INTP22-ML-4

STEEL DEFECT DETECTION USING COMPUTER VISION

PHASE-3 REPORT

Submitted by

PAVAN KUMAR M

Under the mentorship of Devesh Tarasia

PHASE – 3 OBJECTIVES

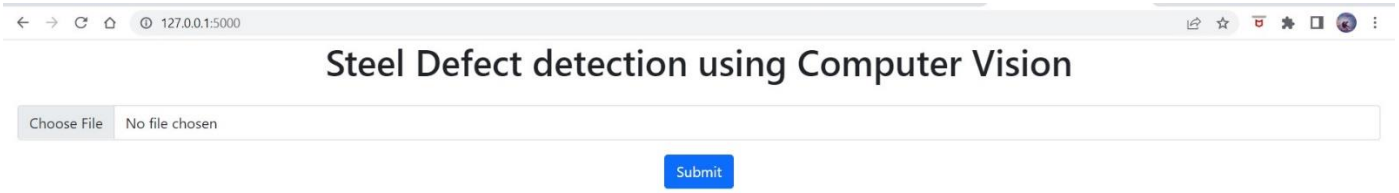
- ❖ Testing in the Local Machine
- ❖ Deployment to Cloud

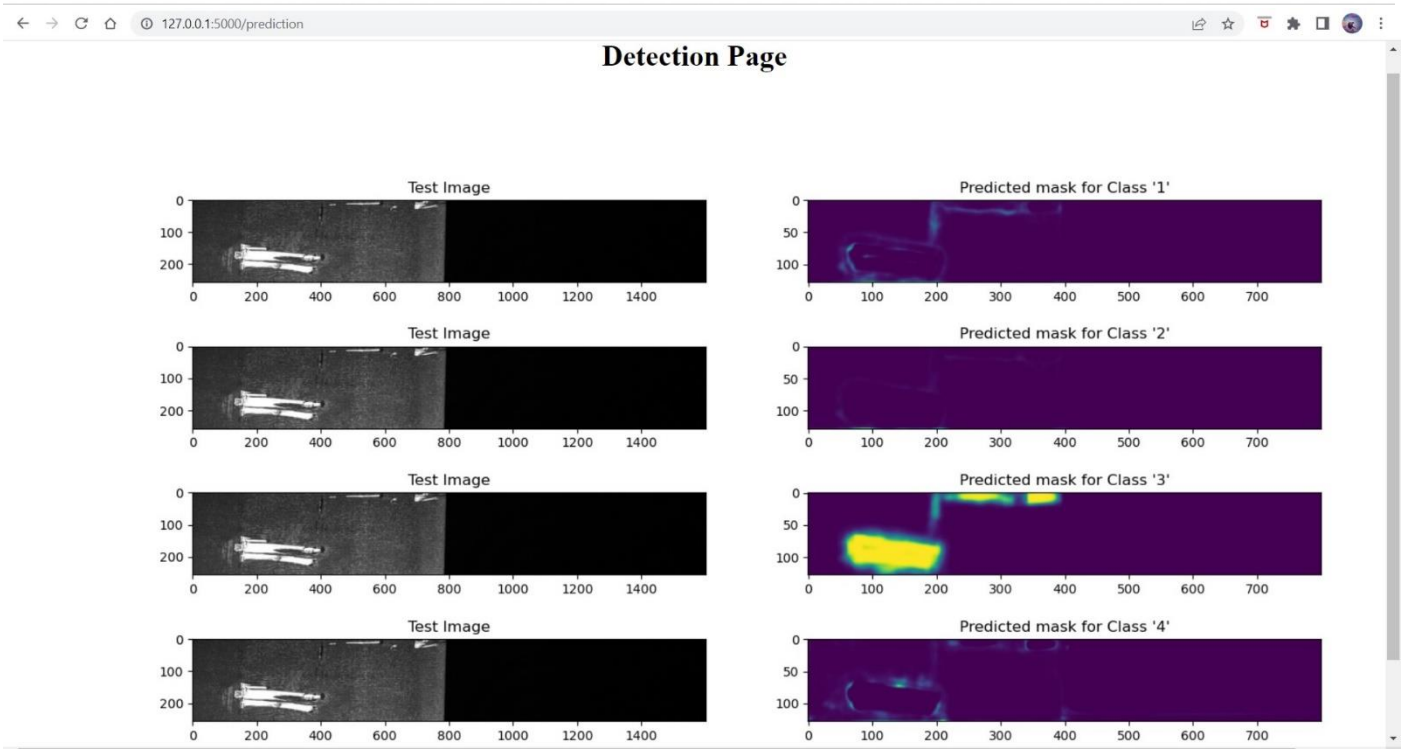
For deployment there were many Frameworks that are available but I used Flask framework which I felt beginner friendly.

Steps followed in Deployment:

- Firstly, I created a Virtual Environment specifically for deploying the ML model that was developed in stage 2.
- Then I installed the Packages (Libraries) required to run this Model.
- After installing all the necessary packages, started importing all of them in Spyder IDE.
- Then I loaded the saved model onto Spyder IDE by passing the Custom Objects and defining the Custom object function.
- Then before actually deploying I just tested whether HTML pages are working as intended, for this I first tried with printing Hello World in HTML page and got Positive result.
- Then started coding for deploying the original Model, getting Predictions out of it and displaying it to the user.
- I created 2 simple HTML web page one to take user input i.e. image of steel Image which may or may not have defect. Another to show the Prediction to the user.
- So, once the user uploads the image in HTML page and clicks submit it is passed onto the model. And model predicts whether there is a defect or not. If defect is detected then the model identifies the Defect region and also classifies it to the class it belongs to.
- And this is what expected at the end of this project.
- From the Detection page user could able to concluded whether the image he/she uploaded has defects, if yes it classifies the image into that particular class and identifies the defective region as well.

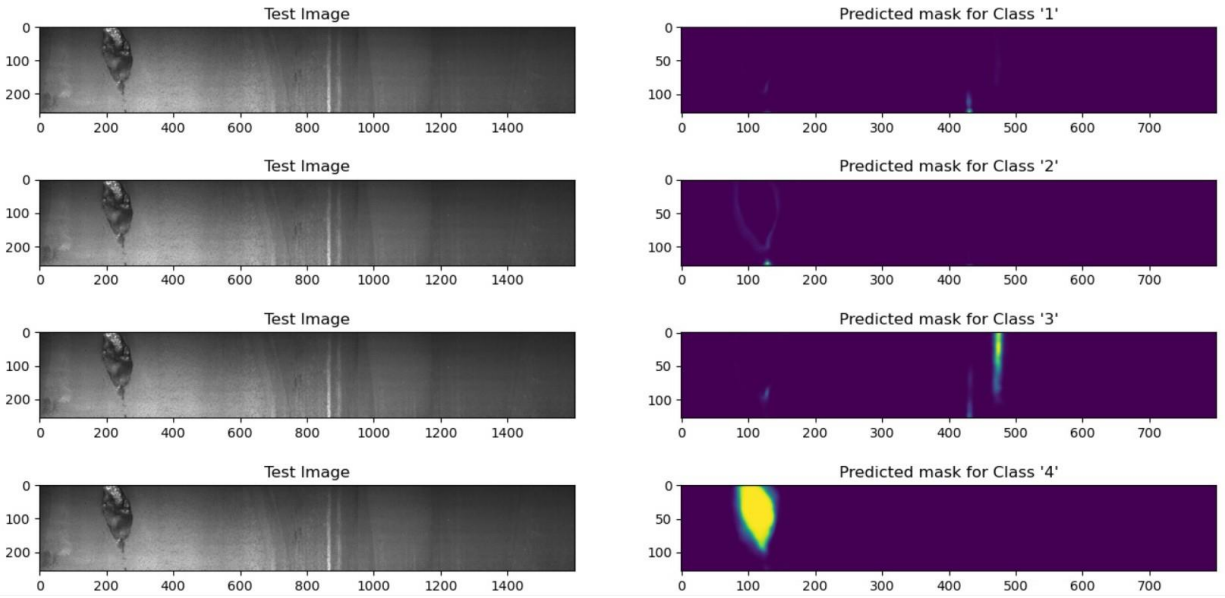
Few examples of above said steps are attached below:


127.0.0.1:5000
Steel Defect detection using Computer Vision
Choose File
No file chosen
Submit


127.0.0.1:5000/prediction
Detection Page
Test Image
Predicted mask for Class '1'
Test Image
Predicted mask for Class '2'
Test Image
Predicted mask for Class '3'
Test Image
Predicted mask for Class '4'

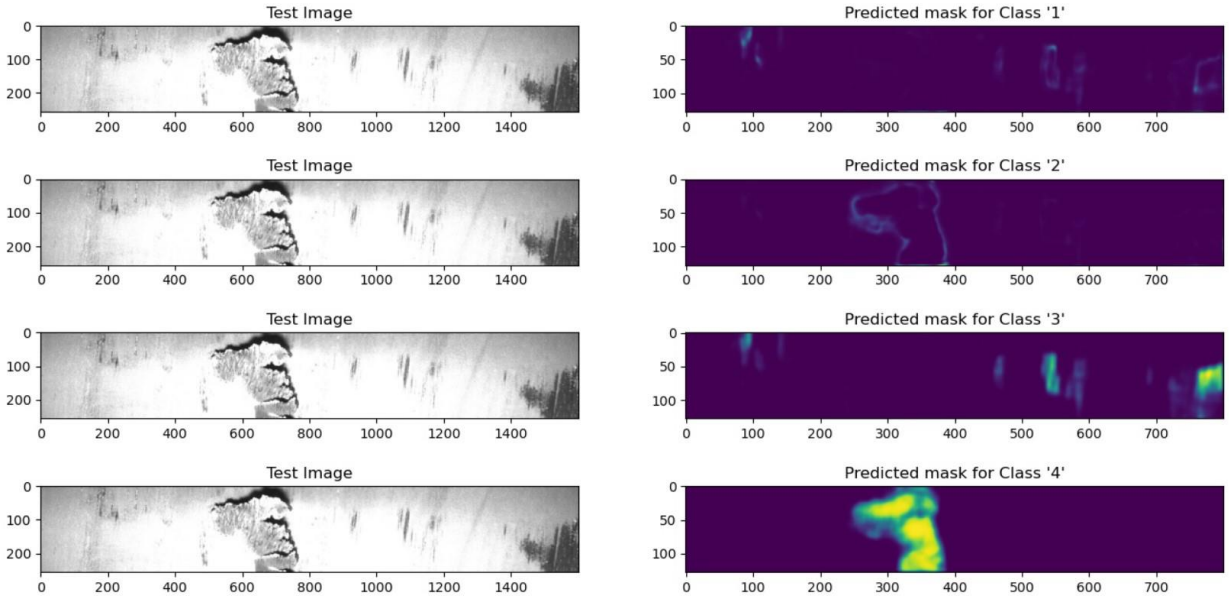
127.0.0.1:5000/prediction

Detection Page



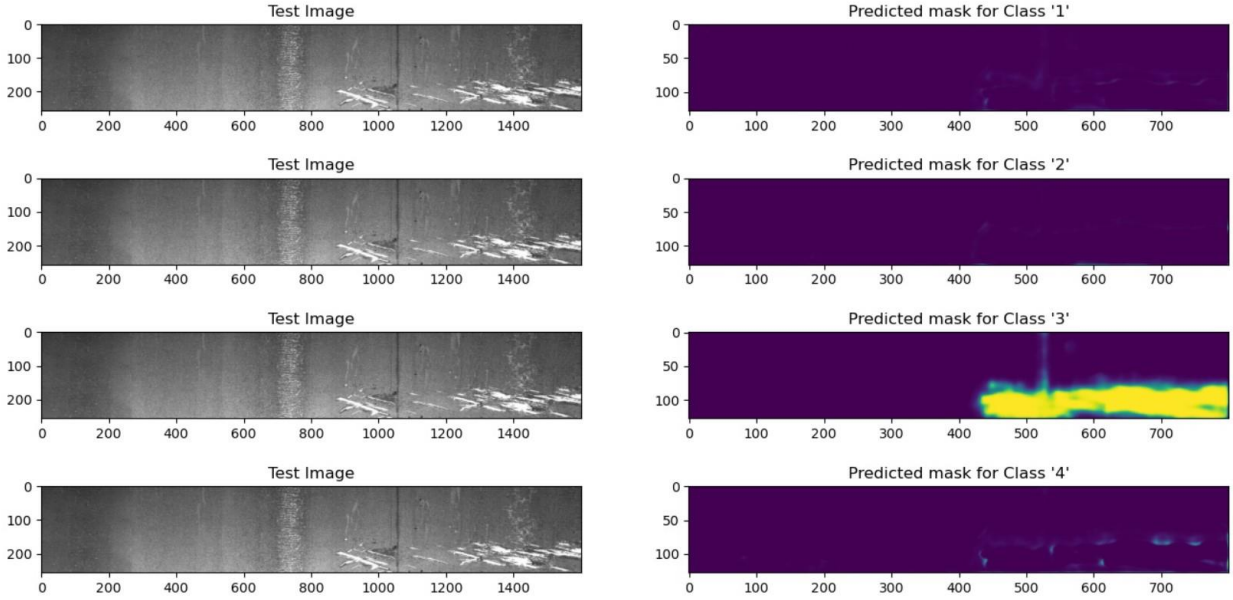
127.0.0.1:5000/prediction

Detection Page

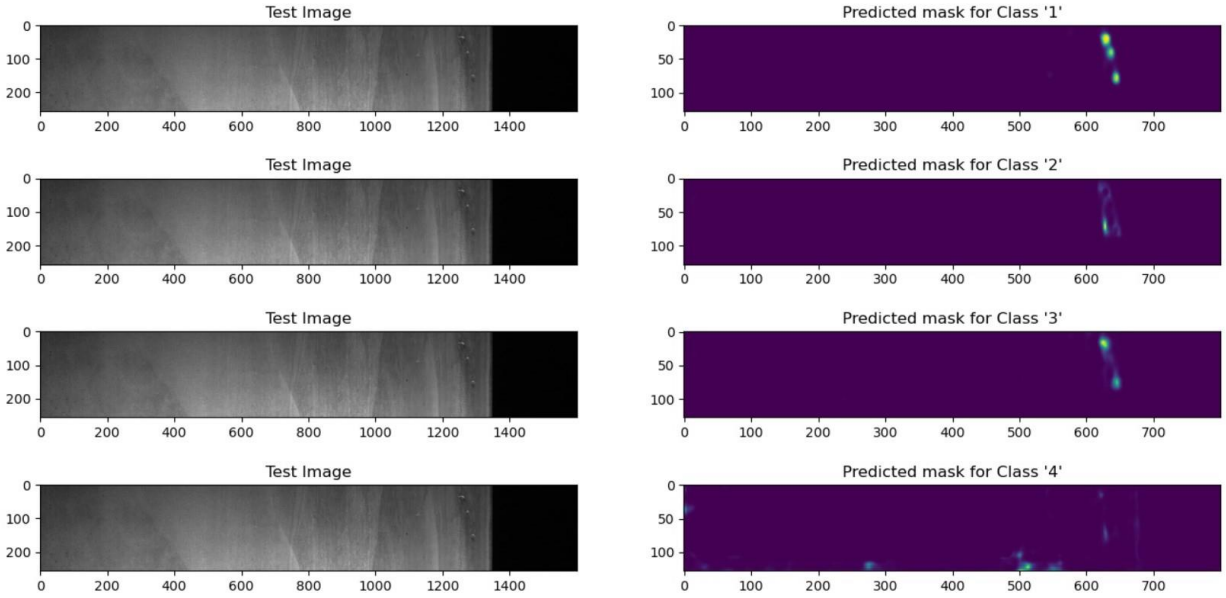


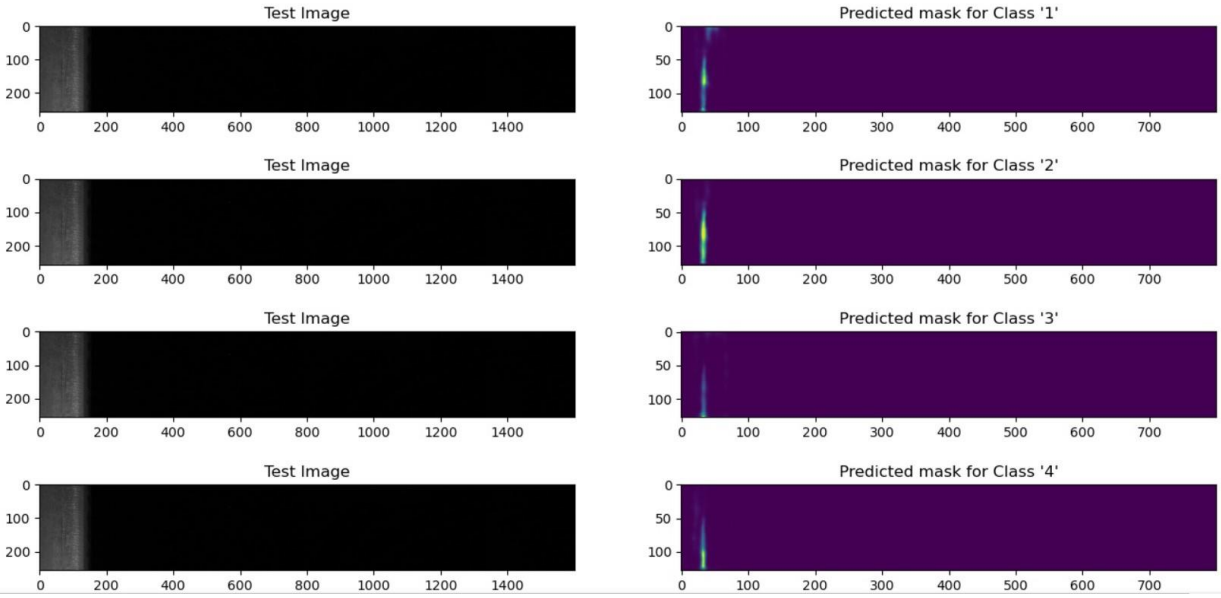
127.0.0.1:5000/prediction

Detection Page



Detection Page



127.0.0.1:5000/prediction

Detection Page