Mobile Food Journalling Application with Convolutional Neural Network and Transfer Learning: A Case for Diabetes Management in Malaysia

Jason Thomas Chew[1], Patrick Hang Hui Then[4]
School of Information and
Communication Technologies
Swinburne University of Technology
Sarawak Campus, Malaysia[1,4]

Yakub Sebastian[2]
College of Engineering, IT & Environment
Charles Darwin University, Australia

Valliapan Raman[3]
Department of AI and DS
Coimbatore Institute of Technology
Coimbatore, India

***Abstract*—Diabetes is an ever worsening problem in modern society, placing a heavy burden on healthcare systems. Due to the association between obesity and diabetes, food journaling mobile applications are an effective approach for managing and improving the outcome of diabetics. Due to the efficacy of nutritional tracking and management in managing diabetes, we implemented a deep learning-based Convolutional Neural Network food classification model to aid with food logging. The model is trained on a subset of the Food-101 and Malaysian Food 11 datasets, including web-scraped images, with a focus on food items found locally in Malaysia. In our experiments, we explore how fine-tuning of the image dataset improves the performance of the model.**

***Keywords*—Convolutional neural network; deep learning; diabetes; food journal; mobile application; nutritional tracking; Malaysia**

I. Introduction

Diabetes is becoming an increasingly prevalent disease in modern society. It was estimated that in 2017 at least 6% of world population were affected by type 2 diabetes (T2DM) [1]. Urbanization has caused a shift in dietary patterns among the populace towards food with a higher associated risk of diabetes [2]. Diet affects an individual's diabetes outcome in various ways, both directly and indirectly. The sugar content of a meal directly influences an individual's blood glucose levels, whereas an indirect factor such as the consumption of high-fat foods causes obesity, which then contributes to the onset of diabetes [3]. It is well-established that there is a strong link between obesity and diabetes, where excess weight plays a part in up to 90% of the cases of T2DM [4]. Consequently, diet management plays a crucial role in managing the disease outcome among the diabetics and in disease prevention among the non-diabetics.

In Malaysia, the incidence rate of diabetes in adults has increased from 15.2% in 2011 [5] to 17.5% in 2015 [6]. A long-standing health problem for the country, diabetes in Malaysia lays an increasingly heavy burden on the country's healthcare system, costing it approximately RM2.04 billion in 2011 alone [7]. Due to the association between diet and diabetes [8], food journaling applications can improve T2DM disease outcomes by helping users to plan and monitor their diet. The use of food journaling software and other related nutritional tracking applications is an emerging approach that may improve an individual TD2M management outcome [9], [10], [11]. For instance, *Diabetes Notepad* was developed to assist diabetics in managing the disease via self-care through diet monitoring, leading to an improved clinical outcome of diabetes in Korea [12]. In Malaysia, despite a consensus that having mobile food journaling app with image-snapping feature would be highly desirable to help with T2DM management, more than half of the people surveyed were unaware of such dietary logging applications [13].

In this paper, we present a new food journaling application tailored to the Malaysian demography. The challenges in developing the application are non-trivial. Because the onset T2DM is strongly linked to environmental, sociodemographic and cultural factors [14], an effective food journaling application should be contextualised to its target population and capable of recognising nutritional components of local diets (e.g. *Diabetes Notepad* is targeted at South Korean demography). Many food databases still contain inconsistent or missing nutritional content of certain food makes it difficult to estimate one's nutritional intake [9]. For instance, they may not sufficiently document non-standard food types such as restaurant food, ethnic food, food prepared by friends and party food. It is therefore important to develop a solution which focuses on a specific demography such as Malaysia.

Our contributions are two-fold. First, we propose a new mobile food journalling application that uses a convolutional neural network (CNN) architecture to enhance the classification of food images taken with mobile phones. The amount of calorie intake from the classified food is automatically calculated for the users to help with personal diet planning and monitoring, which is crucial in managing such chronic diseases as T2DM. Second, we propose a way to enhance the performance of the classification model via transfer learning, a strategy that appears to work well on non-conventional images of local Malaysian food.

The remaining of this paper is organized as follows. Section II reviews some related work, with a specific appreciation of food journalling app development in Malaysian context. Section III describes the proposed system architecture, software components, and the method for training CNN classifier. In Section IV, we explain our experiments and report the results. Section V summarises and discusses the experimental results. The limitations of the system are described in Section VI. Section VII concludes this paper.

II. Related Work

A. Food Recognition and Classification

Food recognition and classification algorithms form the internal machinery of food journaling apps, executed as part of the food entry workflow that reduce the burden of food entry by users. Convolutional Neural Networks (CNNs) are a popular method for food classification due to their increased efficacy over traditional machine learning-based approaches [15].The success of CNNs can be attributed to transfer learning, where a model that is pre-trained on a large-scale image dataset such as ImageNet is retrained on a different dataset [16], [17].

There are several existing works in the literature that apply transfer learning to train CNN-based food classification models [15], [18], [19], [20], [21]. Meyers et al. [19] developed *Im2Calories*, a food diary application that uses an ensemble of CNN classifiers for food detection, classification, segmentation, and volume estimation. Their system is able to detect food from 23 different restaurants as well as perform general food detection. Pan et al. [20] proposed *DeepFood* which leverages CNN-based transfer learning algorithms for deep feature extraction. The author's goal is to classify fresh food ingredient images indexed in the Mealcome image dataset [22]. In the most recent example, Sahoo et al. [21] developed *FoodAI*, a CNN model trained to classify various dishes with a focus on food found locally in Singapore. The model is able to predict a wide variety of Singaporean and South-East Asian dishes, recognizing 756 distinct food classes.

B. Food Journaling Apps Development in Malaysia

There is limited documentation of food journaling apps that specifically target T2DM management for Malaysian population. A recent survey of mobile diabetes management apps showed that existing apps focused only on reporting and setting reminders for meal time, without automatic food recognition ability [23]. Where automatic food image recognition is available, it is not in the context of TD2M management. For instance, [24] used artificial neural network to recognise five different types of Malaysian traditional dessert: *curry puff*, *'kuih ketayap'*, *'kuih kosui'*, *red tortoise cake*, and *'putu piring'*. Calories were subsequently computed from the detected food images but no implication of T2DM management was documented. The algorithm reported 80% food recognition accuracy. [13] proposed a food journalling app that uses deep neural network model trained with InceptionV3 and MobileNetV2 on Food-101 dataset. Although the app was designed to assist T2DM management, it is does not provide the capability to automatically compute calorie amount based on classified images.

III. Method

In this section, we describe the internal workings of our proposed food journalling app. Fig. 1 overviews the essential workflow of the app. The user snaps a picture of food item, which is sent to a web server running the food image classification algorithm. The classification result is subsequently returned and displayed to the user, who then confirms the food type and provides further information about the amount of food that was consumed. The app relays these information back to the server, which computes and returns the estimated consumed calories. Fig. 2 shows the technology components used for the development of the app.

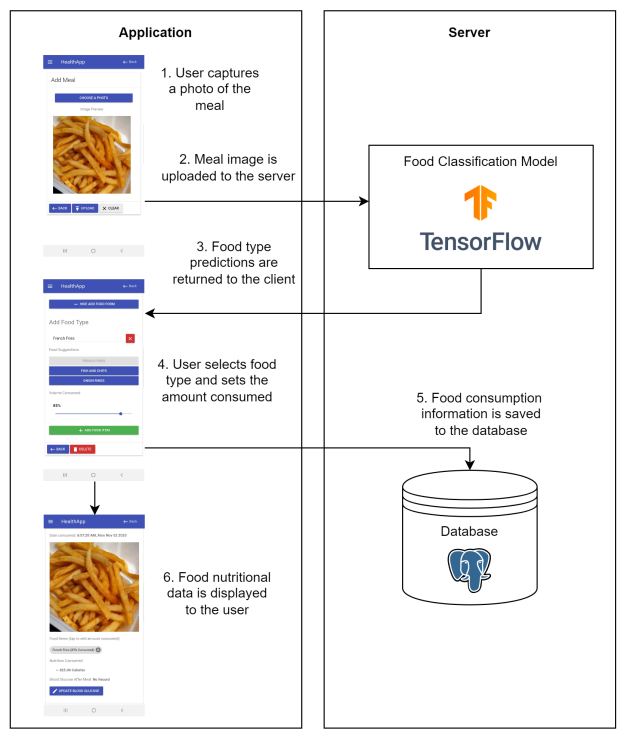

Fig. 1. Workflow of the Proposed Food Journalling App.

A. Food Dataset

The food image classifier forms the backbone of the app. To construct an initial dataset for training the model, we combined images from the Food-101 [25] and Malaysian Food 11[1] datasets. Several food categories regularly consumed by Malaysians have previously been identified by [26]. These include cereal products, beverages, fruits and vegetables, confectioneries, meat products, fish products, milk products,

[1] https://www.kaggle.com/karkengchan/malaysia-food-11

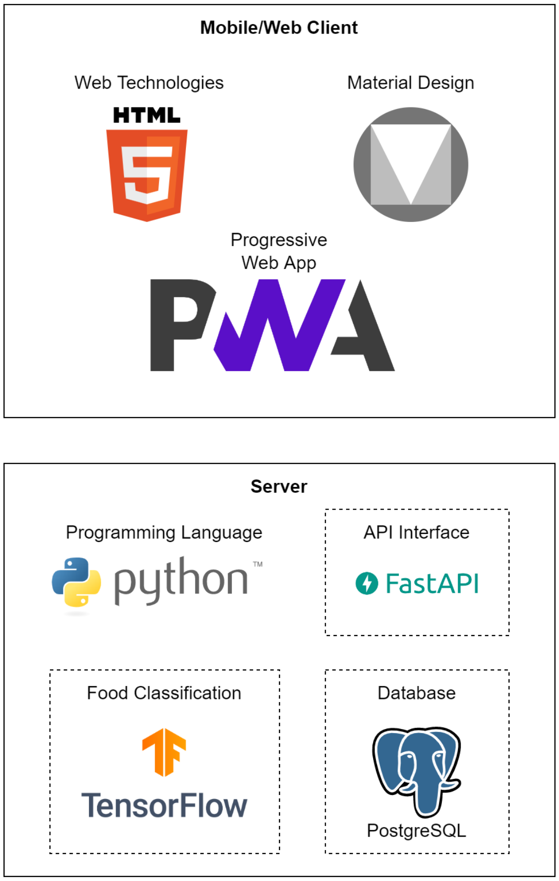

Fig. 2. Technology Components Used by the Proposed App.

condiments, eggs, legumes, and spreads. We use these categories to select food classes from the two datasets for training the model. An additional local noodle dish from Sarawak, *kolo mee*, is included in the dataset. The *kolo mee* food images were scraped from the web.

In cases where both the Food-101 and Malaysian Food 11 datasets have overlapping food classes (*fish and chips, fried rice* and *hamburger*), the images from the Malaysian food dataset are used for training the model and the Food-101 images are discarded. This is to reduce the class imbalance in the dataset while also ensuring that the training images are as visually similar as possible to the way the local dishes are prepared. The *apple pie* and *kolo mee* classes were manually processed for reasons explained later. One image was removed from the *Nasi Lemak* class due to corruption. The remaining food classes were used without any further preprocessing. The final dataset consists of 31335 images across 32 classes. Details of the food dataset for training the model are shown in Table I.

B. Model Training

We implemented the CNN using the *Keras* deep learning framework with *TensorFlow 2.2* backend and trained it on the dataset. To shorten the training duration and improve the model's performance, transfer learning is used for training the model. Transfer learning allows for the features learnt in one domain to be transferred into another domain, such that the resulting performance on the new domain is improved, as opposed to training a new model from scratch [27], [28].

TABLE I. FOOD CLASSES SELECTED FOR TRAINING THE MODEL

Food Class	Images	Food Categories
Apple Pie	883	Confectionaries
Caesar Salad	1000	Fruits & Vegetables
Chocolate Cake	1000	Confectionaries
Donuts	1000	Confectionaries
Dumplings	1000	Meat, Fruits & Vegetables
Fish and Chips	1000	Fish
French Fries	1000	Fruits & Vegetables
Fried Noodles	1000	Cereal
Fried Rice	1000	Cereal
Garlic Bread	1000	Cereal
Hamburger	1000	Meat
Hot Dog	1000	Meat
Ice Cream	1000	Confectionaries
Kaya Toast	1000	Cereal, Condiments
Kolo Mee	473	Cereal, Meat
Laksa	1000	Cereal
Lasagna	1000	Cereal
Mixed Rice	1000	Cereal, Meat, Fruits & Vegetables
Nasi Lemak	999	Cereal
Onion Rings	1000	Fruits & Vegetables
Pancakes	1000	Cereal
Peking Duck	1000	Meat
Pizza	1000	Cereal, Meat
Popiah	1000	Meat, Fruits & Vegetables
Ramen	1000	Cereal
Roti Canai	1000	Cereal
Satay	1000	Meat
Spaghetti Bolognese	1000	Cereal, Meat
Spaghetti Carbonara	1000	Cereal, Meat
Steak	1000	Meat
Tiramisu	1000	Confectionaries
Waffles	1000	Confectionaries

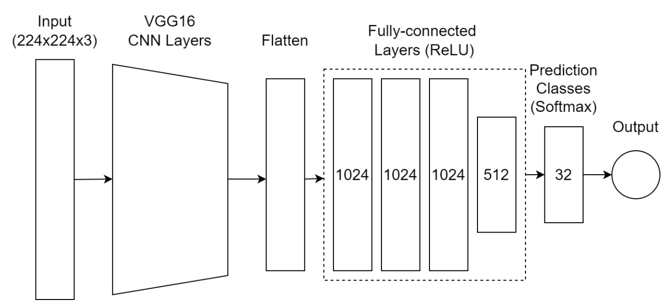

Fig. 3. CNN and Transfer Learning Architecture

Fig. 3 illustrates our CNN and transfer learning training architecture. The base model used for transfer learning is VGG16 [29], initialized with weights pre-trained on the ImageNet dataset [17]. The weights of the convolutional layers were frozen so that they do not update, while the fully connected layer of the model was replaced with three fully-connected layers with widths of 1024, 1024, 1024, and 512, respectively, and the output layer was also updated to 32 classes.

The images in the dataset are split into three: 60% for training, 20% for validation and 20% for testing. In order to prevent the model from overfitting, the images in the training set were augmented. Image augmentation allows the model to generalize better by applying a random transformation on the images according to certain parameters as they are being fed into the model during training [30]. The parameters used for augmentation of the test images that produced the best performing model are the following: image rotation between -90 and 90 degrees, random horizontal flipping of the images, varied brightness ranging between 75% to 100%, and image size rescaling between 80% to 100%.

The model was trained on a local machine equipped with an Nvidia GTX 1060 6GB GPU. Training was performed in three steps with different learning rates and a fixed batch size of 50. The initial learning rate was set to 1×10^{-4} and the model was trained until the accuracy stopped increasing. When the accuracy stopped increasing, the learning rate was updated and the training step was resumed. The learning rate for the second and third training steps were 1×10^{-5} and 1×10^{-6} respectively. A complete model takes an average of 6 hours to train. The best performing model achieved a top-1 accuracy of 76.77% and a top-5 accuracy of 94.37%.

IV. Experiment and Results

A. Initial Results

During the initial training of the model, the original apple pie images from the Food-101 dataset (1000 images) were used without any manual preprocessing. However, during testing of the model, it was found to have performed poorly in predicting apple pie images, with an accuracy of 53.5%.

Upon inspection of the apple pie images, we hypothesized that the model's below average performance was due to a portion the images in apple pie class that poorly represented the features of the food class. In order to confirm this hypothesis, we retrained the model after manually preprocessing the images to remove images that were suspected to be causing the model's poor performance. Three categories of images were identified and removed from the apple pie image class: non-food images, incorrectly labelled images, and images that poorly represent the food class, which includes images where the apple pie is mostly blocked from view or are mixed with different food classes. Fig. 4 shows a sample of the removed images. A total of 117 images were removed from the dataset and the model was retrained. The images in the other 31 classes of the dataset remain unchanged.

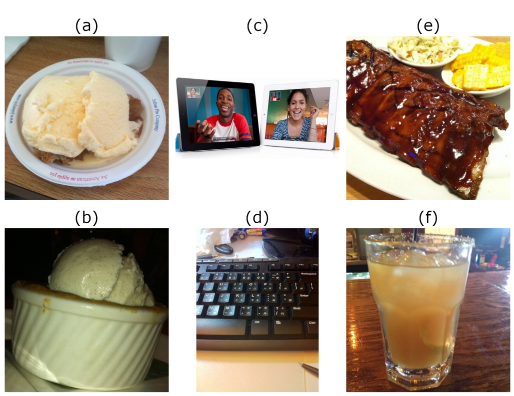

Fig. 4. The Images Removed During Preprocessing of the Apple Pie Class Includes Severely Occluded Images (a, b), Non-Food Images (c, d), and Other Food Images (e, f).

TABLE II. Images Misclassified by the Model when Trained with the Unprocessed and Preprocessed Datasets. The Food Classes with Fewer Misclassifications are Highlighted.

Class	Unprocessed	Preprocessed
All Food Classes	93	72
Caesar Salad	1	1
Chocolate Cake	4	3
Donuts	5	3
Dumplings	3	2
Fish and Chips	2	2
French Fries	0	0
Fried Noodles	0	0
Fried Rice	1	3
Garlic Bread	6	3
Hamburger	1	1
Hot Dog	5	6
Ice Cream	7	4
Kaya Toast	8	4
Kolo Mee	1	0
Laksa	0	0
Lasagna	2	5
Mixed Rice	0	2
Nasi Lemak	3	0
Onion Rings	2	3
Pancakes	12	4
Peking Duck	6	3
Pizza	3	1
Popiah	0	5
Ramen	0	0
Roti Canai	3	7
Satay	2	0
Spaghetti Bolognese	0	0
Spaghetti Carbonara	1	0
Steak	0	0
Tiramisu	10	8
Waffles	5	2

After retraining using the preprocessed apple pie class, the model's performance on apple pie images noticeably improved from 53.5% to 59.32%. Table II shows the incorrect classifications of the two models, where *Unprocessed* denotes the model trained on the original 1000 *apple pie* images and *Preprocessed* denotes the model trained on the preprocessed apple pie images. There is an overall reduction in misclassifications observed in the retrained model. In the table, the model which made fewer misclassifications on a specific food class is highlighted. Several of the classes with the largest improvements observed are those which share visual similarities (*pancakes* and *kaya toast*), and *ice cream*, which is commonly served as a topping or alongside an *apple pie*.

Additionally, the retrained model was found to have misclassified fewer non-*apple pie* images as an *apple pie* despite the images in the other food classes remaining unchanged. The total number of images from the other food images which were incorrectly classified as *apple pie* was reduced from 93 to 72. These results suggest that the model's accuracy can be further improved by manually processing the images in the remaining food classes to ensure that each image in the dataset significantly represents the features of their respective classes.

B. Effects from Incorporating Additional Local Food Classes

Seven additional food classes were included to the dataset in order to increase the model's coverage of local food classes, namely: *Ais Kacang*, *Ayam Pansuh*, *Beef Noodle Soup*, *Boiled Eggs*, *Fried Eggs*, *Kampua Noodles*, and *Layer Cake*. The new images were obtained by scraping the web and manually processing the images to remove unrelated images. We trained two models on the new dataset, one with the same parameters

TABLE III. Prediction Accuracy for the Models Trained on the Updated Dataset with and without Focal Loss (FL). The Columns with Higher Accuracy are Highlighted

Food Class	Images	Accuracy (No FL)	Accuracy (FL)
Ais Kacang	435	80.46%	72.41%
Ayam Pansuh	120	58.33%	66.67%
Beef Noodle Soup	520	73.08%	76.92%
Boiled Eggs	623	87.20%	85.60%
Fried Eggs	398	81.25%	77.50%
Kampua Noodles	292	63.79%	63.79%
Layer Cake	669	91.04%	86.57%
Model Accuracy	34412	75.51%	74.17%

as the original model, and another using focal loss optimizer [31]. The focal loss optimizer is introduced in an attempt to account for the high class imbalance in the new food classes. Focal loss puts greater focus on the classes that the model has difficulty classifying during the training process. The images contained in each class and the results of the two models are shown in Table III.

Upon evaluation of the two new models, the model trained using the same parameters as described in Section III-B achieved an accuracy of 75.51% whereas the model using the focal loss optimizer achieved an accuracy of 74.17%. The lower accuracy of the two new models can be attributed to the class imbalance of the newer classes due to a lack of available images on the web. The results of both models are shown in Table III. The model with focal loss is observed to perform better in classifying *Ayam Pansuh*, which has the greatest class imbalance of all the food classes. Although focal loss is observed to allow for higher performance on the highly imbalanced class, the model performs poorer in classifying the other food classes in the dataset.

C. Sample Use Case

In this section, we describe the meal recording process on the web application. In order for users to use the application, they must first register an account and log in.

Fig. 5(a) and 5(b) shows the account creation and log in interfaces. After the user has logged in to the application, they can begin tracking their meals and view their meal history via the *Smart Diet Watcher* module. The module is accessible from the application's main menu as shown in Fig. 5(c).

The first step in the meal logging process is uploading an image of the meal. Fig. 5(d) shows the meal creation interface for uploading a meal image. The user can choose between taking a photo of the meal or choosing an image from the device. The meal image that the user has selected is previewed before upload.

After uploading the meal image, the user can add food items to the meal. Fig. 6(a) shows the interface for adding food items to the meal. The food item suggestions are provided by the model's predictions based on the food image. Users can add multiple food items to the meal in cases where the food image contains multiple dishes. Once food items are added to the meal, users can edit or delete them as shown in Fig. 6(b). The user can optionally record their blood glucose levels after the meal.

Once the user has completed logging their meal, they can view a summary of the recorded meal as shown in Fig. 6(c).

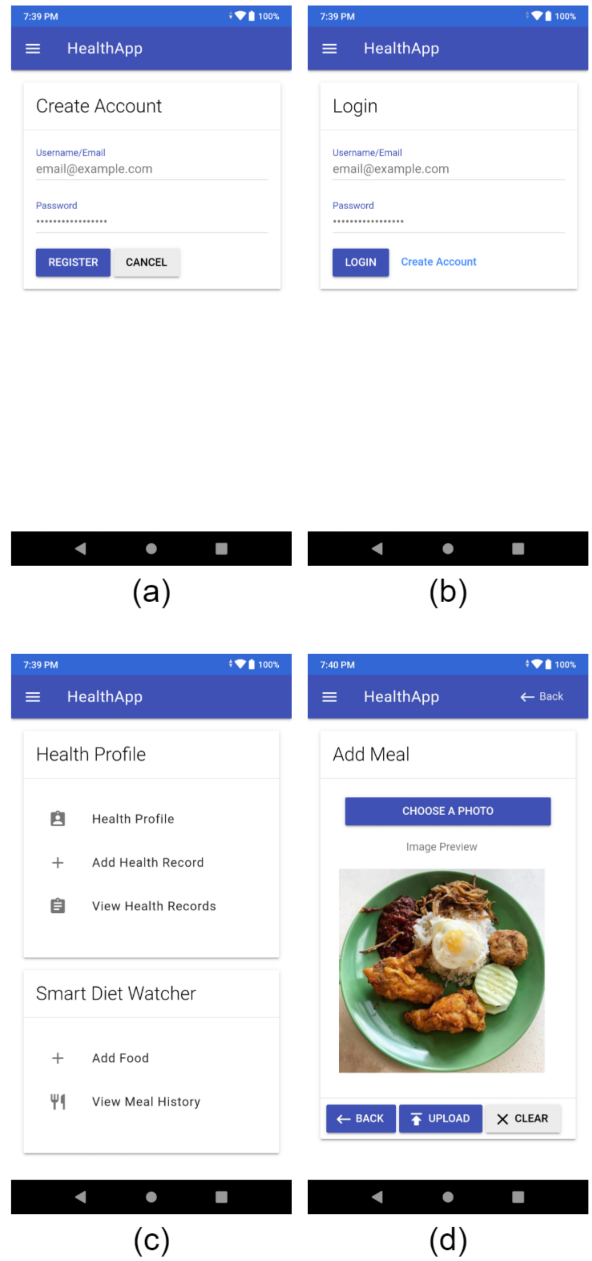

Fig. 5. (a) Shows the Account Creation, (b) Shows the Login Interface, (c) Shows the Main Menu Interface, and (d) Shows the Meal Creation Interface.

The meal summary shows the food items contained in the meal image, the meal's nutritional value and the user's blood glucose level after the meal. The nutritional values shown in the meal summary is calculated based on the percentage of each consumed food item that was set by the user. The baseline nutritional values that are used for the nutrition calculations are based on the amount of a single full portion serving. Lastly, the user can view their past meal records from the meal history menu as shown in Fig. 6(d).

V. Discussion

To summarise, we trained a CNN model to predict food items found locally in Sarawak, Malaysia. Two publicly avail-

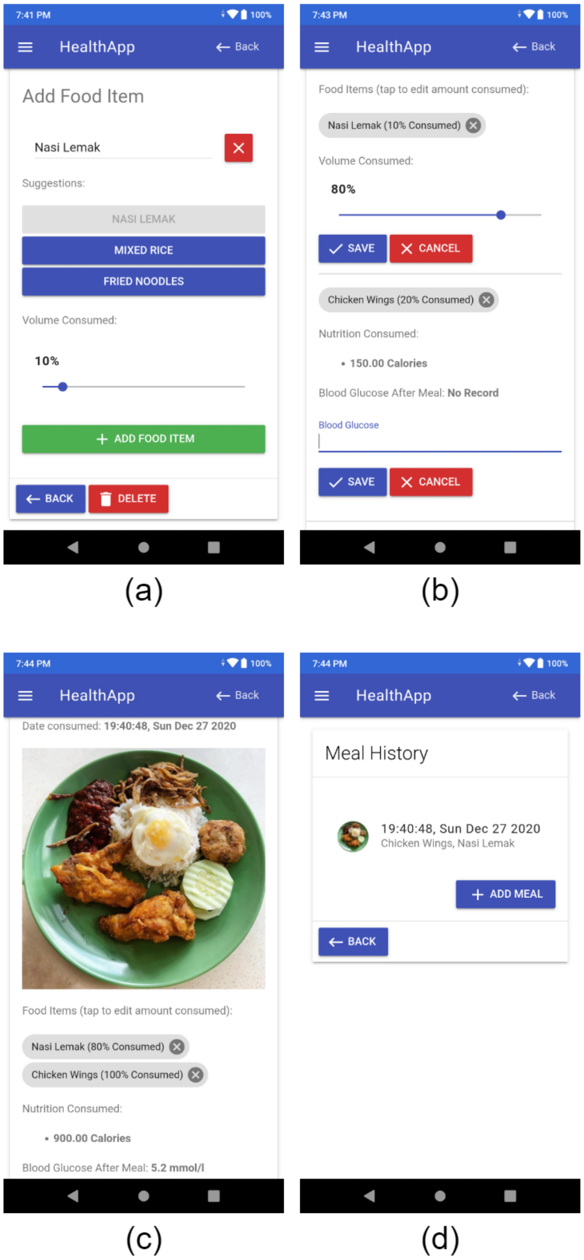

Fig. 6. (a) Shows the Interface for Adding Food Items, (b) Shows the Interface for Updating Food Items and the Post-Meal Blood Glucose Record, (c) Shows a Summary of the Meal, and (d) Shows the Meal History Menu.

able food datasets, Food-101 and Malaysian Food 11 datasets, were used for the initial model training. This was followed by incorporating local food items that were scraped from the web.

Upon training the initial model, we found that the *apple pie* food class was poorly performing. Thus, we investigated the effect of further preprocessing the images to improve the model's performance. We removed poor quality and unrelated images from the dataset, retrained the model and reported the effects of the preprocessing step in detail, shown in Table II. There was an improvement in accuracy despite the smaller number of training images, which is likely due to the images in the preprocessed dataset better representing the apple pie class.

Next, we incorporated images of local Sarawakian food items scraped from the web into the dataset and retrained the model. Due to concerns of class imbalances for the web scraped images, focal loss was introduced. We explored the effect of focal loss on the model's performance, as shown in Table III. Interestingly, we found that the model without focal loss performed better overall, achieving a higher accuracy on most of the web scraped food classes despite the class imbalance. One notable case however is the *Ayam Pansuh* class which is severely imbalanced. The results obtained indicates that while focal loss is generally expected to improve performance when there is a class imbalance, it may be beneficial to train another model that does not implement focal loss as well in case the use of focal loss unexpectedly results in a poorer performing model.

VI. Limitations

Currently, the number of food classes that the model can predict is limited. We plan to expand the total number of food classes that the model can predict by collecting local food images and using them to retrain the model. Our final goal is for the model to recognize most local food consumed in Malaysia.

The model's performance has been shown to improve through manual preprocessing of the apple pie class, indicating that there is a possible performance gain if all the images were manually processed. In particular, *FoodAI* [21] managed to achieve a top-1 accuracy of 83.2% where the authors put significant effort into manual inspection of the food images by domain experts while constructing the dataset. However, in our case, time and resource constraints limit the manual preprocessing of the dataset. This limitation can be addressed in future work by looking into methods of automating the dataset preprocessing via CNN-based feature extraction and clustering [32] to group visually similar images for batch labeling.

VII. Conclusion

In this paper we implement a food classification model with a CNN-based architecture. Transfer learning was applied to boost the model's performance and reduce training time. We explored the effect of manually preprocessing of the dataset on the model's performance. Results indicate that the model's performance can be improved further by carefully curating the images that are used for training the model. In future work, we plan to expand the food classes that the model can predict to classify more local Malaysian dishes.

Acknowledgment

The authors extend appreciation to collaborators in the Clinical Research Centre, Sarawak General Hospital for validating our works and testing our applications. The project is funded under Prototype Research Grant Scheme from the Malaysia Ministry of Higher Education, Ref: PRGS/1/2019/ICT02/SWIN/01/1.

REFERENCES

[1] M. A. B. Khan, M. J. Hashim, J. K. King, R. D. Govender, H. Mustafa, and J. Al Kaabi, "Epidemiology of type 2 diabetes–global burden of disease and forecasted trends," *Journal of epidemiology and global health*, vol. 10, no. 1, p. 107, 2020.

[2] V. Mohan, V. Ruchi, R. Gayathri, M. R. Bai, V. Sudha, R. M. Anjana, and R. Pradeepa, "Slowing the diabetes epidemic in the world health organization south-east asia region: the role of diet and physical activity," *WHO South-East Asia Journal of Public Health*, vol. 5, no. 1, pp. 5–16, 2016.

[3] R. J. Johnson, L. G. Sánchez-Lozada, P. Andrews, and M. A. Lanaspa, "Perspective: a historical and scientific perspective of sugar and its relation with obesity and diabetes," *Advances in Nutrition*, vol. 8, no. 3, pp. 412–422, 2017.

[4] S. Verma and M. E. Hussain, "Obesity and diabetes: an update," *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, vol. 11, no. 1, pp. 73–79, 2017.

[5] Institute for Public Health Malaysia, "National health and morbidity survey 2011 (nhms 2011)," *Vol. II: Non-Communicable Diseases, Risk Factors & Other Health Problems*, 2015.

[6] Ministry of Health Malaysia, "National health & morbidity survey 2015 non-communicable diseases, risk factors & other health problems," 2015.

[7] F. I. Mustapha, S. Azmi, M. R. A. Manaf, Z. Hussein, J. Mahir, F. Ismail, A. N. Aizuddin, and A. Goh, "What are the direct medical costs of managing type 2 diabetes mellitus in malaysia," *Med J Malaysia*, vol. 72, no. 5, pp. 271–277, 2017.

[8] W. Sami, T. Ansari, N. S. Butt, and M. R. Ab Hamid, "Effect of diet on type 2 diabetes mellitus: A review," *International journal of health sciences*, vol. 11, no. 2, p. 65, 2017.

[9] F. Cordeiro, D. A. Epstein, E. Thomaz, E. Bales, A. K. Jagannathan, G. D. Abowd, and J. Fogarty, "Barriers and negative nudges: Exploring challenges in food journaling," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015, pp. 1159–1162.

[10] A. Darby, M. W. Strum, E. Holmes, and J. Gatwood, "A review of nutritional tracking mobile applications for diabetes patient use," *Diabetes technology & therapeutics*, vol. 18, no. 3, pp. 200–212, 2016.

[11] J. Jung, L. Wellard-Cole, C. Cai, I. Koprinska, K. Yacef, M. Allman-Farinelli, and J. Kay, "Foundations for systematic evaluation and benchmarking of a mobile food logger in a large-scale nutrition study," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 2, pp. 1–25, 2020.

[12] Y. J. Kim, S. Y. Rhee, J. K. Byun, S. Y. Park, S. M. Hong, S. O. Chin, S. Chon, S. Oh, J.-t. Woo, S. W. Kim *et al.*, "A smartphone application significantly improved diabetes self-care activities with high user satisfaction," *Diabetes & metabolism journal*, vol. 39, no. 3, pp. 207–217, 2015.

[13] K. K. Chan and N. Suki, "Mobile food management and dietary management for T2DM patients in malaysia," *International Journal of Scientific & Technology Research*, vol. 8, no. 11, pp. 574–578, 2019.

[14] T. Dendup, X. Feng, S. Clingan, and T. Astell-Burt, "Environmental risk factors for developing type 2 diabetes mellitus: a systematic review," *International journal of environmental research and public health*, vol. 15, no. 1, p. 78, 2018.

[15] V. Bruno and C. J. Silva Resende, "A survey on automated food monitoring and dietary management systems," *Journal of health & medical informatics*, vol. 8, no. 3, 2017.

[16] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[18] G. Ciocca, P. Napoletano, and R. Schettini, "Cnn-based features for retrieval and classification of food images," *Computer Vision and Image Understanding*, vol. 176, pp. 70–77, 2018.

[19] A. Meyers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. P. Murphy, "Im2calories: towards an automated mobile vision food diary," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1233–1241.

[20] L. Pan, S. Pouyanfar, H. Chen, J. Qin, and S.-C. Chen, "Deepfood: Automatic multi-class classification of food ingredients using deep learning," in *2017 IEEE 3rd international conference on collaboration and internet computing (CIC)*. IEEE, 2017, pp. 181–189.

[21] D. Sahoo, W. Hao, S. Ke, W. Xiongwei, H. Le, P. Achananuparp, E.-P. Lim, and S. C. Hoi, "Foodai: Food image recognition via deep learning for smart food logging," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2260–2268.

[22] H. Chen, J. Xu, G. Xiao, Q. Wu, and S. Zhang, "Fast auto-clean cnn model for online prediction of food materials," *Journal of Parallel and Distributed Computing*, vol. 117, pp. 218–227, 2018.

[23] S. Izahar, Q. Y. Lean, M. A. Hameed, M. K. Murugiah, R. P. Patel, Y. M. Al-Worafi, T. W. Wong, and L. C. Ming, "Content analysis of mobile health applications on diabetes mellitus," *Frontiers in Endocrinology*, vol. 8, p. 318, 2017.

[24] N. A. A. N. Muhammad, C. P. Lee, K. M. Lim, and S. F. A. Razak, "Malaysian food recognition and calorie counter application," in *2017 IEEE 15th Student Conference on Research and Development (SCOReD)*. IEEE, 2017, pp. 445–450.

[25] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 – mining discriminative components with random forests," in *European Conference on Computer Vision*, 2014.

[26] I. Pondor, W. Y. Gan, and G. Appannah, "Higher dietary cost is associated with higher diet quality: a cross-sectional study among selected malaysian adults," *Nutrients*, vol. 9, no. 9, p. 1028, 2017.

[27] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: A survey," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 5, pp. 1019–1034, 2014.

[28] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*. Springer, 2018, pp. 270–279.

[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[30] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[32] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5147–5156.