



Food Recognition of Malaysian Meals for the Management of Calorie Intake

Nursuriati Jamil¹, Shahrul Raimie Roslan², Raseeda Hamzah³, Izzad Ramli⁴

^{1,2,3,4}Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Selangor, lizajamil@computer.org, shah.rymie@gmail.com, raseeda@fskm.uitm.edu.my, izzad@fskm.uitm.edu.my

ABSTRACT

This work served the weight management module of a complete self-management e-Health intervention program to improve patient activation and self-management behaviours among patients with metabolic syndrome in Malaysia. Since calorie intake is a crucial factor in weight management, we proposed calorie counting of food intake using food images to ease the burden of manual calorie counting. Even though there are several similar applications available, the recognition of Malaysia's common meals was very low as the availability of the food images was scarce. Therefore, we collected a total of 1,000 images comprising 50 categories of common Malaysian meals. Food recognition is done using a pre-trained MobileNet learning model on Tensorflow Lite deep learning environment. At the time of writing, the average accuracy of food recognition was at 80 percent. A lot of work need to be done in the future to increase the food recognition's accuracy, particularly with food of high resemblance.

Key words : Image recognition, MobileNet, Tensorflow Lite, calorie counter, e-Health.

1. INTRODUCTION

In many countries, the use of mobile health (m-Health) apps to enable patients' self-management of long-term conditions is very encouraging. International clinical guidelines highly recommend the inclusion of self-management programs in the management of metabolic syndrome, and such programs have been associated with improved health conditions for patients with obesity, hypertension, and diabetes. Obesity is a growing health problem that affected people all around the world, Malaysia included [1]. Numerous studies showed that Malaysians lacked nutritional knowledge and neglected the nutritional aspects of food [2]. With today's technological advancement, the process of acquiring information of food's nutritional facts is effortless with the help of computer vision and food database. Many researches were done involving mobile-based food image recognition and produced positive results. DeepFood [3] is a computer-aided dietary assessment system developed using deep learning-based food image recognition, and it achieved high accuracy at recognizing food using UEC-256 and Food-101 database. Another example is Nutrinet [4], a deep learning food and drink recognition system containing 520 classes which achieved an accuracy of 94.47% on a dataset containing 130,517 images.

Some of the mobile applications are commercially available such as CalorieMama [5], Nutritionix [6] and BiteSnap [7]. However, all of them have low applicability in Malaysia due to lack of information available on Malaysian food. They only focused on recognising food that were commonly consumed globally such as steak, burger, sushi, and pre-packaged food. This is a concern for Malaysians who wanted to exercise a healthy diet because of the lack of nutritional information of Malaysian food. This paper aims to provide a platform containing a library of nutritional information on Malaysian food, and the ability to recognise Malaysian food conveniently based on image recognition by utilising the strength of deep transfer learning technique [8].

Currently, the image recognition tasks no longer requires feature engineering process due to the availability of large-scale annotated datasets such as ImageNet [9] and the existence of ImageNet Large Scale Visual Recognition Challenge (ILSVRC), a competition that evaluated algorithms used for large-scale object detection and classification. The revival of deep convolutional neural network (DCNN) increased the computational power and availability of massive training datasets [10] [18-19] and contributed to the simplicity of the image recognition due to the automation of feature engineering in the deep convolutional neural network architecture. DCNN, however, comes with its challenge in implementing it in practice. To fully utilized the architecture, the number of data set required for training process to perform optimally is huge [11], which cause the data collection and training process to become expensive and time-consuming. Therefore, we proposed to adopt the transfer learning DCNN model for our proposed food recognition. Transfer learning is a deep learning design technique in which a trained model in one task is repurposed for another task [12]. It eliminates the need for a large dataset for training and allows rapid progress or improved performance by transferring the knowledge of a model that has been trained on a large and complex dataset such as GoogleNet into a new task [13]. In this paper, we reviewed three popular transfer learning models that are AlexNet, GoogleNet and MobileNet.

2. RELATED WORK

AlexNet DCNN architecture proved its success in recognition and classification performance by winning the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012. It is trained on an enormous dataset: 1.2 million images of size 256 x 256 divided into 1000 categories. AlexNet consists of five convolutional layers, three pooling, and two fully-connected layers with approximately 60 million free parameters. GoogleNet, compared to other DCNN architecture, is much more powerful and sophisticated architecture. It consists of two convolutional layers, two pooling layers, and nine 'Inception' layers, which is a new module introduced by GoogleNet. The Inception layer is made up of six convolutional layers and one pooling layers, the results of the process in each layer in Inception layer are then concatenated into one output. GoogleNet is currently the most significant DCNN architecture in ILSVRC challenge, with 5.5% top-5 classification error, compared to AlexNet at 15.3% error ([14]). Mobilenet is a class of a more efficient deep learning models created specifically for mobile and embedded vision applications. It is based on a streamlined architecture that used depth-wise separable convolutions to build lightweight deep neural networks. The goal of the model that efficiently trade off accuracy and latency and allowed for calibration of parameters based on the problem's need [15]. Mobilenet introduced the concept of depth-wise-separable convolutional which is a further improvement of the convolutional layer that can significantly reduced computation and model size. Table 1 summarizes the comparisons of the three models. Based on the comparisons, Mobilenet transfer learning is chosen due to the flexibility of images size, the accuracy of the model, size of the model, and the suitability in mobile devices.

Table 1: Comparisons of popular transfer learning models

Model	Dataset size	Accuracy	Model size
AlexNet	Medium	15.3% top-5 classification error	Small
GoogleNet	Large	5.5% top-5 classification error	Large
Mobilenet	Any size	70% accuracy using ImageNet	Small

3. RESEARCH METHODS

The overall research process flow diagram can be seen in Figure 1. The general process is divided into two stages, that is the offline and online processes. The offline processes include data collection, training of the MobileNet transfer learning model and cloud database setup. Meanwhile, the online processes are the mobile application development

comprising user profile, weight management, nutritional facts, and food recognition modules.

3.1 Data Collections

As the images of Malaysian meals were not readily available, secondary data collections were done. A total of 1,000 images were gathered using Google Images and from food image database UEC Food 256, and further labelled into 20 categories. The nutritional information of each food category was also collected from [6] [17-18] and uploaded to a cloud database to be used in calorie counting and weight management module. Examples of the collected food images and nutritional information can be seen in Figure 1.

3.2 Training of MobileNet Network

Out of the 1,000 collected image, 800 images (80%) were then used to train the pretrained Mobilenet network as demonstrated in Figure 2.

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5x Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

Figure 2: Network Architecture of MobileNet

3.3 Cloud Database Setup

Since the food recognition in our work is to be used in a weight management module, the Body Mass Index (BMI) and Basal Metabolic Rate (BMR) of the user need to be calculated. Therefore, the user's profile must include basic physical attributes such as weight, height and age. To minimize storage requirement of the mobile device, we implemented a cloud-hosted database using Firebase database. Firebase was used to manage three functionalities that are user's authentication, nutritional facts of the food, and the food images. In this stage, the Firebase SDK need to be integrated with the mobile environment to manage the database in real-time.

3.4 E-Health Mobile Application

There are four modules in the e-Health application that allowed the users to monitor their calorie intake and assist in managing their weights. All modules are described in the following subsections. The process flow of the e-health application can be seen in Figure 1.

3.4.1 User Profile Module

The function of the user's profile is to register new users where information of the user is obtained to calculate their BMR and BMI. The BMR is used to calculate the user's daily calorie needs, while the BMI is used in the Weight Management module to determine the user's weight category.

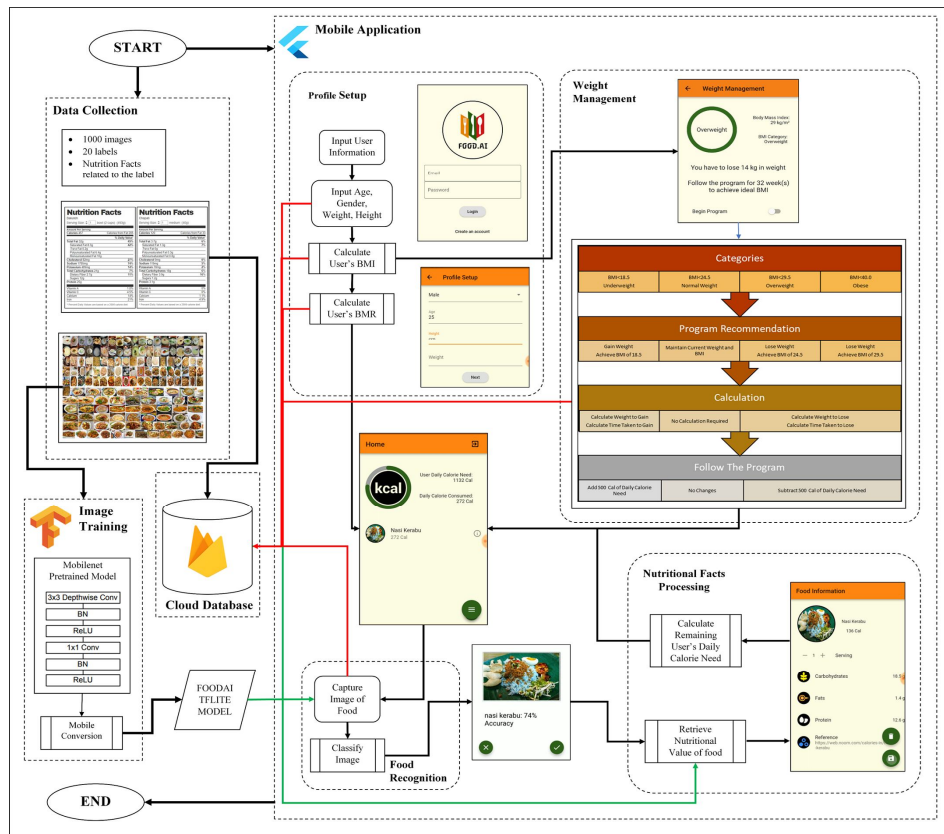


Figure 1: Research Process Flow Diagram

3.4.2 Weight Management Module

In Weight Management module, the user is divided into four categories (i.e. underweight, normal, overweight and obese) depending on the BMI. The category is used to recommend a suitable weight management program (i.e. weight gain, weight lost and weight maintain) based on the user's ideal weight. This module calculated the weight difference and the duration taken to achieve the ideal weight. Equation (1) is used to calculate the weight difference, followed by the duration calculation in (2).

$$\text{Weight Difference}(kg) = \left(\text{Ideal BMI} \left(\frac{kg}{m^2} \right) * \text{User's Height}^2(m^2) \right) - \text{User's Weight}(kg) \quad (1)$$

$$\text{Program Duration}(week(s)) = \frac{\text{Weight Difference}(kg)}{0.45(\text{constant value})(kg/week)} \quad (2)$$

Once the user started the weight management program, the user's daily calorie needs are determined depending on the weight category and the daily calorie consumed. All data are

stored in the cloud database as the user progressed with the weight management program. The interface of this module can be seen in Figure 3.



Figure 3: Weight Management Module Interface

3.4.3 Food Recognition Module

In the Food Recognition module, the user can capture a meal image using a mobile phone camera or select an image from the gallery. The image is fed into the trained MobileNet learning model on the Tensorflow Lite platform for the recognition task. The results of the food recognition are displayed according to the ranking of confidence level. The user is allowed to confirm the correct meal. The selected food image and its corresponding label is then stored in the cloud database. After confirming the selection, the user is directed to the Nutritional Facts Processing module. Figure 4 demonstrates the food recognition results.

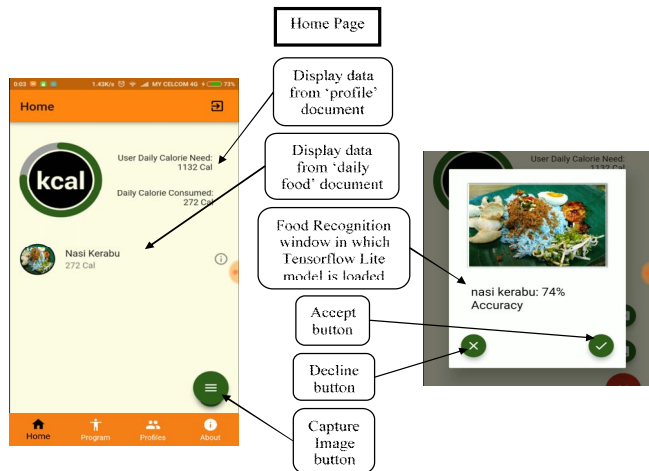


Figure 4: Results of the Food Recognition

3.4.4 Nutritional Facts Processing Module

Nutritional Facts Processing is a module in which the nutritional facts of the recognized food is retrieved from the cloud database and displayed to the user. The user can also specify the serving size of the food that was consumed. The daily calorie needs of the user is updated by subtracting the calorie consumed from the daily allowable calorie intake calculated in the Weight Management module. The nutritional information of *Nasi Kerabu* is presented in Figure 5.

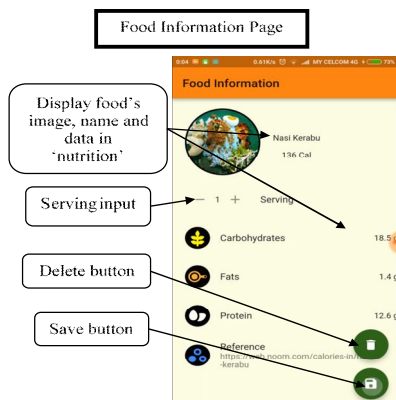


Figure 5: Nutritional Information of the Consumed Food

4. RESULTS AND DISCUSSION

In this section, results of training, validation and testing of the MobileNet transfer learning model are presented.

4.1 Training and Validation of MobileNet network

As mentioned earlier, 80% of the collected images were used for training MobileNet. The other 20% of the dataset is used for validation. Results of the training and validation are presented in Figure 6 and 7. Figure 6 shows the accuracy of MobileNet against the number of epoch, while Figure 7 demonstrates the Cross-Entropy loss of MobileNet. Loss that measures the performance of a classification model and increases as the predicted probability diverges from the actual label.

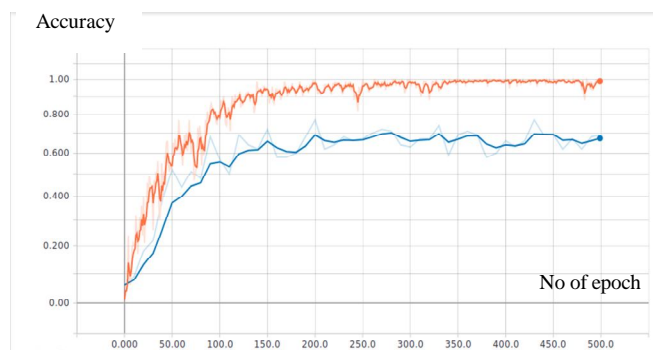


Figure 6: Accuracy of MobileNet Training (Red) and Validation (Blue)

Figure 6 show that the accuracy and performance of the model started poorly with an accuracy below 0.1 for both training and validation. The accuracy improved over time and it reached optimal value during the 200th epoch where the accuracy of the model is at the range of 0.9 for training and 0.7 for validation. Validation produced less accuracy rate as the dataset size is much smaller than training dataset.

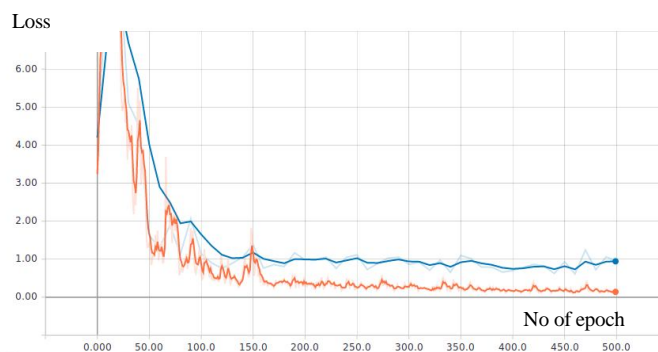






Figure 7: Cross-Entropy Loss of MobileNet Training (Red) and Validation (Blue)

The performance loss in Figure 7, however, shows a loss of more than 3.00 for both training and validation datasets at the beginning. As the epoch increased, the loss increased to over 6.0 and started to reduce at epoch 50 and stabilized after epoch 200 to almost 0 for training and 1 for validation. In the following section, another set of images outside of the training and validation datasets are tested to measure the accuracy of the model in practice.







4.2 Testing the MobileNet network

The food recognition's accuracy is tested using 25 test images.

Twenty images (80%) were correctly labelled, while 5 images were incorrectly labeled. In Figure 8, ten test cases are presented to be discussed.

Test Case 1: Nasi Goreng				
Test Image	Accuracy Test Result: Positive			
	Label	Nasi goreng	Nasi minyak	Nasi ayam
	Confidence (%)	99.70	0.25	0.01
Test Case 2: Murtabak				
Test Image	Accuracy Test Result: Positive			
	Label	Murtabak	Nasi kukus	Satay
	Confidence (%)	92.68	7.96	0.12
Test Case 3: Kway Teow				
Test Image	Accuracy Test Result: Positive			
	Label	Kway teow	Mee goreng	Pasembur
	Confidence (%)	95.84	3.64	1.77
Test Case 4: Nasi Ayam				
Test Image	Accuracy Test Result: Positive			
	Label	Nasi ayam	Roti jala	Mihun

The result is stated as positive if the model correctly labeled the food in the image and negative, otherwise. The label and similarity values depicted the model's prediction of the test image and how certain it is of the label. Test case 1-6 produced positive results. It is due to the distinct features, colour, and texture of the food, both during training phase and testing. The model can correctly recognised the label of food even though the features are scattered throughout the image, or it has a distorted shape. For example, the chicken in test case 4 was very much different in shape compared to the whole chicken of *nasi ayam* in the training datasets. The same scenario applied to test case 5 and 6, where the shape of *roti*

	Confidence (%)	92.09	5.79	0.44
Test Case 5: Roti Naan				
Test Image	Accuracy Test Result: Positive			
	Label	Roti naan	Laksa	Murtabak
	Confidence (%)	0.79	6.47	1.21
Test Case 6: Chapati				
Test Image	Accuracy Test Result: Positive			
	Label	Chapati	Roti canai	Roti naan
	Confidence (%)	87.12	6.18	3.33
Test Case 7: Nasi Lemak				
Test Image	Accuracy Test Result: Negative			
	Label	Nasi kukus	Nasi lemak	Nasi kerabu
	Confidence (%)	91.45	8.30	0.22
Test Case 8: Nasi Kukus				
Test Image	Accuracy Test Result: Positive			
	Label	Nasi lemak	nasi kukus	nasi kerabu
	Confidence (%)	81.21	16.58	1.35
Test Case 9: Mee Kari				
Test Image	Accuracy Test Result: Negative			
	Label	Pasembur	Mee kari	Roti jala
	Confidence (%)	33.16	25.97	13.87
Test Case 10: Laksa				
Test Image	Accuracy Test Result: Positive			
	Label	Laksa	kway teow	Mee kari
	Confidence (%)	99.73	0.24	0.02

naan and *chapati* are irregular in the testing dataset as opposed to the regular round shape in most of the training images.

The complication arisen when two or more labels exhibited nearly similar features, as shown in test case 7 to 10. Most of the training images for *nasi lemak* and *nasi kukus* in test cases 7 and 8 contains similar side dishes, and therefore confused the learning model in testing phase. Thus, both test cases showed high confidence of the different labels. However, in test cases 9 and 10, both *laksa* and *mee kari* have similar features, in term of the sauce's colour and the shape of the noodles, and the model has a higher tendency to label an image as *laksa* if it has similar features.

5. CONCLUSION

A MobileNet transfer learning model was trained using Malaysian meals dataset and a food recognition engine was developed to assist a person in monitoring his/her calorie intake by capturing the food image he/she consumed. It is hoped that the mobile application will be able to encourage more Malaysians to monitor their calorie intake in an easy manner for a healthier lifestyle. There are many improvements to the work presented in this paper. Firstly, the varieties of each food category can be increased for better training of the learning model. As the food recognition module is to be used in a weight management plan, the portion of different types of food consumed and the nutrition information based on the portions should also be considered. The accuracy of the food calorie and user's daily calorie also need to be validated by a dietitian. As of now, the food calorie was obtained from a food database online and the user's daily calorie was calculated mathematically.

REFERENCES

1. Y.Y. Chan, K.K. Lim, K. H. Lim, C.H. Teh, C.C. Kee, S.M. Cheong, and N.A. Ahmad. **Physical activity and overweight/obesity among Malaysian adults**, Findings from the 2015 National Health and morbidity survey (NHMS) [Abstract]. *BMC Public Health*, Vol. 17, No. 1, pp. 733, 2017. <https://doi.org/10.1186/s12889-017-4772-z>
2. S. Ismawati, M. Zainalabidin, and G. Rezai, G. (n.d.). **Healthy eating: The preventive factors among Malaysians**. Retrieved from <http://joebm.com/papers/135-C00007.pdf>
3. C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma. **DeepFood: Deep learning-based food image recognition for computer-aided dietary assessment**. In *Inclusive Smart Cities and Digital Health, Lecture Notes in Computer Science*, 2016, pp. 37-48.
4. S. Mezgec, and B.K. Seljak. **NutriNet: A deep learning food and drink image recognition system for dietary assessment** [Abstract]. *Nutrients*, Vol. 9, No. 7, pp. 657, 2017.
5. **CalorieMama: Instant Food Recognition**. Retrieved from <https://caloriemama.ai/#CalorieMama>
6. **Nutritionix**, A Syndigo Company. Retrieved from <https://www.nutritionix.com/>
7. **Bitesnap, the easier way to track what you eat**. Retrieved from <https://getbitesnap.com/>
8. E. Crookshanks, E. *Development methodologies and SDLC. Practical software development techniques*, 2014, pp. 91-100. https://doi.org/10.1007/978-1-4842-0728-4_6
9. A. Krizhevsky, I. Sutskever, and G.E. Hinton, G. E. **ImageNet classification with deep convolutional neural networks**. *Communications of the ACM*, Vol. 60, No. 6, pp. 84-90, 2017.
10. R. Wu, S. Yan, Q. Dan, and G. Sun. [1501.02876] **Deep image: Scaling up image recognition**. Retrieved October 19, 2018, from <https://arxiv.org/abs/1501.02876>, 2015.
11. C.Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu. **A survey on deep transfer learning**. In *Artificial Neural Networks and Machine Learning – ICANN 2018 Lecture Notes in Computer Science*, 2018, pp. 270-279. https://doi.org/10.1007/978-3-030-01424-7_27
12. I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. Cambridge, MA: The MIT Press, 2017.
13. E.S. Olivas. *Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques*. Hershey, PA: IGI Global (701 E. Chocolate Avenue, Hershey, Pennsylvania, 17033, USA), 2010.
14. H. Shin, H.R. Roth, M. Gao, L. Lu, Z. Xu. I. Nogues, and R.M. Summers. **Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning**. *IEEE Transactions on Medical Imaging*, Vol. 35, No.5, 2016, pp. 1285-1298. <https://doi.org/10.1109/TMI.2016.2528162>
15. A.G. Howard, Z. Menglong, C. Bo, K. Dmitry, W. Wang, T. Weyand, M. Andreetto, and H. Adam. **MobileNets: Efficient convolutional neural networks for mobile vision applications**, Retrieved from <https://arxiv.org/abs/1704.04861>, 2017, April 17.
16. **Noom: Lose weight for good**. Retrieved from www.noom.com
17. **Myfitnesspal: Fitness starts with what you eat**. Retrieved from www.myfitnesspal.com
18. L.Y.H. Teng, K.K. Kuok, M. Imteaz, W.Y. Lai, D.K.X. Ling, **Development of whale optimization neural network for daily water level forecasting**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, No.3, May - June 2019, pp 354-362. <https://doi.org/10.30534/ijatcse/2019/04832019>
19. L. Maddisetti, R.K. Senapati, JVR. Ravindra. Training neural network as approximate 4:2 compressor applying machine learning algorithms for accuracy comparison, *International Journal of Advanced Trends in Computer Science and Engineering* Vol. 8, No.2, pp. 211-215. <https://doi.org/10.30534/ijatcse/2019/17822019>