Methodology/Procedure

(i) Dataset Collection and Training Using Roboflow Collaborative Notebook

o Utilized the Roboflow Collaborative Notebook provided by Roboflow for training custom object detection models.

o Curated a comprehensive fire image dataset from Roboflow, ensuring diversity in lighting conditions, perspectives, and scenarios.

o Applied the YOLOv5 model architecture to the dataset using the notebook, training the model to achieve the desired level of accuracy.

o Obtained the trained weights file from the YOLOv5 training process, which would serve as the model for fire detection.

(ii) Raspberry Pi Setup and Package Installation

o Installed essential packages on the Raspberry Pi, including OpenCV, YOLOv5, NumPy, and other dependencies to facilitate smooth model execution.

o Configured the Raspberry Pi to ensure compatibility with the trained YOLOv5 model.

o Used the Raspberry Pi Imager to install Bullseye v11 with a 64-bit architecture onto the SD card.

(iii) Model Deployment on Raspberry Pi

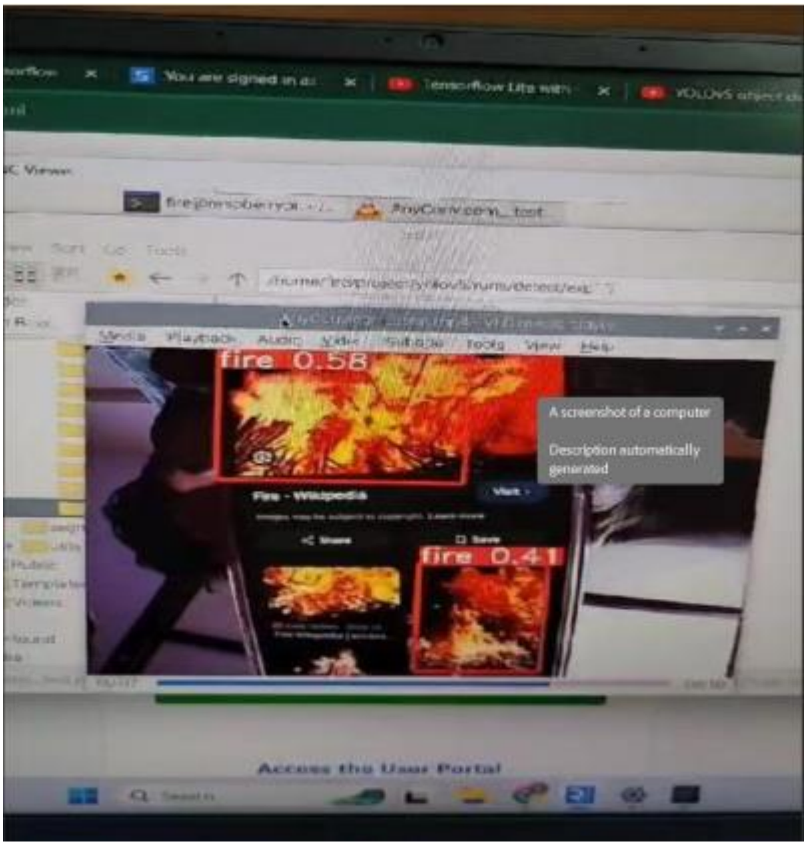
o Transferred the obtained weights file from the training phase to the Raspberry Pi.

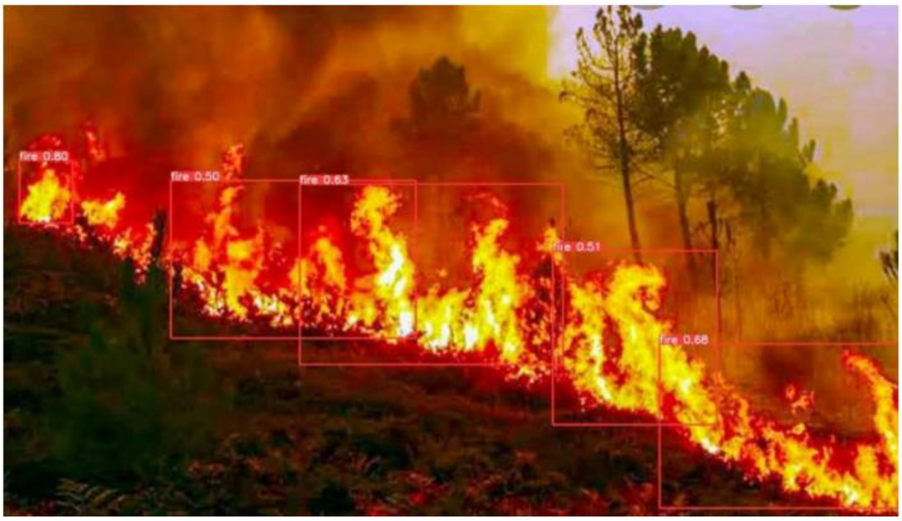
o Utilized the detect.py script provided by YOLOv5 to set the path to the trained weights file and ran inference on the Raspberry Pi.

o Observed and analyzed the model's real-time performance in fire detection on the Raspberry Pi, ensuring its functionality within the constraints of the device

Results

We ran inference on video (.mp4 file) and the results are displayed below. We split the video into separate frames and the model proceeded to analyse the images.





NOTE: It also works on real time with a Rpi compatible camera.