Republic of Yemen

Emirates International University

Faculty of Engineering

AI Department

Bank Vault Protection and Surveillance System

AI-Based Intrusion Using Computer Vision

Team Members: Hamza Mahdi, ALI Marish,

Akram Al-Sa`Awani, Mohammed AL-Hazar

Supervisor: Prof. Ayman Al-Tina

Course: Computer Vision

Submission Date: 17 February

Repository Link: (hamza-mahdi/Bank-Vault-Protection-and-Surveillance-System-AI-Based-Intrusion-Using-Computer-Vision)

Abstract

This project presents a real-time intelligent protection and surveillance system designed for bank vault security. The system integrates motion detection, human detection, object tracking, automatic alarm activation, and intruder image capture using computer vision and artificial intelligence techniques.

The architecture prioritizes real-time responsiveness by elevating process priority, minimizing computational overhead during tracking, and dynamically switching between detection and tracking phases. The system captures evidence images, activates an alarm for a defined duration, and automatically resets after event completion.

The implementation demonstrates a complete end-to-end intelligent surveillance pipeline suitable for secure environments requiring autonomous intrusion monitoring.

1. Introduction

Bank vaults represent highly sensitive and high-risk security environments. Traditional surveillance systems rely heavily on manual monitoring, which increases response time and the risk of human error.

This project proposes a fully automated AI-driven monitoring system capable of:

- Detecting suspicious motion
- Verifying human presence
- Tracking intruders in real time
- Capturing evidence images
- Triggering and managing an alarm system

The system is designed to operate continuously, autonomously, and with minimal latency.

2. System Overview

The system operates through a sequential multi-stage pipeline:

1. Frame acquisition from a live camera.
2. Background modeling and motion detection.
3. Human verification using AI-based detection.
4. Object tracking after confirmation.
5. Alarm activation.
6. Snapshot capture for forensic documentation.
7. Automatic reset after tracking failure or alarm timeout.

The design ensures that computational resources are prioritized during active events and optimized during idle states.

3. System Architecture

The project consists of four core modules:

3.1 Main Control Module

This module manages:

- Video capture initialization
- Background modeling
- Motion analysis
- State transitions (Detection ↔ Tracking)
- Alarm duration control
- Snapshot storage
- Process priority elevation

It acts as the central coordination layer of the system.

3.2 Human Detection Module

Human detection is implemented using the Histogram of Oriented Gradients (HOG) descriptor combined with a pre-trained Support Vector Machine (SVM) classifier provided by OpenCV.

Operational Conditions for Human Detection

- Detection is applied only to Regions of Interest (ROI) generated by motion detection.
- ROI must satisfy minimum size constraints to avoid false positives.
- The ROI is resized before evaluation to improve detection consistency.
- Detection parameters are configured to balance speed and accuracy.

Human verification is required before alarm activation to prevent triggering due to non-human motion.

3.3 Tracking Module

After confirming human presence, the system initializes a CSRT (Discriminative Correlation Filter with Channel and Spatial Reliability) tracker.

Tracking Conditions

- Tracking is activated only after successful human detection.
- During tracking, repeated human detection is disabled to reduce computational load.
- If tracking fails, the system automatically stops tracking and returns to detection mode.
- The bounding box of the tracked intruder is continuously updated and visualized.

CSRT was selected due to its robustness in handling scale variation and partial occlusion.

3.4 Alarm Module

The alarm system operates in a separate execution thread to prevent blocking the main video processing loop.

Alarm Behavior

- The alarm starts immediately after confirmed human detection.
- It runs continuously for a predefined duration (30 seconds).
- The alarm can be stopped automatically after timeout.
- It can also be manually terminated through user input.
- The alarm resets when tracking is lost.

This multi-threaded design ensures uninterrupted real-time frame processing.

4. Motion Detection Mechanism

Motion detection is implemented using adaptive background subtraction.

Technical Details

- A running average background model is maintained.
- Each new frame is compared against the background.
- Absolute frame difference is computed.
- Thresholding is applied to isolate motion regions.
- Morphological operations are used to reduce noise.
- A minimum area threshold prevents triggering due to small disturbances.

Motion Constraints

- Motion must exceed 0.5% of total frame area.
- Background model update rate is controlled to adapt gradually to lighting changes.
- Small noise and environmental fluctuations are filtered.

This approach ensures that only significant movement is considered for further processing.

5. Evidence Capture Mechanism

When an intruder is confirmed:

- The system captures a snapshot of the detected human region.
- Images are saved in a dedicated directory.
- Filenames contain timestamps to preserve chronological order.
- Snapshots serve as digital forensic evidence.

Snapshot capture occurs only once per confirmed detection event to prevent redundancy.

6. Performance Optimization

To ensure real-time performance:

- The system elevates process priority to reduce latency.
- Human detection is disabled during tracking to conserve resources.
- Detection is restricted to motion-based ROI instead of the entire frame.
- Alarm runs in a separate thread to avoid blocking execution.

These design decisions ensure stable performance under continuous operation.

7. System Operational Workflow

Stage 1 – Idle Monitoring

The system monitors the scene and updates the background model.

Stage 2 – Motion Detection

Significant motion triggers ROI extraction.

Stage 3 – Human Verification

The ROI is analyzed using the HOG-based human detector.

Stage 4 – Intrusion Confirmation

If human presence is verified:

- Alarm activates.
- Snapshot is saved.
- Tracker initializes.

Stage 5 – Active Tracking

The intruder is tracked continuously.

Stage 6 – Termination

Tracking failure or timeout results in:

- Alarm deactivation.
- Tracker reset.
- Return to idle monitoring.

8. System Requirements

Hardware Requirements

- Surveillance camera with stable positioning
- Minimum 15–25 FPS capture rate
- Multi-core CPU recommended
- Adequate storage for snapshots

Software Requirements

- Python 3.x

- OpenCV
- NumPy
- psutil
- Operating system with camera access permissions

9. Security and Reliability Considerations

- Human verification reduces false alarms.
- Motion area threshold prevents noise-triggered events.
- Snapshot storage ensures traceability.
- Multi-threaded alarm design maintains system responsiveness.
- Automatic recovery after tracking failure increases reliability.

10. Limitations

- HOG-based detection may be less accurate in extreme lighting conditions.
- Performance depends on camera resolution and processor capability.
- Very small or distant subjects may not satisfy ROI constraints.
- Tracking may fail under full occlusion.

Despite these constraints, the system provides a functional and reliable intrusion detection mechanism for controlled environments.

11. Conclusion

This project successfully implements a complete AI-based vault protection system combining motion detection, human verification, object tracking, alarm control, and evidence capture.

The system demonstrates:

- Real-time autonomous monitoring
- Intelligent resource management
- Structured event lifecycle handling
- Practical forensic documentation capability

The integration of classical computer vision techniques with optimized execution design results in a reliable and deployable bank vault protection solution.