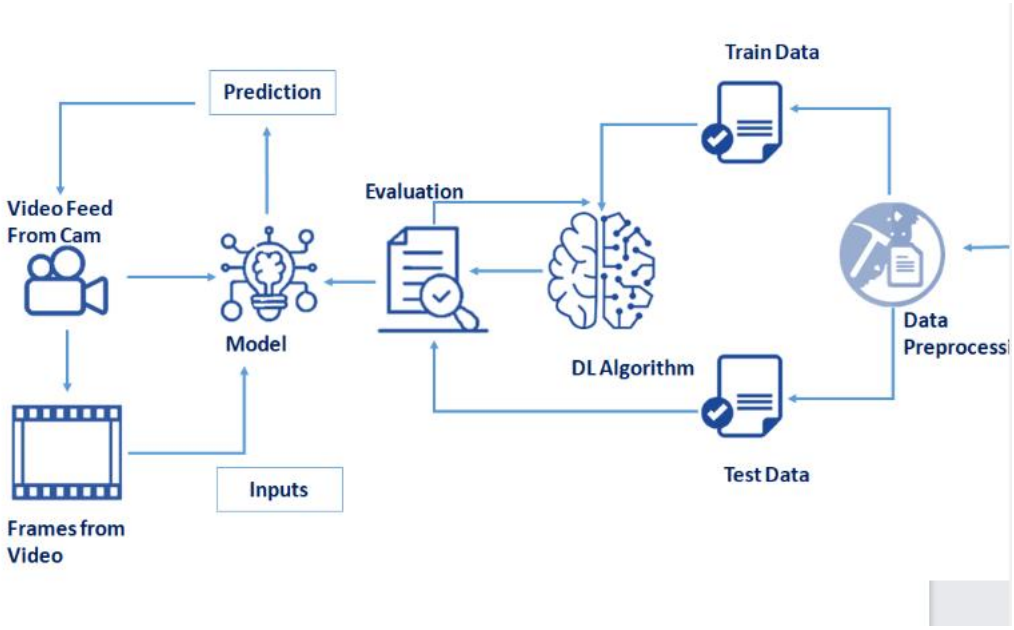
Natural Disasters Intensity Analysis and Classification using Artificial Intelligence

Abstract:

Natural disasters not only disturb the human ecological system but also destroy the properties and critical infrastructures of human societies and even lead to permanent change in the ecosystem. Disaster can be caused by naturally occurring events such as earthquakes, cyclones, floods, and wildfires. Many deep learning techniques have been applied by various researchers to detect and classify natural disasters to overcome losses in ecosystems, but detection of natural disasters still faces issues due to the complex and imbalanced structures of images. To tackle this problem, we developed a multilayered deep convolutional neural network model that classifies the natural disaster and tells the intensity of disaster of natural The model uses an integrated webcam to capture the video frame and the video frame is compared with the Pre-trained model and the type of disaster is identified and showcased on the OpenCV window.

Technical Architecture:



Project Objectives

By the end of this project you will:

- know fundamental concepts and techniques of the Artificial Neural Network and Convolution Neural Networks

- Gain a broad understanding of image data.
- Work with Sequential type of modeling
- Work with Keras capabilities
- Work with image processing techniques
- Work with Opencv

Project Flow

- The user interacts with the UI (User Interface) to open the integrated webcam.
- The video frames are captured and analyzed by the model which is integrated with flask application.
- Once model analyses the video frames, the prediction is showcased on the UI and OpenCV window

To accomplish this, we have to complete all the activities and tasks listed below

- Data Collection.
    - Collect the dataset or Create the dataset
- Data Preprocessing.
    - Import the ImageDataGenerator library
    - Configure ImageDataGenerator class
    - ApplyImageDataGenerator functionality to Trainset and Testset
- Model Building
    - Import the model building Libraries

Project Structure:

Create a Project folder which contains files as shown below

- Dataset folder contains the training and testing images for training our model.
- We are building a Flask Application that needs HTML pages stored in the templates folder and a python script app.py for serverside scripting
- we need the model which is saved and the saved model in this content is a disaster.h5
- templates folder contains home.html,intro.html,upload.html pages.

Project requirements:

to build Machine learning models you must require the following packages

- Flask
- Python packages: type “pip install numpy” and click enter.
- Type “pip install pandas” and click enter.
- Type “pip install scikit-learn” and click enter.
- Type “pip install opencv-contrib-python” and click enter.
- Type “pip install tensorflow==2.3.0” and click enter.
- Type “pip install keras==2.4.0” and click enter.
- Type “pip install Flask” and click enter.

DATASET:
https://drive.google.com/file/d/11-FdbTaJVrpwQmaCLV5gYYDQlfTeD0uz/view?usp=sharing

MODEL BUILDING: We are ready with the augmented and pre-processed image data, Lets begin our model building, this activity includes the following steps

- Import the model building Libraries
- Initializing the model
- Adding CNN Layers
- Adding Hidden Layer
- Adding Output Layer
- Configure the Learning Process
- Training and testing the model
- Saving the model

ACTIVITES:

- Initializing the model
- Adding CNN layers
- Adding Dense Layers
- Configure The Learning Process
- Train The Model
- Save The Model
- Test The Model

IMAGE PREPROCESSING:

- Import The ImageDataGenerator Library
- Configure ImageDataGenerator Class
- Home.htmlApply ImageDataGenerator Functionality To Trainset And Testset

TEMPLATE: It Stores the key HTML files such as:

- Home.html
- Intro.html
- Upload.html

APPLICATION BUILDING:

1)BUILD HTML PAGES

- We use HTML to create the front end part of the web page.
- Here, we have created 3 HTML pages- home.html, intro.html, and upload.html
- home.html displays the home page.
- Intro.html displays an introduction about the project
- upload.html gives the emergency alert

2)BUILD THE PYTHON CODE:

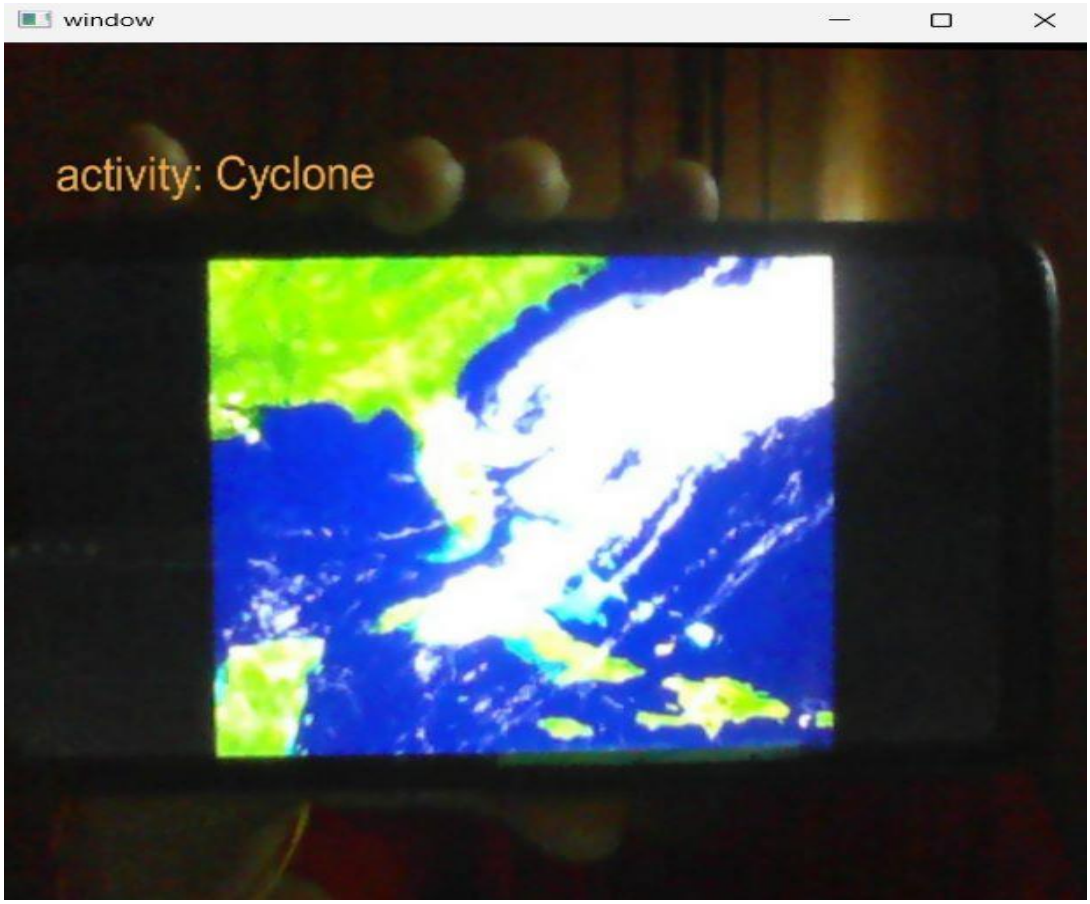
- Importing Libraries such as flask modules and so on
- Creating our flask application and loading our model by using load_model method
- Routing to the html Page
- using in-built webcam
- using video file residing on the disk

- Convert the image from BGR to RGB
- Predicting the results
- Displaying the result

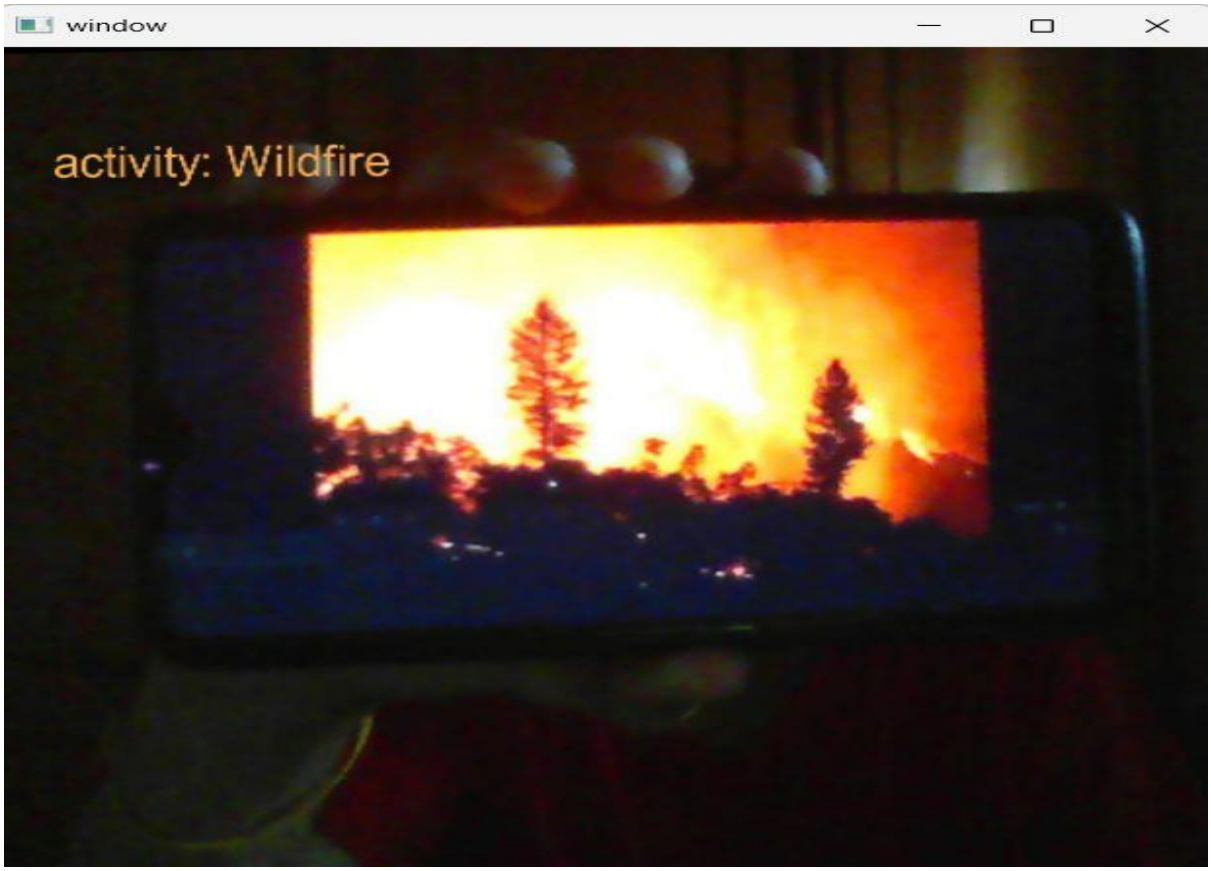
Run The Application:

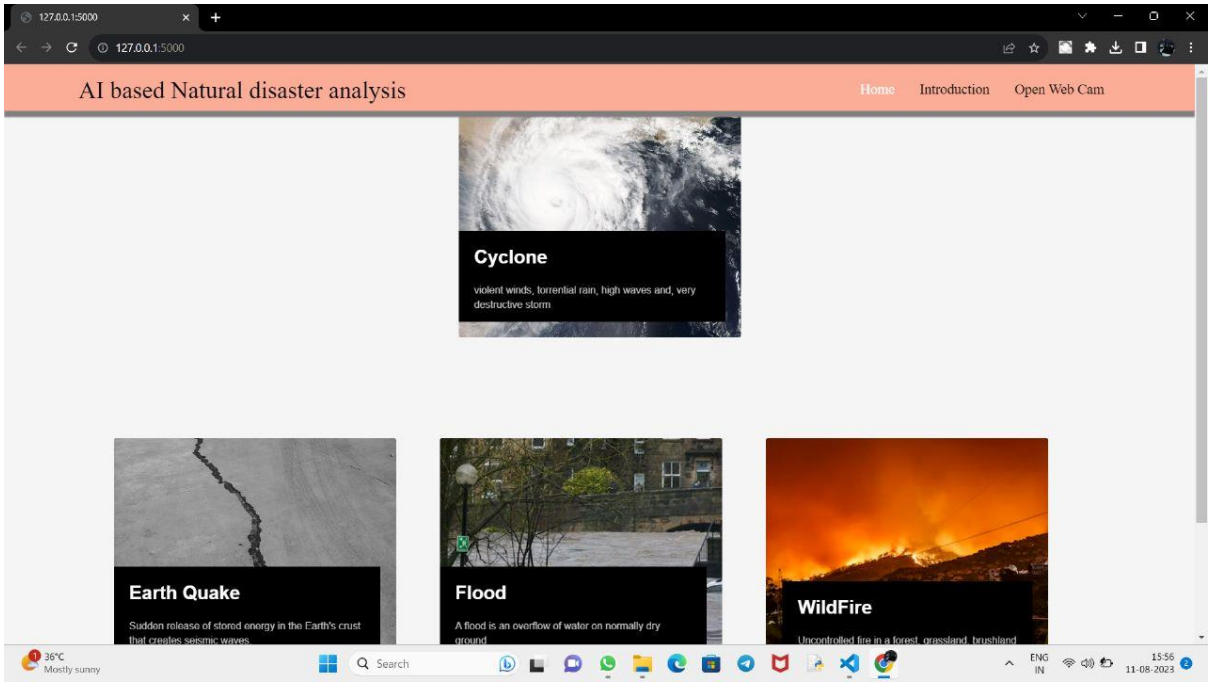
- Open vs code from start menu
- Navigate to the folder where your app.py resides.
- Now type “python app.py” command.
- It shows link on http://127.0.0.1:5000
- Copy that url and go to localhost to check the application
- Enter the required input to get predictions

Screenshots:

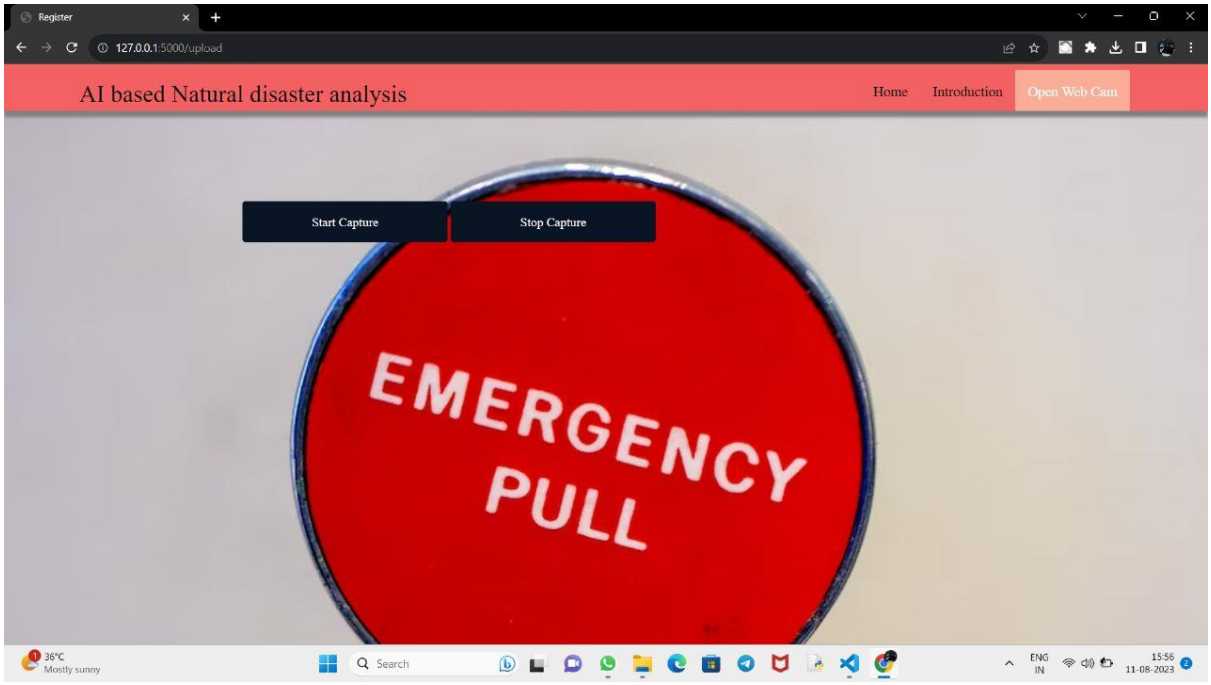

window
activity: Cyclone


window
activity: Earthquake


window
activity: Wildfire


127.0.0.1:5000
127.0.0.1:5000
AI based Natural disaster analysis
Home
Introduction
Open Web Cam
Cyclone
violent winds, torrential rain, high waves and, very destructive storm
Earth Quake
Sudden release of stored energy in the Earth's crust that creates seismic waves
Flood
A flood is an overflow of water on normally dry ground
WildFire
Uncontrolled fire in a forest, grassland, brushland
36°C
Mostly sunny
Search
ENG
IN
15:56
11-08-2023

China, India and the United States are among the countries of the world most affected by natural disasters.
Natural disasters have the potential to wreck and even end the lives of those people, who stand in their way. However, whether or not you are likely to be affected by a natural disaster greatly depends on where in the world you live, The objective of the project is to human build a web application to detect the type of disaster . The input is taken from the in built web cam, which in turn is given to the pre trained model . The model predicts the type of disaster and displayed on UI.



CONCLUSION:

In conclusion, this project presents a significant advancement in the field of natural disaster intensity analysis through the utilization of artificial intelligence. The successful development and implementation of the Convolutional Neural Network (CNN) architecture demonstrate its efficacy in processing diverse data sources and providing accurate intensity analysis. By overcoming the limitations of traditional methods, our project contributes to more precise, timely, and scalable disaster management practices.

The implications of this work extend to enhancing disaster response planning, resource allocation, and mitigation strategies. The successful integration of real-time analysis and adaptive algorithms underscores the potential for rapid decision-making in dynamic disaster scenarios. Through intuitive visualizations, the project enables stakeholders to comprehend intensity analysis results effectively.

Moving forward, the project sets the stage for further research and innovation in AI-driven disaster management systems. The culmination of our efforts underscores the transformative role of artificial intelligence in safeguarding communities and environments from the impacts of natural disasters.