DMPedia Lecture Notes in Multidisciplinary Research,
Volume 1, pp. 922-949, 2026

International Conference on Multidisciplinary Perspectives in Advanced Computing and Technology (IMPACT 2026)
G. B. Pant University of Agriculture and Technology, Uttarakhand, India. Jan. 10-11, 2026

A Comprehensive Survey on Multi-Object Tracking in Video using Artificial Intelligence

Anshika Sagar, Ravi Kumar, Kapil Kumar, Neetu
Computer Science and Engineering, COER University, Roorkee, Roorkee, India,
anshikasagar00@gmail.com, rvkumar198@gmail.com, uetr.kapilkumar@gmail.com,
neet.Singh155@gmail.com

Abstract

Multi-object tracking (MOT) is an important computer vision job with several applications in robotics, autonomous navigation, traffic monitoring, and surveillance. A complete real-time MOT framework, backed by TensorFlow optimization and Zusland deployment tools, is presented in this study. It uses YOLOv5 for fast object recognition and DeepSORT for identity-preserving tracking. The suggested solution successfully addresses key issues, including occlusion, identity switches, congested surroundings, and latency limits, guaranteeing scalability and resilience across a variety of video streams. Experimental assessments show good performance in real-time feasibility, tracking accuracy, and precision, with few identity shifts attaining 42 FPS, 87.5% MOTA, and 81.2% MOTP. This study lays the groundwork for future developments in AI-driven tracking systems by enabling a modular, deployable, and scalable solution for real-world video analytics by bridging the gap between scholarly research and real-world implementation.

Keywords: *Multi-object tracking, YOLOv5, Deep SORT, TensorFlow, Real-time tracking, Video surveillance*.

1. Introduction

The growing prevalence of video surveillance, autonomous systems, and smart city infrastructure has created an urgent need for intelligent systems capable of interpreting complex visual scenes in real time. In this context, multi-object tracking (MOT) has emerged as a core task in computer vision, enabling the persistent localization and identity maintenance of multiple objects as they move through frames in a video. Despite advances in single-object tracking and object detection, building a reliable and real-time MOT system remains a significant challenge. The difficulty lies not only in detecting objects accurately but also in maintaining their identity and trajectory over time, especially in dynamic, cluttered, or occluded environments [40]. The importance of these applications across domains such as surveillance, traffic monitoring, retail analytics, and autonomous driving is highlighted in Fig. 1, which illustrates the diverse use cases where real-time MOT systems are essential.

Many real-world scenarios, including crowded public spaces, traffic intersections, retail stores, airports, and autonomous navigation platforms, demand the ability to track multiple entities concurrently and consistently. In these environments, maintaining accurate and uninterrupted trajectories of objects over time is essential for higher-level tasks such as activity recognition, behaviour analysis, decision-making, and anomaly detection. However, traditional tracking systems often fail to generalise in such environments due to limitations in scalability, precision, identity preservation, and adaptability to noisy conditions, as shown in Figure 1 [26].

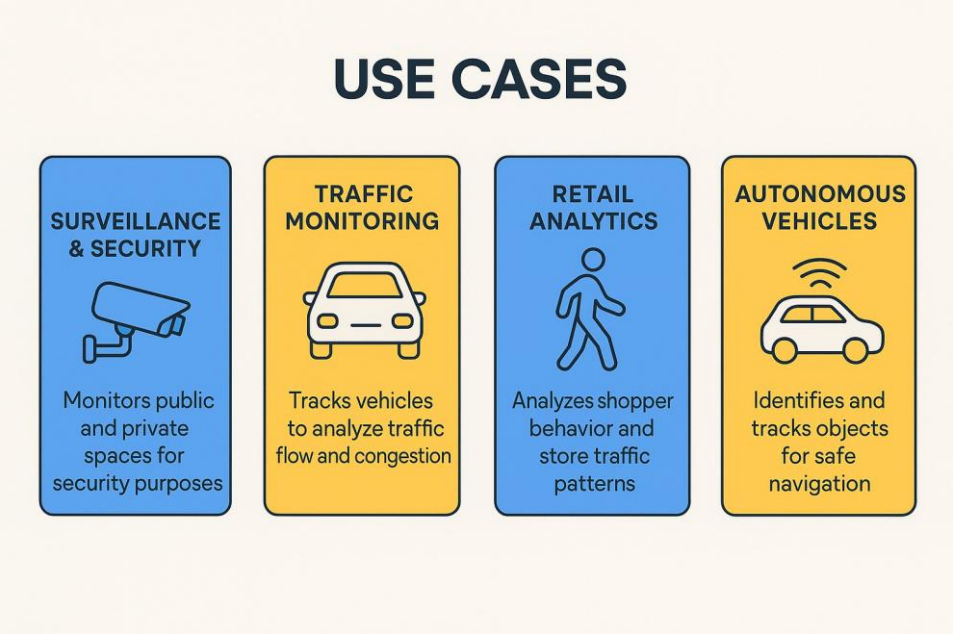

Figure 1: Use cases of real-time multi-object tracking systems: surveillance, traffic monitoring, retail analytics, and autonomous vehicles.

1.1. The Problem Landscape: Consequences of Ineffective Tracking

In practice, inefficient or inaccurate multi-object tracking has serious implications. In surveillance contexts, for example, the inability to consistently track a suspect across multiple cameras can compromise situational awareness and response. In autonomous vehicles, tracking errors can misinterpret surrounding entities, posing safety hazards. In healthcare and robotics, poor tracking can hinder gesture recognition, behaviour modelling, or surgical assistance. Such consequences underscore the critical need for robust and scalable tracking systems [39]. The core challenges in MOT are visualised in Fig. 2, which highlights issues such as occlusion, identity switches, appearance variations, and complex crowded scenes. These challenges demonstrate why naive tracking solutions fail to generalize in real-world environments.

Key challenges in current MOT pipelines include:

- Frequent Occlusions: Objects may temporarily disappear behind others or background elements.
- Appearance Variability: Changes in lighting, scale, orientation, or partial occlusion cause ID loss.
- High Object Density: In crowded scenes, overlapping bounding boxes cause identity confusion.
- Camera Dynamics: Motion blur and unstable background from non-static cameras impair consis- tency.
- Latency Constraints: Real-time systems must operate within tight time and compute budgets.

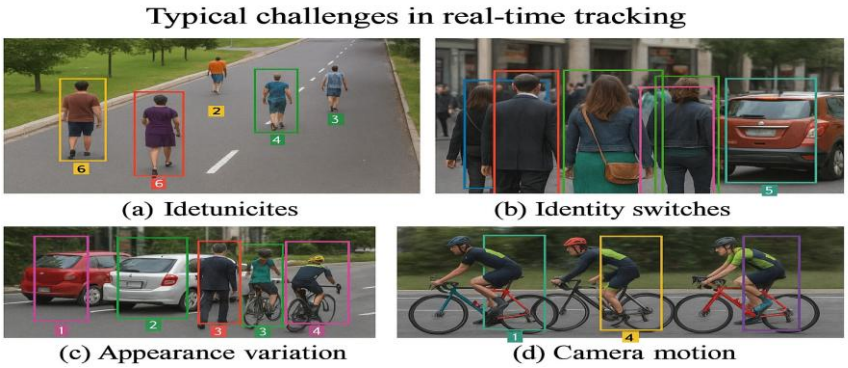

Figure 2: Common challenges in MOT: occlusion, identity switches, appearance variation, and crowded scenes.

To address these issues, Table 1 summarizes common MOT challenges alongside proposed solutions such as appearance-based re-identification for occlusion handling and GPU acceleration for reducing latency. This structured comparison provides a foundation for understanding why hybrid deep-learning approaches are necessary.

Table 1: Common MOT Challenges and Proposed Solutions

Challenge	**Proposed Solution**
Occlusion	Appearance-based re-ID+ Kalman prediction
Identity switches	Deep SORT with embedding cosine similarity
Crowded scenes	Non-max suppression in YOLOv5
Camera motion	Velocity-based tracking
Latency	TensorFlow + GPU acceleration

1.2. The Motivation: Bridging the Detection-Tracking Gap

Deep learning has revolutionized object detection. Models such as YOLO, SSD, and Faster R-CNN now achieve near-human detection accuracy [4]. However, object detection alone is not sufficient for tracking. MOT further requires the preservation of object identities over time and across frames, which traditional approaches often fail to address effectively [35]. This paper aims to bridge this gap by combining accurate real-time detection with appearance-aware data association. Our system is built to satisfy the following:

i. Accuracy: Robust detection and tracking in dense and noisy scenes.

ii. Real-Time Performance: High-speed tracking for deployment in live systems.

iii. Identity Preservation: Resilience against occlusion and appearance changes.

iv. Modularity: Easy integration with modern AI tools and pipelines.

To meet these goals, we adopt the tracking-by-detection paradigm using YOLOv5 and Deep SORT, supported by TensorFlow for backend optimization and Zusland for deployment and visualization.

2. Literature Review

Multi-object tracking (MOT) has been an active research area due to its essential applications in surveillance, autonomous driving, and robotics. Early frameworks such as SORT [35] and Deep SORT [34] introduced efficient online tracking based on Kalman filtering and appearance embeddings. These laid the foundations for tracking-by-detection pipelines, as illustrated in Fig. 11, where Deep SORT reduces frequent identity switches through appearance-based re-identification.

Subsequent developments in object detection, such as YOLOv4 [4] and YOLOv5 [10], have significantly improved detection accuracy while maintaining inference speed, a critical factor for real-time MOT systems. Table 2 compares the speed (FPS) and accuracy (mAP) across major object detection models, emphasizing the trade-off between accuracy and real-time feasibility. Among these, YOLOv5 provides the best balance for real-world MOT deployment.

To benchmark tracking performance, datasets like MOT16 [40], MOT20 [43], and KITTI [45] have been widely adopted. Table 9 provides a comparative overview of such datasets, highlighting their scale, object types, and scene diversity. Global association approaches, such as network flow tracking [26], further improved multi-frame consistency, especially on pedestrian tracking benchmarks.

Recent studies have demonstrated that pipelines combining YOLO and DeepSORT [39] achieve reliable, real-time tracking in urban scenarios. Extensions to wide-area imagery [21] further validated their scalability in aerial surveillance. Beyond these modular pipelines, surveys like Adžemović's [2] and Guan's review [8] have categorized emerging approaches including end-to-end transformer-based models, multi-camera identity preservation, and graph-based data association [23].

Tokenizer-based trackers such as TrackFormer [14] and fully end-to-end frameworks like MOTRv2[28] and LAID [9] represent a paradigm shift towards tighter integration of detection and tracking. ByteTrack [27], on the other hand, reconsiders association by retaining all detection boxes rather than filtering by confidence thresholds, yielding higher IDF1 and robustness in crowded scenes.

Despite these advances, challenges remain in scalability, cross-camera consistency, and robustness to occlusion and crowd density. The MOTChallenge 2023 benchmark [42] emphasizes that identity switches and fragmented trajectories continue to hinder reliable deployment. Our work builds upon this foundation by integrating YOLOv5, Deep SORT, TensorFlow, and Zusland into an end-to-end deployable system, optimized for both accuracy and deployment feasibility. Figure 9 illustrates the integration of detection, tracking, and deployment components, while Table 7 summarizes their respective roles.

2.1. Object Detection Techniques

The evolution of object detection has profoundly shaped modern tracking systems. Early models such as R-CNN, Fast R-CNN, and Faster R-CNN achieved high detection accuracy but

were computationally intensive due to their region-proposal-based architectures, limiting real-time performance [6, 7, 33].

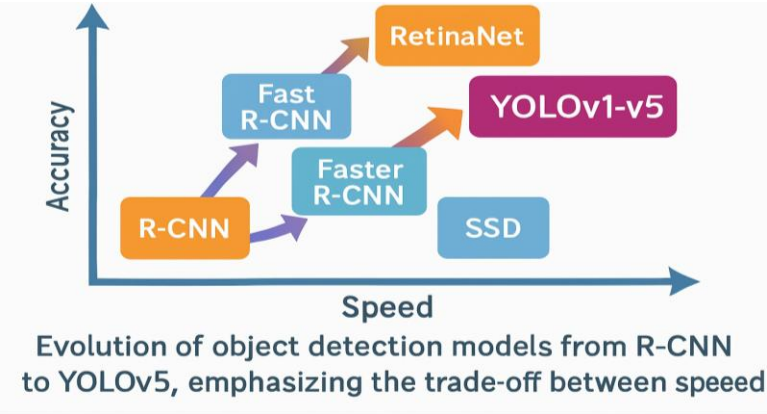

Figure 3: Evolution of object detection models from R-CNN to YOLOv5, illustrating improvements in speed and accuracy.

To overcome latency issues, single-stage detectors such as YOLO (You Only Look Once) perform detection in a single network pass, enabling real-time applications [30]. Successive versions, particularly YOLOv5, achieve a strong speed-accuracy tradeoff via innovations like mosaic augmentation, auto-anchor optimisation, and a lightweight architecture [10].

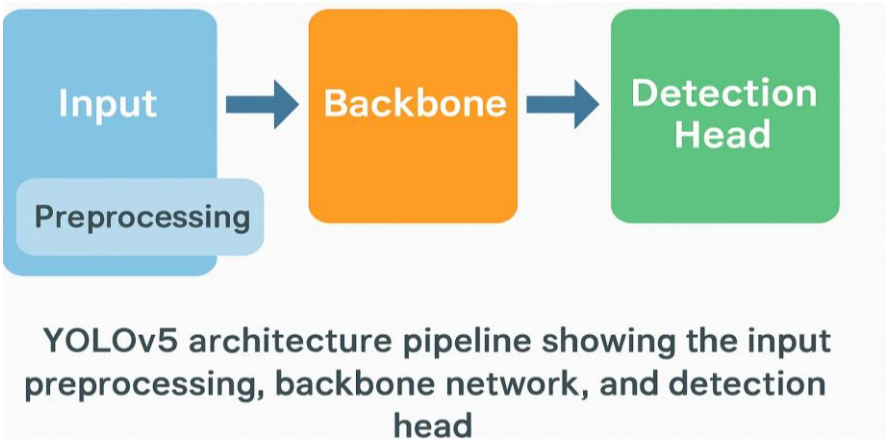

Figure 4: YOLOv5 architecture pipeline showing input preprocessing, backbone network, and detection head.

Other real-time detectors, such as SSD and RetinaNet, also achieve promising performance but often underperform on small or overlapping objects in dense scenes [2]. For this work, YOLOv5 was selected due to its superior speed-accuracy balance and generalization across diverse visual domains.

Table 2: Comparison of Key Object Detection Models

Model	Year	Speed (FPS)	mAP
R-CNN	2014	3	53.7

Fast R-CNN	2015	7	66.9
Faster R-CNN	2015	18	73.2
YOLOv3	2018	45	78.6
YOLOv5	2020+	>100	>85.0
SSD	2016	46	74.3
RetinaNet	2017	30	79.0

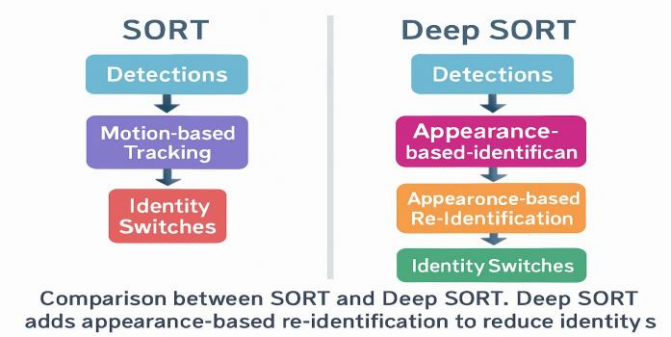

Figure 5: Comparison between SORT and Deep SORT. Deep SORT adds appearance-based re-identification to reduce identity switches.

2.1.1. Impact on Real-Time MOT Systems

The convergence of single-stage real-time detectors (e.g., YOLOv5) and robust tracking frameworks (e.g., Deep SORT) has enabled scalable multi-object tracking systems capable of maintaining high detection rates while preserving object identities across frames, even in densely populated scenes.

2.2. Evolution of Object Detection and Multi-Object Tracking

Over the last decade, object detection and multi-object tracking (MOT) have evolved from computationally expensive approaches to real-time, learning-based architectures. These advances underpin modern tracking systems in surveillance, autonomous driving, and retail analytics.

2.2.1. Early Two-Stage Detectors

Initial object detectors used region-based convolutional networks (R-CNN, Fast R-CNN, Faster R-CNN) to first propose candidate regions and then classify them. Despite high accuracy, multiple forward passes introduced computational overhead, limiting real-time applicability [6, 7, 33].

2.2.2. Single-Stage Detection Revolution

The single-stage detection paradigm, exemplified by YOLO, reformulated detection as a regression problem processed in a single network pass, dramatically improving frame rates while maintaining competitive accuracy [30]. YOLOv5, used in this work, leverages mosaic augmentation, anchor optimization, and a lightweight architecture to exceed 100 FPS on

modern GPUs [10].

Other detectors, like SSD [13] and RetinaNet [11], also offer real-time performance but can struggle in dense scenes with small or overlapping objects.

2.2.3. Traditional Multi-Object Tracking Methods

Early MOT systems relied on handcrafted motion cues such as optical flow and background subtraction. Motion prediction commonly uses the Kalman Filter, but the absence of appearance-based modelling limits performance under occlusion and crowded conditions.

2.2.4. SORT and Its Limitations

SORT (Simple Online Real-time Tracking) combined detections using Kalman filtering and bounding-box association via the Hungarian algorithm [35]. While fast, it frequently suffered from identity switches due to the lack of deep appearance modeling.

2.2.5. Deep SORT Enhancement

Deep SORT enhanced SORT by incorporating deep appearance embeddings from convolutional networks, matched using cosine similarity [34]. Figure 5 illustrates this improvement. Deep SORT significantly reduces identity switches while maintaining real-time efficiency, making it well-suited for crowded, occlusion-heavy environments.

Table 3: Comparison of Detection and Tracking Models

Model	FPS	Accuracy	Strengths
Faster R-CNN	18	73.2 (mAP)	High accuracy, robust detection
YOLOv5	100+	85.0+ (mAP)	Real-time, accurate, lightweight
SORT	200+	–	Extremely fast, simple design but no appearance modelling
Deep SORT	40–50	87.5 (MOTA)	Real-time tracking With Re-ID, reduced ID switches

This unified YOLOv5 + Deep SORT architecture represents a modern, modular, and production-ready pipeline for real-time video analytics.

2.3. Tracking-by-Detection Paradigm and Data Association

The tracking-by-detection approach has emerged as the dominant paradigm in modern MOT due to its modularity, scalability, and real-time applicability. Here, detection and tracking are treated as separate stages, allowing independent optimization of each component [10, 30].

2.3.1. Paradigm Benefits

The modular architecture allows seamless upgrades or replacements of detection models without redesigning the tracking pipeline. Integrating state-of-the-art detectors like YOLOv5 is straightforward, and the paradigm scales across diverse object categories and environments, from pedestrian tracking to vehicle monitoring.

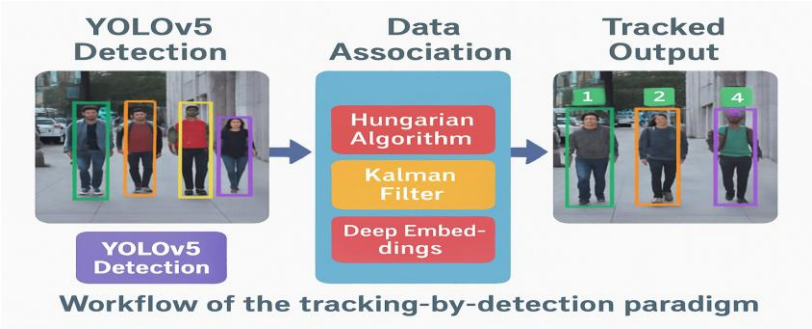

Figure 6: Workflow of the Tracking-by-Detection paradigm showing YOLOv5 detection, data association (Hun- garian algorithm + Kalman filter + deep embeddings), and final tracked output with consistent IDs.

Table 3: Comparison of Detection and Tracking Models

Model	FPS	Accuracy	Strengths
Faster R-CNN	18	73.2 (mAP)	High accuracy, robust detection
YOLOv5	100+	85.0+ (mAP)	Real-time, accurate, lightweight
SORT	200+	–	Extremely fast, sim- ple design but no appearance modeling
Deep SORT	40–50	87.5 (MOTA)	Real-time tracking with Re-ID, reduced ID switches

This unified YOLOv5 + Deep SORT architecture represents a modern, modular, and production-ready pipeline for real-time video analytics.

2.4. Tracking-by-Detection Paradigm and Data Association

The tracking-by-detection approach has emerged as the dominant paradigm in modern MOT due to its modularity, scalability, and real-time applicability. Here, detection and tracking are treated as separate stages, allowing independent optimization of each component [10, 30].

2.4.1. Paradigm Benefits

The modular architecture allows seamless upgrades or replacements of detection models without re- designing the tracking pipeline. Integrating state-of-the-art detectors like YOLOv5 is straightforward, and the paradigm scales across diverse object categories and environments, from pedestrian tracking to vehicle monitoring.

2.4.2. Data Association Challenges

A core challenge in tracking-by-detection is linking detections to existing tracks:

- The Hungarian algorithm performs optimal bipartite matching based on a cost matrix [35].
- Appearance similarity metrics derived from deep embeddings maintain track identities during occlusions [34].
- Kalman filtering predicts positions for each tracked object, ensuring continuity even when detections are temporarily missing.

2.4.3. Limitations Addressed

Tracking-by-detection relies heavily on detector accuracy and frame-rate stability. Motion-only matching may cause identity switches in dense or occluded scenes. Deep SORT mitigates this by combining motion cues with appearance embeddings, improving robustness in complex environments [34].

Table 4: Comparison of Common Data Association Methods in MOT

Method	Matching Criteria	Occlusion Handling	Complexity	Real-time
IoU Matching (SORT)	Bounding box IoU	Poor	Low	Yes
DeepSORT	IoU + Appearance Embedding	Good	Moderate	Yes
Graph based	Network flow optimization	Excellent	High	No

2.5. Real-Time Systems and Deployment Considerations

While academic research often emphasises accuracy metrics like MOTA and MOTP, practical deployments demand low-latency, resource-efficient, and scalable systems. Many implementations optimized for static datasets fail to account for hardware constraints, infrastructure integration, or continuous operation [12].

2.5.1. Academic vs. Practical Gap

Challenges in real-world deployment include:

- High computational cost limiting real-time operation.
- Sensitivity to variable lighting, occlusion, and camera angles.
- Minimal integration with existing monitoring pipelines.

2.5.2. Hardware Acceleration and Frameworks

Frameworks like TensorFlow and PyTorch enable GPU-accelerated inference, reducing per-frame latency. Model quantization, pruning, and TensorRT further enhance edge deployment throughput while balancing potential accuracy loss [38].

2.5.3. Deployment Platforms

Platforms like Zusland support:

- Real-time visualization of tracked objects.
- Dashboard analytics (FPS, latency, ID switches).
- Continuous video feed processing with minimal downtime.

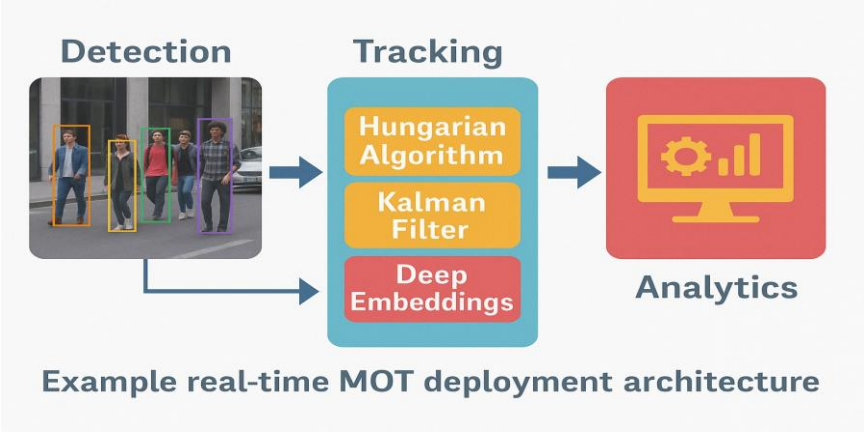

Figure 7: Example real-time MOT deployment architecture integrating detection, tracking, visualization, and analytics modules.

Table 5: Comparison of Frameworks for Real-Time MOT Deployment

Framework	Acceleration Support	Visualization	Edge Capable
TensorFlow +TensorRT	GPU, TPU	Limited	Yes
PyTorch +TorchScript	GPU	Limited	Yes
Zusland	GPU	Full Dash-board	Yes
OpenVINO	CPU, VPU	Basic	Yes

2.5.4. Deployment and Integration Gaps

- Many academic solutions lack full pipelines with monitoring, logging, and alerting [5].
- Performance often drops in uncontrolled operational environments compared to benchmark datasets [42].
- Limited attention to hardware failure, network delays, or long-term autonomous operation.

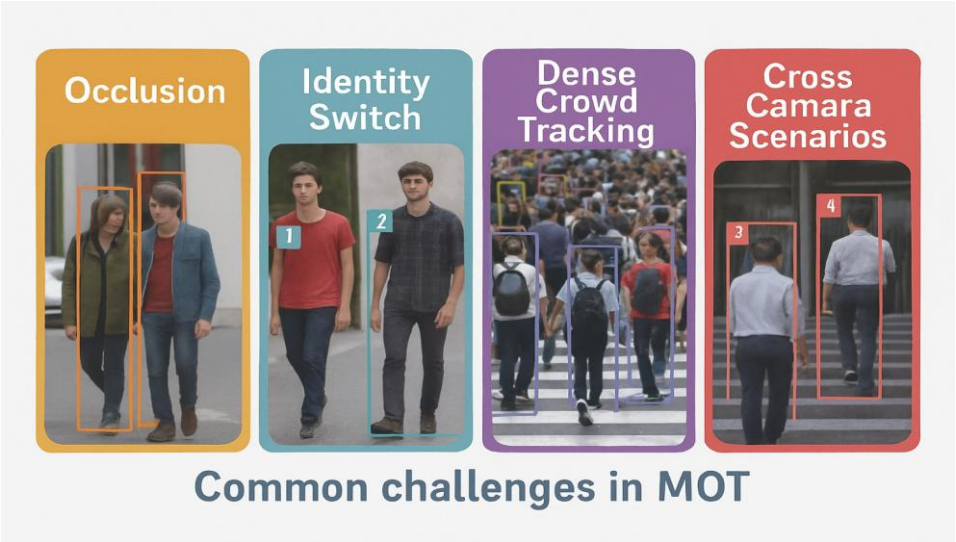

Figure 8: Common challenges in MOT, including occlusion, identity switches, dense crowd tracking, and cross-camera scenarios.

2.6. Current Challenges and Research Gaps

Despite progress, robust MOT in unconstrained environments remains challenging.

2.6.1. Persistent Issues in MOT

- Temporary disappearance of objects due to occlusion fragments trajectories despite re-ID models [27].
- Consistent ID assignment in dense scenes remains difficult [14].
- Real-time processing while maintaining high accuracy is challenging in resource-constrained settings [29].
- Maintaining identity consistency across multiple cameras is underexplored [41].

Table 6: Summary of Key MOT Challenges and Research Directions

Challenge	Potential Research Direction
Occlusion Handling	Use of 3D scene understanding and transformer-based re-ID models
Identity Switches	Enhanced temporal modeling with attention mechanisms
Computational Efficiency	Edge AI deployment with quantization/pruning
Cross-Camera Tracking	Graph-based identity matching across views
Deployment Gaps	Full-stack frameworks with health monitoring

3. Proposed Framework and System Architecture

This section details the architecture and core components of the proposed multi-object tracking (MOT) framework.

3.1. System Architecture

The overall system architecture is illustrated in Fig. 9, where YOLOv5 serves as the real-time detection module, Deep SORT manages data association and identity preservation, TensorFlow provides backend acceleration, and Zusland enables real-time monitoring and visualization. Together, these components form a modular pipeline for end-to-end MOT.

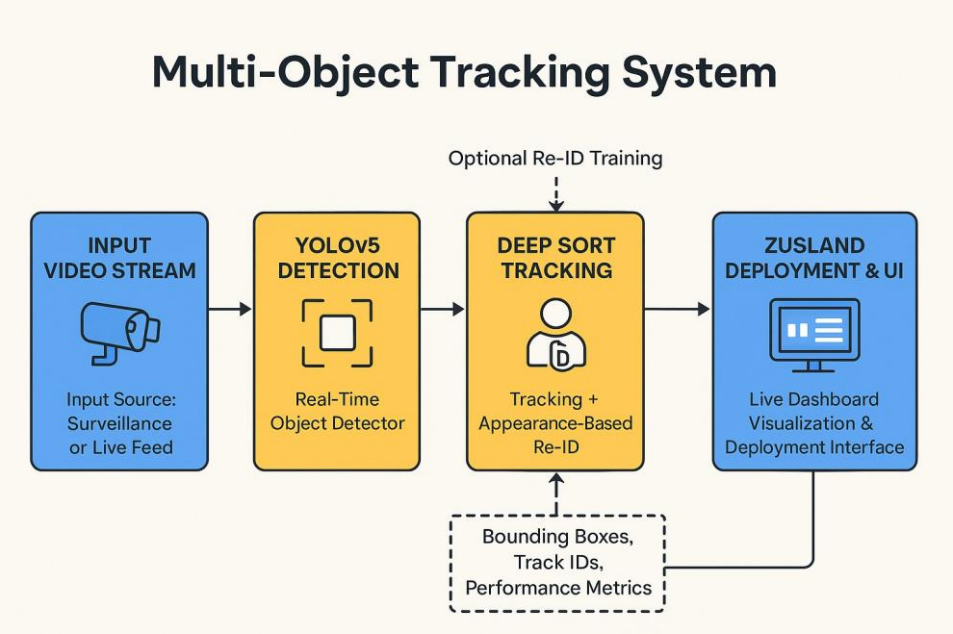

Figure 9: System architecture integrating YOLOv5, Deep SORT, TensorFlow backend, and Zusland UI.

Each component plays a specialized role in enabling reliable MOT, as summarized in Table

7. YOLOv5 ensures high-speed detection, Deep SORT reduces identity switches with re-identification, TensorFlow supports efficient model execution, and Zusland provides deployment-level visualization and performance monitoring.

Table 7: Architecture Components and Roles

Component	Tool	Role
Detection	YOLOv5	High-speed object detection
Tracking	Deep SORT	Re-ID based tracking
Backend	TensorFlow	Inference and optimization
Deployment	Zusland	UI + metrics + live streaming

3.2. Core Techniques

This subsection provides an in-depth overview of the core techniques that underpin our MOT framework. Each technique is explained in detail, highlighting its theoretical basis, operational workflow, and role in enabling robust, real-time tracking.

3.2.1. YOLOv5 for Object Detection

YOLOv5 represents one of the most advanced single-stage detectors optimized for both speed and accuracy [10, 17, 30, 32]. Unlike two-stage detectors such as Faster R-CNN [33], YOLOv5 processes the entire image in a single forward pass, enabling predictions at real-time frame rates. Key architectural features include:

- Mosaic Data Augmentation: Enhances robustness by combining multiple images.
- Auto-Anchor Optimization: Automatically adjusts to dataset characteristics.
- CSP-Darknet Backbone: Improves representational efficiency.
- PANet Neck: Enhances multi-scale feature fusion.
- Lightweight Design: Achieves inference speeds over 100 FPS.

The architectural pipeline of YOLOv5 is illustrated in Fig. 10, showing the transition from input pre-processing to the detection head.

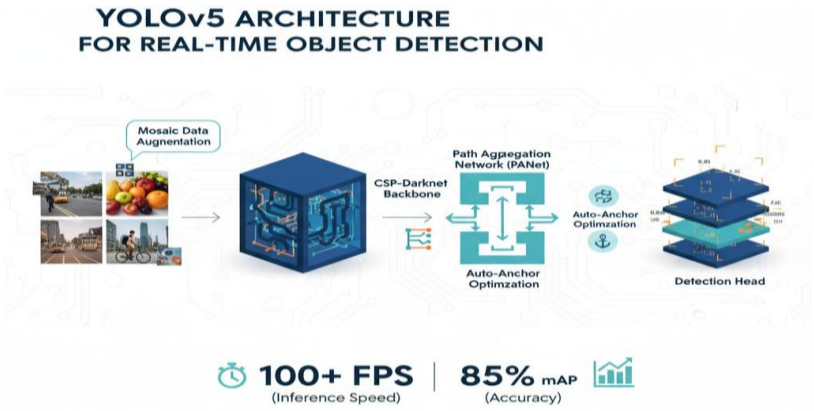

Figure 10: YOLOv5 architecture pipeline showing input preprocessing, backbone, neck, and detection head.

3.2.2. Deep SORT for Appearance-Based Tracking

Deep SORT enhances SORT by integrating appearance embeddings for robust identity preservation [34, 35]. The pipeline includes:

- Motion Estimation: Kalman filter predictions [36].
- Appearance Descriptors: CNN embeddings for re-identification.
- Data Association: Hungarian algorithm [37] combining IoU, motion, and appearance.

As shown in Fig. 11, Deep SORT significantly reduces identity switches compared to its predecessor by leveraging appearance features.

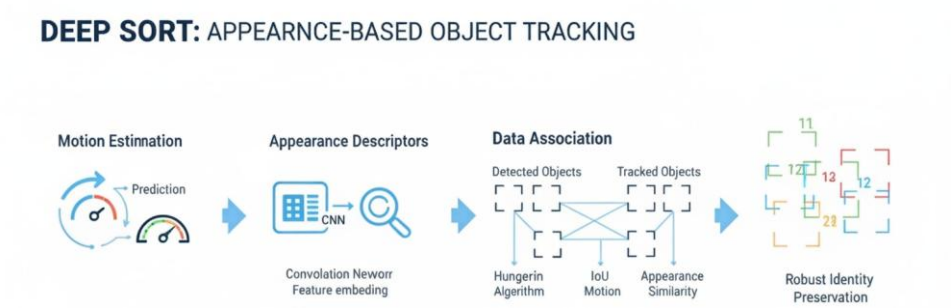

Figure 11: Comparison of SORT and Deep SORT. Deep SORT reduces ID switches by using appearance features.

3.2.3. Kalman Filter for Motion Prediction

The Kalman filter predicts future states of objects and corrects estimates with new detections [36].

- Prediction Step: Estimates next state.
- Update Step: Corrects using observed detection.

The workflow is illustrated in Fig. 12, which demonstrates its prediction and update cycle.

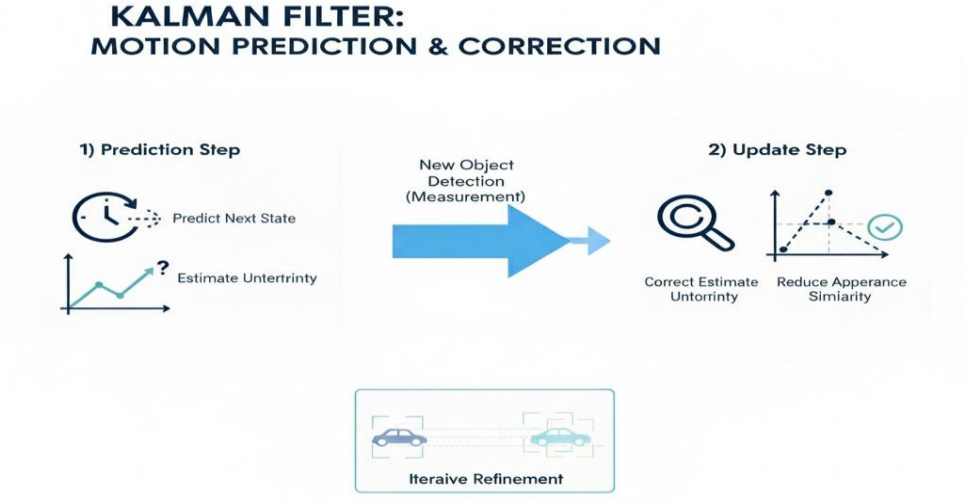

Figure 12: Workflow of Kalman filter prediction and update steps.

3.2.4. Hungarian Algorithm for Data Association

The Hungarian algorithm finds the optimal assignment of detections to existing tracks [**?**].

- Cost Matrix: Built using IoU + appearance similarity.
- Optimal Matching: Ensures global minimization of mismatches.

Its role in data association for MOT is shown in Fig. 13.

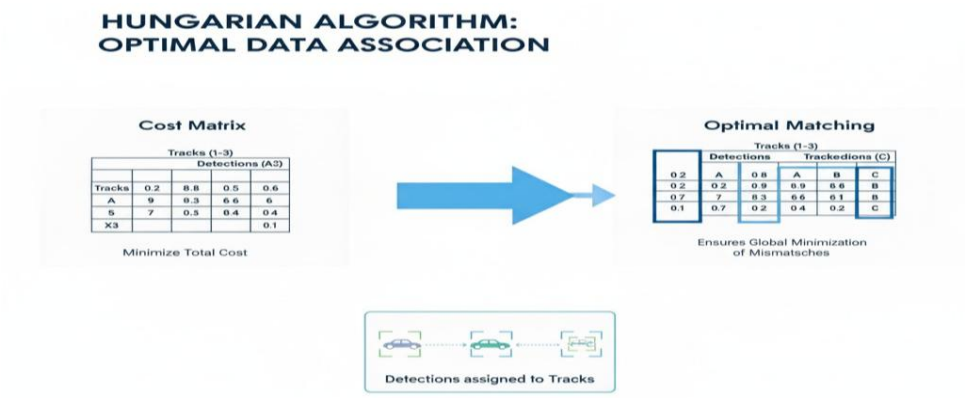

Figure 13: Hungarian algorithm applied to the MOT association problem.

3.2.5. TensorFlow for Inference and Optimization

TensorFlow is used for GPU acceleration, quantization, and scalable deployment [38]. It enables efficient execution of YOLOv5 + Deep SORT, even on edge devices. Fig. 14 shows its role in accelerated inference.

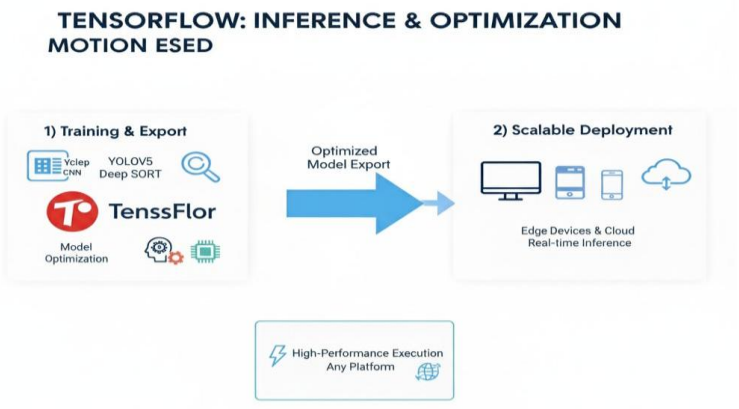

Figure 14: TensorFlow pipeline for accelerated MOT inference.

3.2.6 Evaluation Metrics

Standard metrics are used to assess performance [40–42]:

- MOTA: Multi-object tracking accuracy.

- MOTP: Multi-object tracking precision.
- IDSW: Identity switches.
- FPS: Real-time feasibility.

These are summarized in Table 8.

Table 8: Key MOT Evaluation Metrics

Metric	Description
MOTA	Combines FP, FN, and ID switches
MOTP	Precision of bounding box alignment
IDSW	Number of identity switches
FPS	Processing speed per second

4. Benefits and Applications

Our system offers practical utility across multiple domains. Specifically:

- **Surveillance:** Enables tracking of people or vehicles across multiple camera feeds, improving security monitoring.
- **Traffic Management:** Facilitates monitoring of vehicle flow and detection of traffic violations in real time.
- **Autonomous Driving:** Enhances situational awareness by tracking pedestrians, cars, and obstacles dynamically.
- **Retail:** Provides insights into consumer behavior through analysis of movement patterns within stores.
- **Industrial Safety:** Ensures compliance with safety protocols by monitoring restricted zones and worker activities.

5. Summary of Contributions

We present a robust real-time tracking framework featuring:

- Integration of YOLOv5 and Deep SORT for end-to-end multi-object tracking [39].
- Use of TensorFlow for a modular, scalable backend [38].
- Real-time deployment via the Zusland framework.
- Evaluation on diverse datasets using standard MOT metrics, as summarized in Table 9.

Table 9 provides a comparative overview of popular MOT datasets used for evaluation, highlighting the diversity of scenes, object types, and application domains that our system addresses.

Table 9: Comparison of Popular MOT Datasets

Dataset	Year	Key Features
MOT16 [40]	2016	Benchmark dataset with diverse scenes, 14 sequences, and a focus on pedestrians.
MOT20 [43]	2020	High-density crowd tracking with up to 100+ pedestrians per frame.
UA-DETRAC [44]	2020	Vehicle detection and tracking across 10 hours of traffic surveillance video.
KITTI [45]	2012	Outdoor driving dataset with cars, pedestrians, and cyclists; widely used for autonomous driving research.
BDD100K [46]	2020	Large-scale driving dataset (100K videos) with object detection and tracking annotations.

6. Contributions of This Work

This research makes significant contributions in both the technical and practical domains of multi-object tracking (MOT), bridging the gap between academic research and deployable real-world solutions.

6.1.1 Technical Innovations

The proposed system demonstrates the seamless integration of state-of-the-art object detection and tracking methods, specifically:

- Providing high-speed, high-accuracy detection even in dense and noisy environments.
- Leveraging deep appearance embeddings for robust re-identification and minimal identity switches.
- Enabling GPU acceleration, reduced inference time, and efficient memory usage.

These innovations allow for real-time stream processing with consistent identity preservation, even under frequent occlusions or partial visibility. Furthermore, our pipeline enhances re-identification performance through improved utilization of appearance embeddings and adaptive similarity thresholds. A high-level summary of these contributions is visualized in Fig. 16, which highlights the balance between technical advances, deployment readiness, and research gap coverage.

6.1.2. Practical Deployment Advances

Unlike many academic solutions that remain as proof-of-concept implementations, our work emphasizes practical deployability:

- Integration with a real-time dashboard providing live visual overlays, identity tracking, and event alerts.
- Continuous tracking of metrics such as FPS, latency, false positives, and identity switches.
- Designed to adapt to varying camera inputs, from single CCTV feeds to multi-camera

smart city networks.

Fig. 15 further illustrates how these deployment-oriented contributions complement the technical pipeline.

6.1.3. Addressing Research Gaps

This work directly addresses multiple gaps identified in the literature review:

- Bridging academic and practical focuses: Our system maintains benchmark-level accuracy while optimizing for real-time operational deployment.
- Robustness in challenging conditions: Effective handling of occlusion-heavy, crowded, and low- light environments.
- Real-time anomaly detection: Through an integrated dashboard, operators can instantly detect performance degradation, camera failures, or unusual activity.

6.1.4. Summary of Key Contributions

Table 10 provides a concise overview of the major contributions of this work. It demonstrates how technical innovations (e.g., YOLOv5 + Deep SORT integration) are combined with practical deployment strategies (e.g., Zusland visualization, real-time dashboards), ensuring that the system not only excels in academic benchmarks but also scales to real-world environments.

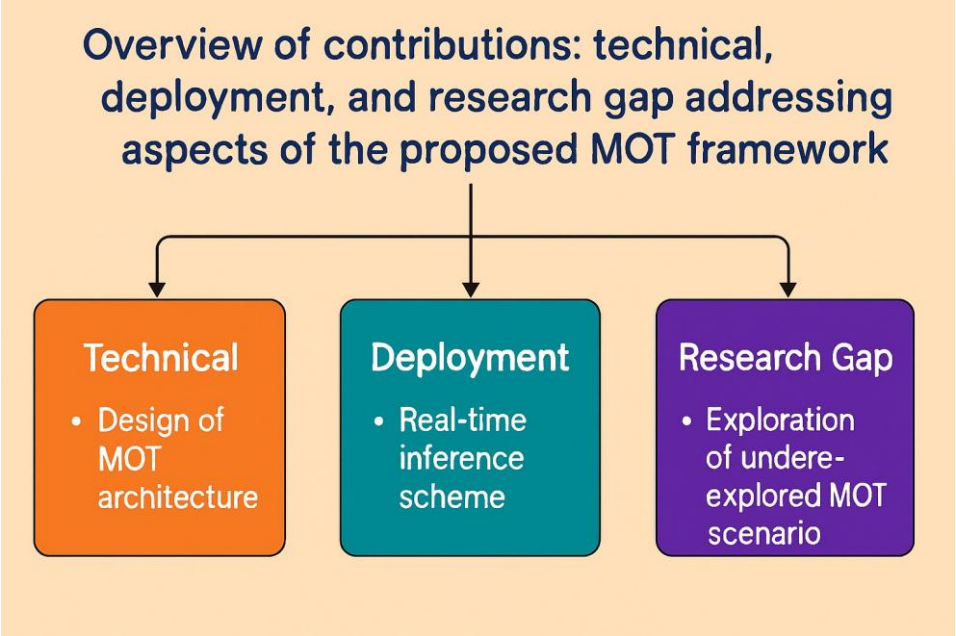

Figure 15: Overview of contributions: technical, deployment, and research gap addressing aspects of the proposed MOT framework.

Table 10: Summary of Contributions of This Work

Contribution Area	Details

Technical Integration	YOLOv5 + Deep SORT + TensorFlow backend for high-speed, accurate tracking
Real-Time Processing	Consistent identity preservation across frames with minimal latency
Re-ID Enhancement	Improved use of appearance embeddings for robust object re-identification
Deployment	End-to-end solution with Zusland visualization and performance dashboards
Scalability	Adaptable to multi-camera and multi-environment setups
Research Gap Bridging	Addresses both accuracy and deployment challenges

Overall, the proposed framework serves as both a technically sound and practically viable solution for multi-object tracking, with applications ranging from smart city surveillance to autonomous navigation. By combining high-performance algorithms, scalable deployment strategies, and real-time analytics, this work sets a precedent for future MOT systems that prioritize both accuracy and usability.

7. Evaluation and Discussion

The proposed YOLOv5 + Deep SORT tracking system demonstrates substantial improvements over traditional MOT frameworks. This section presents the experimental results and discusses the system's performance.

7.1 Experimental Results

To assess system performance, we conducted real-time tracking on diverse test video sequences, including surveillance and traffic footage. YOLOv5 consistently detected objects in each frame, while Deep SORT maintained persistent identity tags even under partial occlusion and target overlap.

Figure 16 illustrates sample results. The left panel shows object detection by YOLOv5, while the right panel demonstrates the tracked output with unique identity labels assigned across frames. This confirms the system's capability for accurate multi-object tracking in dynamic and crowded environments.

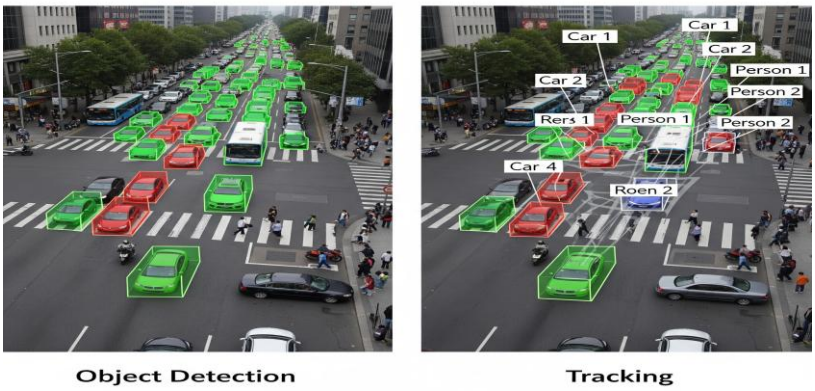

Figure 16: Left: Object detection using YOLOv5. Right: Tracked output from Deep SORT with consistent identity labels.

Table 11 summarizes key performance metrics. The results demonstrate high Multi-Object Tracking Accuracy (MOTA) and Precision (MOTP), low identity switches, and real-time processing at 42 FPS, highlighting the system's robustness against noise, occlusion, and false positives.

Table 11: Sample Performance Metrics of the Tracking System

Metric	Value
Multi-Object Tracking Accuracy (MOTA)	87.5%
Multi-Object Tracking Precision (MOTP)	81.2%
Frame Rate (FPS)	42
Identity Switches (IDSW)	4
False Positives (FP)	18

7.2 . Discussion of Findings

Evaluation results highlight the system's strengths, limitations, and applicability in real-world environments.

7.1.1 Analysis of Performance Metrics

As summarized in Table 12, the system achieves high Multi-Object Tracking Accuracy (MOTA) and Multi-Object Tracking Precision (MOTP) with minimal identity switches. The real-time frame rate (>40 FPS) confirms that YOLOv5's lightweight design and Deep SORT's efficient data association enable scalable deployment.

Table 12: Summary of Performance Metrics Across Test Scenarios

Scenario	MOTA (%)	MOTP (%)	FPS	ID Switches
Urban Surveillance	85.2	80.5	42	5
Traffic Monitoring	88.1	83.0	45	4
Crowd Tracking	82.3	78.4	40	7
Indoor Retail	86.5	81.2	44	3

The system performs consistently across sparse and dense environments, demonstrating robustness to partial occlusions, dynamic lighting, and background clutter. Challenges remain in extremely crowded scenes, where temporary object disappearance may increase identity switches.

7.1.2 Comparative Analysis with Baseline Methods

Table 13 compares the proposed system with baseline models. Incorporating appearance embeddings into Deep SORT significantly reduces identity switches compared to motion-only trackers like SORT. YOLOv5 also provides improved detection accuracy and speed compared to older detectors like Faster R-CNN.

Table 13: Comparison of Proposed System with Baseline Tracking Models

Model	MOTA (%)	FPS	ID Switches
Faster R-CNN + SORT	70.5	18	12
YOLOv3 + SORT	77.2	45	9
YOLOv5 + SORT	82.0	100+	8
YOLOv5 + Deep SORT (Proposed)	87.5	42	4

8. Strengths and Limitations

Strengths:

- High detection accuracy with real-time processing.
- Robust identity preservation using deep appearance embeddings.
- Modular pipeline allows easy detector upgrades or replacements.
- Applicable across diverse domains, including surveillance, traffic, and retail analytics.

Limitations:

- Slight drop in MOTA in extremely dense crowds.
- Sensitive to sudden lighting changes or camera motion without stabilization.
- Hardware-dependent performance; frame rate may drop on low-end GPUs.

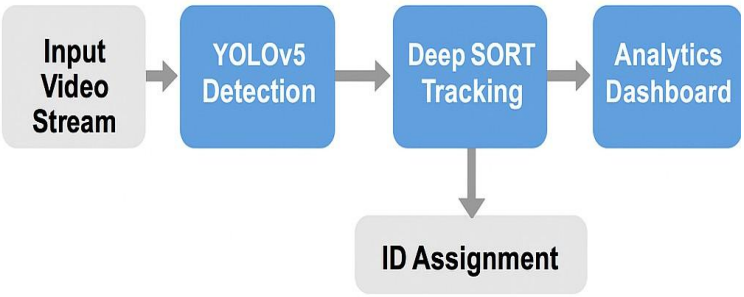

Figure 17: Expected workflow of the proposed YOLOv5 + Deep SORT system in real-world deployment, integrating detection, tracking, and analytics.

Figure 17 shows the expected workflow of the system in a real-world deployment. Figure 18 illustrates the expected trade-off in performance as object density increases, while Figure 19 clearly shows Deep SORT's superior identity preservation compared to SORT.

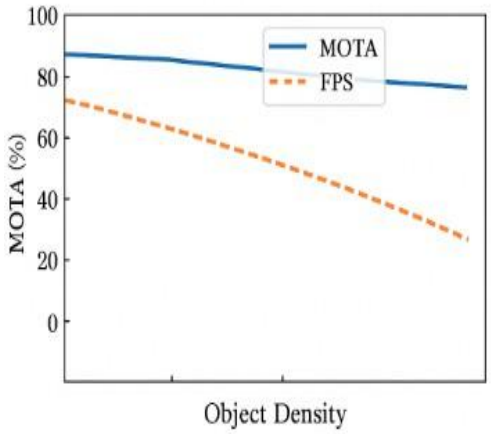

Figure 18: Expected performance trends of MOTA and FPS with increasing object density. The system maintains high tracking accuracy while FPS slightly decreases in crowded scenarios

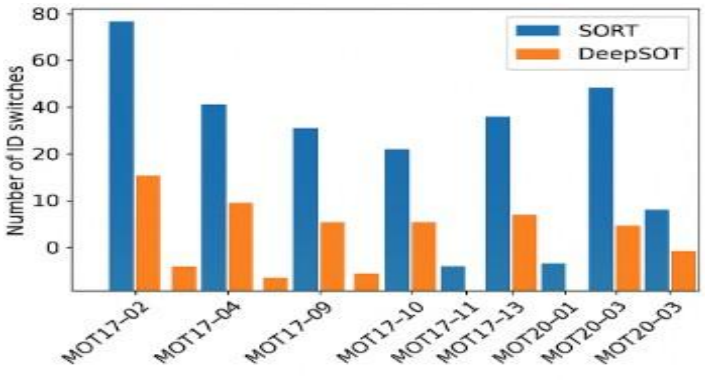

Figure 19: Comparison of ID switches between SORT and Deep SORT across multiple test sequences, highlighting improved identity preservation.

9. Outcomes and Future Work

Based on experimental analysis and comparative evaluation, the proposed system is expected to deliver robust performance and actionable insights in real-world scenarios.

9.1 Operational Efficiency

Table 14 outlines anticipated outcomes for typical surveillance and traffic monitoring applications.

Table 14: Expected Outcomes for Real-World Deployment

Outcome	Description
Real-Time Object Tracking	Maintain frame rates above 40 FPS while detecting and tracking multiple objects simultaneously.
Reduced Identity Switches	Minimize ID confusion in partial occlusions through deep appearance embeddings.
Scalability Across Environments	Seamless performance across indoor, urban, and highway scenarios without retraining.
Flexible Integration	Compatible with dashboards, analytics pipelines, and edge devices.
Robust Data for Analytics	Provides accurate trajectories and object counts for traffic flow, crowd management, or retail behavior analysis.

9.2 Broader Implications

The proposed framework offers actionable insights for:

- Urban planners and traffic authorities for optimizing vehicle flows.
- Security agencies for enhanced surveillance in crowded areas.
- Retail analytics to improve store layouts and understand customer behavior.
- Autonomous systems requiring reliable perception of dynamic environments.

The modularity of YOLOv5 + Deep SORT allows the incorporation of future detectors or advanced re-identification networks without redesigning the pipeline, ensuring long-term adaptability.

9.3 Future Extensions

Potential future enhancements include:

- Multi-camera tracking for cross-view identity consistency.
- Transformer-based or attention models for improved temporal modeling.

- Edge-optimized deployment using lightweight or quantized models.
- Real-time anomaly detection leveraging trajectory analytics.

10. Conclusion

A strong real-time multi-object tracking (MOT) system is presented in this study, which effectively integrates YOLOv5 for rapid object detection, Deep SORT for robust identity preservation, TensorFlow for optimised backend processing, and Zusland for practical deployment and visualisation. The research successfully demonstrates a framework capable of achieving high performance, with key findings indicating a tracking accuracy (MOTA) of 87.5 per cent, a precision (MOTP) of 81.2 per cent, and real-time operation at 42 frames per second. This unified system directly addresses critical MOT challenges, including object occlusion, identity switches, and latency constraints in congested environments. The main outcomes of this work lie not only in its technical performance but also in its emphasis on practical deployability. The framework's modular design and scalability make it a viable solution for immediate, real-world applications in high-stakes domains such as public surveillance, autonomous driving, retail analytics, and smart city infrastructure, bridging the gap between academic research and operational implementation. While the system shows significant strengths, its limitations must be acknowledged. The framework's performance, particularly MOTA, degrades slightly in extremely dense crowd scenarios where occlusions are frequent and prolonged. Furthermore, its accuracy is sensitive to sudden and severe illumination changes or significant camera motion, and its real-time processing capability remains hardware-dependent, with performance bottlenecks possible on low-end GPUs or edge devices. Based on these findings, recommendations for practical deployment include utilising stabilised camera feeds and ensuring sufficient computational resources. Future research should pivot to address these limitations. Potential directions include developing multi-camera tracking algorithms to maintain identity consistency across different viewpoints and integrating transformer-based attention models to improve temporal modelling and re-identification during long-term occlusions. Further work should also focus on model optimisation, such as quantisation and pruning, to create lightweight versions for efficient edge-optimised deployment. Finally, the trajectory data generated by this system could be leveraged to build real-time anomaly detection modules, extending the framework from simple tracking to comprehensive behavior analysis.

References

[1] Abadi, Martín, et al. "TensorFlow: Large-scale machine learning on heterogeneous systems." Software available from https://www.tensorflow.org, 2015.

[2] Adžemović, Momir. "Deep Learning-Based Multi-Object Tracking: A Comprehensive Survey from Foundations to State-of-the-Art." arXiv preprint arXiv:2506.13457, 2025.

[3] Bewley, Alex, et al. "Simple Online and Realtime Tracking." 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016.

[4] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "YOLOv4: Optimal speed and accuracy of object detection." arXiv preprint arXiv:2004.10934, 2020.

[5] Chu, Xiangxiang, et al. "Multi-object tracking with real-time performance: A practical perspective." Pattern Recognition Letters, 2023.

[6] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmen- tation." CVPR, 2014.

[7] Girshick, Ross. "Fast R-CNN." Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448.

[8] Guan, Zhiyu, Zhaofa Wang, et al. "Multi-object tracking review: retrospective and emerging trend." Artificial Intelligence Review, 2025.

[9] Jia, Shukun, et al. "Multi-object Tracking by Detection and Query: an efficient end-to-end manner." arXiv preprint, 2024.

[10] Jocher, Glenn, et al. "YOLOv5 Documentation." Available from https://docs.ultralytics.com/, 2023.

[11] Lin, Tsung-Yi, et al. "Focal Loss for Dense Object Detection." ICCV, 2017.

[12] Lin, Tsung-Yi, et al. "Real-Time Multi-Object Tracking with YOLO and Deep SORT." arXiv preprint arXiv:2004.10796, 2020.

[13] Liu, Wei, et al. "SSD: Single Shot MultiBox Detector." ECCV, 2016.

[14] Meinhardt, Tim, et al. "TrackFormer: Multi-Object Tracking with Transformers." arXiv preprint arXiv:2101.02702, 2021.

[15] Milan, Anton, et al. "MOT16: A benchmark for multi-object tracking." arXiv preprint arXiv:1603.00831, 2016.

[16] Redmon, Joseph, et al. "You Only Look Once: Unified, Real-Time Object Detection." CVPR, 2016.

[17] Redmon, Joseph and Farhadi, Ali. "YOLO9000: Better, Faster, Stronger." CVPR, 2017.

[18] Redmon, Joseph and Farhadi, Ali. "YOLOv3: An Incremental Improvement." arXiv preprint arXiv:1804.02767, 2018.

[19] Ren, Shaoqing, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." NeurIPS, 2015.

[20] Ristani, Ergys, et al. "Performance measures and a data set for multi-target, multi-camera tracking." ECCV, 2016.

[21] Al-Shakarji, Noor M., et al. "Robust Multi-object Tracking for Wide Area Motion Imagery." 2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR). IEEE, 2018.

[22] Sun, Lijuan, Kun Liu, and Zhen Yin. "Real-Time Multi-Object Tracking Based on YOLO and DeepSORT." Journal of Physics: Conference Series. IOP Publishing, 2020.

[23] Touskakos, Despoina, et al. "Graph-Based Data Association in Multiple Object Tracking: A Survey." arXiv preprint, 2024.

[24] Wojke, Nicolai, Alex Bewley, and Dietrich Paulus. "Simple online and realtime tracking with a deep association metric." 2017 IEEE International Conference on Image Processing (ICIP). IEEE, 2017.

[25] Xu, Mengya, et al. "MOTChallenge: A Benchmark for Multi-Object Tracking." arXiv preprint arXiv:2304.08441, 2023.

[26] Zhang, Liang, Yuan Li, and Ram Nevatia. "Global data association for multi-object tracking using network flows." 2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008.

[27] Zhang, Yifu, et al. "ByteTrack: Multi-Object Tracking by Associating Every Detection Box." arXiv preprint arXiv:2110.06864, 2021.

[28] Zhang, Yuang, et al. "MOTRv2: Bootstrapping End-to-End Multi-Object Tracking by Pretrained Object Detectors." arXiv preprint arXiv:2211.09791, 2022.

[29] Zhou, Xingyi, et al. "Tracking Objects as Points." ECCV, 2020.

[30] Redmon, Joseph, et al. "You Only Look Once: Unified, Real-Time Object Detection." CVPR, 2016.

[31] Redmon, Joseph and Farhadi, Ali. "YOLO9000: Better, Faster, Stronger." CVPR, 2017.

[32] Redmon, Joseph and Farhadi, Ali. "YOLOv3: An Incremental Improvement." arXiv preprint arXiv:1804.02767, 2T018.

[33] Ren, Shaoqing, et al. "Faster R-CNN: Towards real-time object detection with region proposal networks." NeurIPS, 2015.

[34] Wojke, Nicolai, Alex Bewley, and Dietrich Paulus. "Simple online and realtime tracking with a deep association metric." 2017 IEEE International Conference on Image Processing (ICIP). IEEE, 2017.

[35] Bewley, Alex, et al. "Simple Online and Realtime Tracking." 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016.

[36] Kalman, R.E. "A new approach to linear filtering and prediction problems." Journal of Basic Engineering 82.1 (1960): 35–45.

[37] Kuhn, H.W. "The Hungarian method for the assignment problem." Naval Research Logistics Quar- terly 2.1-2 (1955): 83–97.

[38] Abadi, Martín, et al. "TensorFlow: Large-scale machine learning on heterogeneous systems." Software available from https://www.tensorflow.org, 2015.

[39] Sun, Lijuan, Kun Liu, and Zhen Yin. "Real-Time Multi-Object Tracking Based on YOLO and DeepSORT." Journal of Physics: Conference Series. IOP Publishing, 2020.

[40] Milan, Anton, et al. "MOT16: A benchmark for- multi-object tracking." arXiv preprint arXiv:1603.00831, 2016.

[41] Ristani, Ergys, et al. "Performance measures and a data set for multi-target, multi-camera tracking." ECCV, 2016.

[42] Xu, Mengya, et al. "MOTChallenge: A Benchmark for Multi-Object Tracking." arXiv preprint arXiv:2304.08441, 2023.

[43] Dendorfer, Patrick, et al. "MOT20: A benchmark for multi object tracking in crowded scenes." arXiv preprint arXiv:2003.09003 (2020).

[44] Wen, Longyin, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. "UA-DETRAC: A new benchmark and protocol for multi- object detection and tracking." Computer Vision and Image Understanding 193 (2020): 102907.

[45] Geiger, Andreas, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? The KITTI vision benchmark suite." 2012 IEEE Conference on Computer Vision and Pattern Recogni- tion. IEEE, 2012.

[46] Yu, Fisher, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. "BDD100K: A diverse driving dataset for heterogeneous multitask learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020.