FLOWER RECOGNITION MODEL BASED ON DEEP NEURAL NETWORK WITH VGG19

(Date received: 13.01.2022/Date accepted: 03.04.2022)

Zi Yuan Ong[1], Kah Kien Chye[1], Huay Wen Kang[1], Chi Wee Tan[1]*

[1] *Faculty of Computing And Information Technology, Tunku Abdul Rahman University College, Kampus Utama, Jalan Genting Kelang, 53300, Wilayah Persekutuan Kuala Lumpur, Malaysia.*

**Corresponding author: chiwee@tarc.edu.my*

ABSTRACT

Computer vision is one way to streamline processes like robotic process automation and digital asset management. It has come a long way in terms of its capabilities and what it can provide and do for different industries. Applications provided by computer vision include object detection and image detection. This field of technology is still relatively young and faces many challenges however. Challenges faced in this field include the lack of comprehensively annotated images to use for training the optimal algorithms, and lack of accuracy for application to real-life images which differ from the training dataset. To tackle these issues, this paper is aiming to adjust pre-trained machine learning models, which are ResNet50 and VGG19 respectively, while also training and tuning a new SqueezeNet inspired model to create a flower recognition model that is able to process and remember large amounts of flower species data. From the research carried out, VGG19 was discovered to have the best performance on both the 5 Categories and Flower-102 dataset, with an accuracy of 88 percent and 84 percent respectively.

Keywords: VGG19, Transfer Learning, Deep Learning, Flower Recognition, Neural Network

1.0 INTRODUCTION

The world has approximately 369,000 named flowering plant species (Liu *et al.*, 2016). In general, experienced plant taxonomists are able to recognize plants based on the flower characteristics such as sepals, petals, stamens, and carpels. However, distinguishing these flowers is difficult for most people. Furthermore, similar flower species can lead to confusion. This is where object recognition comes in, as it is able to understand and analyse images effortlessly and instantaneously (Ong *et al.*, 2021). Therefore, the main objective of this project is to create a flower recognition model that is able to accurately recognise the class of flower in an image. This model is able to analyse the input image and determine the specific type of flower in the image. The end-goal is to train a computer to do what comes naturally to humans, which is to understand what is included in an image and gain insight from it. This project is interested in both modifying existing pre-trained machine learning models and training a flower recognition model based upon SqueezeNet. It is critical to both pursue novel and innovative methods, and also to conduct thorough research on existing methods in order to gain new insights and make new discoveries.

2.0 LITERATURE REVIEW

Lv *et al.* (2021) proposed a model to do flower classification with saliency detection and VGG-16 deep neural network model that is trained on the Oxford Flower-102 data set. An optimization algorithm of stochastic gradient descent was done to reduce computing consumption and training time to improve the model. In order to reduce model overfitting, a dropout method was used by randomly removing training information. To deal with the issue of insufficient image data and also to reduce model training time, transfer learning methods were used and an accuracy of 91.9 percent was achieved, which demonstrates better results than other conventional methods for image classification tasks and proves the possibility of flower identification using this model.

Cibuk *et al.* (2019) used pre-trained DCNN models, AlexNet and VGG-16, for feature extraction and concatenated features from both models to construct efficient feature sets. For the feature selection algorithm, the minimum Redundancy Maximum Relevance (mRMR) model was implement. The extracted features were then used to classify the flower species using a support vector machine (SVM) classifier with a Radial Basis Function (RBF) kernel. Their experimental results showed that they were able to achieve an accuracy performance of 96.39 percent and 95.70 percent for the Flower-17 and Flower-102 datasets, respectively.

Feng, Wang, Zha and Cao (2019) proposed the approach of using transfer learning and Adam deep learning optimization algorithm to fix the defects of current mainstream CNN, which are deep depths, long parameters, long training time and slow convergence. A modified and supplemented VGG-16 model was used, and the transfer learning method and Adam optimization algorithm are used to accelerate the network convergence. They used partial sets of images of the Flower-102 dataset combined with the Flower-17 dataset to form 30 sets of images, which are then randomly divided with Stratified Shuffle Split. With this they were able to obtain a 98.99 percent accuracy on their test set, while maintaining fast convergence.

Liu *et al.* (2016) proposed a flower classification approach which uses a convolutional neural network to extract features. They have also obtained the luminance map which is created by converting RGB pixels to YUV, and the brightness of the colour is extracted from the Y component, which allows better performance as flowers have high brightness. They compute a bottom-up saliency map using a regional contrast-based salient object detection algorithm, which simultaneously evaluates global contrast difference and spatial weighted coherence scores. The algorithm is simple, efficient, and naturally multi-scale, and it generates full-resolution, high-quality saliency maps, which improves performance. They achieved an accuracy of 76.54 percent in their dataset and 84.02 percent in the Oxford Flower-102 dataset.

2.1 Existing Method

SqueezeNet is a novel convolutional neural network notable for having 112 times fewer parameters than another CNN, Alexnet, while also being able to maintain an accuracy top-5 performance comparable to that of AlexNet (Iandola *et al.*, 2016). Because it is such a small mode, SqueezeNet is more suitable for on-chip implementations on FPGAs (Iandola *et al.*, 2016). The SqueezeNet model has been studied in various use cases, and the results have been promising. Sayed, Soliman, and Hassanien (2021) used a SqueezeNet model optimised with a bald eagle search (BES) optimization to find the best hyperparameter to predict melanoma skin cancer on ISIC 2020 and ISIC 2019. The proposed melanoma skin cancer prediction model obtained an overall accuracy of 98.37 percent, specificity of 96.47 percent, sensitivity of 100 percent, f-score of 98.40 percent, and area under the curve of 99 percent. The results showed the robustness and efficiency of the proposed model compared with VGG-19, GoogleNet, and ResNet50. Therefore, a SqueezeNet-inspired model was chosen to be explored.

ResNet, which is an abbreviation for Residual Networks, enables engineers to train hundreds or even thousands of layers while still achieving impressive results (He *et al.*, 2016). The model managed to win the ImageNet challenge in 2015. It has been discovered that increasing training error in deep neural networks is caused by the network's initialization, optimization function, or one of the most well-known problems, the vanishing gradient problem (He *et al.*, 2016). It is an issue that occurs during the training of artificial neural networks with gradient-based learning and backpropagation. Gradients are known and can be used to update the weights in a network during backpropagation. However, the gradient can become increasingly small at times, thereby functionally preventing the weights from changing values. Because the same values are propagated over and over again, the network stops training, resulting in no useful work being done. To solve such problems, residual neural networks are used. ResNet alleviates this vanishing gradient problem through employing skip connections, which works by adding the output of an earlier layer to a later layer (He *et al.*, 2016).

VGG is a novel object-recognition model with support for up to 19 layers (Simoyan and Zisserman, 2015). It is pre-trained with ImageNet datasets and is still able to outperform with other unseen datasets, which makes it one of the most used image recognition architectures. The VGGNet has several variants, like the VGG-16 and VGG-19 variants, which differ only in the total number of layers in the neural network. Several studies have been conducted using the VGG-19 model, with impressive results. Victor Ikechukwu *et al.* (2021) conducted research using ResNet-50, ChexNet, VGG-19, and their own proposed Iyke-Net models to identify pneumonia from chest x-ray images, where VGG-19 achieved a high accuracy of 93.5 percent, coming in close second after their proposed Iyke-Net which is 93.6 percent accurate.

3.0 METHODOLOGY

This section explains and elaborates on the datasets used and the design of the models.

3.1 Datasets

The first dataset is the Kaggle flower recognition dataset, which includes 4242 images from Flickr, Google Images, and Yandex Images (Mamaev, 2021). Daisy, tulip, rose, sunflower, and dandelion images are divided into five categories. Each class has about 800 images, with each image measuring about 320x240 pixels. The photos are not reduced to a single size, but rather come in a variety of sizes. It is henceforth referred to as the 5-category dataset.

The Oxford Flower-102 dataset, which consists of 102 flower categories, is also employed (Nilsback & Zisserman, 2008). This dataset is even more specific than the 5-category dataset in that the category is based on flower species specified in their scientific name, instead of a general flower category like “daisy”. The images depict flowers that are common in the United Kingdom. Each class contains between 40 and 258 images, with varying scales, poses, and lighting. The difficulty of classification is exacerbated by large variations within the same category and several very similar categories. There are 8189 images in the dataset. This dataset is henceforth referred to as the 102-category dataset.

3.2 Model Architecture

The first model developed is based upon the original SqueezeNet model (Iandola *et al.*, 2016). It utilises the Fire module architecture as suggested by the original developers, consisting of a squeeze convolution layer of only 1x1 filters, which feeds into an expand layer that has a mix of 1x1 and 3x3 convolution filters. The abundant use of 1x1 filters greatly reduces the parameters by 9 times compared to using 3x3 filters. The parameters in the layer are further reduced by using the squeeze layers, which reduces the number of input channels to 3x3 filters. Therefore, SqueezeNet is capable of achieving a size more than 50 times smaller than the AlexNet model while still achieving a reasonable accuracy. This particular implementation is a stripped down version of SqueezeNet with fewer layers which consists of one input into a conv2d layer, followed by batch normalisation, the first fire module, the first MaxPooling2D layer, the second fire module, the second MaxPooling2D layer, the third fire module, the first GlobalAveragePooling2D layer and the final Dense layer with softmax activation to obtain the categories to be predicted.

With comparisons to different ResNet variants like ResNet18, ResNet34, ResNet101 and ResNet152, ResNet50 is chosen for transfer learning because of its lower requirement of computational power and encouraging accuracy (He *et al.*, 2016). Firstly, the pre-trained ResNet50 model (resnet50 weights tf dim ordering tf kernels.h5) is downloaded from Github (Fchollet, 2016) and which uses the weights already pre-trained from the imagenet datasets. The first layer of ResNet50 is frozen and made untrainable to avoid taking a long time during the training process by having lesser trainable parameters. Extra dense layers shall also be added to the final layers of the pre-trained network, which allows learning the combinations of the previously learned features which are useful in recognising new objects in the new dataset. Therefore, additional layers were implemented after the output of the ResNet50 model. A total number of 7 layers were added in this project including the flatten layer, batch normalization layers, two customised layers, a Rectified Linear Unit (ReLU) activation function layer and the softmax layer. The flatten layer converts data into a 1-dimensional array for input to the next layer, and batch normalisation is a layer that allows each layer of the network to learn more independently. Finally, a softmax layer is included as the output layer in order to predict fixed types of flowers whereby 5 or 102 classes of classification are produced, depending on whether it is for the 5-category or 102-category dataset. The 5 classes are created in order to predict the 5 types of flowers for the 5-category dataset, which are daisy, tulip, rose, sunflower and dandelion. So, after applying this layer, a transfer learning model that can classify the input images into various types of flowers based on predictions from the pre-trained ResNet50 model is developed.

VGG-19 is composed of 16 convolutional layers, 5 pooling layers, 3 fully-connected layers and a final layer of softmax function (Simonyan & Zisserman, 2015). The matrix was shaped (224, 224, 3) as the fixed input size of 224x224 RGB image is passed into this network. Small kernel size of 3x3 with stride of 1 pixel is used to cover every part of the image rather than using a large kernel size, whereas a 2x2 pixel window with a stride of 2 pixels is used to perform max pooling. The multiple layers of small kernels are able to effectively cover the images without the use of large kernels such as 11x11 kernel in AlexNet and 7x7 kernel in ZFNet. Therefore, the number of parameters and the overfitting problem is mitigated. All hidden layers are equipped with ReLU which uses tanh or sigmoid functions to introduce non-linearity for better classification compared to the previous models. In this project, the pre-trained weights are used by setting the parameter weights to the one trained with imagenet. The first 19 layers are frozen to prevent the weights from being modified and similarly to the aforementioned ResNet50 model, additional layers are added after the pre-trained model. A max-pool layer is added to downsample the input features. A flattening layer is then added before a dense layer with a softmax function since the dense layer accepts 2D input.

3.3 Logical Flow

Data Loading: The flowers dataset consists of images of flowers with different class labels and stored in respective directories. For the 5-category dataset, there are 4242 images and 5 class labels. For the 102-category dataset, there are 8189 images and 102 class labels. This stage loads the data from their directories and concatenates it into a single dataframe. As a result, the flower images end up with an image dimension of 244 x 244, which minimises image dimensions while maintaining image readability with efficient computational complexity and accommodating the input shape for the pre-trained models.

Data Understanding: The flowers dataset contains examples of labelled flower images. Each example includes a JPEG flower image as well as the class label. The exploration of the image data helps to authenticate the class distribution of each type of flower in the dataset and ensures a balanced dataset, in

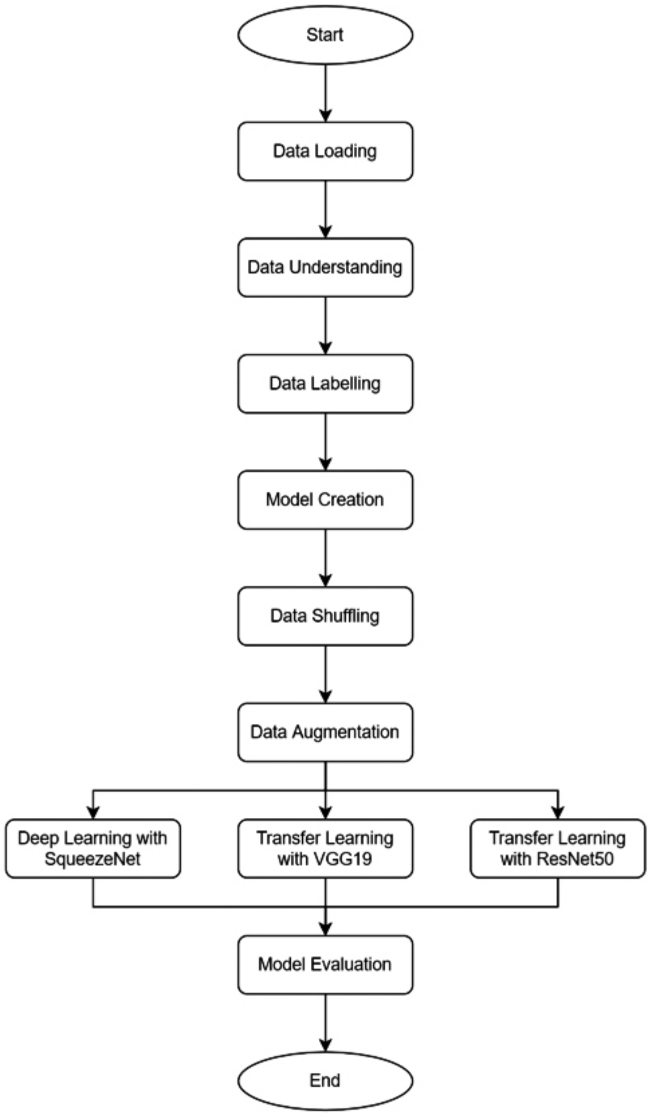


Figure 1: Logical Flowchart

order to an imbalanced dataset that can lead to poor predictive power. Data visualization is needed to sample and study the input data to ensure image readability by randomly viewing the 10 images in the dataframe via 2D representation with the selected new image dimensions.

Data Labelling: This stage is used to transform categorical data (textual data) into numerical values for the prediction functions so that the deep learning predictive models can understand the input data. Label encoding technique is performed to convert categorical values to numbers in this step.

Model Creation: The same optimizer and loss function is used to compile all models. Cross-entropy is the loss function chosen to evaluate a set of weights in multi-class classification problems for flower recognition. Furthermore, the Adam optimizer with the learning rate of 0.01 is used to search through different weights for the network. Finally, because this is a classification problem, classification accuracy is collected and reported, which will be defined via the metrics argument.

Data Shuffling: This stage is used to redistribute the training and testing data samples in the dataset to ensure that each data sample produces an "independent" change on the model, without being influenced by the points that came before it. Data shuffling is required since the images were added sequentially from subfolders into the dataframe during data loading Otherwise, the model can only learn what is "daisy" from the first 800 images, which does not optimise the model's parameters. The seed is set to 100 and is applied to both images and labels to ensure that each image matches the correct label.

Data Augmentation: This stage is used to increase the amount of data by adding slightly changed copies of already existing data or newly created artificial data from existing data. This refers to randomly changing the images in ways that shouldn't impact their interpretation, such as horizontal flipping, zooming, and rotating. Through data augmentation, this stage regularises the images and prevents overfitting problems when training a machine learning model. This technique is used to overcome the problem of overfitting by creating more data and making the model generalise well on the unseen data.

Model Training: The model is using the predefined train-test split for the Oxford Flower-102 dataset, with 1020 training samples and 6149 testing samples. The 5-category dataset is split into train-test sets using a ratio of 80:20. Different batch sizes and epochs are adjusted in different models in order to achieve the optimal result. This stage generates tensor image data in batches, which will be looped over for both training and testing. The neural network in each model takes in inputs, which are then processed in hidden layers using weights that are adjusted during training. Then the model spits out a prediction. The weights are adjusted to find patterns in order to make better predictions.

Model Evaluation: In this process, validation accuracy measures were calculated after the model had gone through all the data. The network had been fully trained when these scores were calculated. The final model selection is ultimately based on validation accuracy.

4.0 RESULTS AND DISCUSSION

This section shows the results obtained for each model, on both datasets. The metric used is the validation accuracy, because metrics such as training accuracy do not accurately reflect real-world performance. The validation accuracy is calculated by taking the True Prediction/Total Number of Predictions using the validation dataset. The models were validated with N = 1020 samples for the Oxford Flower-102 dataset and N= 848 samples for the 5-category dataset.

Table 1: Validation accuracy for each model for the two datasets

	Validation Accuracy (5 Category)	Validation Accuracy (102 Category)	Status
SqueezeNet Inspired Model	77%	67%	Rejected
ResNet50	67%	42%	Rejected
VGG19	88%	84%	Accepted

According to our findings, the partially pre-trained VGG-19 model performed the best on both datasets, achieving a high validation accuracy of 88 percent on the 5 Categories dataset and 84 percent on the Flower-102 dataset. This could be due to the VGG-19's simple structures and hidden layers with ReLU functions, which can better introduce non-linearity for better classification than other models. The model's generalisation improves as the number of features produced decreases.

The SqueezeNet Inspired model had the second-highest validation accuracy of 77 percent and 67 percent on the 5 Categories and Flower-102 dataset respectively. This might be due to the fact that while this SqueezeNet Inspired model is not quite as complex and has less layers and performance than both the other models, this model is not pre-trained. Instead, the whole model was trained solely relying on the two datasets individually for each scenario, and therefore was able to perform better than the partially pre-trained ResNet50. This is a lightweight model, even when compared to the already lightweight original SqueezeNet implementation, and therefore it should be expected performance was sacrificed for it's small size.

Finally, the ResNet50 model performs the worst, with 67 percent and 42 percent on the 5 Categories and Flower-102 datasets, respectively. This could be because the architecture of the ResNet50 model is overly complicated for this task, resulting in poor generalisation. The optimal learning rate, batch size, and identification of the best freezing layer all play important roles in the model's performance.

5.0 CONCLUSION

The objectives to create a flower recognition model that is able to accurately recognise the class of flower in an image have been met because a model based on VGG19 has been developed that can perform flower classification tasks with 88 percent accuracy with 5 flower categories and 84 percent accuracy with 102 flower species. However, there are some limitations in this research, such as hardware limitations. This project's algorithms and models are all computationally intensive due to the complexity of the models and the size of the dataset. This project is also more concerned with which models perform better on the two flower datasets, rather than why the models perform the way they do. Other pre-trained models such as AlexNet, VGG16 and so on, which were not used in this project, are another potential avenue for investigation. More comprehensive studies on other models can be conducted in the future.

6.0 ACKNOWLEDGMENTS

Authors thank the Faculty of Computing and Information Technology, Tunku Abdul Rahman University College for financial support and resources to carry out this project. ■

REFERENCES

[1] Cıbuk, M., Budak, U., Guo, Y., Cevdet Ince, M. and Sengur, A., (2019) 'Efficient deep features selections and classification for flower species recognition'. Measurement, 137, pp.7-13.

[2] Fchollet, (2016). Release VGG16, VGG19, and ResNet50 · fchollet/deep-learning-models., GitHub. Available at: https://

github.com/fchollet/deep-learning-models/releases/tag/v0.1 (Accessed: 10 September 2021)

[3] Feng, J., Wang, Z., Zha, M. and Cao, X. (2019) 'Flower Recognition Based on Transfer Learning and Adam Deep Learning Optimization Algorithm'. Proceedings of the 2019 International Conference on Robotics, Intelligent Control and Artificial Intelligence - RICAI 2019.

[4] He, K., Zhang, X., Ren, S., Sun, J. (2015) 'Deep Residual Learning for Image Recognition' 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778

[5] Iandola, F.N., Moskewicz, M.W., Ashraf, K., Han, S., Dally, W.J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size. ArXiv, abs/1602.07360.

[6] Liu, Y., Tang, F., Zhou, D., Meng, Y. and Dong, W. (2016) 'Flower classification via convolutional neural network'. 2016 IEEE International Conference on Functional-Structural Plant Growth Modeling, Simulation, Visualization and Applications (FSPMA), Qingdao, China, pp. 110-116.

[7] Lv, R., Li, Z., Zuo, J., Liu, J. (2021) 'Flower Classification and Recognition Based on Significance Test and Transfer Learning'. 2021 IEEE International Conference on Consumer Electronics and Computer Engineering, ICCECE 2021. Institute of Electrical and Electronics Engineers Inc., pp. 649–652.

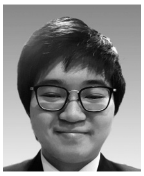
[8] Mamaev, A. (2021). Flower Recognition, Kaggle [Online]. Available at: https://www.kaggle.com/alxmamaev/flowers-recognition (Accessed: 19 April 2021)

[9] Nilsback, M., Zisserman, A. (2008). 102 Category Flower Dataset, Visual Geometry Group [Online]. Available at: https://www.robots.ox.ac.uk/~vgg/data/flowers/102/ (Accessed: 19 April 2021)

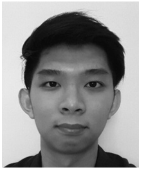
[10] Simonyan, K., Zisserman, A. (2015) 'Very deep convolutional networks for large-scale image recognition', 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, International Conference on Learning Representations, ICLR, San Diego, California, United States of America.

[11] Ong, ZY., Chye, KK., Kang, HW., Tan, CW. (2021) 'A Flower Recognition System using Deep Neural Network Coupled with Visual Geometry Group 19 Architecture', International Conference On Digital Transformation And Applications 2021, ICDXA 2021, pp. 121-128 Kuala Lumpur, Malaysia.

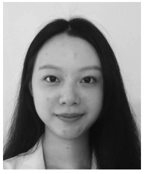
PROFILES



ONG ZI YUAN received his Bachelor's degree of Computer Science (Honours) in Data Science from Tunku Abdul Rahman University College (TAR UC) in 2022. He is a fresh graduate who is passionate about Artificial Intelligence (MI), Machine Learning (ML) and Data Science. His research interests include artificial intelligence, natural language processing, image processing, data analytics as well as computer vision.
Email address: ongzy-wp17@student.tarc.edu.my



CHYE KAH KIEN graduated from Tunku Abdul Rahman University College (TAR UC) in 2022 with a BCompSc (Hons) (Computer Science in Data Science). He is a recent graduate who is passionate about applying data science and machine learning skillsets to SAP systems, an enterprise software to manage business operations and customer relations in which 77% of the world's transaction revenue touches an SAP system. His research interests include natural language processing, image processing, data mining, process mining, as well as advanced analytics.
Email address: chyekk-wp17@student.tarc.edu.my



KANG HUAY WEN received BCompSc (Hons) Computer Science in Data Science from Tunku Abdul Rahman University College (TAR UC) in 2022. She is passionate about Artificial Intelligence (AI) and Data Science and aspires to be a professional data scientist. Her research interests involved Data Analytics, Code Mixed Sentiment Analysis and Computer Vision.
Email address: kanghw-wp17@student.tarc.edu.my

DR TAN CHI WEE received BCompSc(Hons) and PhD degrees in year 2013 and 2019 respectively in Universiti Teknologi Malaysia. Currently, he is a Senior Lecturer cum Programme Leader at Tunku Abdul Rahman University College and actively involved in the Centre of Excellence for Big Data and Artificial Intelligent (CoE) and become the research group leader for Audio, Image and Video Analytics Group under Centre for Data Science and Analytics (CDSA). Dr Tan's main research areas are Computer Vision (CV), Image Processing (IP) and Natural Language Processing (NLP) and Artificial Intelligence (AI). He is an enthusiastic researcher experienced in conducting and supporting research into Image Processing. Being a meticulous and analytical researcher with Train-The-Trainer certificate of many years of educational and hands-on experience, he was invited to Université d'Artois (France) under Marie Skłodowska-Curie Research and Innovation Staff Exchange (RISE) programme for collaborative research between European countries with Southeast Asian countries on motion detection and computer vision and being involved in industry project as professional consultant.
Email address: chiwee@tarc.edu.my