detect the ingredients present in a food preparation and that remains a challenging task. It can only detect food items like chapati, rice, roti or even a curry, provided that it had been trained sufficiently on the images of the same but to tell the calorie content of such items, like curry is a complex task as the ingredients differ across geographies and demographies.