Abstract

Real-time object detection plays a vital role in computer vision applications such as surveillance, industrial safety, smart monitoring, and embedded AI systems. With the advancement of deep learning models like YOLO (You Only Look Once), object detection has become faster and more accurate, enabling deployment in real-time environments. However, practical implementation often requires seamless integration of detection models, user interfaces, and alert mechanisms to make the system usable and deployable in real-world conditions.

Existing System

Existing object detection systems commonly rely on command-line execution or standalone scripts that process images or video streams without an interactive interface. Most implementations focus primarily on model inference and accuracy evaluation rather than usability and deployment flexibility. Additionally, many systems are designed for cloud-based environments or require constant internet connectivity to download models and dependencies.

Traditional setups typically use a single YOLO version and lack the flexibility to switch between models based on performance requirements. They also do not always provide real-time alert mechanisms tailored for safety-critical applications, such as detecting human presence near restricted zones. As a result, these systems may be technically effective but are limited in terms of portability, user interaction, and real-world monitoring capabilities.

Proposed System

To overcome these limitations, the proposed system introduces a **real-time object detection application** built using Python, integrating YOLOv5, YOLOv8, and YOLOv11 models within a user-friendly desktop interface developed using Tkinter. The system allows dynamic selection of different YOLO variants, enabling users to balance speed and accuracy based on their requirements.

The application captures live webcam input through OpenCV and performs real-time inference using PyTorch and the Ultralytics framework. A dedicated event-driven alert mechanism detects human presence (COCO class ID 0) and triggers both a visual overlay and an audible beep notification. A cooldown mechanism ensures alerts are not repeatedly triggered in rapid succession.

Furthermore, the proposed system supports both online and offline installation modes, making it suitable for deployment in restricted-network environments such as industrial facilities or edge AI systems. By combining model flexibility, real-time detection, graphical interface control, and deployment readiness, the proposed solution provides a practical, scalable, and user-friendly object detection framework for modern monitoring applications.