

# Plant Disease Classification using Convolutional Neural Network

Sanskar Singh <sup>1</sup>, Dr. Swati Sah <sup>2</sup>

<sup>1</sup>UG Scholar, Patan College for Professional Studies, Kupondole, Kathmandu, Nepal

<sup>2</sup> HOD, School of Computing, Patan College for Professional Studies, Kupondole, Kathmandu, Nepal

## ABSTRACT

Nepal is an agricultural country, where two-thirds of the population depend upon for livelihood. Pesticides and fertilizers play a key role in increasing the productivity of the crops. However, human inability to distinguish the type of plant disease on their own is leading to improper pesticide management. Previous research suggests, about 15-25% of the food products are lost annually whether in pre- or post-harvest activities reason being plant diseases due to improper pesticide management. This project aims to combine modern technologies and easy internet access to develop a tool for improving pesticide management by identifying and classifying plant diseases at farmers level. This paper presents a comprehensive study and development done on recognition and classification of plant leaf using Convolutional Neural Network. For understanding the requirements, study on several traditional approaches along with modern solutions were done during which gaps and possible improvements were found out. Field study was done to identify most common crops at the local level. Dataset on most used crops such as maize, potato, tomato and pepper were used. Each of around 1000 images on diseased and non-diseased leaves from Kaggle were used. Altogether around 10,000 images have been used for the training and validation. After using various CNN architectures, an architecture to best fit for our dataset was created. The custom architecture was able to generate an accuracy of 96%.

**Keywords:** Plant disease, Pesticide Management, Artificial Intelligence, Rural, Convolutional Neural Network, AI, Agriculture

## 1 Introduction

Nepal, a beautiful country sitting in the laps of Himalayas is rich in favorable climate and a fertile soil (Balasubramanian, 2014). This brings no surprise in Agriculture being the backbone of Nepal's economy. Farming is subsistent in nature and crop is mostly integrated with livestock. The history of farming/ agriculture dates back to several centuries. It is believed that it started with rice, wheat, potato and maize genotypes (Joshi, 2017). Later many species such as tomatoes, paddy, millet, soybean and others were cultivated. The industry was booming in success until the pests came and dried out the entire terraces of cultivating land (Khadka, 2009). Insects, mites, ticks, mice, rats and other rodents were causing a mayhem until farmers made use of pesticides. However, farmers were not knowledgeable on specific types of pests that were consuming the crops, so they used multiple pesticides that were likely. This led to an improper pesticide management which still persists today (Khadka, 2009). It not only increases the cost of the farmer but also affects the natural growth of crops subsequently having significant loss of crop productivity.

## 2 Literature Review

### 2.1 Traditional Approach

Organic production unit in Tikathali, Lalitpur have been using a honeypot, as seen in fig 1. As illustrated in the figure, the bottom of the honeypot contains sweet chemicals that allures the pests. The pests enter through the hole in the top of honeypot (shown in red arrow). Small portion of the crop field is dug and then honeypot is carefully placed inside it.

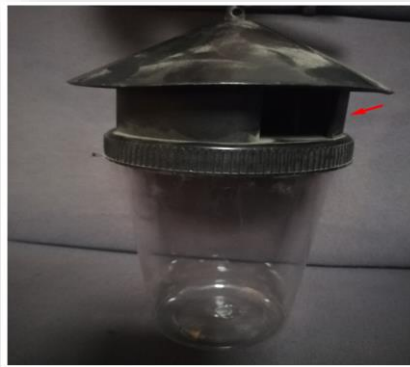


Figure 1: Traditional Approach

Some Nursery use rather simpler form such as putting sticky liquid in the leaves and stem of the crops to trap the pests. After identifying the pest, they apply the proper pesticide.

Not all farmers are aware of this idea, so they simply use Urea, sugar and powder of maize to kill the pests. It is only capable of eliminating a certain species of pest. It works effectively in a small field but fails miserably in a large production. Also, this approach heavily relies upon a certain type of crop such as rice and wheat.

## 2.2 Scientific Approach

Juan F. Molina et al. (2014) introduced a shading-based system to identify an almost blight disease or any sort of infection on tomato leaf lets. The proposed method based on the arrangement of tomato leaves with the assistance of shading descriptors. Shading portrayal is finished utilizing shading structure descriptors (for example shading histogram by quantifies the measure of significant shading) utilizing Hue-Max-Min method, scalable shading descriptor (for example driving spatial shading conveyance) and shading design descriptors (shading design variety) utilizing YCbCr shading space transform. After calculation of all these feature or descriptor values, a novel technique based on settled leaf one out cross validation method is utilized to accomplish better classification proportion. The individual descriptor configuration evaluation performed utilizing an internal loop permit, while the external loop estimates performance appraisal between different descriptors. The creator inferred that shading structure descriptor gave better precision than different methods.

## 2.3 Critical Analysis

A robust system for automatic detection and classification of disease could be a valued tool in order to support early diagnosis. Accuracy generated is ok-ish so it could further be improved by experimenting with other classification methods. A greater dataset would be required for a guaranteed good statistical property along the learning process of the classifier. Wider set of descriptors such as texture, shape, size, etc. could also be considered.

Kuo-YI Huang (2007) presented a methodology utilizing artificial neural networks for recognition of Phalaenopsis seedling disease with surface and color analysis. Lesion regions were portioned utilizing underlining guideline to separate background (pot), object (leaves) and exponentially change a variable parameter to improve the picture and to assess surface highlights of the lesion zone Gray Level Co-event matrix (GLCM) utilized. From that point onward, tricolor mean values of lesion regions are utilized to order ailments. The back propagation neural system classifier is then used to sort different deformities. At long last, creators had the option to recognize and classify noticeable lesion zones, however they were unfit to look at the contaminated territory on the encased cutting edges. The recognition and characterization framework can be likewise helpful for the improvement of

blossoms and plant leaves in greenhouse.

## 2.4 Critical Analysis

The detection capability of the system without being able to classify the disease type is as high as 97%. The model can classify the disease 89.6% of the time. The system can identify and classify visible lesion areas but is unable to inspect infected areas on the covered blades. Training samples of only 145 were used. The accuracy could drop significantly under large number of datasets.

Pantech Solutions (2019) developed a system using BNN classifier to predict plant diseases using texture-based segmentation. Input image is resized, and region of interest selection is performed when required. Color features like mean, standard deviation of HSC color space and texture features like energy, homogeneity, contrast and correlation are extracted for network training and classification. Automatic classifier such as BPN with FF are used for classification. Network uses tangent sigmoid function as kernel function. Finally, authors were able to get simulated result with minimum error during training and better accuracy during classification.

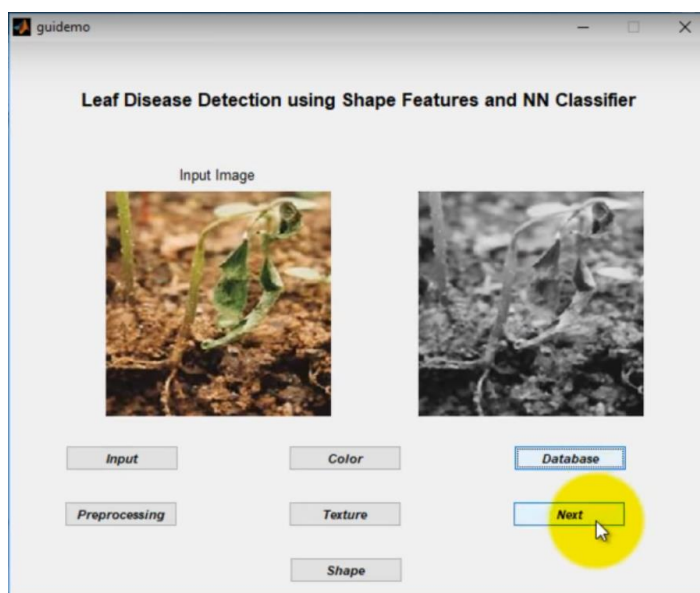


Figure 1: GUI pantech solution (Pantech, 2019)

## 3 STUDIES AND FINDINGS

Field visit was also done for understanding the requirements at the local level. For which following questions were asked:

### 3.1 Analysis of target audience

Can you tell your qualification?

41 responses

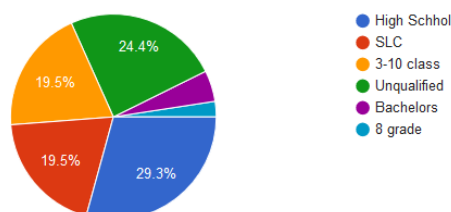


Figure 2: Analysis of target audience

The system to be developed can have complex features and advance UI with respective to the target

audience. Farmer tend to have low qualification. One quarter or 24.4% have not studied at all and about 20% have studied from class 3 - 10 only. So, the application will have simple features and simple UI so that it will be accessible to all people independent of their qualification. The tasks will be performed with as few clicks as possible.

### 3.2 Analysis of Most Grown Crops

It is necessary to get an input on specific detail of the most common crops grown in the locality so that the application can meet the local level demands. The crops were first segregated into four aspects being grain, vegetables, pulses and fruit. Later on, more specific questions were asked.

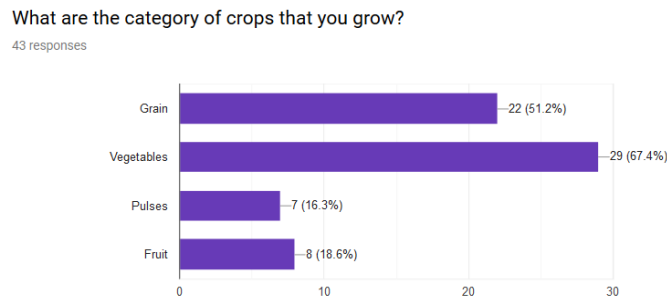


Figure 4: Category of crops

### 3.3 Getting one step closer

The following question was asked as an introductory question to step into the main matter. 88.4% of the farmers get the pests mean these farmers can further be interrogated to obtain more results.

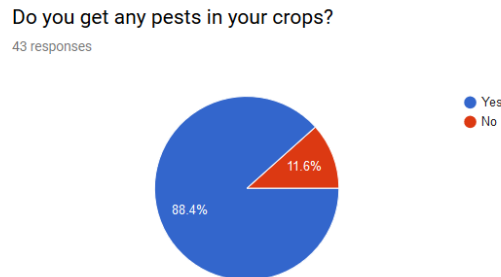


Figure 9: Getting pest in crops

## 4 DATA ANALYSIS AND INTERPRETATION

### 4.1 Model Architecture

For the visualization of model structure, the following function was used `model.summary()`

```

model.summary()
Model: "sequential_1"

```

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 256, 256, 32)	896
activation_1 (Activation)	(None, 256, 256, 32)	0
batch_normalization_1 (Batch Normalization)	(None, 256, 256, 32)	128
max_pooling2d_1 (MaxPooling2D)	(None, 85, 85, 32)	0
dropout_1 (Dropout)	(None, 85, 85, 32)	0
conv2d_2 (Conv2D)	(None, 85, 85, 64)	18496
activation_2 (Activation)	(None, 85, 85, 64)	0
batch_normalization_2 (Batch Normalization)	(None, 85, 85, 64)	256
conv2d_3 (Conv2D)	(None, 85, 85, 64)	36928
activation_3 (Activation)	(None, 85, 85, 64)	0
batch_normalization_3 (Batch Normalization)	(None, 85, 85, 64)	256
max_pooling2d_2 (MaxPooling2D)	(None, 42, 42, 64)	0
dropout_2 (Dropout)	(None, 42, 42, 64)	0
conv2d_4 (Conv2D)	(None, 42, 42, 128)	73856
activation_4 (Activation)	(None, 42, 42, 128)	0
batch_normalization_4 (Batch Normalization)	(None, 42, 42, 128)	512
conv2d_5 (Conv2D)	(None, 42, 42, 128)	147584
activation_5 (Activation)	(None, 42, 42, 128)	0
batch_normalization_5 (Batch Normalization)	(None, 42, 42, 128)	512
max_pooling2d_3 (MaxPooling2D)	(None, 21, 21, 128)	0
dropout_3 (Dropout)	(None, 21, 21, 128)	0
flatten_1 (Flatten)	(None, 56448)	0
dense_1 (Dense)	(None, 1024)	57803776
activation_6 (Activation)	(None, 1024)	0
batch_normalization_6 (Batch Normalization)	(None, 1024)	4096
dropout_4 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 15)	15375
activation_7 (Activation)	(None, 15)	0

```

Total params: 58,102,671
Trainable params: 58,099,791
Non-trainable params: 2,880

```

Figure 13: Model Summary

## 4.2 Grasping the Used Convolutional Filters and Operations

In this model, a total of 5 convolutional layers with 26 filters each have been used. Following steps has been used throughout the model. First the images are imported to the apply the filters and in the process get the feature maps.

```
In [7]: im=plt.imread("potato.jpg")
plt.imshow(im)

Out[7]: <matplotlib.image.AxesImage at 0x292a7c6f2e8>
```

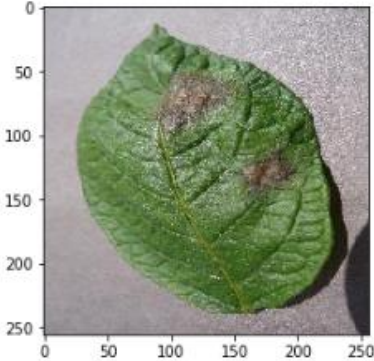


Figure 14: Read Image

For viewing the feature maps and seeing the number of convolutional layers in the network.

```
In [13]:
for layer in model.layers:
    # check for convolutional layer
    if 'conv' not in layer.name:
        continue
    # get filter weights
    filters, biases = layer.get_weights()
    print(i, layer.name, layer.output.shape)

27 conv2d_1 (?, 256, 256, 32)
27 conv2d_2 (?, 85, 85, 64)
27 conv2d_3 (?, 85, 85, 64)
27 conv2d_4 (?, 42, 42, 128)
27 conv2d_5 (?, 42, 42, 128)
```

After that a function was created to plot the feature maps.

```
from keras.models import Model
from matplotlib import pyplot
from numpy import expand_dims
from matplotlib import pyplot as plt
ixs = [0,1,2,3,4]
outputs = [model.layers[i].output for i in ixs]
model = Model(inputs=model.inputs, outputs=outputs)
# Load the image with the required shape
img = load_img('potato.jpg', target_size=(256, 256))
# convert the image to an array
img = img_to_array(img)
# expand dimensions so that it represents a single 'sample'
img = expand_dims(img, axis=0)
img = preprocess_input(img)
# get feature map for hidden layer
feature_maps = model.predict(img)
# plot the output from each block
square = 5
for fmap in feature_maps:
    ix = 1
    for _ in range(square):
        for _ in range(square):
            # specify subplot and turn of axis
            ax = pyplot.subplot(square, square, ix)
            ax.set_xticks([])
            ax.set_yticks([])
            # plot filter channel in grayscale
            pyplot.imshow(fmap[0, :, :, ix-1], cmap='gray')
            ix += 1
    # show the figure
    pyplot.savefig('filters.pdf')
    pyplot.show()
```

Figure 15: Feature Map plotting

- First Feature Map

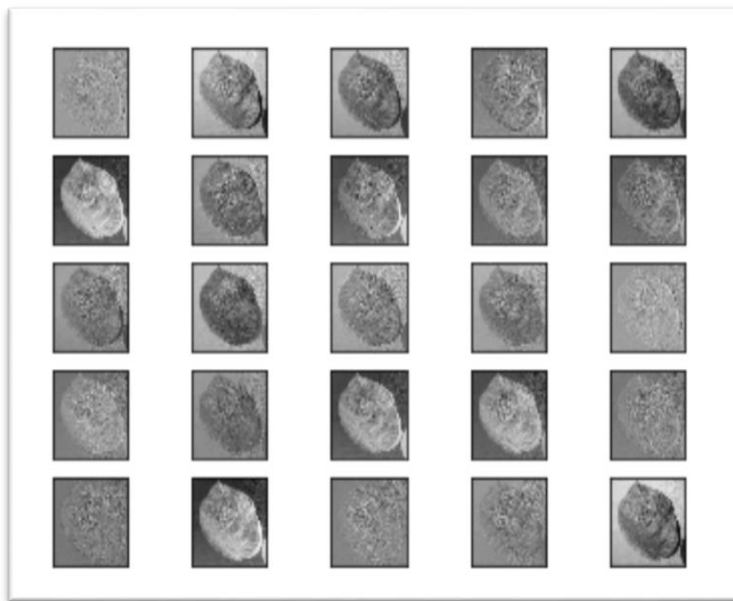


Figure 16: First Feature Map

- Second Feature Map

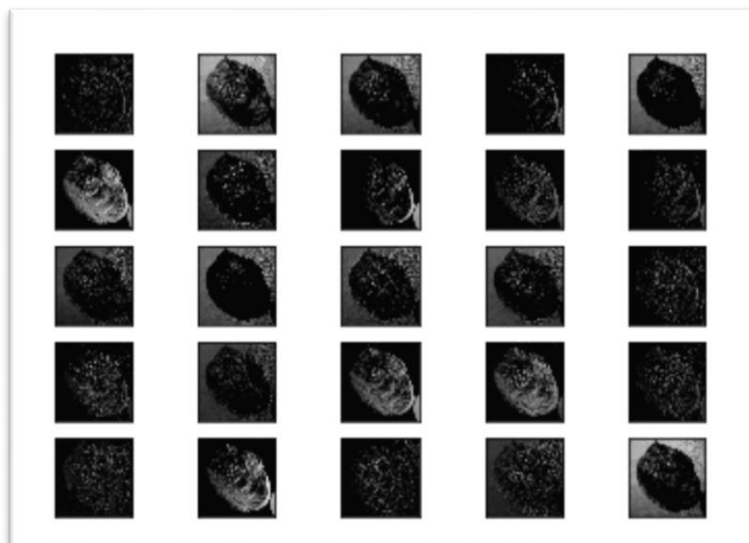


Figure 17: Second Feature Map

- Third Feature Map

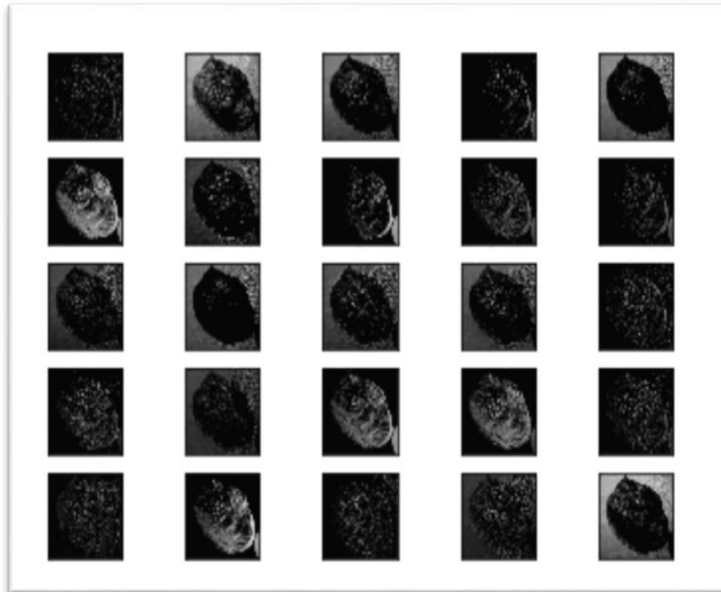


Figure 18: Third Feature map

- Fourth Feature Map

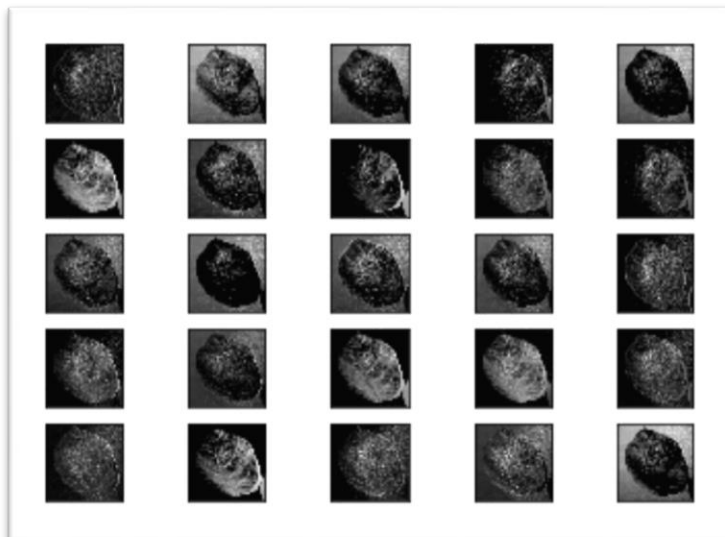


Figure 19: Fourth Feature Map

#### 4.2.1 ReLU Activation Function

Mathematically defined by  $y = \max(0, x)$ , ReLU is a most commonly used activation function in neural networks. Particularly for Convolutional Neural Networks. Using simple math, it drops the neuron with the negative (-ve) value and activates neuron with the value of greater than 0. Below is an example to it.

$$\begin{bmatrix} 1 \\ 4 \\ -1 \end{bmatrix} \xrightarrow{\text{Relu}} \begin{bmatrix} 1 \\ 4 \\ 0 \end{bmatrix}$$



#### 4.2.2 Equation of ReLU

$$f(x) \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

$$\text{Ex: } f(x) \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$$

Figure 20: Equation of ReLU (Sharma, 2017)

#### 4.2.3 Max Pooling: Pooling Layer

Features are stored in matrixes. Applying a pooling layer helps to reduce the computational power. It drastically reduces the size of parameters without omitting the important features. Below is a matrix before the pooling layer has been applied.

2	2	7	3
9	4	6	1
8	5	2	4
3	1	2	6

After the max pool for the matrix above has been applied with a filter of (2x2), the following matrix is obtained.

9	7
8	6

In this model, pool size of (3x3) has been used.

```
model.add(MaxPooling2D(pool_size=(3, 3)))
```

Figure 21: Max Pooling

#### 4.2.4 Regularization

The model maybe able to achieve high accuracy but when test with new images, the model may not be able to achieve good accuracy. This is called overfitting. Reducing the overfitting is done by the slight modification of the algorithm so that it is better generalized.

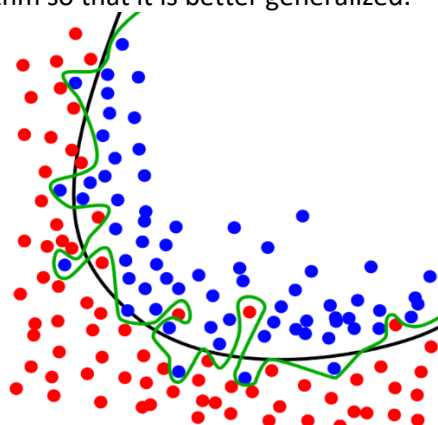


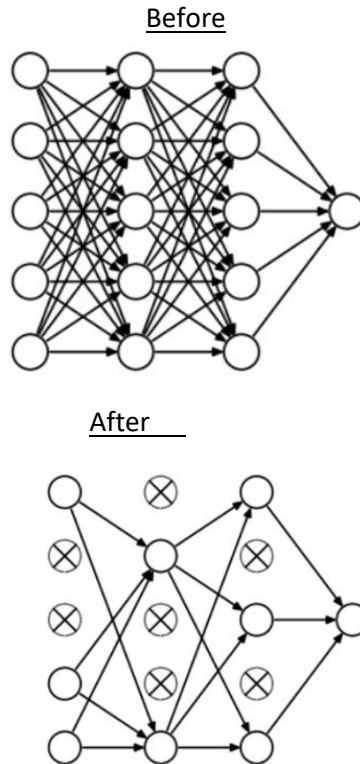
Figure 22: Overfitting (Sharma, 2017)

In the provided image above, the black line is the generalized fitting whereas the green line shows overfitting. In this model, a ratio of 0.25 has been used:

```
dropout_1 (Dropout) (None, 85, 85, 32) 0
model.add(MaxPooling2D(pool_size=(3, 3)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), padding="same"))
```

Figure 23: DropOut

In every iteration, dropout carefully deactivates or drops a selected number of neurons.



### 4.2.5 Flatten Layer

This layer converts the data into a 1-dimensional array so that matrix can be processed into dense layer.

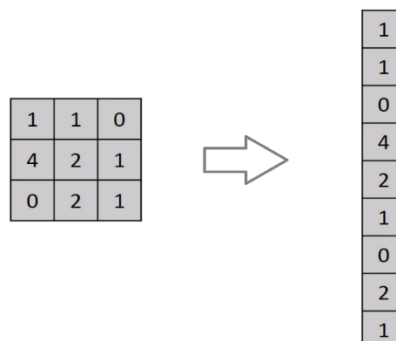


Figure 24: Flatten Layer (Paudel, 2012)

Usage in this model

Flatten_1 (Flatten)	(None, 56448)	0
---------------------	---------------	---

Figure 25: Flatten Layer usage

### 4.3 Snapshot of The Home Page of Application

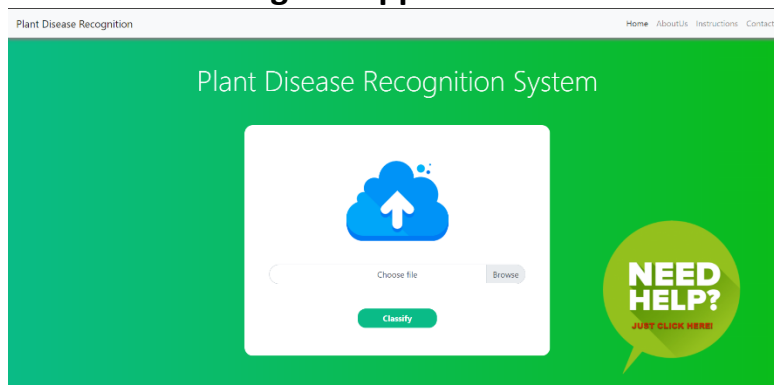


Figure 26: Snapshot of the home UI

#### 4.3.1 Description

A user-centric design has been made. The target audience of the project are farmers and farmers can have difficulty navigating in multipage designs. So, the main feature of the project has been displayed in the first page of the website. Proper icon selection has been done so that the image itself can help farmers identify the area they need to focus on. In case they have any difficulty, a big help icon is added on the side. After clicking on it, they are navigated to a new interface where they can view the instructions on how to use the page. For the home UI, a green color gradient has been selected as the project title is "Plant Disease Recognition System".

## 5 Conclusion and future work

### 5.1 Critical Evaluation

Even though there are already few applications that are designed and developed to identify and classify plant diseases, this project focuses more towards the real-life needs in the local level. Field study and domain research has been done to understand the requirement and needs of the farmers in the local level. There are hundreds and thousands of diseases a single plant can have so the researcher has focused more towards the most often occurring diseases in the locality.

The major obstacle the researcher had is the colour segmentation. As most of the plants have the leaves in green colour, it was hard to identify the type of plant. The Brightside is the system is capable of identifying and classifying the disease 97% of the time. The accuracy in the beginning was about 90-ish percentage and was later trained for multiple epochs to achieve such high accuracy. Another problem is, some of the disease have only about 200 samples. So, the system may identify other disease as shown in the confusion matrix. However, dataset for most of the disease is in good numbers and has consistent result even in the real-life scenario.

Another issue of the matter is it is not able to detect any image other than plant image. Let's say a user uploads cat image, the system is not able to say that it is not a valid plant leaf image. Due to the time limitations it could not be incorporated. This problem along with other will be processed in later version of the project. The greatest strength of the UI of the system is that is very simple, minimalistic and sticks to the core. However, upon doing an evaluation study, it was found that users want more features to have. Their recommendations have been recorded and will be valued. Speaking of recommendations, after data analysis another must have feature was identified. Farmers can do very less with just the identified disease name. They may or may not know the solution to the identified disease, so a recommendation page was also added.

## 5.2 Conclusion

The system proposed is able to predict high level of accuracy. This was possible only after finding the gaps of previous researchers by doing secondary research i.e. literature review. Also, primary research methods and tools was used to get a detailed idea on the existing technologies available in the market. Primary researched helped me find the current necessity of features in the product. As discussed, secondary researched helped me find the gaps in the existing technologies. It was found that the most of the researchers used images under controlled environments. So, this project makes use of image with improper lighting conditions. Although, in this project itself a higher number of datasets could yield more precise results. More feedbacks have been collected from users through user evaluation. These feedbacks will be analysed and used for the further modifications and improvement of the developed system.

The project researcher was planning to make aimed at creating a complete solution to diseases in most common plants. After diving into the world of plant diseases, researcher found out that there are hundreds and thousands of diseases in every single plant. Since the researcher was already in the middle of the project, he needed a solution. So, just like the most common plants in the locality, the researcher/developer has predicted and classified the most common plant diseases.

The project followed agile methodology. For dividing the work flow and setting a milestone, a Gantt chart had been created. The report has been segregated systematically and chronologically in order to provide information on the research flow and development process. Every component and techniques have been illustrated in form of explanations, figures and tables. This system is based on a computer science and agriculture field to connect both of them and make the work more efficient.

## 5.3 Future Work

Excellent feedbacks have been received from the users from during the Evaluation Testing. Following additions will be done in the next version of the project. A loading screen will be added during the processing of the system. More and more plants will be added for the plant disease classification. A log in system will added so that people can record and view their history. Last but nonetheless, Nepali font integration could be done for the accessibility for the farmers.

## References

- Oo, Y. and Htun, N. (2018). Plant Leaf Disease Detection and Classification using Image Processing. *International Journal of Research and Engineering*, 5(9), pp.516-523.
- Cortes, E. (2015). Plant Disease Classification Using Convolutional Networks and Generative Adversarial Networks. Stanford University, pp. 4321-4325
- Nepal Economy 2019, CIA World Factbook. 2019. Nepal Economy 2019, CIA World Factbook
- Singla, S. Design for the Methodology of Plant Leaf Disease Detection using Image Processing, *International Journal of Latest Trends in Engineering and Technology*, 7(3), pp. 148-152
- Adhikari, P. (2017). An overview of pesticide management in Nepal. *The Journal of Agriculture and Environment*, pp. 95-105
- Rani, S. and Kautish, S., 2018, June. Association Clustering and Time Series Based Data Mining in Continuous Data for Diabetes Prediction. In *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1209-1214). IEEE.
- Mohan, K.J., Balasubramanian, M. and Palanivel, S., 2016. Detection and recognition of diseases from paddy plant leaf images. *International Journal of Computer Applications*, 144(12).
- Orillo, J.W., Cruz, J.D., Agapito, L., Satimbre, P.J. and Valenzuela, I., 2014, November. Identification of diseases in rice plant (*oryza sativa*) using back propagation Artificial Neural Network. In *2014 International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)* (pp. 1-6). IEEE.
- Molina, J.F., Gil, R., Bojacá, C., Gómez, F. and Franco, H., 2014, September. Automatic detection of early

- blight infection on tomato crops using a color-based classification strategy. In 2014 XIX Symposium on Image, Signal Processing and Artificial Vision (pp. 1-5). IEEE.
- Mahlein, A.K., Steiner, U., Hillnhütter, C., Dehne, H.W. and Oerke, E.C., 2012. Hyperspectral imaging for small-scale analysis of symptoms caused by different sugar beet diseases. *Plant methods*, 8(1), p.3.
- Kautish, S. and Thapliyal, M.P., 2012. Concept of Decision Support Systems in relation with Knowledge Management–Fundamentals, theories, frameworks and practices. *International Journal of Application or Innovation in Engineering & Management*, 1, pp.1-9.
- Kai, S., Zhikun, L., Hang, S. and Chunhong, G., 2011, January. A research of maize disease image recognition of corn based on BP networks. In 2011 third international conference on measuring technology and mechatronics automation (Vol. 1, pp. 246-249). IEEE.
- Kautish, S., 2013. Knowledge sharing: A contemporary review of literature in context to information systems designing. *ACADEMICIA: An International Multidisciplinary Research Journal*, 3(1), pp.101-114.
- Zhang, M. and Meng, Q., 2011. Automatic citrus canker detection from leaf images captured in field. *Pattern Recognition Letters*, 32(15), pp.2036-2046.
- Sharma, D., Thapa, R., Manandhar, H., Shrestha, S. and Pradhan, S. (2013) "Use Of Pesticides In Nepal And Impacts On Human Health And Environment", *Journal of Agriculture and Environment*, 13, pp. 67-74. doi: 10.3126/aej.v13i0.7590
- Baker, S., and Gyawali, B.K. (1994). Promoting proper pesticide use in Nepal. Policy analysis in agriculture and related resource management research report 28.
- Kaur, R., & Kautish, S. (2019). Multimodal Sentiment Analysis: A Survey and Comparison. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET)*, 10(2), 38-58.
- VC Web Design. 2019. What is Waterfall? and Why Do We Use It? - VC Web Design. Available at: <http://vcwebdesign.com/uncategorized/what-waterfall-why-do-we-use-it/>.
- Aduwo, J.R., Mwebaze, E. and Quinn, J.A., 2010. Automated Vision-Based Diagnosis of Cassava Mosaic Disease. In *Industrial Conference on Data Mining-Workshops* (pp. 114-122).
- Types of Software Testing: Different Testing Types with Details. 2019. Types of Software Testing: Different Testing Types with Details. Available at: <https://www.softwaretestinghelp.com/types-of-software-testing/>
- Wikipedia. 2019. Agriculture in Nepal - Wikipedia. Available at: [https://en.wikipedia.org/wiki/Agriculture\\_in\\_Nepal](https://en.wikipedia.org/wiki/Agriculture_in_Nepal)
- Gyawali, K. (2018) "Pesticide Uses and its Effects on Public Health and Environment", *Journal of Health Promotion*, 6, pp. 28-36. doi: 10.3126/jhp.v6i0.21801
- Dhital, Sushma & Rupakheti, Dipesh & Tripathee, Lekhendra & Sigdel, Shalik. (2015). A Review on Status of Pesticides Use in Nepal
- Wikipedia. 2019. Qualitative research - Wikipedia. [ONLINE] Available at: [https://en.wikipedia.org/wiki/Qualitative\\_research](https://en.wikipedia.org/wiki/Qualitative_research)
- Wikipedia. 2019. Qualitative research - Wikipedia. [ONLINE] Available at: [https://en.wikipedia.org/wiki/Qualitative\\_research](https://en.wikipedia.org/wiki/Qualitative_research)
- Social Research Methods - Knowledge Base - Survey Research. 2019. Social Research Methods - Knowledge Base - Survey Research. [ONLINE] Available at: <https://socialresearchmethods.net/kb/survey.php>
- The Web framework for perfectionists with deadlines | Django. 2019. The Web framework for perfectionists with deadlines | Django. [ONLINE] Available at: <https://www.djangoproject.com/>
- Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D. and Stefanovic, D., 2016. Deep neural networks-based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016.
- Variables and Hypotheses. 2020. Variables and Hypotheses. [ONLINE] Available at: <http://web.csulb.edu/~msaintg/ppa696/696vars.htm>
- Santurkar, S., Tsipras, D., Ilyas, A. and Madry, A., 2018. How does batch normalization help optimization? In *Advances in Neural Information Processing Systems* (pp. 2483-2493)

- Ide, H. and Kurita, T., 2017, May. Improvement of learning for CNN with ReLU activation by sparse regularization. In 2017 International Joint Conference on Neural Networks (IJCNN) (pp. 2684-2691). IEEE.
- Wu, H. and Gu, X., 2015, November. Max-pooling dropout for regularization of convolutional neural networks. In International Conference on Neural Information Processing (pp. 46-54). Springer, Cham.
- YouTube. 2020. How to Make a UML Sequence Diagram - YouTube. [ONLINE] Available at: <https://www.youtube.com/watch?v=pCK6prSq8aw&t=302s>
- YouTube. 2020. UML Use Case Diagram Tutorial - YouTube. [ONLINE] Available at: <https://www.youtube.com/watch?v=zid-MVo7M-E&t=2s>
- Singh, C. and Kautish, S.K., Page Ranking Algorithms for Web Mining: A Review. International Journal of Computer Applications, 975, p.8887.
- Cruz, J.M., Chua, L.O. and Roska, T., 1994, December. A fast, complex and efficient test implementation of the CNN universal machine. In Proceedings of the Third IEEE International Workshop on Cellular Neural Networks and their Applications (CNNA-94) (pp. 61-66). IEEE.
- Maria Vasilenko. 2020. Don't Be Confused by a Confusion Matrix - Maria Vasilenko - Medium. [ONLINE] Available at: <https://medium.com/@marie.vasilenko/dont-be-confused-by-a-confusion-matrix-25f7ea759163>
- Kautish, S., 2008. Online Banking: A Paradigm Shift. E-Business, ICFAl Publication, Hyderabad, 9(10), pp.54-59.
- How to Create a Gantt Chart. 2020. How to Create a Gantt Chart. [ONLINE] Available at: <https://www.gantt.com/creating-gantt-charts.htm>
- Wikipedia. 2020. MindView - Wikipedia. [ONLINE] Available at: <https://en.wikipedia.org/wiki/MindView>
- Zhang, S.W., Shang, Y.J. and Wang, L., 2015. Plant disease recognition based on plant leaf image. The journal of animal & plant sciences, 25(3 Supplement 1), pp.42-45.
- Lathrop, L.D. and Pennypacker, S., 1980. Spectral classification of tomato disease severity levels. Photogram. Eng. Remote Sens, 143, pp.439-445.
- Sampathkumar, A., Rastogi, R., Arukonda, S., Shankar, A., Kautish, S. and Sivaram, M., 2020. An efficient hybrid methodology for detection of cancer-causing gene using CSC for micro array data. Journal of Ambient Intelligence and Humanized Computing, pp.1-9.
- Kaur, R. and Kaur, M., 2017. A brief review on plant disease detection using in image processing. International journal of computer science and mobile computing, 6(2), pp.101-106.
- Arivazhagan, S., Shebiah, R.N., Ananthi, S. and Varthini, S.V., 2013. Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. Agricultural Engineering International: CIGR Journal, 15(1), pp.211-217.
- Naik, M.R. and Sivappagari, C.M.R., 2016. Plant leaf and disease detection by using HSV features and SVM classifier. International Journal of Engineering Science, 3794.
- Henderson, D., Williams, C.J. and Miller, J.S., 2007. Forecasting late blight in potato crops of southern Idaho using logistic regression analysis. Plant disease, 91(8), pp.951-956.
- Prabhakar, M., Prasad, Y.G., Desai, S., Thirupathi, M., Gopika, K., Rao, G.R. and Venkateswarlu, B., 2013. Hyperspectral remote sensing of yellow mosaic severity and associated pigment losses in Vigna mungo using multinomial logistic regression models. Crop protection, 45, pp.132-140.
- Ferentinos, K.P., 2018. Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture, 145, pp.311-318.
- Rani, S. and Kautish, S., 2018. Application of data mining techniques for prediction of diabetes-A review. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 3(3), pp.1996-2004.
- Nagasubramanian, K., Jones, S., Singh, A.K., Singh, A., Ganapathysubramanian, B. and Sarkar, S., 2018. Explaining hyperspectral imaging-based plant disease identification: 3D CNN and saliency maps. arXiv preprint arXiv:1804.08831.

- YouTube. 2020. Convolutional Neural Network (CNN) | Convolutional Neural Networks with TensorFlow | Edureka - YouTube. [ONLINE] Available at: [https://www.youtube.com/watch?v=umGJ30-15\\_A&list=PLbD7UUcOkY4QhEgOqF9rviEMdY\\_B-gxFO&index=56](https://www.youtube.com/watch?v=umGJ30-15_A&list=PLbD7UUcOkY4QhEgOqF9rviEMdY_B-gxFO&index=56)
- Kumar, A., Rajpurohit, V.S. and Kautish, S., 2020. A Study on Technology-LED Solutions for Fruit Grading to Address Post-Harvest Handling Issues of Horticultural Crops. In Modern Techniques for Agricultural Disease Management and Crop Yield Prediction (pp. 203-221). IGI Global.
- Kaustumbh Jaiswal. 2020. How to deploy a Machine Learning model using Django: Part-1. [ONLINE] Available at: <https://medium.com/saarthi-ai/deploying-a-machine-learning-model-using-django-part-1-6c7de05c8d7>
- Bootsnipp.com. 2020. Bootstrap Aboutus Examples. [ONLINE] Available at: <https://bootsnipp.com/tags/aboutus>
- Stepshot. 2020. How to Create a User Instruction Manual | Stepshot. [ONLINE] Available at: <https://stepshot.net/how-to-create-a-user-instruction-manual/>
- Philip Hodgson. 2020. Tips for writing user manuals. [ONLINE] Available at: <https://www.userfocus.co.uk/articles/usermanuals.html>