ORIGINAL ARTICLE

MyPlate: A Diet Monitoring and Recommender Application

Anusha Achuthan[1], Shu Hui Yii[1], Ali Fawzi Mohammed Ali Alkhafaji[2]

[1] School of Computer Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia.
[2] Department of Computer Sciences, College of science, University of Baghdad, Baghdad, Iraq.

ABSTRACT

Introduction: Diet monitoring is of great interest in practicing a healthy diet to ensure good health by keeping track of the type and amount of food consumed. The food recommender system could provide personalized suggestions about food choices that best suit user preferences, making diet monitoring a valuable tool for maintaining good health, preventing diseases, and achieving a balanced lifestyle. Methods: This article presents a diet monitoring and recommender Android mobile application utilizing a Mask Region-based Convolutional Neural Network (Mask-RCNN) and portion analysis with a content-based filtering recommender algorithm. The application, named MyPlate, will provide a daily calorie limit and nutrition guidelines based on the user profile record, such as current and ideal weight. At the same time, MyPlate can also perform food image recognition and portion analysis from the picture captured and estimate the total calories. Finally, MyPlate provides a personalized food intake recommendation based on the previous intake record. Results: The developed MyPlate application can achieve accurate food image recognition, with the mean average precision at 0.825. Besides that, the proposed MyPlate application can perform portion analysis and provide food recommendations based on intake history at 80% similarity. Conclusion: The proposed MyPlate Android application has shown the effectiveness of incorporating machine intelligence into a diet monitoring technique that can promote healthy eating habits in the community.
Malaysian Journal of Medicine and Health Sciences (2024) 20(2): 196-203. doi:10.47836/mjmhs.20.2.26

Keywords: Image Recognition, Portion Analysis, Recommender Application, Mask Region-based Convolutional Neural Network (Mask-RCNN), Content-based Filtering

Corresponding Author:
Anusha Achuthan, PhD
Email: anusha@usm.my
Tel: +60164753029

INTRODUCTION

Food is an essential part of everyone's lives, and it is listed in a modern list of six human needs that benefit from nearly seventy-five years of psychology, neuroscience, and sociology research (1). However, today's world has adapted to a diet practice that has several harmful effects on human health. Modern lifestyle changes and globalization have compelled humanity so much that one has no time to pay attention to a healthy diet and forced us to consume fancy, high-calorie fast food and processed meat (2). On the other hand, some people with busy schedules will always skip meals and have irregular meal timing, leading to a poor diet. Recently, modern research into the effects of poor diet has given insight into avoiding them, but unfortunately, the measures taken are not as effective as expected (3).

Apart from that, The Ministry of Health Malaysia has developed the Malaysian Healthy Plate (4) that emphasizes the Quarter-quarter Half concept to serve as a meal intake guide to Malaysians, but most of us are not following it. When someone is practicing unhealthy eating habits such as under and over-eating, it can contribute to illness and other health problems. Hence, user-friendly diet monitoring systems are highly required to serve as a guideline to track diet progress and monitor weight loss goals. There are some diet monitoring mobile applications that are available in the market. HealthifyMe is a mobile application that receives input in the form of keywords, voice, and snap (5) to track the calories and food taken. In this application, the user must tap on the food item in the image that is intended to be tracked and add it one by one to the list. It also enables the user to track the water drink and comes with a virtual assistant Ria, to assist the user. However, HealthifyMe could not provide the user with a free customizable diet plan, and its auto-generated meal suggestion also does not strictly follow the user's intake record. Calorigram is another mobile application designed to help users record their calorie intake by searching using keywords or by scanning the bar code on the food packaging. For now, Calorigram is still unable to track the calorie in Asian food such as dimsum. Moreover, Calorigram only provides statistics charts on calorie intake, macro balance, and weight tracker for free, and users must upgrade their profile for more advanced features. Calorigram also does not provide food intake suggestions.

Calorie Mama is another mobile application that allows users to record the calories in their daily meal intake by searching the keywords or taking a photo of the meal. It comes with a simple and clean user interface and is easy to use (6). However, some Asian food, such as nasi lemak, is missed, and there is no report generated for users on what they consume along the progress. Besides that, the free meal plan available on Calorie Mama is fixed and is not customizable based on user needs and preferences. Eventually, the user is asked to pay for full access to detailed nutrition information and food intake recommendations. Out of so many diet monitoring applications, none provide portion analysis with respective calories of the food from the picture taken and simultaneously provide a free personalized food intake recommendation. The users must input the food taken manually or by scanning the bar code which is unavailable on home-cooked food, to record the calorie intake. Besides that, the user needs to set the food portion manually, and the user does not know whether the meal taken contains balanced nutrition. These will result in inaccurate diet monitoring, and the user will not benefit from it.

In order to understand and develop a relevant diet monitoring application, a preliminary brainstorming, survey, and a short interview with potential users are performed. First, the brainstorming generates rough ideas on how the mobile application should be delivered. Next, a survey on the diet monitoring process is carried out in electronic form using Google Forms. It is a basic and free-to-use tool to gather responses from the targeted users. According to the collected data, an analysis is made to verify the required information needed for the system development.

Additionally, research on vendor solutions with similar situations has been done. From the available applications in the market, we can conclude the requirements of users using the reviews and comparisons made to figure out the improvement. At the same time, short interviews are undertaken with potential system users. A survey alone is not enough to fully understand user requirements. Therefore, short interviews and online one-to-one conversations are made through Webex to understand the user requirements further and gather useful insight into what they expect from the system. Lastly, the results obtained from these approaches are combined, filtered, and organized to determine the system's user requirements to meet the business and users' needs.

From all the surveys and analysis, the Diet Monitoring and Recommender Application MyPlate is proposed to overcome that problem by utilizing the mask region-based convolutional neural network (Mask-RCNN) model with image recognition abilities to count the calories in the food consumed based on the portion of food recognized and then provide recommendation on the food intake by using the content-based filtering recommendation algorithm. With a vast Asian food database, MyPlate allows user to track their intake record accurately and effectively monitor their diet progress.

MATERIALS AND METHODS

This study proposes a new method for diet monitoring and a recommender application called MyPlate using deep learning, portion analysis, and a content-based filtering recommender algorithm. MyPlate is an Android-based mobile technique accessed with an internet connection. This application has three main modules, including the user module, food recognition module, and recommendation module. This application will utilize the Mask Region-based Convolutional Neural Network (Mask-RCNN) approach to perform the food image recognition and portion analysis, while a content-based filtering algorithm is adopted to perform the food recommendation. The recognition and recommender model is hosted on the local server and is connected with the mobile application through the network via Flask API. The database server is a Firebase cloud computing development platform maintained by Google (7). Fig. 1 shows the system architecture design of MyPlate. Following the sequence indicated by the numbering in Fig. 1, first, the user login to the application by sending a login request to the database for Firebase authentication through the internet. If the user is authenticated, the response will be directed back to the user notifying the login is successful. For the image recognition and portion analysis, the user must upload an image from the mobile device to the network and send it to the Mask-RCNN model hosted in the local server via Flask API. Then, the obtained result is sent to the database to save as the intake record. In the recommendation module, first, the diet intake record of the user is retrieved from the database, then recommended food is returned from the server via Flask API by using the food consumed as the input.

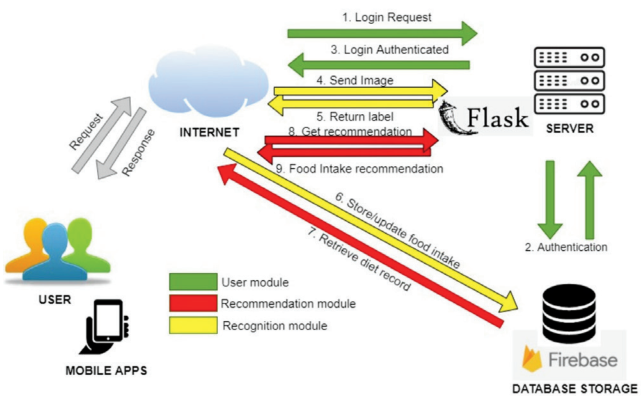

Figure 1: Architecture design diagram for MyPlate.

The Firebase Realtime Database is stored as JSON objects. For the User Interface Layer, Android Studio View and ViewGroup are suggested to interact with the user to display output and receive input. The activity or fragment running on the ViewModel is proposed to interact with the domain layer. In the Domain Layer (Business Logic), the Mask-RCNN model and recommender algorithm process the input data and display output to the user. The data flow between the model and the Android application is conducted via the API developed by the Flask framework. For the Data Access Layer, the NoSQL database is used in Firebase to store all the data. The Firebase Authentication is proposed to store user login information such as email and password and generate user identification.

Functional Requirements

The main functionalities involved in this USM exergame application are summarized in Table I.

Table I: Main functionalities of MyPlate

Functional Requirements	Description
Login	Allow users to login to their account if it exists.
Register new account	Allow users to register a new account.
Edit user profile	Allow users to edit user profile information such as password, current weight, ideal weight, and age.
View report	Allow users to view report generated on diet monitoring process
Track calorie	Allow users to track their calorie intake in each meal, either by image recognition or manually input.
Input image to recognize food and portion	Allow users to upload a picture of food and recognize for the food portion with its calorie content.
Search food by text	Allow users to manually search for the food intake by using text search.
View foods recommendation	Allow users to view the food recommend-ed by the recommendation algorithm in MyPlate.
Set preferences	Allow users to set preferences to obtain a more suitable foods recommendation.

User Module

This module handles user account management, including user information. It keeps the user's account information, such as email address, username, and password. This submodule also records the user's body measurements, such as age, weight, height, body mass index (BMI), diet-related data (i.e., lifestyle), and diet progress. This module is also responsible for account authentication during login and user profile editions. Moreover, the user module enables the visualization of the statistics, such as the food intake over a period of time, the detailed nutrition information on the food, and the diet progress. It allows users to have a better insight into the data.

Food Recognition Module

The Food Recognition Module handles calorie tracking through image recognition or text searching. This module aims to maintain a food database containing all popular and daily Asian food, besides some Western food. In addition, it also handles the storing of detailed nutrition information such as calorie content, cholesterol, protein, fat, sodium, and cholesterol in food. In the text search system, the user can search the food consumed manually and adjust the serving size. This is because there might be a time when the user forgets to take the food image before consuming it, and the user still can track their calorie. This module handles the food image recognition and determines its portion for image recognition. The model will be trained on the food image set and tuned for better accuracy.

Weiqing Min and friends proposed the Ingredient-Guided Cascaded Multi-Attention Network (IG-MAN) (8) for food image recognition. The proposed technique can sequentially localize multiple informative image regions with multi-scale from category-level to ingredient-level guidance in a coarse-to-fine manner. In 2020, Jiangpeng He et al. (9) also proposed a multi-task image-based dietary assessment for food recognition and portion size estimation. The multi-task learning utilized the L2-norm-based soft parameter sharing, cross-domain feature adaptation, and normalization to improve the result.

This study proposes a deep learning architecture called Region-based Convolutional Neural Network (Mask R-CNN) based transfer learning for food segmentation and portion size computation. The Mask R-CNN is an extension of Faster R-CNN designed as the feature extractor for food segmentation, combining object detection and semantic segmentation. It consists of multiple layers, including the convolutional, pooling, and fully connected input and output layers (10). In Mask-RCNN, the exact pixels of each object can be located and localized instead of just bounding boxes, which is useful in this technique for food recognition and portion analysis as the exact pixels of a food can be determined. Technically, the Mask-RCNN model is a supervised learning approach trained on a labeled food image set before conducting the inferring phase. The inputted image will pass through a series of convolution filter layers and pooling, then pass through flattened, fully connected layers to classify an object with probabilistic values between 0 and 1. The label and serving size are then returned to the database for calorie counting and finally returned to the user. The implementation steps for Mask R-CNN are defined as follows:

Data Preparation

The food images with object instances and corresponding instance masks are first collected and annotated. The VGG Image Annotator (VIA) is an image annotation tool used to define regions in an image and label those regions to serve as the training input for Mask-RCNN. The set of food images is then divided into training and testing sets.

Model Architecture
After annotating the dataset, the training process is carried out by reusing the learning weights of the pre-trained network and changing the final output layer to detect the food classes. Mask-RCNN utilizes the dual stages of Region Proposal Network (RPN) and Fast-RCNN to produce the regions of interest for the potential object instances. In the second stage, the produced regions are warped into fixed-sized feature maps using a Pool layer and execute the classification process. The Feature Pyramid Network is then used to advance the standard feature extraction, which passes high-level features to lower layers. After that the Bounding boxes are obtained from RPN, and then object instances are categorized. Finally, Mask R-CNN generates the binary masks for each object, and the class labels for each object are finally predicted.

Loss Functions
The loss functions are used to measure how far are the model's predictions from the true labels. The loss function used for multi-task Mask R-CNN is defined as Loss = LRPN + Lclass + Lbox + Lmask for each Region of Interest. The loss function in this model consists of 4 terms for different tasks:
1. LRPN is the RPN loss used to assess the accuracy of suggested regions.
2. Lclass is the classifier loss used to evaluate the object classification.
3. Lbox is the Bounding box loss used to evaluate the bounding box predictions.
4. Lmask is the Mask loss used to evaluate the mask predictions.

Training
The proposed MyPlate approach is trained and tested on three thousand food images. The training was conducted on a Core i7 CPU with 16 GB RAM, Windows 10, and MATLAB R2019a. The transfer learning is used to initialize the R-CNN with pre-trained weights. The R-CNN model is trained on the training set of the food images using the defined loss functions. We use Adam optimizer to update the model's weights with the default settings of the initial learning rate.

Inference
The unseen tested set of the food images is passed through the Mask-RCNN model to obtain suggested regions, classification, bounding boxes, and predicted masks.

Recommendation Module
The recommendation module in MyPlate handles the recommendations on food intake based on user preferences and the previous intake record. The most related food can be seen as the food intake recommendation by calculating the food's similarities. Users also can narrow down the food list by setting some filters. The content-based filtering method is proposed as the recommendation algorithm to suggest a customizable food intake that best suits the user. Content-based filtering is an algorithm that uses keywords or attributes to describe an item, and the user profile also has these attributes (11). Then, the items are ranked by how closely they match the user attribute profile, and the best-matched items will be recommended. In 2016, Kundan Shumsher Rana proposed a food recommendation system based on a content-filtering algorithm (12). This system enables users to search for a food item using a query box and display the nutritional value using the recommendation algorithm. For implementing a content-based recommender, a CSV file containing all foods' information and tags is prepared and further used as the input file for recommendation.

Dataset
The image dataset of fifteen classes of foods is collected by downloading from online resources such as Google Images and by capturing foods from different angles. The foods include the general type, such as rice and noodles, and other commonly found food, including eggs, bread, and seafood. The dataset contains approximately three thousand food images, around two hundred images for a food class. After that, each image is annotated using the annotator tool, which is the VGG Image Annotator (VIA), by specifying the exact location of each food in the image. Then, these images are divided into training and testing datasets in an 8:2 ratio. Fig. 2 shows some examples of annotated dataset images.

Technology Deployed
This section defines the hardware and software specification, programming language and algorithm deployed during the development of the application. The proposed MyPlate approach was assessed on Intel(R)

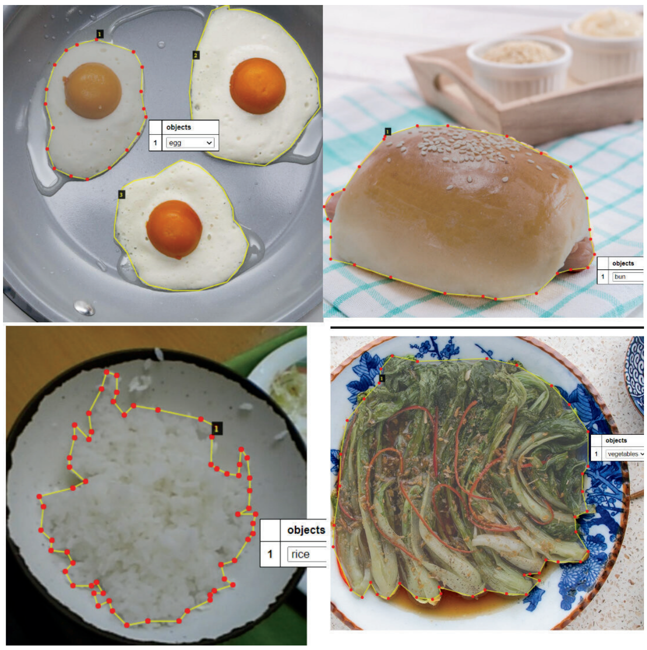

Figure 2: Example annotated image set for MyPlate. Image soure from (13,16)

Core (TM) i5-8250U CPU @ 1.60GHz 1.80 GHz CPU with 4 GB RAM and Windows 10. The specification of the hardware used to run the mobile application can be summarized into three:

- The mobile device must be running on Android operating system
- The mobile device is equipped with a back camera
- The mobile device requires internet connection service

The specification on software specification, programming language, and algorithms deployed during the development of the application can be summarized into the following:

1. Software tools:
- Android Studio is used for all code scripting during the application development.
- Draw.io is used for drawing UML diagrams and other diagrams.
- VGG Image Annotator (VIA) is an image annotation tool is used to define regions in an image and label those regions to serve as the training input for Mask-RCNN.
- Anaconda is a distribution of Python programming language for scientific computing that aims to simplify deployment. The APIs for the Mask-RCNN model and recommender algorithm with Android applications are built and run.

2. Programming Language:
- Java is the main programming language used in the MyPlate Android mobile application.
- Python scripting trains and builds the model for recognition, portion analysis, and recommendation.

3. Library and Framework:
- Firebase is used for user authentication and other serverless services.
- TensorFlow is used for machine learning logic.
- Flask is used as a micro web framework for creating Android Programming Interface (APIs) in Python.

4. Algorithms:
- Mask region-based convolutional neural network (Mask-RCNN) is proposed as the machine learning method to develop MyPlate's main recognition feature, portion analysis, and food recognition. Mask-RCNN, as a class of deep, feed-forward artificial neural networks, has been used widely in analyzing visual imagery, which is suitable for us to recognize the food portion.
- Content-based filtering algorithm is proposed for food intake recommendation.

Contribution Of The Proposed MyPlate Application

This study proposes a new robust diet monitoring and recommender mobile technique, utilizing Mask-RCNN and portion analysis with a content-based filtering recommender algorithm. In the food recognition part, instance segmentation using Mask-RCNN enables MyPlate to segment each instance of food image as a single entity with a masking area besides performing object classification. This masking region has information about the area of the food, which can be further used to perform the food portion analysis by calculating the pixel of each masking area and extracting it with the preset ratio to obtain the portion of each food. The Mask R-CNN detects objects and provides pixel-level segmentation masks. This is crucial for accurately outlining the boundaries of individual food items in an image, which is vital for portion size estimation. With the portion analysis features, users of MyPlate can easily be aware of the amount of each kind of food consumed by calculating the portion of the food in everyday meals. To our knowledge, the Mask R-CNN deep learning model has not yet been introduced for diet monitoring and recommender applications. Here, we confirm the utility of the Mask R-CNN approach at a different spatial scale of food images by producing a new technique to detect and segment food items from plate images. The Mask R-CNN is pre-trained on large image datasets, significantly reducing the number of labeled data needed for training and expediting the model inference process. Therefore, MyPlate can handle diverse types of food items and various serving sizes. Once food items are segmented, the model can estimate portion sizes. This information is valuable for dietary assessment and calorie counting. The proposed MyPlate application consists of all the necessary functions as a diet monitoring application, such as creating a user account, tracking intake records, and viewing diet reports. Besides that, it also enhances vital features such as food recognition from the image captured with portion analysis and provides food intake recommendations.

Table II summarizes the comparison between the proposed MyPlate model and the available diet monitoring applications. In MyPlate, a deep learning algorithm is integrated to provide a better outcome. The combination of multi-layered neural networks and Mask-RCNN allows data flow between nodes and can perform complex tasks in food recognition. In terms of portion analysis, none of the existing applications provide portion recognition, so the user must manually set the food portion. In contrast, MyPlate utilized the Mask-RCNN and OpenCV to recognize the portion of each food recognized. Then, the calorie estimate is more accurate. At the same time, the food intake suggestion provided in HealthifyMe and Calorie Mama is just randomly produced and not customized, which cannot meet user preferences. However, in MyPlate, the effective recommendation algorithm, which is content-based filtering, is used to recommend food intake. Therefore, the recommendation is suitable for the user and can promote health.

RESULTS

The proposed MyPlate model is evaluated qualitatively and quantitatively to ensure the modules of the application work as expected. Unit testing is conducted

Table II: Summary of test cases

Test Case Name	Test Procedure	Expected Result	Actual Result	Status / Comment
Insert data into Firebase when registering.	Users create an account on the register page with the correct information.	Register success. The information is saved to Firebase.	As expected.	The information is successfully saved to the Firebase. Passed.
Update user account testing.	The user changes the updated information to the form and clicks to update	The information must be updated in the database and shown correctly on the user profile page.	As expected.	The test is carried out to ensure users can update their information easily. Passed.
Show user progress report testing.	Users choose the week for the progress report.	The intake record charts are fetched from the database and displayed correctly.	As expected	Passed.
Create, edit, and delete intake records with different serving	Users create an intake record of different serving sizes for meals on different days. Update the serving size and delete after that.	The record must be saved to the database and shown on the home page with the correct serving and nutrition information calculated.	As expected.	Passed.
Searching the food by name	Users input a food name or some alphabet into the search bar	If the food data exists and the logic is correct, foods with matching names will be shown.	As expected.	Passed.
Get the food recommendation.	Users click on the meal to get recommendations on it.	The top three food in the record is obtained and sent to the recommender algorithm in the server. The list of results is sent back and displayed without duplication.	As expected.	Passed.
Filter down the recommendation list.	Users select some filter criteria for food recommendations,	The matched food will be displayed.	As expected.	Passed.

for each unit function to check if it behaves as expected. At this level, the testable modules of the MyPlate are tested independently with the testing plan, such as the function of accepting input images. The functionality in each module is tested one by one. For example, in the recognition module, unit testing is carried out to ensure the recognition of foods from input images can be performed successfully. Besides that, in the user module, a series of tests, such as checking with Firebase, login, and register activity, are tested. Table II summarizes the test cases conducted. The following subsections explain these experiments in detail.

Test Cases

The test case consists of data input, the application's conditions, and the expected output. It is used to check whether the mobile application meets the requirements. The test cases are conducted to assess the main functionalities of MyPlate, such as input image and view recommendation. This test case is implemented to confirm that MyPlate enables the user to get the food recommendation.

Test case of updating user's accounts

This test case is performed to verify that MyPlate enables the user to update the user account. Fig.3 illustrates that MyPlate in this test case was capable of (a) the input fields are editable, and there is checking to ensure only valid input is entered. (b) The new account information is saved and acknowledged to the user. (c) The new calorie intake limit is calculated and updated in the database, and the user can view it on the main page.

Test case on view food intake recommendation

This test case is implemented to confirm that MyPlate enables the user to get the food recommendation. Fig.3 illustrates that MyPlate could (d) ensure that the most frequently consumed food items are obtained from

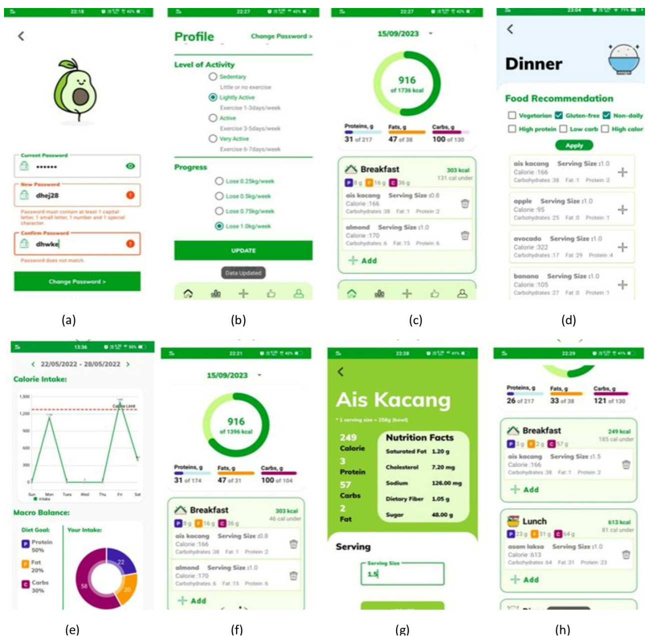

Figure 3: Screenshot of (a) editable input field with validation, (b) update account information, (c) recalculate of calorie intake limit , (d) food recommendations based on the three most frequently consumed food items, (e) view intake progress, (f) all food intake shown correctly on the home page, (g) updated food serving size, (h) deleted food intake record.

the database. (d) Provide the food recommendation based on the most frequently consumed food items is prompted back to the user from the server via Flask API with no error.

Test case on show user progress report

This test case is applied to verify that MyPlate can show user progress report testing. Fig.3 (e) illustrates that all progress reports are correct, and the user can select different weeks for the different reports.

Test case on creating, editing, and deleting intake record
This test case is performed to verify that MyPlate enables users to create, edit, and delete intake records with different servings. Fig.3 illustrates that (f) all intake records show correctly. (g) The food serving size is updated correctly and shown. (h) The deleted foods are no longer saved in the database, and the acknowledgment is sent to the user.

Image Recognition Module
The training and testing have of the image recognition module been performed using the Mask-RCNN model and the portion analysis. Since a custom dataset is needed for MyPlate, transfer learning with Mask-RCNN is carried out in developing the Mask-RCNN model. After annotating the dataset, the training process is conducted by reusing some parts of the pre-trained network and changing the final output layer to detect the food classes. The backbone used is Resnet101, and five epochs are set during the training phase. Parameters such as batch size and learning rate are modified from time to time, and the training outcome is observed to get satisfactory results in terms of accuracy and processing time. The model has been trained with 2 batch sizes and a 0.001 learning rate, resulting in a training loss at 0.332, validation loss at 0.527, MRCNN class loss at 0.108, and mrcnn box loss at 0.225. After obtaining the weight by training, the model is further evaluated by loading the trained weight. Ten random images are chosen for testing, and the mean average precision (mAP) is calculated using the intersection over union (IoU) threshold of 0.5, which is used to measure the overlap of a predicted versus actual bounding box of an object. Seven of the ten random images have been tested correctly, with an accuracy of 0.7 and 0.825 mAP. An experiment on portion analysis has been carried out to validate its performance further. OpenCV module is used together with the Mask-RCNN to calculate the contour mask area of an object. It is then rounded to the ratio and converted as the serving size of the food. A few experiments on different ratios are applied to the images taken by phone camera to determine the suitable ratio that gives the best outcome. All the pictures are resized to 1000 x 1000 pixels before sending to the model. Assuming a quarter of the image is appropriate to 1 serving size as indicated in the Quarter-quarter Half concept. The results of 40 pixels equivalent to 1cm give the most accurate results. Fig.4 shows the sample result of food recognition combined with portion analysis. All the foods are recognized with the bounding box, and the masks are correctly annotated around the localized food. The reading beside the foods' name is the serving size detected.

Recommender Module
A content-based filtering algorithm provides food intake recommendations by measuring the similarity between foods and returns the most related results. Different methods used to calculate these similarities include cosine similarities, Euclidean distance, and K-nearest Neighbour (KNN). Ten foods are fed as the input to the content-based filtering algorithms, and the average results for the top ten outcomes are computed. The Cosine similarity obtained the highest percentage of related outcomes(i.e., 80%); hence, it is selected as the similarity metric. Meanwhile, Euclidean distance and KNN achieve 70% and 60%, respectively.

Figure 4: Food recognition using Mask-RCNN on custom dataset. Image soure from (14-16).

DISCUSSION

The proposed MyPlate application consists of all the necessary functions as a diet monitoring application, such as creating a user account, tracking intake records, and viewing diet reports. Besides that, it also enhances vital features such as food recognition from the image captured with portion analysis and provides food intake recommendations.

Table III summarizes the comparison between the proposed MyPlate model and the available diet monitoring applications. In MyPlate, a deep learning algorithm is integrated to provide a better outcome. The combination of multi-layered neural networks and Mask-RCNN allows data flow between nodes and can perform complex tasks in food recognition. In terms of portion analysis, all the existing applications do not provide portion recognition, so the user must manually set the food portion. In contrast, MyPlate utilized the Mask-RCNN and OpenCV to recognize the portion of each food recognized. Then, the calorie estimate is more accurate. At the same time, the food intake suggestion provided in HealthifyMe and Calorie Mama is just randomly produced and not customized, which cannot meet user preferences. However, in MyPlate, the effective recommendation algorithm, which is content-based filtering, is used to recommend food intake. Therefore, the recommendation is suitable for the user and can promote health.

Table III: Features comparison among MyPlate and existing application

Features	HealthifyMe	Calorigram	Calorie Mama	MyPlate
Food image recognition	✓		✓	✓
Database for Asian food	✓			✓
Detailed nutrition information		✓		✓
Report on statistics		✓	✓	✓
Food portion analysis				✓
Free customisable food intake recommendation				✓

CONCLUSION

The work in this article presented a new diet monitoring and recommender application termed MyPlate, combined with image recognition with portion analysis using Mask-RCNN and OpenCV. The proposed MyPlate model integrates portion recognition with food recognition to track the calorie content and provide food intake recommendations simultaneously based on previous diet records by utilizing a content-based filtering algorithm on the Android platform. According to the online survey, the portion recognition and food intake recommendations have received positive feedback, which is preferable for diet monitoring due to the higher accuracy and effective monitoring. For future work, the application can be improved in terms of food recognition, portion analysis, and food recommendation. Regarding food recognition, the classes of food recognition can be increased to cover different types of food. Besides, the model can be improved to recognize all food portions, even if they overlap. Additionally, the food recommendation can be more specific by considering the nutrition needs of the user. Another improvement can be applied to the food searching function that allows rea-ltime searching in the sense that each time users enter a character. The system will search through the database, and the obtained results will match the sequence of characters inserted.

REFERENCES

1. Aubrey A. Bad Diets Are Responsible For More Deaths Than Smoking, Global Study Finds. [Online].; 2019 [cited 2021 November 7. Available from: https://www.npr.org/sections/thesalt/2019/04/03/709507504/bad-diets-are-responsible-for-more-deaths-than-smoking-global-study-finds.
2. Gravagna N. Six Fundamental Human Needs We Need To Meet To Live Our Best Lives, Forbes. [Online].; 2018 [cited 2021 November 7. Available from: https://www.forbes.com/sites/quora/2018/02/05/six-fundamental-human-needs-we-need-to-meet-to-live-our-best-lives/?sh=358a3b22344a.
3. Hawkes C. Uneven dietary development: linking the policies and processes of globalization with the nutrition transition, obesity and diet-related chronic diseases. Global Health. 2006;2:4. doi: 10.1186/1744-8603-2-4.
4. Yap R. Quarter Quarter Half: A Guide to Healthy, Balanced Meals. [Online].; 2020 [cited 2021 December 17. Available from: https://mypositiveparenting.org/2020/06/30/quarter-quarter-half-a-guide-to-healthy-balanced-meals/.
5. HealthifyMe. [Online]. [cited 2021 November 3. Available from: https://www.healthifyme.com/my/.
6. Calorie Mama Instant Food Recognition. [Online]. [cited 2021 November 3. Available from: https://caloriemama.ai/.
7. Firebase Realtime Database. [Online]. [cited 2022 July 1. Available from: https://firebase.google.com/docs/database.
8. Weiqing M., Linhu L., and Zhengdong L. Ingredient-Guided Cascaded Multi-Attention Network for Food Recognition. In the 27th ACM International Conference; October 2019.
9. Jianpeng H., Zeman S., Janine W., Deborah A.K., Carol B., and Fengqing Z. Multi-Task Image-Based Dietary Assessment for Food Recognition and Portion Size Estimation. In 2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR); 2020.
10. Ray. Medium. [Online]. [cited 2022 April 12. Available from: https://raychunyin00.medium.com/image-segmentation-extending-faster-r-cnn-into-mask-r-cnn-953fe4923d64.
11. Soroka S. Quora. [Online]. [cited 2022 April 5. Available from: https://www.quora.com/Recommender-Systems-1/How-does-content-based-filtering-recommendation-algorithm-work.
12. Rana K.S. Food Recommendation System Based On Content Filtering Algorithm, Project Report for Bachelor's Degree in Computer Science and Information Technology, DWIT College; 2016.
13. Choy MC. Berrys [Online]. [cited 2023 October 7. Available from: https://www.berrys.com.my/products-single-buns.html
14. Seasongroup. [Online]. [cited 2023 October 7. Available from: https://www.seasongroupshop.com.my/product/polo-bun/
15. Cookpad. [Online]. [cited 2023 October 7. Available from: https://cookpad.com/tw/%E9%A3%9F%E8%AD%9C/9463737-%E6%B8%85%E7%82%92a%E8%8F%9C
16. Kaggle. Food-11 image dataset. [Online]. [cited 2023 October 7. Available from: https://www.kaggle.com/datasets/trolukovich/food11-image-dataset