SMART SURVEILLANCE WITH MULTI-AGENT SYSTEMS

DRONES, CAMERAS, AND AI FOR URBAN SECURITY

Victor Jaziel Coronado FLores A01644090
Diego Iván Morales Gallardo A01643382
Luis Daniel García Espinosa A01643058
Jorge Antonio Arizpe Cantú A01637441
Michael Rice Radilla A00837460

THE PROBLEM

LARGE OPEN AREAS REQUIRE EFFECTIVE SECURITY SOLUTIONS.

Challenges:

- Insufficient coverage with limited resources.
- Delays in detecting and responding to threats.
- Human limitations in continuous monitoring.

Objective:

Develop a coordinated system to detect suspicious individuals and enable quick and efficient responses.

THE SOLUTION

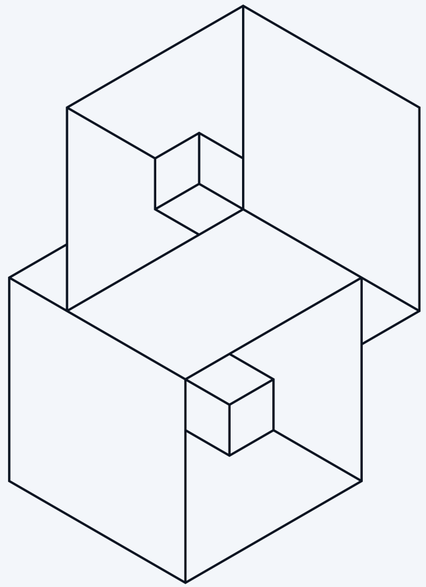
Intelligent Multi-Agent System:

- Autonomous Drones: Patrol and investigate suspicious individuals.
- Surveillance Cameras: Detect potentially dangerous individuals.
- Security Personnel: Analyze and respond to confirmed threats.

Key Features:

- Communication via efficient protocols (KQML).
- Real-time computer vision using YOLOv5.
- Decision-making based on auctions and voting mechanisms.

THE PROCESS



01 Cameras detect suspicious individuals using YOLOv5.

02 An alert is sent to the drone through a KQML message.

03 The drone moves to the detection area and streams live footage.

04 Security personnel analyze the footage, initiate a voting process, and decide on the response.

05 The drone returns to its base once the situation is resolved.

AGENTS AND ENVIRONMENT

Security

Evaluates and responds to alerts generated by the drone and cameras. Issues commands to drones and cameras after assessing alerts. Can initiate alarm or dismiss false alerts.

Drones

Autonomous agent that patrols the area and alerts other agents. Uses YOLO in order to detect suspicions. Shares location and status with other agents.

Camera

Static agent that monitors specific areas. Also uses YOLO and reports to other agents location of of thief.

Environment

Simulation of an open area with patrol zones and a drone landing station. Environmental factors (e.g., wind) integrated to test drone adaptability.

What We Measure

Metrics:

Patrol Time: Efficiency in covering the area.

Battery Usage: Resource management of the drone.

Confirmation Time: Speed in identifying and responding to threats.

Detection Accuracy: Success rate in identifying actual suspicious individuals.

Why It Matters:

Ensures system reliability.

Highlights areas for improvement in detection and resource allocation.

Simulation Success

When Is It Successful?

- A suspicious individual is detected and confirmed within the simulation.
- The drone has sufficient battery to return to the base safely.
- Personnel classify the situation correctly (real threat or false alarm).

Communication and Coordination Among Agents

Protocol: KQML (Knowledge Query Manipulation Language)

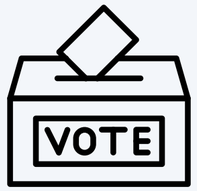
Key Actions:

- Call-for-Proposal: To assign tasks (e.g., drone patrol).
- Inform: To share detection results.
- Accept-Proposal: To confirm task allocation.

Auctions

The Contract Net Protocol (CNET) assigns tasks to the most capable agents.

Cameras and the drone compete for tasks based on battery level and proximity.



Voting

Validates whether a situation is a real threat or a false alarm.

Security personnel analyze images and decide collectively.

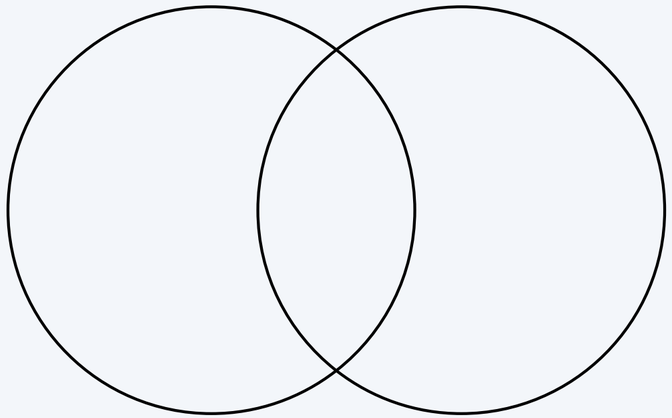
System Benefits

Ensures clear roles and quick decisions.

Optimizes resource use, such as drone battery and visual coverage.

Improves decision-making accuracy and fairness.

What Architecture Did We Use?



Deductive Architecture:

The system relies on clear rules and processes for decision-making.

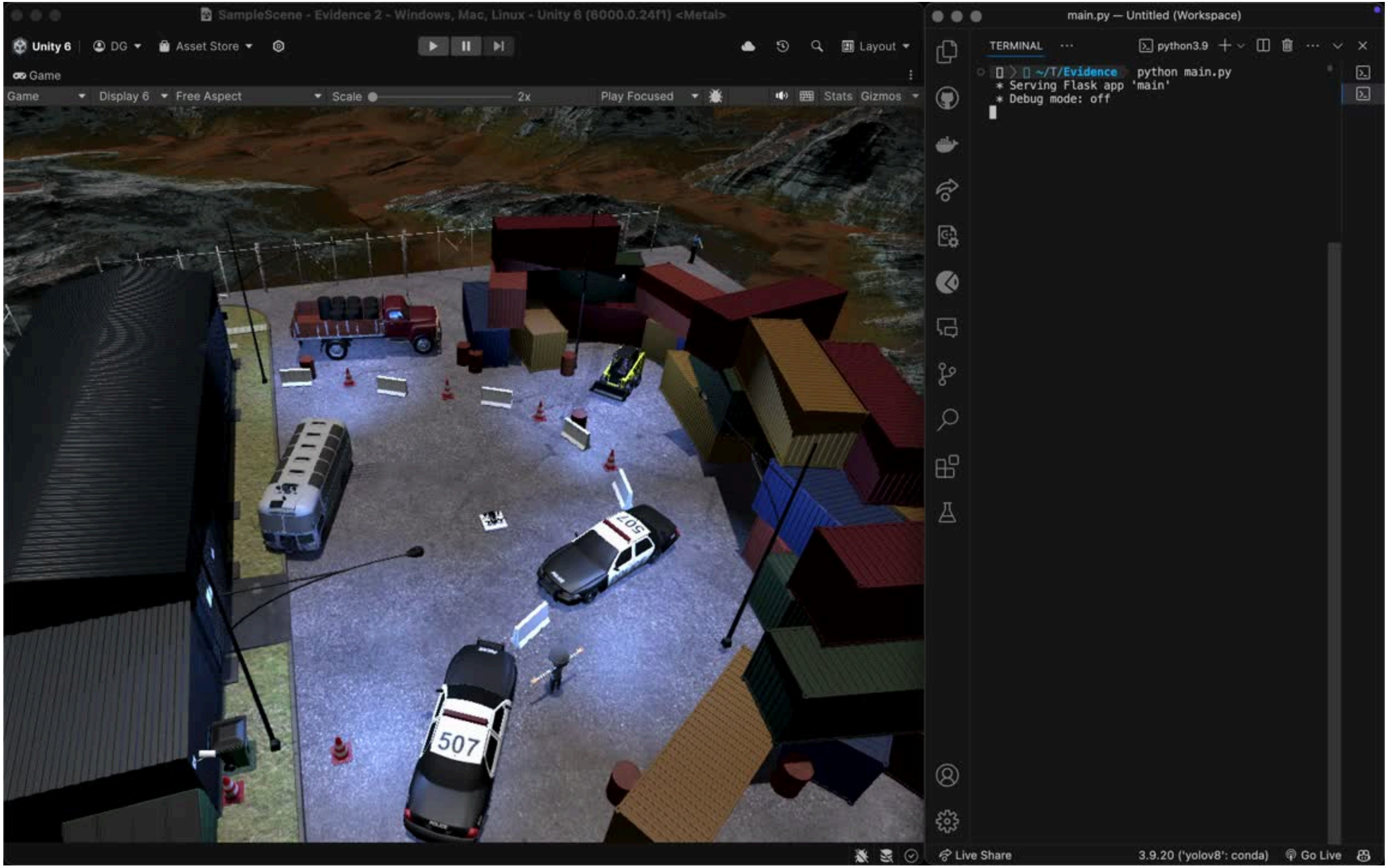
Example:

- The drone autonomously decides based on alerts and available resources.
- Human personnel complement the system by analyzing and validating threats.

Why It Works:

- Allows the drone to be autonomous in basic tasks.
- Ensures critical decisions are reviewed by humans.

Demo


SampleScene - Evidence 2 - Windows, Mac, Linux - Unity 6 (6000.0.24f1) <Metal>
Unity 6
DG
Asset Store
Layout
Game
Game
Display 6
Free Aspect
Scale
2x
Play Focused
Stats
Gizmos
507
main.py — Untitled (Workspace)
TERMINAL
python3.9
~/T/Evidence
python main.py
* Serving Flask app 'main'
* Debug mode: off
Live Share
3.9.20 ('yolov8': conda)
Go Live