Fire Detection

Explaining Approach selection :

Since this task will be used in real-time to detect fires, my choice was based on selecting a fast and accurate model that can effectively perform object detection. For this reason, I reviewed the top models for object detection, considering their inference speed, as time is crucial. A delay of just a few minutes can significantly impact firefighters' ability to respond effectively.

After evaluating various options, the best models that meet these criteria are **YOLOv8**, **YOLOv5**. Each of these models offers a balance of speed and accuracy, making them suitable for real-time applications in fire detection.

Selection of top-performing model:

I conducted several training experiments with different models and hyperparameters, evaluating their performance on the validation dataset using various metrics. This comprehensive approach allowed me to assess each model's strengths and weaknesses, enabling me to identify the most effective configurations for fire detection. The evaluation included **mean Average Precision (mAP)**, which measures the overall accuracy of the model at a specific Intersection over Union (IoU) threshold; **mean Average Precision across IoU thresholds of 0.50 to 0.95 (mAP50-95)**, which provides a more comprehensive assessment of model performance across varying levels of detection precision; **precision**, which indicates the proportion of true positive detections among all positive predictions, helping to minimize false alarms; and **recall**, which measures the proportion of true positive detections among all actual instances of fire, ensuring that as many fires as possible are detected. Together, these metrics provided valuable insights into the models' accuracy and reliability in detecting fire-related objects.
The results are presented at the end of the notebook, and the best-performing model was **YOLOv8**.

Further work :

In the coming month, I plan to continue training other approaches and assess their performance on my dataset, such as Fast R-CNN, DETR, and EfficientNet. I will also investigate ways to enhance the dataset, either by increasing its size or by applying various data augmentation techniques to improve model robustness. Additionally, I will dedicate time to interpreting the model's results using methods such as Grad-CAM, EigenCAM, HiResCAM, and LayerCAM, which will help in understanding the model's decision-making process and identifying areas for improvement.Furthermore, I aim to optimize the model's speed by reducing its inference time as much as possible through techniques like quantization and pruning, which can significantly accelerate detection.

I will also work on developing a user-friendly interface for firefighters to easily access detection results and analytics and to provide feedback based on real scenarios. This feedback will be crucial for continuously improving the model's performance in practical usage. For example, I can utilize Gradio to create an interactive interface that facilitates this process.