

DATASET:

We have assembled the largest satellite image collection. It's called the "Weather Phenomenon Database" (WEAPD). The collection of satellite images consists of 5636 images. The collection is divided into four categories. The satellite image classification dataset, RSI-CB256, was the one we used. Four distinct classifications, a combination of Sensors and Google Map pictures, are included in this collection. With this dataset, we want to increase weather image classification accuracy and efficiency.

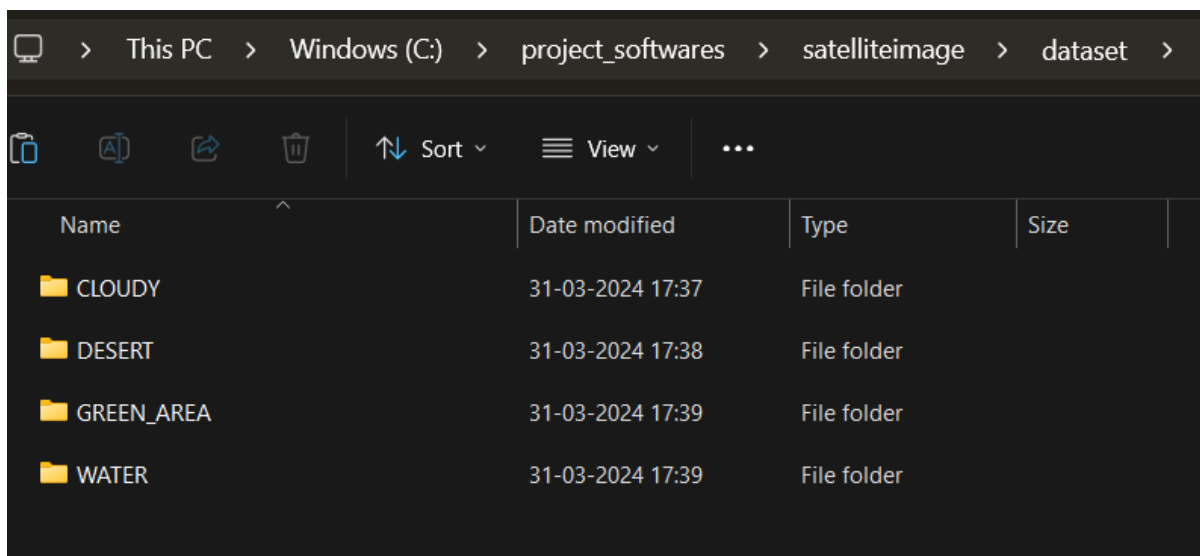


Fig-1: Dataset

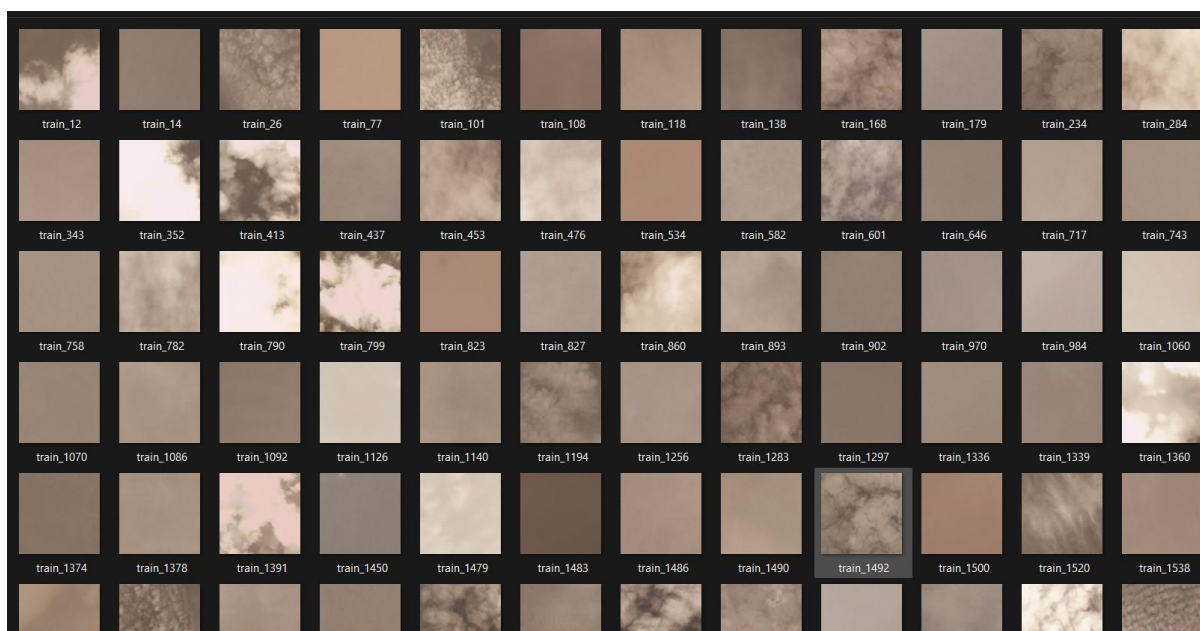


Fig 2: Cloud image dataset

Weather Prediction using CNN project

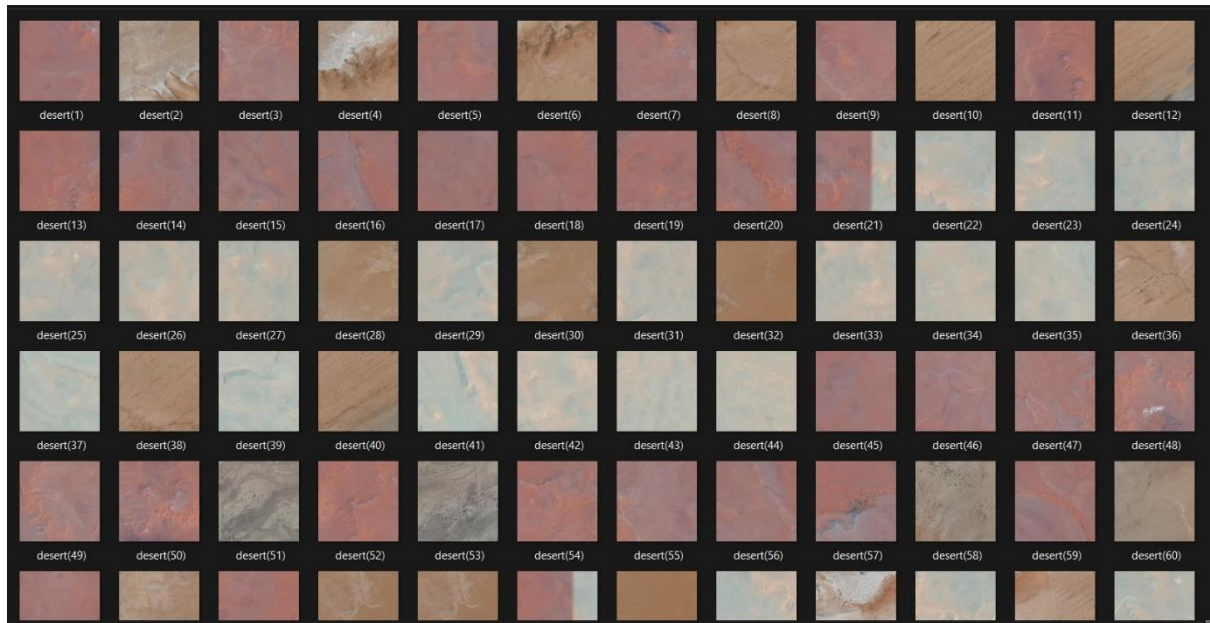


Fig 3: Desert dataset

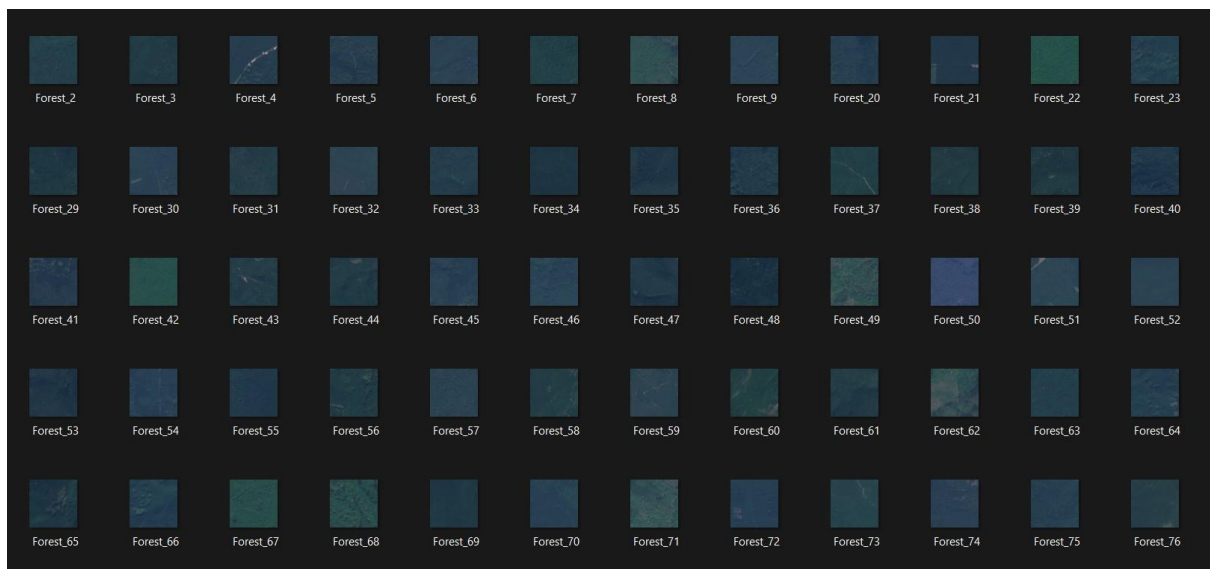


Fig 4: Green Area

Weather Prediction using CNN project

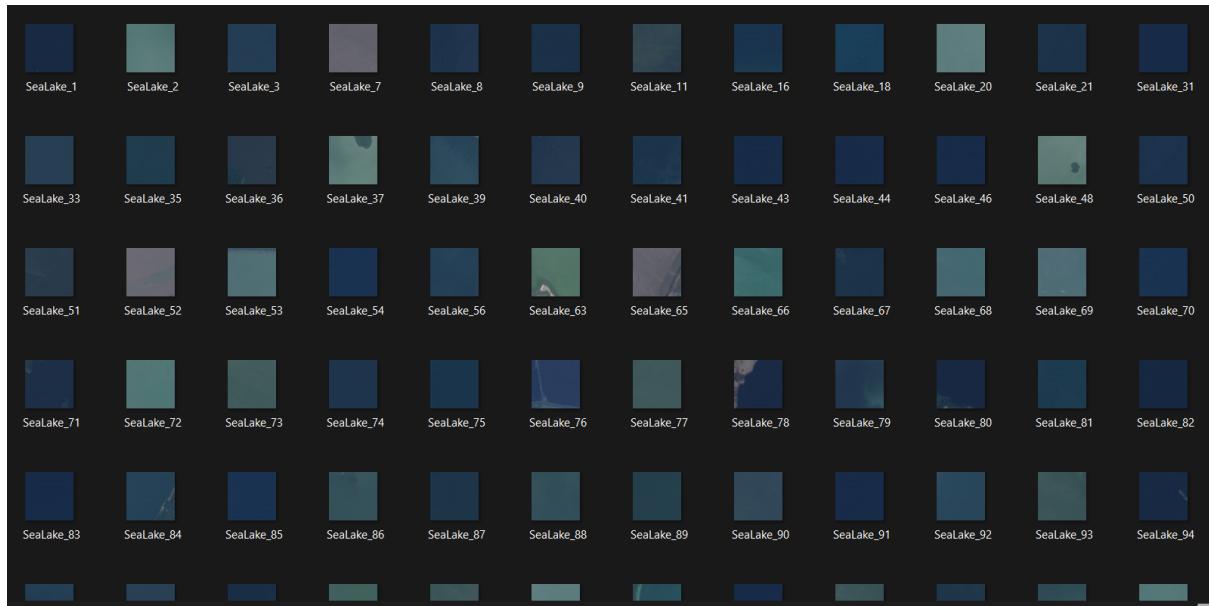


Fig 5: Water

The trained model will preprocess the dataset and give the output based on the following classification. They are “Cloud”, “Desert”, “Green area” and “Water”. Figure 1 displays the model's dataset. Figure 2 displays the Cloud dataset. The Desert dataset is displayed in Figure 3. The Green area dataset is displayed in Figure 4. Figure 5 displays the Water dataset.

EVALUATION GRAPHS

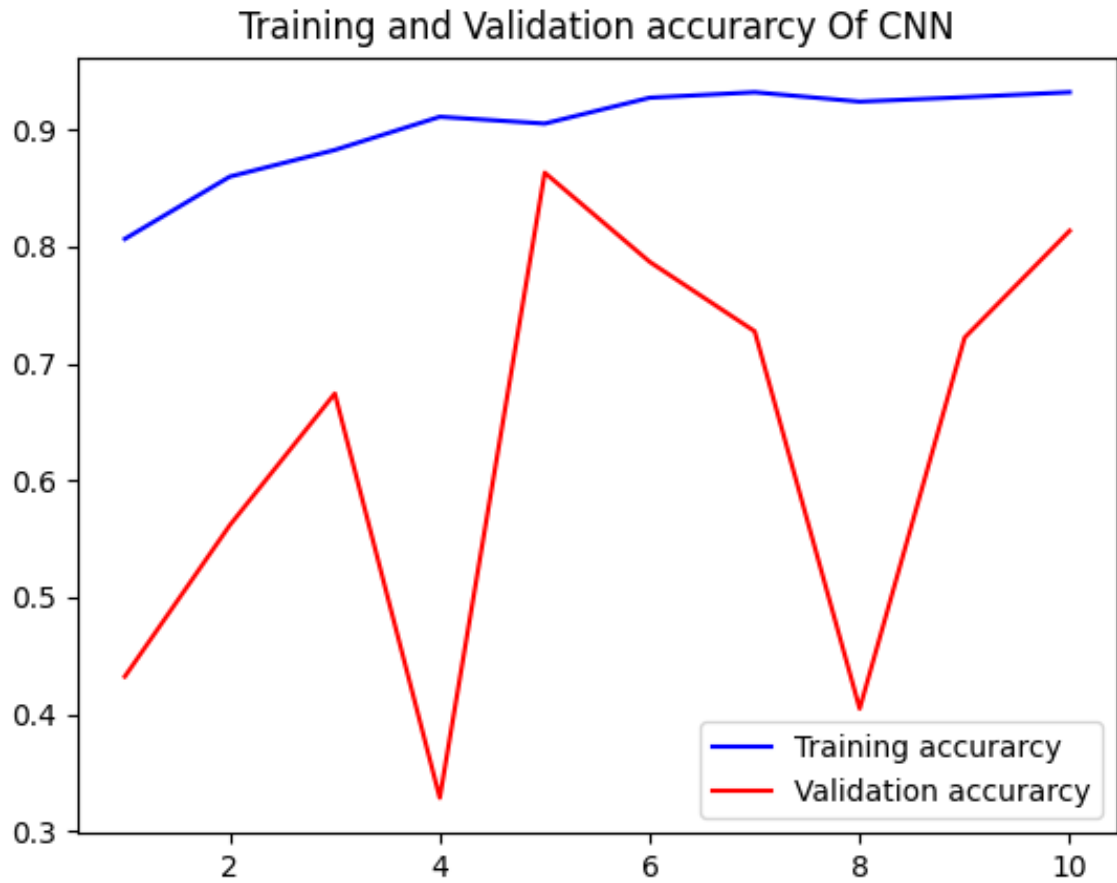


Fig :cnn_accuracy

The above figure shows the Training and Validation Accuracy of CNN. This was taken from the analysis of CNN accuracy using matlab libraries. This graph shows the analysis of the model using CNN.

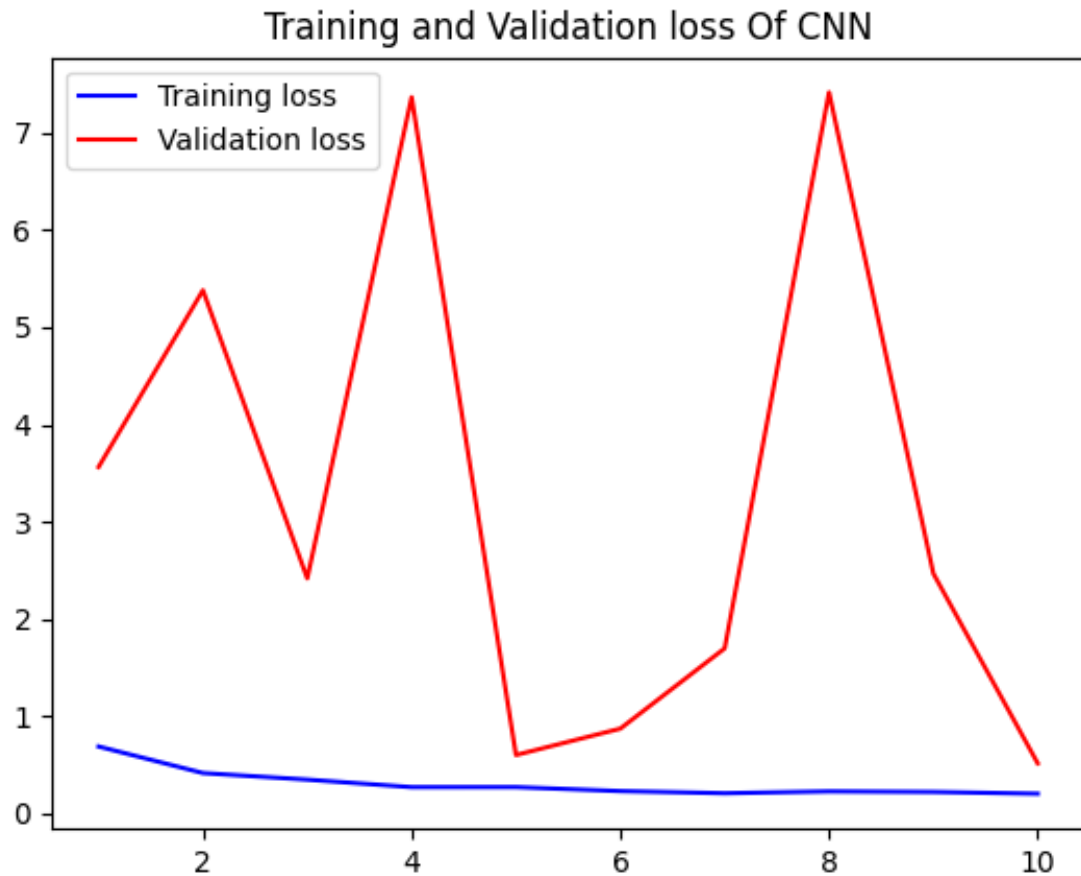


Fig : cnn_loss

The above figure shows the Training and Validation Loss of CNN. This was taken from the analysis of CNN loss using matlab libraries in python coding. This graph shows the analysis of the model using CNN.

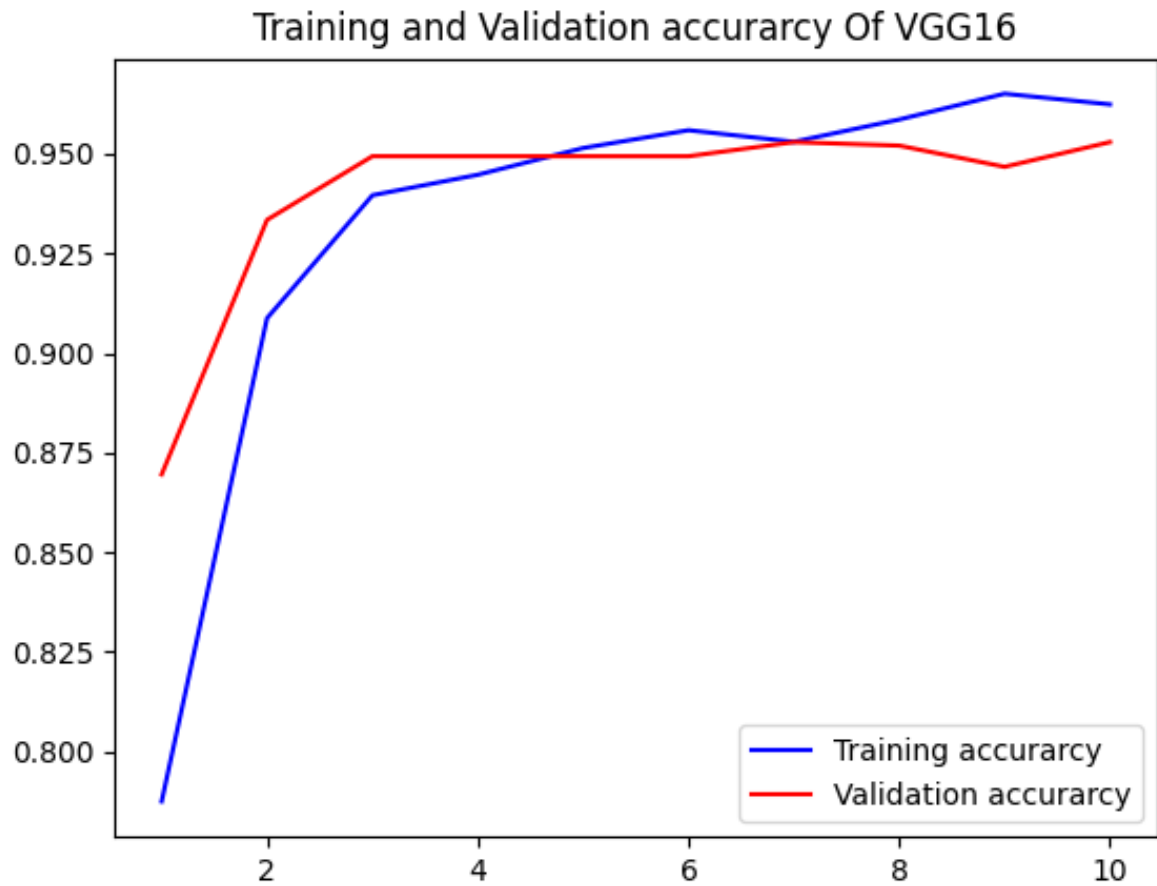


Fig : vgg16_accuracy

The above figure shows the Training and Validation Accuracy of VGG16. This was taken from the analysis of VGG16 accuracy using matlab libraries in python coding. This graph shows the analysis of the model using VGG16. From the figure we can say that the accuracy using VGG16 is very high and it gives more accurate results.

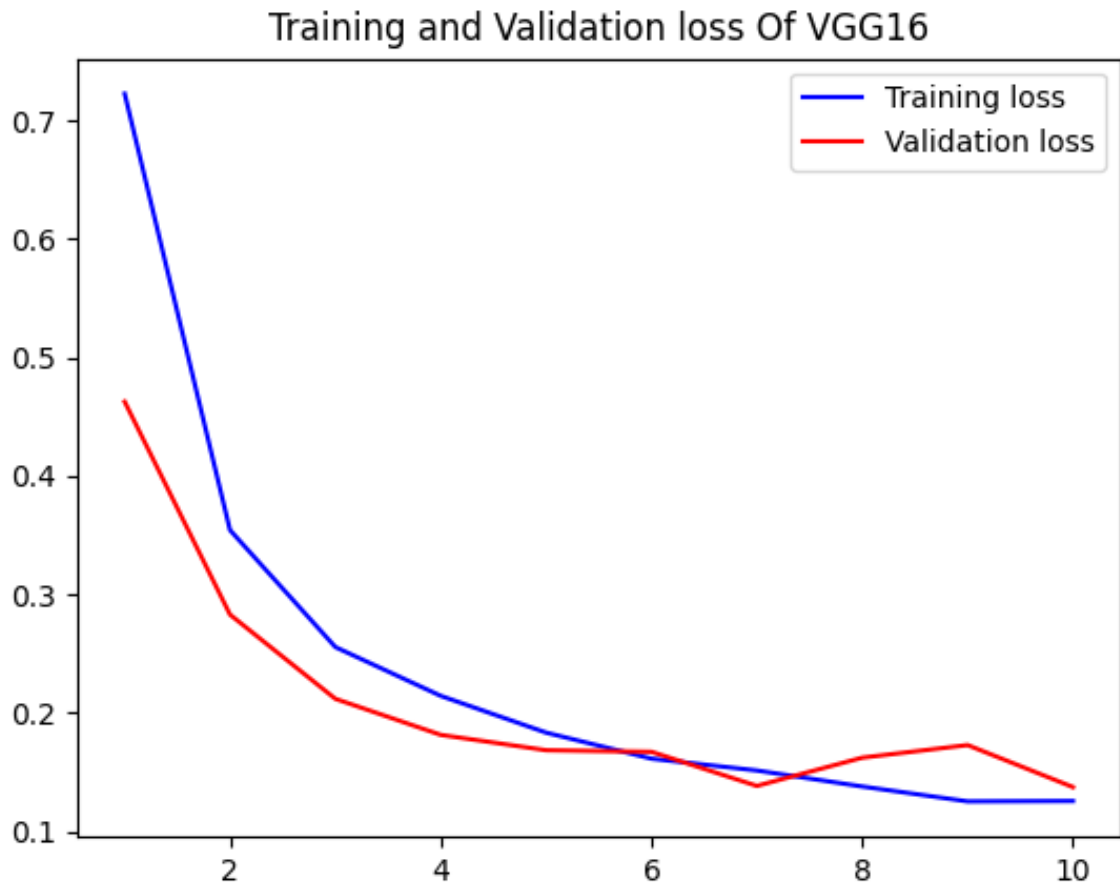


Fig : vgg16_loss

The above figure shows the Training and Validation Loss of VGG16. This was taken from the analysis of VGG16 Loss using matlab libraries in python coding. This graph shows the analysis of the model using VGG16. From the figure we can say that the loss using VGG16 is very less and it gives more accurate results.

RESULT:



Fig : Result 1

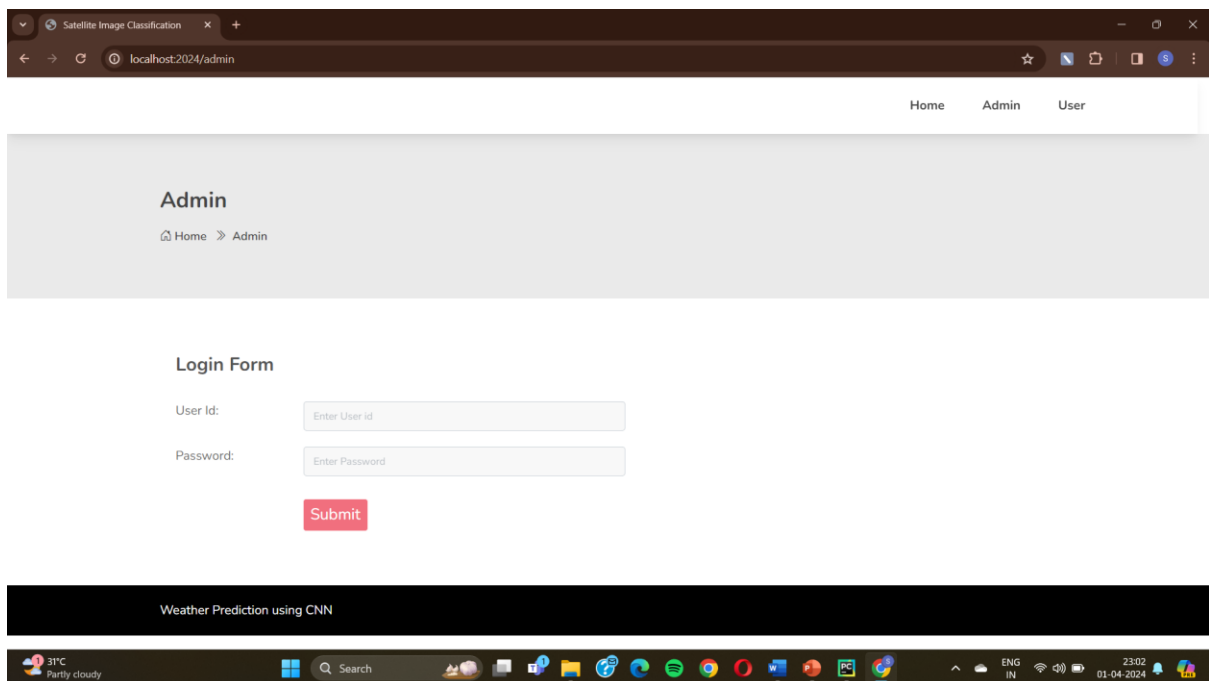


Fig : Admin Interface

Weather Prediction using CNN project

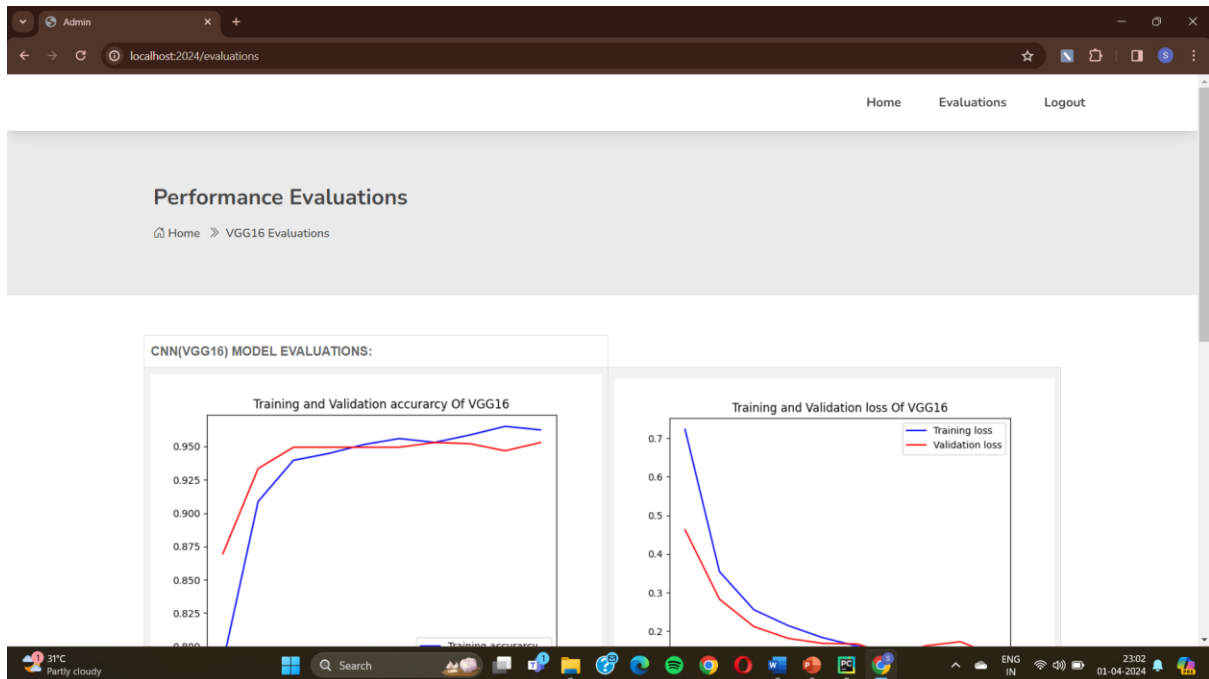


Fig : Performance Evaluation

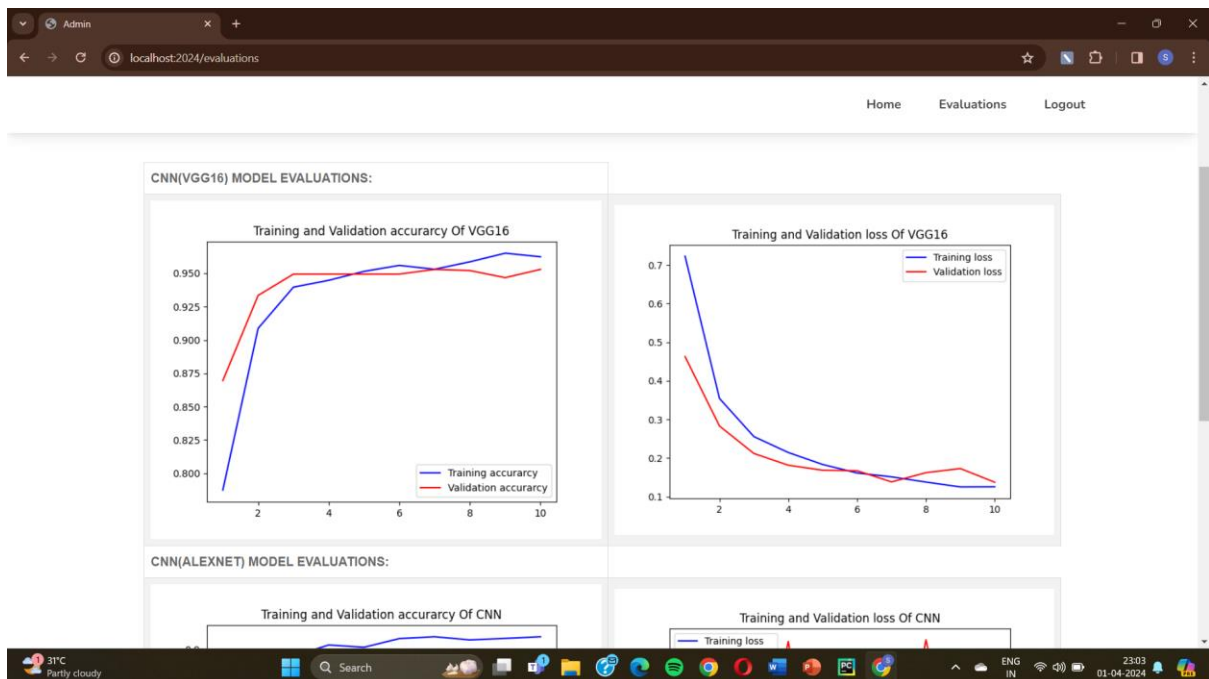


Fig : VGG16 Model Evaluation

Weather Prediction using CNN project

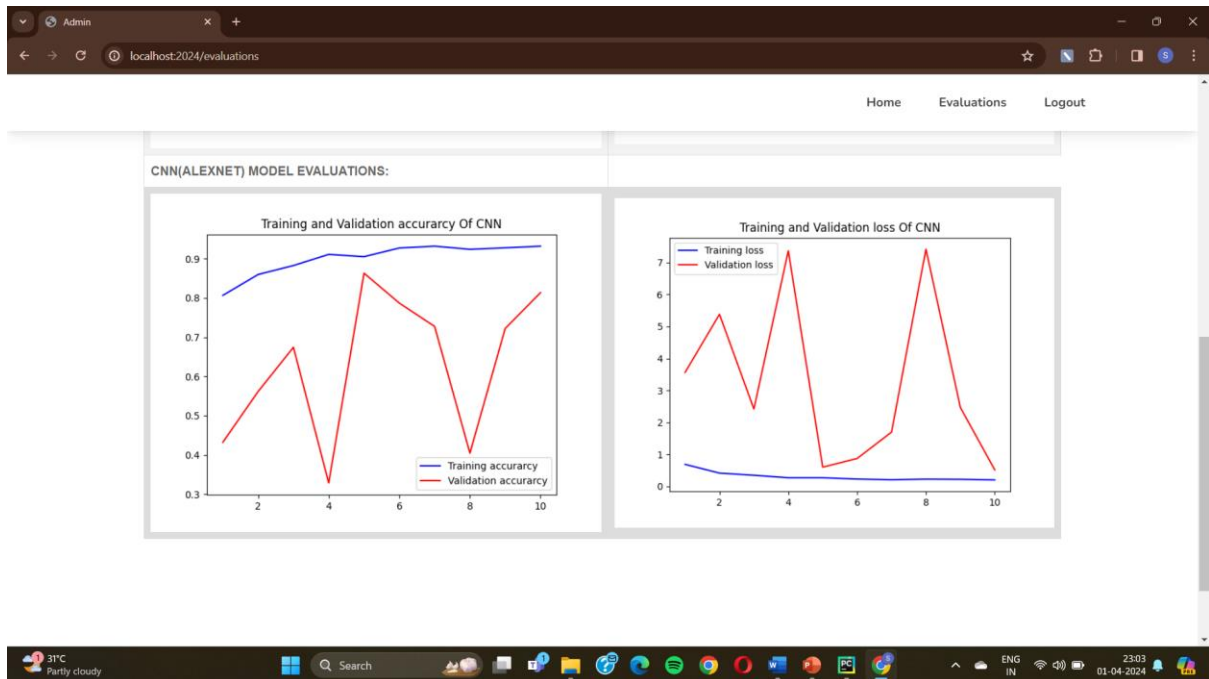


Fig : CNN Model Evaluation

User

Home Admin User

Home > User Login

Login Form

User Id:

Password:

[Register Here](#)

Weather Prediction using CNN

Fig : User Interface

Weather Prediction using CNN project

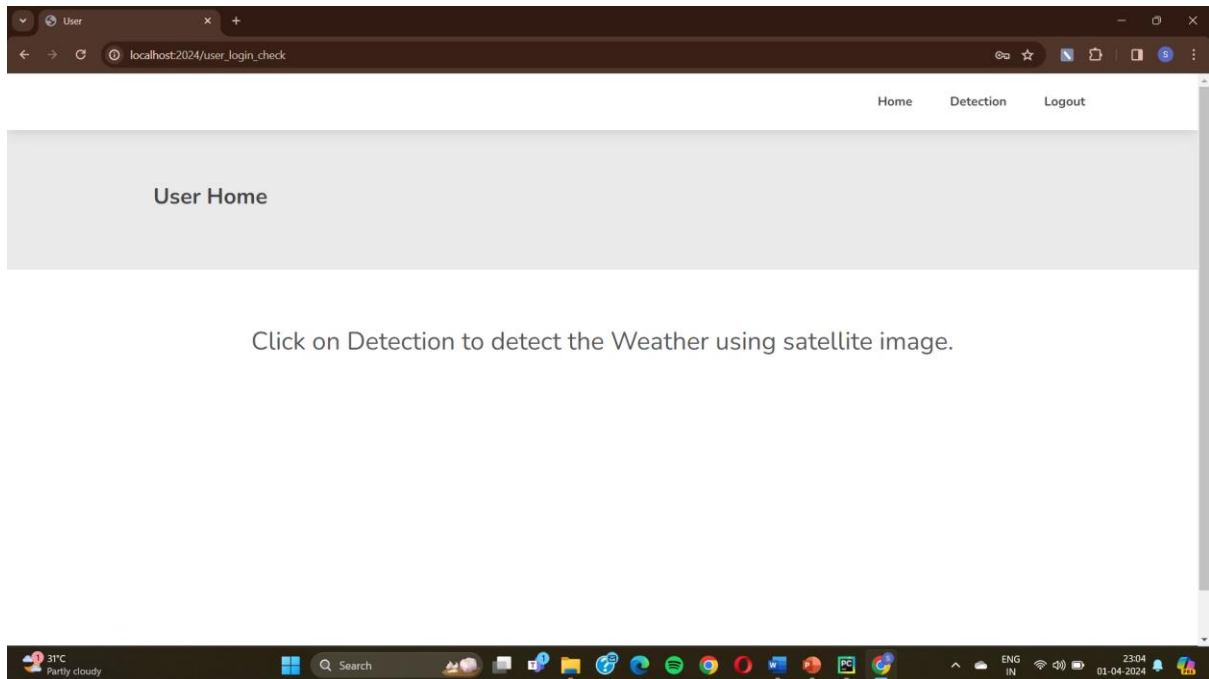


Fig : User Home

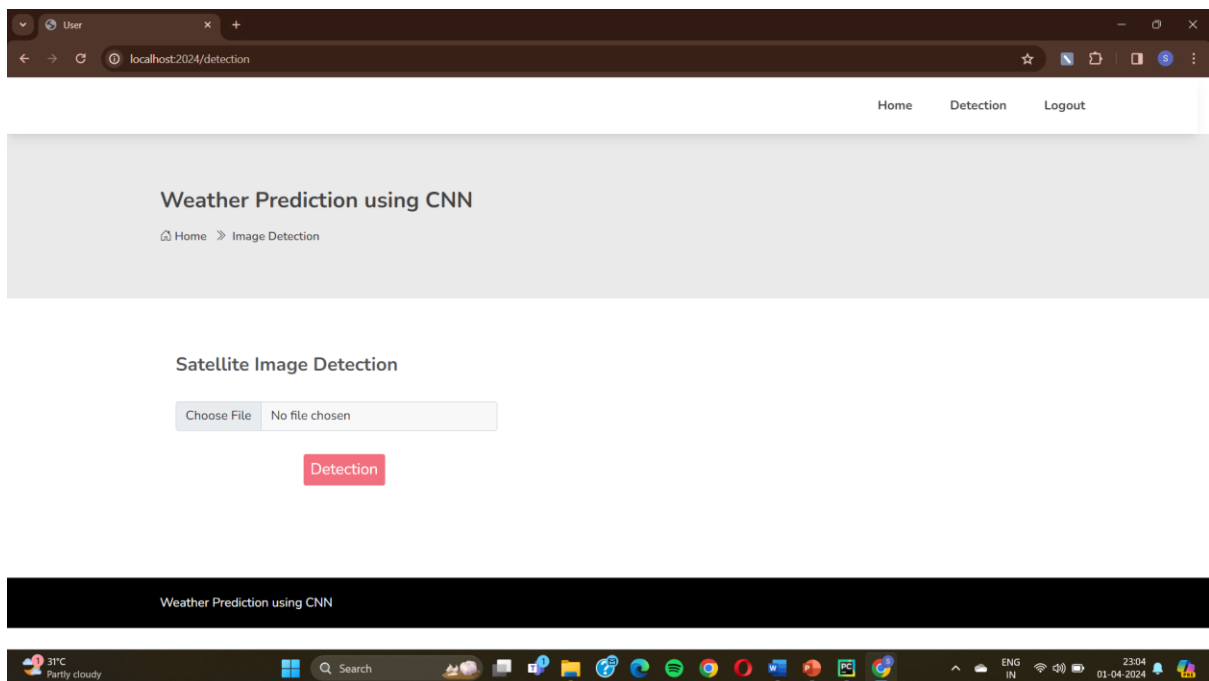


Fig : Satellite Image Detection

Weather Prediction using CNN project



Fig : Output-1 Cloudy

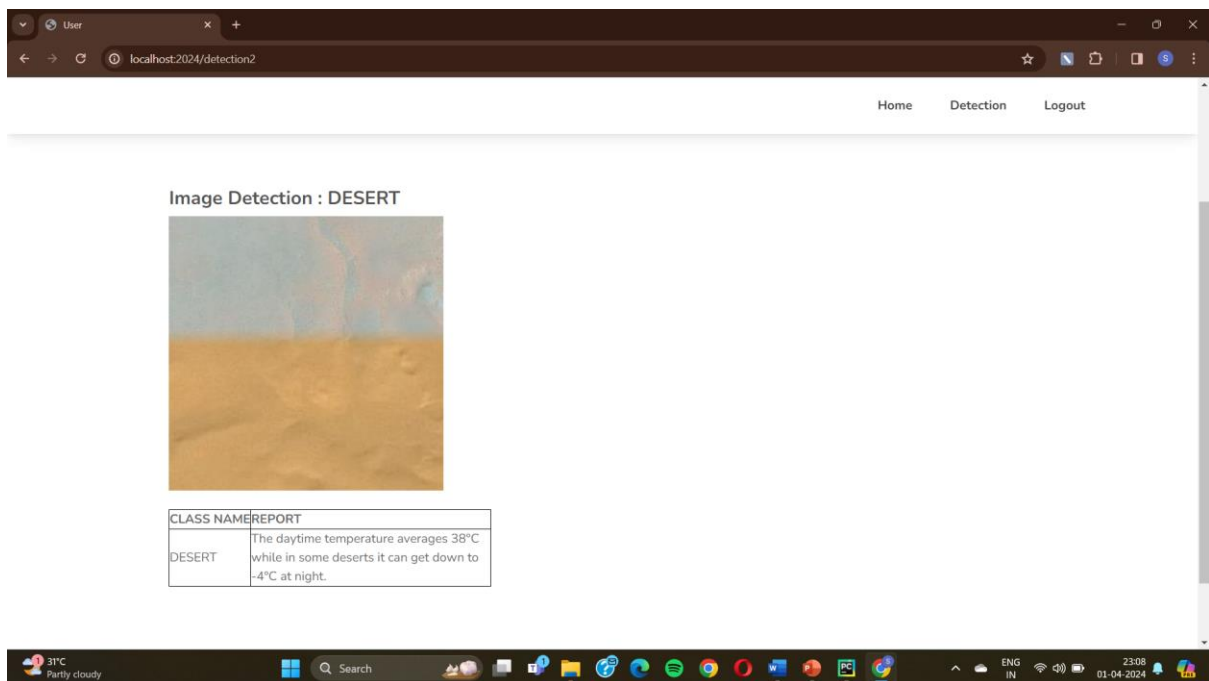


Fig : Output-2 Desert

Weather Prediction using CNN project

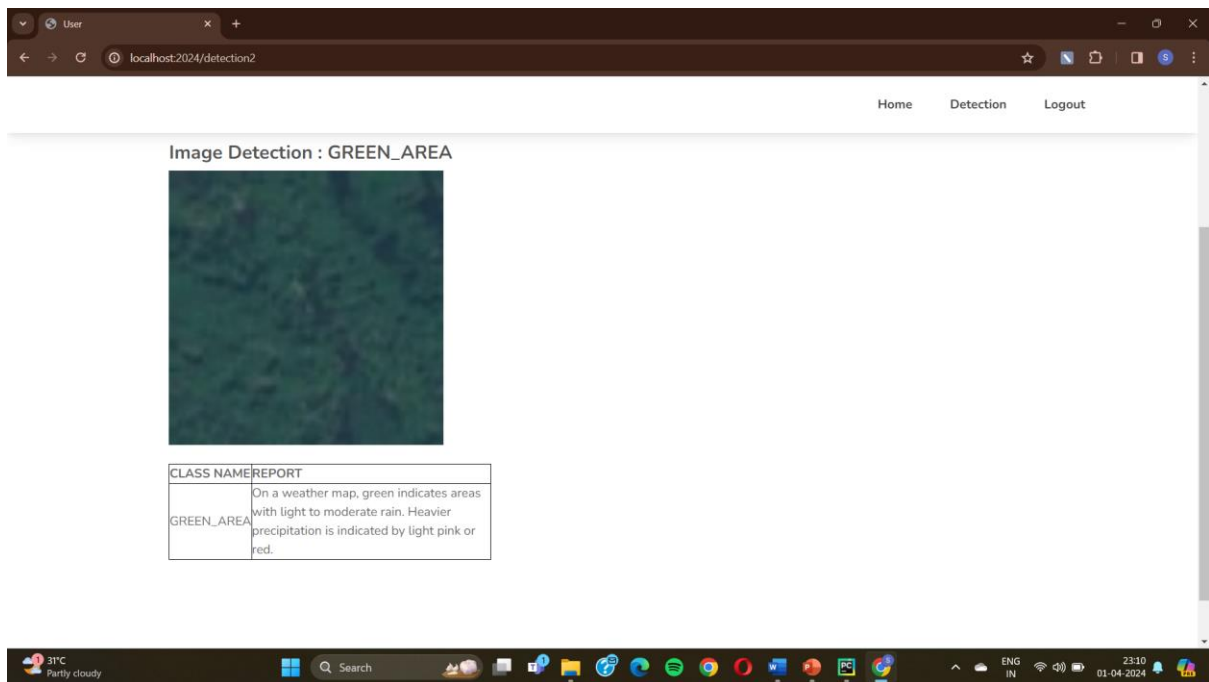


Fig. : Output-3 Green_Area

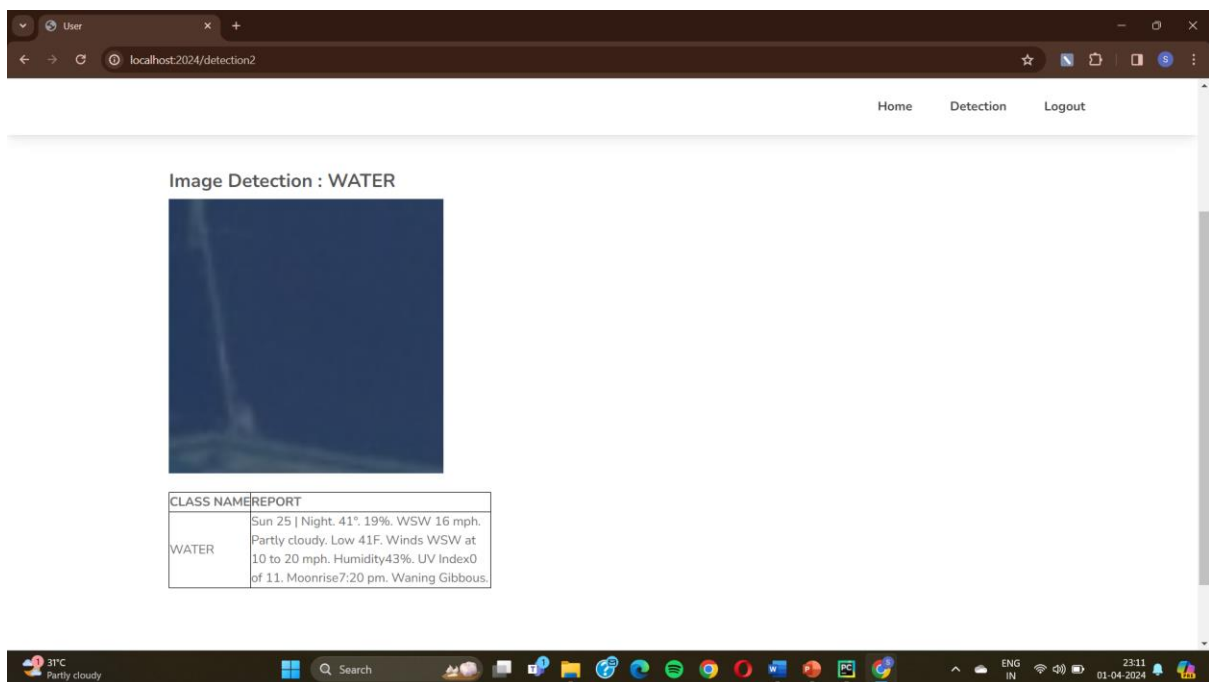


Fig.: Output-4 Water

The figures show the output of the project. Our project aim is to predict the weather using satellite images. For that we have developed a web page using flask application in python. This web application is a user-friendly application. The figures shows the first page of the web application. It shows with the title as Weather Prediction using Satellite Images CNN(VGG16). In the same page we can also see three tabs at the right corner. They are Home, Admin and User. By clicking on admin, we go to another page where it shows the validation graphs. By clicking on the user, we go to the login page of the user shown in fig 8. In this page we can also see registration tab where we can also register with our details. This helps to login easily. After login, it is directed to another page where the user needs to upload the image. It is shown in fig. In this page, the user can upload an image of jpeg or png format. This image is used as an input in our model. After uploading the image, it gives the output as shown. If the image is cloudy, then it will be shown as in fig. If the image is desert, then it will be shown as in fig. If the image is green area, then it will be shown as in fig. If the image is water, then it will be shown as in fig. These are the outputs of our project.

Through rigorous testing and validation procedures, we demonstrate the model's effectiveness in accurately predicting weather conditions. The model achieves an impressive classification accuracy of approximately 95%, validating its reliability and utility for real-world applications.