

Critical Pedestrian Behavior Detection for Public Safety Monitoring: Transformer-Based Pedestrian Anomaly Identification

Mohammad Reza Mohebbi, Josef Ressel Center Vision2Move, University of Applied Sciences Kufstein Tirol, Kufstein, Austria

Mayank Sudarsi Kotaiah, University of Passau, Passau, Germany

Manu Gupta, Josef Ressel Center Vision2Move, University of Applied Sciences Kufstein Tirol, Kufstein, Johannes Kepler University Linz, Linz, Austria

Mario Döller, Josef Ressel Center Vision2Move, University of Applied Sciences Kufstein Tirol, Kufstein, Austria

Abstract: Surveillance camera networks produce far more video than operators can reliably assess in real time, particularly in crowded scenes where critical events are brief, partially occluded, and spread across multiple views. This paper presents a transformer-based pedestrian anomaly detection module for public-safety decision support, capable of detecting and categorizing abnormal behaviors in surveillance footage. We train RF-DETR Nano with a DINOv2 backbone on the IITB-Corridor dataset with nine behavior classes (Normal plus eight anomaly types). To improve temporal consistency, we combine lightweight tracking with temporal smoothing and anomaly persistence to reduce flickering alerts. The approach attains 91.1% mAP@50 and a 0.51% false-alarm rate for Normal-versus-Anomaly alerting, indicating high reliability for operator workflows. We discuss failure modes (small objects, motion blur) and runtime trade-offs, and outline integration into broader multi-camera monitoring pipelines.

Keywords: Pedestrian anomaly detection, Video surveillance analytics, Transformer-based object detection, RF-DETR, DINOv2, Real-time public safety monitoring

1. INTRODUCTION

Crowded public spaces such as transport hubs, campuses, and event venues are increasingly monitored by networks of surveillance cameras. Yet the volume of video streams far exceeds what human operators can continuously observe, and critical incidents may unfold within seconds under occlusions, motion blur, and rapidly changing crowd dynamics [1]. As a result, early warning cues—ranging from suspicious object placement to violent interactions—can be missed or detected too late to enable timely intervention [1].

Automated video analytics can reduce this operational gap by flagging unusual situations and helping operators focus attention where it matters most. For public-safety monitoring, however, an effective solution must meet practical constraints: it should be robust to crowded-scene noise, produce stable alerts rather than flickering detections, offer interpretable categories that match operator workflows, and run fast enough for near-real-time use. In practice, minimizing false alarms is often as important as achieving high detection accuracy, because excessive alerts quickly erode trust and usability [1].

A large body of work on video anomaly detection focuses on learning “normality” from data and identifying deviations, often using unsupervised or weakly supervised objectives [1]. Common strategies include reconstruction- and prediction-based models that learn spatiotemporal regularities and mark frames with high reconstruction/prediction error as abnormal [2, 3]. While these approaches can generalize beyond a predefined anomaly list, they frequently provide limited interpretability (what kind of anomaly occurred), can be sensitive to appearance changes (lighting, viewpoint), and may struggle in crowded scenes where “normal” contains complex interactions and frequent occlusions [1].

In parallel, recent advances in deep visual recognition—especially transformer-based detectors and strong self-supervised visual backbones—have improved person-centric perception in surveillance-like imagery [4, 5]. Detection- and tracking-driven pipelines are attractive because they naturally produce localized evidence (bounding boxes) and enable event-level reasoning over time [4]. Nevertheless, many such systems still face practical challenges: temporal instability in frame-wise predictions, brittleness for small or distant objects, and compute-heavy additions (e.g., segmentation) that can compromise real-time operation [1]. Moreover, multi-camera monitoring introduces further complexity (cross-view identity association), which motivates modular designs where reliable single-view perception is a necessary first step.

These limitations motivate our approach: we develop a transformer-based pedestrian anomaly detection module that is explicitly optimized for operator-facing reliability and temporal consistency. We train an RF-DETR Nano detector with a DINOv2 backbone to detect and categorize nine corridor behaviors (Normal plus eight anomaly types). On top of per-frame detections, we integrate lightweight tracking, temporal smoothing, and anomaly persistence (“anomaly locking”) to reduce flicker and produce stable alerts. In addition to accuracy, we report runtime profiling to quantify deployment trade-offs and identify bottlenecks, positioning this module as a practical building block for broader multi-camera safety analytics.

Our contributions are summarized as follows:

- A **transformer-based pedestrian anomaly detection** pipeline using **RF-DETR Nano + DINOv2** for multi-class behavior categorization in surveillance video.
- A **temporal stabilization strategy** (tracking, smoothing, and anomaly persistence) that reduces flickering alerts and supports operator workflows.
- An **operator-oriented evaluation setup** including Normal-vs-Anomaly alerting and multi-class anomaly categorization.
- A **deployment-focused runtime analysis** highlighting performance trade-offs and computational bottlenecks relevant to real-time monitoring.

2. METHODOLOGY

2.1. System Overview and Processing Pipeline

We develop a pedestrian anomaly detection module for surveillance video that localizes actors, assigns interpretable behavior labels, and produces operator-stable alerts under occlusion and motion blur (Fig. 1). Video is sampled at 8 FPS and preprocessed (640×640

resize, ImageNet normalization). An RF-DETR Nano detector with a DINOv2 backbone predicts bounding boxes and 9-class behaviors. Predictions are filtered (confidence + NMS), linked across frames with a lightweight tracker, and stabilized using temporal smoothing and anomaly persistence (“anomaly locking”) to reduce alert flicker. The system outputs both Normal-vs-Anomaly alerts and fine-grained anomaly categories. Optionally, SAM v2 is applied to detected regions for latency–accuracy profiling.

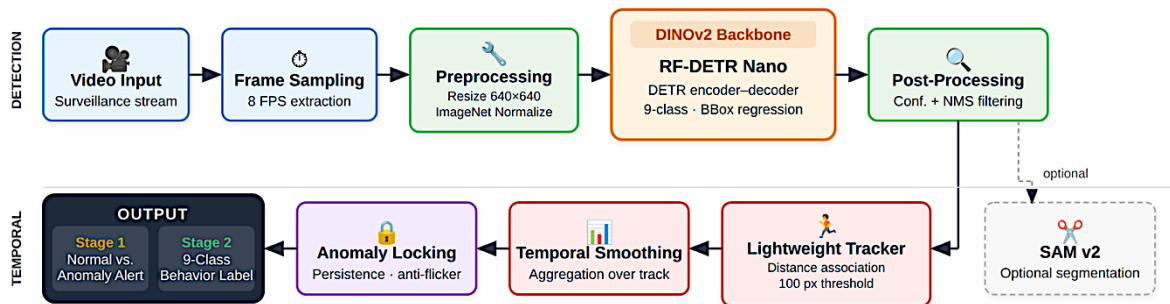


Figure 1: Overview of the proposed pedestrian anomaly detection pipeline.

2.2. Dataset and Annotation

Experiments are conducted on the IITB-Corridor dataset [6], which contains surveillance footage recorded in corridor environments. We model nine behavior classes:

- **Normal:** Standard Walking
- **Anomalies:** Bag Exchange, Cycling, Suspicious Object, Running, Fighting, Hiding, Playing Ball, Protest

All instances are annotated with bounding boxes using an open-source bounding-box annotation tool, providing supervised localization and class labels for detector training. The annotated data is partitioned into training, validation, and test sets; model selection is performed on the validation set via the best validation loss checkpoint.

2.3. Model Architecture: RF-DETR Nano with DINOv2

Our detector is based on RF-DETR Nano, a compact detection transformer that follows the DETR paradigm [4].

Backbone (DINOv2). We employ DINOv2 as the feature extractor to obtain robust visual representations under lighting variation and viewpoint changes typical in surveillance data [5].

Transformer encoder–decoder. The encoder is a ViT-style self-attention module that aggregates global context. The decoder uses a fixed set of learned object queries that attend to the encoded features and produce a set of candidate detections [4].

Prediction heads. For each query, the model outputs:

- class logits over the 9 behavior classes, and

- bounding-box regression parameters.

Training objective. The loss combines standard detection terms:

- cross-entropy for classification,
- L1 loss for box regression, and
- GloU loss for improved localization, especially under partial occlusions.

This architecture provides localized, interpretable outputs (boxes + labels) and supports downstream temporal stabilization.

2.4. Preprocessing and Training

We apply horizontal flipping, scaling, and color jitter for augmentation, to improve robustness to appearance changes. Training uses AdamW with a batch size of 8; the final model is selected by lowest validation loss.

2.5. Inference Post-processing and Temporal Stabilization

2.5.1. Detection filtering

For each frame, we retain detections above a confidence threshold and remove duplicates using NMS:

- confidence threshold: 0.25
- NMS IoU threshold: 0.5

2.5.2. Lightweight tracking (single-camera association)

To obtain short-term temporal continuity at negligible cost, detections are linked across consecutive frames using distance-based matching. Each detection is associated with the nearest active track if the distance is below a fixed threshold:

- association threshold: 100 px (image space)

Tracks that cannot be matched are ended after a short tolerance period (implementation-dependent), and new detections initiate new tracks. This tracker is intentionally simple: its role is not identity re-identification but enabling track-level temporal aggregation.

2.5.3. Temporal smoothing

Frame-wise class predictions can fluctuate due to partial occlusion and motion blur [1]. We therefore aggregate class evidence over each track using a short temporal history (windowed majority), producing a more stable class estimate and suppressing short-lived label spikes.

2.5.4. Anomaly persistence (“anomaly locking”)

Even with smoothing, anomalies can momentarily disappear when the actor is partially occluded or blurred. To avoid alert flicker, we apply anomaly locking: once a track is confidently classified as anomalous, the anomaly state is retained for a short persistence

horizon and released only when sufficient evidence supports a return to Normal. This converts unstable frame-level predictions into more consistent, operator-friendly alerts.

2.5.5. Optional SAM v2 segmentation (profiling)

For deployments that require pixel-accurate highlighting, we optionally run SAM v2 on detected regions to produce segmentation masks [2]. In this study, segmentation is evaluated primarily for its runtime impact and is not required for the reported detection/classification accuracy.

2.6. Outputs and Evaluation Protocol (Operator-Oriented Stages)

We evaluate the system in three complementary stages aligned with operational use:

- **Stage 1 — Alerting (Normal vs. Anomaly).**
All anomaly classes are merged into a single “Anomaly” label. This reflects the primary operator question: does this situation require attention?
- **Stage 2 — Categorization (9-class).**
The model outputs one of the nine classes to provide interpretable context (e.g., Fighting vs. Suspicious Object), which can support prioritization and response selection.
- **Stage 3 — Runtime profiling.**
We report per-component latency and end-to-end throughput (FPS) for the detector, optional segmentation, tracking, and the full pipeline.

Metrics include standard detection quality (mAP) for localization/classification and classification metrics (precision/recall/F1) for Stage 1 and Stage 2. Unless stated otherwise, evaluations are performed on the test split after applying the inference filtering and temporal stabilization described above.

3. EXPERIMENTAL RESULTS

3.1. Setup and Evaluation Protocol

The proposed pipeline is evaluated on the IITB-Corridor dataset, which consists of nine categories — one normal class and eight anomaly types. Performance is reported at three levels:

(i) overall detection quality (mAP and Recall@100), (ii) operator alerting by collapsing all anomaly classes into Normal vs. Anomaly, and (iii) fine-grained 9-class categorization. Runtime is measured per component and for the end-to-end pipeline.

3.2. Quantitative Results

3.2.1. Overall detection

RF-DETR Nano (DINOv2) achieves strong localization and classification performance on corridor surveillance footage (Table 1), indicating reliable detection under typical occlusion and viewpoint constraints.

3.2.2. Alerting (Normal vs. Anomaly)

For operator-oriented alerting, the method attains 97.4% accuracy and a 0.51% false-alarm rate (Table 1), showing that the temporal stabilization strategy yields highly reliable alarms without excessive spurious triggers.

3.2.3. Fine-grained categorization

The model performs consistently across most anomaly categories (Table 2). Classes with clear visual signatures such as Fighting and Hiding show the highest F1, while motion-dominant behaviors (e.g., Cycling, Running) remain more challenging due to blur and viewpoint variation.

Table 1: Summary results (test set)

Task	Metric	Value
Detection (9-class)	mAP@50	91.1%
	mAP@50:95	74.0%
	Recall@100	82.5%
	AP (small)	49.7%
Normal vs. Anomaly	Accuracy	97.4%
	F1-score	92.82%
	False-alarm rate	0.51%

Table 2: Per-class performance (F1-score)

Class	F1	Class	F1
Normal	98.39%	Fighting	98.06%
Bag Exchange	92.86%	Hiding	98.99%
Cycling	77.78%	Playing Ball	97.25%
Suspicious Object	95.38%	Protest	94.26%
Running	86.99%		

3.3. Runtime Performance

Table 3 reports average latency and throughput. Detection alone runs at 14 FPS, while adding SAM v2 reduces end-to-end speed to 2.93 FPS, making segmentation the dominant bottleneck for real-time deployment in its current configuration.

Table 3: Runtime profiling

Component	Avg latency (ms)	FPS
RF-DETR detection	71.42	14.00
SAM v2 segmentation	267.78	3.73
Tracking	1.71	585.72
Total pipeline	340.92	2.93

3.4. Qualitative Results

These qualitative examples (Fig. 2) illustrate the output of our pipeline on IITB-Corridor: pedestrians are detected and assigned behavior labels, with Normal dominating and anomalies highlighted when present. The top example shows correct recognition of Fighting (left) and Protest in a group scene (right). The bottom example shows detection of Hiding (left) and Running (right). The colored overlays correspond to the optional SAM v2 masks used for visualization/profiling, while labels are produced by RF-DETR.



Figure 2: Qualitative results on IITB-Corridor. The pipeline detects pedestrians and assigns behavior labels, correctly identifying anomalies such as Fighting, Protest, Hiding, and Running. Colored overlays show optional SAM v2 masks; labels are produced by RF-DETR.

3.5. Failure Modes and Discussion

Two consistent limitations emerge. First, small/distant targets reduce recall for brief anomalies, reflected in $AP(\text{small}) = 49.7\%$. Second, motion-heavy behaviors (Cycling, Running) suffer from motion blur and partial occlusions, producing missed detections or temporary class flips. Temporal smoothing and anomaly persistence substantially reduce flickering, and the very low false-alarm rate indicates strong suitability for operator alerting despite class imbalance between Normal and Anomaly instances.

4. CONCLUSION

We presented a transformer-based pedestrian anomaly detection module for surveillance video that supports operator decision-making by detecting and categorizing abnormal behaviors. Using RF-DETR Nano with a DINOv2 backbone on the IITB-Corridor dataset, the system achieved strong detection performance (91.1% mAP@50) and reliable alerting, reaching 97.4% accuracy for Normal-versus-Anomaly with a 0.51% false-alarm rate. Lightweight tracking with temporal smoothing and anomaly persistence reduced flickering predictions and produced more stable alerts.

Remaining challenges mainly concern small/distant targets and motion-dominant behaviors (e.g., Cycling and Running) affected by blur and occlusion. Runtime profiling showed 14 FPS for detection, while adding SAM v2 segmentation reduced throughput to 2.93 FPS, making segmentation the primary bottleneck for real-time deployment. Future work will target robustness to small objects and motion, improve runtime efficiency, and extend the module toward multi-camera integration with privacy-by-design considerations.

Bibliography

- [1] Zhu, Liyun; Wang, Lei; Raj, Arjun; Gedeon, Tom; Chen, Chen (2024). Advancing Video Anomaly Detection: A Concise Review and a New Dataset. arXiv preprint, arXiv:2402.04857.
- [2] Huang, Yuzhi; Li, Chenxin; et al. (2025). Track Any Anomalous Object: A Granular Video Anomaly Detection Pipeline. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 8689–8699.
- [3] Liu, Wen; Luo, Weixin; Lian, Dongze; Gao, Shenghua (2018). Future Frame Prediction for Anomaly Detection – A New Baseline. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6536–6545.
- [4] Carion, Nicolas; Massa, Francisco; Synnaeve, Gabriel; Usunier, Nicolas; Kirillov, Alexander; Zagoruyko, Sergey (2020). End-to-End Object Detection with Transformers. Computer Vision – ECCV 2020 (Lecture Notes in Computer Science), 12346, 213–229.
- [5] Oquab, Maxime; Darcet, Timothée; Moutakanni, Théo; et al. (2023). DINOv2: Learning Robust Visual Features without Supervision. arXiv preprint, arXiv:2304.07193.
- [6] https://opendatalab.com/OpenDataLab/IITB_Corridor