CANELA: An AI-Driven Edge Camera System for Active Object Tracking in Intelligent Surveillance Systems

José Manoel Silva, Leonardo Lima, Stefano Loss, Thais Batista, Everton Cavalcante, Aluizio Rocha Neto, Daniel Araújo, Eduardo Nogueira Cunha, Nélio Cacho, Frederico Lopes

Federal University of Rio Grande do Norte, Natal, Brazil

{manoel.freitas.071, leonardo.lima.163, everton.cavalcante}@ufrn.br
{momoloss10, thaisbatista}@gmail.com, neliocacho@dimap.ufrn.br
{aluizio, daniel, eduardo, fred}@imd.ufrn.br

Abstract. *Intelligent surveillance systems have been increasingly relying on edge-based Artificial Intelligence for real-time object detection and tracking. However, fixed cameras suffer from a limited field of view and degraded image quality at long distances, while resource-constrained edge devices struggle to run complex vision models without incurring high latency and thermal issues. This paper presents CANELA (Camera with Autonomous Navigation and Envisioning Led by AI), a low-cost pan-tilt edge camera system for active object tracking. CANELA combines a custom gimbal, parallel control algorithms, and a Raspberry Pi 5 with a Hailo-8L M.2 Entry-Level accelerator to enable efficient on-device inference and dynamic camera repositioning to keep targets centered. Experimental results in a face-detection pipeline leveraging hardware acceleration demonstrate that CANELA maintains high frame rates while preserving CPU availability by offloading neural network inference to the accelerator. Compared to fixed cameras, it increases the duration of high-quality target capture, improving detection performance and tracking stability.*

1. Introduction

Intelligent surveillance systems (ISS) leverage artificial intelligence (AI), computer vision, and Internet of Things (IoT) devices to analyze video and audio data, enabling real-time proactive monitoring while reducing human intervention and accelerating incident response [Kim et al. 2010]. These systems augment traditional camera systems by incorporating capabilities such as facial recognition, object detection and tracking, and behavior analysis to identify threats and trigger alerts. Supported by advances in edge computing, communication networks, and deep learning, modern ISS have been widely applied in domains such as smart cities, transportation, industry, and home security [Adrian et al. 2018].

Despite these advances, ISS still face significant challenges in dynamic environments, particularly in reliably tracking moving objects while maintaining high image quality. Traditional fixed cameras are constrained by limited fields of view (FoVs) and inherent visibility gaps, preventing adaptation to rapid changes in target trajectory [Hsia et al. 2022]. Moreover, running continuous, complex computer vision models on resource-constrained edge devices introduces substantial computational overhead, often resulting in high latency and CPU thermal throttling that degrade system responsiveness and tracking stability [Benoit-Cattin et al. 2020, Sharma and Kanwal 2024]. These limitations prevent current ISS solutions from simultaneously achieving wide-area coverage, real-time responsiveness, and efficient edge processing.

To address these issues, this paper introduces CANELA (*Camera with Autonomous Navigation and Envisioning Led by AI*), an intelligent pan-tilt camera system designed as a low-cost edge IoT solution. CANELA integrates a customized 3D-printed gimbal, parallel control algorithms, and a Raspberry Pi 5[1] coupled with a Hailo-8L M.2 Entry-Level AI acceleration module[2] to enable efficient on-device inference. The system autonomously adjusts its orientation to keep the target centered in the frame, overcoming the restricted FoV of traditional fixed cameras, particularly at close range [Niu et al. 2022]. Furthermore, by offloading neural network inference to the acceleration module, CANELA minimizes CPU utilization and prevents thermal throttling, ensuring stable real-time performance and precise tracking.

Face detection and recognition represent applications that can significantly benefit from CANELA's dynamic tracking capabilities [Bashir and Arief 2025]. By continuously adapting the FoV, the system increases the likelihood of capturing high-quality frames of moving targets. To evaluate this capability, we implemented a face-detection pipeline and conducted an experimental analysis comparing the CANELA prototype and a traditional fixed camera under single- and multi-person scenarios. In our scenario, face detection is treated as a downstream task to assess tracking effectiveness. The focus is not on model accuracy but on how active camera motion improves reliable detection.

The contribution of this paper is twofold. First, it shows that active object tracking mitigates the FoV limitations of static cameras at short distances, significantly increasing usable frame capture. Second, it demonstrates the efficiency of the CANELA architecture by offloading neural network inference to a dedicated AI accelerator, yielding longer, high-quality capture durations and improved detection performance and tracking stability compared to fixed cameras.

The remainder of this paper is organized as follows. Section 2 presents the background of this work. Section 3 discusses related work. Section 4 describes the CANELA architecture. Section 5 describes the face-detection use case. Section 6 presents the experimental results. Section 7 brings some conclusions.

2. Background

This section introduces the background of this work, focusing on pan-tilt camera systems for active object tracking and the role of dedicated AI accelerators in overcoming the computational limitations of edge devices.

Camera pan-tilt. Fixed cameras in ISS are limited by narrow FoVs and visibility gaps, reducing their ability to capture moving targets clearly. To address these limitations, pan-tilt and pan-tilt-zoom (PTZ) cameras offer mechanical flexibility through horizontal (pan) and vertical (tilt) movements, enabling real-time adjustment of the viewing angle [Bashir and Arief 2025]. By actively tracking objects, these systems maintain the target within the frame, increasing the time available for accurate analysis by AI models.

Autonomous tracking requires tight integration between hardware and software

[1] https://www.raspberrypi.com/products/raspberry-pi-5/

[2] https://hailo.ai/products/ai-accelerators/hailo-8l-m-2-ai-acceleration-module-for-ai-light-applications/

components. Typically, servomotors are mounted on a gimbal control camera orientation, while computer vision algorithms process the video stream to locate the target. The system computes the spatial error between the target centroid and the image center, which is then used as input to a proportional-integral-derivative (PID) controller, a common approach for controlling machines and processes that require continuous control and automatic adjustment. Based on this error, the controller continuously adjusts motor actuation for smooth, stable camera movements that keep the target centered and avoid overshoot [Niu et al. 2022].

AI acceleration for edge devices. Real-time computer vision in edge environments requires running computationally intensive deep learning models. Although modern microcomputers, such as the Raspberry Pi 5, offer multi-core processing, general-purpose CPUs remain inefficient for sustained AI inference workloads. As a result, running continuous vision models solely on CPUs often incurs higher latency and reduced system performance [Yu et al. 2025].

To overcome these limitations, edge-computing-based systems are increasingly using dedicated AI accelerators like GPUs or NPUs, which optimize tensor operations and handle neural networks more efficiently than CPUs. By interfacing with the host device via high-speed connections (e.g., PCIe 3.0), they offload inference tasks and free the main processor to handle control logic and sensor management. A representative example of this approach is the Hailo-8L accelerator, which delivers high Tera-Operations Per Second (TOPS) with low power while supporting higher frame rates and reduced thermal throttling. Combining AI acceleration with edge computing architectures is essential for enabling real-time, responsive, and scalable ISS.

3. Related Work

ISS are using AI processing at the network edge to enable real-time monitoring while reducing reliance on cloud infrastructure and the consequent latency [Bashir and Arief 2025, Wang et al. 2025]. Traditional surveillance depends on static camera configurations, such as fixed-point monitoring systems, which means they miss parts of the scene. Convolutional Neural Networks (CNNs) work well for image recognition [Bashir and Arief 2025], but their success in surveillance depends on capturing relevant visual information, which fixed cameras cannot always guarantee.

To overcome these limitations, recent research has explored active camera systems, particularly PTZ configurations. Niu et al. [Niu et al. 2022] and Bashir and Arief [Bashir and Arief 2025] employ PID-based control to automatically track moving targets and keep them within the camera's FoV. Similarly, Garcia et al. [Garcia et al. 2024] propose a tracking system based on embedded platforms, such as NVIDIA Jetson, for face recognition.[3] These approaches demonstrate the advantages of active tracking, but they often rely on computationally intensive processing pipelines.

Continuous computer vision workloads on edge devices pose substantial computational and energy challenges. Although techniques such as hardware acceleration and lightweight neural network optimization reduce inference latency, existing work does not address the impact of sustained workload on thermal behavior and long-term performance stability. This issue is critical in real-time surveillance, where long operations

[3]https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/

can cause thermal throttling and degrade performance. Specialized hardware, such as FPGAs [Singh et al. 2017], can achieve low latency in active tracking systems, but the increased cost and complexity undermine scalability for widespread edge-based surveillance systems.

In summary, the literature reveals a gap in (i) low-cost hardware suitable for large-scale deployment, (ii) continuous active tracking through mechanical camera control, and (iii) thermally stable, real-time AI inference at the network edge. To address this gap, we propose CANELA as an affordable pan-tilt IoT camera powered by a Raspberry Pi 5 and the Hailo-8L M.2 Entry-Level AI accelerator. By offloading neural network inference to the accelerator, the system reduces CPU load, prevents thermal throttling, and maintains stable, real-time performance and precise tracking.

4. CANELA

We designed CANELA as an AI-driven edge-camera system for autonomous object tracking in ISS. Operating as an edge IoT node, the system enables real-time object tracking while maintaining low latency and efficient resource utilization. Its modular architecture supports deployment across multiple domains, including traffic monitoring for vehicle tracking and flow analysis, industrial inspection to identify defects, and smart city applications such as monitoring public spaces and infrastructure.

The computational core of CANELA is based on a Raspberry Pi 5 equipped with 8 GB of RAM and a quad-core ARM Cortex-A76 CPU (Broadcom BCM2712 SoC), with an approximate cost of $80. To enable efficient AI inference, the platform is augmented with a Hailo-8L M.2 Entry-Level accelerator, capable of delivering up to 13 TOPS with a power consumption between 1.5W and 2.5W. The accelerator is connected via the Raspberry Pi M.2 HAT+ PCIe 3.0 interface, which approximately provides 900 MB/s of bandwidth, critical for reducing inference latency.

For video acquisition and tracking, CANELA incorporates an autonomous pan-tilt camera mechanism that dynamically adjusts the FoV to follow moving targets, thereby significantly improving tracking continuity and increasing the likelihood of capturing high-quality frames. The employed camera is a high-definition webcam operating at 30 frames per second (FPS), aiming to balance image quality and computational efficiency while ensuring sufficient detail for object detection. The following subsections detail the hardware components, control pipeline, and AI acceleration process.

4.1. CANELA Components

The CANELA prototype is housed in a gimbal chassis manufactured using 3D printing with PLA filament. This mechanical structure enables camera movement along the horizontal and vertical axes, with angular ranges of 360° (pan) and 180° (tilt), respectively. The model used in this work is available on Thingiverse[4] and is licensed under Creative Commons (CC BY-NC-SA). The assembly consists of 12 parts, including structural supports and gear components, as illustrated in Figure 1. Components *a*, *b*, and *c* form the main structural housing, which accommodates the motors and protects internal components. Components *d*, *e*, and *f* support the camera mount, while elements *g* through *l*

[4]https://www.thingiverse.com/thing:314027

implement the gear reduction system responsible for transmitting motor motion to the gimbal axes.

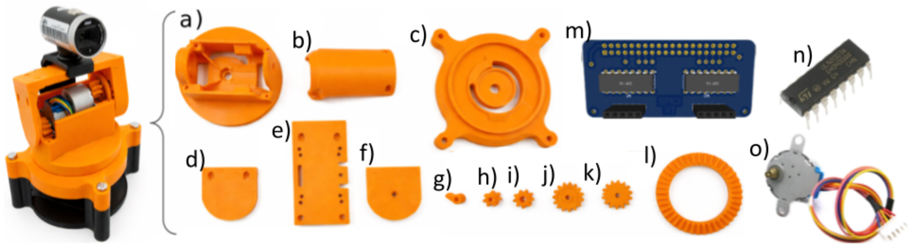

Figure 1. Gimbal parts printed in 3D with PLA filament: (a) main housing (body); (b) housing cover; (c) arm plate or base; (d) lateral support; (e) camera base plate or mount; (f) lateral support; (g) pointer; (h) 7-tooth gear; (i) 9-tooth gear; (j) 10-tooth gear; (k) 14-tooth gear; (l) 39-tooth gear. Hardware setup: (m) driver circuit; (n) ULN2003 module; (o) 28BYJ-48 stepper motor.

The gimbal is actuated by two 28BYJ-48 stepper motors controlled via ULN2003 driver modules, which run at 5V and are common in IoT prototyping for their simplicity. The control circuit connects to the Raspberry Pi 5 through a 26-pin header and dedicated motor-control connectors. This hardware setup, shown in Figure 1 (components *m*, *n*, and *o*), enables the precise, repeatable camera positioning required for active tracking. We have chosen stepper motors over servo motors due to their low cost, ease of control, and ability to provide precise incremental positioning without complex feedback, making them suitable for real-time edge-based tracking applications.

4.2. Object Tracking Control Loop

At the software level, CANELA implements a continuous control loop that adjusts camera orientation based on the detected position of an object of interest, enabling responsive object tracking at the edge. We have chosen to adopt a custom PID controller algorithm because our stepper motors operate in discrete steps, eliminating the need for complex closed-loop corrections and minimizing processing overhead. Our algorithm directly translates the AI-processed visual offset into exact step commands, keeping the target centered while enabling continuous tracking and target switching.

As illustrated in Figure 2, each iteration begins with frame acquisition, followed by object detection executed on the Hailo 8L M.2 Entry Level accelerator. Upon detection, the identifiers of all objects are populated into a dynamic tracking list. If multiple objects are present, the system processes them sequentially, prioritizing the first element in the list and initiating the tracking sequence, while actively adjusting the camera to centralize the object. We made this design choice for sequential processing to simplify the control logic and minimize computational overhead.

To ensure stable tracking, the centralization process is maintained for 10 consecutive frames. We determined this value empirically through initial experiments, starting with a 30-frame hold and progressively reducing it to optimize responsiveness until reaching a balance between sufficient visual stability for downstream tasks and fast camera transitions between multiple targets. After this period, the object is considered actively tracked, and relevant information is processed before the loop continues. Once the tracking cycle for the current object is completed, the pipeline advances to the next element

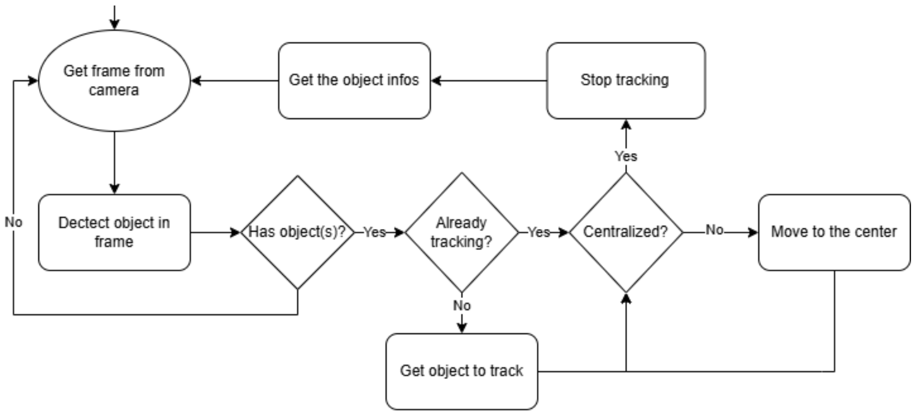

Figure 2. Flowchart of the camera control and object tracking pipeline

in the list and repeats the centralization process for all detected targets. If no objects are detected or once all elements in the tracking list have been processed, the system enters a standby state for 90 frames. If the absence of targets persists, the camera returns to its original position by reversing previous movements, ensuring a consistent reference state before restarting the acquisition cycle.

4.3. Hailo 8L M.2 Entry-Level AI Acceleration Module

CANELA leverages the Hailo 8L M.2 Entry-Level AI acceleration module to efficiently run deep learning models at the network edge. Unlike conventional platforms, the Hailo 8L M.2 Entry-Level requires models to be compiled into the proprietary Hailo Executable Format (.HEF), rather than using standard formats such as ONNX or TensorFlow.

Figure 3 illustrates the model conversion workflow, which consists of three stages implemented within the Hailo AI Software Suite.[5] First, the trained model is converted to the Hailo Archive (.HAR) format, preserving the network structure and parameters. Next, the model is optimized and quantized to INT8 precision. Although quantization may introduce minor accuracy trade-offs, calibration techniques are applied to minimize performance degradation, which requires a representative dataset of at least 1,024 samples matching the model's input characteristics. Finally, the optimized model is compiled into the deployable (.HEF) format. This conversion enables efficient execution on the accelerator, significantly improving inference performance, reducing CPU load, and enabling stable real-time operation.

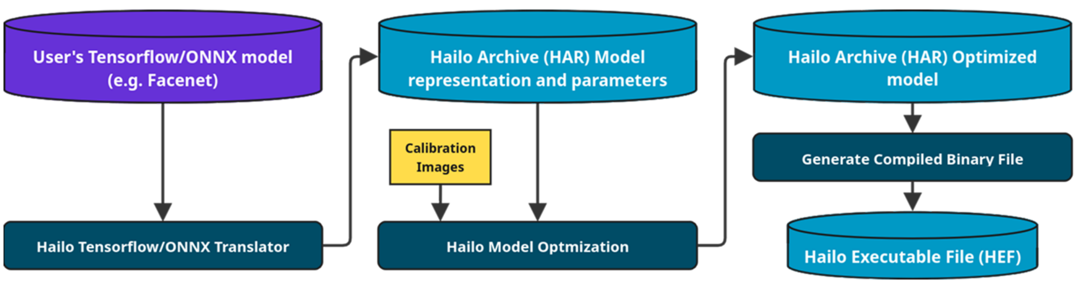

Figure 3. Hailo 8L M.2 Entry-Level conversion workflow

[5] https://hailo.ai/products/hailo-software/hailo-ai-software-suite/

5. Object Tracking Use Case: Face Detection

This section presents a face detection-based use case to demonstrate CANELA's capabilities. Face detection is a representative object-tracking scenario for ISS as it involves dynamic targets whose positions, scales, and motions vary over time, thus making it suitable for assessing how effectively the system maintains continuous target tracking under realistic conditions. This setup allows assessing how variations in target motion, camera distance, and spatial position affect tracking performance and system responsiveness.

Neural networks and integration with the Hailo 8L M.2 Entry-Level accelerator. The proposed face-detection solution integrates the SCRFD[6] [Guo et al. 2022] and FaceNet[7] [Schroff et al. 2015] neural networks, both open-source and distributed under the MIT License. In this work, we use the SCRFD-500m variant, a lightweight model optimized for mobile and edge environments with approximately 626,400 parameters. This model detects faces and provides spatial bounding boxes, which serve as the primary input to the tracking control loop. FaceNet is incorporated as a secondary, downstream component to generate 128-dimensional embeddings from detected faces, enabling the search for a face in a vector database. It is important to note that evaluating the accuracy of face recognition is out of the scope of this work.

Both models are converted using the process described in Section 4.3 for execution on the Hailo-8L M.2 Entry-Level, making all computationally intensive inference tasks offloaded to the accelerator. Raspberry Pi's CPU is responsible for lightweight operations, including post-processing SCRFD outputs and performing L2 normalization (Euclidean norm) on FaceNet embeddings. These embeddings are then queried against a pre-registered dataset using a vector database to enable identification and trigger notifications. Although this implementation focuses on face detection, the pipeline is model-agnostic and can be extended to other object detection tasks supported by the Hailo framework, thereby reinforcing CANELA's applicability to object tracking.

Pipeline implementation. The implemented pipeline controls CANELA's pan-tilt mechanism using the detected face position as the primary input. As illustrated in Figure 4, the system continuously captures frames and processes them using the Hailo-8L M.2 Entry-Level AI accelerator. In the first stage, the SCRFD model detects faces and outputs bounding boxes. These detections are then optionally processed by FaceNet to generate embeddings. Because all inference operations are executed on the accelerator, the system maintains high frame rates while avoiding CPU overload and thermal degradation.

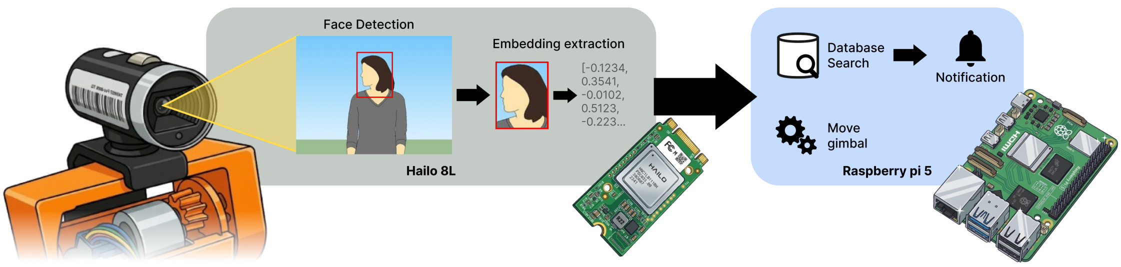

Figure 4. Pipeline implementation for face detection

[6] https://github.com/deepinsight/insightface
[7] https://github.com/davidsandberg/facenet

Once a face is detected, the Raspberry Pi computes the spatial error between the bounding box's centroid and the image's center. This error is used as input to parallel PID control threads, which determine the required motion of the stepper motors. The system then adjusts the camera orientation in real time to keep the tracked face centered in the frame. By maintaining the target within the FoV, CANELA increases the temporal window for detection and downstream processing. This behavior directly improves tracking continuity and highlights the benefits of active camera control in dynamic environments.

6. Evaluation

We conducted an experiment comparing a traditional fixed camera and CANELA to evaluate the effectiveness of the active object-tracking mechanism using the face-detection use case. The goal is to assess how autonomous camera motion improves tracking performance, particularly in maintaining target visibility, increasing the number of usable frames, and enhancing the reliability of downstream perception tasks. While the fixed camera remained static, CANELA actively tracked the target and maintained the face centered in the frame.

Experimental setup. We conducted the evaluation in a controlled indoor environment with stabilized lighting to minimize external variability. To ensure a fair comparison, both the fixed-mount camera and the CANELA system used identical hardware: a Raspberry Pi 5 with 8 GB of RAM and a Microsoft LifeCam HD-5000 webcam operating at 640×480 pixel resolution and 30 frames per second (FPS). We positioned both systems side-by-side at a height of 95 cm above the floor and operated them simultaneously.

We defined two experimental setups, one with a single participant and another with two participants. In each case, we performed the experiments at distances of 2 to 6 meters (see Figure 5). We also evaluated two movement profiles. In the first profile, participants moved laterally while maintaining a frontal orientation toward the camera. In the second profile, participants moved laterally while maintaining a fixed forward gaze relative to their initial position, simulating a side-view scenario. In all cases, subjects traversed a 4.10-meter-wide path for 35 seconds.

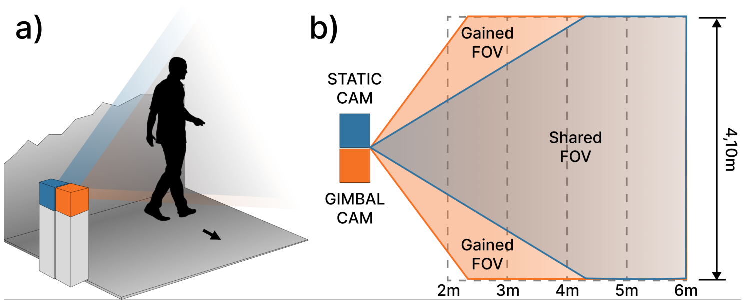

Figure 5. (a) Experimental scenario; (b) Comparative diagram of the FoV between the fixed (static) and moving (gimbal) cameras.

During the experiments, we logged data at the frame level to capture tracking performance, system behavior, and hardware efficiency. For each processed frame, the system recorded identifiers, frame resolution, and experimental configuration parameters, including device type (static or gimbal), number of participants, distance, and facial orientation. Performance metrics included total pipeline processing latency, number of detected faces per individual, and confidence scores for vector database matching. Additionally, we collected telemetry data to monitor CPU and accelerator temperatures, CPU utilization, and RAM consumption, all synchronized using absolute Unix timestamps.

Results and analysis. In both single- and two-person scenarios, both systems exhibited similar performance at distances of 5 and 6 meters across both movement profiles. This suggests that, at longer distances, both systems operate under comparable effective FoVs, reducing the relative advantage of active tracking. At intermediate distances (4 meters), both systems showed improved detection rates. However, we observed clear differences in cases below 3 meters. In the single-person scenario, CANELA consistently captured a higher number of frames at 2 and 3 meters for both frontal and side-view profiles (see Figure 6). This result highlights the advantage of active tracking in mitigating FoV constraints, particularly when the target occupies a larger portion of the scene.

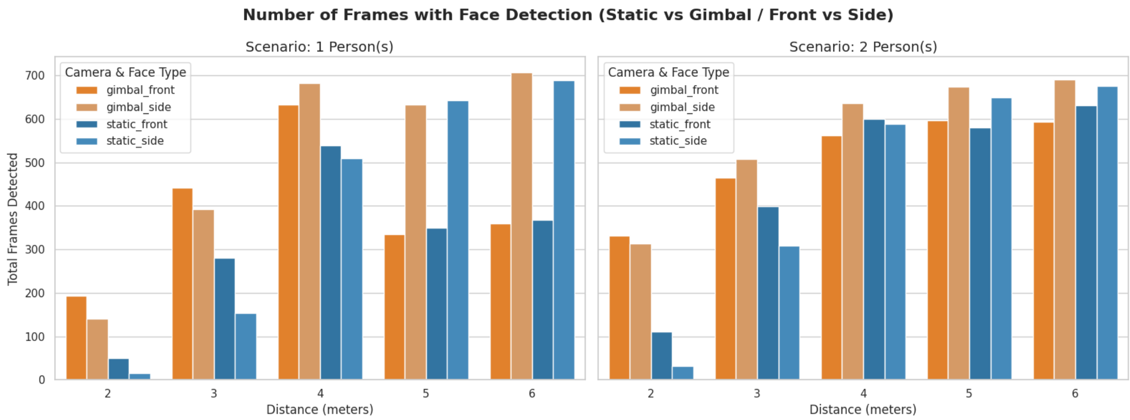

Figure 6. Comparison between the number of frames with face detection

In the two-person scenario, the performance gap became more noticeable at closer distances. The static camera experienced a significant drop in detections, especially for side-view movements, because subjects frequently moved partially out of the FoV. In contrast, CANELA maintained more stable detection rates by dynamically adjusting its orientation to follow the targets. These results demonstrate the effectiveness of active tracking in handling multi-target scenarios under spatial constraints. The gimbal-based system achieved more successful detections, with 4,514 and 5,372 frames in the single- and two-person scenarios, respectively, compared to 3,593 and 4,575 detections for the static camera. Both systems captured a similar total number of frames (22,054 for CANELA and 22,645 for the static camera), indicating that the observed improvements are due to more effective tracking rather than increased frame acquisition.

Figure 7 presents the number of faces successfully associated with identities in the database. The distribution closely follows the detection results shown in Figure 6, indicating that most detected faces were assigned an identity label. This behavior primarily reflects the permissive matching threshold in the vector database used rather than the recognition model's performance, which is not the focus of this work. In practical deployments, stricter confidence thresholds would be required to mitigate false positives.

Figure 8 shows the average pipeline processing time per frame. In the single-person scenario, both systems exhibit shorter processing times at longer distances, which

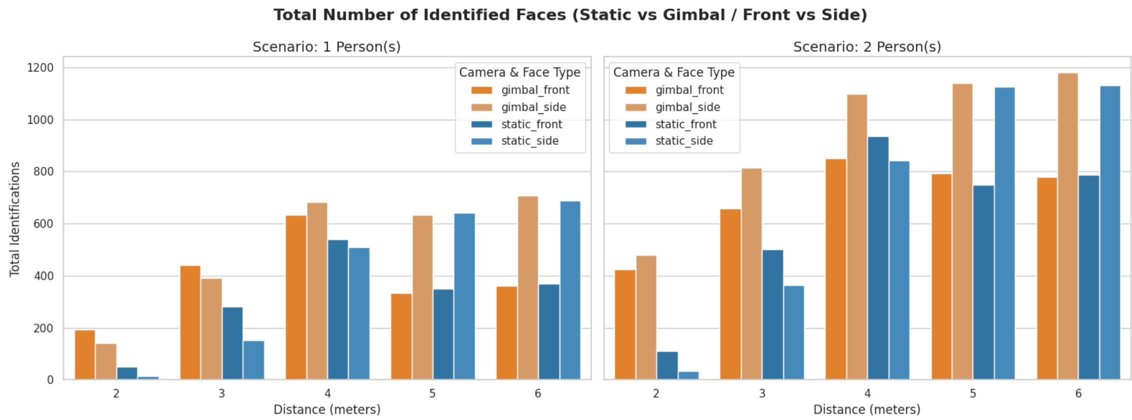

Figure 7. Comparative analysis of the number of faces identified

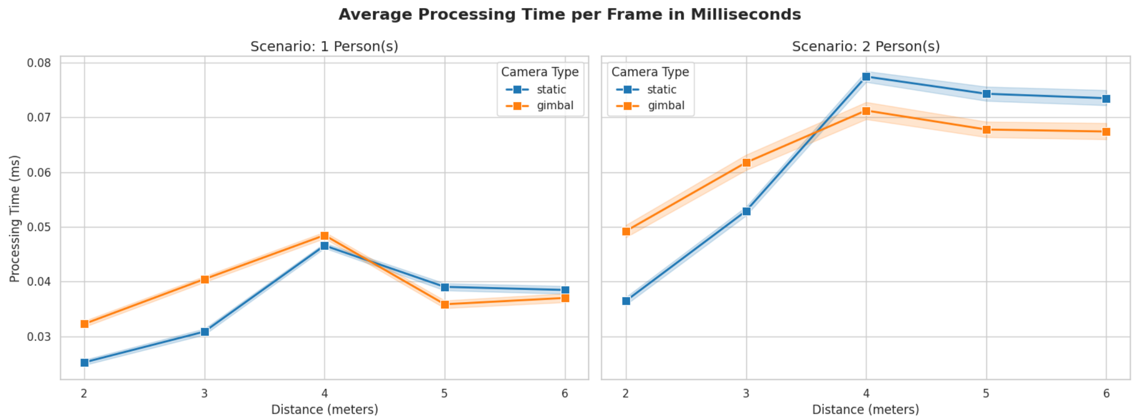

Figure 8. Comparative analysis of processing time

may be associated with reduced detection complexity. In the two-person scenario, processing time increases up to 4 meters due to the higher number of detected faces, and then decreases as subjects move further away. These trends suggest that processing time is influenced more by scene complexity than by the tracking mechanism itself.

Telemetry analysis also provides relevant insights into system efficiency and thermal stability. By offloading neural network inference (SCRFD and FaceNet) to the Hailo-8L M.2 Entry-Level AI accelerator, CANELA maintains low CPU utilization, averaging between 12% and 16%, and RAM usage between 22% and 26%. This allows the CPU to focus on control tasks, including spatial error computation and PID-based motor control. Regarding thermal measurements, the accelerator operated within a stable temperature range of 44°C to 50°C, while the CPU reached temperatures between 75°C and 85°C in both setups. More importantly, we observed no evidence of thermal throttling in CANELA, demonstrating that hardware acceleration effectively mitigates performance degradation under sustained workloads.

In summary, the comparative analysis highlights clear trade-offs and advantages. CANELA consistently outperformed the static camera at shorter distances and in multi-person scenarios, where FoV limitations are most critical. While both systems exhibited similar behavior at longer distances, the static camera showed a substantial decrease in detection performance at closer ranges, particularly for side-view movements. In contrast, CANELA maintained stable tracking by actively adjusting its viewpoint, demonstrating its effectiveness in dynamic surveillance scenarios. These results validate the effectiveness of the CANELA design, demonstrating that integrating active camera control with dedicated AI acceleration significantly improves tracking continuity, system robustness, and real-time performance in edge-based intelligent surveillance applications.

7. Conclusion

This paper presented CANELA, an AI-driven pan-tilt camera system designed as a versatile edge-computing solution for ISS. As a model-agnostic platform, CANELA enables real-time, active object tracking by integrating perception and actuation directly at the edge. The system addresses key limitations of traditional ISS, including limited FoV and computational constraints of embedded devices. The CANELA's design combines a Raspberry Pi 5 with a dedicated Hailo-8L M.2 Entry-Level AI accelerator, enabling efficient execution of deep learning models at the network edge. By offloading computationally intensive inference tasks to the coprocessor, the system frees CPU resources to manage parallel feedback control routines and real-time mechanical tracking. This design ensures stable performance while avoiding the latency and thermal degradation commonly associated with CPU-bound edge AI systems.

Experimental results in a controlled environment validated both the effectiveness of the active tracking mechanism and the efficiency of CANELA. Telemetry analysis showed that the AI accelerator maintained low, stable temperatures, confirming the system's ability to sustain continuous inference workloads without thermal throttling. From a tracking perspective, CANELA significantly improved detection performance at shorter distances (2 to 3 meters), particularly in multi-person scenarios where FoV limitations are most critical. At longer distances, the system performed comparably to a static camera, indicating that the benefits of active tracking are most noticeable under spatial constraints. Our results showed that CANELA achieved a higher number of successfully detected frames, highlighting its ability to increase the temporal window for perception tasks.

Future work will focus on both hardware and software enhancements to further improve robustness and applicability. On the hardware side, upgrading the camera module to include autofocus and optical zoom will improve image quality and detection reliability. On the software side, incorporating stricter confidence thresholds for face detection and extending support for additional object detection models will enhance system flexibility. These improvements will enable CANELA deployment in more complex real-world ISS dynamic scenarios, such as police patrolling and traffic monitoring.

Acknowledgments

This work is supported by the SPICI project, funded by FINEP (grant 2827/22).

References

Adrian, A. I., Ismet, P., and Petru, P. (2018). An overview of intelligent surveillance systems development. In *2018 International Symposium on Electronics and Telecommunications (ISETC)*, USA. IEEE.

Bashir, F. and Arief, S. (2025). Utilization of deep learning in PTZ (pan-tilt-zoom) camera control systems for geospatial-based intelligence surveillance. *Remote Sensing Technology in Defense and Environment*, 2(2):117–131.

Benoit-Cattin, T., Velasco-Montero, D., and Fernández-Berni, J. (2020). Impact of thermal throttling on long-term visual inference in a CPU-based edge device. *Electronics*, 9(12).

Garcia, D. R., Serna, A. A., Crespo, D. S., Yu, X., and Saniie, J. (2024). AI smart security camera for person detection, face recognition, tracking and logging. In *2024 IEEE International Conference on Electro Information Technology (eIT)*, USA. IEEE.

Guo, J., Deng, J., Lattas, A., and Zafeiriou, S. (2022). Sample and computation redistribution for efficient face detection. In *2022 International Conference on Learning Representations*. https://iclr.cc/virtual/2022/poster/6369.

Hsia, S.-C., Wang, S.-H., Wei, C.-M., and Chang, C.-Y. (2022). Intelligent object tracking with an automatic image zoom algorithm for a camera sensing surveillance system. *Sensors*, 22(22).

Kim, I. S., Choi, H. S., Yi, K. M., Choi, J. Y., and Kong, S. G. (2010). Intelligent visual surveillance – A survey. *International Journal of Control, Automation and Systems*, 8(5):926–939.

Niu, H., Liu, J., Yu, Z., Zheng, D., He, P., and Wang, F. (2022). Real-time object tracking system using PTZ camera. In *2022 IEEE 4th International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, pages 471–478, USA. IEEE.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, USA. IEEE.

Sharma, H. and Kanwal, N. (2024). Video surveillance in smart cities: Current status, challenges & future directions. *Multimedia Tools and Applications*, 84(16):15787–15832.

Singh, S., Saini, R., Saurav, S., and Saini, A. K. (2017). Real-time object tracking with active PTZ camera using hardware acceleration approach. *International Journal of Image, Graphics and Signal Processing*, 9(2):55.

Wang, X., Tang, Z., Guo, J., Meng, T., Wang, C., Wang, T., and Jia, W. (2025). Empowering edge intelligence: A comprehensive survey on on-device AI models. *ACM Computing Surveys*, 57(9).

Yu, C., Wang, M., Chen, S., Wang, W., Fang, W., Chen, Y., and N.Xiong, N. (2025). Efficient NPU–GPU scheduling for real-time deep learning inference on mobile devices. *Journal of Real-Time Image Processing*, 22.