Pothole Detection System using Convolution Neural Networks

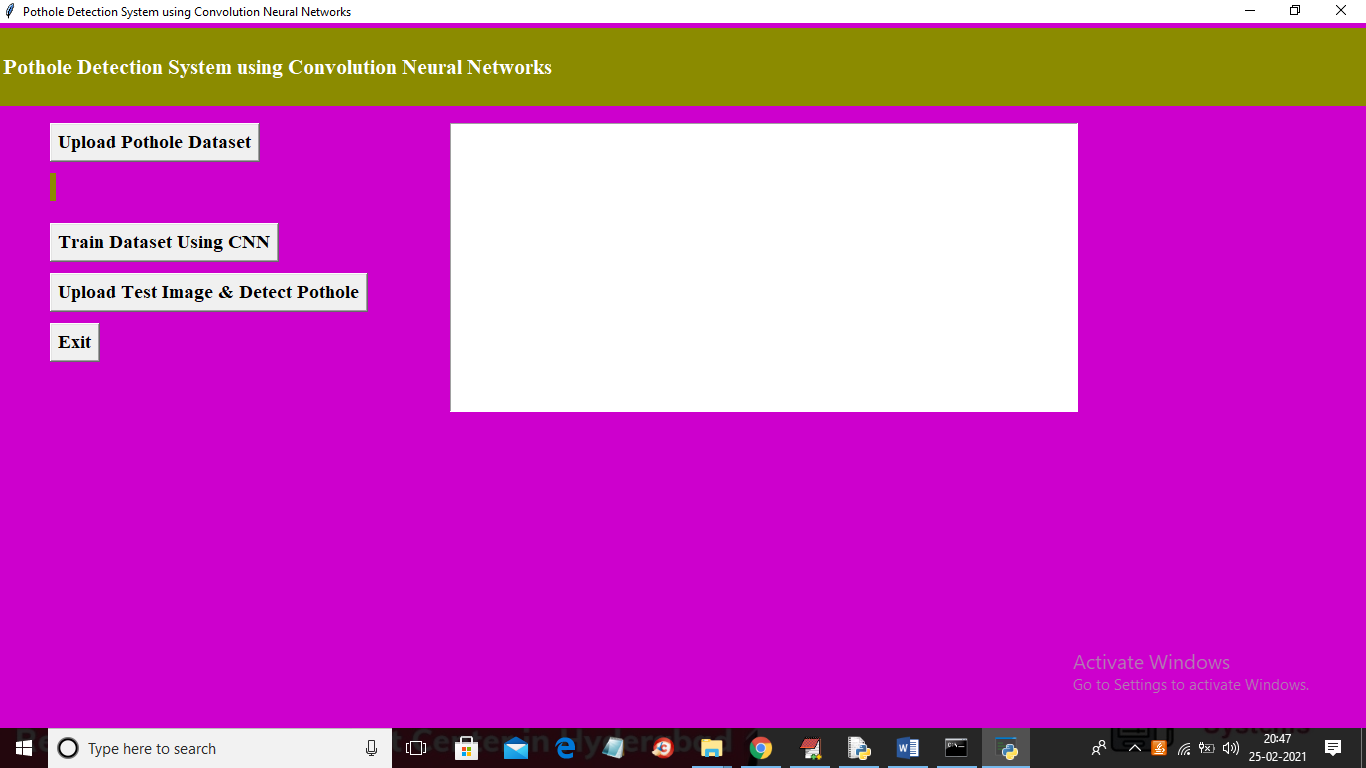
In this project we are using Convolution Neural Network (CNN) to detect pothole from given images. To implement this project we have designed following modules

1) Upload Pothole Dataset: using this module we can upload train and test images of pothole to application and then application will read all images and convert all images into train and test array

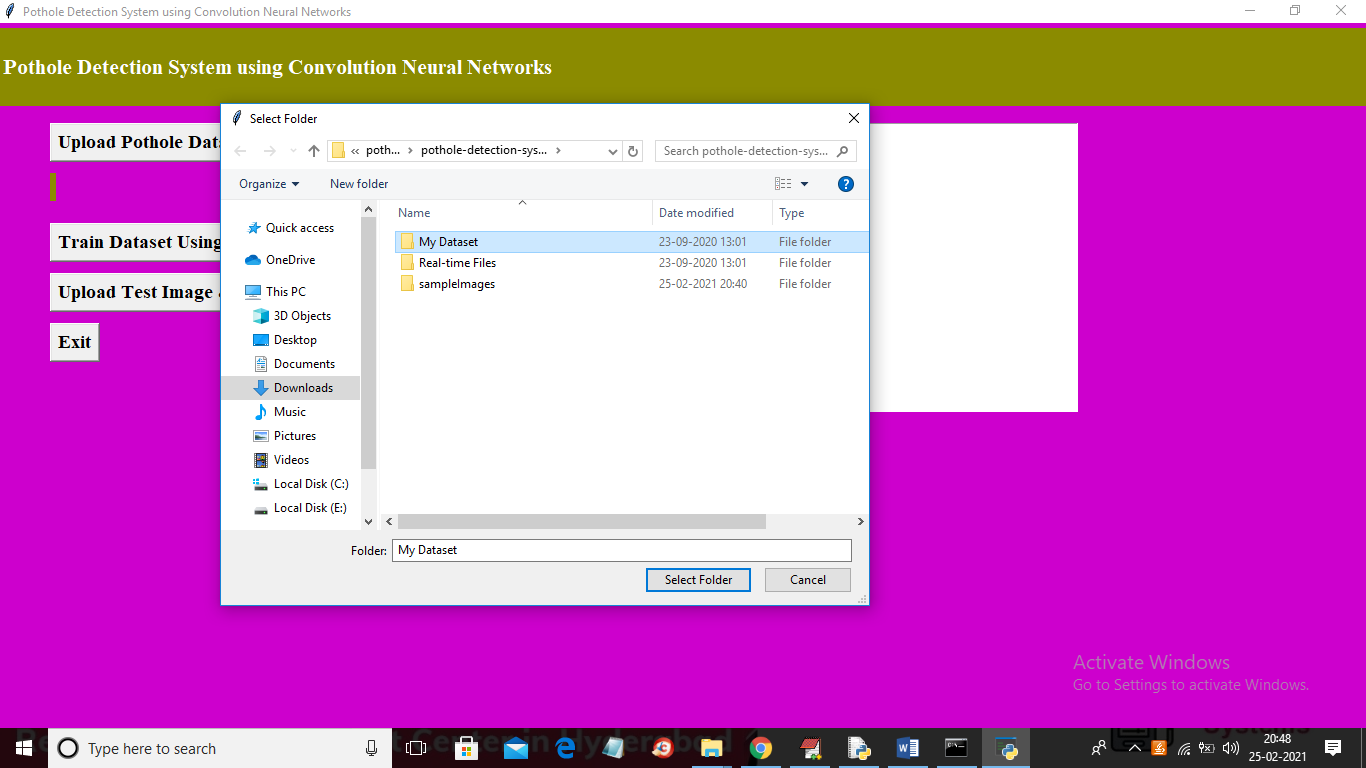
2) Train Dataset Using CNN: using this module all train and test array will be pass to CNN layers and then this layers will filter all images array and then build classification model where each image will be classified into one of two types such as ‘Pothole Detected’ or ‘No Pothole Detected’.

3) Upload Test Image & Detect Pothole: using this module we will upload test image and then CNN will classify that image as pothole contains or not.

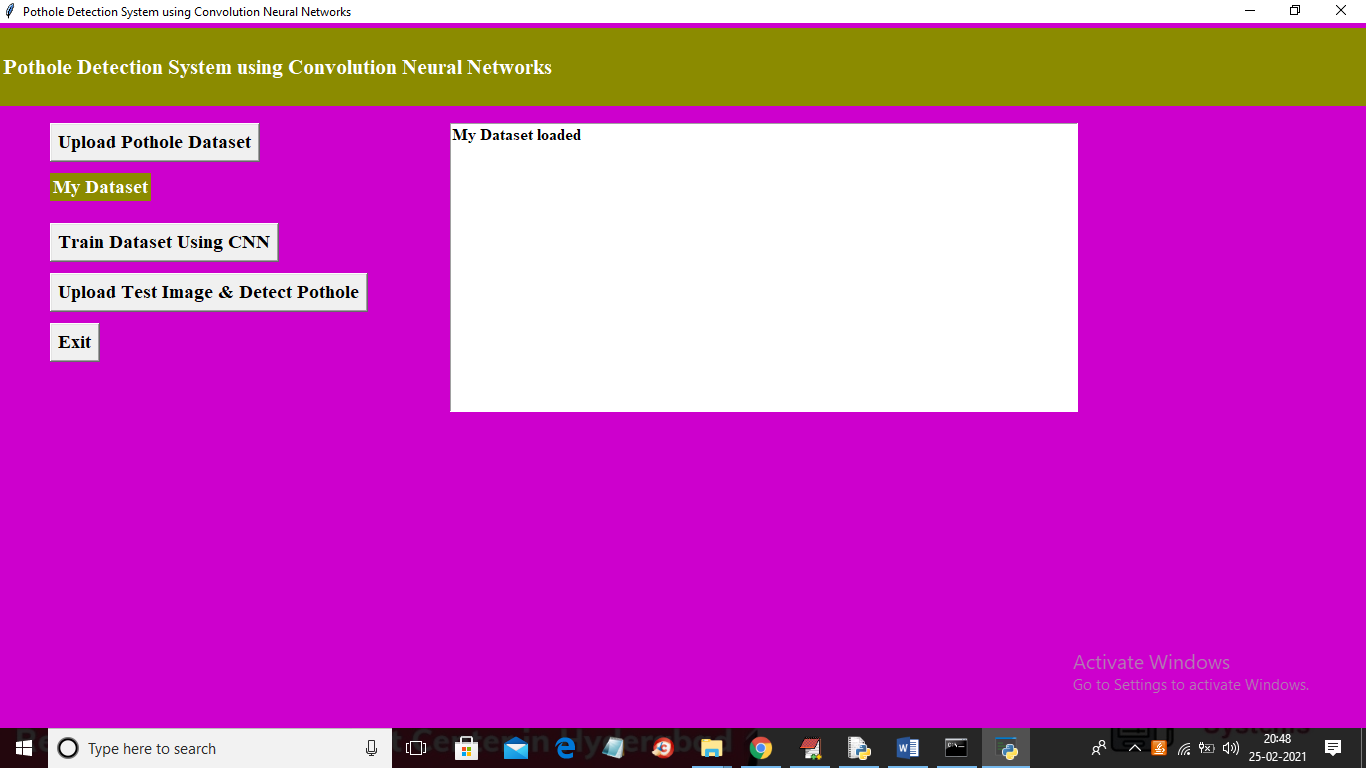
To run project double click on ‘run.bat’ file to get below screen



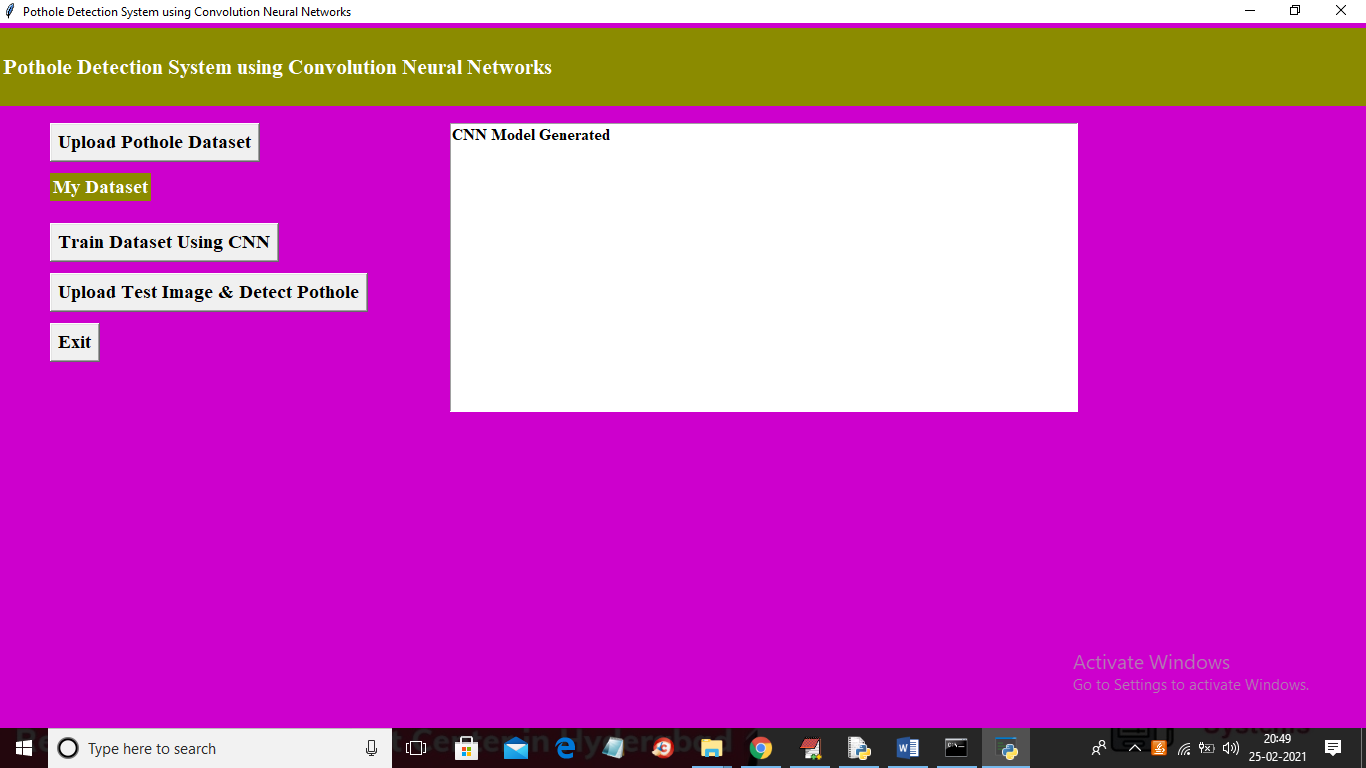
In above screen click on ‘Upload Pothole Dataset’ button and upload dataset



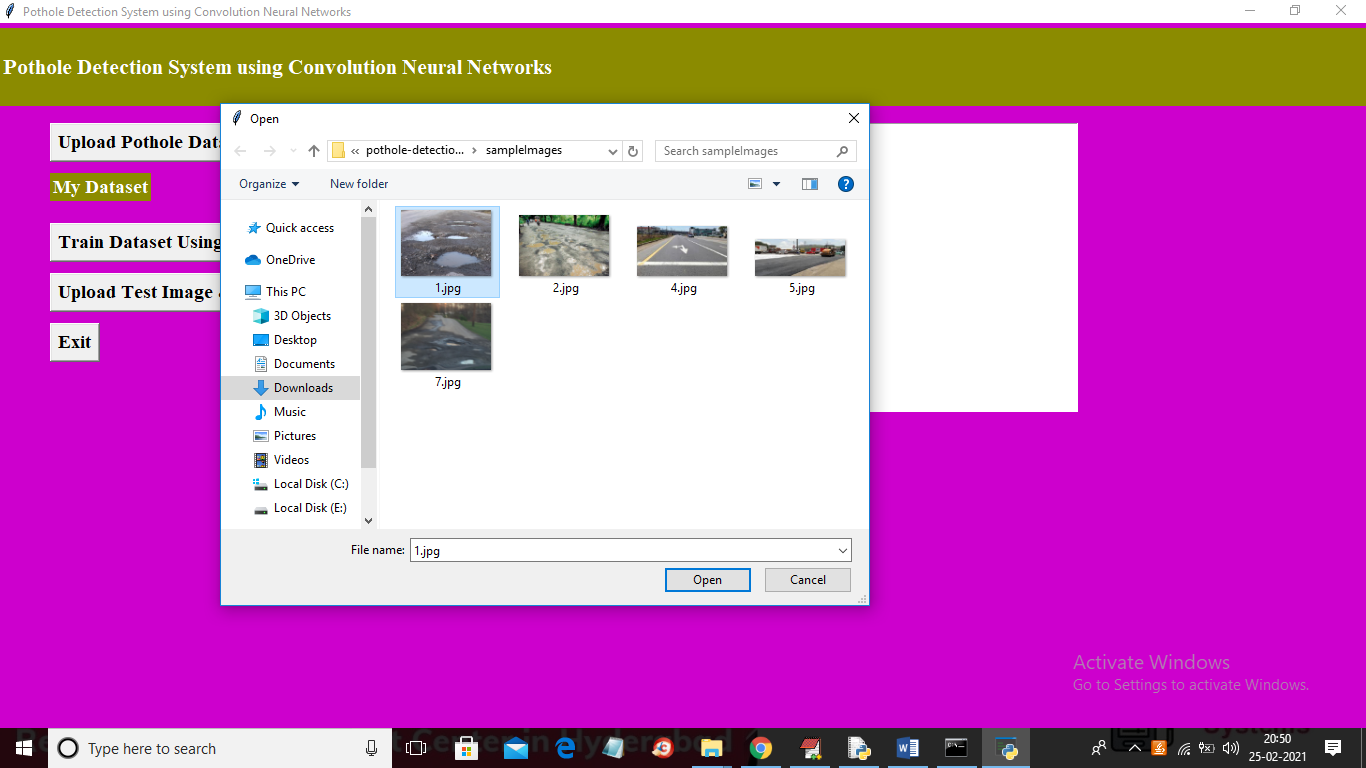
In above screen selecting and uploading ‘My Dataset’ folder and then click on ‘Select Folder’ button to load dataset and to get below screen



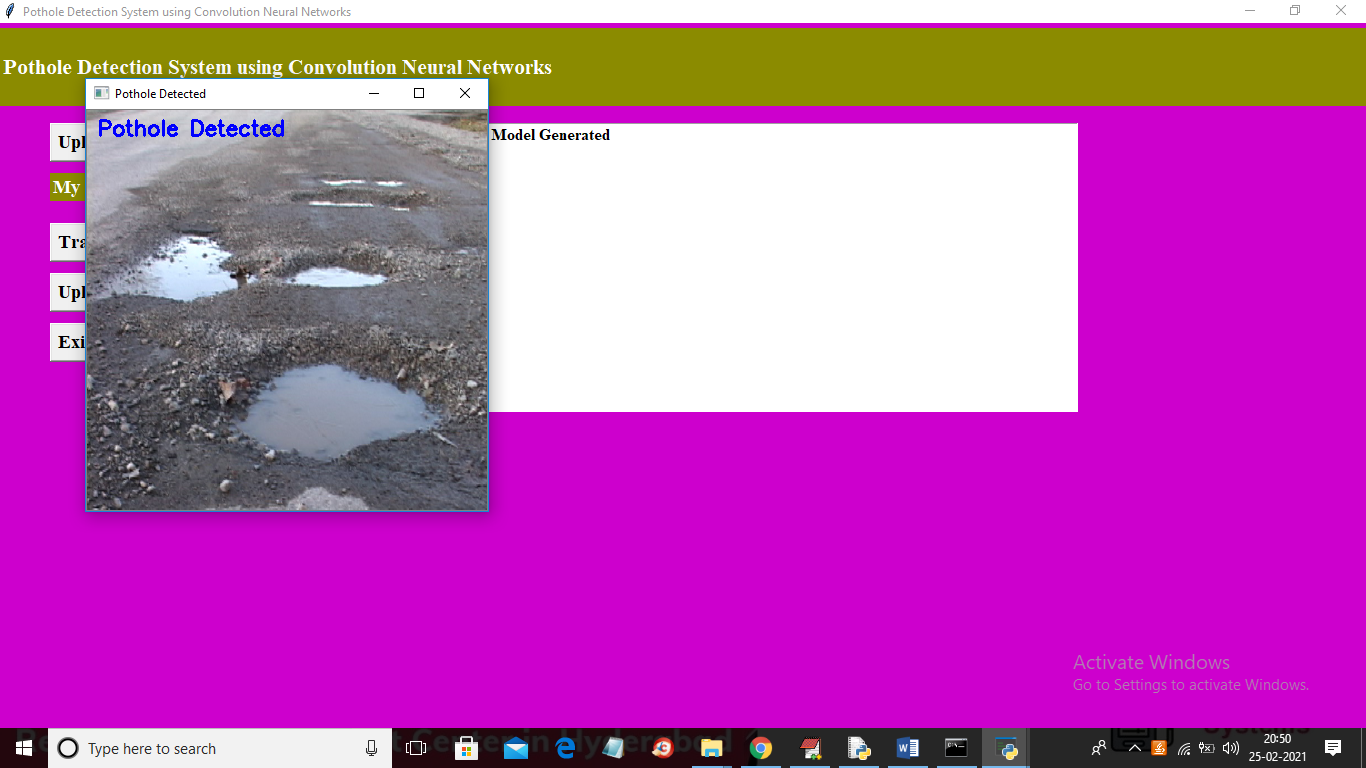
In above screen dataset loaded and now click on ‘Train Dataset using CNN’ button to train all images using CNN and to get classification model



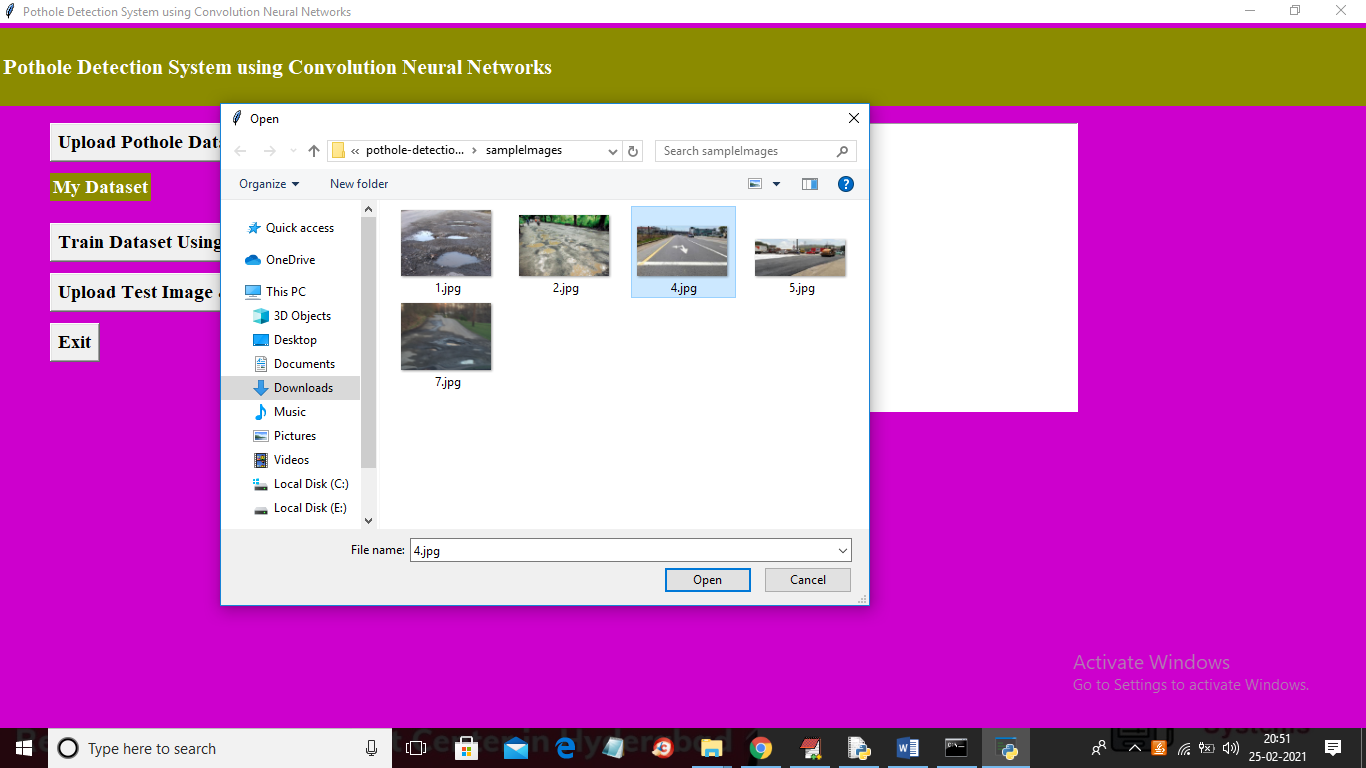
In above screen CNN model generated and now click on ‘Upload Test Image & Detect Pothole’ button to upload test image and to get below result



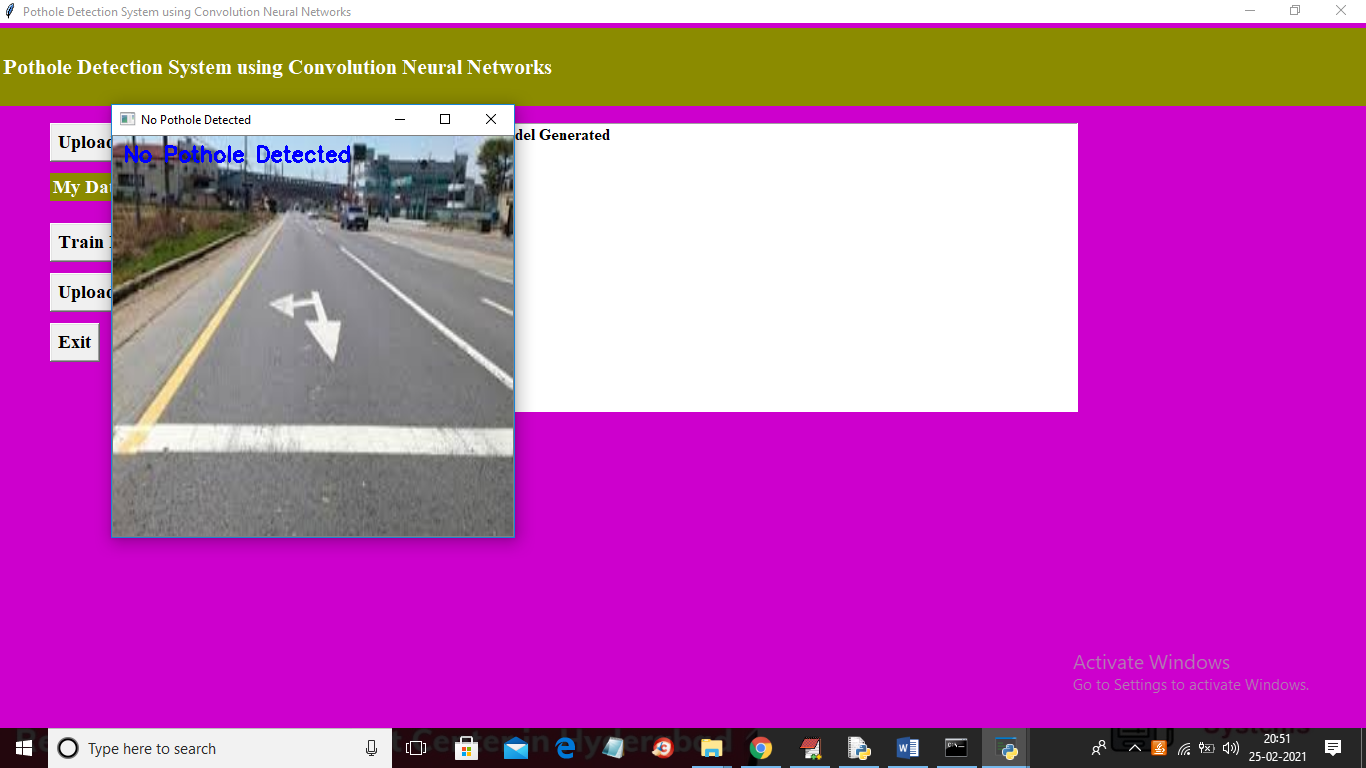
In above screen selecting and uploading ‘1.jpg’ file and then click on ‘Open’ button to get below screen



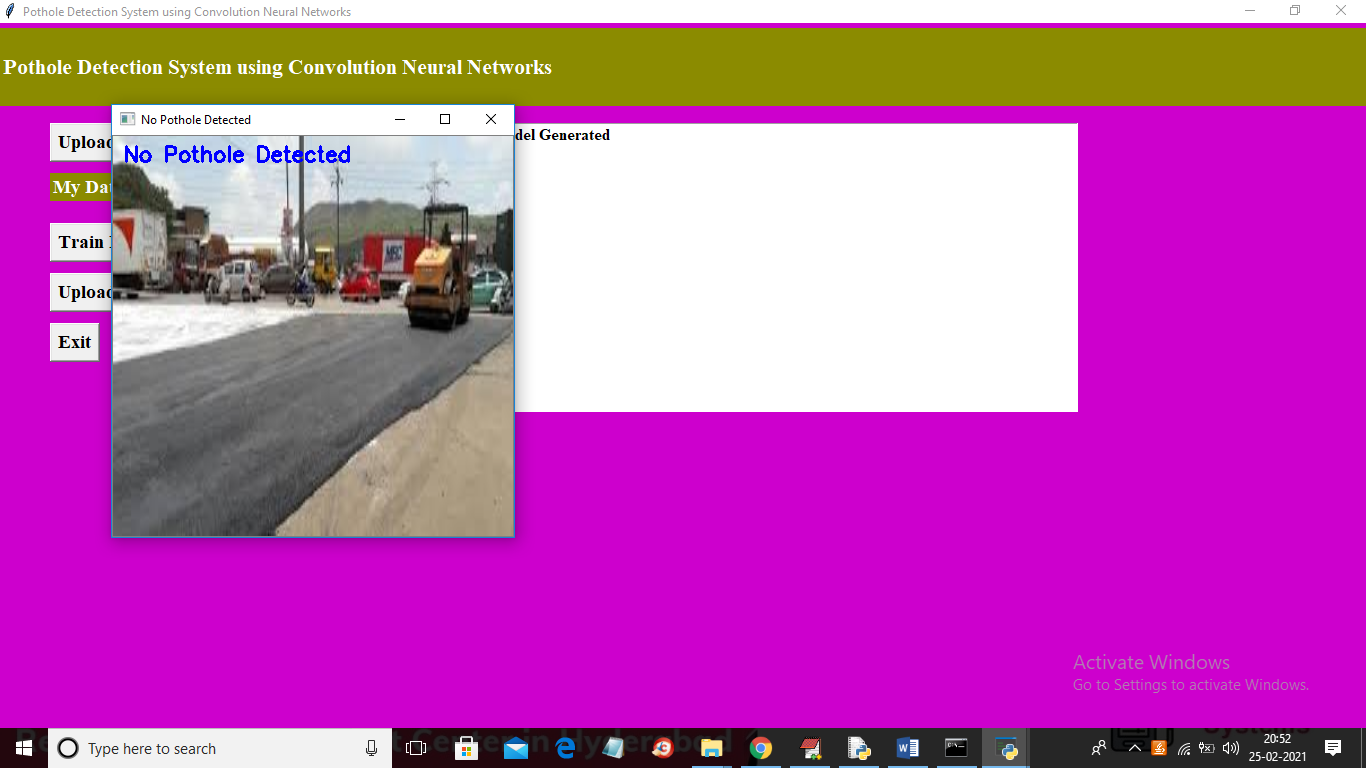
In above screen uploaded image classified as ‘Pothole Detected’ and now test with other image



In above screen selecting and uploading ‘4.jpg’ file and then click on ‘Open’ button to get below result



In above screen on image or image title bar we can see output as ‘No Pothole Detected’



Similarly you can upload any image and get result