



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 5, May 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

 9940 572 462

 6381 907 438

 ijrset@gmail.com

 www.ijrset.com



Food Image Recognition and Calorie Prediction

Lankesh Milind Gedam, Dr.S.D. Shirbahadurkar

P.G. Student, Department of Electronics and Telecommunication, ZEAL College of Engineering & Research, SPPU, Pune, Maharashtra, India

Professor, Department of Electronics and Telecommunication, ZEAL College of Engineering & Research, SPPU, Pune, Maharashtra, India

ABSTRACT:The increasing number of health issues reported due to obesity and overeating, people have become cautious about their diet intake to prevent themselves from the diseases such as hypertension, diabetes, and other heart related problem which are caused due to obesity. As per the data shared by WHO, at least 2.8 million people are dying each year because of being overweight or obese. The important part of any healthy diet plan is its calories intake. Hence, we propose a deep learning-based technique to calculate the calories of the food items present in the image captured by the user. We used a layer-based approach to predict calorie in the food item which include Image Acquisition, Food item classification, Surface area detection and calorie prediction.

In this paper, a web-based application for estimating fruit calories and improving individual's utilization propensities for wellness is developed. The tensor flow is one of the best process to classify the machine learning method. This method is implementing to calculate the food calorie with the help of a Convolutional Neural Network. Calories calculation in food is now a common task. We use a machine-learning-based approach to predict the calories from food images. Our system only require an image of the food item. First, the type of the food item in the image is identified. Second, the size of the food item is estimated in grams. Finally, by considering both the phases of output, the amount of calories in the food item is predicted. These three phases are based on supervised machine learning.

KEYWORDS:Mask R-CNN, ROI (Region of Interest), IOU (Intersection Over Union), Resnet 50Food Computing, Image Recognition, Smart Food Logging

I. INTRODUCTION

It is very important in today's time that people should be aware of what they are consuming and what will be its impact on the body. So, a system that can help individuals to maintain their calories intake is very important. Most of the world's population live in countries where overweight and obesity kills more people than any other health disease. The problem here is not about having enough food, it is about the people not knowing what is in their diet. If people could estimate their calorie intake during a day, they can easily decide on the number of calories they want to consume. However, managing calorie intake is a very cumbersome task which involves the people to manually keep a track of food item they have consumed throughout the day and they must determine the calories they have consumed. This process is not only manual but also inaccurate as the calorie estimation not only depends on what you are eating but also depends on how much are having.

With the advancement in the field of image processing techniques, the image recognition models are in demand. Researchers are aggressively deploying image recognition model for various uses such as self-driving cars, cancer detection, video frame analysis etc. Researchers have also shown keen interest in predicting the calories present in the food item with the help of the image. Researchers have used various machine learning and deep learning techniques to perform the task of calories estimation with the help of supplied images. Manal Choker and Shady Elbassuoni proposed a solution in which they used the supervised learning technique to perform the single food classification and its calorie prediction. They used the model which takes an image of the food item as input and provides the calorie as output. They developed a MathWorks Image Process tool which was used to extract features from the image. Extracted features were then compressed and fed to a classifier and regressor to identify the food type and determine the size of the food item respectively. Finally, the output of both classifier and regressor is fed to another regressor



which provides the calories as the output. Parisa Pouladzadeh¹, Abdulsalam Yassine¹, Shervin Shirmohammadi developed a deep learning-based model which takes a food image clicked from the mobile camera which is capable of estimating calorie for the mixed portion of the food item as well. The dataset used contained 3000 images clicked under different condition with different camera model and then the clicked image is given as input. They used color segmentation, k-mean clustering, and texture segmentation tools. They employed Cloud SVM and deep neural network to increase the performance of the image identification model. For calorie prediction they used the reference object approach wherein they mandated the presence of the thumb in the image so that their model can use the thumb present in the image as reference for the size estimation of food item present in the image which helped in calorie estimation. In earlier researches, the researchers have collected 101 different food images such as Chicken Wings, Tacos, Bread pudding etc. The dataset has 101 food categories with 101000 images where most of the images contains mixed food items. They used 750 images for model training purpose and 250 images for testing purpose. They used model based on Random Forests to mine discriminative visual components and efficient classification of the images. However, they used the model for the image classification only.

In recent years, due to a rise in health consciousness, many mobile applications for recording everyday meals have been released. Some of them enlist fruit image recognition, which estimates not only fruit names but also fruit calories. It's important to recognize that high-fat fruits have greater calorie density since a gram of fat has over twice the calories of a gram of protein or carbohydrate. Fresh fruits are not sold with nutrition facts. Whether the goal is to limit carb intake, count calories, or simply try to eat more whole fruit, the calorie chart will pinpoint which fruits best fit into your healthy eating plan. Convolutional Neural Networks (CNN) is one class of deep neural networks. Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input which helps to interpret the data such as images, audio, and text. The concept of Deep Learning arises from the study of Artificial Neural networks, Multilayer Perceptron which contains more hidden layers is a Deep Learning structure.

II. RELATED WORK

MATERIALS AND METHOD

Overview

Our model takes Image of the food items as input and give calories of the food item as output. In this there are number of middle process which are done to achieved to this. Firstly, food item whose calorie need to be predicted is identified in the captured image. After the food item identification its size and volume are determined and at last food calorie is estimated to 128*128 pixels before they are used in the model. Mask R-CNN Algorithm is used for image recognition and calorie prediction is done using approximate proportion approach format.

Dataset

Our dataset has six different food items. Dataset is custom generated by selecting food item image from internet and by selecting images from the existing dataset such as Food 101. Food Item are Banana, Pizza Toast, Orange, Idle, Hot Dog, Omelette.

Food Item	Image Count
Banana	113
Pizza Toast	104
Orange	101
Idle	113
Hot Dog	102
Omelette	105

Fig:-Data set Details

As images are gathered from different sources it is important to scale them, so they are scaled to 128*128 pixels before they are used in model. Also images are manually annotated using Pixel Annotation tool and masks are created for model training purpose.

Food-Item Identification

We use instance segmentation to create a pixel wise mask of object in the image. This technique gives us more granular understanding of the food data in the image. Here we use Mask R-CNN for image segmentation which used ROI and IOU to generate bounding boxes, provide labels and mask.As in figure 2. Bounding box is created across the identified food item and label omelet is assigned to the food item.



Fig.2: Mask output of omelet

Mask R-CNN is a deep neural network aimed to solve instance segmentation in machine learning. It separates different objects in an image or a video and gives out the objects bounding boxes, classes and masks. There are two stages in mask R-CNN first is to generate a proposal about the reason where there might be an object based on the input image second is to predict the class of the object, refines the boundary boxes and generate mask in pixel level. Both stages are connected to the backbone structure which helps in feature extraction. Here we are using Resnet 101 as backbone. Mask R-CNN Model is initialized with pretrained weights.

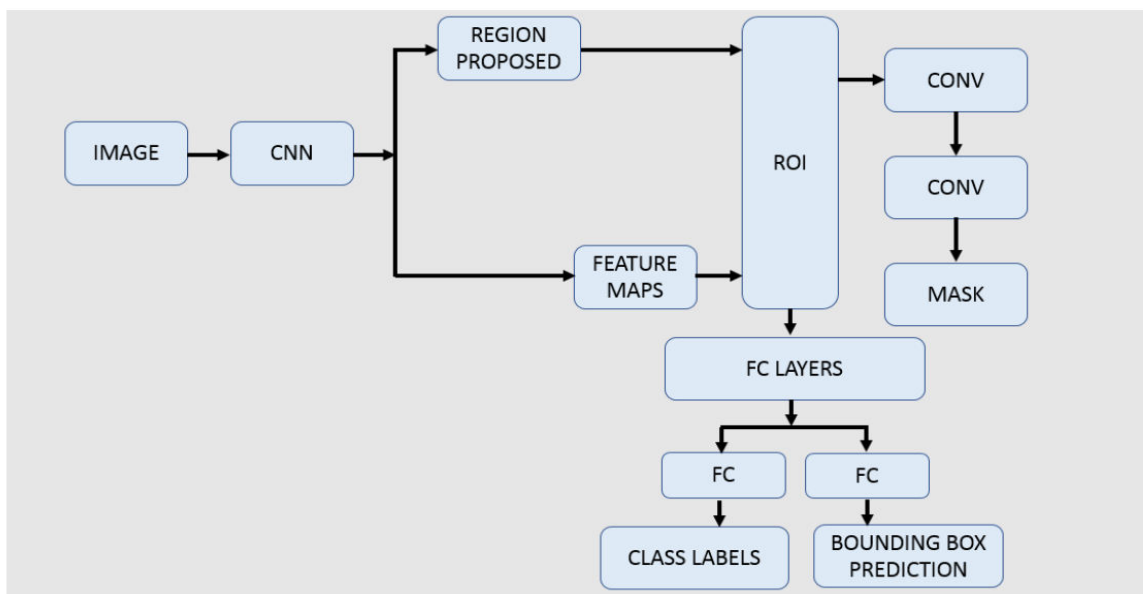


Fig.3: Mask R-CNN Model



To train a Mask R-CNN based image recognition model, we need many images to train deep learning models. Since the dataset we collected is not large enough to train the model, we used transfer learning approach from matterport repository. With the help of transfer learning, instead of training a model from scratch, we started with a weights file that is been trained on the COCO dataset. COCO dataset contains lot of images (~120K), so the trained model weights have already learned a lot of the features common in natural images, which really helps.

Food Calorie Prediction

As the same food can be taken at different depths to generate different picture sizes, we need a method to calculate calorie or estimate the size of the food in a real-world scenario. After we get the desired food items detected along with their masks, we need the real object sizes, which is not possible through a pin-hole camera image alone. So, we take a referencing approach that references the food-objects to the size of the pre-known object to extract the actual size of the food contained in that specific image. For food calorie prediction, method used is approximation of proportions. In this approach calorie per mask of the food class is taken as reference for predicting calorie of the input image. A spread sheet has been prepared containing “Class”, “Calorie Per Mask”, “Minimum Calories”, “Maximum calorie”.

Food	Calorie-per-mask	Minimum-Calorie	Maximum Calorie
Banana	0.02739	72	135
Hot Dog	0.0375	100	290
Omelette	0.01725	50	80
Orange	0.0385	130	320
Pizza Toast	0.051555	200	242
Idle	0.00942	25	40

In Proportion Approximation Approach each image has been scaled to the size of (128*128) and segmented area of the food class is calculated and further multiplied with “Calorie per Mask “.

IV. METHODOLOGY

1. WORK

Food computing has evolved as a popular research topic in recent years. Thanks to the proliferation of smart phones and social media, a large amount of food-related information (such as food images, cooking recipes, and food consumption logs) is often shared online. This has led to us having access to rich heterogeneous information for several important tasks. In particular, this gives the community an opportunity to conduct extensive analysis of food-related topics. Effectively utilizing food computing impacts our lives in multi-faceted ways, including our behaviour, culture, and health. In the past, such analysis has led to impact in medicine, biology, gastronomy, agronomy, etc. One of the most important tasks is food image recognition using deep learning. Many of these techniques rely on the recent success of deep learning for visual recognition, and use these state-of-the-art models to train a deep convolution network that can recognize a variety of food items. Using their feature extraction ability, researchers adapt pre-trained ImageNet models to their own food datasets. Among various efforts to build food image recognition models, Food AI has been trained on the largest food dataset for recognition tasks, with almost 400,000 images.

In addition to Food AI, there are several existing commercial and academic food image recognition systems which can be employed to reduce the burdens of traditional mobile food journaling. These include Calorie Mama 5 , AVA6 , Salesforce Research’s Food Image Model7 , Google Vision API8 , Amazon Rekognition 9 , and many others. To the best of our knowledge, Food AI is the most comprehensive food image recognition solution, with an ability to recognize over 756 different visual food categories (over 1,166 food items), specifically covering a wide variety of Southeast Asian cuisines and Asian cuisines in general.



2. FOOD-AI

We now present our proposed Food AI. We first describe data collection challenges, model training and addressing class imbalance issues. Then, we present extensive experiments and qualitative analysis to shed light onto the insights for deploying such a technology in the real world. We obtain insights from the development environment (analysis on our data collected for training), and the production environment (analysis of query data from users).

2.1 Constructing Food Image Dataset

During the early days, the primary objective was to develop a robust dataset of several food images. Special attention was given to food items commonly consumed in Singapore. We first defined 152 "super categories" representing generic types of foods and drinks. Some of these super categories are: Beer, Fried Rice, Grilled Chicken, Ice Cream, etc.. For each of these super categories, we identified several food-items that likely belong to the category. For example, the Fruit super category would have pineapple, jackfruit, and lychee as the food items. In this manner, we identified a total of 1,166 different food items. Ideally, this is the total number of classes. However, it turns out many items are not visually distinguishable (e.g., coffee with sugar and coffee without sugar are visually indistinguishable, but the difference is important as it significantly affects the users' total caloric intake). Thus, we introduced a notion of Visual Foods as a way to further group the items in a specific category according to their visual similarity. By visually inspecting the images and consulting with domain experts, we merged the 1,166 different food items to 756 visual food categories. We also added a category for non-food items, for which we randomly sampled about 10,000 different images from ImageNet dataset.

The prediction model would be trained to make predictions on these visual food categories (or classes). In the case where finer grained predictions were needed at the application level (e.g., during food logging), the users would have the option to manually select a sub-item from the prediction made. For example, given an image of coffee beverage, the model would predict Coffee With Milk and the user can further choose whether it was With Sugar or Without Sugar. As our target market was primarily Singapore, we laid special focus on foods commonly consumed in Singapore. Out of the 152 categories and 756 visual food categories, 8 categories (e.g. Indian, Chinese, Desserts, Malay, etc.) and 100 visual foods are tailored to this purpose. During the dataset collection, we ensured that we had at least 500 images for each of these 100 visual foods. The images were collected by crawling from Google, Bing, Instagram, Flickr, Facebook and other social media, for each of the food categories. These were manually vetted to confirm whether each image crawled indeed belonged to the food class being searched for. Based on requests from the stakeholders and our continued efforts to improve Food AI, we continue to update the dataset by including new images and visual food classes. At present, we have 756 visual foods, comprising about 400,000 images in total. We have a minimum of 174 images and maximum of 2,312 images per visual food. We then split the dataset into train, validation, and test data, for training and evaluating the model. The dataset (henceforth FoodAI-756) is summarized in Figure.

2.2 Food Annotation Management System (FAMS).

The original image crawling process was labour-intensive and inefficient, requiring users to manually query a search item and select appropriate images to download a given food category. To alleviate these and streamline the process, we developed Food Annotation Management System (FAMS), a web-based tool for automatic crawling and annotating food images. An annotator will define and submit a list of keywords (i.e., food items) through FAMS. The back-end crawler will then retrieve several thumbnail images and present them to the annotator in FAMS. The number of images retrieved is based on the input provided by the user. The annotator is then able to view several thumbnails and is able to quickly identify relevant images for the given keyword (e.g., by checking all, and unchecking the few irrelevant images). After this, the annotator can confirm the labelling and the full-size images are automatically downloaded and saved to the database. An example of collecting images for Orange Juice is shown in Figure . There are two important roles in FAMS, manager and annotator. The manager manages the annotation processes and assigns annotation tasks to one or more annotators. Annotators complete the tasks assigned by the manager. After the manager has confirmed the annotation results, FAMS initiates the download of the full-resolution images from various sources to its backend. Finally, the new annotated images are merged with the existing data to form a new version of the training set.

2.3 Training the Model

With the growing success of deep learning for visual recognition, we use a deep convolution network as the model to recognize food images.



2.4 Transferable Features for Image Recognition

In recent years, it was observed that using networks pre-trained on the ImageNet dataset and transferring those to other datasets provided a significant boost to performance than training new models from scratch. This was due to the ability of Deep Convolution Networks to learn general features applicable to several computer vision tasks. Following this success, we use pre-trained ImageNet models and fine-tuned them on our food dataset. During the course of this project, we have tried several models, and updated them as new state-of-the-art models that have been invented. During the earlier iterations on Food AI we tried older models such as Alex Net, VGG, Google Net. In this, we focus our attention to more recent models and report their performance: ResNet, ResNeXt and SENet. We also considered Dense Net, but found its performance comparable to ResNet models. During training, we followed the standard approaches for data augmentation such as rotation, random crop, random contrast, etc.

2.5 Class Imbalance in the Dataset

A specific problem we face in the Food AI dataset is the extremely imbalanced data, i.e., there is huge variance in the number of instances per class (see histogram of instance distribution in Figure). This imbalance would induce a bias in the model favouring a better performance to those classes with more data, even if those food items are relatively easy to classify. To address this issue, we modify the traditional cross-entropy loss to focal loss, during the training phase. This loss will dynamically vary the scale of the loss of the instances such that the focus is more on the difficult examples during training. Specifically, instead of using cross-entropy we modify the loss as:

$$F L(p_t) = -\alpha (1 - p_t)^\gamma \log(p_t)$$

Here α is a factor that balances the weightage of samples from different classes, and $\gamma > 0$ is the focusing parameter, which regulates the importance of the sample based on its ease of classification. If the sample is easy to classify, its importance gets reduced.

3. System Architecture and Deployment

After having trained the model at the backend, we deployed the Food AI model for production. The System Architecture. Food AI has been deployed as a RESTful web service accessible via HTTP/HTTPS protocol. At the front end, the client is platform and language independent. It can be a mobile app, web application, or desktop application subject to clients' business requirement. At the backend, the Apache Nginx web server is responsible for receiving user requests, redirecting to the UWSGI web application server. The UWSGI calls Caffe inference engine to get the food prediction scores and returns it to the Nginx web server, which returns the response to the user. Food AI web service is written in Python using Flask framework. Food AI web service mainly provides two interfaces, classify and feedback. The classify interface receives HTTP/HTTPS request, including either the URL of image or image data, saves data and image to database and returns the classification results. The feedback interface receives the user feedback about the classification result and saves to database. In Food AI, we use MongoDB as database. The classification interface supports both GET and POST methods. The feedback supports GET method only. To perform food image classification, the UWSGI sends image data to the Caffe inference engine. The Caffe inference engine is written in C++ and hosted on a web server. The web server loads the Food AI model at startup time and makes inference in the GPU mode. To facilitate the cross-language communication between API service and inference engine, we use the Apache Thrift framework. The Apache Thrift framework is a scalable cross-language service solution that works efficiently and seamlessly among a various range of languages, such as Java, Python, C++ etc. An overview of the entire deployed architecture is shown in Figure.

3.1 User Experience

To use Food AI web service, users are required to register their interests on Food AI website (www.foodai.org). After the requests are approved, an API key is assigned to them. The API key is required to validate user's identity for using Food AI service.

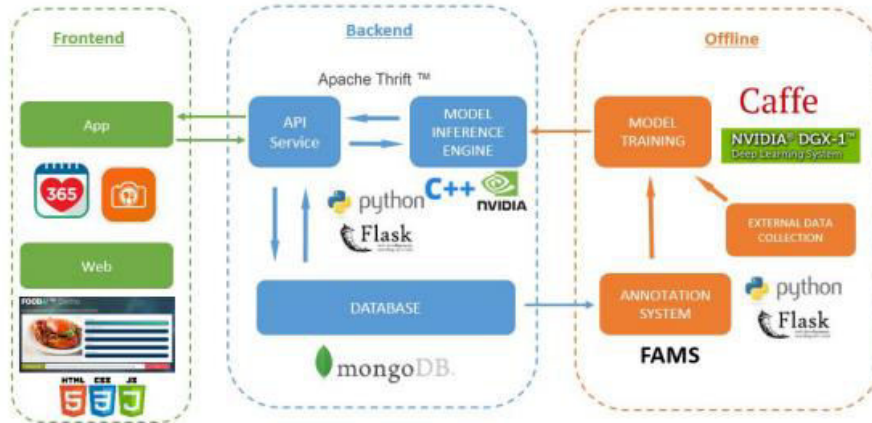


Fig:-System architecture for food AI

Users can integrate FoodAI web service in their application or a system independent from the platforms. The API documentation can be found at FoodAI website. The response in Json format is returned to the user.

The Json object contains the attributes as follows:

- food result: a list of top 10 visual food name sorted by the classification score.
- food results by category: a list of top 10 super category sorted by the classification score
- non-food: an indicator to show if the image is not a food
- qid: query id of the request
- status code: an indicator of the response status
- status message: a descriptive message based on status code
- time cost: the time spent on inference

V. EXPERIMENTAL RESULTS

4.1 Evaluation of Model during Development

Here we present the results of Food AI in the training phase, where we evaluate the performance on test data of the original dataset. Performance of Fine-Tuning Pre-Trained ImageNet Models. While several models have been explored over the course of Food AI development, in this paper, we present the performance of ResNet50, ResNet-101 (50-layer and 101-layer ResNet), ResNeXT-50 (50 layers), and SENet trained with ResNeXt-50. We present the results of the basic models in Table . Among the models, the best top-1 accuracy of 80.86% and top-5 accuracy of 95.61% was obtained by a combination of SENet with ResNeXt-50. ResNet-101 did not do very well (possibly due to convergence challenges). We also look at the inference speed of the models and see that we can make predictions at the rate of 80-120 images per second for the 50-layer models, or 1 image in 0.01 seconds. This is fairly fast and the end-to-end inference result returned to the user thus depends on the round-trip latency of transferring the image to our server and getting the result back. The models occupy close to 100MB for the 50 layer models. Since this model is going to be stored only on the server, the model does not cause a memory constraint.

Performance After Incorporating Focal Loss. We present the results of training the model with focal loss in Table . Here, due to the usage of focal loss, we have improved the convergence of the model to a better optimum. We obtain a top-1 accuracy of 83.2%, achieved with a combination of SENet and ResNeXt-101, outperforming the previous best of 80.86%. This demonstrates the ability of focal loss to improve the performance on imbalanced datasets like the FoodAI-756 dataset by dynamically changing the scale of the loss during training and giving less importance to easy examples. Insights from Performance on the Test Dataset. Next, we look closer at some of the results so as to obtain new insights. Specifically,



Table 1: Performance of various models on food AI-dataset. The models were pre-trained on image net and fine tuned on our dataset.

Network	Top-1 Accuracy	Top-5 Accuracy	Testing Speed (img/sec)	Model Size
Res Net -50(14)	0.7870	0.9427	80	96 MB
Resnet-101(14)	0.7645	0.9366	32	168 MB
ResneXt-50(38)	0.7898	0.9473	112	94 MB
Senet ResneXt-50	0.8086	0.9561	122	103 MB

we focused on the most misclassified instances. Our objective was to understand why these specific classes would get misclassified. Was it a possible short-coming in our training strategy? Was it that the data was just too difficult or noisy? Was it that many of these classes looked visually very similar? Or possibly a combination of any of these factors. We show some of these highly misclassified results in Table . In many of the cases, from the visual food name, we can see that some of the items and their predictions have very similar ingredients, which leads to confusion. In particular the first row in the table shows the same dish with soup predicted as being without soup (dry).One possible explanation of this is that our dataset has more instances of the dry version of the dish than the soupy version. Another interesting case is the last row: Mee Kueh, and Mee Rebus. See Figure for images of both classes, where we can see that they look very similar. At the time of dataset collection, we assumed that these should have been different categories. Upon further research, we found that these categories are often considered as the same item. This suggests merging them into a single visual food. Performance of the Model. How do we measure the model performance in the real world? One approach could be to manually annotate data that has been queried, and compare this against the model predictions.

This approach is extremely expensive for two reasons: (i) It requires substantial manual effort, which would be time-consuming; and (ii) This labelling process requires an expert who is familiar with all 756 visual food categories (for a variety of cuisines) and finding/hiring such experts for labelling is not an easy task. Note that annotating these images is significantly harder than collecting them. This is because during data collection, we just query a keyword and retrieve several images immediately which requires a cursory look to confirm. In the case of user queries, the annotator has to look at one image and assign it one of the 756 categories (some of whom are visually very similar). Another approach to measure model performance is based on the feedback given by the users. Despite not receiving feedback for all queries, this is a useful indicator. Based on the feedback, we score an accuracy of about 50% in top-1 accuracy, and about 80% in top-5 accuracy. There could be several factors contributing to a worse performance in the real world than on our test dataset. We hypothesize the following possible (not exhaustive) reasons:

- **Domain Shift.** Possibly, the distribution of query data and our training data is different. Our data is relatively cleaner, and has higher quality photos in comparison to the real world photos taken by the users. This domain shift may cause a degradation of performance of the model.
- **Poor Quality of Data Query.** Closely related to domain shift is the poor quality of image queries. Many users may submit queries which are of poor quality (e.g. upside down images, food is mostly consumed, etc.). These images would result in a poor performance of the model.
- **Different Imbalanced Distribution.** As noted before, the FoodAI-756 dataset is highly imbalanced, creating a bias in the model learnt. This bias may affect model performance in the real world, where the distribution of instances per class queried may be different from that in the train dataset.

Visual Food	Recall	Most common incorrect prediction
Mushroom & minced pork noodles soup	0.2	Dry minced pork & mushroom noodles
Vegetable u main	0.24	Dry ban mian
Stewed taupok	0.28	Bak kut tch
Tauhu goreng	0.3	Fried tau kwa
Pork chop western set	0.32	Chicken chop
Tikka	0.4	Chicken curry
Beef ball soup	0.42	Beef ball kway teow soup

Fig:-table -Difficult cases in the food AI test dataset



Fig:- Sample food images

• **Poor Feedback Quality.** It is possible that users may not have the knowledge of food item and may give an arbitrary feedback. They may also be intending to "play" around with the technology and intentionally giving a wrong feedback. Next, we will explore some of these factors through case studies. Analysis of User Queries. We first look at some of the queries sent by users, and identify some of the key properties. We show several examples in Figure , where we have categorized them into 8 categories based on the associated challenges. While several queries are easy to classify (A), there are many challenges including inter-class similarity (B), intra-class diversity (C), incomplete food (D), non-food (E), poorly taken photos (F), multiple food items (G), and unknown foods (H). Detailed descriptions are in the caption. Query Images vs Our Dataset. Here we present a couple of case studies based on Food AI behaviour we wanted to investigate.

VI. CONCLUSION

In this paper, we have used the deep learning based modal to predict the total calories of the food item present in this image . to develop this solution, we used mask r-CNN technique to create mask and bounding boxes. This in turn helped the model to calculate the surface area occupied by the different food items in the image which further facilitated the model with the ability to satisfactorily predict the calories associated with each food item. Calorie are estimated with the help of mathematical formulas which compares the proportion of the image-occupied by each food item and determines the calories associated with it. in future work, we plan to extend the scope of model by increasing the ability of the model to identify a greater number of images of food items instead of 6 food item, which we used in our current dataset. The dataset we used contain 638 images of food item with 6 different categories, which will be extended in our next model. Finally, we would like to calculate the calories of the food item based on their volume with the help of 3D images. As the image as input and predict the calories of the food item with better results.

REFERENCES

1. Yash Chugh, Reema Gupta, Ranjeeta Kaushik, (IJETMAS) International Journal of Engineering Technology, Management and Applied Sciences, "Image Enhancement Using Morphological Operators," Volume 3, Special Issue, ISSN 2349-4476, May 2015.
2. Kiran Ambhore, Prof. N.A. Dawande, (IJARCCE), International Journal of Advanced Research in Computer and Communication Engineering, "Measuring Calories and Nutrition from Food Image," Vol. 5, Issue 6, June 2016.
3. Tejwant Singh, Mr..Manish Mahajan, (IJARCSEE), International Journal of Advanced Research in Computer Science and Software Engineering, "Performance Comparison of Fuzzy CMeans with Respect to other Clustering Algorithm," Volume 4, Issue 5, May 2014.
4. World Health Organisation <https://www.who.int/features/factfiles/obesity/en>.
5. K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017.
6. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.



7. Bossard L., Guillaumin M., Van Gool L. (2014) Food-101 – Mining Discriminative Components with Random Forests. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8694. Springer, Cham.
8. Ciocca, Gianluigi & Napolitano, Paolo & Schettini, Raimondo. (2016). Food Recognition: A New Dataset, Experiments, and Results. IEEE Journal of Biomedical and Health Informatics. PP. 1-1. 10.1109/JBHI.2016.
9. Ge, ZongYuan & McCool, Chris & Sanderson, Conrad & Corke, Peter. (2015). Modelling local deep convolutional neural network features to improve fine-grained image classification.
10. Cao, Yuanzhouhan, et al. "Exploiting Depth From Single Monocular Images for Object Detection and Semantic Segmentation." IEEE Transactions on Image Processing, vol. 26, no. 2, Feb. 2017, pp. 836–46. DOI.org/27
11. Chokr, Manal and Shady Elbassuoni. "Calories Prediction from Food Images." AAAI (2017).
12. Pouladzadeh, Parisa & Yassine, Abdulsalam & Shi Mohammadi, Shervin. (2015). Food: Food Detection Dataset for Calorie Measurement Using Food Images. Lecture Notes in Computer Science.
13. Food recognition: a new dataset, experiments and results (Gianluigi Ciocca, Paolo Napolitano, Raimondo Schettini) In IEEE Journal of Biomedical and Health Informatics, volume 21, number 3, pp. 588-598, IEEE, 2017 [11] He Y, Xu C, Khanna N, Boushey CJ, Delp EJ. FOOD IMAGE ANALYSIS: SEGMENTATION, IDENTIFICATION AND WEIGHT ESTIMATION. Proc (IEEE Int Conf Multimed Expo). 2013
14. Estrada, FJ. & Jepson A. d int J Comput Vist (2009)
15. Liang, Yanchao & Li, Jianhua. (2017). Computer vision-based food calorie estimation: dataset, method, and experiment. <https://arxiv.org/pdf/1705.07632.pdf>.
16. P. Pouladzadeh, S. Shirmohammadi and R. Almaghrabi, "Measuring Calorie and Nutrition from Food Image", IEEE Transactions on Instrumentation & Measurement, vol. 63, no. 8, pp.1947-1956, August 2014.
17. World health statistics 2012, Available: <http://www.who.in/gho/publications>
18. Obesity study, October 2011, Available:- <http://www.who.int/mediacentre/factsheets>



INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.423



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 **6381 907 438** **ijirset@gmail.com**



www.ijirset.com

Scan to save the contact details