AI-POWERED TARGET RECOGNITION SYSTEM

Submitted by

SUJAN S

TABLE OF CONTENTS

ABSTRACT

The increasing prevalence of public safety threats in sensitive areas such as airports, schools, and government facilities necessitate the development of intelligent, proactive surveillance systems. This project introduces an AI-powered target recognition system designed to detect weapons like guns and knives in real-time using advanced deep learning techniques. At the core of the system lies a fine-tuned YOLO-based object detection model that has been trained specifically on a curated dataset of weapon images, addressing the precision limitations commonly associated with generic object detection frameworks. By integrating this model with real-time video feed analysis, motion detection algorithms, and dual-mode imaging (normal and thermal), the system provides immediate, accurate alerts upon threat identification. Additionally, it includes functionalities such as live annotations, automated frame capture upon detection, and audio-visual alert mechanisms, all of which contribute to heightened situational awareness. Unlike traditional surveillance setups that rely heavily on manual monitoring, this system minimizes human error and ensures uninterrupted vigilance. It is optimized to run on standard computing platforms with GPU support, eliminating the need for expensive specialized hardware. Furthermore, the system leverages thermal imaging to detect potential bombs or explosive devices based on their heat signatures, expanding its threat detection capabilities beyond conventional weapons. This project thus represents a scalable, cost-effective, and high-accuracy solution that redefines real-time security through automation, making it suitable for implementation in diverse environments ranging from educational institutions to critical infrastructure facilities. Its contribution lies not only in the technological innovations it presents but also in its potential societal impact—preventing harm, saving lives, and establishing a new benchmark for AI-driven security systems.

INTRODUCTION

The rapid advancement of artificial intelligence and computer vision technologies has led to significant transformations in the field of surveillance and security. As threats such as terrorism, armed violence, and unauthorized access continue to pose critical challenges to public and private sectors alike, there is a growing need for intelligent systems that can detect, analyze, and respond to potential threats with speed and accuracy. Traditional surveillance systems rely on human operators to monitor live camera feeds, an approach that is prone to fatigue, distraction, and delayed responses. Such limitations have created opportunities for AI-based systems to take the lead in redefining how surveillance can be managed.

The AI-Powered Target Recognition System is a novel solution that employs machine learning and computer vision to detect weapons in real-time. This project focuses on the development and implementation of a custom-trained YOLO (You Only Look Once) object detection model tailored to identify potentially harmful objects such as guns, knives, and explosive devices. The system is designed to analyze video streams from CCTV or drone cameras and instantly detect threats with high accuracy, even in complex or crowded environments. This eliminates the risk of missed detections due to human oversight and enables faster response times from security personnel.

The concept of AI-powered surveillance is not entirely new, but what sets this project apart is its emphasis on threat-specific training data, system optimization for real-time performance, and integration of auxiliary features such as thermal imaging toggling and motion-based event triggering. Most pre-trained YOLO models are built using general datasets like COCO, which are not sufficient for niche applications like weapon detection. To overcome this challenge, this project leverages a curated dataset composed of weapon-specific images that include various orientations, lighting conditions, and backgrounds to ensure robustness.

Additionally, the system integrates features such as motion detection, alert systems, and live annotations to provide an end-to-end surveillance solution. The motion detection module acts as a supplementary trigger to reduce unnecessary computational load by activating the object detection model only when movement is sensed in the camera's field of view. When a weapon or explosive device is detected, the system generates real-time alerts and highlights the object in the video frame using bounding boxes and labels.

This project also aims to create a deployment-ready solution that can be operated using a standard PC with GPU support, thus eliminating the dependency on high-end infrastructure. This makes the system more accessible and scalable for use in environments where cost and mobility are crucial concerns, such as mobile surveillance units or remote monitoring stations. The inclusion of thermal vision support further enhances detection capabilities in low-light or visually complex environments, while also enabling the detection of bombs based on heat emissions.

In summary, the AI-Powered Target Recognition System represents an innovative blend of deep learning, real-time computing, and intelligent automation. It addresses key challenges in security monitoring by reducing the risk of human error, enhancing threat detection speed, and offering a versatile, hardware-efficient architecture. Through this project, we aim to provide a practical, reliable, and forward-looking security solution that has both technical and societal impact.

SCOPE AND MOTIVATION

In the current global landscape, where violence and security breaches are increasingly prevalent, the need for intelligent and proactive surveillance systems has never been more critical. The **AI-Powered Target Recognition System** is developed with the goal of addressing these challenges by leveraging the capabilities of artificial intelligence, machine learning, and thermal imaging to detect weapons, bombs, and other threats in real time. The system has been carefully designed to provide automated monitoring, identification, and alerting, thereby reducing dependency on human supervision and enhancing response time in critical situations.

The **scope** of this project extends beyond conventional surveillance. It includes real-time detection of firearms, knives, and thermal signatures indicative of explosive devices or bombs—something that traditional security cameras often fail to recognize. By integrating a YOLO-based object detection model with both RGB and thermal camera inputs, the system can function effectively in various lighting and environmental conditions. Whether deployed in military zones, airports, educational institutions, or public venues, this system serves as a scalable, adaptable, and robust security solution.

The **motivation** behind this project is rooted in real-world concerns. Repeated instances of school shootings, terrorist attacks, and border infiltrations have exposed the limitations of manual surveillance systems. In most cases, a faster and more intelligent detection mechanism could have helped avert disaster. This realization has fueled the idea of creating a system that can detect threats in milliseconds and notify security personnel instantly, thereby minimizing damage and potentially saving lives.

Another motivating factor is the **inefficiency of manual CCTV monitoring**. Security guards often need to watch multiple screens simultaneously, which leads to fatigue and oversight. By automating the detection process using AI, the burden on human operators is reduced and efficiency is significantly increased. The system also includes motion detection and automatic screenshot capture upon identifying threats, allowing for immediate recordkeeping and evidence storage.

Furthermore, the integration of **thermal camera technology** extends the scope of this project into **bomb detection** and **heat-based analysis**. Explosives and unauthorized devices typically emit distinct heat patterns which can be identified using colormapped thermal imaging. This feature also makes the system suitable for night surveillance, smoke-obscured environments, and indoor/outdoor deployment where visibility is compromised.

The project also demonstrates the **practical application of classroom knowledge** in solving real-world problems. Using Python 3.10.15, OpenCV, PyTorch, and the YOLOv11l.pt model, the system is built from the ground up in Visual Studio Code and is capable of running on embedded platforms like the Raspberry Pi or Jetson Nano. This makes it accessible not only to high-end institutions but also for low-cost deployment in schools and public infrastructure.

Importantly, the system is **modular and customizable**, allowing it to be extended beyond weapon detection. The same framework can be retrained to detect fire hazards, unauthorized intrusions, or even identify missing persons in large crowds. This scalability ensures that the system remains relevant as new security challenges emerge.

In summary, this project is motivated by a combination of societal needs, technological feasibility, and a vision to build a **next-generation intelligent surveillance system**. Its scope spans across multiple domains, offering a solution that is not only reactive but proactive, ensuring safety, reducing manual dependency, and creating a secure environment for everyone.

DESCRIPTION

The AI-Powered Target Recognition System is a real-time intelligent surveillance tool designed to automatically detect weapons and explosive threats from live video feeds using machine learning and computer vision. It primarily uses the YOLO (You Only Look Once) architecture—one of the most popular and efficient object detection models—finely tuned to identify guns, knives, and explosives in real-time environments. The system operates using both RGB and thermal camera feeds, enhancing its capability to detect visible and hidden threats even under challenging lighting or visual conditions.

At the core of the system lies a custom-trained YOLOv5 or YOLOv11l.pt model, preloaded with a dataset that specifically includes varied images of weapons and explosive components. This model processes each frame from the video feed and draws bounding boxes around identified threats, clearly labeling them with confidence scores. When the confidence level surpasses a predefined threshold (typically around 0.6–0.8), the system triggers an alert mechanism—consisting of both a sound buzzer and annotated detection window—to notify users of a potential threat.

The project is developed using Python 3.10.15 in VS Code, leveraging powerful libraries such as OpenCV for video processing, PyTorch or TensorFlow for running the object detection model, and additional Python packages for file handling, GUI display, and system integration. The solution is hardware-agnostic but optimized to run efficiently on systems equipped with a GPU, ensuring real-time performance without significant lag or resource bottlenecks.

The system includes the following major modules:

1. **Video Feed Capture**

   The video feed is obtained from either a webcam, external CCTV camera, or thermal camera. The system supports seamless switching between visible and thermal modes, enabling use cases in both daylight and dark environments.

2. **Motion Detection Module**

   To conserve computing power and reduce false detections, a motion detection algorithm compares sequential frames and activates the weapon detection model only when significant motion is identified. This reduces background processing when the scene is static.

3. **YOLO Object Detection Module**

   This is the main intelligence unit. Using the custom-trained YOLO model, each active frame is analyzed to detect weapons or bombs. Detected objects are highlighted with bounding boxes and labeled with confidence scores. The model processes multiple objects in a single frame with impressive speed and precision.

4. **Alerting System**

   Upon detection, a buzzer (audio alert) is triggered, and a screenshot of the detection frame is saved. Simultaneously, the bounding boxes on screen help users visually track the threat. These features ensure immediate situational awareness.

5. **Thermal Imaging Support**

   A unique addition to the system is the thermal camera support, especially useful in environments with poor lighting. Thermal input allows detection based on heat signatures—ideal for identifying concealed weapons and bombs. This makes the system suitable for night surveillance or operations in smoke-filled/dusty areas.

6. **Detection Logging & Screenshot Capturing**

   For forensic and analytic purposes, every detection is logged with a timestamp, and screenshots are automatically saved in a dedicated folder. This archive can be used for training, performance evaluation, or legal reference.

7. **User Interface & Display**

   The system offers a simple but functional live display window with real-time bounding boxes and annotations. This allows security personnel to monitor the environment visually while receiving intelligent assistance from the AI model.

Beyond these components, the system is modular and scalable. Developers can expand it further by integrating SMS/email alert systems, database logging, or linking it to access control mechanisms (e.g., locking doors when a threat is detected). Moreover, thanks to its compatibility with platforms like Raspberry Pi or NVIDIA Jetson Nano, the solution can be easily embedded into drones or standalone smart surveillance units.

The choice of YOLO as the backbone detection model was motivated by its balance of speed and accuracy. Unlike two-stage detectors like Faster R-CNN, YOLO performs detection in a single pass, making it suitable for real-time applications where delay can compromise safety. YOLOv5 and later iterations (like YOLOv11) also provide flexibility in input size, inference speed, and fine-tuning capacity.

The project also supports toggling between visible and thermal camera input manually. The dual-mode operation empowers security systems to work in almost all visibility conditions. For instance, during the night, a thermal camera would allow the system to track threats based on heat emissions, particularly useful for identifying bombs or overheating components that could lead to fires or explosions.

Importantly, the project is open-source, allowing customization and future improvements. New detection classes, improved datasets, or domain-specific training can further extend the utility of the system into fields like wildlife monitoring (detecting poachers), industrial safety (detecting fire hazards), or disaster management (finding heat signatures in collapsed structures).

ALGORITHM / PSEUDOCODE

BEGIN

1. IMPORT required libraries: OpenCV, NumPy, Tkinter, PIL, SQLite3, Logging, DateTime, etc.

2. SETUP logging to log errors into a file (error.log)

3. FUNCTION log_detection(conn, camera_index, object_class, confidence, weapon_detected)

   - INSERT detection record into SQLite database with timestamp

4. TRY TO load YOLO model to GPU

   - IF error occurs, log and display error message

5. FUNCTION init_db()

   - CREATE SQLite database and table if not exists

   - RETURN connection

6. CONNECT to the SQLite database

7. INITIALIZE camera list with:

   - Built-in webcam (index 0)

   - Mobile stream URL

8. SET default camera index to 0 and try to open it

   - IF fails, log and display error

9. INITIALIZE background subtractor for motion detection

10. CREATE "output" directory if not exists

11. INITIALIZE global variables:

- thermal_mode ← False
- recording ← False
- out ← None
- detection_count ← 0
- weapon_count ← 0

12. FUNCTION draw_detections(frame, results):

- FOR each detected object in results:
- IF confidence > threshold slider value:
    - INCREMENT detection_count
  - GET bounding box and object class
    - IF object is "knife" or "gun":
      - Set color to red
      - Show weapon warning
      - Set weapon_detected ← True
      - INCREMENT weapon_count
    - ELSE:
      - Set color to green
    - DRAW bounding box and label
    - LOG detection to database
- RETURN modified frame and weapon_detected

13. FUNCTION apply_thermal_effect(frame):

- CONVERT frame to grayscale
- APPLY thermal-like color map
- RETURN modified frame

14. FUNCTION toggle_recording():

- IF recording is False:
  - START recording

- Initialize VideoWriter with current time as filename
- ELSE:
  - STOP recording and release VideoWriter

15. FUNCTION update_video_feed():

-READ frame from camera

- IF thermal_mode is True:

- APPLY thermal effect

- DETECT objects using YOLO
- DRAW detections
- IF recording:
  - SAVE frame
- CONVERT frame for Tkinter
- UPDATE GUI image display
- REPEAT every 10ms

16. FUNCTION toggle_thermal_mode():
- TOGGLE thermal_mode flag
- UPDATE button label accordingly

17. FUNCTION switch_camera():
- SWITCH to next camera in list
- Release current camera
- Open new camera

18. FUNCTION update_statistics(results):
- UPDATE detection, weapon, and average confidence labels
- REPEAT every 1000ms

19. FUNCTION update_metrics():
- CALCULATE FPS based on elapsed time
- UPDATE status bar with FPS, detections, weapons

- REPEAT every 1000ms

20. CREATE main GUI window with Tkinter:
    - Add video label
    - Add stats display
    - Add detection threshold slider
    - Add thermal, switch camera, record buttons
    - Add menu bar (Settings, Help)
    - Add status bar

21. START metrics update loop

22. START video feed update loop

23. RUN Tkinter mainloop

24. ON EXIT:
    - RELEASE camera and video writer
    - DESTROY all OpenCV windows

END

IMPLEMENTATION

The successful implementation of the AI-Powered Target Recognition System involved the integration of several core components including machine learning, computer vision, hardware interfacing, and real-time video stream processing. The system was developed using **Python 3.10.15** as the primary programming language due to its wide range of libraries and community support for AI, image processing, and embedded systems. The development environment used was **Visual Studio Code (VS Code)**, and the model was trained and deployed using **YOLOv11l.pt**, a pre-trained version of the popular YOLOv5 model optimized for object detection tasks.

The following steps describe the major phases of system implementation:

1. **System Architecture**

   The architecture of the system is modular and follows a pipeline approach:

   a) **Video Feed Input** – Captured from either a standard USB webcam or a thermal camera.

   b) **Frame-by-Frame Analysis** – Each frame is processed in real-time.

   c) **Object Detection using YOLOv11l.pt** – The model identifies potential threats like guns, knives, or bombs.

   d) **Thermal Mapping** – For temperature anomalies and hidden threats.

   e) **Motion Detection** – For dynamic threat identification and reducing false positives.

   f) **Alert Generation** – Buzzer activation or on-screen warning.

   g) **Screenshot Storage** – Capturing frames containing detected threats for review.

2. **Model Integration – YOLOv11l.pt**

   The model used is **YOLOv11l.pt**, a variant of YOLOv5 optimized for real-time object detection. This model was selected for its speed, accuracy, and compatibility with both CPU and GPU environments. The weights were loaded using the **Ultralytics YOLOv5 framework** and run through the PyTorch backend. The system utilizes the detect.py interface with a modified post-processing script to:

   - Draw bounding boxes,
   - Display class labels (e.g., ‘gun’, ‘knife’, ‘bomb’),

- Filter confidence scores,
- And forward the detection to the alert mechanism.

The YOLO model was fine-tuned with additional images of weapons and bombs, including thermal images. This fine-tuning improved detection accuracy in low-light and occluded conditions.

3. **Thermal Camera Integration**

A key implementation aspect is the use of a **thermal camera** for bomb detection. The camera outputs a **thermal grayscale image**, which is color-mapped using OpenCV's COLORMAP_JET or COLORMAP_INFERNO to highlight high-temperature regions. This is essential in environments where visual detection fails—e.g., concealed bombs, obscured views, or nighttime monitoring.

The thermal feed is also processed by the YOLO model, allowing simultaneous RGB and thermal analysis. This dual modality adds redundancy and robustness, especially in critical scenarios like detecting hidden IEDs or overheating mechanical devices.

4. **Real-Time Processing with OpenCV**

OpenCV serves as the image processing engine. It captures frames, preprocesses them, feeds them into the detection model, and displays results with bounding boxes and alerts. The implementation includes:

- **VideoCapture** for webcam/thermal camera input.
- **Resize and Normalize** steps for model compatibility.
- **cv2.rectangle** and **cv2.putText** for annotation.
- **imshow** loop for real-time display.

Additional filters are applied to prevent repeated alerts for the same object and ensure fluid motion tracking of targets.

5. **Motion Detection Module**

A lightweight motion detection algorithm is integrated to detect changes between frames. This is especially helpful in:

- Reducing computational overhead by skipping frames with no movement.
- Triggering threat detection only when motion is identified, improving speed and response time. The algorithm compares pixel-wise differences and thresholds the change magnitude to activate the YOLO detection cycle.

6. **Alert System (Audio/Visual)**

   Upon threat detection, the system activates a **buzzer or siren** using GPIO pins (if deployed on Raspberry Pi or Jetson Nano), and displays an **on-screen warning**. Alerts are triggered only if:

   - The object detected is within the specified list (e.g., gun, bomb).
   - The confidence level exceeds a set threshold (usually >0.6).

   Simultaneously, the current frame is saved in a secure folder using a timestamped filename for future investigation or reporting.

7. **Screenshot Capture and Logging**

   Every detection event automatically captures a frame using cv2.imwrite() and logs the following:

   - Date and time of detection
   - Object type and confidence level
   - Source of detection (RGB or thermal)
   - Camera ID (if multi-camera system is deployed)

   This allows system operators to review security logs, improve datasets, and validate system performance during testing.

8. **Compatibility with Embedded Hardware**

   The system was tested and found to be compatible with:

   - **Raspberry Pi 4** (with camera module and optional Coral USB accelerator)
   - **NVIDIA Jetson Nano** (with CUDA-enabled GPU acceleration)

   These platforms enable real-time deployment in compact, low-power environments, perfect for remote or mobile surveillance.

   Python libraries such as torch, cv2, RPi.GPIO, and ultralytics are used, with minimal hardware requirements, making the system suitable for wide deployment.

9. **Optimization and Performance Testing**

   During implementation, the following measures were taken for optimization:

   - Batch frame skipping to reduce latency.
   - CUDA acceleration enabled for GPU setups.
   - Thermal and RGB image resolution reduced slightly to increase FPS without major accuracy loss.
   - Use of threading for parallel capture and detection.

   Testing showed real-time performance at **20–30 FPS** on a mid-tier GPU and **8–12**

FPS on the Jetson Nano using YOLOv11l.pt. The detection accuracy remained consistent across different environments and lighting conditions.

Deployment Notes

- The system can be deployed on any GPU-enabled PC.
- Can be migrated to embedded devices like Raspberry Pi or Jetson Nano with minor code adjustments.
- Requires a thermal camera for heat-based detection.
- Alert and data saving paths are customizable.

OUTPUT

Fig.1: Weapon Detection Using Thermal Camera Feed

Fig.2: Detected Knife with Confidence Level 0.87

APPLICATIONS

The AI-Powered Target Recognition System has a wide range of real-world applications in domains that require high levels of security, proactive threat detection, and situational awareness. Designed for scalability, modularity, and portability, the system can be deployed in various settings, from national defense to civilian safety. Below are some of the most relevant application areas:

1. **Military Surveillance and Defense**
   The system can be deployed on drones, surveillance towers, or mobile ground units for real-time monitoring of sensitive borders or conflict zones. Its ability to detect weapons and explosives using both RGB and thermal imaging makes it ideal for day-night operations and mission-critical environments.
2. **Airport and Metro Station Security**
   Airports and metro stations are high-risk zones where concealed weapons or bombs pose a serious threat. This system can be used to monitor entry points, luggage areas, and waiting zones to detect any suspicious objects automatically and trigger alerts without human intervention.
3. **School and Campus Safety**
   School shootings and campus violence have increased in recent years. This system can be installed at entrances, hallways, and corridors to provide an early warning mechanism if a student or intruder is carrying a weapon.
4. **Smart City Surveillance**
   Integrated into city-wide surveillance networks, the system can enhance public safety by monitoring streets, parks, and crowded areas. Using edge devices like Jetson Nano, this setup can provide intelligent threat detection without constant cloud processing.
5. **Banking and Financial Institutions**
   Banks and ATMs are common targets for armed robbery. This system can be used to detect unauthorized weapons and automatically lock access doors or trigger emergency alarms.

6. **Industrial and Warehouse Security**

   Large facilities can deploy the system to detect if workers are unintentionally carrying sharp tools or if there are overheating machines that could potentially explode, using the thermal vision feature.

7. **Disaster Response and Search Operations**

   Thermal detection can help identify hidden threats like bombs or heat signatures of survivors in collapsed buildings, making it useful for rescue operations after natural disasters.

8. **Government Buildings and Border Security**

   Sensitive government establishments can benefit from continuous monitoring to detect incoming threats from intruders or smuggled explosive devices.

CONCLUSION

The AI-Powered Target Recognition System represents a significant advancement in the field of intelligent surveillance and threat detection. By integrating real-time object detection models like YOLO with advanced imaging technologies such as thermal cameras, the system is capable of identifying potential dangers— including weapons and explosive devices—with remarkable accuracy and speed. This dual-mode capability ensures consistent performance in diverse environments, including low-light and obscured conditions.

One of the system's standout features is its real-time processing capability, enabling immediate responses through alarms and visual alerts. Its modular design ensures that it can be deployed on a wide range of hardware—from powerful GPU-enabled computers to compact edge devices like the NVIDIA Jetson Nano or Raspberry Pi. This makes it not only a powerful tool for law enforcement and military operations but also accessible for civilian applications such as school security and public infrastructure protection.

Moreover, the system's use of deep learning makes it highly adaptable. As new threats emerge, the model can be retrained or fine-tuned with updated datasets, keeping the system relevant and robust over time. Its open- source and customizable nature further encourages innovation and integration into broader smart surveillance ecosystems.

In a world where threats are becoming increasingly sophisticated, the need for intelligent, automated, and proactive surveillance systems has never been greater. This project offers a reliable and scalable solution to that need—bridging the gap between traditional monitoring and AI-driven security. It not only strengthens preventive measures but also significantly reduces the dependence on human vigilance, thus enhancing overall safety and efficiency.