

**Auto Chloro**

A Plant Disease Classifier & Remedies Provider (Using Deep Learning)

Presented by **Team Hydro AI**:

**1. Md. Farhadul Islam**

CSE, Brac University

**2. Sarah Zabeen**

CS, Brac University

***What is Auto Chloro:***

1. A deep-learning based plant disease classifier
2. Provides remedies of the disease
3. In Bengali Language

***Why Auto Chloro:***

1. Bangladesh is an agricultural country
2. Detecting diseases instantly
3. Getting remedies instantly
4. Simple & Easy to use
5. Using AI in agriculture to solve real world challenges

***What makes Auto Chloro Unique:***

1. Bengali Language, so that every user in Bangladesh understands
2. Using a strong Convolutional Neural Network Model
3. Easy to operate User Interface

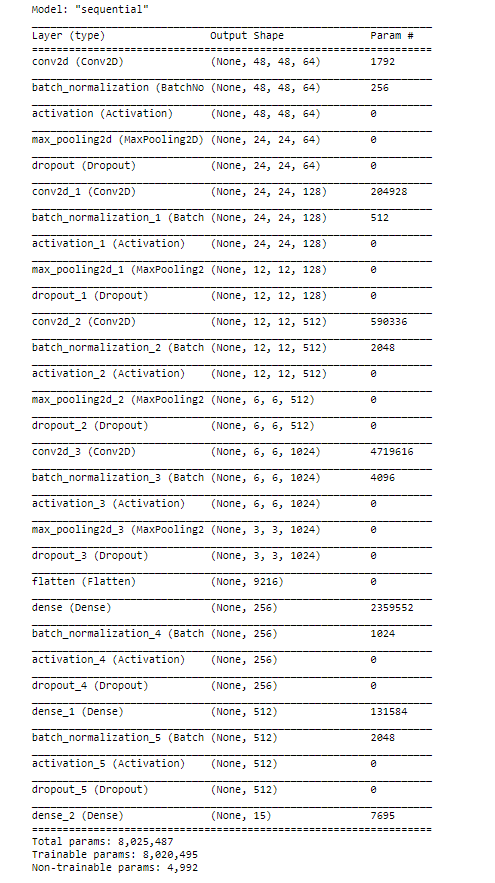
**Details & Breakdown**

**DATSET:**

The dataset we will be using contains 17476 images. The train dataset has 16222 images belonging to 15 classes and the test dataset has 1254 images belonging to 15 classes.

Dataset link: <https://www.kaggle.com/vasanthkumar14/plant-disease>

**LIBRARIES:**

* Numpy
* Matplotlib
* OS
* Tensorflow
* EasyGUI

**CNN MODEL:**

We used a sequential model. The Sequential Model API is a way to build deep learning models that create a sequential class and create and add model layers to it. We used 4 convolutional layers with “Relu” (Rectified Linear Unit) activation functions. The parameters of the first conv2D are, filter-size, kernel-size, Input-shape. **The convolutional layer is then pass to MaxPooling layer,pooliing size is the window size.**We use Flatten to convert data into 1 Dimensional form. Dense layer feeds all outputs from the previous layer to all its neurons, each neuron providing one output to the next layer. Dropout function is a simple way to prevent overfitting. Dropout is a technique where randomly selected neurons are ignored during training. We used ‘Adam’ as our optimizer to optimize our data with learning rate. A metric is a function that is used to measure the model's performance. **Here we take 15 epochs for train our model. More epochs increase the accuracy and decrease the loss (but it takes more time too).**

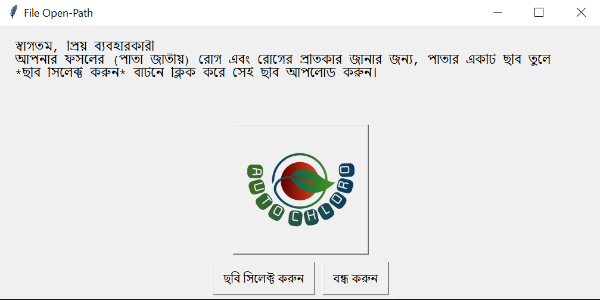
**SAVING THE MODEL (h5):**

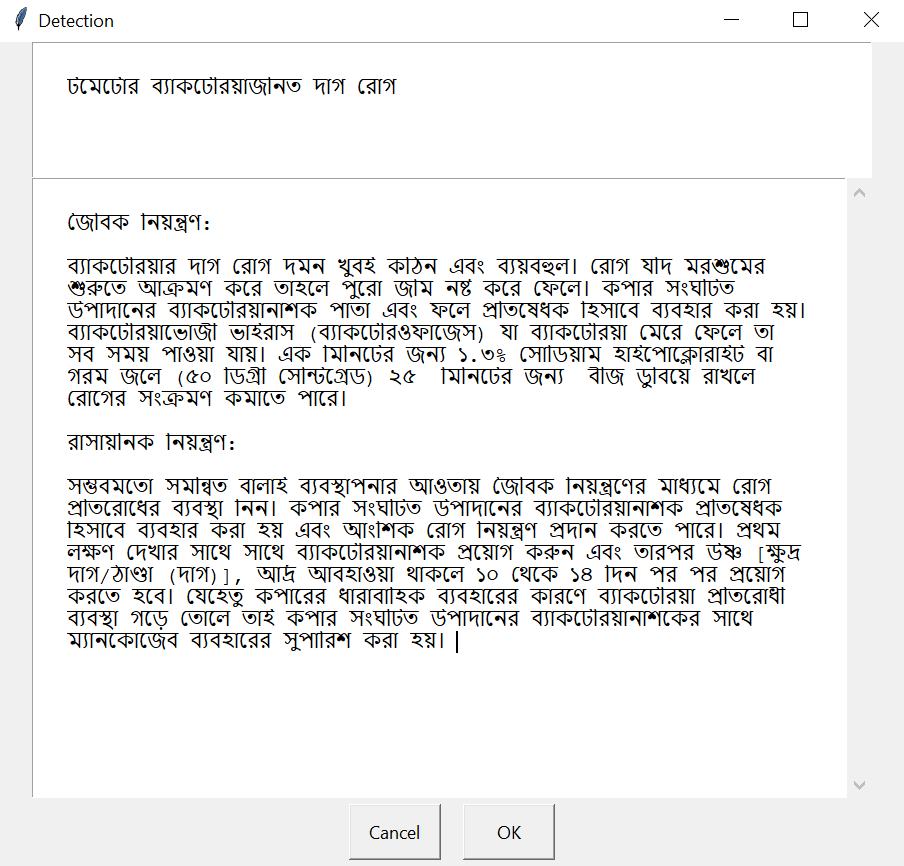
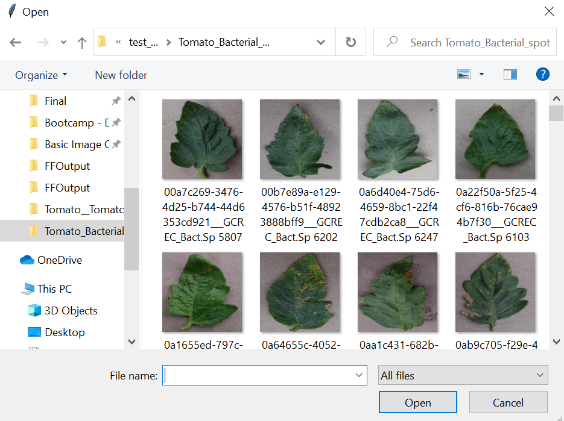
**It takes a lot of time to train the model. Therefore, we save the trained model so that we can save the time. Moreover, we have to use the saved model in our GUI, as it’s an application.**

**GUI:**

**We used easygui, a simple GUI framework based on tkinter. At first, we load the saved model that we trained previously. Basically, we use the fileopenbox function to get the image path. Then, we load the image with load\_img method. After that, we covert the image to array and expand the dimension where axis=0, it defines the index at which dimension should be inserted. If input has D dimensions then axis must have value in range [-(D+1), D].**

We make a list of our labels and use multiple if-else statement to match our prediction with the diseases. Finally, we show the disease name and remedies in textbox.





**CURRENT STATUS AND BUGS**

Currently we can show the disease properly if the image is good. The dataset is not good enough to predict the diseases properly every time. It does show wrong outputs sometimes. We have 3 types of plants. The GUI is pretty simple and it was intentional, although it needs more work. Overall, the CNN model is good enough to show some good results and the whole code is working properly.

**FUTURE PLANS**

* Increasing the data, based on Bangladeshi Crops/Plants
* Creating a website and an app so that it can reach the people easily
* Better design for the GUI
* Adding this project to another one which is an automated cultivation system using IOT and ML.