Plant Recognition System Using Convolutional Neural Network

Wan Muhammad Arief Hilman Bin Wan Zefri 1, Sharifalillah Nordin2

Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Shah Alam
40450 Shah Alam, Malaysia

1ariefhilman08@gmail.com
2sharifa@tmsk.uitm.edu.my

Abstract. There are many plant species available to man, even some that have not been discovered. There are species of plants are reported to be on the verge of extinction. Due to the inhumane nature of humans, forest is cleared to make space for industrial purposes. These purposes will lead to the destruction of nature, including plants. Plants are important to be preserved for the future. Plants should be made aware to the public to avoid such disaster. Ensuring they know the plants species are the reasons for the development of this system. The system will ensure people can identify plants without having flipping through research papers or books. Those plants that are similar to one another can be recognized easily. Using convolutional neural network to recognize plant can help the public gain more awareness towards plants species. Using the traditional convolutional neural network for the training of model to recognize plants. The system was tested to get the best accuracy to recognize plants. The images were increased to ensure a better accuracy. This system requires a good model in order to recognize the correct plants. Convolutional neural network is commonly used for image classification due to its high accuracy. The accuracy for the system created in this project is 78.85%. This project can be improved in the future by increasing the accuracy. There are a lot of ways to increase the accuracy such as trying out other CNN architectures or by feeding the system more images. This project involves 7 phases which is preliminary study, knowledge gathering and acquisition, knowledge representation, system design, system development, system testing and evaluation, and documentation.

1. Introduction

There have been reports of problems when conserving plants. One being that the plants are not identified. Without knowing the plants name and species, they cannot be part of a conservation plan. Another problem would be that names can be changed and multiply, creating confusion for researchers and managers in identifying a plant. Another problem is that there is no single collection of information about all the world’s plants which makes it harder to be able to identify the plant. The name of each plants matters because the names should match the properties of the organisms. Identifying plants only by using the naked eye may cause the plants to have the wrong name and description. Using traditional method by searching the plant everywhere to confirm its species may take time and effort because the taxonomic information of plants is scattered throughout a vast collection of journals, theses, popular articles, books, and electronic sources. Some of these can be difficult to access and it may be hard to identify these plants even using these methods [1].
The system should be able to identify the plants and provide the functions or specialty that the plant has. Some plants are used for medical purposes and some plants are poisonous to even be touched. This information is what the system will be providing. By using the system, users can understand and appreciate the importance of plants for humans, animals, and the environment. They may be more tempted to protect the plants from any danger that may occur such as deforestation. By having awareness, we can manage plants effectively. The data used for this system are 10 plants collected from Belum, Temenggor. The 10 plants collected are from different family. The following plants and its family are shown in table 1.

| No | Plant                                                | Family          |
|----|------------------------------------------------------|-----------------|
| 1  | Leea Rubra Blume Ex Spreng.                         | Vitaceae        |
| 2  | Heliotropium Indicum L.                              | Borahinaceae    |
| 3  | Alternanthera Sessilis (L.) R.Br. Ex DC.             | Amaranthaceae   |
| 4  | Bulbophyllum Medusae (Lindl.)                        | Orchidaceae     |
| 5  | Rafflesia Cantleyi Solms                            | Rafflesiaceae   |
| 6  | Thunbergia Grandiflora (Roxb. Ex Rottler) Roxb.      | Acanthaceae     |
| 7  | Crinum Asiaticum L.                                 | Amaryllidaceae  |
| 8  | Zingiber Spectabile Griff.                           | Zingiberaceae   |
| 9  | Centella Asiatica (L.) Urb.                          | Apiaceae        |

2. Literature Review

2.1. Plant Classification
Plants classification that uses the leaves usually have taken descriptions from botanist that has described the different features including the shape, texture, and veins. Plant identification is usually by observing the morphological characteristics of the plant such as the general character, structures of the stem, roots and leaves, embryology, and flowers. This is followed by the consultation of a guide or a known database. Most of its important information about taxonomic identity is contained in its leaves of a plant. The leaves are permanent on a plant whereas the flowers are only temporary, they may only last a few weeks. This is the reason why most plant identification tools that are based on Content-Based Image Retrieval techniques mainly work with leaf images. Colour, texture, and shape are features that can characterized a leaf. Leaf may have more than one colour due different climates and season [2]. Plant classification is important because it can greatly speed up the process of identifying plant species, collecting, and monitoring. Botanists can have the world’s herbaria right at their fingertips. There are tools that can be used to make the botanical information from the world’s herbaria accessible to anyone. It can be used anywhere, in a remote jungle or rainforest to plants that grow in the city or park [3].

2.2. Convolutional Neural Network
Convolutional neural network is widely used for image classification [4]. It has had ground-breaking results over the past decade from image recognition to voice recognition [5][6][7]. Convolutional Neural Network were used in projects such as diagnosis of cancer using histopathological images and traffic sign classification [8]. There was a research where AlexNet architecture used CNN for classification purpose and produced good results. It was observed that the images are classified correctly even for the portion of the test images, and this shows the effectiveness of deep leaning
algorithm [9]. CNNs are expected to achieve significantly better results than standard feed-forward networks for many tasks due to the fact that they impose appropriate constraints on how the function mapping is learnt [10]. CNNs give the best performance in solving pattern or image problems and can even outperform humans in certain cases [11]. Convolutional neural network process images as tensors which are frameworks of number with extra dimensions. Shown below is a $2 \times 3 \times 2$ tensor. A CNN usually takes 3 tensors, which is a higher-order matrix, as its input. This usually includes an image with $H$ rows, $W$ columns, and 3 channels. These 3 channels are R, G, B colour channels. Higher order tensor inputs can be handled by CNN in the same way. The input goes through several processes in sequence. One processing step is normally called a layer, which can be a convolution layer, a pooling layer, a normalization layer, a fully connected layer, a loss layer and other several types of layers [12]. The key layers in a CNN model are convolution layer, pooling layer, and ReLU [13].

![Tensor Representation](image)

**Figure 1.** Representation of a tensor

### 2.3. Related Works

**Plant Leaf Recognition Using a Convolution Neural Network** [14] is proposed to improve classification performance by using a CNN that can extracts and learn feature points. This is an improvement for a previous work which used a shape-based search method to distinguish between different plants instead of using traditional machine learning methods, which classify data after extracting features and pre-processing. The system proposed in this paper will be studying a method for learning and recognizing types of leaves using CNN model, a deep learning technique. This is to improve the classification performance.

**Android Based Smart Phone Data Image Recognition** [15] is used to recognize cursive handwriting. The cursive handwriting is extracted using a mobile phone’s camera feature and the image is then processed by a pre-processing technique which is segmented based on the stroke of the letters. The handwriting is recognized with the help of Support vector machine (SVM). Printed cursive writing can also be used on the app. The main advantage of this app is to turn text into digital text. This can make any work more efficient because they do not require the need to turn to the script repeatedly. This is also a less hassle way where it does not require much space to save these digital texts.

**Herbal Plant Recognition Using Deep Convolutional Neural Network** [16] used deep Convolutional Neural Network (CNN) for herbal plant recognition using leaf identification. The old method of plant identification consumes time because of the varieties as well as similarities that are within the plant species. This study shows that using multiple parameters can create and enhance a deep CNN model to boost recognition accuracy performance. Another purpose of this study is to show the significant effects of the multi-layer on sample that are small to achieve reasonable performance. There are methods that can provide a better significant benefit on the overall performance which is data augmentation. Resize, flip, and rotate are simple augmentation that will increase accuracy significantly. This is by creating invariance and preventing the model from learning irrelevant features.
This study constructed a new dataset of leaves from herbal plants that are found in Malaysia and the results from the experiment achieved 99% accuracy.

The research paper Deep Plant: Plant Identification with Convolutional Neural Network [17] studies CNN to learn unsupervised feature representations for 44 different plant species that is collected from Royal Botanic Gardens, Kew, England. Deconvolutional networks is the technique in which the CNN model is based on to gain intuition on the chosen features of the plants. Representing each plant, it is uniquely chosen from venations of different order. CNN features with different classifiers produces results that are much better compared to state-of-the-art solutions that are reliant on hand-crafted features.

3. Methodology

3.1. Data Collection

The collection of data was done in processes where list of plants from Belum, Temenggor was used to identify the plants needed to be collected for the system. The images were collected from google images where images of each plant were hand-picked for the system. The raw images collected from google images was in different pixels and sizes. The images collected will be transformed to suit the system. Table 2 shows the examples of image of each plant collected.

| No | Plant | Example of Image |
|----|-------|-----------------|
| 1  | Leea Rubra Blume Ex Spreng. | ![Leea Rubra Blume Ex Spreng.](image1) |
| 2  | Heliotropium Indicum L. | ![Heliotropium Indicum L.](image2) |
| 3  | Alternanthera Sessilis (L.) R.Br. Ex DC. | ![Alternanthera Sessilis (L.) R.Br. Ex DC.](image3) |
| 4  | Bulbophyllum Medusae (Lindl.) Rchb.F. | ![Bulbophyllum Medusae (Lindl.) Rchb.F.](image4) |
| No | Plant                                      | Example of Image                                      |
|----|-------------------------------------------|-------------------------------------------------------|
| 5  | Piper Umbellatum L.                       | ![Image of Piper Umbellatum L.](image1.png)          |
| 6  | Rafflesia Cantleyi Solms                 | ![Image of Rafflesia Cantleyi Solms](image2.png)     |
| 7  | Thunbergia Grandiflora (Roxb. Ex Rottler) Roxb. | ![Image of Thunbergia Grandiflora](image3.png)       |
| 8  | Crinum Asiaticum L.                      | ![Image of Crinum Asiaticum](image4.png)             |
| 9  | Zingiber Spectabile Griff.               | ![Image of Zingiber Spectabile Griff.](image5.png)   |
| 10 | -Centella Asiatica (L.) Urb.             | ![Image of Centella Asiatica](image6.png)            |
Table 3. Number of images collected.

| No | Scientific Name                                           | Number of Images Collected |
|----|-----------------------------------------------------------|----------------------------|
|    |                                                           | Train | Test | Total |
| 1  | Leea rubra Blume ex Spreng.                              | 69    | 13   | 82    |
| 2  | Heliotropium indicum L.                                  | 80    | 10   | 90    |
| 3  | Alternanthera sessilis (L.) R.Br. ex DC.                 | 40    | 10   | 50    |
| 4  | Bulbophyllum medusae (Lindl.) Rchb.f.                    | 46    | 10   | 56    |
| 5  | Rafflesia cantleyi Solms                                 | 54    | 10   | 64    |
| 6  | Alternanthera sessilis (L.) R.Br. ex DC.                 | 60    | 10   | 70    |
| 7  | Thunbergia grandiflora (Roxb. ex Rottler) Roxb.          | 40    | 10   | 50    |
| 8  | Crinum asiaticum L.                                      | 51    | 10   | 61    |
| 9  | Zingiber spectabile Griff.                               | 100   | 13   | 113   |
|    | Total                                                    | 678   |      |       |

3.2. Image Pre-processing
Images collected was resized to ensure that it is the same pixels and can ensure the efficiency of the system. This enables the system to understand the data without any errors. Therefore, pre-processing is required to be done to all data collected. An example of image that has been pre-processed is as below. After resizing, we can see that the image has been added padding on the top and bottom. This is done to ensure the image stays intact even after resizing. The image will then be processed easily during training of the model.

Figure 2. Image after pre-processing

3.3. Data Augmentation
Data augmentation is to increase the number of images to get a more accurate result. It is a technique to artificially create a new set of data from existing data. Image data augmentation includes various range of operations to manipulate the image such as shifts, zooms, flips and much more. CNN learns features that are invariant to their location in the image and data augmentation can further aid CNN in this invariant approach and also aids the model in learning features that are invariant.
3.4. System Design
System design consist of three parts which is the system architecture, system flowchart and user interface.

3.4.1. System Architecture. The first step in the system architecture will be the image acquisition where the images are collected and stored for the use of the system. The next step is image pre-processing where the images is resized to the same height and weight. The next step would be data augmentation where the images are increased using data augmentation method.

![System Architecture Diagram]

Figure 3. System Architecture

3.4.2. System Flowchart. The first step in system flow is uploading an image to be recognize by the system. The image uploaded is then classified based on the model trained. The plant details are the output of the classification.

3.4.3. User Interface Design. The user interface is user friendly even to those who are a novice to technology. The name of the system is shown on the top of the user interface. The upload an image button is placed below the title and the uploaded image is shown below the button after uploading. The classify button will appear after uploading an image and when it is clicked, the output will be shown below the image. The output includes the name of plant, its family and a short description of the plant.
3.5. Testing and Evaluation

Some text. Testing was done using test dataset. The details related to the image path and their respective class labels is in a csv file called Test2.csv. The extraction of image path and labels was done using pandas. Then to predict the model, images was resized to $30 \times 30$ pixels and numpy array was created containing all image data. From the sklearn.metrics, we imported the accuracy_score and observed how our model predicted the actual labels. The model was saved using the Keras model.save() function.

![Figure 4. Plant recognition GUI](image)

![Figure 5. Graph Output](image)
4. Analysis and Discussions

4.1. Data Augmentation Results

The number of images before and after data augmentation has been recorded to ensure maximum
number of images. Table 4 shows the number of images before and after flipping and mirroring using
ImageOps module. The highest being Zingiber spectabile Griff. and lowest being Centella asiatica (L.)
Urb. because of the lack of images collected. Table 4 shows images before and after using
ImageDataGenerator. The range of the images are around 300 to 400 images which increases the
images by 3 to 4 times the normal amount. The table also shows images before and after using
ImageOps and ImageDataGenerator. This is done to maximize the images for training. The images can
be combined because ImageDataGenerator does not produce images that are flipped or mirrored. It is
randomly generated by the module.

Table 4. Images before and after flipping and mirroring using ImageOps module.

| No. | Scientific Name                           | Number of Images (Train) |
|-----|-------------------------------------------|--------------------------|
|     |                                           | Before | After flipping and mirroring using ImageOps module | After using ImageDataGenerator | After using ImageOps and ImageDataGenerator |
| 1   | Leea rubra Blume ex Spreng.               | 69     | 207                                      | 357                           | 495                           |
| 2   | Heliotropium indicum L.                   | 80     | 240                                      | 407                           | 567                           |
| 3   | Alternanthera sessilis (L.) R.Br, ex DC.  | 40     | 120                                      | 313                           | 393                           |
| 4   | Bulbophyllum medusae (Lindl.) Rchb.f.    | 46     | 138                                      | 347                           | 439                           |
| 5   | Piper umbellatum L.                       | 54     | 162                                      | 330                           | 438                           |
| 6   | Rafflesia cantleyi Solms                  | 60     | 180                                      | 341                           | 461                           |
| 7   | Thunbergia grandiflora (Roxb. ex Rottler) Rxb. | 40     | 120                                      | 278                           | 358                           |
| 8   | Crinum asiaticum L.                       | 51     | 153                                      | 387                           | 489                           |
| 9   | Zingiber spectabile Griff.                | 100    | 300                                      | 394                           | 594                           |
| 10  | Centella asiatica (L.) Urb.               | 34     | 102                                      | 272                           | 340                           |

Example of the images can be seen in table 5. The table presents i) the images without
augmentation i.e., the images that has been pre-processed to the same height and width by inserting
padding; ii) flip images using ImageOps i.e., the images that has been flip 180 degrees; iii) mirror
using ImageOps which created the images that are mirrored; iv) ImageDataGenerator produces images
that are randomly augmented. The spaces are white in color to avoid any noise and unwanted image
being a part of recognizing the plant. Other choices that can be made was making the parts fill with
grey color, images near to it or mirroring the image. These are choices which was not chosen as it
would cause difficulties in recognizing the plant properly.
Table 5. Example of image after data augmentation.

- i) Without augmentation
- ii) Flip using ImageOps module
- iii) Mirror using ImageOps module
- iv) Using ImageDataGenerator

4.2. System Testing Results

The accuracy was tested using different data augmentation methods, ImageOps and ImageDataGenerator. In Table 6, the accuracy was tested without any data augmentation. The result produced was low and it reached the lowest for 400 epoch which was 50 percent. In the table, data augmentation was done using ImageOps, which flips and mirrors the images. This has resulted in higher result compared to without using any data augmentation, but the results were not far. The accuracy ranges from 50 to 60 percent. This led to trying out other method, which was ImageDataGenerator. This method produced images which were randomly augmented to increase the number of images for training. The test using ImageDataGenerator was recorded, and it is higher compared to both results using ImageDataGenerator and using ImageOps and ImageDataGenerator. Another test was done which was combining both data augmentation method, ImageOps and ImageDataGenerator. This has produced the best result out of the methods before. The accuracy reached 78% when using this method compared to the other methods which barely reached 70% (as shown in Table 6).

The epoch was the parameter that was changed in the testing because the number of epochs will decide how many times the weights of the network will change. As the number of epochs increases, the same number of times weights are changed in the neural network and the boundary goes from underfitting to optimal to overfitting. Both underfitting and overfitting should be avoided. The number of epoch should be set as high as possible and the training should be terminated when validation error starts increasing. The number of epoch is tested to produce the best accuracy.
**Table 6. Accuracy without data augmentation.**

| Epoch | Accuracy without data augmentation (%) | Accuracy with data augmentation using ImageOps | Accuracy with data augmentation using ImageDataGenerator | Accuracy with data augmentation using ImageOps and ImageDataGenerator |
|-------|----------------------------------------|-----------------------------------------------|--------------------------------------------------------|---------------------------------------------------------------|
| 50    | 63.46                                  | 55.77                                         | 65.38                                                  | 71.15                                                         |
| 100   | 56.73                                  | 54.81                                         | 65.38                                                  | 66.35                                                         |
| 150   | 43.27                                  | 61.54                                         | 60.58                                                  | 69.23                                                         |
| 200   | 62.50                                  | 58.65                                         | 70.19                                                  | 69.23                                                         |
| 300   | 57.69                                  | 61.54                                         | 67.31                                                  | 75.00                                                         |
| 400   | 50.00                                  | 55.77                                         | 64.42                                                  | 78.85                                                         |
| 500   | 44.23                                  | 50.96                                         | 67.30                                                  | 74.04                                                         |
| With  | 60.58                                  | 62.50                                         | 63.46                                                  | 73.08                                                         |
|       | EarlyStopping                          |                                               |                                                        |                                                               |

**Acknowledgement**

This work is supported by UiTM Internal Research Grants Scheme (LESTARI) (600-RMC/LESTARI SDG-T 5/3 (146/2016)) from the Research Management Center (RMC), Universiti Teknologi MARA, Shah Alam, Selangor. The appreciation also goes to the Faculty of Computer and Mathematical Sciences (FSKM), Universiti Teknologi MARA, Shah Alam, Selangor for giving a moral support in the production of this paper.

**References**

[1] Roux, M. Le. (2018). Why plants need an identity. The Conversation. [https://theconversation.com/why-plants-need-an-identity-96737](https://theconversation.com/why-plants-need-an-identity-96737)

[2] Sharma, S., Gupta, C., Gu, X., Du, J., Wang, X., & Reserved, I.-A. R. (2015). A Review of Plant Recognition Methods and Algorithms. International Journal of Innovative Research in Advanced Engineering, 2(6).

[3] Belhumeur, P. N., Chen, D., Feiner, S., Jacobs, D. W., Kress, W. J., Ling, H., Lopez, I., Ramamooorthi, R., Sheorey, S., White, S., & Zhang, L. (2008). Searching the world’s Herbaria: A system for visual identification of plant species. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5305 LNCS(PART 4), 116–129. [https://doi.org/10.1007/978-3-540-88693-8-9](https://doi.org/10.1007/978-3-540-88693-8-9)

[4] Suo, J., Wu, T., Zhu, S., Shan, S., Chen, X., & Gao, W. (2008). Design sparse features for age estimation using hierarchical face model. 2008 8th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2008. [https://doi.org/10.1109/AFGR.2008.4813314](https://doi.org/10.1109/AFGR.2008.4813314)

[5] Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2018). Understanding of a convolutional neural network. Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017, 2018-Janua(August), 1–6. [https://doi.org/10.1109/ICEngTechnol_2017.8308186](https://doi.org/10.1109/ICEngTechnol_2017.8308186)

[6] Sharifuddin, M. S. I., Nordin, S., and Ali, A. M., Voice control intelligent wheelchair movement using CNNs. Proceedings of 2019 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS 2019), pp. 40–43, 2019, [https://doi.org/10.1109/AiDAS47888.2019.8970865](https://doi.org/10.1109/AiDAS47888.2019.8970865)

[7] Sharifuddin, M. S. I., Nordin, S., and Ali, A. M., Comparison of CNNs and SVM for voice control wheelchair, IAES International Journal of Artificial Intelligence, vol. 9, no. 3, pp. 387–393, 2020, [https://doi: 10.11591/ijai.v9.i3.pp387-393](https://doi: 10.11591/ijai.v9.i3.pp387-393).
[8] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53(8), 5455–5516. https://doi.org/10.1007/s10462-020-09825-6

[9] Krishna, M. M., Neelima, M., Harshali, M., & Rao, M. V. G. (2019). Image classification using deep learning. Journal of International Pharmaceutical Research, 46(3), 745–749. https://doi.org/10.14419/jiij.v712.7.10892

[10] Browne, M., & Ghidary, S. S. (2003). Convolutional neural networks for image processing: An application in robot vision. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2903(May), 641–652. https://doi.org/10.1007/978-3-540-24581-0_55

[11] Hijazi, S., Kumar, R., & Rowen, C. (n.d.). What Is a CNN? Using Convolutional Neural Networks for Image Recognition. 1–12.

[12] Wu, J. (2017). Convolutional Neural Networks. Medical Imaging and Information Sciences, 34(2), 109–111. https://doi.org/10.11318/mii.34.109

[13] Haridas, R., & R L, J. (2019). Convolutional Neural Networks: A Comprehensive Survey. International Journal of Applied Engineering Research, 14(3), 780. https://doi.org/10.37622/ijaer/14.3.2019.780-789

[14] Jeon, W. S., & Rhee, S. Y. (2017). Plant leaf recognition using a convolution neural network. International Journal of Fuzzy Logic and Intelligent Systems, 17(1), 26–34. https://doi.org/10.5391/IJFIS.2017.17.1.26

[15] Kundavinachiyar, G., & Muthumani, I. (2014). Android based Smart Phone Data Image Recognition. 5(1), 849–851. https://www.semanticscholar.org/paper/Android-based-Smart-Phone-Data-Image-Recognition-Kundavinachiyar-Muthumani/debed88fc215397784cbf5e7294a70644c74b4a4

[16] Zin, I. A. M., Ibrahim, Z., Isa, D., Aliman, S., Sabri, N., & Mangshor, N. N. A. (2020). Herbal plant recognition using deep convolutional neural network. Bulletin of Electrical Engineering and Informatics, 9(5), 2198–2205. https://doi.org/10.11591/eei.v9i5.2250

[17] Lee, S. H., Chan, C. S., Wilkin, P., & Remagnino, P. (2015). Deep-plant: Plant identification with convolutional neural networks. Proceedings - International Conference on Image Processing, ICIP, 2015-Decem, 452–456. https://doi.org/10.1109/ICIP.2015.7350839