**CLASSIFICATION OF MEDICINAL PLANT SPECIES THROUGH**

**DEEP LEARNING USING CONVOLUTIONAL**

**NEURAL NETWORK**

**JOHN ROMMEL BADUEL DE ROBLES**

**BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING**

**(Major in Computer Engineering)**

**JUNE 2018**

**CLASSIFICATION OF MEDICINAL PLANT SPECIES THROUGH**

**DEEP LEARNING USING CONVOLUTIONAL**

**NEURAL NETWORK**

**JOHN ROMMEL BADUEL DE ROBLES**

**SUBMITTED TO THE FACULTY OF THE**

**DEPARTMENT OF ELECTRICAL ENGINEERING**

**COLLEGE OF ENGINEERING AND AGRO-INDUSTRIAL TECHNOLOGY**

**UNIVERSITY OF THE PHILIPPINES LOS BAÑOS**

**IN PARTIAL FULFILLMENT OF THE**

**REQUIREMENTS FOR THE**

**DEGREE OF**

**BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING**

**(Major in Computer Engineering)**

**JUNE 2018**

1. **INTRODUCTION**
   1. **Background of the Study**

Medicinal plants are without a doubt, useful at the minimum if not vital to people’s health and welfare. To put in perspective their usage, it has been cited that globally, roughly 11% of flowering plants are used for such purpose (Abe & Ohtani, 2013). With the technological advancements in the field of medicine, the utilization of these plants are continually and further enhanced of their traditional uses long since ancestral times.

Of varying effects and conditions that could be alleviated or remedied, it is crucial to distinguish among the medicinal plants which is which. With an android phone and an internet connection, a reverse image search could be done through Google, Bing, or any other available search engine. From an online database, the image would be recognized, and then classified correspondingly to what it is perceived to be. However, this may come problematic in view of the stated means. A reliable reception is necessary for the internet connectivity with the need of accessible internet itself. Furthermore, the accuracy and completeness of the accessed data may not align with the locally-specific plant in question.

Image recognition, and relatedly image classification, had their development significantly heightened with the reintroduction of deep learning through convolutional neural networks (CNNs). Deep learning, a branch of machine learning, is the field in which features extracted from data originally hand-engineered (in machine learning) are instead inherently learned by the model without any intermediate interference from the data (Lecun, Bengio, & Hinton, 2015). It is labeled as ‘deep’ because of the multiple hidden layers between the input and the output that map these two to each other. And so, with the exhibited feature extraction, image classification, for instance, becomes achievable to the deep learning-based system.

The architecture of deep learning responsible for image-related examinations are CNNs. They operate with filters that have weights, which dictate what features to extract. These weights, originally arbitrary in value, are adjusted through what is termed as training, where data is fed to the CNN model for its independent evaluations. Having fine-tuned the weights, the first set of filters of a model locates basic patterns or low-level features from an input image. The filters activate areas of the input where the features are recognized, and the data is outputted as activation or feature maps. Consequently, these maps serve as the input to the next set of filters, which then considers higher level of features. This hierarchical amplifying feature extraction executes repeatedly until the output is reached, where the classification or recognition is concluded. (Lecun et al., 2015)

Deep learning has existed since the 1950s, only that it was addressed differently. Back then, it was called cybernetics. And eventually, it turned to connectionism. The field stayed inactive and forgotten, until the processing machines, together with the abstractions, became available. However, a trend would not transpire without having its pioneers. In the midst of the revolution of image classification stands Krizhevsky et al. (2012) with their 2012 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) CNN-based entry, winning them the competition against other networks by a large margin. ILSVRC is an annually held object recognition competition which uses a subset of the ImageNet dataset amounting to millions of images, where it is established to report the top-1 and top-5 error rates. These error rates are the percentages in which the right image labels are not within the top labels (i.e. not the one label considered for top-1 and not among the five labels considered for top-5) evaluated by the network. Their CNN model achieved a top-5 error rate of 16.4% compared to the second best performing network with 26.2%, paving the way to substantially progressing computer vision. (Krizhevsky, Sutskever, & Geoffrey E., 2012; Goodfellow et al., 2016)

Different configurations of deep CNNs have been employed for various fields and applications since then, with increasing accuracy and complexity. With that, it goes without saying that plant classification CNNs are already existing and performing well for their subjects of interest. Even so, these CNNs are particularly limited to the plants they are tasked to classify. Hence, in the study, a deep CNN would be employed to categorize specific medicinal plants, making use of limited data present in the environment.

* 1. **Significance of the Study**

It is challenging for someone not versed in the classification of plant species to tell apart which is which upon encountering them. With the increasing reliance to and availability of mobile phones, developing a mobile application which could classify plants, in this case medicinal ones, not only could aid in distinguishing them from each other, but could also serve as an educational material for those interested in the area. Moreover, the study could serve as a stepping stone to further develop and increase the extent of predictive ability of models with larger datasets and larger classifications to consider.

* 1. **Objectives of the Study**

**General Objectives**

The study generally aims to develop a mobile application for the classification of Caesalpinia pulcherrima (Bulaklak ng paraiso), Cassia alata (Acapulco), Blumea balsamifera (Sambong), and Mentha arvensis (Herba buena) plant species.

**Specific Objectives**

More specifically, this study aims to:

1. employ a CNN architecture that could classify images based on the specified categories with limited training data; and
2. evaluate results in terms of testing accuracy and F-measure.
   1. **Scope and Limitations of the Study**

The study only addresses the plant classification of the involved medicinal plants existing within the vicinity of the University of the Philippines Los Baños (UPLB) campus, using a customized architecture of CNN applicable to a limited amount of training data. That is, plant species not included in the training data, or differing features of the plant species existing within other regions could not be classified at all or would lead to an erroneous prediction from the model. As for the data, at least 100 plant samples per species would be considered, in which 80%, 10%, and the remaining 10% would be respectively used for training, tuning, and testing.

* 1. **Time and Place of the Study**

The study will be conducted from August 2018 to December 2018 at the computer laboratory of the Department of Electrical Engineering, College of Engineering and Agro-Industrial Technology, University of the Philippines Los Baños, Laguna.

1. **REVIEW OF LITERATURE**

Plant classification, more specifically of medicinal plants, should be addressed as it is crucial in the implementation of technologies that concern them. Of the means to do so, artificial intelligence has advanced immensely and developed deep learning, which provided CNN models to be utilized for image classification. Consequently, this entailed a variety of particular uses of CNNs, with one of which involving plants.

* 1. **Medicinal Plants in the Philippines**

The adoption of the National Drug Policy in the Philippines has led to the significant spur of attention and reliance to plants in their use as medicinal treatment. Traditional healers, who were one of the first to promote medicinal plants, reported the use of the following to address various health concerns: decoction of the leaves of Blumea balsamifera (Sambong) for cough treatment, poultice of the leaves of Cassia alata (Acapulco) for skin diseases, and decoction of the leaves of Mentha arvensis (Herba buena) for stomach ache. In addition, of the ten scientifically-proven medicinal plants endorsed by the Department of Health (DOH) for any health concerns, two of the mentioned ones are included in them. These two are Blumea balsamifera, promoted as a diuretic, and Cassia alata, recommended for skin fungal infection and scabies. The former also prosper in the manufacture of its products, which further poses its effectivity and economic relevance in the nation. As Western medicine dominates the medical field, the importance and accessibility of herbal drugs most especially in rural areas should not be abandoned. Rather, it should be advocated and studied so as to complement at least the Western medical system. (Tomlinson & Akerele, 2015)

* 1. **Evolution of Artificial Intelligence**

**Machine Learning**

In order to associate a set of objects to their designated classifications or detections, say, translate an English sentence to its French counterpart, or classify an image as to whether it is a dog or a cat, an intermediary must be considered. This mediator, in the area of artificial intelligence, is where machine learning algorithms come into play. Simply put, features are extracted from the objects of interest, the inputs, to let the machine learn patterns from them. This is done by having adjustable arbitrarily set weights for the feature extractors. And so, the process of learning through establishing the correct weight values, also known as training, enables the machine to differentiate and classify through the motifs (Lecun et al., 2015). Accordingly, these features serve as representations of the data in which the performance of the machine substantially depend on (Goodfellow et al., 2016).

**Representation Learning**

A major problem of the conventional machine learning systems, however, is their need for considerable human intervention, as well as the expertise, in choosing what features to extract (LeCun et al., 2015). That is, feature extraction is designed by the human operator himself. What comes as problematic in this case, is that with the increasing complexity of systems, it becomes an extremely complicated task to manually design such feature models that would generate optimal solutions. And so, to deal with this obstruction, people have thought of having the machines learn these features by themselves, through what is known as representation learning.

Still, this approach falls short in handling necessary abstract features for more complex tasks. With varying and accumulating influences in the composition of the data (e.g. orientation and illumination in a picture of a dog), especially unobservable factors, the difficulty in data representation approaches the difficulty in solving the problem itself (Goodfellow et al., 2016). For this reason, simple representation learning becomes ineffective.

**Deep Learning**

Addressing this issue, a discipline under machine learning called deep learning was developed. It is a robust form of representation learning in which a series of hierarchical mapping of representations are executed. To put into more detail, this field starts with the recognition of features from the input. These simple features would then form the perceived inputs in the next layer of feature detectors, with the process of using preceding outputs as succeeding inputs done repetitively. Upon reaching the set output, higher and more abstract level of representations are achieved. This sequence of data representations is what made it possible to accurately solve complicated and more complex real-life problems. Needless to say, this is why it is termed deep. With this, the potential for advancement in different areas in the application of AI was extended. (Goodfellow et al., 2016; LeCun et al., 2015)

Being experimented on since the 1950s, referred to as cybernetics then connectionism, it is only now that deep learning is reaping its fruits for the following reasons: increase in available datasets for training, increase in processing and memory capacities of GPUs and CPUs, reduction in costs of computing hardware, and improvement in the algorithms (Goodfellow et al., 2016; Guo et al., 2016).

* 1. **Convolutional Neural Networks (CNNs)**

**Fundamentals of CNN**

CNN is a deep learning architecture which makes use of a number of layers to perform a hierarchical feature extraction to a set of data that are generally images (Lecun et al., 2015). Essentially, these layers are composed of convolutional layers, pooling layers, and fully connected layers, which are trained through tuning the arbitrarily set parameters of these layers by being fed with (usually labeled) data.

To put into more detail, convolutional layers are the basic blocks of CNNs which utilize feature filters, also termed as kernels, to output feature maps through discrete convolution (Guo et al., 2016; Lecun et al., 2015). The network comprises these layers being the first ones, each or some of which are oftentimes followed by pooling layers mainly for dimension reduction and translation invariant feature maps, and then fully connected layers that transform the feature maps to vectors and can link these vectors to categories for an image classification, for instance (Guo et al., 2016). Incidentally, for handling large datasets, non-saturating linearities are applied to the convolutional and fully connected layers for faster learning, leading to better performance of the architecture (Krizhevsky et al., 2012). Meanwhile, to prevent overfitting, dropouts and a number of data augmentation techniques could also be implemented. To define, overfitting is the erroneous generalization of the trained model that makes it recognize only the data it was fed.

**CNN in Computer Vision**

Of the areas in which deep learning is applied, computer vision stands to be the most prominent of them all. This is after amassing the interest of the populace in an outcome of the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). ILSVRC is an annually held competition which started in 2010, aimed to realize algorithms or models of the lowest error rates in image recognition. In the contest, it is customary to report the top-5 and top-1 error rates, which are the percentages in missing the right label of the image in the considered top labels of the algorithm. During the ILSVRC-2012, Krizhevsky et al. toppled other models and achieved a 9.8% top-5 error rate disparity with the next best performing model. They did so by employing CNNs and notably, this had served to be the baseline in generating better structures of the network. (Krizhevsky et al., 2012)

**Development of CNN Architectures**

Having that settled, it is essential then to take note of the progressing and significant modifications to the structures of the existing state-of-the-art CNNs. Getting back to the first ILSVRC CNN-based entry and winner, the network of Krizhevsky et al. (2012) operated with five convolutional layers and three fully connected layers. As they have remarked, removing a single convolutional layer would be vital to the performance of the network. They have also made use of the nonlinearity called rectified linear units (ReLUs), max pooling for the pooling layers, and image translations and horizontal reflections for data augmentation.

Speaking of which, it is important to note that they have the model undergo supervised learning. This is a type of learning where the images labeled with their right classifications are fed to the model, so that the model would learn and accomplish these correspondence in its generalizations. Of the networks to be considered, only those that have gone through supervised learning would be evaluated, and the pre-trained ones, those with layer parameters already adjusted beforehand, would be excluded to examine the networks fully from scratch.

With that, the first documented modification of the mentioned structure was done by Zeiler et al. (2014). Using their own formulated Deconvolutional Networks (deconvnets) to visualize which features stimulate each layer of the network, they were able to pinpoint and better the inadequacy of the previous work. By changing the filter sizes in the first convolutional layer to 7 by 7 (from 11 by 11), the stride, the amount of shift of the filters, in the first convolutional layer to 2 (from 4), and the size of the third to fifth convolutional layers (increased to 512, 1024, & 512, respectively), they were able to decrease the top-5 error rate by 1.7%. They also experimented on and observed the change in performance of the model upon removing or adjusting specific layers, in which it was perceived to have an increased error rate resulting from the removal of one or some convolutional layers and/or the increase in size of the fully connected layers.

Another remarkable concept introduced in the area of CNN is the Inception model. Realized by Szegedy et al. (2015), this idea gave them the win in the ILSVRC 2014 with their 22-layered “GoogLeNet” architecture. Notably, they have made the network not only deeper, but also wider, in the sense that convolutional layers with varying filter sizes are applied in parallel and operated supplementarily with 1x1 convolutional layers for dimension and computational cost reduction. They employed paralleled 1x1, 3x3, and 5x5 convolutional layers, and pooling layers, with the 3x3 and 5x5 layers preceded by 1x1 layers, to reduce the computational cost from the convolutions with their inputs. This constitutes the Inception modules, and they reasoned that this arrangement makes computational complexity of each stage of the network controllable even with a desired extensive increase in units, and that it follows the rationale of having visual information of varying degrees process simultaneously. Despite having fewer parameters (i.e. 12 times fewer) than the architecture by Krizhevsky et al., they achieved top-5 error of 6.67% where their more exhaustive data augmentation (i.e. cropping approach) have played a part as well. It is also noteworthy that their implementation is 3 to 10 times faster than their non-Inception counterparts. Moreover, they have emphasized that the use of Inception modules are only advantageous at higher layers.

Of the stated architectures developed and implemented, it is crucial to take into account the datasets they used. All that were mentioned are entries of the ILSVRC, and this competition makes use of a subset of the ImageNet dataset with about 1000 images for each of its 1000 categories, amounting to millions of data. Hence, if these networks are to be employed as is to an area with limited training data, intuitively, it is possible that the networks would overfit and fail to function accordingly, mainly due to the huge amount of parameters they consider.

**Performance Evaluation of CNN Architectures**

It is common to make use of a typical accuracy measurement to assess CNN architectures applied to various fields. Given an object, the value can be calculated by obtaining the percentage of its properly classified images over the quantity of its total testing images. This is an acceptable and usual practice, and would be observed in the examined literatures to follow. However, when working with the classification of several objects, it is more complete to implement an additional metric. This metric is called the F-measure, which considers and maximizes both precision and recall. Precision is the percentage of the correct instances over the total predictions in a class, while recall is the percentage of the correct instances over a class’ actual total samples (POWERS, 2011). These quantities range from 0 to 100 when expressed as percentages, with 100 being the ideal and desired value.

The precision, recall, and F-measure of a network are determined easier with the help of a confusion matrix. A confusion matrix is a table which positions the quantity of the predictions (columns) against their actual classifications (rows). It establishes the true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP and TN are the correctly classified cases, while FN and FP are those that are not. Equation 2-1, Equation 2-2, and Equation 2-3 present the formulas for the measurements.

|  |  |  |
| --- | --- | --- |
|  |  | *(Equation 2-1)* |
|  |  | *(Equation 2-2)* |
|  |  | *(Equation 2-3)* |

* 1. **Plant-related CNN Applications**

With a reasonable background on the general existing architectures of CNNs, it is paramount, in this case, to be mindful of those specifically aimed at plants. That being said, in 2016, Sladojevic et al. (2016) made use of the CaffeNet architecture in their plant disease recognition through leaf image classification. This is an architecture strictly based off of the first CNN model having five convolutional and three fully connected layers, making use of the ImageNet dataset for parameter adjustments. Through transfer learning – that is, retaining the learned parameter weights and biases, they have modified the network by changing the last layer to fit to the number of classes they require, and further fine-tuned the layer parameters with the use of their dataset. This works because the earlier layers of the network generates low-level features only and are not object-specific, to say the least. Additionally, they applied affine transformation, perspective transformation, and simple image rotations for data augmentation. Due to the substantial difference in dataset size of theirs and the original application of the architecture (i.e. thousands vs. millions), they reduced the initial learning rate of the layers. With that, they achieved an overall accuracy of 96.3%.

On a related note, in terms of plant classification, Yalcin et al. (2016) dealt with the same reference architecture as well, only adjusting the filter sizes and parameters of the layers and classifying 16 plant species on their network. Working with 4800 images and considering the different growth stages of each plant, they achieved a 97.47% accuracy.

Meanwhile, Dyrmann et al. (2016) devised their own model having seven convolutional layers and two fully-connected layers. Their first convolutional layer had weights set from the VGG16 network which was also trained on the ImageNet dataset. With their data amounting to only tens of thousands, they designed their network to consist of just about 1.2 million learnable parameters, compared to the 60 million-parameter network of the standard five convolutional, three fully-connected layer CNN. They utilized 50% dropouts before the fully-connected layers, and batch normalization, ReLUs, and max pooling after the convolutional layers. Also, they applied excessive green segmentation to prevent biases of the network to species with specific backgrounds. They acquired an 86.2% accuracy with their 22 category CNN model, and claimed that the low amount of training data caused the poor individual classification accuracies of some of the plant species.

And so, as the last to mention, Andrea et al. (2017) compared four existing and accessible architectures of CNN (i.e. LeNet, AlexNet, cNET, & sNET) and have chosen to use the one with the highest accuracy and lowest loss. They have realized this to be cNET which has 2 convolutional layers and three fully-connected layers, attaining 96.4% accuracy on their dataset. They have segmented and masked the training data for feature detection and made geometrical transformations of 30-degree differences for data augmentation. To be able to be used in an embedded system, they have reduced the filters from 64 to 16 to enhance the processing speed while still taking into account the accuracy of the network.

1. **MATERIALS AND METHODS**

This study will employ a CNN architecture in classifying medicinal plants, and will evaluate the performance of the network after using it. The data specifics, as well as the architecture and evaluation details are discussed as follows.

* 1. **Data Acquisition, Partition, and Augmentation**

An android phone will be used to acquire pictures of the selected plant species in their natural environment. Of the 4 types chosen, 100 samples will be considered for each. For every one of these samples, four vertically-tilted overhead orientations at most (i.e. North, South, East, and West) will be taken and captured, with a corresponding top view whenever possible. The varying orientations will be helpful in making the network less dependent to a specific background, as well as perceive other plant angles, in order to generate predictions more reliably by alleviating the overfitting problem. Accordingly, as long as the distinctive features of the plants are present and perceivable, that is, as long as the leaves and flowers (if there are any) could be seen, the plant images will be included in the dataset. With that, approximately 2000 images can be accumulated.

Now, isolating for the meantime the taken top view photographs, the remaining 400 images per plant species will be divided and used as follows: 250 for training, 50 for cross-validation, and 100 for testing. Before being used for training though, the taken pictures will be rescaled to 256x256 to reduce computational cost and training time. Then, of the top view pictures which consist of 100 per plant species, 70 will go to the training dataset, 10 will go to cross-validation, and 20 will go to testing. As implemented by Dyrmann et al. (2016), to further increase the training and validation data, the photographed top views will be rotated with 90° increments then mirrored. This augmentation of data multiplies it eight times, and also contributes to make the network somehow rotation invariant.

* 1. **Proposed CNN Architecture**

It is vital to consider an architecture that is not as intricate as the one designed in GoogLeNet (Szegedy et al., 2014) to prevent overfitting when working with limited training data. On the mentioned researches focused on agriculture, comprising of only thousands of datasets (compared to millions in the ILSVRC), the CNN architectures are basic stacks of convolutional and fully-connected layers which achieved satisfactory, if not exceptional outcomes. Among these, it could be observed that the earliest structure of five convolutional, three fully-connected network was implemented on some of them (Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, 2016; Yalcin & Razavi, 2016). And so, taking into account the simplicity and reliability of the performance, this will be the one employed in this research.

The overall succession of layers can be seen in Table 3-1. Filter sizes and strides, as well as the order and construction of the proposed network, are mainly influenced and sourced from the quintessential architecture by Krizhevsky et al. (2012) with implementation of some of the modifications by Zeiler et al. (2014). Pooling layers considered are the typical max pooling, which simply select the largest values within every examined region of the layers they are applied to. Accordingly, these layers function by introducing translation invariance (Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, 2016; Yalcin & Razavi, 2016) and spatial reduction to the network, which aid in the betterment of the network’s performance and moderation of computational load. In addition, they are set to be overlapping pooling where the different scanned sections have mutual areas, which is reported to further increase the classification accuracy (Krizhevsky et al., 2012). Moreover, making use of rectified linear units (ReLUs) allow the network to learn faster, while batch normalizations (BN) notably reduce the necessary training iterations (Dyrmann, Karstoft, & Midtiby, 2016; Krizhevsky et al., 2012). As for the last fully-connected (FC) layer, a 4-way softmax function will be worked with to output the probabilities of the four classes given an image for testing.

Table 3-1. Proposed CNN architecture.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| LAYER | NUMBER OF FILTERS | FILTER SIZE | STRIDE | PADDING | OUTPUT SIZE |
| Input |  |  |  |  | 256x256x3 |
| Convolutional 1 | 96 | 7x7x3 | 2 | No | 125x125x96 |
| ReLU 1 |  |  |  |  | 125x125x96 |
| BN 1 |  |  |  |  | 125x125x96 |
| Pooling 1 |  | 3x3 | 2 |  | 62x62x96 |
| Convolutional 2 | 256 | 5x5x96 | 2 | No | 29x29x256 |
| ReLU 2 |  |  |  |  | 29x29x256 |
| BN 2 |  |  |  |  | 29x29x256 |
| Pooling 2 |  | 3x3 | 2 |  | 14x14x256 |
| Convolutional 3 | 384 | 3x3x256 | 1 | Yes | 14x14x384 |
| ReLU 3 |  |  |  |  | 14x14x384 |
| Convolutional 4 | 384 | 3x3x384 | 1 | Yes | 14x14x384 |
| ReLU 4 |  |  |  |  | 14x14x384 |
| Convolutional 5 | 256 | 3x3x384 | 1 | Yes | 14x14x256 |
| ReLU 5 |  |  |  |  | 14x14x256 |
| Pooling 3 |  | 3x3 | 2 |  | 6x6x256 |
| FC 1 |  | 6x6x4096 |  |  | 1x1x4096 |
| ReLU 6 |  |  |  |  | 1x1x4096 |
| FC 2 |  | 1x1x4096 |  |  | 1x1x4096 |
| ReLU 7 |  |  |  |  | 1x1x4096 |
| FC3 |  | 1x1x4096 |  |  | 1x1x2 |
| Softmax |  |  |  |  | 1x1x2 |

Other details concerning the training of the network is summarized in Table 3-2. Stochastic Gradient Descent (SGD), the method to be used in training the network, works by adjusting layer weights upon computing gradients of some set of the examples, whose endpoint is minimizing the average of the error function (i.e. objective function) which evaluates the difference between the desired and the actual performance of the model (Lecun et al., 2015). The initial weights will be randomized from a zero-mean Gaussian distribution with 0.01 standard deviation.

Table 3-2. Learning details of the network.

|  |  |
| --- | --- |
| LEARNING DETAILS | VALUE/TYPE |
| Training Method | SGD |
| Batch Size | 128 |
| Momentum | 0.9 |
| Weight Decay | 0.0005 |
| Initial Neuron Biases  (of 2nd, 4th, 5th conv, & FC layers) | 1 |
| Initial Neuron Biases  (of remaining layers) | 0 |
| Initial Learning Rate | 0.01 |
| Epochs | 1000 |

* 1. **Evaluation of Results**

With the testing data, Equation 2-1, Equation 2-2, and Equation 2-3 will be used to calculate the magnitudes per class of the precision, recall, and F-measure, respectively. The overall accuracy of the network will be calculated by obtaining the percentage of the correctly classified images over the quantity of the total testing images. To make sense of the reason as to why these statistical values are acquired as they are, activation maps of the convolutional layers will be visualized and observed. Additionally, the plot of the training process (epochs vs. validation loss and training loss) will be recorded to ascertain the right number of iterations (epochs) for the network. This is the instant the gap between the training and the validation loss is minimum (Dyrmann et al., 2016).

1. **REFERENCES**

Abe, R., & Ohtani, K. (2013). An ethnobotanical study of medicinal plants and traditional therapies on Batan Island, the Philippines. *Journal of Ethnopharmacology*, *145*(2), 554–565. https://doi.org/10.1016/j.jep.2012.11.029

Andrea, C. C., Daniel, B. B. M., & Misael, J. B. J. (2017, October). Precise weed and maize classification through convolutional neuronal networks. In *Ecuador Technical Chapters Meeting (ETCM), 2017 IEEE* (pp. 1-6). IEEE.

Dyrmann, M., Karstoft, H., & Midtiby, H. S. (2016). Plant species classification using deep convolutional neural network. *Biosystems Engineering*, *151*(2005), 72–80. https://doi.org/10.1016/j.biosystemseng.2016.08.024

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning* (Vol. 1). Cambridge: MIT press.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, *187*, 27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Krizhevsky, A., Sutskever, I., & Geoffrey E., H. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25 (NIPS)*, 1–9.

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

POWERS, D. M. W. (2011). Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, *2*(1), 37–63. https://doi.org/10.1.1.214.9232

Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, D. S. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, *2016*. Retrieved from http://dxdoi.org/10.1155/2016/3289801

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … Rabinovich, A. (2014). Going Deeper with Convolutions. *arXiv:1409.4842*. https://doi.org/10.1109/CVPR.2015.7298594

Tomlinson, T. R., & Akerele, O. (Eds.). (2015). *Medicinal plants: their role in health and biodiversity*. University of Pennsylvania press.

Yalcin, H., & Razavi, S. (2016). Plant classification using convolutional neural networks. *2016 Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, 1–5. https://doi.org/10.1109/Agro-Geoinformatics.2016.7577698

Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Thesis Gantt Chart** | | | | | | | | | | | | | | | | | | | | | |
|  |  |  |  | Plan Duration | | | | | *Starting from August 2018* | | | | | |  |  |  |  |  |  | |
| **ACTIVITY** | **PERIODS (IN WEEKS)** | | | | | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** | **21** |
| **Acquisition of Plant Images** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Comp. Lab. Setup of Program** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Pre-processing of Images** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Setup of the CNN Architecture** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Training of the Network** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Evaluation of Performance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Thesis Write-Up** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |