Diabetic Retinopathy Detection using ResNet101 and DenseNet121

In this project we are using DenseNet121 and Resnet101 to predict diabetic retinopathy disease. This disease may cause permanent blindness and if not diagnose early and for diagnosis we need heavy resource which may not available in remote areas. So we are developing deep learning models which require patient iris image to detect retinopathy disease.

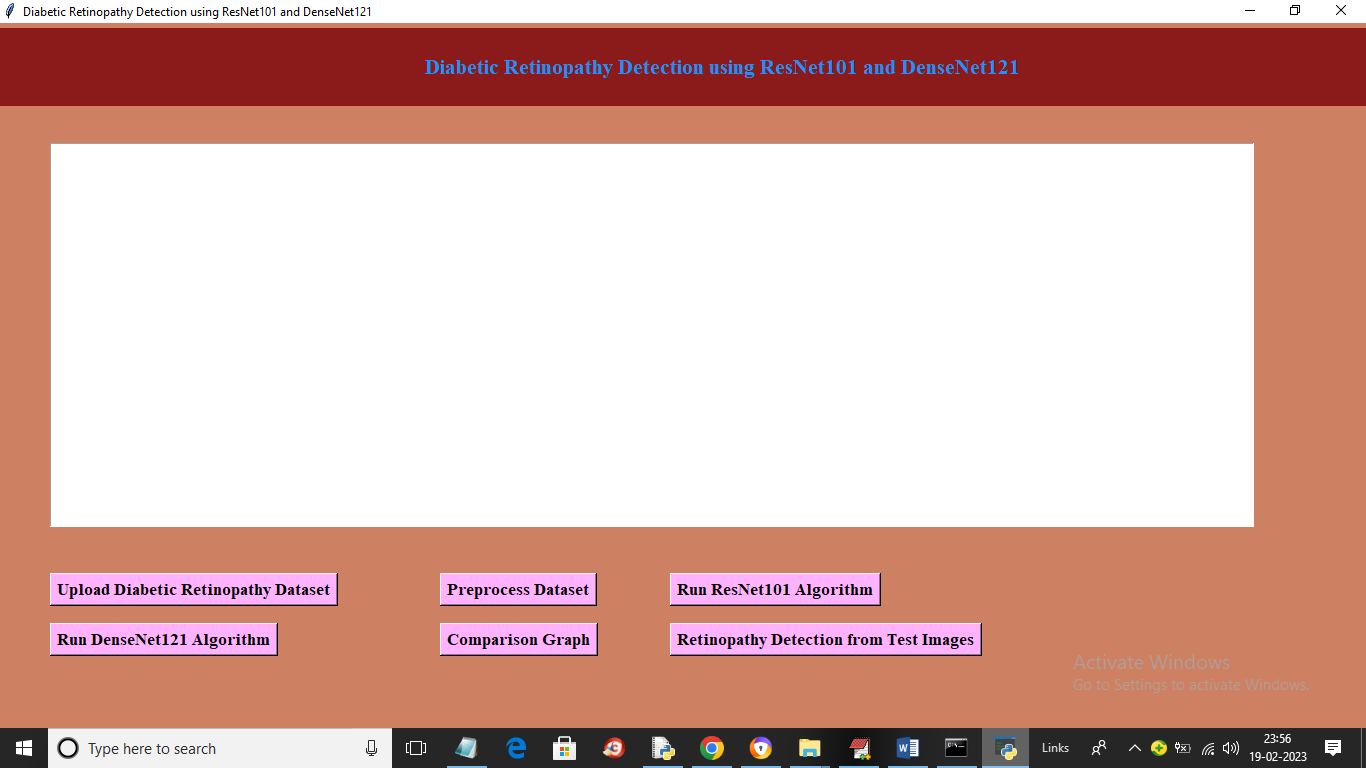
To train both algorithms we have used same KAGGLE dataset given by your and after training model we can upload folder which will contains left and right eye and this algorithm will analyse images to predict retinopathy.

To implement this project we have designed following modules

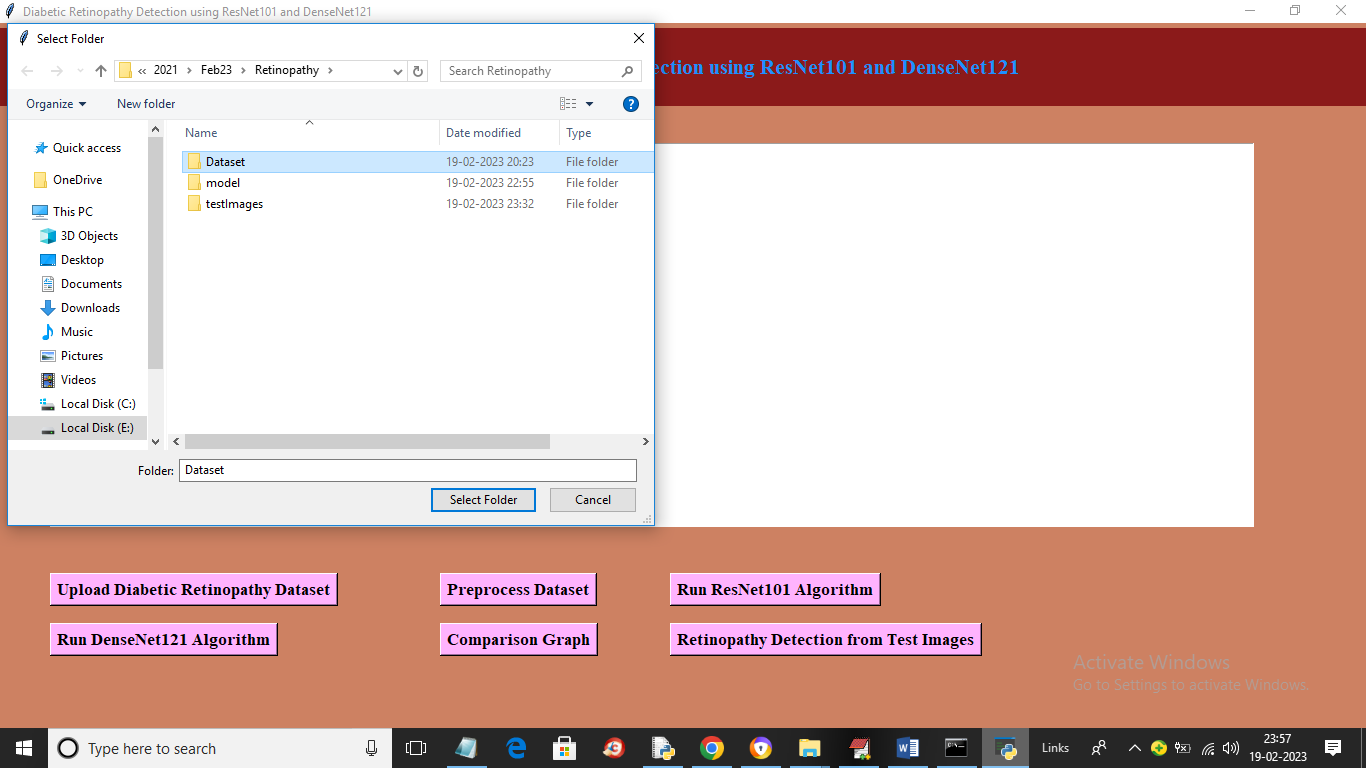
1. Upload Diabetic Retinopathy Dataset: using this module we will upload dataset to application
2. Preprocess Dataset: this module will read each image and then resize all images to equal size, normalize pixel values and then shuffle dataset images and then split all images into train and test where application will be using 80% images for training and 20% images will be applied on trained model to calculate prediction accuracy
3. Run ResNet101 Algorithm: using this module we will input 80% images to ResNet101 to train a model and then model will be applied on 20% test images to calculate prediction accuracy
4. Run DenseNet121 Algorithm: using this module we will input 80% images to DenseNet121 to train a model and then model will be applied on 20% test images to calculate prediction accuracy
5. Comparison Graph: using this module we will plot accuracy graph between both algorithms
6. Retinopathy Detection from Test Images: using this module will upload a folder of 2 images such as left and right eye and then algorithm will analyse those images to predict retinopathy

SCREEN SHOTS

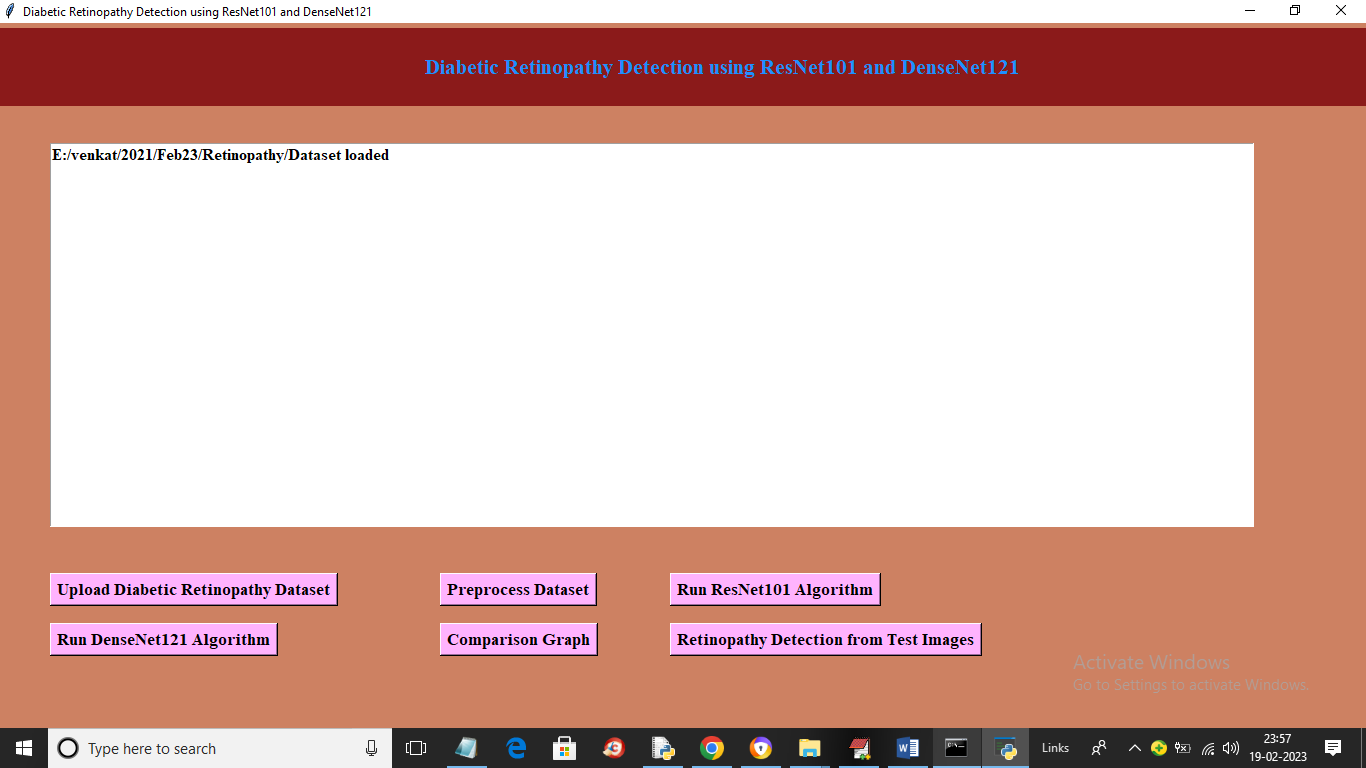
To run project double click on ‘run.bat’ file to get below screen



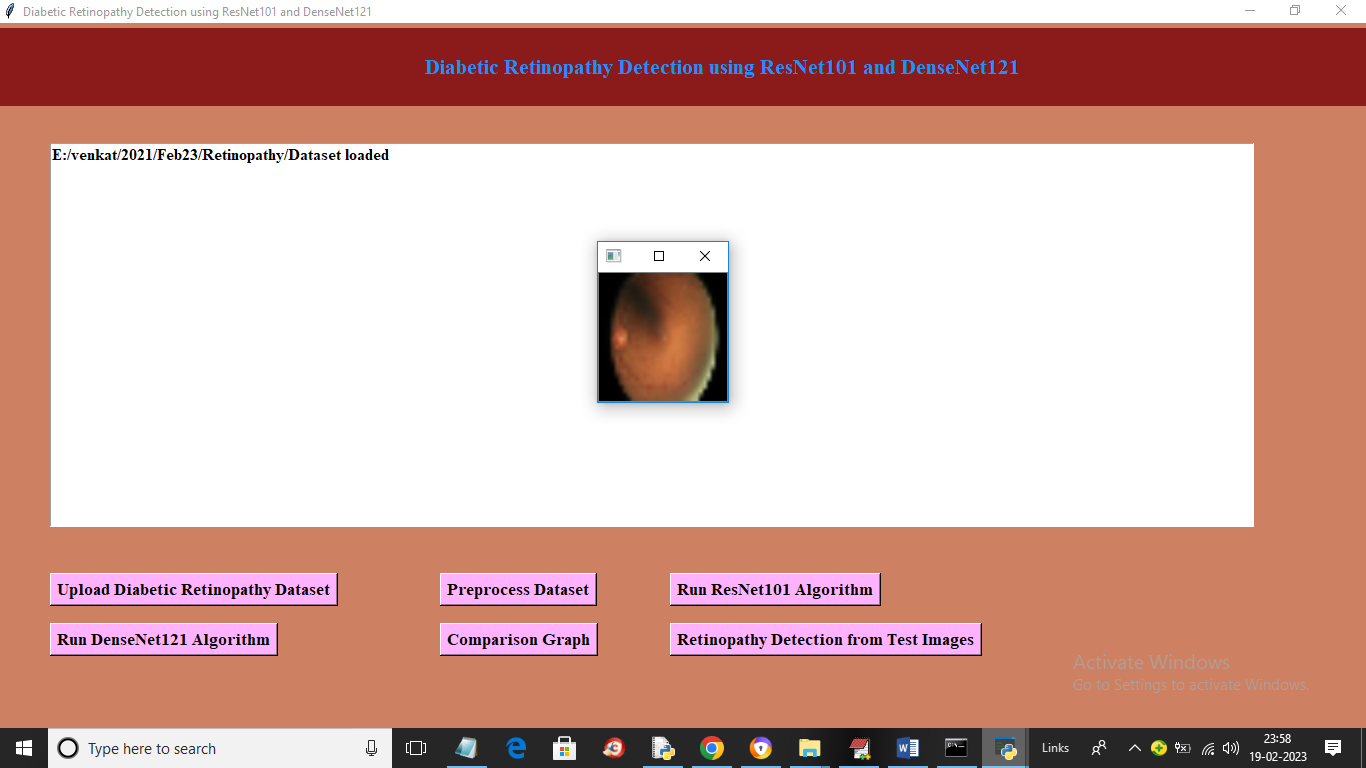
In above screen click on ‘Upload Diabetic Retinopathy Dataset’ button to upload dataset and get below output



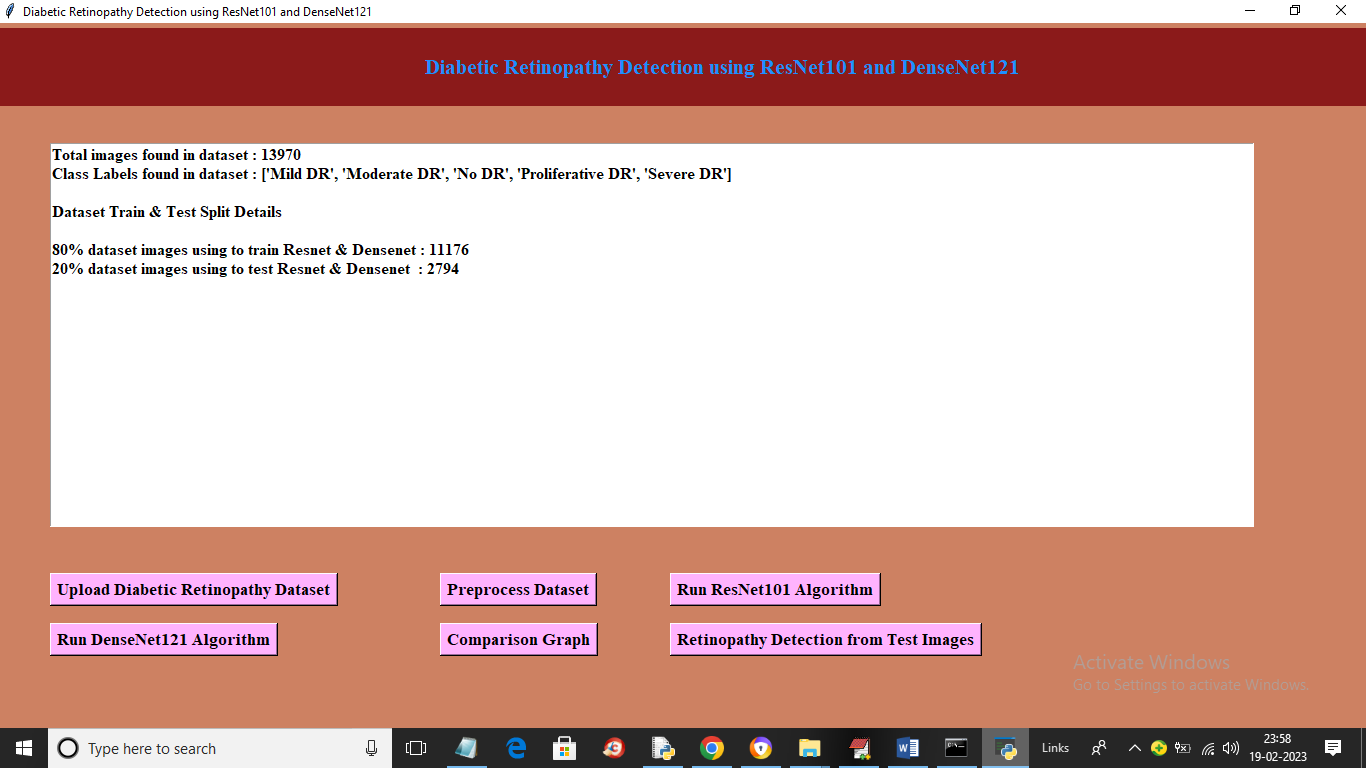
In above screen select and upload ‘Dataset’ folder and then click on ‘Select Folder’ button to load dataset and get below output



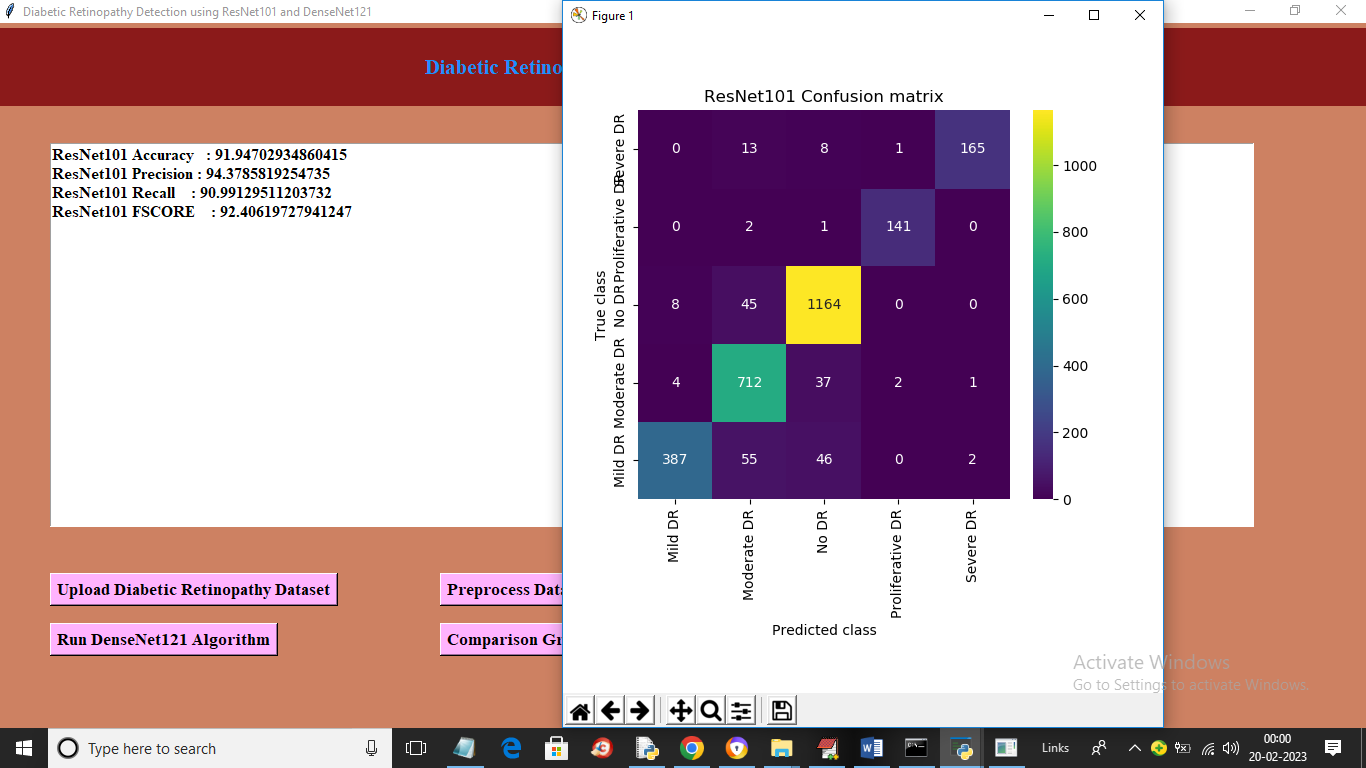
In above screen dataset loaded and now click on “Preprocess Dataset’ button to process images and get below output



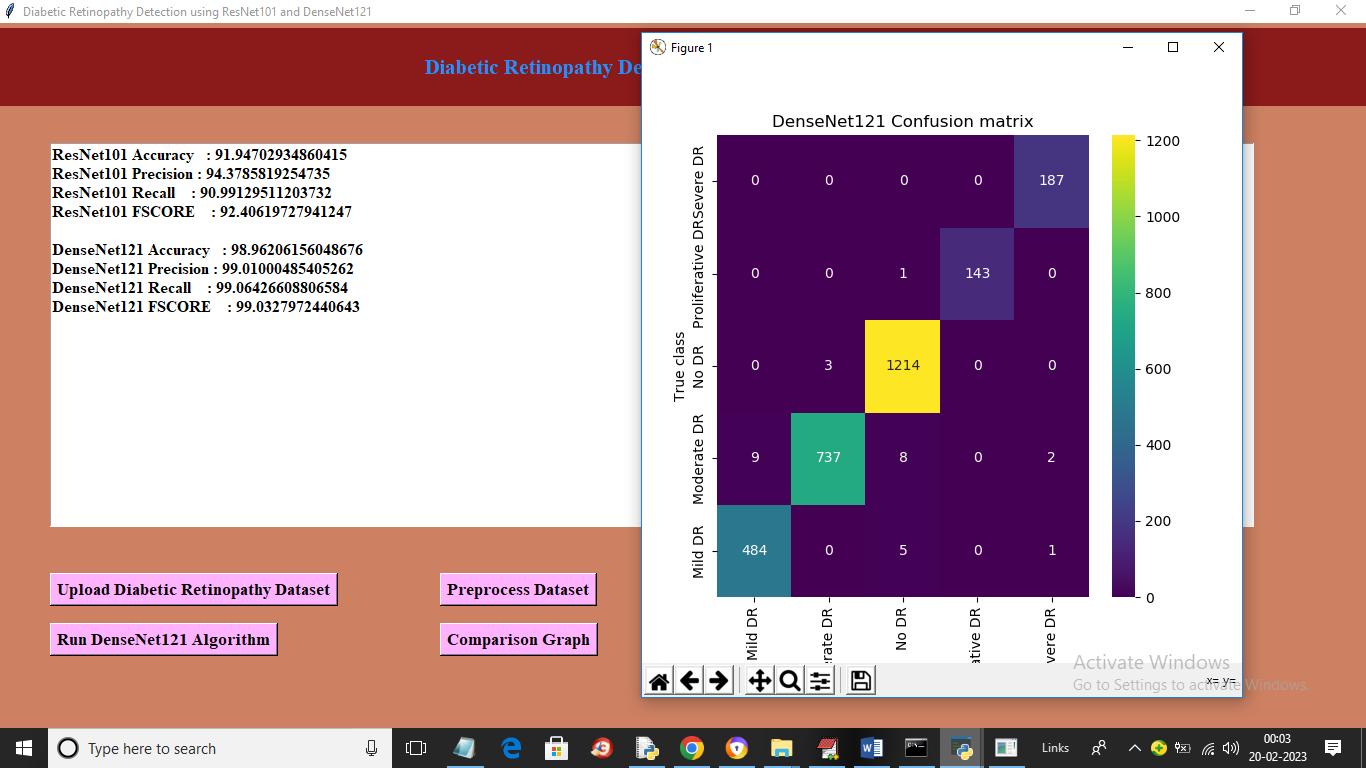
In above screen all images processed and displaying one sample loaded processed image and now close above image to get below output



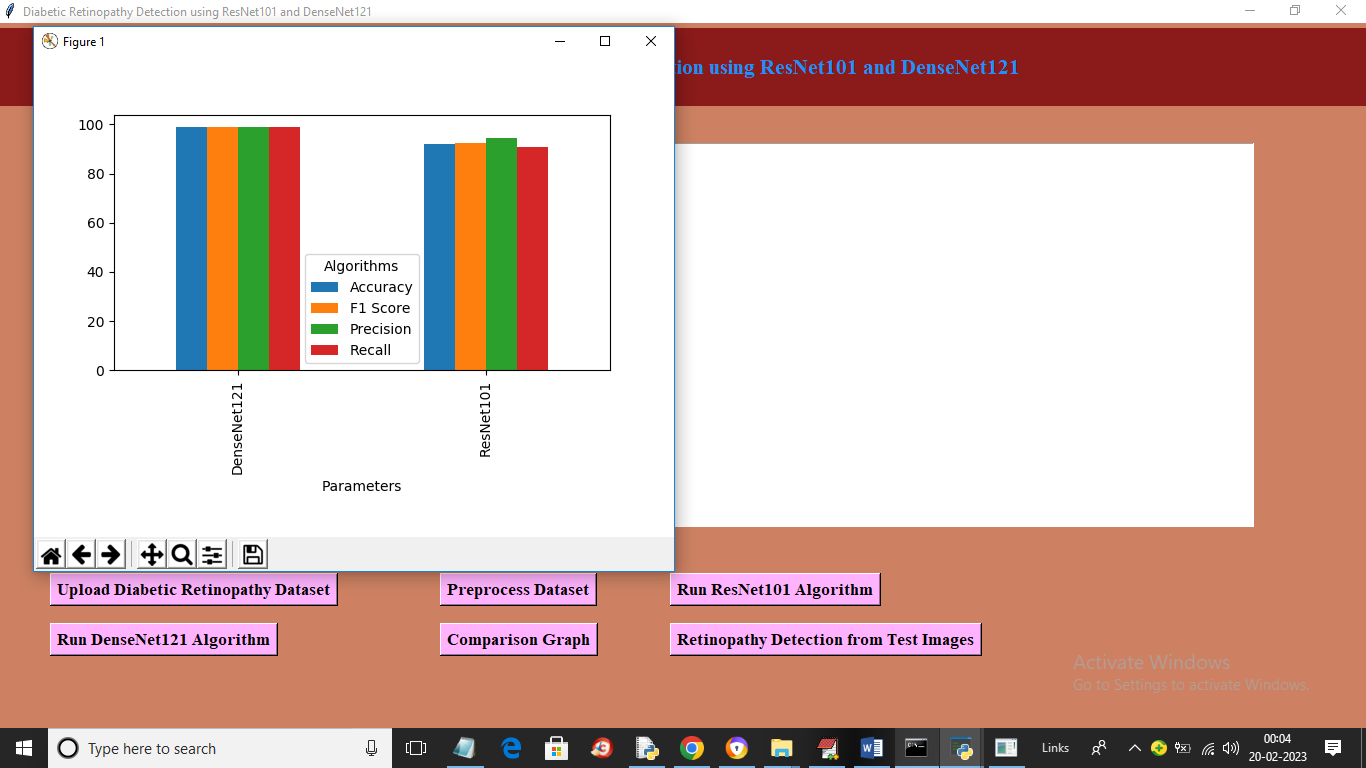
In above screen we can see dataset contains 13970 images and then application using 80% (11176) images for training and 1794 images for testing and now click on ‘Run ResNet101 Algorithm’ button to train Resnet101 and get below output



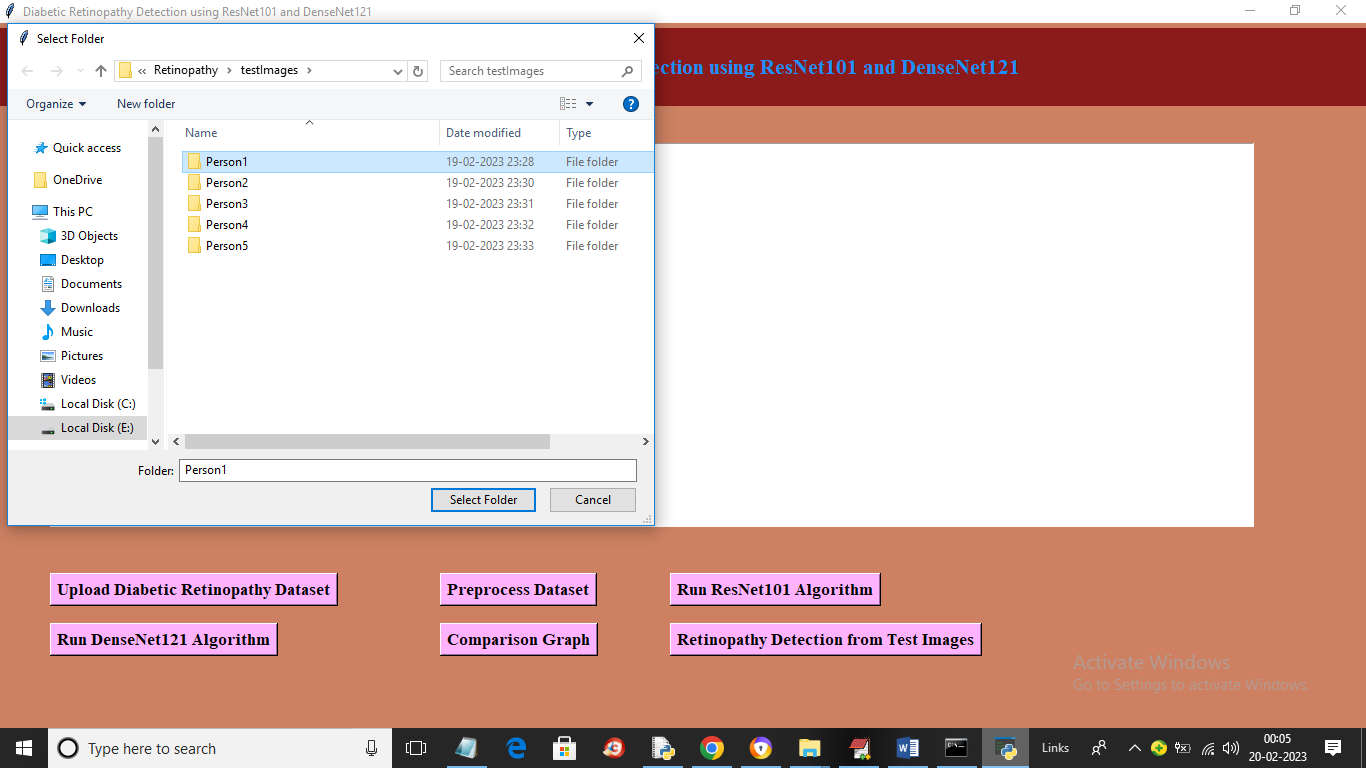
In above screen with Resnet101 we got 91% accuracy and we can see other metric output also and in confusion matrix graph x-axis contains Predicted Labels and y-axis contains True Labels and all different colour boxes in diagnol represents correct prediction count and remaining blue colour boxes contains incorrect prediction count which are very few so Resnet101 can predict retinopathy 90% accurately. Now close above graph and then click on ‘Run Densenet121 Algorithm’ button to train DenseNet121’ and get below output



In above screen with DenseNet121 we got 98% accuracy and now close above graph and then click on ‘Comparison Graph’ button to get below output



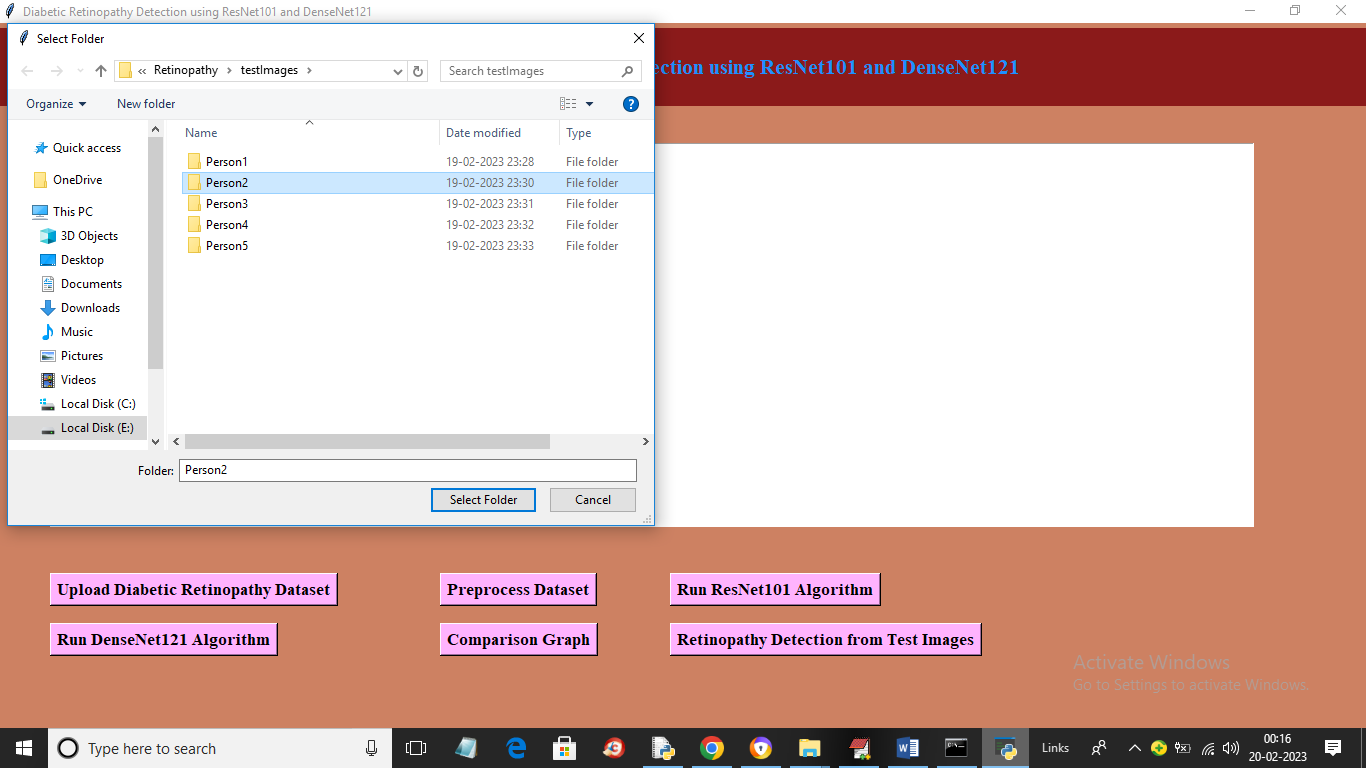
In above graph x-axis represents algorithm names and y-axis represents accuracy and other metric in different colour bar and in both algorithms DensetNet121 got high performance. Now close above graph and then click on ‘Retinopathy Detection from Test Images’ button to upload folder and get below output



In above screen selecting and uploading ‘Person1’ folder and then click on ‘Select Folder’ button to load images and get below prediction output



In above screen for both side images we predicted retinopathy which you can see in red colour text and similarly you can upload other person images and test it



In above screen uploading another person folder to get below output

