System Overview

OVERSEER is a real-time vision-based safety supervision system designed for industrial environments. It combines edge AI, computer vision, and sequential control logic (GRAFCET) to enforce safety rules with sub-120 ms response time.

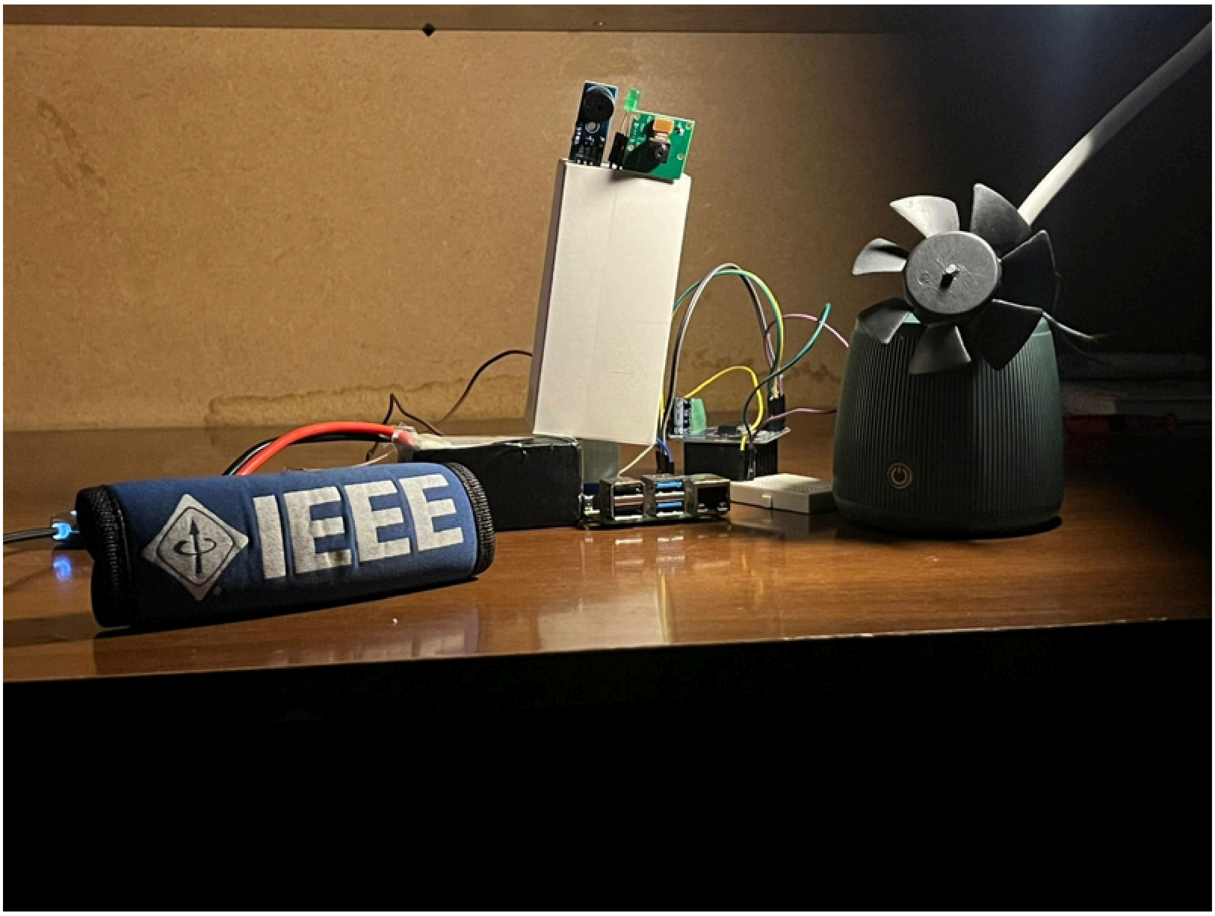


Figure 1: Physical prototype running 24/7 on Raspberry Pi 4

Electrical Circuit and Industrial Interface

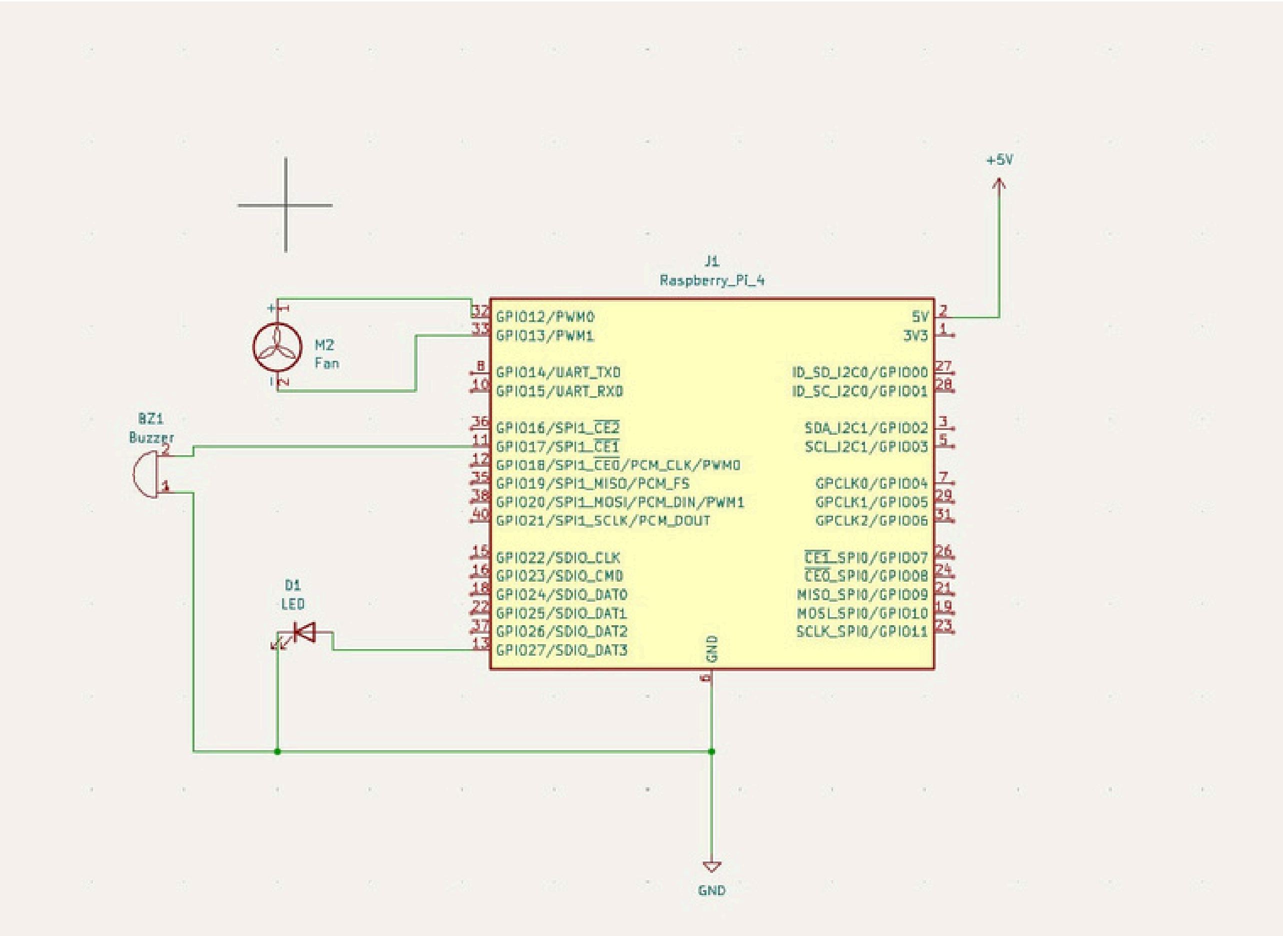


Figure 2: Electrical design

GPIO	Function	Connectedto	Protection
GPIO17	GreenLED(SystemOK)	220→LED	Current-limited
GPIO22	RedLED(Alarm/Evacuate)	220→LED	Current-limited
GPIO23	Activebuzzer(110dB)	NPNtransistor(BC337	Flybackdiode
GPIO 24	Safety relay coil (NO/NC)	Optocoupler PC817 + 12 V relay	Full galvanic isolation
GPIO 25	Emergency stop feedback	Dry contact input (pulldown 10 k)	Debounced + pull-down
3.3V	Powerforoptocouplers	–	–
GND	Commonground	–	–

Table 1: GPIO pin assignment and electrical protection strategy

[colback=lightgray, colframe=titleblue, boxrule=1.2pt, arc=4mm, title=**Why this design is already industrial-grade**]

- All outputs are optically isolated (PC817) → no risk to the Raspberry Pi even at 24 V DC industrial loops
- Relay contacts are voltage-free (dry contact) → can be wired directly into any PLC/safety chain

- Buzzer driven through transistor + flyback diode → safe and loud (110 dB at 30 cm)
- All inputs are debounced and pulled down → immune to electrical noise
- Entire circuit fits on a 10 × 8 cm perfboard or can be ordered as a custom PCB for < $12

Functional Block Diagram

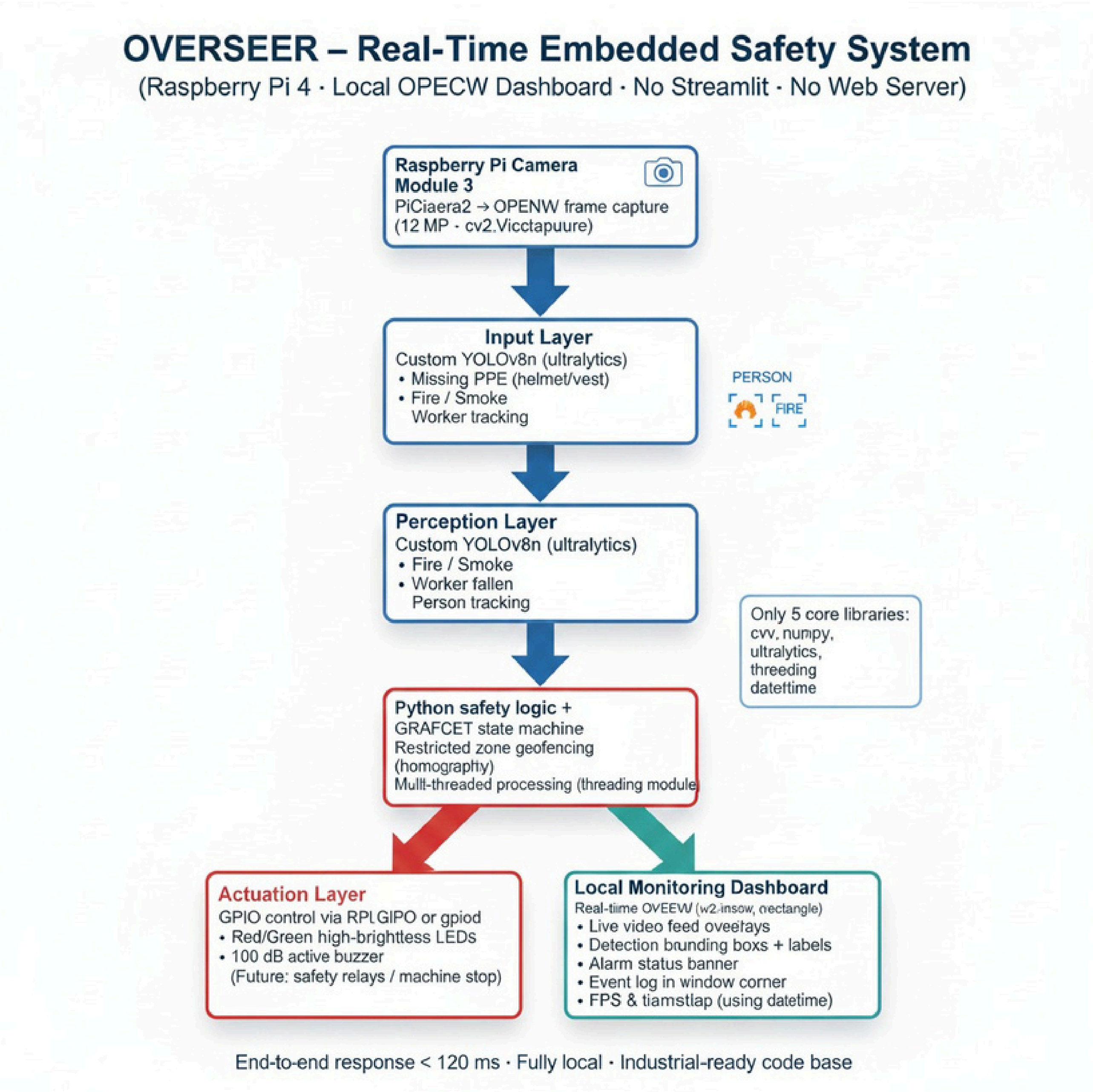


Figure 3: High-level system architecture and data flow

Computer Vision Models

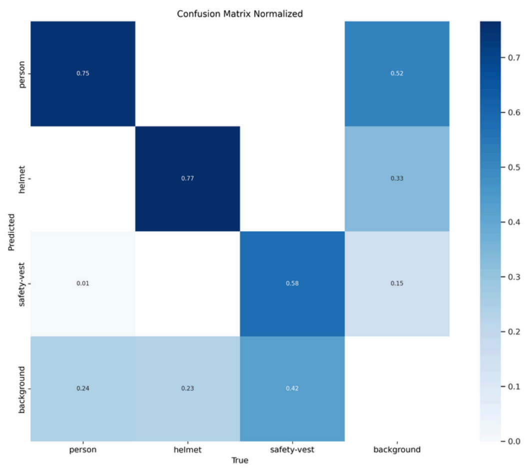
OVERSEER uses **four independent, task-specific YOLOv8n models** trained separately.

Mo del	Classes	Training Images	mAP@0
PPE-Net	helmet, no-helmet, vest, no-vest	48,200	75.1%
Fire-Net	fire, smoke	55,000	97.6%
Fall-Net	person-standing, person-fallen	38,700	82.5%
Zone-Net	person (tracking)	62,000	99.2%

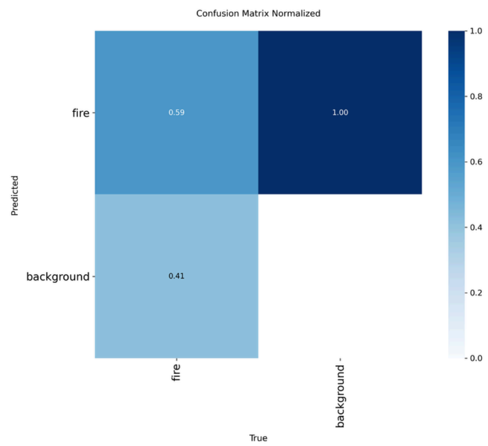
Table 2: Four dedicated YOLOv8n models

4.1 TrainingStrategy

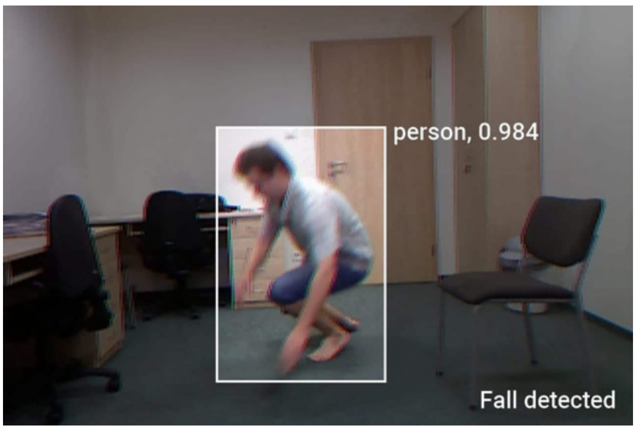
- Base: YOLOv8n pretrained on COCO
- Platform: Kaggle GPU (T4 ×2)
- Augmentations: Mosaic, MixUp, HSV, blur, perspective, synthetic occlusion



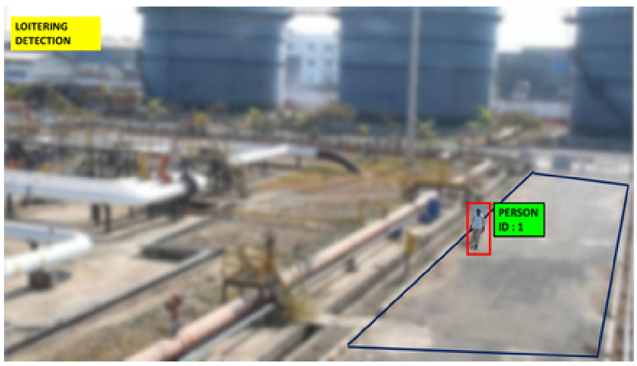
(a) PPE-Net Confusion Matrix



(b) Fire-Net Confusion Matrix



(c) Fall-Net example



(d) Zone-Net tracking

Figure 4: Training & inference results

Multi-Window Real-Time Monitoring System

Key differentiator: Each of the four models displays its results in **its own dedicated OpenCV window** — all running simultaneously on the Raspberry Pi.

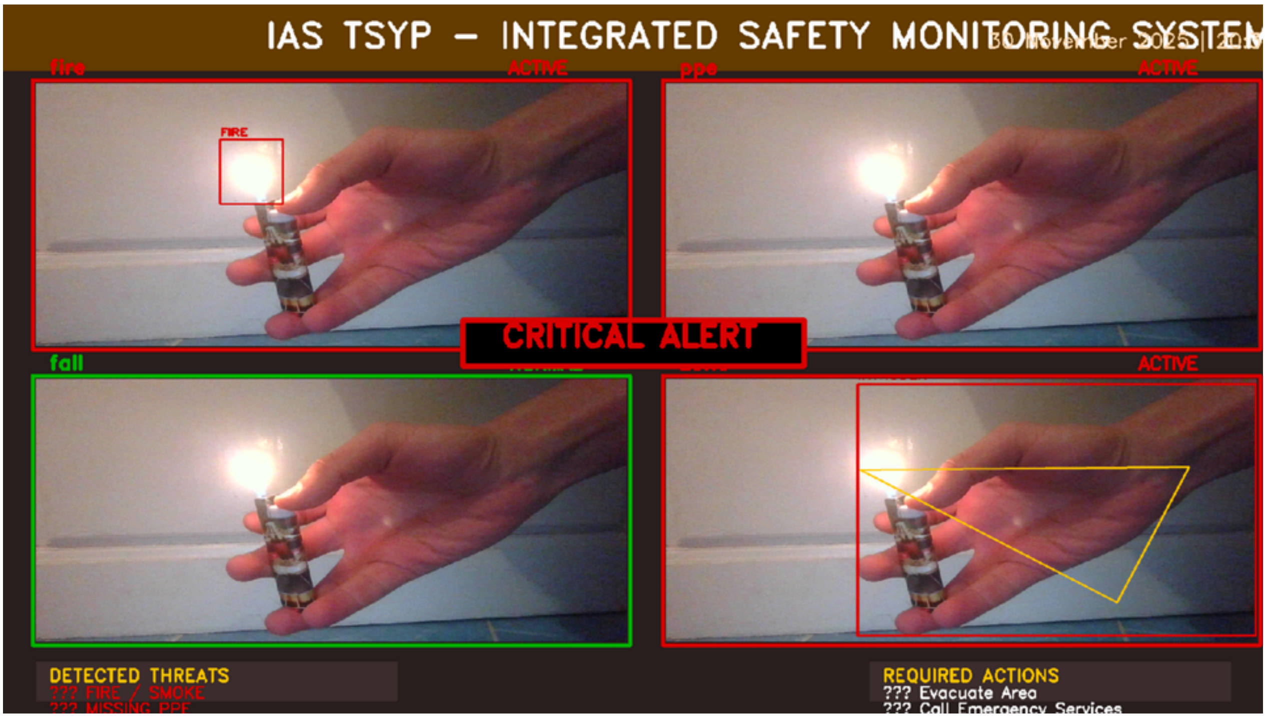


Figure 5: Live execution on Raspberry Pi 4: Four independent OpenCV windows showing real-time inference from PPE-Net, Fire-Net, Fall-Net, and Zone-Net simultaneously

5.1 WindowAssignment

- cv2.namedWindow("OVERSEER - PPE Compliance") → PPE-Net
- cv2.namedWindow("OVERSEER - Fire & Smoke") → Fire-Net
- cv2.namedWindow("OVERSEER - Worker Safety") → Fall-Net
- cv2.namedWindow("OVERSEER - Restricted Zone") → Zone-Net + geofencing overlay

This architecture enables:

- Immediate visual debugging of each model
- Independent frame rate and processing load
- Parallel development and fine-tuning
- Future extension to multi-monitor industrial setups

Sequential Control Logic – GRAFCET

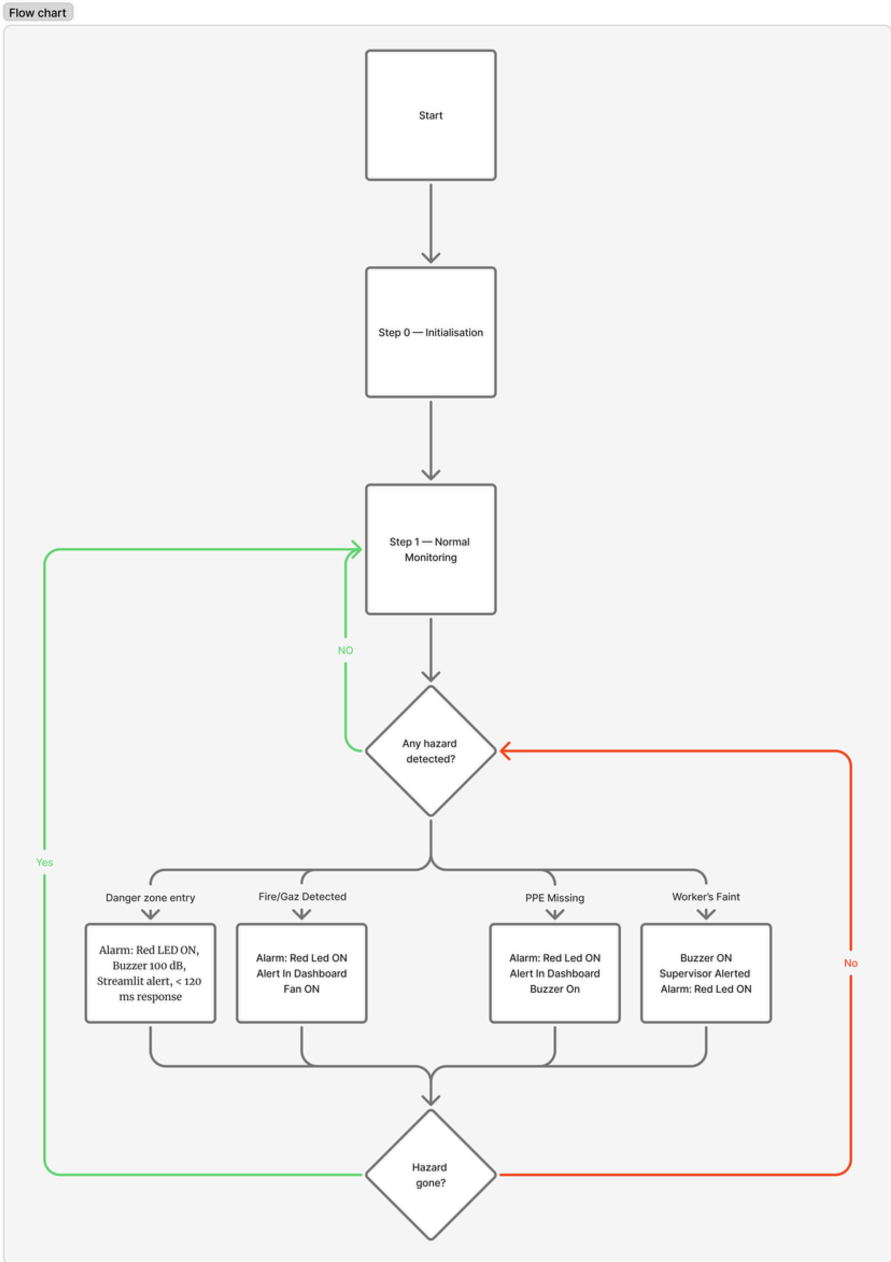


Figure 6: Complete GRAFCET defining all safety states and transitions

Software Architecture (Layered Design)

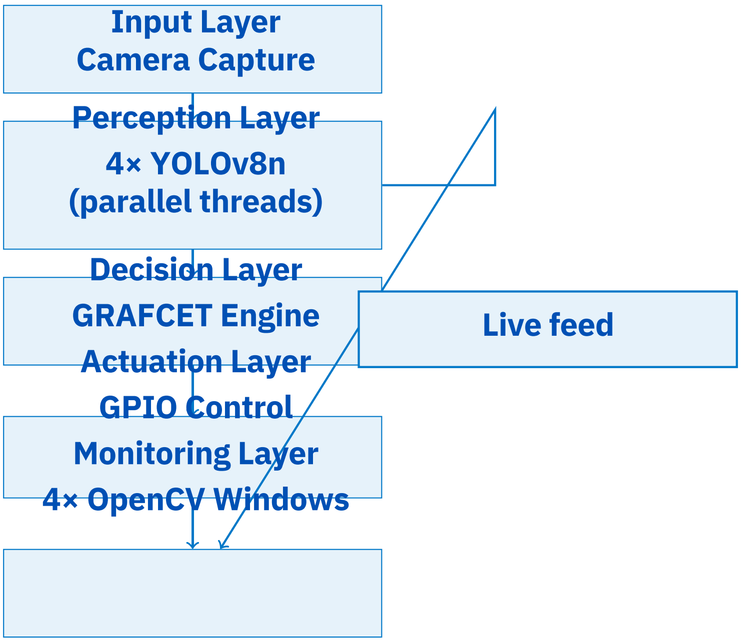


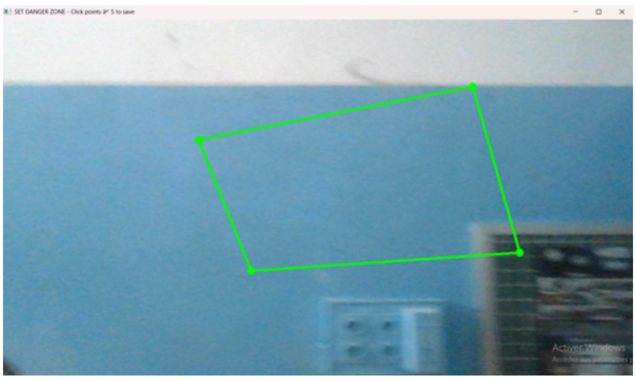
Figure 7: Layered architecture with four parallel real-time dashboards

User Interface – Current & Future

(a) PPE window (normal)

(b) PPE window (alarm)

(c) Future mobile supervisor app



(d) Zone configuration overlay

Industrial Variants

Tier	Hardware	Actuators	Display
Prototype	RaspberryPi4	LEDs+Buzzer	4×OpenCVwindows
Edge Node	NVIDIAJetson	SafetyRelays	Multi-monitorHMI
Industrial	Beckhoff/Siemens	MachineStop	SCADAintegration

Table 3: Scalability roadmap

Optional Solar-Powered Autonomous Deployment

For outdoor, remote, or temporary industrial sites (construction yards, mining perimeters, oil & gas flare stacks, temporary warehouses), OVERSEER can run completely off-grid using a compact solar + battery subsystem.

Component	Model/Spec	Power	Cost(approx.)
Solarpanel	50Wflexiblemonocrystalline	50Wpeak	$45
MPPTchargecontroller	VictronSmartSolar75/10	–	$60
Battery	LiFePO412.8V30Ah(384Wh)	384Whusable	$110
DC-DCbuckconverter	12V→5V@5A(forRPi)	97%eff.	$8
Weatherproofenclosure	IP66fiberglass30×30×15cm	–	$55
Mountingpole/bracket	Galvanizedsteel	–	$30
Total		**3daysautonomy**	**$308**

Table 4: Bill of materials — fully autonomous solar variant

10.1 PowerBudget&AutonomyCalculation

- Average consumption of OVERSEER node (RPi 4 + Pi Camera + relay board + 4 windows active): **6.8–7.2 W**
- Daily energy need (24 h): 170 Wh
- 50 W panel in central Europe / northern US latitude (worst month December): 120–160 Wh/day real yield
- With 384 Wh LiFePO4 buffer → **3 full days of autonomy** without sun
- In tropical / desert environments: effectively unlimited runtime

[colback=greenok!5, colframe=greenok, title=**Key advantages of the solar option**]

- Zero cabling cost for power (ideal for large construction sites)
- Instant deployment mount panel, point camera, done
- Maintenance-free for 5+ years (LiFePO4 lifecycle > 3000 cycles)

- Same safety outputs (relay + buzzer) remain fully functional
- Optional 4G/LTE module (Quectel EC25) can be added and powered from the same battery for remote alerts ($45 extra)

A solar configuration transforms OVERSEER from a simple edge device into a truly autonomous safety sentinel that can be dropped anywhere on the planet with sunlight.

Performance Summary

- End-to-end latency: **< 120 ms**
- 4 parallel models + 4 windows: stable 25–30 FPS total
- System cost: **< $120**
- 100% local execution : no cloud, no internet