International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V69I5P110

Volume 69 Issue 5, 69-76, May 2021
© 2021 Seventh Sense Research Group®

Original Article

Food Item Calorie Estimation using YOLOv4 and Image Processing

Samidha Patil[1], Shivani Patil[2], Vaishnavi Kale[3], Mohan Bonde[4]

[1,2,3]*Final Year B.Tech IT Student,*[4]*Assistant Professor, Department of Information Technology, UMIT, SNDT Women's University, Juhu Tara Road, Santacruz(W) Mumbai Maharashtra India.*

Received Date: 15 April 2021
Revised Date: 29 May 2021
Accepted Date: 31 May 2021

Abstract - *In last decade or two, an increase in growth of obesity has been seen all around the world. There has been increasing research to tackle obesity using food logging and food item calorie analysis. An increase in healthy living has led to numerous food management applications, which have image recognition to automatically record meals. To achieve healthy living it's important for someone to observe his/her daily calorie intake. The project aims to incorporate modern technique for object detection together with image analysis techniques to determine a more accurate calorie count from images of food items. The strategy employed involves determining the calorie count of the food item through mathematical calculations of the features extracted from food image by image segmentation. In this paper, we propose a mobile software for food calorie estimation from images of food items. By using YOLO- You Only Look Once for Object detection and Image segmentation for calorie estimation we are able to detect the food and thereby calculate the required food calories from the varied datasets of Indian cuisine.*

Keywords - *Calorie Estimation, Object Detection, YOLO.*

I. INTRODUCTION

Food is necessary for human life. Since it is the main source of energy and all the nutrients we get from the food, it is an important source of health and well-being. However, the human's diet is changing from the last one or two decades, as it is becoming high in saturated and trans fats and salt, and low in vegetables and fruits. These changes in diet are causing various diseases such as obesity, diabetes, cardiovascular diseases and cancer. To deal with these issues, most of the people are changing their diet plans by focusing on what type of food they are consuming.

To know what we consume, we regularly create a record of everyday meals. Such recording of the food is manual exercise and it is complex and time consuming task. It's not easy task for consumer to create healthy food selections as the nutritional information available is not easily interpretable. Thus, to make it straightforward for the customers, this project has come into existence. Automatic food classification is beneficial in real-world applications like waiter-less restaurants and private health. As an example, mobile food classification has been used to tell the users their daily dietary requirements and calorie intake. This work aims to develop a mobile application that may record real time images of meal and analyse it for calorie content, so that people will improve their dietary habits and lead a healthy life.

In this project, we have used YOLO (You Only Look Once) and Image Segmentation for recognition and calorie estimation of food items respectively. YOLO (You Only Look Once) is a real time object detection algorithm. It is the algorithmic rule or strategy behind how the code is going to detect the objects from the image. It looks at the image just once, then goes through the network and detects the object. The image segmentation technique from image processing is used to estimate the amount of calories present in the detected food item.

II. RELATED WORK

Tatsuya Miyazaki, Gamhewage C de Silva and Kiyoharu Aizawa [2] had mentioned an image analysis approach to calorie count estimation for dietary assessment. They had designed a dataset of 6000 images contained in food log calorie count that had been calculated by the nutritionist. The food image was compared with food log dataset from the aspect of multiple options like Correlograms, color Histograms and SURF options.

Parisa Pouladzadeh, Shervin Shirmohammadi, and Rana Al-Maghrabi [3] had presented a food calorie and nutrition measurement system that helped patients and dietitians to manage daily food intake. The paper projected to build a mobile application which will offer the measure of calorie and nutrition contents by clicking the photo before and after the consumption of food.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi [6], they had introduced YOLO, a unified model for object detection. This model is uncomplicated to construct and can be trained directly on full images In contrast to classifier-based approaches, YOLO is trained on a loss function that directly corresponds to detection performance and hence the entire model is trained together.

Manal Chokr, Shady Elbassuoni [9] projected the simplest way to estimate the calorie content of the food item (pointed towards eatables like pizzas, doughnuts, chicken, and sandwiches) by measuring the size of the item by passing the compressed image through a regressor.

Joseph Redmon and Ali Farhadi [8], they had introduced YOLOv2 and YOLO9000, the detection systems that were real time. YOLOv2 is quite faster than other detection systems. YOLO9000 is a real-time framework for detection of more than 9000 various objects categories with both optimizing detection and classification.

Sujata Chaudhari, Nisha Malkan, Ayesha Momin, Mohan Bonde [12] used YOLO for real time object detection. This paper aims to involve YOLO as a modern technique for object detection with the goal of achieving high accuracy. The increment in accuracy throughout training process is explained in this paper. They have explained advancement of CNN in this paper and with the assistance of YOLO and CNN object detection is performed.

Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao [13], their main objective of this work is to build a fast operating speed of an object detector in production systems and maximizing for parallel computations. They have developed an efficient and powerful object detection model. The state of art techniques are made more effective for single GPU training. A huge numbers of features are verified for boosting up the accuracy of both classifier and detector.

III. PROPOSED METHODOLOGY

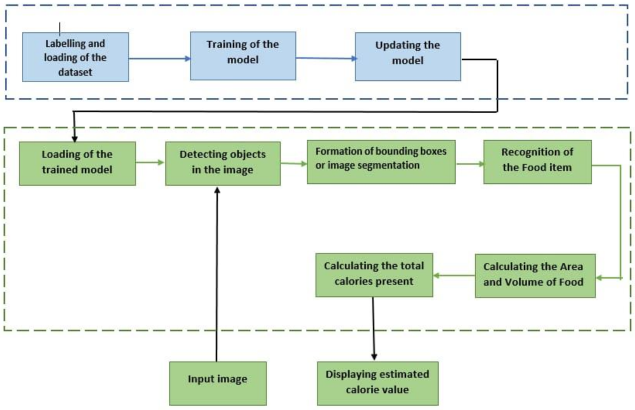
Labelling and loading of the dataset
Training of the model
Updating the model
Loading of the trained model
Detecting objects in the image
Formation of bounding boxes or image segmentation
Recognition of the Food item
Calculating the total calories present
Calculating the Area and Volume of Food
Input image
Displaying estimated calorie value

Fig. 1 Block Diagram of Proposed Methodology

A. Data Acquisition

The dataset comprises of food images from Kaggle datasets and also a custom dataset created by the team members. This dataset consists of images belonging to different classes such as Apple, Banana, Orange, Pizza, etc. Since finger is used as a calibration object, all the images are taken in such a way that finger is placed near the food item. [17], [18].

B. Image Preprocessing

a) Image Cropping

Image cropping is performed to crop out any unnecessary part of the image.

b) Image Resizing

After image cropping, image resizing is done to make all the images of equal size.

C. Labelling of the Dataset

The graphical image annotation tool named LabelImg is used for labelling of the images. The images are labeled by creating boundary box or rectangular box around the object in the image. The annotations are saved as txt file or xml file in Pascal voc format.

D. YOLO (You Only Look Once)

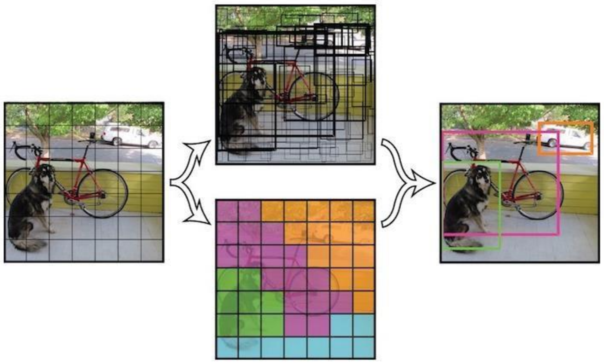

Fig. 2 You Only Look Once

YOLO (You Only Look Once) is an algorithm for object detection in real time. YOLO applies single neural network to the image, then divides the image into regions and predicts bounding box for each possible region. YOLO can be applicable to multiple objects in a single image and able to predict multiple bounding boxes and class probabilities for those boxes.

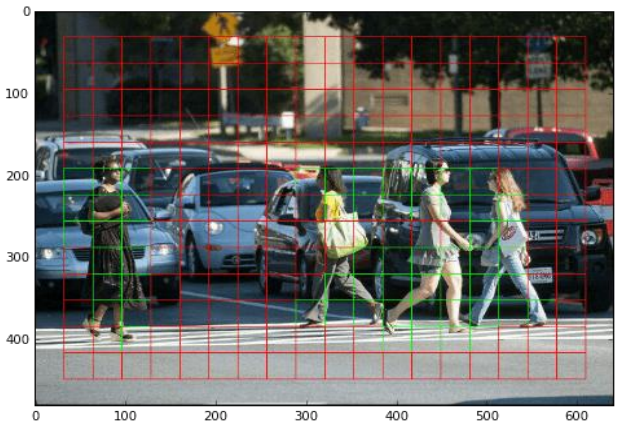

Fig. 3 Division of input image into grid

YOLO Algorithm works according to the following three techniques -

a) Residual Block

Here the image in Figure 3 is divided into various grids with S*S dimension for each grid. In the image, there are multiple grids of equal dimensions. Every grid cell will detect objects that appear within the particular grid.

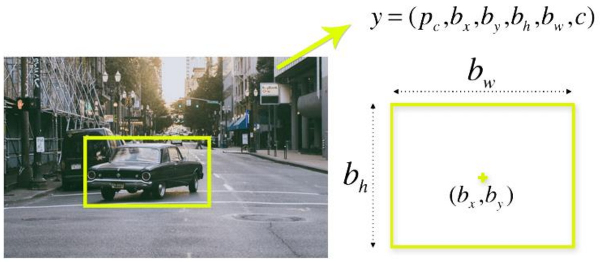

Fig. 4 Bounding Box Representation

b) Bounding Box Regression

The bounding box shown in Figure 4 is used to highlight the object in an image. YOLO uses single bounding box to find the width, height, center and the class of the object. It consists of four attributes; they are Width (bw), Height (bh), Class (for example. person, car, traffic light, etc.) This is represented by letter c, and the last attribute is bounding box center (bx, by).

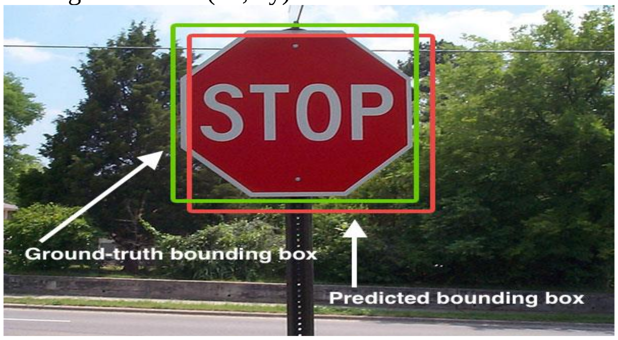

Fig. 5 An example of detecting stop sign in an image

c) Intersection Over Union (IOU)

Intersection Over Union describes overlapping of the boxes as shown in Figure 5. It is a measure to calculate accuracy of an object detector. To apply Intersection Over Union (IOU), we need two things; the ground-truth bounding boxes and the predicted bounding boxes from the image.

$$\text{IOU} = \text{Area of Overlap} / \text{Area of Union} \qquad \text{— (1)}$$

Using the above formula, IOU can be computed.
If ground-truth bounding box is same as the predicted bounding box then IOU is equal to 1 as shown in Figure 6.

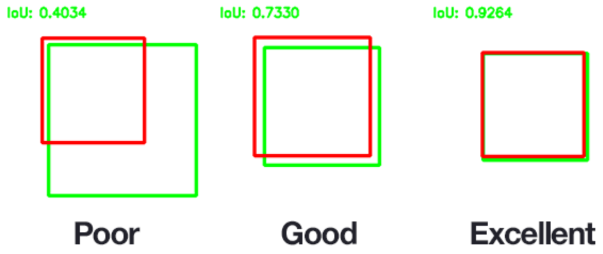

Fig. 6 An example of computing IOU for various bounding boxes

E. *The YOLOv4 Network*

The YOLOv4 network consists of three parts -

a) Backbone

Backbone refers to the feature extraction architecture. It is a Convolution Neural Network (CNN) that combines and forms image features at different granularities. The different versions of YOLO are differentiated depending on the backbone. In YOLOv4 CSPDarknet53 is used. CSP stands for Cross-Stage-Partial connections. [14]

b) Neck

The neck block is used to add extra layer between the backbone and the head (dense prediction block). The YOLOv4 uses modified Pan Aggregation Network (PAN) [10], modified Spatial Attention Module (SAM) [11] and Spatial Pyramid Pooling (SPP) to add extra information in a layer. It is the series of layers to mix and combine image features to pass them forward for prediction.

c) Head

The head block is used to locate bounding boxes and classify the objects present in the image. It analyzes the features from the neck block, detects bounding box and classifies it into various classes.

F. *Convolutional Neural Network*

The Convolutional Neural Network (CNN) is a feed-forward neural network that is used to detect and classify objects in an image. It is also known as ConvNet [4]. A convolutional neural network consists of an input and an output layer, and multiple hidden layers. The hidden layers of a CNN consist of convolutional layer, RELU layer which is activation function, pooling layer, fully connected layer and normalization layer. In CNN, every image is represented in the form of array of pixel values. It is a network which can take input image, assign weights to various objects from the image and differentiate objects from one another. In CNN pre-processing required is much lower than the other classification algorithms. The task of convolutional neural network is to transform the images into a format that is easier to process, without losing the features which are necessary for getting a good prediction.

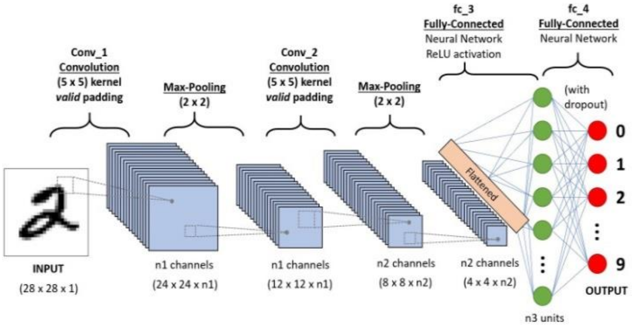

Fig. 7 An example of CNN Architecture representing sequence to classify handwritten digits

CNN is comprised of three types of layers

a) Convolutional Layer (The Kernel)

Convolutional layer is used to extract the various features from the input images. In this layer, convolution is performed between the input image and filter of size M*M. The element used for carrying out the convolution operation is termed as the Kernel or filter. The result of the convolution is termed as the feature map. Feature map gives the information about the edges and corners of the image. This feature map is passed to the other layers to get more other features from the input image.

b) Pooling Layers

Convolutional layer is followed by pooling layer. Pooling layer reduces the size of the convolved feature map to reduce the computational cost required to process the data. This is done by decreasing the connection between the layers and independently operating on each feature map. There exists two types of pooling which are max pooling and average pooling.

Max Pooling returns maximum value from the portion of the image covered by the kernel. Max pooling is considered as a noise reducer. It removes the noisy activations and performs de-noising along with dimensionality reduction.

Average Pooling returns the average of all the values that are covered by the kernel. Dimensionality reduction is achieved using average pooling. Performance wise max pooling is better than the average pooling.

c) Fully Connected Layer

The head block is used to locate bounding boxes and classify the objects present in the image. The Fully Connected (FC) Layer consists of the weights, biases and neurons which are used to connect the neurons of two different layers. These layers are placed before the output layer and form the last few layers of CNN architecture. In this layer, the input images from the previous layers are flattened and fed to the Fully Connected layer. The flattened vector then undergoes some more Fully Connected layers where the mathematical functions operation usually happens. In this step, the process of classification begins to take place.

G. Major Enhancement in YOLOv4

The YOLOv4 is based on Darknet. The difference between previous version of YOLO i.e., YOLOv3 and YOLOv4 is only the backbone. Yolov3 uses Darknet53 backbone while YOLOv4 uses CSPDarknet53 backbone. CSPDarknet53 backbone is used to enhance the learning capability of CNN. YOLOv4 has acquired an AP value of 43.5 percent on the COCO dataset and a real-time speed of 65 FPS. In YOLOv4, the AP and FPS has been incremented by 10 percent and 12 percent, respectively as compared to YOLOv3.

H. Calorie Estimation

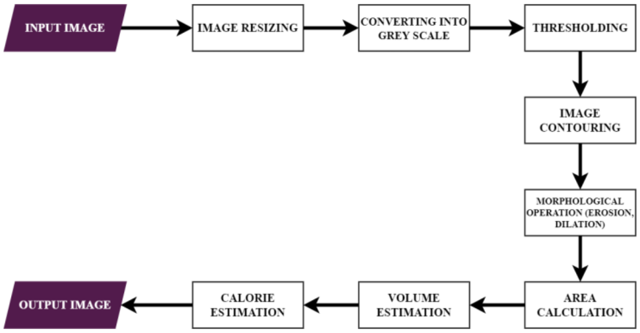

Fig. 8 Method of Calorie Estimation

We use various image processing techniques to calculate the volume of food from images. There are various methods of image segmentation such as canny edge detection, watershed segmentation, morphological operators and Otsu’s method which are used to segment the food item to find the contour of the fruit and thumb. Thumb finger is used as a calibration object. The thumb is kept next to the dish while clicking the photo and it helps us to estimate the real-life size of the food item and volume accurately.

After the food item is identified, the volume of food items by approximating it to a geometric shape like sphere, cylinder, etc is calculated. Once we get the volume, the mass of the food item is calculated using the standard value of density. The amount of calories is calculated using the values of the volume and mass of the food item.

These are the steps for performing Calorie Estimation

a) Image Segmentation

Image segmentation is a technique in digital image processing which is used to partition an image into multiple parts or regions with the help of the pixels of the image. It is mostly used to locate boundaries and objects in images. Following are the image segmentation steps.

i) Resizing of the original image

Image resizing is required to alter the number of pixels. The image is resized to 400×400.

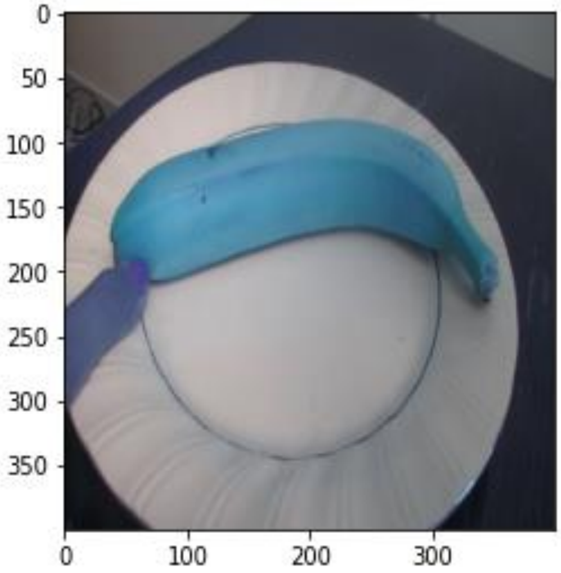

Fig. 9 Image Resizing

ii) Conversion of BGR image to Grey Scale

There are two methods for converting the image from BGR to Gray Scale as follows:

Average Method: In this method, the average value of R, G and B is taken as grayscale value.

Grayscale = (B+G+R) / 3 — (2)

The Weighted Method: This method weights red, green and blue according to their wavelengths.

Grayscale = 0.299R + 0.587G + 0.114B — (3)

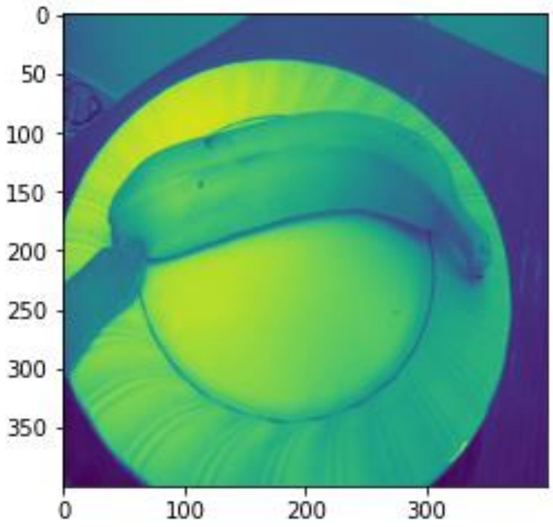

Fig. 10 BGR to Grayscale

iii) Threshold of an entire Image

Thresholding of an image is one of the simple of image segmentation. It is useful for creating binary image from grayscale image. The result of the thresholding separates the object or foreground pixels from the background pixels to perform image processing.

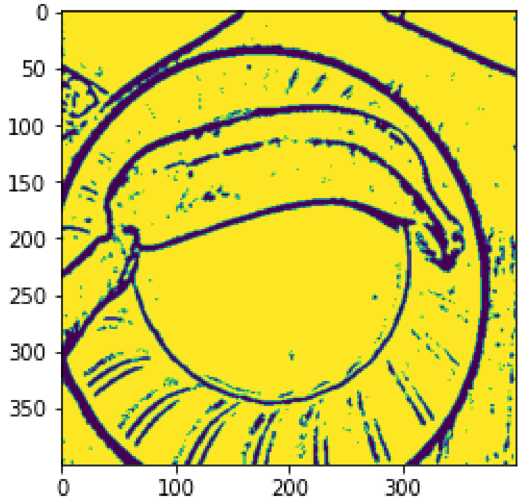

Fig. 11 Thresholding of entire image

iv) Finding Contours

A contour is defined as a curve connecting all the continuous points which are along the boundary and having same color or intensity. Binary images are used to get more accurate contours. It is required to apply threshold or canny edge detection before finding contours.

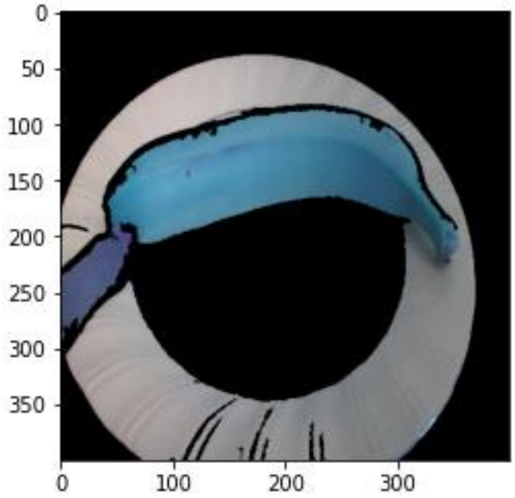

Fig. 12 Finding Contours

v) HSV Color spaced based segmentation

Thresholding of an image is one of the simple of image segmentation. HSV stands for Hue, Saturation and Value. HSV uses color, saturation and brightness values. HSV color space is closer to the RGB color space. Hue is the dominant color observed by humans. Saturation refers to the white light mixed with hue. Value is the brightness/ Intensity. HSV is used to detect the object in a certain color and to decrease the influence of light intensity. The distance measured from the central axis of HSV cylinder determines Saturation(S).

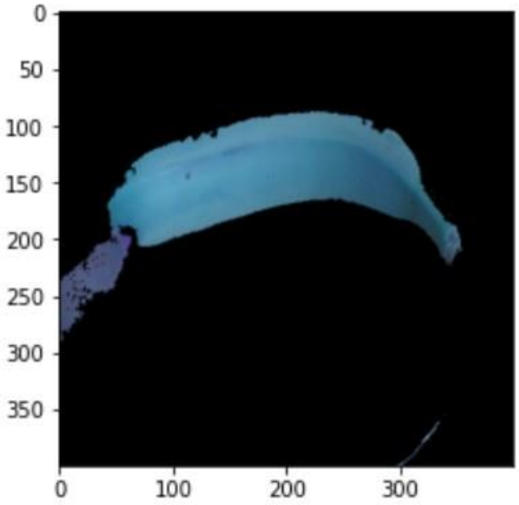

Fig. 13 Conversion to HSV

vi) Erosion and Dilation

Image resizing is required to alter the number of pixels. Erosion and Dilation are two morphological operations. Dilation adds pixels to the boundaries of the image while erosion removes pixels from the boundaries of the image. The size and shape of structuring element determines the number of pixels added or removed.

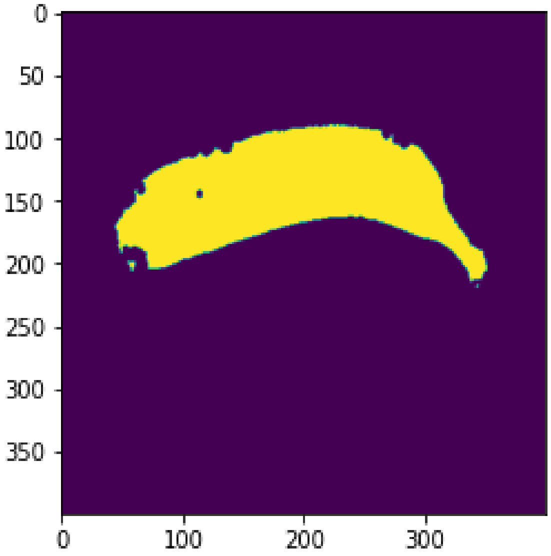

Fig. 14 Erosion

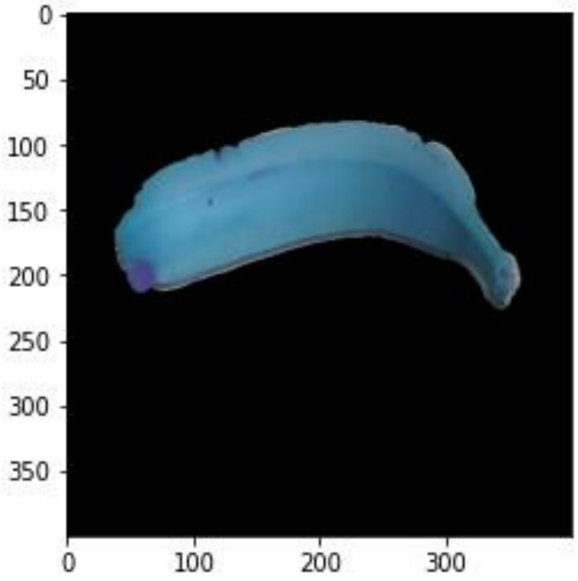

Fig. 15 Dilation

b) Area Calculation

After performing image segmentation steps on image, we get 3 factors namely Food pixel area, skin pixel area and actual skin area. Food pixel area and skin pixel area is calculated by converting an image to HSV. We know that our thumb size is approximately 5cm, so 5*2.3cm is taken as a skin multiplier. To calculate the actual skin area, following formula is used.

$$\text{pix\_to\_cm\_multiplier} = 5.0 / \text{pix\_height} \qquad \text{— (4)}$$

With the help of above 3 factors, estimated food area can be calculated.

Estimated Food Area = (Foods Pixel Area × Actual Skin Area) / Skin Pixel Area — (5)

c) Volume Estimation

The volume estimation method is used to measure the mass of food portion. Mass is required as all the nutritional tables are based on food mass. Once the value of the mass is calculated, then we can easily calculate the amount of calories present in the food item. Volume of food item is been calculated using the total area of food in the image with respect to finger i.e. the calibration object used. Then by approximating these detected item to a geometric shape like sphere, cylinder, rectangle, etc. the volume of food item is calculated.

Once we have the volume of food item, we can use the following formulae to calculate the mass of the food item.

$$M = \rho \times V \qquad \text{— (6)}$$

Where, M = mass of the food item, ρ = density

Food density can be obtained from readily available tables.

d) Calorie Estimation

To estimate the amount of calories present in the food item, we values of the mass and volume. Following equations are used to calculate the amount of calories.

Estimated Weight = Actual Density × Estimated Volume — (7)

Estimated Calories = (Estimated Weight × Calories per 100gm) /100 — (8)

IV. RESULT

The result of the model developed using the YOLOv4 method is represented by the graph obtained during the training process. At iteration 3400, the blue curve indicates the loss during training, which is 0.293 and the red curve indicates the mean Accuracy Precision (mAP) which is around 83%. The sudden drop in mAP curve near iteration 1200 is mostly due to mean precision is lower for that particular mini-batch in our dataset as compared to other mini-batches.

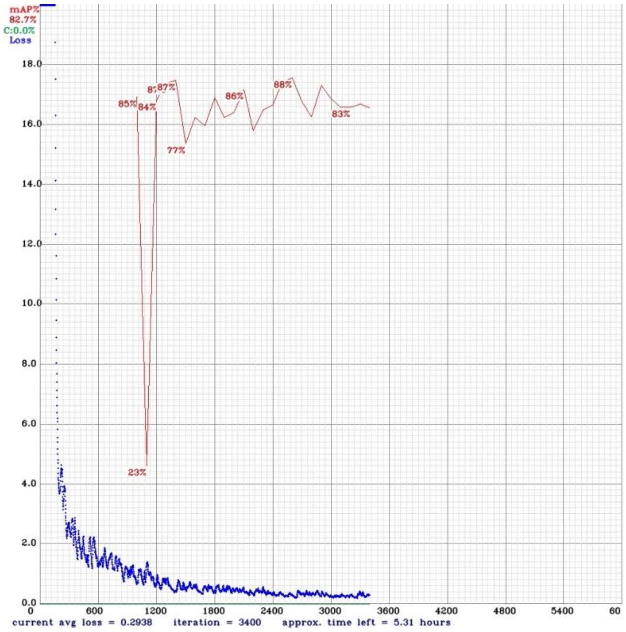

Fig. 17 Graph showing Loss and mean Accuracy Precision (mAP)

Further the values of calorie that are obtained using the model are compared with actual calories referenced from the database. [20]

Depending upon these actual and estimated value of calories, Table I is created.

Table 1. Comparison of actual and estimated calorie values

Food Item	Actual Calorie value	Estimated Calorie value	Difference (in %)
Apple	44	50	12.76
Banana	65	71	8.82
Orange	30	37	20.58
Carrot	41	45	9.30
Cake	297	260	13.26
Pizza	266	275	3.44
Broccoli	32	36	11.76

V. CONCLUSION AND FUTURE SCOPE

Our model is able to detect and recognize the food item accurately and also predict the calorie value. The prediction is in reference with the volume of a food item and finger calibration that is available in the image provided to the system. The system is able to give an output in the form of a bounding box over the food item with a label stating the name of the food item and calorie value that it holds.

Further work can be done in order to increase the accuracy of the system and more classes can be included so that the system is capable of estimating calories for a variety of food item.

ACKNOWLEDGMENT

We would like to express gratitude to our Guide Mr. Mohan Bonde, for giving us an opportunity to do the project work on ‘Food Item Calorie Estimation Using YOLOv4 and Image Processing’. The constant encouragement and support has been a guiding post throughout. We would also like to extend our gratitude to HOD of IT Department Dr. Sanjay Shitole, project coordinators Ms. Anita Morey for guiding us in selecting this topic as well as motivating us to do better.

REFERENCES

[1] Anil K Jain and Farshid Farrokhnia., Unsupervised texture segmentation using Gabor filters, Pattern recognition. 24(12) (1991) 1167–1186.

[2] Tatsuya Miyazaki, Gamhewage C de Silva, and Kiyoharu Aizawa, Image-based calorie content estimation for dietary assessment, In2011 IEEE Inter-national Symposium on Multimedia. (2011) 363–368.

[3] Parisa Pouladzadeh, Shervin Shirmohammadi, and Rana Al-Maghrabi, Measuring Calorie and Nutrition from Food Image, IEEE Transactions on Instrumentation and Measurement, 63(8) (2014) 1947–1956.

[4] Karen Simonyan & Andrew Zisserman, Very deep convolutional networks for large-scale image recognition , ICLR (2015)

[5] Patrick McAllister, Huiru Zheng, Raymond Bond, and Anne Moorhead, Semi-automated system for predicting calories in photographs of meals. In2015 IEEE International Conference on Engineering, Technology and Innovation/International Technology Management Conference (ICE/ITMC) (2015) 1–6. IEEE.

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016) 779–788.

[7] Koichi Okamoto and Keiji Yanai, An automatic calorie estimation system of food images on a smartphone, In Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management (2016) 63–70.

[8] Joseph Redmon and Ali Farhadi, Yolo9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 7263–7271.

[9] Manal Chokr, Shady Elbassuoni, Calories Prediction from Food Images, Proceedings of the Twenty-Ninth AAAI Conference on Innovative Applications (IAAI-17)) (2017).

[10] Shu Liu,Lu Qi,Haifang Qin,Jianping Shi, Jiaya Jia, Path Aggregation Network for Instance Segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 8759-8768.

[11] Sanghyun Woo, Jongchan Park, Joon-Young Lee and In So Kweon,CBAM: Convolutional Block Attention Module, Proceedings of the European Conference on Computer Vision (ECCV) (2018) 3-19.

[12] Sujata Chaudhari, Nisha Malkan, Ayesha Momin, Mohan Bonde, Yolo Real Time Object Detection, SSRG International Journal of Computer Trends and Technology 68(6) (2020) 70-76.

[13] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection, arXiv preprint arXiv:2004.10934 (2020).

[14] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh, CSPNet: A New Backbone That Can Enhance Learning Capability of CNN (2020) 390-391.

[15] Retrieved from https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[16] Retrieved from https://www.upgrad.com/blog/basic-cnn-architecture/

[17] Food dataset from https://www.kaggle.com/moltean/fruits

[18] Food dataset from https://www.kaggle.com/rahulbhalley/food-101

[19] Aqua-calc Food Volume to Weight Conversions. [Online]. http://www.aqua-calc.com/page/density-table

[20] Actual calorie database [Online] https://www.uncledavesenterprise.com/file/health/Food%20Calories%20List.pdf