Graphical User Interface

Graphical user interface (GUI) for our system mainly aims to give its users the freedom to interact with the functionalities. The GUI is designed to provide the user the suitability of selecting the dataset for the training of the system, then performing some operations of the dataset and then further acting on the results obtained from the training process to accurately classify the new images into several known categories.

1.1 Design Philosophy

The GUI is designed by keeping in mind the different users that will be using the system for the image classification. For example, let the user be an intelligent quality inspector who wants to check the defects that are being developed in the material during their processing. Another case would be of the manufacturing engineer who observes and captures some new kind of defects and has maintained a database of that defect and wants the system to learn about this new defect. Therefore, it is important for different users to understand what kind of cases they would act upon and would like to know how the GUI and the model will help them in these cases.

Thus, considering the above requirements, the GUI was designed in such a way that it should have several windows allowing the uploading of dataset to be trained, validating accuracy window, capturing new image and selecting the training model and showing the defect for the new image on the final window.

1.2 System Architecture

In order to understand how the system works, what happens within the system and interaction of the system with the outside environment, a system architecture diagram was developed in Figure 1. This diagram can help in understanding how to organize the system and what parameters should the GUIs have.

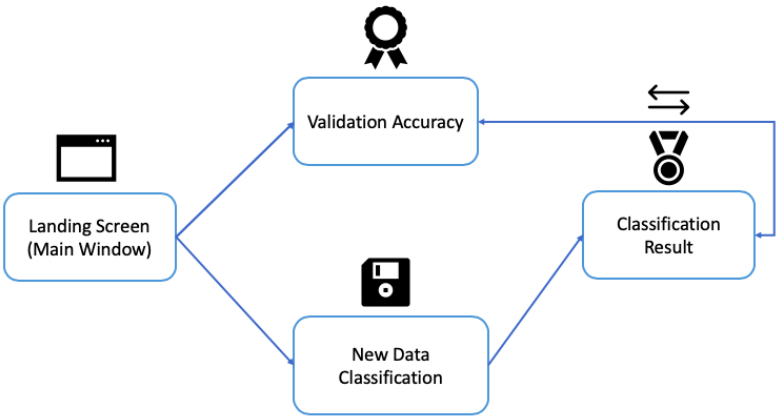

Figure 1 system architecture diagram

The diagram below describes the organization and distribution of the system. The final idea is to have a main window that will open to the user and let them feed the dataset (images) and select the training model. The main window will break into two more windows, the first window will show the validation accuracy window which shows the testing accuracy of the system. The second window will

let another user to feed a new kind of data (image) and let them select the training model to classify the image. Another window will have the result of the classification and will let the new user to inspect the type of defect for quality inspection.

1.3 The GUIs

A GUI used for training by the manufacturing engineer is needed. This GUI needs to allow the manufacturing engineer to be able to select defect images for training, select which models to use, and see the classified defect and the associated accuracy. Next, another GUI used by the operator is needed. This GUI needs to allow the operator to select an image or image set for classification, select which type of model to be used for classification, run the analysis, view the results, and export the data. Two GUIs are designed, one for training for the manufacturing engineer and one for testing for the end user, which is the operator.

1.3.1 Training GUI

a) Training Interface

The first task of opening and acting on the GUI will be carried out by the manufacturing engineer. The first window and the options available for the engineer are shown in Figure 2.

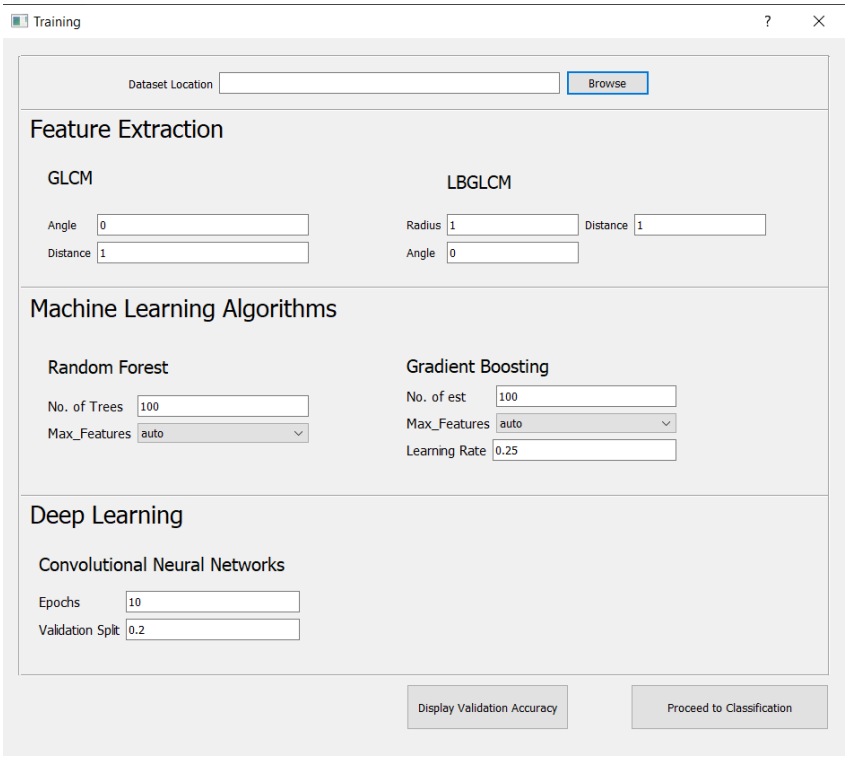

Figure 2 Training window

b) Working process

The training interface is the window used for selection of parameters required for training algorithms. Firstly, the location of the image dataset should be chosen. Secondly, each feature extraction algorithm, machine learning or deep learning algorithm has specific parameters which are

open for the engineer to fill values. After inputting the desired parameters, a click on “Display validation accuracy” button will train all algorithms based on the selected dataset and parameter values. This will take a few minutes. Then the system will show the training results and be ready to retrain.

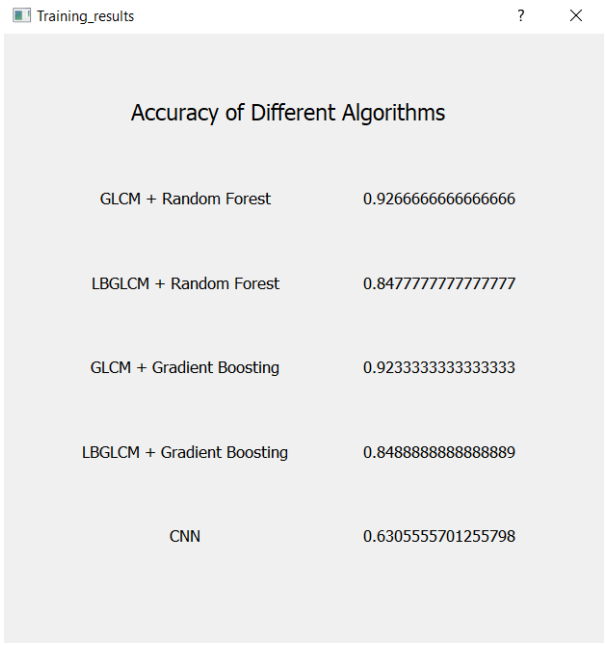

Figure 3 Training results

1.3.2 Testing GUI

a) Testing Interface

The testing interface is the window used for selection of parameters required for the testing algorithms. The parameters include the defect image to be classified and the type of algorithm to be used.

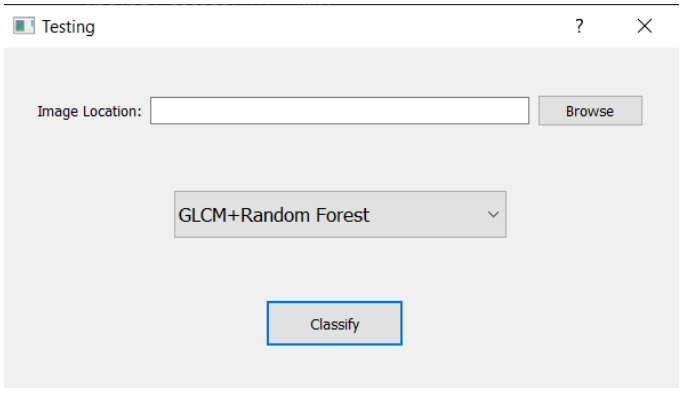

Figure 4 Testing window

b) Working process

After training all the algorithms, the inspection operator can classify the defect of one image which is of interest. The image is not in the training and validating dataset. Some testing images are located in the “Prediction” folder. For classifying the image, the image location should be specified. And the

classifier should be determined. Another GUI will be activated after clicking the “classify” button which will show results of the given image. The results window will display the type of defect in the image along with the image and model used.

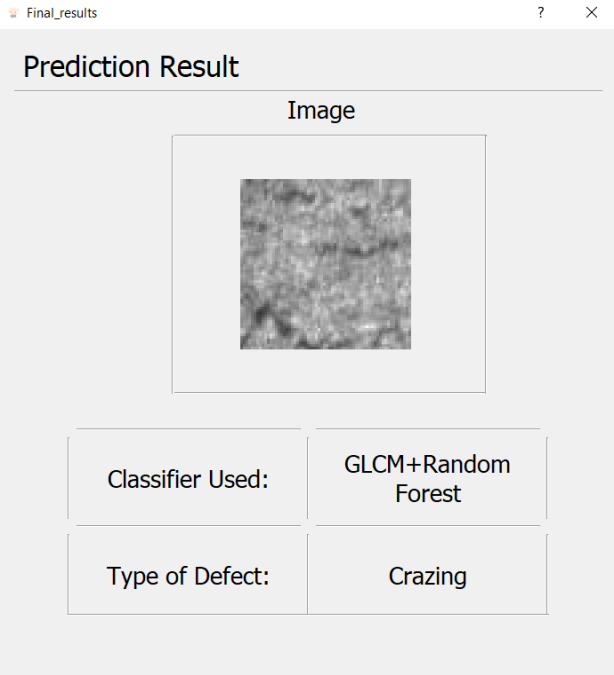

Figure 5 The prediction result