## **PROJECT DEVELOPMENT PHASE**

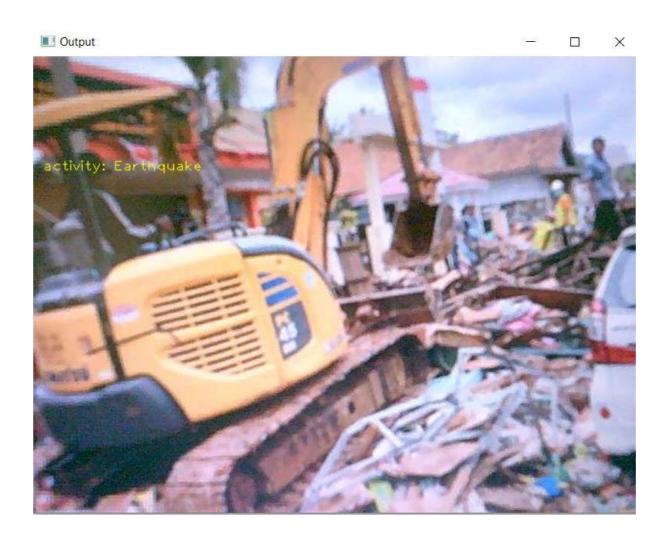
## **SPRINT-4**

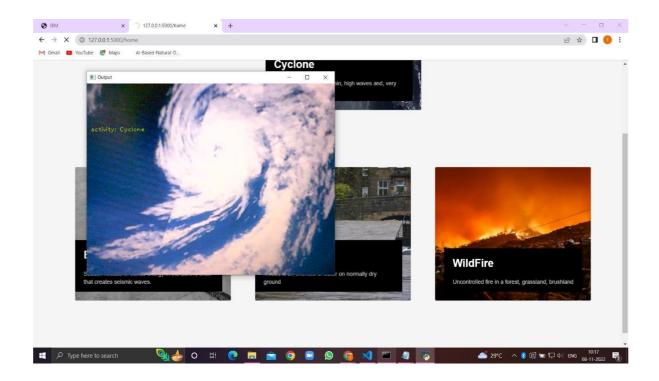
Date	07 November 2022
Team ID	PNT2022TMID22608
Project Name	Natural Disaster Intensity Analysis and Classification using Artificial Intelligence

## INTEGRATE THE WEB APP WITH AI MODEL:

After creating the Model, the Model should be integrated with the web app using the Flask application. The coding part is named as app.py and it will be running in the local host through the generated link. By navigating the local host the webpage will be visible.

```
output = frame.copy()
73
               #print("apple")
74
               frame = cv2.cvtColor(frame, cv2.COLOR BGR2RGB)
               frame = cv2.resize(frame, (64, 64))
76
               #frame = frame.astype("float32")
               x=np.expand_dims(frame, axis=0)
78
               result = np.argmax(model.predict(x), axis=-1)
79
               index=['Cyclone','Earthquake','Flood','Wildfire']
30
31
               result=str(index[result[0]])
32
               #print(result)
33
               #result=result.tolist()
34
               cv2.putText(output, "activity: {}".format(result), (10, 120), cv2.FONT_HERSHEY_PLAIN,
85
                            1, (0,255,255), 1)
               #playaudio("Emergency it is a disaster")
37
               cv2.imshow("Output", output)
38
39
               key = cv2.waitKey(1) & 0xFF
90
91
                  ## if the `q` key was pressed, break from the loop
               if key == ord("q"):
92
                   break
93
94
           # release the file pointers
95
           print("[INFO] cleaning up...")
96
97
           vs.release()
98
           cv2.destroyAllWindows()
           return render_template("upload.html")
99
90
31
02 if __name__ == '__main__':
93
         app.run(debug=False, threaded=True)
34
```





## **MODEL DEPLOYMENT:**

The trained model which is running in the localhost without any error is deployed in the IBM Cloud for making available for the users to predict the Disaster's type and its intensity. It is integrated with the Flask application.

