HEALTHCARE APPLICATION FOR LOW-FODMAP VEGETABLES AND FRUITS IMAGE IDENTIFICATION USING DEEP LEARNING

Watcharachai Wiriyasuttiwong[1*], Kamon Chaiyasit[2], Maethaphan Kitporntheranunt[3] Varit Srisukthaveerat[4], Natthapol Tiawattananon[1], Wittayatorn Tokaew[5] Supaporn Sridee[5], and Vibhakorn Permpoon[6]

[1]Faculty of Engineering, Srinakharinwirot University, Nakhon Nayok, 26120, Thailand.
[2]Vitallife Scientific Center, Bumrungrad International Hospital, Bangkok, 10110,Thailand.
[3]Faculty of Medicine, Srinakharinwirot University, Nakhon Nayok, 26120, Thailand.
[4]The Senizens Wellness Center , Bangkok,10160,Thailand.
[5]School of Communication Arts, Sukhothai Thammathirat Open University, Nonthaburi, 11120,Thailand.
[6]Division of Gastroenterology and Hepatology, Bumrungrad International Hospital, Bangkok, 10110,Thailand.

ABSTRACT

This paper presents the design and development of a healthcare application using deep learning in low-FODMAP vegetables and fruits image identification for patients with gastrointestinal problems or irritable bowel syndrome. FODMAP is fermentable and other specific type of carbohydrate can cause gas in GI tract. This research uses MobileNetV2 as a deep learning model, a convolutional neural network architecture. In training the model, the dataset of 10 types of low-FODMAP and high-FODMAP vegetables and fruits were used. The results of testing the application functionality by experimenting with the testing dataset of 2,000 images. It was found that the predictive efficiency of the system was average precision of 96.55%, average recall and balance accuracy of 96.30% and average F1-score of 96.29%. There is a probability that it can be classified to select only the classes of interest. This is due to the presence of a large number of true positives (TP) and a small number of false positives (FP), and a good average F1-score or harmonic mean between precision and recall.

Keywords: Low-FODMAP, Healthcare Application, Machine Learning, Irritable Bowel Syndrome

Manuscript received on 11 May, 2023; revised on 20 June, 2023, accepted on 30 June, 2023.
*Corresponding author E mail: watch@g.swu.ac.th, Department of Computer Engineering, Faculty of Engineering, Srinakharinwirot University, Thailand.

1. INTRODUCTION

Today, we find that patients with digestive problems are increasing. The most common disease is irritable bowel syndrome (IBS). According to statistics, 10-15% of patients with IBS worldwide have intestinal disorders which cause chronic constipation and diarrhea alternately followed. Each patient had a different degree of symptoms: 40% mild, 35% moderate, and 25% severe. [1] The symptoms of the disease meet the diagnostic criteria of Rome III [2] and American College of Gastroenterology. [3] IBS can be divided into four types: diarrhea predominant IBS (IBS-D), constipation predominant IBS (IBS-C), mixed IBS (IBS-M) and undetermined IBS (IBS-U). In addition, Small Intestine Bacterial Overgrowth (SIBO) is one of the causes of constipation, flatulence, diarrhea, included as cramps, bloating, and excessive gastrointestinal wind. [4] This is caused by an overgrowth of bacteria in the upper gastrointestinal tract. The bacteria produce gas that interferes with intestinal motility. The diagnosis of SIBO is evaluated through a breath test. That is, drinking a glucose or lactulose solution and analyzing the results. [5-6] We also found that both IBS and SIBO in the upper gastrointestinal tract were observed. In some patients there was a correlation. That is, excessive microbial overgrowth in the upper gastrointestinal tract causes irritable bowel syndrome. The patient therefore suffers from the combination of both symptoms. [7] It also found that more than a third of IBS patients had symptoms of SIBO. [8] In addition to taking care of the medication as ordered by the physician. These is also a need for a diet together. Because certain foods can result in exacerbation of symptoms. That is, the small intestines are poorly absorbed by carbohydrate foods, therefore these foods are fermented by intestinal bacteria, especially in the small intestine until causing gas and weak acid in the intestines to increase and lead to sudden changes in the intestines environment. As a result, patients have symptoms of flatulence, stomachache or diarrhea also known as “IBS”. The concept of food was introduced as some sugar when not digested in ileum, it is transferred to the large intestine and fermentation occurs to become short-chain fatty acids and gasses such as H_2, CH_4 or CO_2. Another part of sugar when not digested, it will draw water outside the small intestine cells into the digestive tract which causes the water balance to be lost. These sugars are a group of carbohydrates that the body is not well absorbed at all, known as fermentable oligosaccharides, disaccharides, monosaccharides and polyols (FODMAPs).

FODMAPs were defined and developed by the researchers from Monash University in Australia, which they aim to study the relationship with diseases associated with Crohn's disease. [9] FODMAPs are abbreviations for different types of sugars, which is Fermentable, Oligosaccharides (= Fructans and Galacto-oligosaccharides), Disaccharides (= Lactose and Maltose), Monosaccharides (= Fructose), And Polyols (= Sugar alcohols such as Sorbitol and Mannitol) where FODMAPs are divided into two groups: High-FODMAP and Low-FODMAP. In patients with IBS and bacterial overgrowth in the upper gastrointestinal tract, eating foods in the High-FODMAP group will stimulate the symptoms even more. On the other hand, if the patient is able to eat a Low-FODMAP diet, the symptoms of the disease can be controlled. It shows that the symptoms of the disease are related to diet. [10] Therefore, it is recommended for patients with either IBS or SIBO to choose a Low-FODMAP diet. [11-13] However, the Low-FODMAP guidelines list several types of fruits and vegetables which it may cause discrepancies in the communication of understanding between healthcare providers and patients. One of the reasons for the unsuccessful treatment is due to the patient’s inability to remember food items and thus unable to follow the instructions. Moreover, from research and application development surveys related to recommending Low-FODMAP fruit and vegetable diet according to patient health. It was found that no mobile applications were developed for this purpose in Thailand. Therefore, this research presents the design and development of a mobile application for visual identification of Low-FODMAP fruits and vegetables using machine learning techniques and giving advice on how to eat those vegetables or fruits. Finally, patients can use their mobile phones as a tool to help them choose fruits and vegetables which are suitable for one’s own health conveniently.

2. HEALTHCARE APPLICATION DESIGN

System design and development process can be shown in figure 1. This step starts with preparing a training dataset, each of which contains input data and its output data (Label). Then, that dataset will be used to create and train the model for supervised learning. This trained model will be used for fruit and vegetable detection and classifying. Then develop a mobile application to use as a user interface (UI) which is a way for users to import pictures of fruits and vegetables of interest as input and apply trained models to predict results. To indicate the type of such vegetables and fruits and the FODMAP type by showing the user through the UI application page.

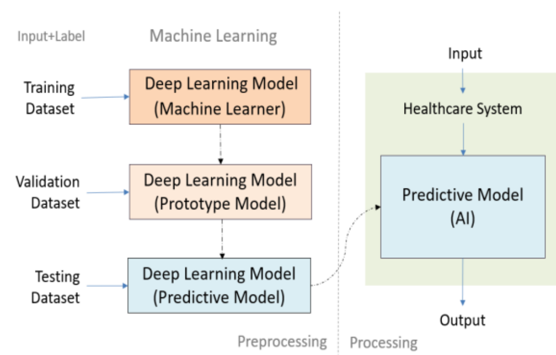


Figure 1. Machine Learning Process and Application.

2.1 Deep Learning Models

Most of the deep learning models are complex. Due to having more than millions of model parameters. It is the parameters obtained during the model training process, such as the neural network or the coefficients obtained from the linear regression.

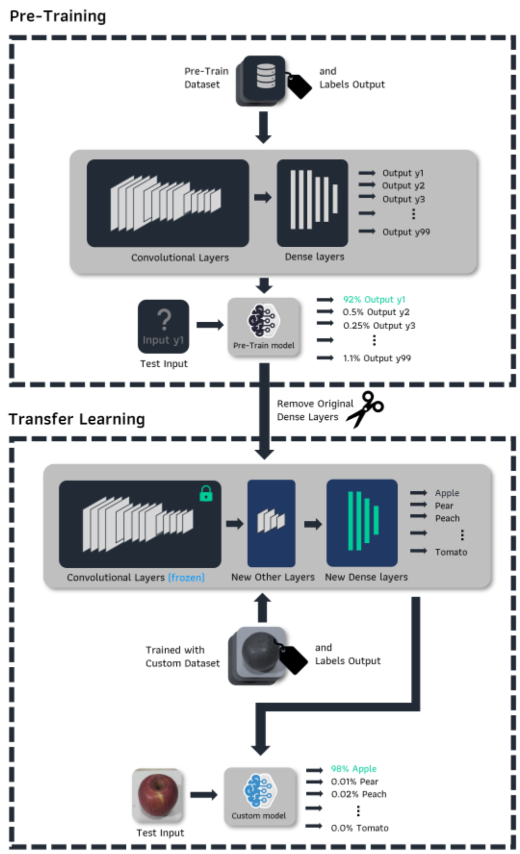


Figure 2. An Overview Process of Applying the Pre-training Model to Transfers Learning.

Keras applications have deep learning models available along with pre-trained weight training. These models can be used for prediction, feature extraction and fine-tuned through transfer learning. [14] This is a technique that reduces the time for training the deep learning models. By retraining some of the previously trained models with similar tasks. Become part of a new, more specialized model by extracting important features that need new ones and use a specialized dataset with the steps shown in figure 2.

These pre-training models were trained on the ImageNet dataset, which is a large-scale dataset that consists of 1.2 million labeled images from thousands of different classes. These classes cover a broad range of subjects, such as animals, plants, everyday objects, and various scenes. (Labeled images are images that have been annotated with descriptive tags or categories, which convey the content or position depicted in the image. These annotations are essential for training machine learning models using a supervised learning approach. The process involves assigning informative labels to images, enabling the models to learn and make predictions based on the provided labels during the training phase.

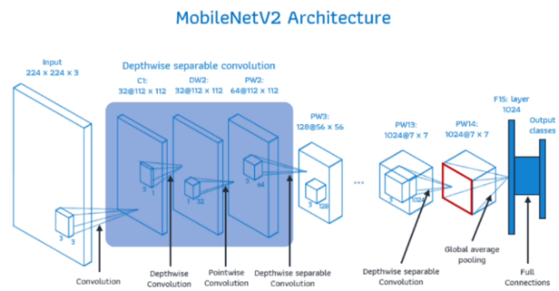


Figure 3. Architecture of MobileNetV2.

This research uses MobileNetV2 proposed by Howard AG, et al. [15], a deep learning model in Keras applications. MobileNetV2 is a convolutional neural network architecture which is designed to use few resources for work with limited resources. It's a small model which was developed to work well on mobile devices with the structure as shown in figure 3. The model is fast and has a low response time, scoring Top-1, Top-5 accuracy of 0.713 and 0.901 respectively. It is one of the pre-trained models of Keras suitable for use in classification or detection.

2.2 Data Preparation

This operation has steps as shown in figure 4. Initially, the images are prepared for use in the model training system and stored in a total folder of approximately 42,700 images. By dividing the image into 20 class, choosing to use 10 type of Low-FODMAP and High-FODMAP, consisting of fruits and vegetables as follow: Low-FODMAP fruits and vegetables: Kiwi, Lime, Lemon, Raspberry, Papaya, Dragon fruit, Orange, Sweet Pepper, Eggplant and Tomato as shown in table 1. High-FODMAP fruits and vegetables: Apple, Pear, Peach, Watermelon, Rambutan, Lychee, Pineapple, Beetroot, Onion and Corn as shown in table 2.

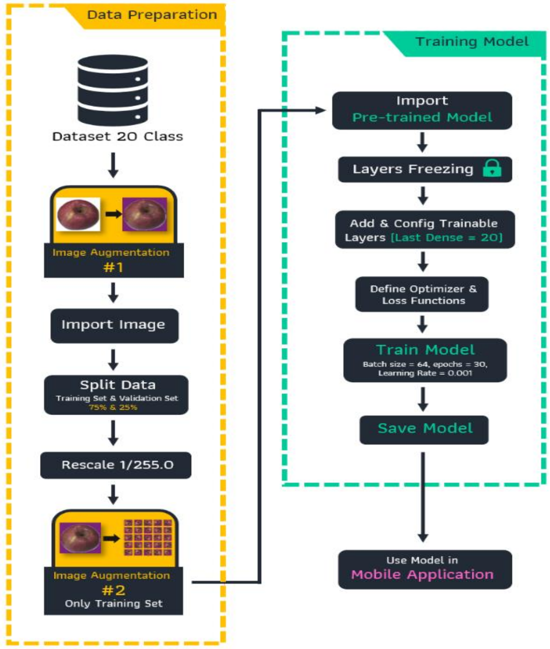


Figure 4. Action Steps in the Data Preparation and Model Training Section.

Table 1. List of Low-FODMAP Fruit and Vegetable Images Used.

Low FODMAP			
Fruit-Vegetable Name	Class Name	Number of Datasets	Number of Types/Species
Kiwi	L1_Kiwi	1284	1
Lime	L2_Lime	1013	1
Lemon	L3_Lemon	1533	1
Raspberry	L4_Raspberry	1062	2
Papaya	L5_Papaya	1187	1
Dragon Fruit	L6_DragonFruit	1112	1
Orange	L7_Orange	1287	2
Sweet Pepper	L8_SweetPepper	4608	4
Eggplant	L9_Eggplant	1014	1
Tomato	L10_Tomato	8830	9

Table 2. List of High-FODMAP Fruit and Vegetable Images Used.

High FODMAP			
Fruit-Vegetable Name	Class Name	Number of Datasets	Number of Types/Species
Apple	H1_Apple	10190	8
Pear	H2_Pear	8759	8
Peach	H3_Peach	2193	2
Watermelon	H4_Watermelon	1165	2
Rambutan	H5_Rambutan	924	1
Lychee	H6_Lychee	953	1
Pineapple	H7_Pineapple	1777	2
Beetroot	H8_Beetroot	817	1
Onion	H9_Onion	2388	3
Corn	H10_Corn	1644	2

In which some types of fruits and vegetables are covered by subspecies as well. In addition, images are also used that can easily identify the main characteristics of the breed. It is expected that users can see and take photographs while shopping in real life. For example, watermelon that has red and yellow flesh inside and is often cut in half to sell in stores. Therefore collecting such images in both manners uncut watermelon with watermelons cut in half, both varieties with red and yellow flesh inside.

Because most of the pictures usually the background color is white as shown in figure 5. This can lead to inefficient coaching from recognizing a feature that doesn't really represent the class such as remembering the number and white characteristics of the background instead of remembering the feature that indicates the fruit or vegetable. In practical use, it is unlikely that users will be able to take photographs of interesting fruits and vegetables along with a background that is clearly colored and white as in the dataset.

Figure 5. The Apple Dataset Image Styles.

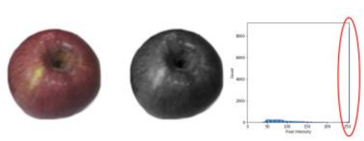


Figure 6. Histogram of the Pixel Intensity.

We plot histogram to show the color intensity of the image pixels converted to grayscale images. It was found that the intensity of 255 has the highest number as shown in figure 6, where the intensity of the grayscale image is between 0 - 255 (0 = black, 255 = white). We need to prevent this problem with the first image augmentation. We create new images or modify existing images (Only images that do not use the background from a general shooting but were replaced by white as mentioned above). In this operation, we randomly selected the white part of the background, specifying the RGB color values by assigning the 3 primary colors R (Red), G (Green) and B (Blue) representing the intensity of each color set. The random number range is between 0 and 255, as shown in figure 7. However, we have retained some of the images with white background, restoring the current dataset image to figure 8. and save the new image file to the specified folder location.

Figure 7. The specifying RGB color values.

Figure 8. The Apple Dataset Images After Random Background Color Change.

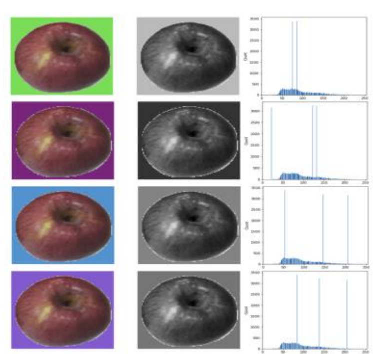


Figure 9. Histogram Comparison When Randomly Changing the Image Background Color.

The figure 9 showed comparing the histogram appearance of the image after randomly changing the background color. This results in instability in the background color of the dataset image while training the model, the importance of the feature can be reduced. Then extract all images into 2 parts which are the part used for training (Training dataset) and another used for validation after training (Validation dataset), divided into 75% and 25% respectively. This gives the number of images for each class as shown in figure 10.

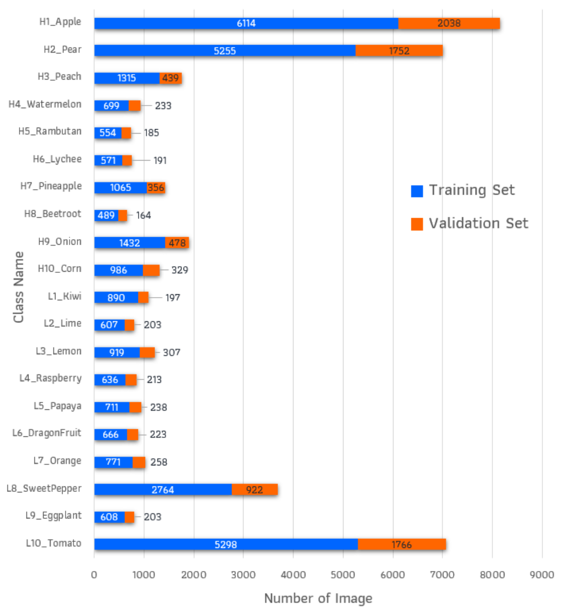


Figure 10. Number of Images Shared for Training and Validation.

In the process of rescale the image to 1/255 and require shuffle or randomly all datasets have a distribution of data or have a distribution without leaning in either way.

Then bring the image for training dataset into the second image augmentation process. In this step, we will determine to modify the image. By having a random image of no more than 10 degree, randomly rotating images up to 25 degrees, randomly zooming images up to 10% moving images left-right and up-down random on more than 10% and flip the horizontal image with both orientations randomly as shown in figure 11. It is also required to be filled with the part that is rotated beyond the edge of the image, making it the nearest feature, making the closest part. In this case the color of the background to fill in the excess space. The difference is shown in figure 12.

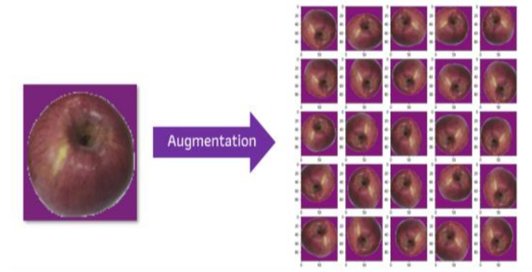


Figure 11. Before and After Image Through Image Augmentation Process.

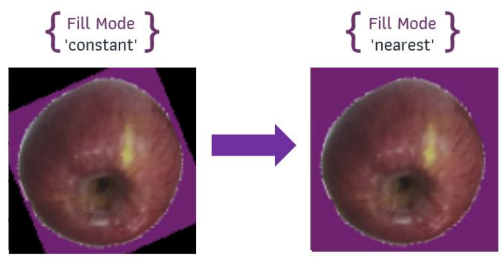


Figure 12. Image Difference Given Static Complement and an Image that Defines the Fill Based on the Closest Part.

For image augmentation, we mainly focus on adjusting the characteristics of each class of vegetables and fruits. Because if we bring the image to training immediately, it may cause overfitting which is a problem caused by ineffective recognition. For example, it might be remembered that when this number of red objects were centered, the image was apple, or you might remember that apple has to have 100% color and style accordingly. If the corners of the image are slightly distorted, it's not an apple right away. The others memorizing details in the images that may not be related to being apple.

2.3 Training the Model

After going through the image augmentation process, train the data to classify it with packets from the Keras and TensorFlow libraries. This process will provide training by running the pre-trained model, MobileNetV2, with the following steps:

- Bring layers from the old model to freeze the data, to avoid destroying any data which is in the future training cycle as an example of the steps in Figure 3.
- Adding new trainable layers on top of frozen layers to allow the model to learn how to transition from old properties to predictions with new datasets using the sequential layers addiction method.
- Complete the model settings and adjust each layer until the end, and assign the final layer to the shape according to the number of outputs that can be generated, 20 classes, which will get a new modified model.

Then define the optimizer part, which is the main component that drives the model to learn what we want. In this experiment, we used ADAM technique (Adaptive moment estimation), an adaptive gradient descent optimization tool commonly used in deep neural networks. It is also used to define the loss functions, which are tools used to measure how well the neural network is performing in each training session. In this research, we used standard multiclass cross-entropy loss.

Began the training by setting batch size = 64, set the maximum number of rounds of a training session called epochs = 30 rounds. Finally, set the learning rate, a hyperparameter that controls how much the model’s weight changes in one step of the training model process to be equal to 0.001.

While training the model, the loss and accuracy values of both the training and validation side are displayed in every epoch. The learning efficiency of the model is measured by these loss and accuracy values. The less the model’s loss or error value is well learned. However, on the other hand, the higher the accuracy, the better the model’s learning.

Generally, the more epochs cycle count, the loss decreases and accuracy increases until it reaches a fixed point. When finish training the model, it saves the model and converts it to a “tflite” file extension called TensorFlow Lite (TFLite). This is a tool that allows developers to bring TensorFlow models to run on mobile devices with smaller models, work faster and lower latency. It may reduce the accuracy a bit so that this model can be run in the android application that will be developed in the next section.

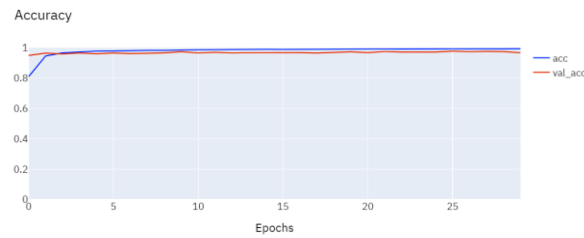


Figure 13. Performance of Learning Curves.

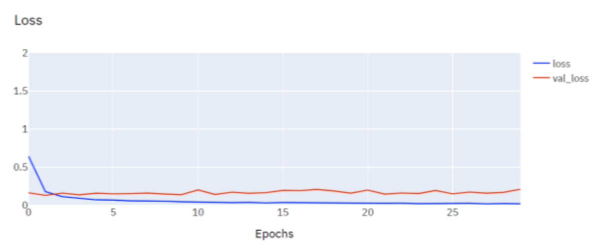


Figure 14. Optimization of Learning Curves.

The figure 13 and figure 14 shows the relationship of accuracy to the increasing number of epochs (Performance learning curves) and the relationship of loss to the number of epochs increasing (Optimization learning curves) respectively. Where the blue line is the accuracy or loss of training and the red line is the accuracy or loss of the validation. This allows the model’s learning to be diagnosed. To determine from the learning curve whether the training dataset and validation dataset represent the appropriate data in the model development process.

Healthcare Application Development: In developing healthcare applications, Flutter framework is a cross-platform framework used to develop native mobile applications that can be used for both android and iOS using Dart language. We can show the overall working process of mobile application from the beginning of both user side and mobile application side as shown in figure 15.

On the mobile application side, import the model file with the extension .tflite and configure the label or the result indicating the fruits and vegetables of all 20 classes according to the order from which they have been trained. On the user’s side, there are two ways to choose the format of the image input: select a file from within the device or choose to press the capture button at that moment as shown in figure 16.

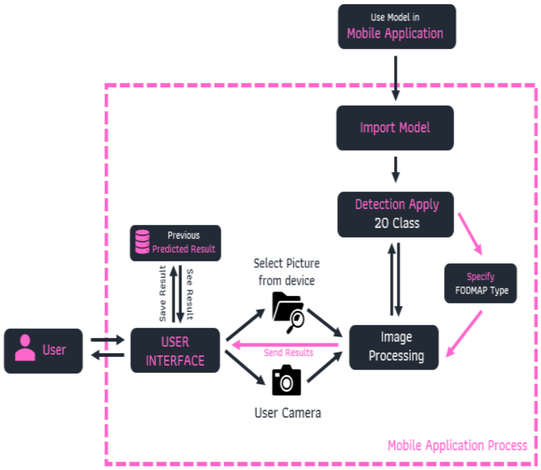


Figure 15. An Overview of the Mobile Application Operation in this Research.

Figure 16. Application Example in the Main Page for Selecting the Detection Method.

When the user selects an input image, it goes through image processing which gives the appropriate image for prediction. Then begin predicting the class from the image above. When it was concluded that it was any type of fruit and vegetable from the selection of the greatest probabilities. For example, when a user takes a picture of a lime after predicting and concluding that it is a lime. It is also indicated to classify whether it is Low-FODMAP or High-FODMAP. In this example case, its class L2_Lime is Low-FODMAP, the display color will be green which indicates that it can be eaten safely. And vice versa, High-FODMAP emphasizes the use of red which indicates caution such patients should not eat it as shown in figure 17.

Figure 17. Screen Preview for Low-FODMAP /High-FODMAP Fruit and Vegetable Prediction.

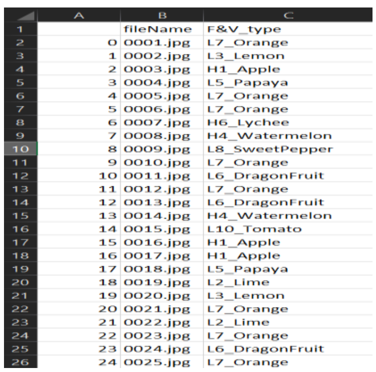
In addition, when the results are displayed on the screen users can choose to save the results for later viewing. The system will save the history of images and data of the prediction results obtained in the result page and show the date and time that the user has selected that kind of fruit and vegetable to predict as well. Moreover, the system also allows users to click to see detailed information and recommendations of the aforementioned fruits and vegetables as shown in figure 18.

Figure 18. Preview of the Prediction History Results Screen that the User Chooses to save.

3. TESTS AND RESULTS

In this research, we test the performance of the application. The experiment used 2,000 testing dataset to find the prediction results from the model obtained from the training. The test dataset used here consisted of 20 classes of fruits and vegetables divided into 100 equal classes of each class, stored in a random folder, as an example in figure 19. And create an annotation file for identifying the image names based on the actual results that the program should predict, as an example in figure 20.

Figure 19. An Example Image for Testing the Model's Performance.



	A	B	C
1		fileName	F&V_type
2	0	0001.jpg	L7_Orange
3	1	0002.jpg	L3_Lemon
4	2	0003.jpg	H1_Apple
5	3	0004.jpg	L5_Papaya
6	4	0005.jpg	L7_Orange
7	5	0006.jpg	L7_Orange
8	6	0007.jpg	H6_Lychee
9	7	0008.jpg	H4_Watermelon
10	8	0009.jpg	L8_SweetPepper
11	9	0010.jpg	L7_Orange
12	10	0011.jpg	L6_DragonFruit
13	11	0012.jpg	L7_Orange
14	12	0013.jpg	L6_DragonFruit
15	13	0014.jpg	H4_Watermelon
16	14	0015.jpg	L10_Tomato
17	15	0016.jpg	H1_Apple
18	16	0017.jpg	H1_Apple
19	17	0018.jpg	L5_Papaya
20	18	0019.jpg	L2_Lime
21	19	0020.jpg	L3_Lemon
22	20	0021.jpg	L7_Orange
23	21	0022.jpg	L2_Lime
24	22	0023.jpg	L7_Orange
25	23	0024.jpg	L6_DragonFruit
26	24	0025.jpg	L7_Orange

Figure 20. Annotation File Preview to Prepare to Compare Test Results.

Then load the trained model and use it to predict the images used for such tests. The result is probabilities from the prediction of each image in the form of an array. For example [0.0000001943, …, 0.9000451565, 0.0000001111, 0.0999539122] according to the location of all 20 defined classes. Which class has the most probabilities is considered the class as the result of the prediction. From figure 21, the example case is considered class L8_SweetPepper because it has probabilities at 0.9000451565 or confidence at about 90%.

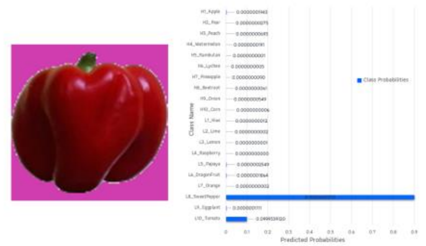
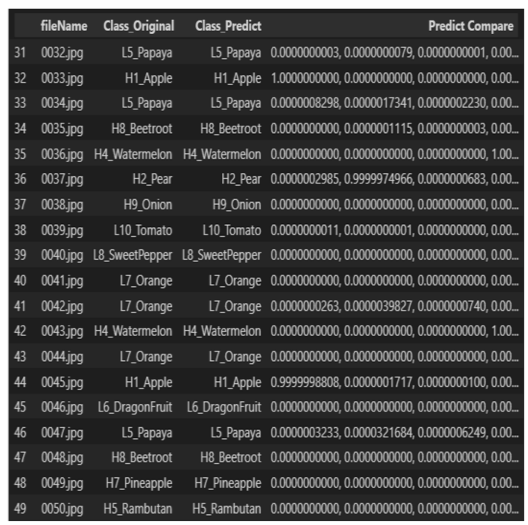


Figure 21. An Example of a Prediction with the Probabilities of L8_SweetPepper Most Valuable.

Bring the results to be stored as a data frame and linked to the data from the previous annotation file, and collect the probabilities shown in figure 22. When we bring the plot together with the image used for testing, it will be as shown in figure 23.



	fileName	Class_Original	Class_Predict	Predict Compare
31	0032.jpg	L5_Papaya	L5_Papaya	0.0000000003, 0.0000000079, 0.0000000001, 0.00...
32	0033.jpg	H1_Apple	H1_Apple	1.0000000000, 0.0000000000, 0.0000000000, 0.00...
33	0034.jpg	L5_Papaya	L5_Papaya	0.0000008298, 0.0000017341, 0.0000002230, 0.00...
34	0035.jpg	H8_Beetroot	H8_Beetroot	0.0000000000, 0.0000001115, 0.0000000003, 0.00...
35	0036.jpg	H4_Watermelon	H4_Watermelon	0.0000000000, 0.0000000000, 0.0000000000, 1.00...
36	0037.jpg	H2_Pear	H2_Pear	0.0000002985, 0.9999974966, 0.0000000683, 0.00...
37	0038.jpg	H9_Onion	H9_Onion	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
38	0039.jpg	L10_Tomato	L10_Tomato	0.0000000011, 0.0000000001, 0.0000000000, 0.00...
39	0040.jpg	L8_SweetPepper	L8_SweetPepper	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
40	0041.jpg	L7_Orange	L7_Orange	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
41	0042.jpg	L7_Orange	L7_Orange	0.0000000263, 0.0000039827, 0.0000000740, 0.00...
42	0043.jpg	H4_Watermelon	H4_Watermelon	0.0000000000, 0.0000000000, 0.0000000000, 1.00...
43	0044.jpg	L7_Orange	L7_Orange	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
44	0045.jpg	H1_Apple	H1_Apple	0.9999998808, 0.0000001717, 0.0000000100, 0.00...
45	0046.jpg	L6_DragonFruit	L6_DragonFruit	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
46	0047.jpg	L5_Papaya	L5_Papaya	0.0000003233, 0.0000321684, 0.0000006249, 0.00...
47	0048.jpg	H8_Beetroot	H8_Beetroot	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
48	0049.jpg	H7_Pineapple	H7_Pineapple	0.0000000000, 0.0000000000, 0.0000000000, 0.00...
49	0050.jpg	H5_Rambutan	H5_Rambutan	0.0000000000, 0.0000000000, 0.0000000000, 0.00...

Figure 22. A Prediction Results of the Model with 32-50 Test Images and Actual Results.

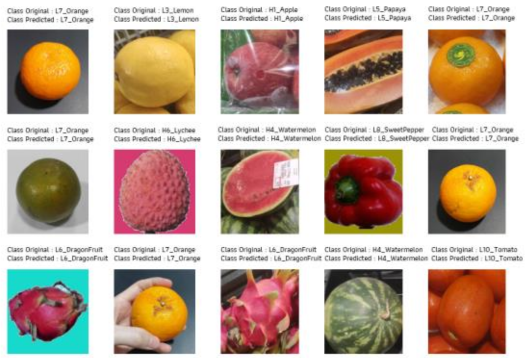


Figure 23. Example of a Test Image, the Predicted Results of the Model with Actual Results.

The values from this comparison table are then calculated as the confusion matrix, which is an important tool for evaluating prediction results. By finding the ratio between the predicted label and the actual result (actual label) from a total of 2,000 images as shown in figure 24.

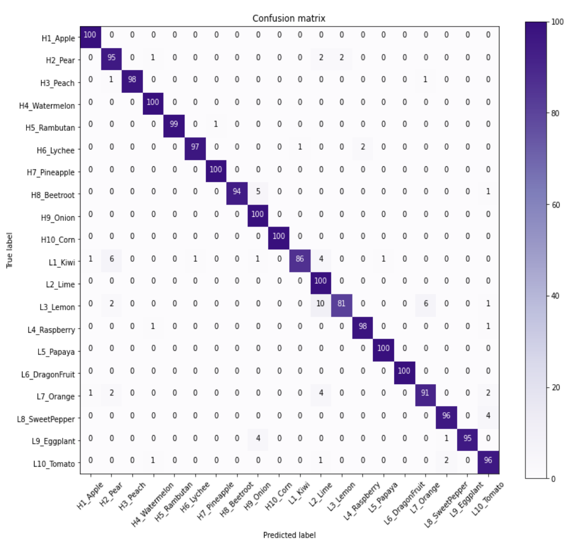


Figure 24. Confusion Matrix of Models including 20 Classes.

From figure 24, the confusion matrix model can explain the comparison of predicted results with actual results in all classes. For example, a class H3_Peach result can be predicted to match the expected result of up to 98 of 100 images. However, some images were predicted to be class H2_Pear and class L7_Orange, one each. The numbers within the confusion matrix can be used to calculate the values of true positive, true negative, false positive and false negative as follows.

- True positive (TP) is the case when both the predicted result and the actual result are also class X (correct because the prediction is class X).
- True negative (TN) is the case when both the predicted result and the actual result are not class X either (correct because the prediction is not class X).
- False positive (FP) is if the predicted result does not match the actual result. The prediction is class X, but the actual result is not class X (wrong because the prediction is class X).
- False negative (FN) is the case when the prediction result does not match the actual result. In this case, it predicts that it is not class X, but the actual result is class X (wrong because it predicts that it is not class X).

And all such values can be calculated to evaluate the effectiveness of the model prediction. Including precision, recall and F1-score of each class [20] as follows.

- Precision is the value of accuracy as equation (1). Comparison between predicted to be class X correct and true to predicted to be class X, Precision=((TP))/((TP+FP)) but the actual result is not.

$$Precision = \frac{(TP)}{(TP+FP)} \qquad (1)$$

- Recall (True Positive Rate : TPR) is the value that tells the part that the program predicts to be class X, how much ratio of all must be class X as equation (2). It is like an accuracy that considers only one class of interest.

$$Recall = \frac{(TP)}{(TP+FN)} \qquad (2)$$

F1-score is the harmonic mean between precision and recall as equation (3), to measure the model's ability.

$$F1\text{-}score = 2 \times \frac{(Precision \times Recall)}{(Precision+Recall)} \qquad (3)$$

Therefore, when bringing the confusion matrix from figure 24 to display a new result as a separate class that is mainly interested, by calculating the true positive, true negative, false positive and false negative values of each class. It is shown in figure 25.

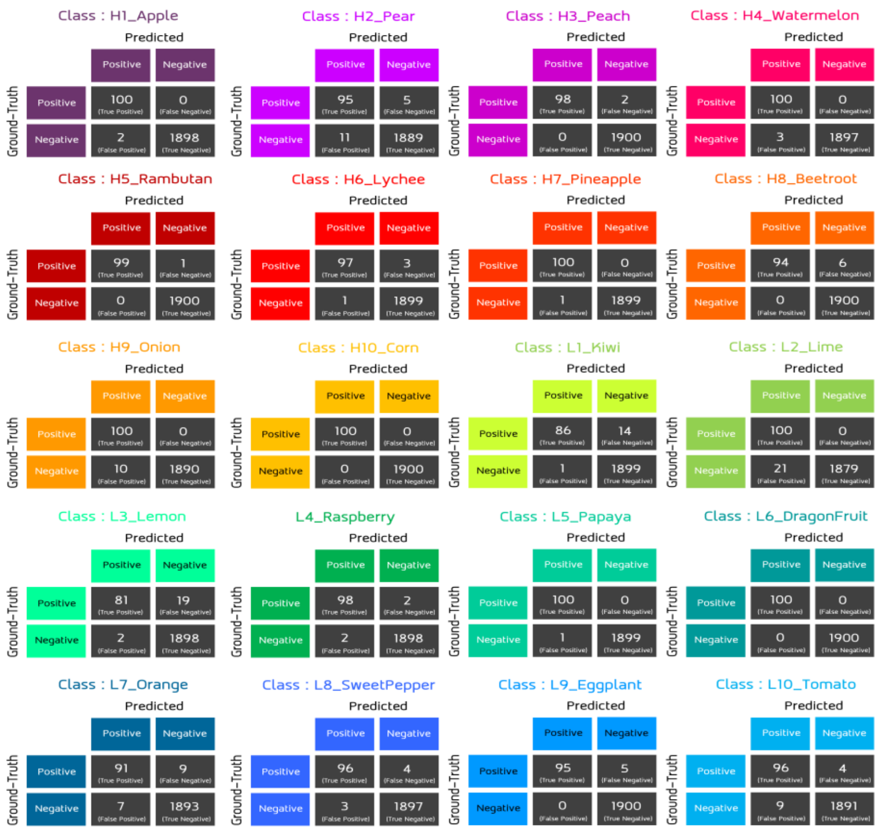


Figure 25. Confusion Matrix Identifies a Separate Class Target.

And bring these values to calculate the three performance estimators and put them in the table as shown in figure 26.

	precision	recall	F1-score	support
H1_Apple	0.9804	1.000	0.9901	100
H2_Pear	0.8962	0.9500	0.9223	100
H3_Peach	1.0000	0.9800	0.9899	100
H4_Watermelon	0.9709	1.0000	0.9852	100
H5_Rambutan	1.0000	0.9900	0.9950	100
H6_Lychee	0.9898	0.9700	0.9798	100
H7_Pineapple	0.9901	1.0000	0.9950	100
H8_Beetroot	1.0000	0.9400	0.9691	100
H9_Onion	0.9091	1.0000	0.9524	100
H10_Corn	1.0000	1.0000	1.0000	100
L1_Kiwi	0.9885	0.8600	0.9198	100
L2_Lime	0.8264	1.0000	0.9050	100
L3_Lemon	0.9759	0.8100	0.8852	100
L4_Raspberry	0.9800	0.9800	0.9800	100
L5_Papaya	0.9901	1.0000	0.9950	100
L6_DragonFruit	1.0000	1.0000	1.0000	100
L7_Orange	0.9286	0.9100	0.9192	100
L8_SweetPepper	0.9697	0.9600	0.9648	100
L9_Eggplant	1.0000	0.9500	0.9744	100
L10_Tomato	0.9143	0.9600	0.9366	100

Figure 26. Prediction Performance for Each Class of the Model from the Same Number of Datasets, Each Class of 100 Images.

When we summarized the predictive performance of each class, the overall performance was considered good. Most of the images can be predicted to match the expected results. But there are still some images from the class that are misleading from reality. Part of this may be due to the dataset, which is a perspective image based on real life photography. There are still quite a few characteristics of some fruits and vegetables that are similar.

Finally, we plot area under the curve (AUC) and calculate receiver operator characteristic (ROC) of each of the 20 class images using true positive rate (TPR) and false positive rate (FPR). ROC curve is a way to tell that the test can separate the class of vegetables and fruits well apart from each other. A good graph should have a corner point that is closest to the top left. Because the upper left corner of the ROC curve is the point where sensitivity (recall) = 100% and specificity (true negative rate) = 100%. It can also indicate the value of AUC, which indicates overall test accuracy. If the closer the AUC value is to 1, the greater the test accuracy, because it represents the probability that the prediction set will give the most accurate results as shown in figure 28.

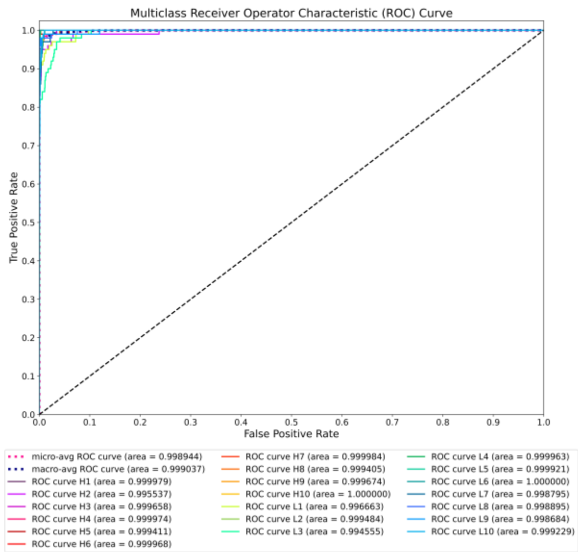


Figure 27. Model ROC Curve from Testing with the Datasets.

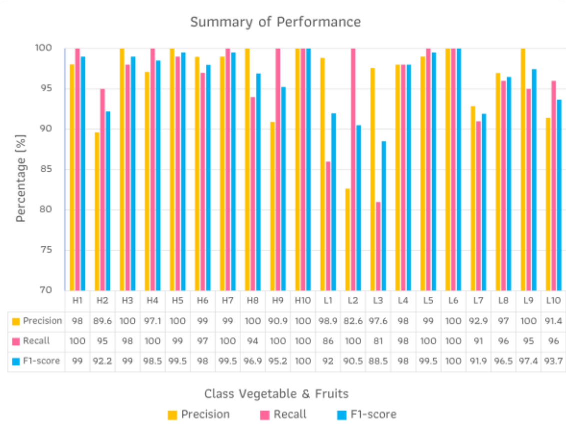


Figure 28. Graph Summarizing the Predictive Performance Across All Classes.

4. DISCUSSION AND CONCLUSIONS

In conclusion, from the fruits and vegetables prediction performance test to identify and classify the type of fruits and vegetables. It was found that the overall prediction was satisfactory. It has average precision of 96.55%, average recall (true positive rate) with balance accuracy of 96.30% and average F1-score of 96.29%. From the recall, it shows that overall, if using images according to the class of interest, it can be predicted correctly. When we compare the total number of images that should result in those classes of interest. This is due to a large number of true positive (TP) values and a small number of false negative (FN) values. However, there was still a minority of images that were incorrectly predicted at 74 of the 2,000 images. The above average precision represents the accuracy of the prediction. That is, it has the probability that can be distinguished to select only the classes of good interest due to the presence of a large number of true positive (TP) and a small number of false positive (FP), and the average F1-score or harmonic mean between precision and recall at a good level.

In addition, comparing the efficiency of each class from Figure 28, it was found that vegetables and fruits in class H10_Corn (Corn) and class L6_DragonFruit (DragonFruit) had the best predictive performance. Because it can predict correctly with recall = 100% and there is no wrong prediction from other classes that it is this type of class as well. This makes precision and F1-score equal to 100%. On the other hand, fruits and vegetables of class L3_Lemon (Lemon) had the least predictive performance. Because the nature of this fruit is very close to other classes of fruit, especially oranges and lemons. As for the recall value caused by the wrong prediction of another class, the F1-score is the least, at 81.00% and 88.52% respectively.

This results in a class L2_Lime (Lime) that even has its own true positive rate or recall of 100% (Lime images can predict lime are all correct). However, there is another class that is wrongly predicted to be the class with the largest number of limes at 21 images, so the precision is the lowest at 82.64%.

When considering the ROC curve as shown in Figure 27. It is found that class 12 or L3_Lemon clearly has less ROC value than other classes. It shows that the model is capable of separating this class from the other classes the least, having an area of about 0.994. On the other hand, class H10_Corn and L6_DragonFruit have the best ability to distinguish a class image from another class, having the area = 1.00. This can help verify performance in addition to the precision, recall and F1-score calculations above. However, it was found that all classes had AUC values closer to 1.0 and the angular characteristics of the graphs were very aligned to the top left, there was little difference in the graphs, and most of the lines were overlapping. Therefore, demonstrating the presence of a very high overall test accuracy, there is a probability that this prediction set which the results are true as that actual.

When we compare the efficiency of the coaching model for each cycle of epochs as shown in Figure 13 and 14. It shows that the performance and optimization learning curves are graphed as the good fit learning curve. That is, both training loss and validation loss are continuously decreasing to a fixed point. The two graph lines have a small gap between them, which indicates that the model is well learned. This trained model can be used to predict images of fruits and vegetables that have never been found from the precise training (The placement or shooting style other than the image used for training but must be a specified species). We call this case the generalization of new data. Therefore, it can be concluded that the model from the above training can be used as a prototype system for further practical use. However, we should add images dataset for better performance, especially Lemon and Lime classes, for example, which may increase the overall performance by up to 99%.

However, the restrictions on the use of the application due to the images collected does not cover the level of ripeness and abnormal damage from fungi in each fruit. Therefore, if the user takes or selects an image of fruits and vegetables that have too much ripeness or unusual appearance than the images used in the training such as fruits and vegetables that are close to spoil or surrounded by fungi, it can result in erroneous predictions. In addition, there is also the issue of characteristic constraints that are an important feature to indicate the class of vegetables and fruits as actual life fruits and vegetables which do not look the same in 360 degrees. In some ways, it can be very similar to other fruits and vegetables. This can result in erroneous predictions. For efficiency in use, users should choose to photograph the fruits and vegetables they are interested in from the perspective that is most clearly descriptive of the type. Because the system for classifying the types of vegetables and fruits is a form of multi-class classification. It is able to classify the classes of fruits and vegetables with multiple classes, but only selects the label output that is the final answer as the only answer. It does not use the multi-label classification format in which multiple label outputs can be specified in the event that the image contains more than one type of fruit and vegetable. Therefore, in order to use it, users need to take pictures of the fruits and vegetables they are interested in with only one type per image.

Finally, species restrictions are not present in the dataset due to the type-restricted images. Therefore, it does not cover other types of vegetables and fruits other than the images in the dataset mentioned, and other selected subspecies of it. In future developments, a prediction result class should be added in addition to the Low-FODMAP and High-FODMAP classes in order to be able to identify such fruit and vegetable types. There are also other environmental factors such as the quality of the device camera, photography brightness, that may affect the predictive ability to classify fruits and vegetables as well.

ACKNOWLEDGEMENTS

This research was funded by Srinakharinwirot University (Grant number 554/2564). We would like to thank the Strategic Wisdom and Research Institute, Srinakharinwirot University, Thailand for supporting this research until it was successfully completed.

REFERENCES

[1] International Foundation for Gastrointestinal Disorders [https://www.aboutibs.org/facts-about-ibs/statistics.html.] Retrieved May 25, 2022.

[2] Longstreth GF, Thompson WG, Chey WD, Houghton LA, Mearin F, Spiller RC. Functional bowel disorders. Gastroenterology. 2006 Apr;130(5):1480-91.doi:

10.1053/j.gastro.2005.11.061. Erratum in: Gastroenterology. 2006 Aug;131(2):688. PMID: 16678561.

[3] American College of Gastroenterology Task Force on Irritable Bowel Syndrome, Brandt LJ, Chey WD, Foxx-Orenstein AE, Schiller LR, Schoenfeld PS, Spiegel BM, et al. An evidence-based position statement on the management of irritable bowel syndrome. Am J Gastroenterol. 2009 Jan;104 Suppl 1:S1-35. doi: 10.1038/ajg.2008.122. PMID: 19521341.

[4] Dukowicz AC, Lacy BE, Levine GM. Small intestinal bacterial overgrowth: a comprehensive review. Gastroenterol Hepatol (N Y). 2007 Feb;3(2):112-22. PMID: 21960820; PMCID: PMC3099351.

[5] Rezaie A, Buresi M, Lembo A, Lin H, McCallum R, Rao S, et al. Hydrogen and Methane-Based Breath Testing in Gastrointestinal Disorders: The North American Consensus. Am J Gastroenterol. 2017 May;112(5):775-784. doi: 10.1038/ajg.2017.46. Epub 2017 Mar 21. PMID: 28323273; PMCID: PMC5418558.

[6] Pimentel M, Saad RJ, Long MD, Rao SSC. ACG Clinical Guideline: Small Intestinal Bacterial Overgrowth. Am J Gastroenterol. 2020 Feb;115(2):165-178. doi: 10.14309/ajg.0000000000000501. PMID: 32023228.

[7] Aziz I, Törnblom H, Simrén M. Small intestinal bacterial overgrowth as a cause for irritable bowel syndrome: guilty or not guilty? Curr Opin Gastroenterol. 2017 May;33(3):196-202. doi: 10.1097/MOG.0000000000000348. PMID: 28257307.

[8] Chen B, Kim JJ, Zhang Y, Du L, Dai N. Prevalence and predictors of small intestinal bacterial overgrowth in irritable bowel syndrome: a systematic review and meta-analysis. J Gastroenterol. 2018 Jul;53(7):807-818. doi: 10.1007/s00535-018-1476-9. Epub 2018 May 14. PMID: 29761234.

[9] Gibson PR, Shepherd SJ. Personal view: food for thought--western lifestyle and susceptibility to Crohn's disease. The FODMAP hypothesis. Aliment Pharmacol Ther. 2005 Jun 15;21(12):1399-409. doi: 10.1111/j.1365-2036.2005.02506.x. PMID: 15948806.

10. Hayes PA, Fraher MH, Quigley EM. Irritable bowel syndrome: the role of food in pathogenesis and management. Gastroenterol Hepatol (N Y). 2014 Mar;10(3):164-74. PMID: 24829543; PMCID: PMC4014048.

[11] Hill P, Muir JG, Gibson PR. Controversies and Recent Developments of the Low-FODMAP Diet. Gastroenterol Hepatol (N Y). 2017 Jan;13(1):36-45. PMID: 28420945; PMCID: PMC5390324.

[12] Magge S, Lembo A. Low-FODMAP Diet for Treatment of Irritable Bowel Syndrome. Gastroenterol Hepatol (N Y). 2012 Nov;8(11):739-45. PMID: 24672410; PMCID: PMC3966170.

[13] Nanayakkara WS, Skidmore PM, O'Brien L, Wilkinson TJ, Gearry RB. Efficacy of the low FODMAP diet for treating irritable bowel syndrome: the evidence to date. Clin Exp Gastroenterol. 2016 Jun 17;9:131-42. doi: 10.2147/CEG.S86798. PMID: 27382323; PMCID: PMC4918736.

[14] Deep Learning API in Python, Keras Applications, [https://keras.io/api/applications/]

[15] Andrew G. Howard, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. Google Inc., 2017.

[16] Charu C. Aggarwal. Neural Networks and Deep Learning. Springer, 2018.

[17] François Chollet. Deep Learning with Python 1st Edition. Nanning, 2018.

[18] Venkata Reddy Konasani. Machine Learning and Deep Learning Using Python and TensorFlow 1st Edition. McGrawHill., 2021.

[19] Bharathi, Confusion Matrix for Multi-Class Classification., 2020.

[20] Margherita Grandini, et al. Metrics For Multi-Class Classification: an Overview. CRIF S.p.A., 2020.

[21] Jason Brownlee. Weight Initialization for Deep Learning Neural Networks., 2021.

[22] Yuxi (Hayden) Liu. Python Machine Learning By Example. Packt Publishing Ltd., 2020.

[23] Horea Muresan, Mihai Oltean, Fruit recognition from images using deep learning, Acta Univ. Sapientiae, Informatica Vol. 10, Issue 1, pp. 26-42, 2018.

[24] Minut Mihai-Dimitrie. Fruit Dataset Fruit Classification. University "Alexandru Ioan Cuza" of IASI., 2021

[25] Shepherd SJ, Parker FC, Muir JG, Gibson PR. Dietary triggers of abdominal symptoms in patients with irritable bowel syndrome: randomized placebo-controlled evidence. Clin Gastroenterol Hepatol. 2008 Jul;6(7):765-71. doi: 10.1016/j.cgh.2008.02.058. Epub 2008 May 5. PMID: 18456565.

Watcharachai Wiriyasuttiwong received the B.Eng. in Electrical Engineering from Prince of Songkha University (PSU), Thailand in 1994., M.Eng. in Electrical Engineering, Distinguished Thesis, Chiangmai University in 1999. Currently, he is a Assistant Professor in Department of Computer Engineering, Faculty of Engineering at Srinakharinwirot University (SWU), Thailand. His interests include AI, Machine Learning, Knowledge Engineering and Fuzzy Expert System.

Kamon Chaiyasit received MSc in Pharmacology from Chulalongkorn University Thailand (2007), PhD in Nutrition from USA (2012) and Bachelor of Law and PhD in Innovation Communication in Political and Local administration from STOU, Thailand (2020), Palliative care in hospitalist and intervist from Harvard Medical School and Fellow of American College of Nutrition. He holds American Board Certified in Nutritional Wellness. He is a President of Integrative Nutrition and Herbal Medicine Association. He works at Integrative Cancer Center at Esperance Clinic Bumrungrad International hospital.

Maethaphan Kitporntheranunt received the M. D. from Faculty of Medicine, Chiang Mai University, Thailand in 1997. In 2001 he joined the Faculty of Medicine, Srinakharinwirot University, Thailand, as an instructor where he became an Associate Professor in 2018. He had research experience/ subspecialty training at King Chulalongkorn Memorial Hospital, Thailand and KK Women's and Children's Hospital, Singapore in 2003 and 2009 respectively. His interests include Medical Education, Medical Instrument and Innovation, Health Technology and Medical Expert System.

Varit Srisukthaveerat received Bachelor of Business Administration in International Logistics from Chulalongkorn University, Thailand (2015), Master of Business Administration in Marketing and Management from Western University, Thailand (2018), Bachelor of Law from STOU, Thailand (2020), and bachelor of public health from Western University, Thailand (2022). PhD in Innovation Communication in Political and Local administration from STOU. He was invited lecturer in labour law at college of integrative medicine, Dhurakij Pundit University. He is a legal consultant of integrative nutrition and herbal medicine association and Senizen wellness center.

Wittayatorn Tokeaw received Bachelor of Education in Educational Psychology from Chulalongkorn University, Thailand, Master of Education in Audio-Visual Communications from Chulalongkorn University, Thailand, and Doctor of Philosophy in Rural planning and Development, Maejo University, Thailand. He is Associate Professor of School of Communication Arts, Sukhothai Thammathirat Open University and Chairman of Doctor of Philosophy Program in Innovation Communication for Political and Local Administration.

Supaporn Sridee received Bachelor of Arts in Communication Arts from Chulalongkorn University, Thailand (1984), Bachelor of Law from Thammasat University, Thailand (1993), Master of Arts in Communication Arts from Chulalongkorn University, Thailand (1996), Doctor of Philosophy in Mass Communication, Thammasat University, Thailand (2005). She is Associate Professor of School of Communication Arts, Sukhothai Thammathirat Open University.

Vibhakorn Permpoon received *Doctor of Medicine from* Ramathibodi Hospital, Mahidol University, Thailand (1990), Fellow of Internal Medicine from Chulalongkorn University Hospital, Thailand (1997), Fellow of Gastroenterology from Chulalongkorn University Hospital, Thailand (1999), and Master Degree of Science from Chulalongkorn University, Thailand (1999). She is a gastroenterologist at Gastroenterology & Hepatology, Bumrungrad International hospital.

Natthapol Tiawattananon received the B.Eng. in Computer Engineering from Srinakharinwirot University (SWU), Thailand in 2022. His interests include AI, Machine Learning, Image Processing, Healthcare System, Software Engineering and UX/UI Design.