IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 January 2025, accepted 7 February 2025, date of publication 13 February 2025, date of current version 27 February 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3541177

RESEARCH ARTICLE

Practical Evaluation Framework for Real-Time Multi-Object Tracking: Achieving Optimal and Realistic Performance

JUMABEK ALIKHANOV[1], DILSHOD OBIDOV[2], MIRSAID ABDURASULOV[2], AND HAKIL KIM[1], (Member, IEEE)

[1]Department of Electrical and Computer Engineering, Inha University, Incheon 22212, South Korea
[2]HUMBLEBEE R&D, Incheon 22207, South Korea

Corresponding author: Hakil Kim (hikim@inha.ac.kr)

This was supported in part by the BK21 Four Program funded by the Ministry of Education (MOE) and National Research Foundation (NRF), South Korea, and in part by HUMBLEBEE R&D.

ABSTRACT This paper introduces an enhanced evaluation framework to assess the real-world efficacy of multi-object tracking (MOT) systems, focusing on holistic assessment encompassing detection, ReID (Re-Identification), and tracking components. The Lightweight Integrated Tracking-Feature Extraction (LITE) paradigm is proposed as a novel method that seamlessly integrates ReID features within the tracking pipeline, minimizing computational overhead. Unlike conventional frameworks, which often overlook real-world constraints, our approach benchmarks tracker performance in practical scenarios using off-the-shelf detectors. A significant insight derived from our framework indicates that practitioners can attain a HOTA (Higher Order Tracking Accuracy) score of up to 30% by customizing input resolutions and confidence thresholds. In contrast, those who are unaware of these optimizations may only achieve a HOTA score of 10%. This finding underscores the critical advantage offered by our evaluation method. Comprehensive experiments reveal that LITE enables ReID-based trackers to operate with similar speeds to motion-only systems (uses only motion cues, such as object trajectory and velocity, to detect and track objects over time without incorporating appearance features), without compromising accuracy. Our findings underscore the LITE paradigm's potential to shift the dynamics of MOT, offering a balanced solution between computational efficiency and high-performance tracking. The evaluation framework not only standardizes tracker assessment but also highlights the versatility of LITE across diverse datasets and edge devices. The source code for this research is publicly available at https://github.com/Jumabek/LITE.

INDEX TERMS Multiple object tracking (MOT), real-time tracking, evaluation framework, LITE, ReID.

I. INTRODUCTION

Multi-object tracking (MOT) is a core challenge in computer vision, with applications across surveillance, autonomous driving, and sports analytics [1], [2], [3], [4]. The primary goal of MOT is to accurately maintain object identities across consecutive video frames, a task that becomes increasingly complex in dynamic or densely populated environments. To address this, re-identification (ReID) techniques have been integrated into MOT frameworks, leveraging appearance features to improve identity consistency. However, this often comes at the cost of increased computational load, particularly in real-time applications [5], [6].

The Lightweight Integrated Tracking-Feature Extraction (LITE) paradigm offers a resource-efficient solution by incorporating ReID capabilities directly into the tracking pipeline [7]. This approach removes the requirement for separate inference models or extensive training for identity matching, thereby minimizing computational demands while ensuring competitive tracking performance. It stands out as an effective option for real-time applications in resource-constrained environments.

Existing evaluation frameworks for MOT and ReID-integrated systems typically emphasize specific benchmarks

The associate editor coordinating the review of this manuscript and approving it for publication was Muammar Muhammad Kabir.

that may not fully reflect real-world constraints, particularly in edge computing environments [8], [9]. This paper addresses this gap by introducing a comprehensive evaluation framework designed to assess the entire tracking pipeline—including detection, ReID, and tracking updates—in real-time, focusing on edge-device performance.

Building upon LITE's contributions to MOT, this work integrates LITE within a novel, holistic evaluation framework. We apply it across several benchmark datasets to assess its efficiency and practical value. Our extensive experiments demonstrate the proposed framework's reliability and adaptability, underscoring LITE's strengths for resource-constrained deployment.

The main contributions of this paper are summarized as follows:

- A novel evaluation framework that assesses the complete MOT pipeline provides a realistic tracker performance measure suitable for practical, edge-device scenarios.
- An extension of the LITE paradigm within this framework to highlight its ability to deliver high-speed, low-overhead tracking performance.
- Comprehensive evaluations across diverse datasets and edge environments, establishing the robustness and applicability of both the framework and LITE-enhanced tracking methods.

This work addresses the following research questions:

- How can trackers be evaluated on benchmark datasets in a way that reflects real-world deployment scenarios?
- What is the practical utility of ReID components in MOT, and how can their contributions be quantitatively assessed?
- Can existing tracking-by-detection models be reliably evaluated without considering the full tracking pipeline?
- How do various trackers perform in terms of FPS when deployed on edge devices?

The remainder of this paper is structured as follows: Section II reviews prior work on MOT evaluation and ReID integration. Section III outlines the proposed evaluation framework. Section IV describes the experimental setup, followed by results and analysis in Section V. Finally, Section VII concludes with directions for future research.

II. RELATED WORK

Trackers are typically classified based on their incorporation of ReID components, as illustrated in Table 1. Pure motion-based trackers boast speed but lack appearance feature integration, whereas motion and ReID-based trackers demonstrate identity recognition but at the cost of reduced speeds. The FairMOT model integrates both functionalities yet necessitates specific detector training. In contrast, LITE-applied trackers merge advantages while imposing negligible ReID computation costs, achieving commendable HOTA scores alongside notable speed advantages.

Significant advances in MOT include ByteTrack [8] and OC-SORT [9], known for focusing on motion cues, and DeepSORT [10] and StrongSORT [6], which integrate appearance-based features. End-to-end frameworks such as MOTR [11] offer comprehensive models but face slow inference speeds. LITE makes tracking distinct by seamlessly integrating appearance feature extraction, maintaining real-time performance without additional deep learning models.

Despite advancements, deploying MOT algorithms on edge devices remains limited. Prior studies [12] conducted on similar hardware address this but overlook efficient ReID mechanisms essential for real-world applications. LITE addresses such gaps by introducing lightweight, efficient ReID integration tailored for edge environments. Demonstrated across comprehensive benchmarks (see Section V), proposed LITE trackers offer a blueprint for achieving significant computational efficiency while sustaining high HOTA scores with minimal ReID overhead.

A. BYTETRACK'S EVALUATION FRAMEWORK

Traditional MOT evaluation methods face several shortcomings. Benchmarks such as MOT17 and MOT20 depend on detectors trained specifically on their datasets, limiting the evaluation results' generalizability. Additionally, prior studies often evaluate only specific stages of the tracking pipeline, such as tracking and matching, while reporting frames per second (FPS). This approach neglects the impact of detection and pre-processing on the overall performance of the tracker. Another critical issue is the lack of consideration for resource limitations inherent in edge devices, leading to discrepancies between reported and actual performance metrics in real-world scenarios.

ByteTrack's evaluation framework(Fig. 2) exemplifies these limitations while serving as the foundation for many state-of-the-art (SOTA) trackers. ByteTrack employs a train-validation split where half of the MOT17 training set is used for training and the other half for validation during ablation studies. While effective in controlled settings, this practice often overestimates tracker performance due to the similar distributions and characteristics shared between the training and validation sets. Furthermore, ByteTrack incorporates datasets like MOT17 and CrowdHuman [13] into its training pipeline, optimizing specifically for benchmark performance. This strategy introduces bias and limits the framework's applicability to general scenarios. Additionally, ByteTrack evaluates trackers using public detections provided by benchmarks like MOT17, often generated using detectors trained on the same dataset. This reliance on benchmark-specific configurations reduces the robustness of the reported results and undermines the generalization of trackers to unseen environments.

While ByteTrack's framework is robust within controlled benchmark settings, its methodology highlights significant challenges, including limited generalization due to dataset-specific validation and benchmark overfitting, which

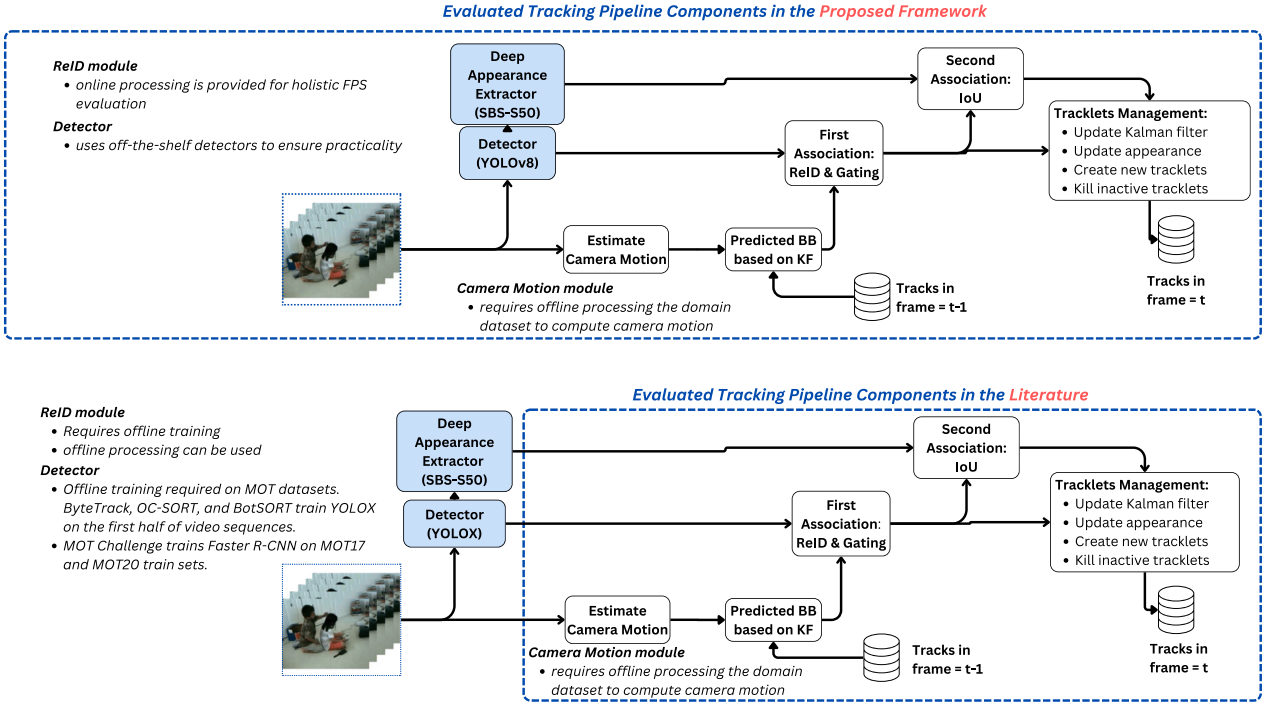


FIGURE 1. Proposed evaluation framework for multi-object tracking. The main novel idea is to evaluate the whole tracking pipeline in real time to provide a practical, realistic evaluation. This allows all hidden costs such as pre-processing, cropping, and resizing done by ReID-enabled trackers to be uncovered. The second main idea is to use a generic and same detector for all trackers. This allows us to evaluate tracker strength by isolating the detector's role. Finally, the importance of considering the detector as part of the tracking pipeline can be seen in Fig. 15 and Fig. 16.

TABLE 1. Comparison of different trackers and their components.

Tracker	Has ReID	No model inference cost	No extra training	Real-time (30FPS)
Motion only				
SORT [14]	–	–	–	✓
OC-SORT [9]	–	–	–	✓
ByteTrack [8]	–	–	–	✓
Motion and ReID				
DeepSORT [5]	✓	–	–	–
Deep OC-SORT [15]	✓	–	–	–
Bot-SORT [16]	✓	–	–	–
StrongSORT [6]	✓	–	–	–
Integrated Motion + ReID				
FairMOT [17]	✓	✓	–	✓
LITE:Trackers (proposed)	✓	✓	✓	✓

obscure tracking algorithms' true performance and practical applicability in diverse real-world scenarios.

III. PROPOSED EVALUATION FRAMEWORK

A. MOTIVATION

Current MOT evaluation methodologies often diverge from practical, real-world use cases, especially on edge devices where speed and resource efficiency are critical. Pre-trained, off-the-shelf detectors like YOLOv8 [18] are typically employed in real-world deployments. However, traditional benchmarks such as MOT17 and MOT20 primarily rely on detectors trained specifically on the benchmark's dataset, which limits the generalizability of performance metrics to real deployment scenarios. Our proposed evaluation framework addresses this gap by providing a holistic evaluation more closely aligned with real-world, edge-device applications. Unlike existing protocols, this framework tests trackers with pre-trained detectors, delivering a realistic performance measure in practical tracking-by-detection setups.

Moreover, traditional studies often measure FPS only for tracking and matching stages [6], [8], [9], [15], overlooking the substantial influence of detection and pre-processing on overall speed. In contrast, our framework extends FPS evaluation by incorporating the detector's impact on speed, offering insights into balancing accuracy and performance in resource-limited, edge-based applications.

B. COMPREHENSIVE AND PRACTICAL EVALUATION FRAMEWORK

Our proposed framework evaluates the complete tracking pipeline, from detection pre-processing and ReID inference to tracking updates, providing a realistic measure of pipeline efficiency on edge devices. For example, domains are the first frame of the KITTI [19] sequence, which is "general-purpose 6 ms for pre-processing, 65 ms for inference, and 243a ms for post-processing. Each of these stages contributes to the overall feasibility of real-time processing.

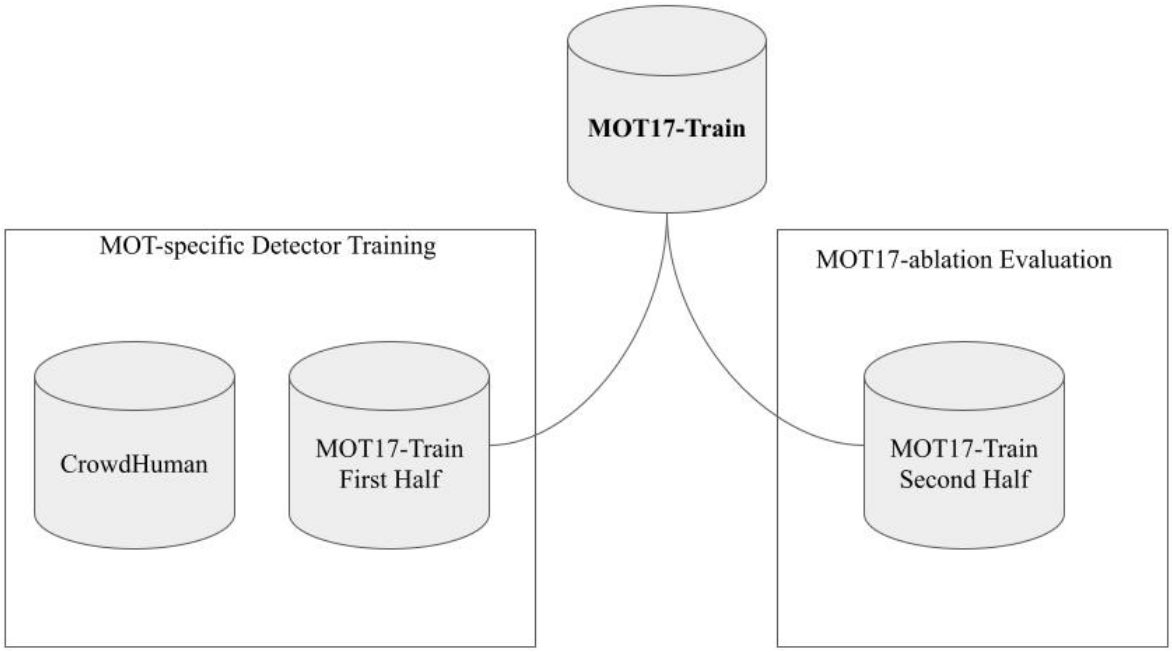


FIGURE 2. The ByteTrack framework splits the MOT17-Train dataset into two halves: the first half is combined with CrowdHuman for detector training, while the second half is used for ablation evaluation, ensuring balanced training and evaluation.

This framework computes an integrated FPS across the pipeline by accounting for each step, particularly valuable for edge devices where GPU memory and processing power are constrained. Capturing the contributions of each component, this framework narrows the gap between reported and actual performance, offering practical insights for deployment scenarios.

Further, current SOTA trackers [8], [9], [15], [16] exploit domain dataset for training their detectors. More specifically, detector YOLOX, which they use, is also a dataset that is used in the MOT17 dataset's first half of video sequences. We argue that this raises the issue of generalizable tracking pipeline performance on unseen domains/datasets. Hence, the proposed pipeline uses general-purpose detectors without training or fine-tuning to offer a practical benchmark. MOT challenge [20], [21], [22] partially addresses this issue by providing public detections (e.g., Faster RCNN's detection) so that each tracker's strength is evaluated without bias on which detector they use. However, the issue in MOT17 and MOT20 is they train Faster RCNN on the train set of the respective dataset [21], [22]. We argue this is not practical enough as when tracking domain changes practical user don't always have the chance to train their detectors for each scenario/domain they want to use tracking. Hence, we propose using a publicly available detection model that does not use data from tracking datasets being evaluated. We can treat such detectors as off-the-shelf detectors as they are readily available and do not require additional user training, making our evaluation framework practical.

Additionally, previous work [7] (see Fig. 15) demonstrated that meaningful tracker comparisons require a detailed exploration of detector settings, including input resolution and confidence thresholds to filter out low-likelihood objects. The main insight was that tracker rankings could not be conclusively determined from experiments based on a single confidence threshold. For example, in the MOT17 dataset, an input resolution of 1280 and a detection confidence threshold of 0.25 provided optimal results.

In this extended version of our work, we aim to further investigate the impact of input resolution, confidence thresholds, and dataset characteristics on tracking performance. To achieve this, we replicated our experiments using the MOT20 dataset (Fig. 16). Results show that the optimal configurations for input resolution and confidence thresholds vary depending on the dataset. Specifically, the MOT20 dataset, which contains denser crowding and smaller object scales, lower confidence thresholds, and higher resolutions, proved more favorable for optimal tracking performance. In contrast, for the MOT17 dataset, previous experiments suggested a confidence threshold of 0.25% as optimal, and different configurations were observed due to the less dense nature and larger object scales in MOT17. Fig. 6 visualizes the height scale ratio distribution for both the MOT17 and MOT20 datasets, emphasizing the variability in detected object heights across videos. The following methodology was used for this analysis: The mean height of bounding boxes for each video is first determined and scaled relative to the video's resolution. These video-level averages are then

averaged across all videos to ensure equal contribution from each video in accordance with the HOTA metric, which ensures fair evaluation across varying video conditions. This approach prevents bias from larger or more object-dense videos and allows for a more balanced comparison of tracking performance. The distribution reveals that due to the denser crowding and smaller object scales in MOT20, the ideal configurations for image resolution and confidence thresholds differ from those of MOT17. Specifically, for MOT20, lower thresholds and higher resolutions proved more favorable, with a confidence threshold of 0.05% emerging as optimal. In contrast, MOT17 benefited from a higher confidence threshold of 0.25%, as shown in the corresponding figures (Fig. 15 and Fig. 16). These insights emphasize the importance of considering dataset-specific characteristics when determining optimal configurations for tracking and evaluation. These findings demonstrate the value of dataset-specific evaluations and the necessity to adjust configuration parameters for diverse tracking scenarios, ensuring that tracking systems adapt to each dataset's unique characteristics.

C. EVALUATION METHODOLOGY OF REID COMPONENT IN TRACKING PIPELINES

Numerous ReID-enabled trackers have been introduced in multi-object tracking (MOT), each demonstrating state-of-the-art performance through improved identity maintenance across frames [5], [6], [15], [16]. Traditionally, ReID components contribute to MOT accuracy by providing appearance-based descriptors critical for distinguishing individuals in dense scenes. However, the computational demands of ReID, including separate training and inference networks, pose challenges for real-time tracking on limited hardware.

To address these constraints, the LITE framework leverages intermediary layer features of the object detection model itself to generate appearance descriptors, eliminating the need for a dedicated ReID model. LITE achieves significant computational efficiency without sacrificing accuracy, as demonstrated in prior benchmarks. Despite these advancements, questions remain regarding the standalone impact of ReID within tracking systems, as motion-based trackers have achieved comparable performance without incorporating ReID [8], [9]. This requires a standardized evaluation framework that isolates and directly assesses the ReID component.

To our knowledge, no existing methodology specifically benchmarks ReID components in MOT pipelines. This study introduces a novel evaluation methodology for ReID, shown in Fig. 3, designed to assess the direct contribution of ReID-based appearance descriptors within MOT systems. In this framework, ReID evaluation is conducted by computing both positive and negative match scores. **Positive match scores** assess ReID performance within a given track across consecutive frames. For instance, in Fig. 3, for Track 1 at Frame 1, three positive match scores are computed by comparing Track 1's descriptor in Frame 1 to its descriptors in subsequent frames. **Negative match scores**, in contrast, assess the distinctiveness of each track's appearance by comparing it against other tracks within the same frame, thus testing ReID's ability to differentiate between individuals.

D. REAL-TIME PROCESSING REQUIREMENTS FOR EDGE DEVICES

Achieving real-time performance on resource-constrained edge devices requires that each module in the tracking pipeline—detection, ReID processing, and track management—operates within real-time bounds. Instead of isolated FPS calculations, this framework computes an integrated FPS that reflects the complexity of each frame, providing a more accurate measure of real-world performance. Both accuracy, as measured by HOTA, and speed are influenced by parameters like detection confidence thresholds and image resolution. By evaluating these effects across the entire pipeline, this framework enables practitioners to capture the practical performance implications for edge devices in real-time applications.

Traditional MOT approaches often separate detection and tracking processes, measuring only tracker update speed [6], [16], [23]. However, this framework evaluates end-to-end pipeline speed, reflecting real-world scenarios where frame processing times fluctuate due to varying detection density, model complexity, input resolution, and other settings. Such a comprehensive approach ensures that practical speed and performance metrics are captured, making the framework highly relevant to real-time applications on resource-constrained devices.

IV. EXPERIMENT SETTING

A. DATASETS

The datasets used in the experiments include MOT17 [21], which consists of 21 sequences with a resolution of 1920 × 1080 and an average of 650 frames (SD 200), serving as a standard multi-object tracking dataset. MOT20 [22] includes 8 sequences at 1920 × 1080 resolution, with an average of 800 frames (SD 250), presenting challenges in crowded scenes. The KITTI dataset [19] features 50 sequences at 1242 × 375 resolution, averaging 120 frames (SD 30), and is used for evaluating autonomous driving scenarios. The VIRAT-S dataset [24] comprises 100 sequences with a resolution of 1280 × 720 and an average of 1000 frames (SD 300), designed for fast-tracking in action detection applications. Lastly, PersonPath22 [25] contains more than 100 sequences with various resolutions, averaging 500 frames (SD 150) and includes various scenarios such as indoor, outdoor, and mobile environments. Dataset characteristics are shown in Table 2. MOT17 and MOT20 datasets include both train and test sequences. However, ground truth for test sequences is unavailable except for submission through portals [21], [22]. Visual samples of datasets are provided in Fig. 4.

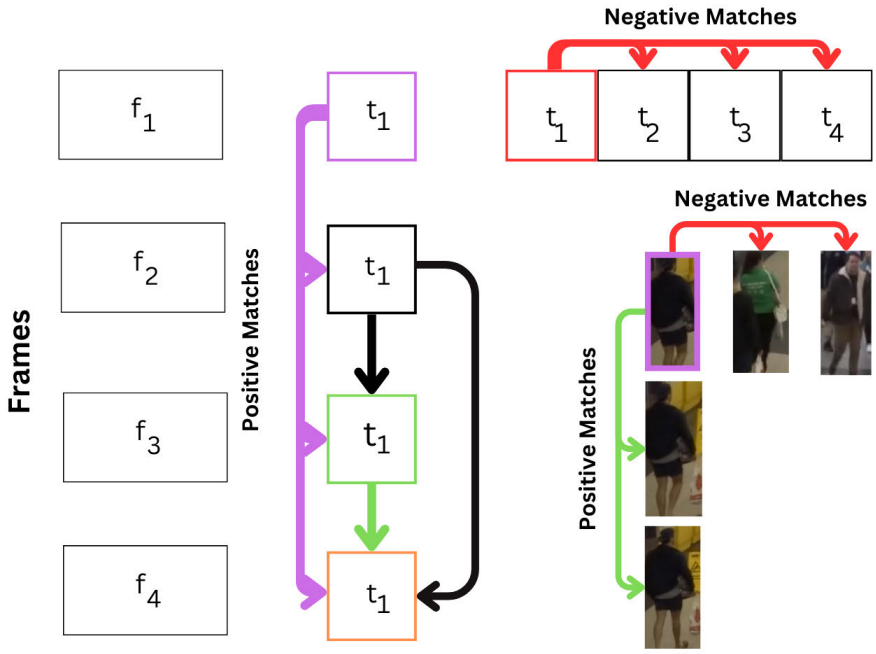


FIGURE 3. Proposed evaluation methodology for ReID networks of MOT pipelines.

TABLE 2. Overview of datasets used in experiments. The average and standard deviation of video length (i.e., number of frames) are shown. For MOT17 and MOT20, only training sequences are included.

Dataset	Description	Resolution	Frames	#Sequences
MOT17	Multi-object tracking	1920x1080	650 ± 200	7
MOT20	Crowded scenes	1920x1080	800 ± 250	4
KITTI	Autonomous driving	1242x375	120 ± 30	21
VIRAT-S	Action detection	1280x720	1000 ± 300	100
PersonP22	Diverse scenarios	Various	500 ± 150	98

B. IMPLEMENTATION DETAILS

Experiments use two code repositories, The first is StrongSORT [26] selected to implement all SORT-like trackers, which are relatively simple and contain fewer settings and parameters. This is achieved by inhering the author's implementation of StrongSORT and DeepSORT. With simple adaptation add SORT. To add LITE:DeepSORT, replace the external ReID module used in DeepSORT with LITE to obtain ReID features without external inference or pre- or post-processing steps. While this code is sufficient to show the strengths of LITE, more recent trackers are also added by adopting the repository BoxMOT (also known as yolo_tracking) [23] which contains trackers such as ByteTrack, OC-SORT, Deep OC-SORT, BoTSORT. Both repository's code is adjusted to follow proposed evaluation framework requirements such as holistic evaluation, and real-time tracking.

Experiments use YOLOv8m for real-time application, with a confidence threshold of 0.25 and an image resolution of 1280. Despite some state-of-the-art trackers using larger architectures for better accuracy, YOLOv8m is chosen for its real-time tracking capabilities. The FRCNN version of ground truth is used for MOT17. For KITTI and VIRAT-S, only pedestrian and person classes are evaluated. The HOTA metric is used to evaluate tracker accuracy. Experiments were conducted on an Intel Core i9-12900K at 5.2 GHz, an NVIDIA GeForce RTX 3090 with 24 GB VRAM, and 64 GB DDR4 at 3200 MHz.

Edge Device Experiment: StrongSORT and DeepSORT's ReID models were excluded from experiments due to insufficient documentation on TensorRT conversion. For all experiments, an NVIDIA Jetson Orin NX with an ARMv8 Processor (rev 1, v8l) with 6 cores at 2.0 GHz and an NVIDIA Tegra Orin (nvgpu) with 7.2 GB memory were used. The device settings were Jetson Clocks—inactive and NV Power Mode—MAXN. FPS benchmarks were measured with no other significant processes running, using the first video sequence of each dataset.

C. EVALUATION METRICS

Traditional metrics like Multi-Object Tracking Accuracy (MOTA) [27] and Identification F1 Score (IDF1) [28] has limitations that can skew tracking system evaluation. MOTA emphasizes detector accuracy and potentially misleading results when detection is challenging. IDF1 rewards association capabilities but can overlook detection accuracy. MOTA is computed based on False Positives (FP), False Negatives

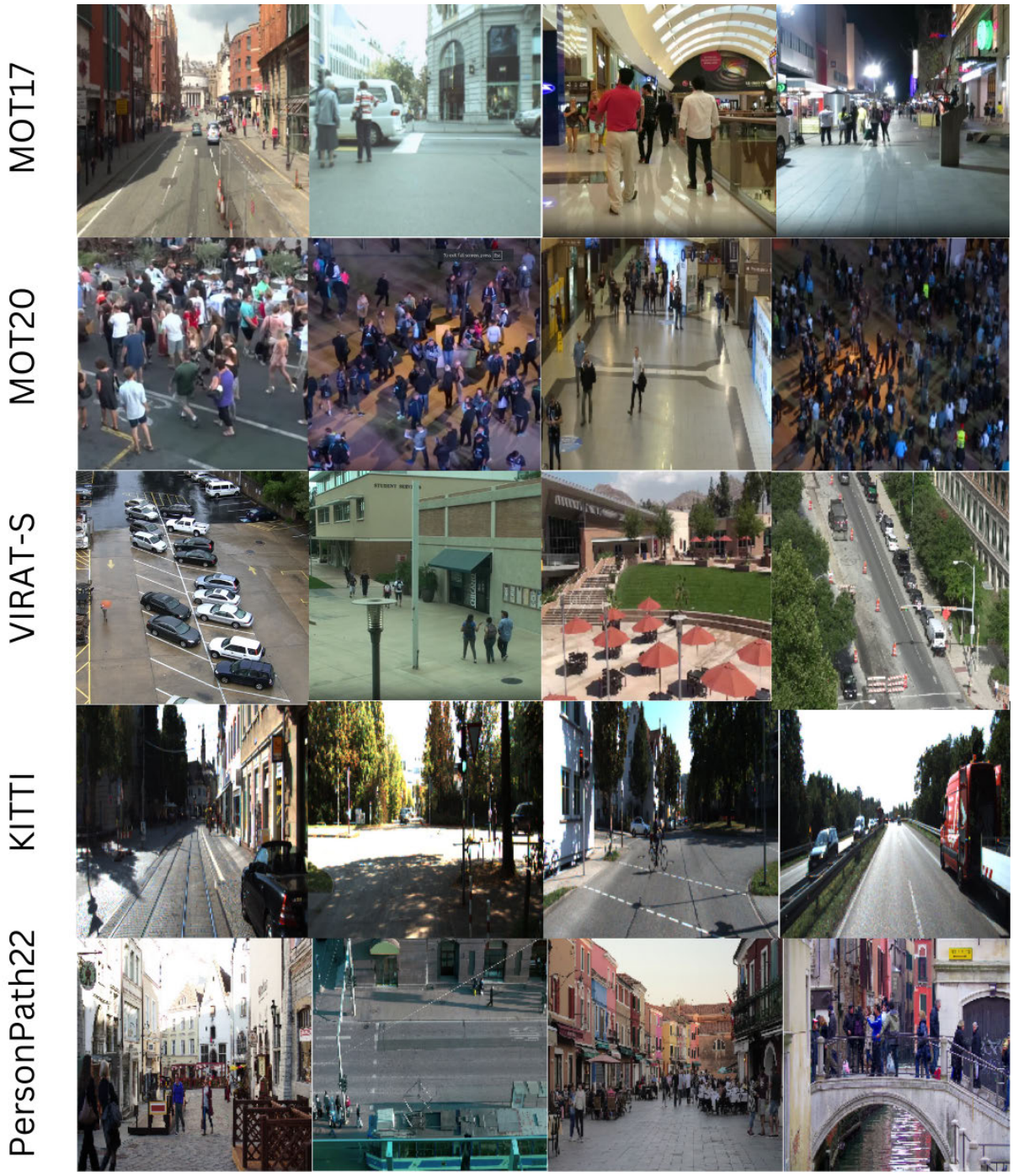


FIGURE 4. Sample frames from various datasets were used in the analysis.

(FN), and Identity Switches (IDSW), while IDF1 is computed based on True Positives (IDTP), Identification False Positives (IDFP), and Identification False Negatives (IDFN).

$$\text{MOTA} = 1 - \frac{\text{FP} + \text{FN} + \text{IDSW}}{\text{Total Detections}} \tag{1}$$

$$\text{IDF1} = \frac{2 \times \text{IDTP}}{2 \times \text{IDTP} + \text{IDFP} + \text{IDFN}} \tag{2}$$

In contrast, HOTA balances detection accuracy and identity association quality, providing a comprehensive evaluation of a tracker's capability. By focusing on HOTA, our framework offers a holistic view of tracker performance, ensuring genuine advancements are reflected in tracking technology evaluations. AssA is the metric to measure association accuracy and DetA is for detection accuracy.

$$\text{HOTA} = \sqrt{\text{DetA} \times \text{AssA}} \tag{3}$$

V. EXPERIMENTS

A. COMMON PITFALLS OF CONVENTIONAL EVALUATION FRAMEWORKS

The experiments aim to highlight common pitfalls in previous evaluation settings and to compare the LITE versions of ReID-based trackers against their original counterparts in terms of speed. In particular, while emphasizing conventional comparisons, we use the same generic detector, YOLOv8m, applied consistently across all trackers to ensure fairness. To assess the robustness of the trackers, we conducted experiments in various scenarios using diverse datasets. Specifically, all trackers, including those that use LITE, were evaluated for their speed advantages and generalizability on five datasets - MOT17, MOT20, PersonPath22, KITTI and VIRAT-S, employing the overall MOT accuracy metric HOTA and the real-time usage metric FPS in Table 3, where HOTA serves as the accuracy metric and FPS as the real-time speed measurement of the tracking pipeline. Qualitative results are also provided to underscore the differences between trackers, highlighting the impact of ReID capabilities, for instance. Fig. 11 and Fig. 14 illustrate qualitative comparisons across various trackers.

Additionally, Table 4 presents a thorough performance analysis, comparing motion-based and ReID-enabled trackers across front-view, top-view, and general settings. This analysis reveals differences in tracking accuracy, association quality, and detection efficiency, particularly under challenging conditions such as occlusions and dynamic movement. In top-view scenarios, motion-based trackers are often highly effective, as they can accurately track straightforward movement patterns without relying on appearance features. In contrast, in front-side views with significant occlusions, ReID-enabled trackers offer enhanced performance by utilizing appearance features to maintain consistent identities despite visual disruptions. This contrast is clearly demonstrated by the performance of BoTSORT and OC-SORT, where BoTSORT excels in handling heavily occluded front-side views, while OC-SORT shows strong performance in top-view scenarios with fewer occlusions.

Fig. 12 and Fig. 13, alongside side-view and top-view comparisons, effectively illustrate these distinctions across various scenarios.

It may be tempting to conclude the best tracker based on the results in Table 3. However, further investigation reveals that altering one parameter of the detector model, specifically the confidence threshold, can yield differing observations and rankings. Fig. 5 shows this ranking discrepancy when a detection confidence threshold changes. This implies by the single parameter of the tracking pipeline (in this case confidence threshold), the ranking of trackers may change. For instance, when the detector confidence threshold is adjusted from 0.25 to 0.05, different results emerge. Interestingly, the trackers perform optimally in MOT17 with a higher confidence threshold, whereas for MOT20, a lower confidence threshold is more effective. This observation necessitates the need for a better evaluation framework for MOT, an effort attempted in this paper.

Insights: Evaluating a tracker at a single confidence threshold can lead to misleading ranking of trackers. Only manipulating the confidence threshold of the detector can change the ranking of the best tracker (DeepSORT as an example in Fig. 5). Hence, tracker comparisons should be made in multiple confidence thresholds. The second insight is that using MOT-train-specific public detections is showing higher results for MOT17 and MOT20. However, this is not the practical performance of the trackers as found in Table 3 for generic detector-based tracking pipelines. This proves the need for the proposed practical evaluation framework.

B. COMPARISON OF TRACKERS WITH PROPOSED EVALUATION FRAMEWORK

This experiment presents key insights derived from the proposed evaluation framework, designed to analyze the effects of varying detector settings on tracking pipeline performance. Fig. 15 and Fig. 16 display HOTA scores of tracking pipelines from the MOT17 and MOT20 datasets, respectively. Each subfigure was generated by evaluating each tracker under different detection confidence thresholds. While no definitive optimal input resolution was identified, the 1280×1280 setting emerged as the most suitable overall. Focusing on the 1280×1280 resolution for both datasets, an optimal confidence threshold of 0.25 was identified for MOT17, whereas 0.05 was optimal for MOT20. This observation highlights that not only tracking the pipeline's performance is very sensitive to underlying factors of the pipeline, but also the optimal confidence threshold depends on the specific tracking scenario (in this case, the dataset). To further explore the unique characteristics of these two datasets, crowd density and object scale distributions are visualized in Fig. 6. As shown, MOT20 contains smaller-scale objects and denser video sequences compared to MOT17. This is a plausible explanation of the disparity in optimal confidence threshold for two datasets.

An additional key insight facilitated by the proposed evaluation framework, originally introduced in [7] and reiterated here for completeness, is presented exclusively for the MOT17 dataset in Fig. 15 and Fig. 17. These figures illustrate how detection settings influence both HOTA and FPS metrics. A primary observation from these results is that minor differences in HOTA scores (1%-2%) across trackers are insufficient to categorically declare one tracker superior. Variability in rankings can arise from numerous factors, including the limited scale of benchmark datasets. Detector settings are often found to impact tracking performance more substantially than inter-tracker differences. Typically, higher resolutions necessitate increased confidence thresholds to balance accuracy and speed. Conversely, lower thresholds yield more detections, potentially leading to additional identity switches and increased computational demands, particularly for trackers incorporating a ReID component.

TABLE 3. The LITE trackers are highlighted to show the boost in FPS while maintaining HOTA scores, demonstrating effective tracking performance across various datasets. Discrepancy between using an off-the-shelf detector (proposed framework) and MOT-trained FRCNN detections from MOT challenge benchmarks already show the necessity of the proposed pipeline. MOT-trained here refers to the case when the detector is trained on the MOT dataset exclusively to learn the domain. Specifically, larger HOTA scores for MOT-trained cases are due to F-RCNN being trained on a similar tracking dataset.

Tracker	MOT17		MOT20		PersonPath22		KITTI		VIRAT-S	
	HOTA ↑	FPS ↑	HOTA ↑	FPS ↑	HOTA ↑	FPS ↑	HOTA ↑	FPS ↑	HOTA ↑	FPS ↑
Pure Motion										
SORT	40.3	39.7	20.1	33.7	35.2	30.8	41.1	43.1	28.3	42.7
OCSORT	43.9	37.2	25.2	32.4	40.3	26.6	43.9	25.5	31.9	37.0
ByteTrack	43.8	39.4	25.2	33.3	40.5	27.0	44.9	27.2	32.7	37.2
Motion and ReID										
DeepSORT	43.7	10.5	24.8	4.8	38.3	15.1	42.6	38.3	33.6	33.9
Deep OC-SORT	43.7	13.0	24.9	8.9	39.9	13.9	43.7	18.7	31.7	24.0
BoTSORT	40.9	24.9	20.8	19.3	39.1	18.4	33.7	22.5	31.1	30.0
StrongSORT	44.5	4.5	26.1	2.4	38.0	5.9	44.0	23.6	33.4	21.8
Integrated Motion + ReID										
LITE:DeepSORT	43.0	26.7	25.2	15.9	38.0	26.1	42.8	40.8	33.7	40.2
LITE:Deep OC-SORT	43.7	34.8	25.3	29.6	39.8	25.0	44.1	23.0	31.5	36.5
LITE:BoTSORT	40.8	38.2	21.1	31.8	39.2	26.4	33.0	24.1	31.2	38.0
MOT Challenge with MOT-trained FRCNN detections										
ByteTrack	47.5	-	41.1	-	-	-	-	-	-	-
OC-SORT	47.6	-	40.6	-	-	-	-	-	-	-
DeepSORT	44.9	-	35.1	-	-	-	-	-	-	-

TABLE 4. Comprehensive tracking performance comparison across various videos and scenarios, evaluating different trackers using multiple metrics.

Video Name	Tracker	HOTA	MOTA	IDF1	AssA	DetA	LocA	FN	FP	TP
uid_vid_00008 (Front View)	OCSORT	30.1	42.6	35.7	20.1	46.2	83.4	1037	386	1106
	BoTSORT	34.3	45.3	39.6	25.6	46.8	84.4	1063	279	1080
uid_vid_00019 (Top View)	OCSORT	43.4	38.4	51.4	57.5	33.8	84.7	4018	446	2097
	BoTSORT	31.8	24.2	32.3	48.3	21.8	85.5	4799	230	1298
uid_vid_00158	DeepSORT	50.4	73.3	61.6	40.4	63.1	83.2	246	137	525
	LITE-DeepSORT	57.4	73.4	70.2	52.4	62.9	83.1	248	139	534

Insights Optimal input resolution and confidence thresholds are highly scenario-dependent, with potential impacts on pipeline performance ranging from HOTA scores of 10% to as high as 30%. The second insight is that a small gap in the HOTA score for trackers (e.g., 1%-4%) is insufficient to rank the tracker's performance.

C. ROBUSTNESS EVALUATION OF REID MODELS

The proposed evaluation framework introduces an isolated methodology for assessing ReID components within MOT pipelines, as illustrated in Sec. III (Also, in Fig. 3). Each feature vector, denoted as t_j, represents the appearance features of a track at a given frame. For each frame f_i, these feature vectors are compared to calculate similarity scores. **Positive match** scores are derived by comparing appearance features within the same track across frames, while **negative match** scores are computed by comparing the track of interest against other tracks. This framework quantifies ReID model performance by evaluating their capacity to accurately identify and distinguish individuals over multiple frames.

ReID model performance is measured through a Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) score. A high-level interpretation of the ROC-AUC metric refers to the probability that the model assigns a higher score to positive matches than to negative matches [29]. AUC is widely recognized as a reliable evaluation metric across various data science fields [30]. The ROC curve visually represents the model's ability to differentiate between true positives and false positives across threshold values, while the AUC score provides a quantifiable measure of this performance. To calculate ROC and AUC

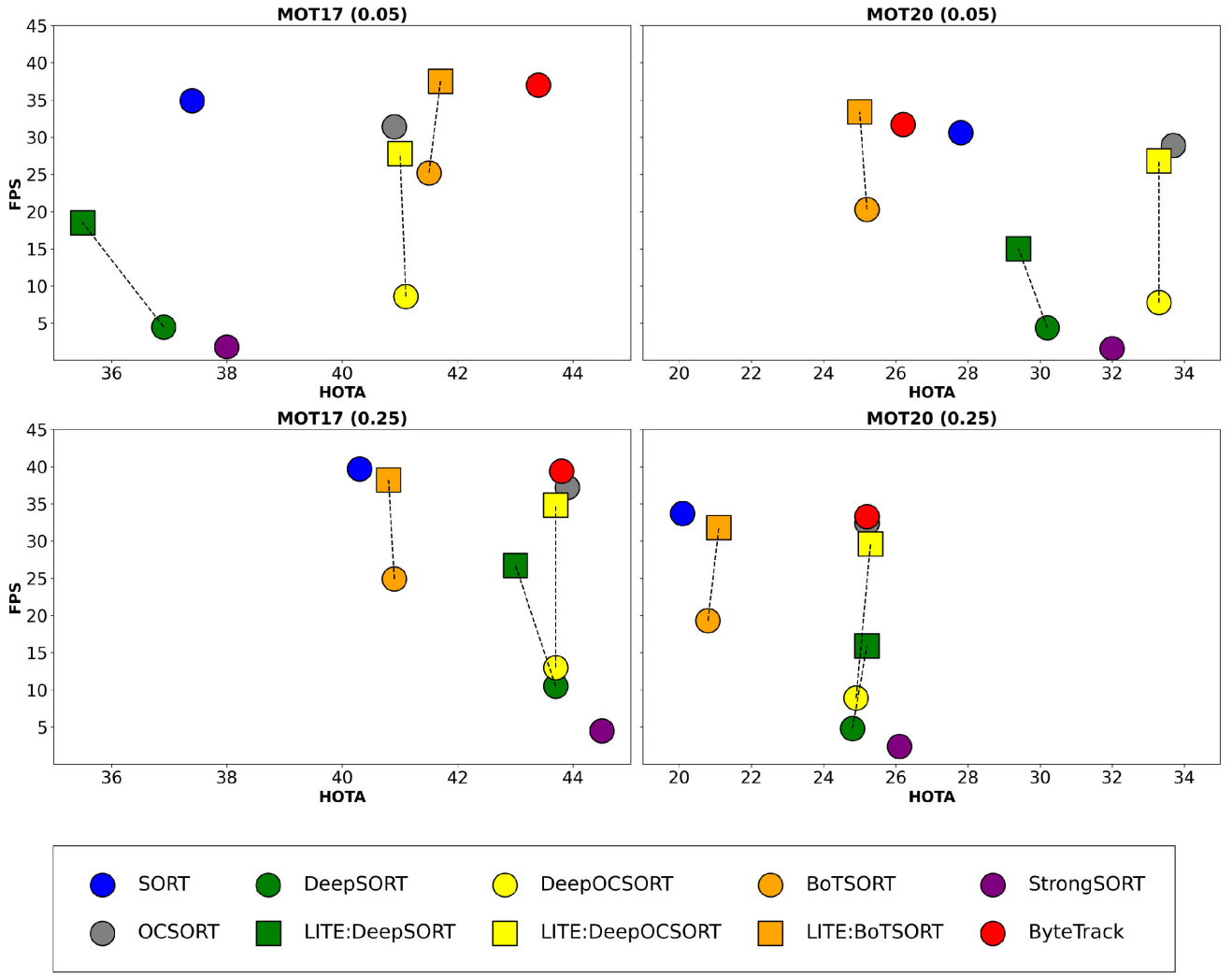


FIGURE 5. Comparison of tracker performance on the MOT17 and MOT20 datasets. The scatter plots display each tracker's HOTA and FPS scores, with colored markers representing individual trackers and dashed lines connecting each LITE variant to its original counterpart. ByteTrack already uses 0.1 as a lower confidence threshold internally which makes it less sensitive to threshold change.

scores, we construct label and score arrays for evaluation: An equal number of positive and negative matches are sampled to ensure balanced data for unbiased evaluation.

$$y_{\text{true}} = [\ \underbrace{1, \ldots, 1}_{\text{len(pos\_matches)}}\ ,\ \underbrace{0, \ldots, 0}_{\text{len(neg\_matches)}}\ ]$$

$$y_{\text{scores}} = [\ \underbrace{0.8, \ldots, 0.7}_{\text{len(pos\_matches)}}\ , \underbrace{0.12, \ldots, 0.35}_{\text{len(neg\_matches)}}]$$

The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) over a range of threshold values, while the AUC score encapsulates the model's discriminative ability as a single value, where scores closer to 1 indicate stronger performance.

For this evaluation, we utilized the MOT20-01 video sequence due to its minimal occlusions and relatively low number of individuals per frame, allowing a clearer assessment of ReID models in a simplified context. Fig 7 shows the distribution of re-identification scores. Notably, LITE's appearance features, though not explicitly trained for identity distinction, demonstrate a discernible separation between positive and negative match scores. Conversely, DeepSORT's ReID model, trained on the Market-1501 dataset [31], shows limited discriminative power, possibly due to its use of early CNN architectures with constrained learning capabilities. Fig. 8 quantitatively compare models using ROC-AUC, revealing that Deep OC-SORT's OSNet [32] and StrongSORT achieve superior scores. Surprisingly, LITE exhibits stronger distinctive power than DeepSORT.

Insight: Despite differences in discriminative power, both Deep OC-SORT and its LITE variant achieve similar HOTA scores. This suggests that ReID components of trackers may be underutilized, with motion-based components (e.g., bounding box estimation with Kalman filters) overshadowing the potential benefits of ReID features.

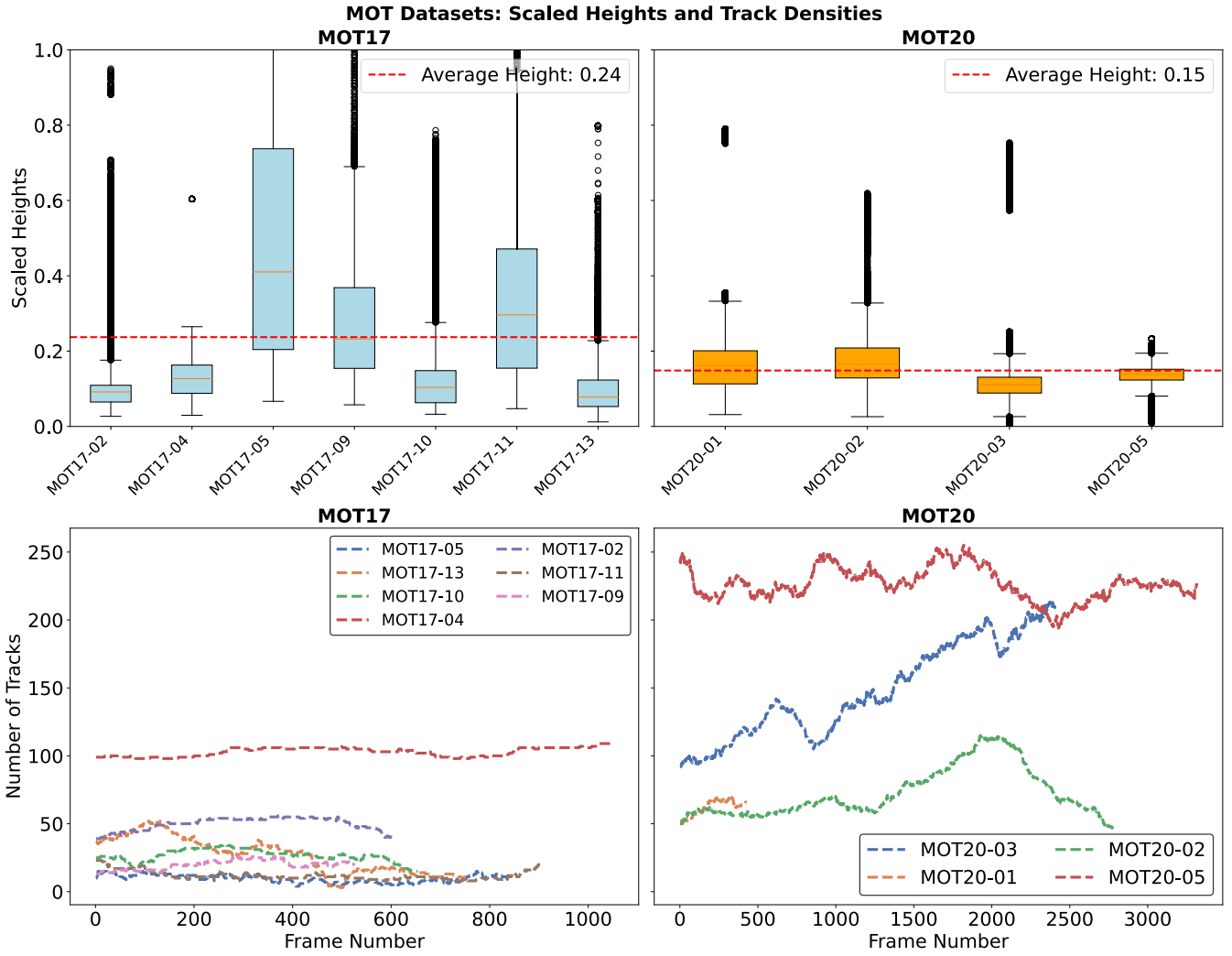


FIGURE 6. Height scale ratio distribution for the MOT17 and MOT20 datasets, illustrating the variability in detected object heights across videos.

D. EDGE DEVICE PERFORMANCE ON JETSON ORIN NX

This experiment aims to assess the real-time performance of tracking pipelines on edge devices, using a framework designed to closely approximate practical benchmarks. Additionally, we compare the speed of ReID-based trackers with the benchmark LITE performance. While OC-SORT and ByteTrack achieve higher HOTA scores than ReID-based trackers (e.g., Deep OC-SORT, BoTSORT, StrongSORT), this paper's primary objective is not to establish the superiority of ReID trackers. Instead, our focus is on benchmarking LITE, an efficient tracker with a ReID component, against other ReID-based trackers to assess its suitability for real-time applications.

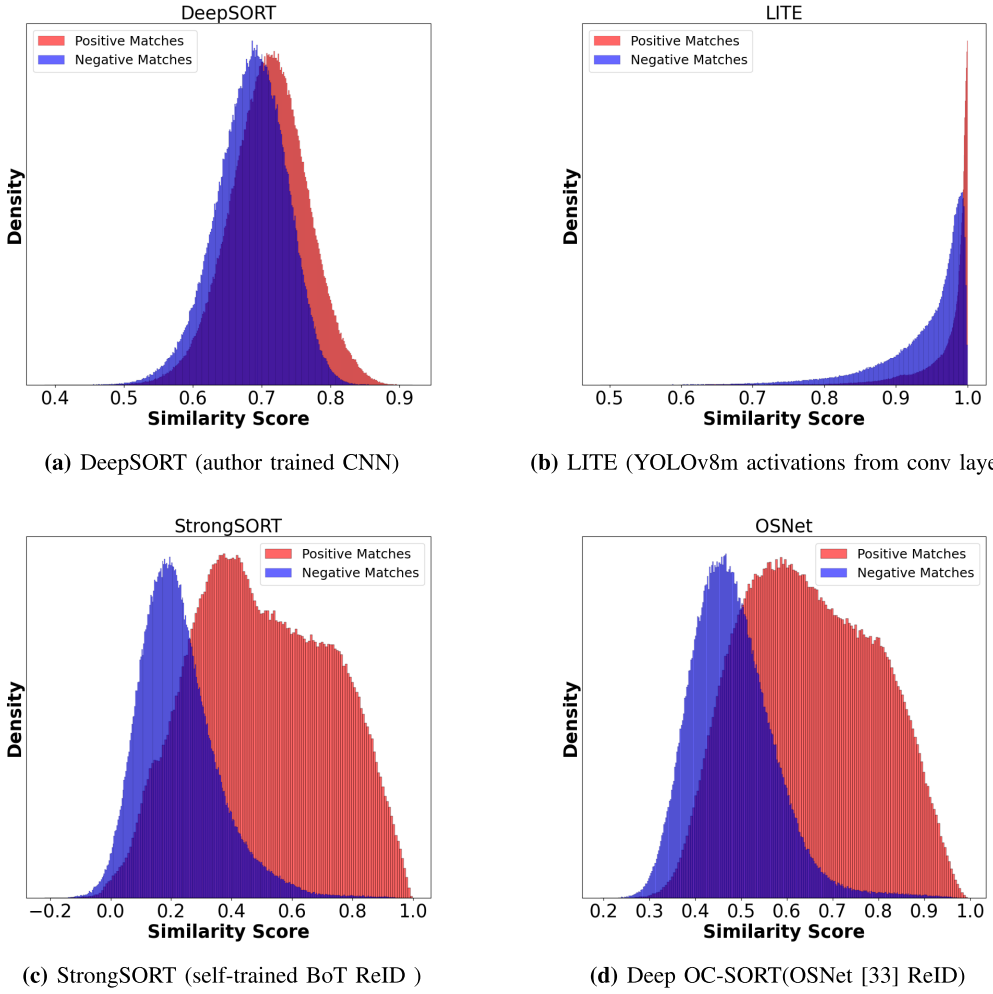
The table 5 presents the FPS reduction for each tracker when deployed on the edge device. Notably, LITE-based trackers sustain FPS levels comparable to those of purely motion-based trackers, even with the added ReID capability, whereas other ReID-based trackers (e.g., Deep OC-SORT and BoTSORT) exhibit at least a 40% reduction in FPS.

Fig. 9 illustrates the impact of detector settings, such as confidence thresholds and input image resolutions, on the performance of the Orin device.

VI. DISCUSSION

This study introduces an advanced evaluation framework that provides critical insights into the multi-object tracking (MOT) domain. It particularly highlights the significant impact of detector settings on tracker performance. Our findings challenge the notion of universally optimal detector confidence thresholds and image resolutions. Instead, we demonstrate that optimal configurations are highly contextual, depending on the specific characteristics of datasets and benchmarks. This understanding emphasizes the complexity and variability inherent in MOT systems, underscoring the necessity for flexible and comprehensive evaluation methods.

Our experiments reveal that variations in detector parameters, such as confidence thresholds and input resolutions,



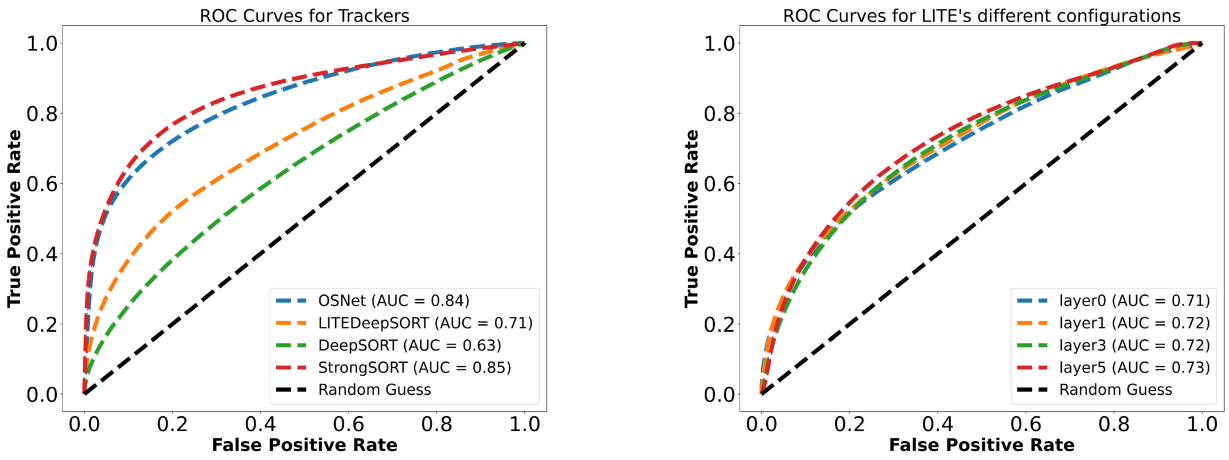
(a) DeepSORT (author trained CNN)

(b) LITE (YOLOv8m activations from conv layer0)

(c) StrongSORT (self-trained BoT ReID)

(d) Deep OC-SORT(OSNet [33] ReID)

FIGURE 7. **Cosine similarity distribution in different ReID components.**



(a) ROC curves for different ReID components.

(b) ROC curves for LITE's different configurations.

FIGURE 8. **a. ReID modules such as OSNet, LITE, DeepSORT, and StrongSORT show AUC values for appearance feature matching. b. Strengths of LITE's different configurations, which correspond to different YOLOv8m architecture's convolutional layers, are depicted in Fig. 7.**

can significantly alter tracker rankings and associated performance metrics. For instance, a confidence threshold that optimizes performance on one dataset may be suboptimal for another due to differences in object scales and

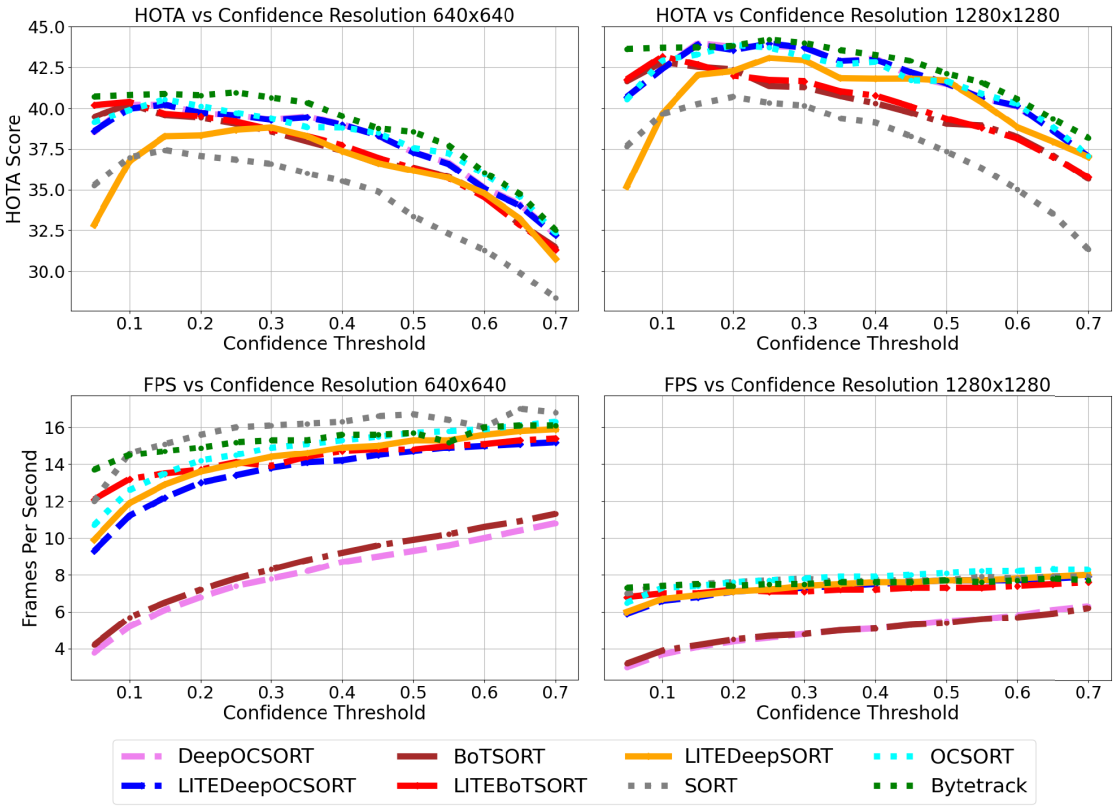


FIGURE 9. Comparative analysis of detection settings' impact on HOTA scores and FPS across different input resolutions (640 × 640 and 1280 × 1280) on MOT17. Experiments conducted on the NVIDIA Jetson Orin NX.

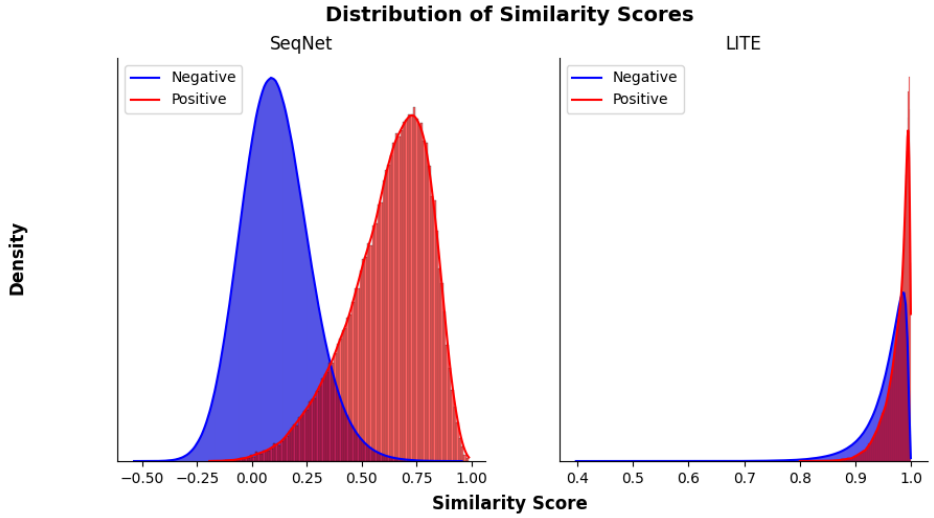


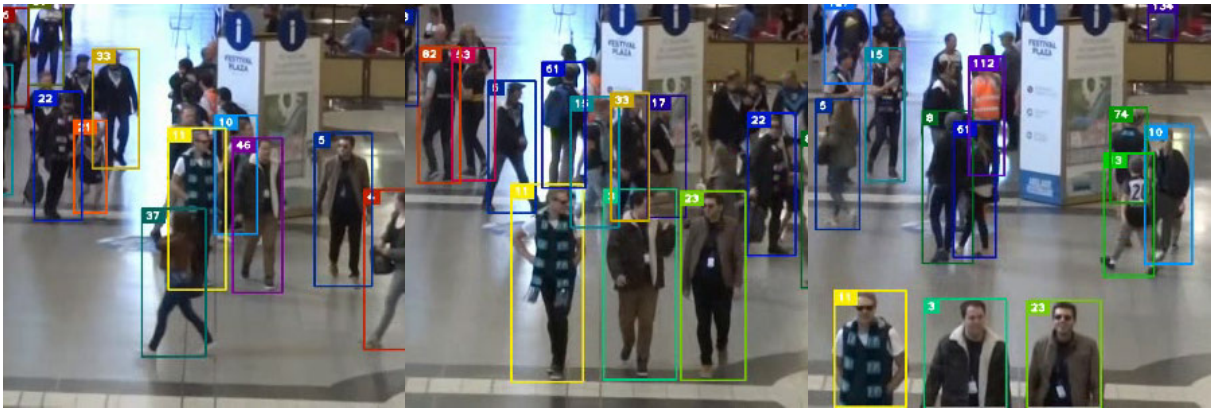
FIGURE 10. Limitation of LITE for person search tasks. SeqNet [35] is a tailored network trained specifically for the person search task.

scene densities. These results underscore the importance of adopting context-sensitive evaluation strategies in tracking research and development.

The proposed evaluation framework offers a practical and robust tool for assessing trackers' generalizability and real-time capabilities across diverse conditions. By incorporating video sequences with varying levels of occlusion and object scale, the framework provides a more accurate measure of a tracker's robustness and adaptability. Notably, our findings demonstrate that minor tweaks to detector settings can lead to substantial differences in the performance rankings of tracking algorithms. This variability highlights

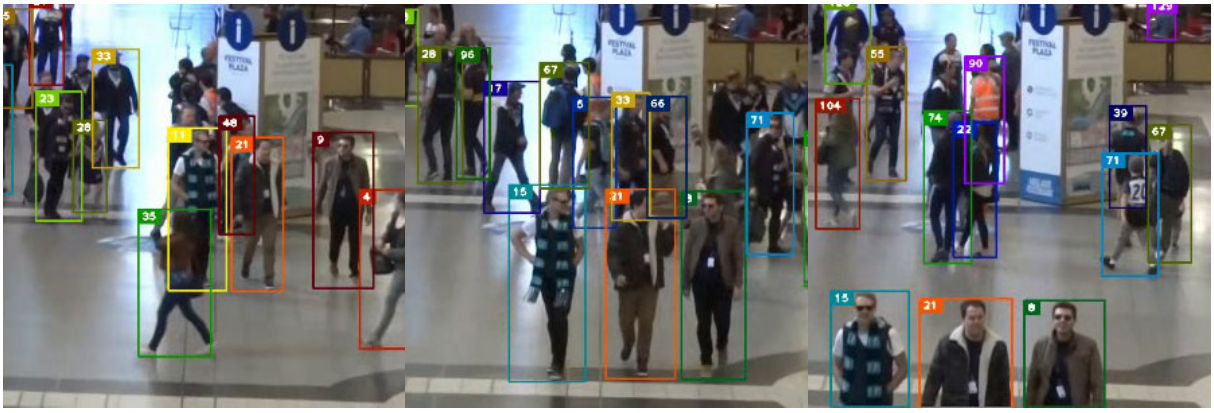
Comparative Analysis of Tracking Methods on MOT20-01

DeepSORT: Few ID switches noted from frame 100 to 200



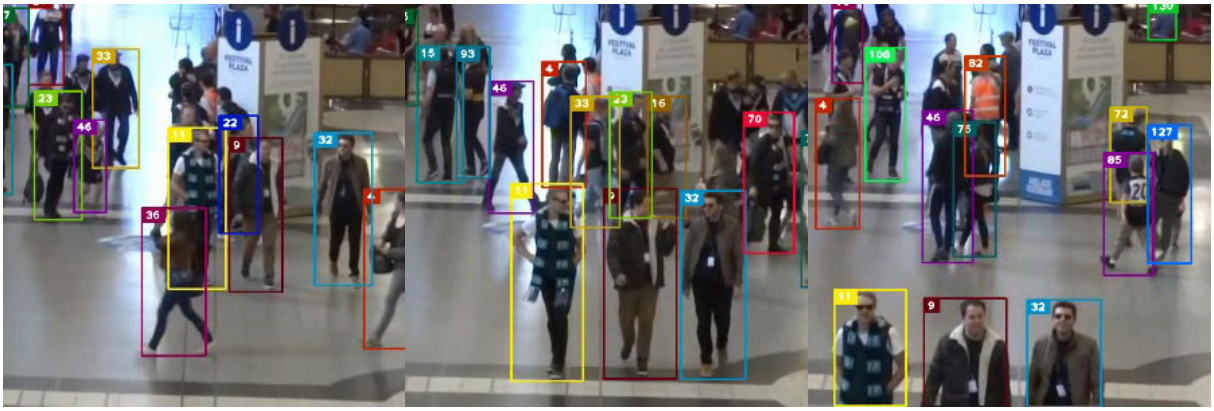
(a) Frame 100 **(b)** Frame 200 **(c)** Frame 300

StrongSORT: Managed tracking with few ID switches



(d) Frame 100 **(e)** Frame 200 **(f)** Frame 300

LITE:DeepSORT: Consistently tracked without ID switches



(g) Frame 100 **(h)** Frame 200 **(i)** Frame 300

FIGURE 11. **In this scenario, three men walking from the center of the frame are tracked from frame 100 to frame 300. The sequence tests each tracker's ability to maintain consistent identification amidst movement and occlusions.**

the limitations of using fixed benchmark datasets and settings, which often fail to capture the full range of real-world complexities.

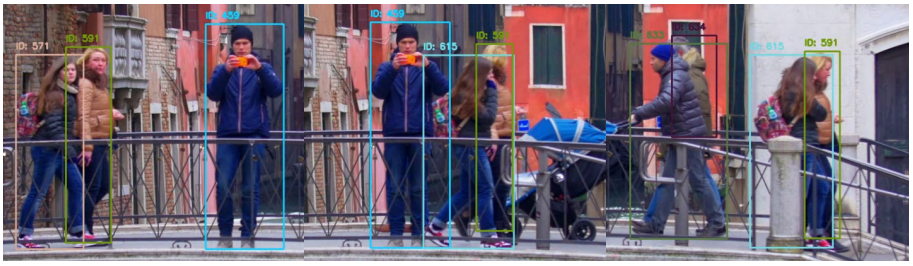
Furthermore, our results indicate that motion-based components often outperform appearance-based ReID features in maintaining track consistency, particularly under the

Comparative Analysis of Tracking Methods on PersonPath22 uid_vid_00008

OC-SORT

(a) Frame 500 **(b)** Frame 540 **(c)** Frame 575

BoTSORT



(d) Frame 500 **(e)** Frame 540 **(f)** Frame 575

FIGURE 12. **Comparison of side-view frames under heavy occlusion: Frame 540 shows two women partially occluding a man with a camera, while in Frame 575, the occlusion increases as more individuals enter the scene. Results highlight the differences between ReID-based BoTSORT and motion-based OC-SORT tracking.**

Comparative Analysis of Tracking Methods on PersonPath22 uid_vid_00019

OC-SORT

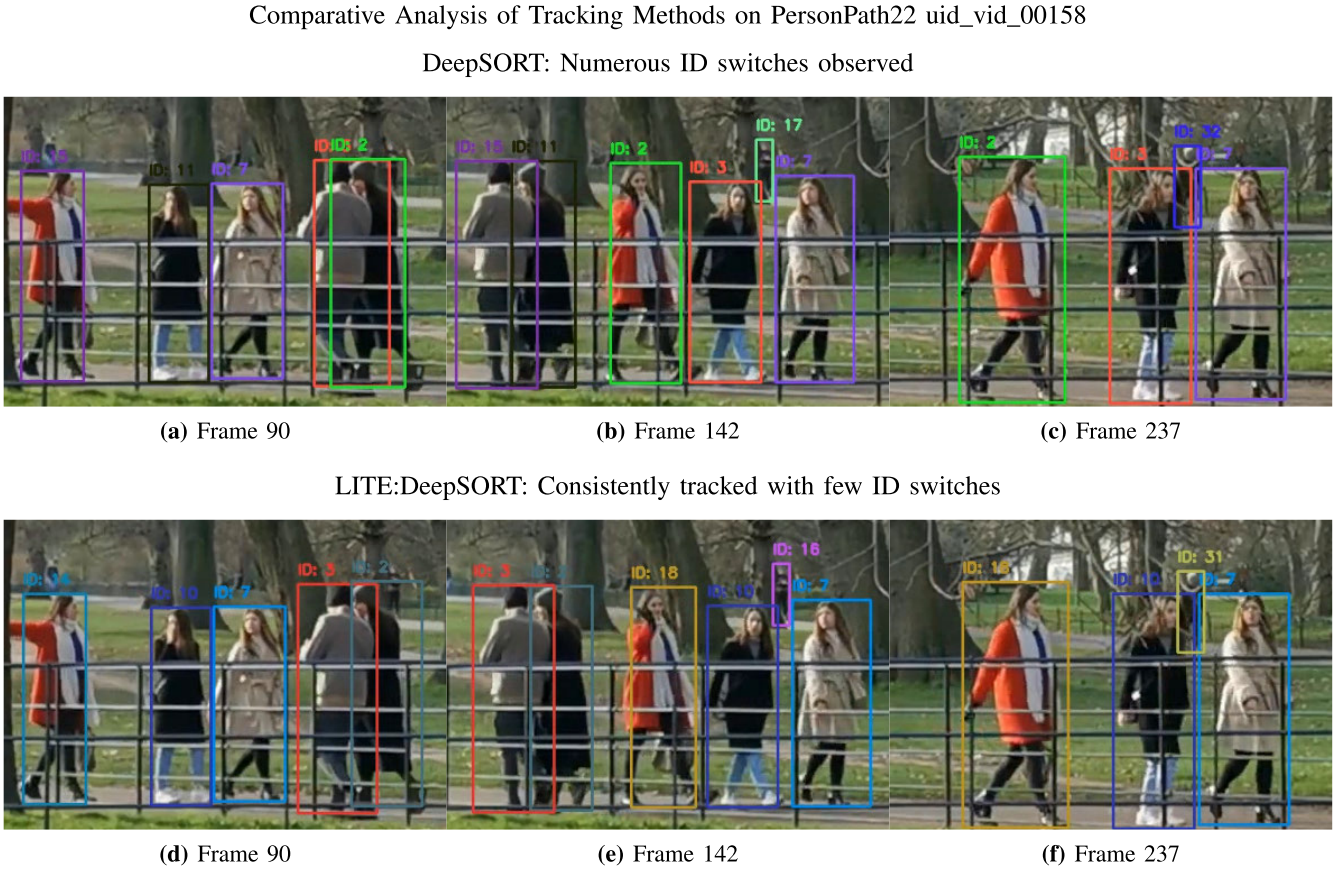
(a) Frame 172 **(b)** Frame 245 **(c)** Frame 335

BoTSORT

(d) Frame 172 **(e)** Frame 245 **(f)** Frame 335

FIGURE 13. **Top-view sequence illustrating the tracking differences between the motion-based OC-SORT and the ReID-enabled BoTSORT.**

practical constraints of edge devices. At the same time, our findings suggest that the matching strategies employed by ReID-enabled trackers may not fully exploit the strengths of appearance features. These observations point to a promising



(a) Frame 90 **(b)** Frame 142 **(c)** Frame 237

(d) Frame 90 **(e)** Frame 142 **(f)** Frame 237

FIGURE 14. The sequence tests each tracker's ability to maintain consistent identification amidst movement and challenging occlusions, where the three women are occluded by two other persons, posing a significant challenge for detection.

TABLE 5. Performance comparison of trackers on MOT17: NVIDIA RTX 3090 Server vs. Jetson Orin NX.

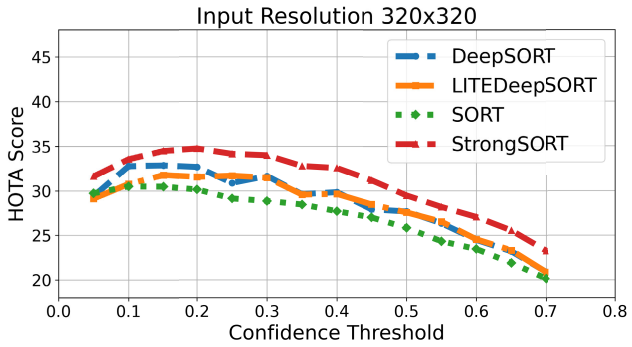
Tracker	NVIDIA RTX 3090		Jetson Orin NX	
	HOTA↑	FPS↑	HOTA↑	FPS↑
Pure Motion				
SORT	40.3	39.7	40.3	7.7
OCSORT	43.9	37.2	43.7	7.7
ByteTrack	43.8	39.4	44.2	7.5
Motion and ReID				
DeepSORT	43.7	10.5	-	-
Deep OC-SORT	43.7	13.0	43.8	4.6
BoTSORT	40.9	24.9	40.6	4.7
StrongSORT	41.7	4.5	-	-
Integrated Motion + ReID				
LITE:DeepSORT	43.0	26.7	43.1	7.2
LITE:Deep OC-SORT	43.7	34.8	44.0	7.2
LITE:BoTSORT	40.8	38.2	40.9	7.1

research direction: the development of more carefully designed matching strategies that can better leverage the strengths of ReID features.

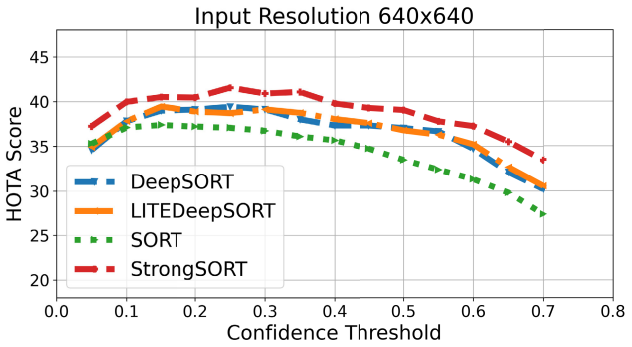
Ultimately, the insights from this study can guide future research in MOT. Future research should focus on creating adaptive, robust tracking solutions that seamlessly adjust to the challenges posed by real-world variability. Such solutions could eliminate the dependency on user-specified settings, such as confidence thresholds, for optimal performance.

Another potential research avenue involves enhancing LITE's discriminatory power by leveraging activation features from multiple layers. These could be optimized using techniques such as PCA [33] or other feature selection algorithms [34]. Such approaches could improve the distinctiveness of appearance features for MOT applications.

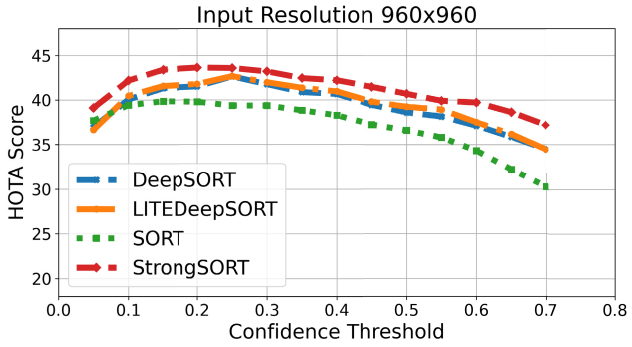
The proposed ReID component evaluation demonstrates LITE's usefulness, particularly compared to dedicated CNN-based ReID models like those in DeepSORT. However, other applications, such as person search, use ReID-capable models more frequently than MOT. LITE's features are extracted directly from the detector without identity-specific training, so its performance in person search is understandably limited. Figure 10 illustrates our findings on this unintended application of LITE. The primary factor contributing to LITE's lower performance in person search is its tendency to generate overly similar embeddings, leading to high similarity scores regardless of whether matches are positive (same person) or negative (different person). This outcome aligns with expectations, as LITE leverages features from YOLOv8m—a model trained exclusively for object detection—resulting in embeddings that capture general visual similarity rather than identity-specific features.



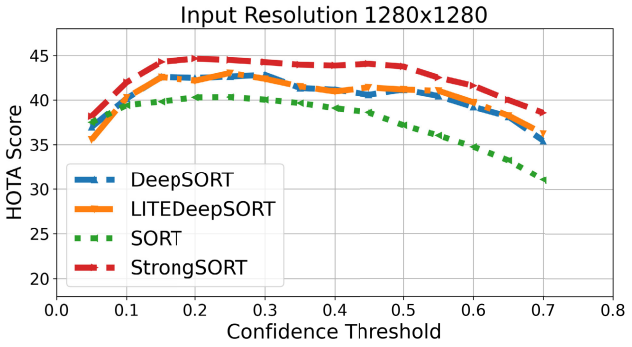
(a) 320x320



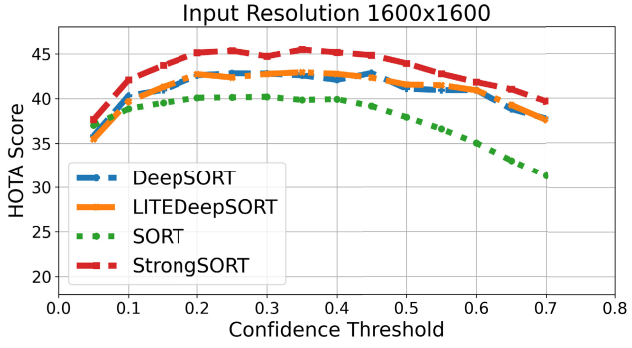
(b) 640x640



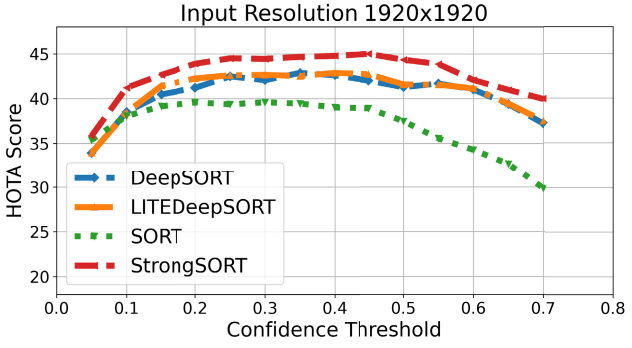
(c) 960x960



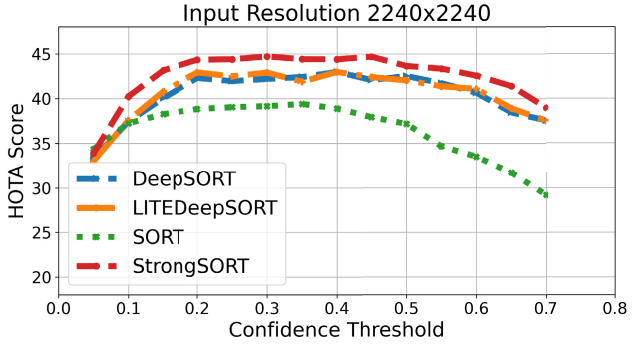
(d) 1280x1280



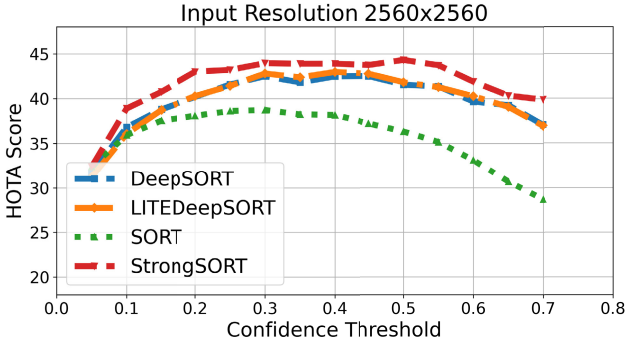
(e) 1600x1600



(f) 1920x1920



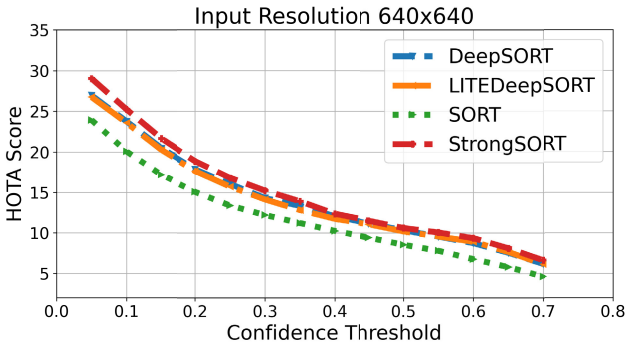
(g) 2240x2240



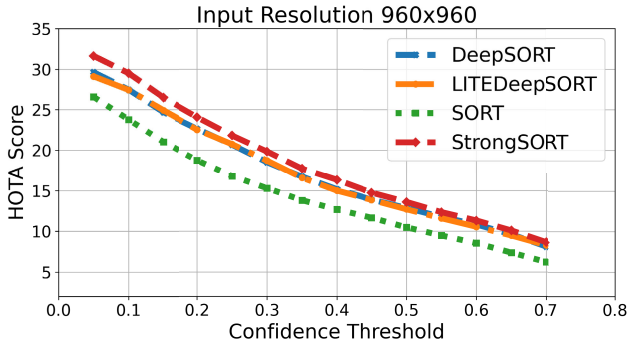
(h) 2560x2560

FIGURE 15. Comparative analysis of detection settings' impact on HOTA scores across different input resolutions. Experiments were conducted on the MOT17 dataset.

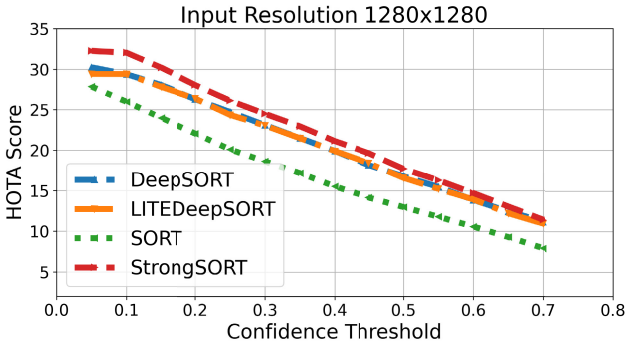
(a) 320x320



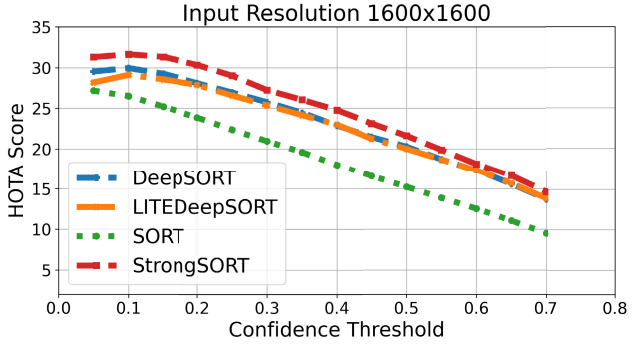
(b) 640x640



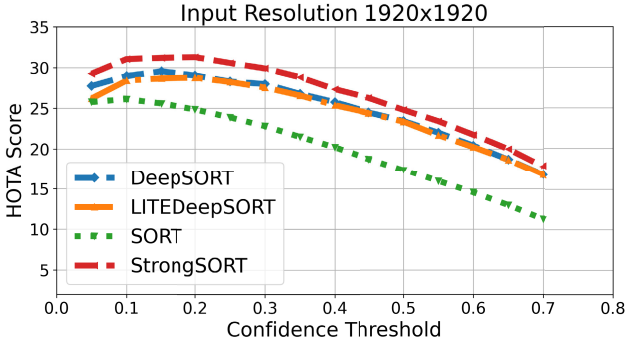
(c) 960x960



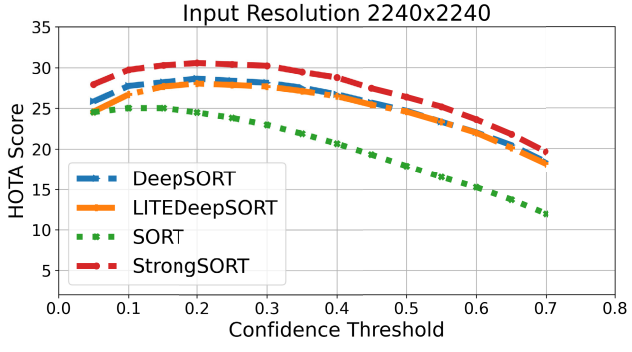
(d) 1280x1280



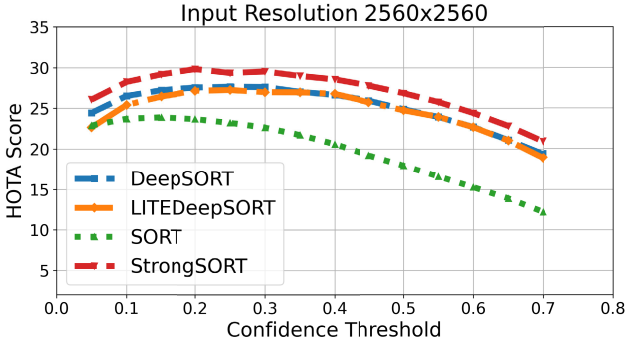
(e) 1600x1600



(f) 1920x1920

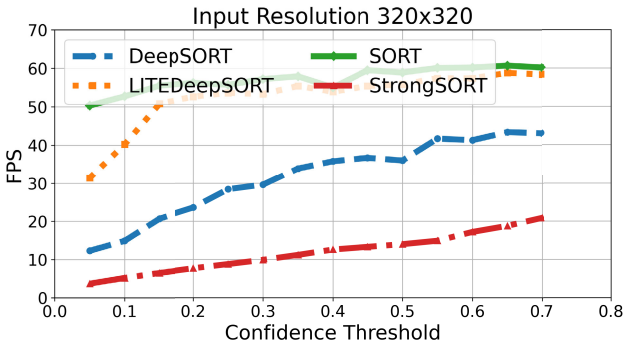


(g) 2240x2240

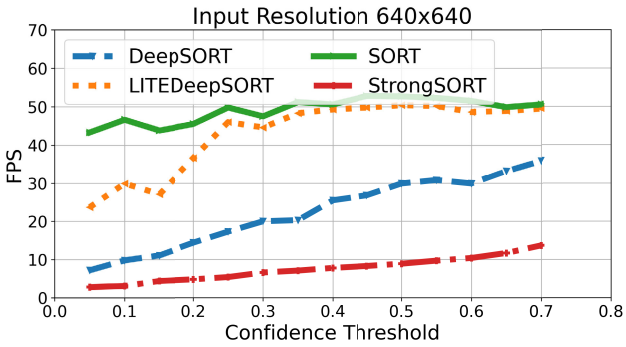


(h) 2560x2560

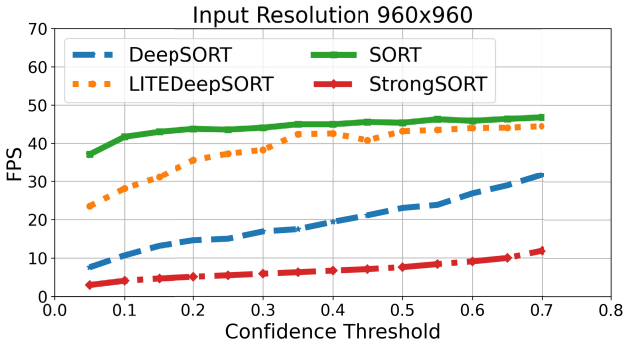
FIGURE 16. **Comparative analysis of detection settings' impact on HOTA scores across different input resolutions. Experiments were conducted on the MOT20 dataset.**



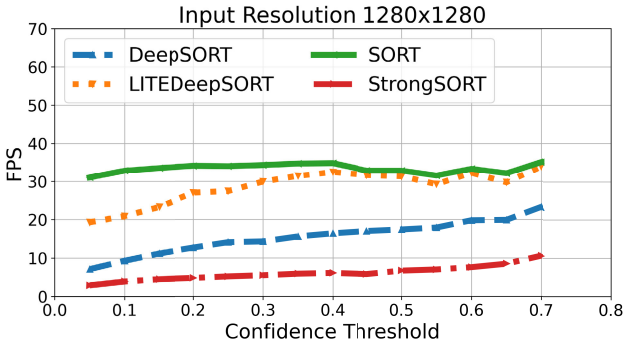
(a) 320x320



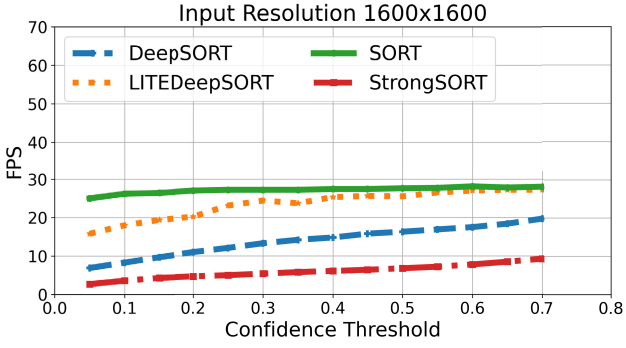
(b) 640x640



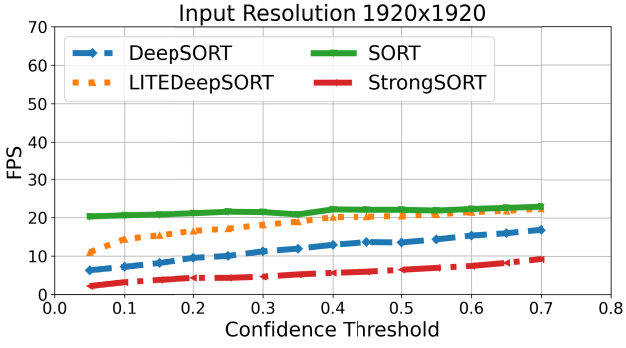
(c) 960x960



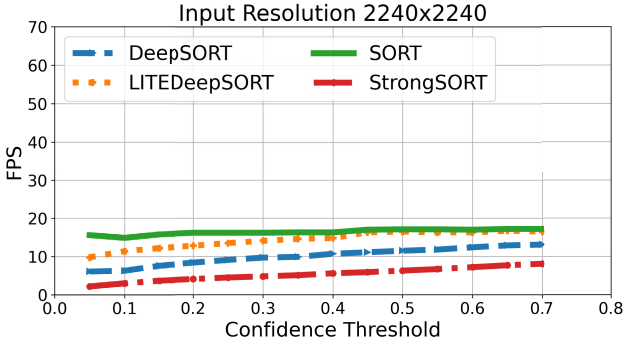
(d) 1280x1280



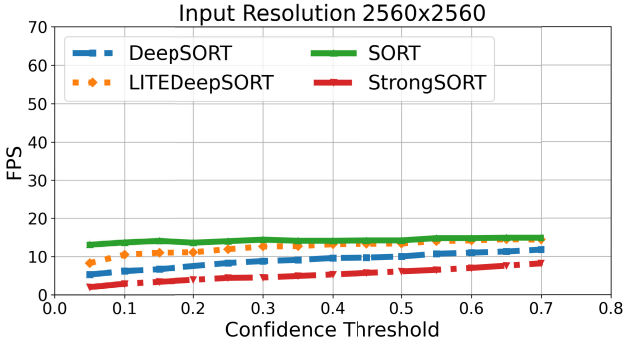
(e) 1600x1600



(f) 1920x1920



(g) 2240x2240



(h) 2560x2560

FIGURE 17. Comparative analysis of detection settings' impact on FPS scores across different input resolutions. Experiments were conducted on the MOT17 dataset.

Interestingly, LITE's poor performance in person search may suggest its suitability for MOT, as crops from the same track in MOT often appear visually similar to one another. This observation further supports LITE's practical relevance to tracking scenarios.

VII. CONCLUSION

This research introduces an advanced evaluation framework for multi-object tracking (MOT) systems, encompassing a holistic approach that integrates detection, re-identification (ReID), and tracking components into the assessment process. The framework highlights the crucial role of optimizing detector settings to significantly enhance performance metrics. Analysis reveals that, through strategic adjustments, HOTA scores can be boosted from 10% to 30%. In other words, these findings demonstrate sharp differences from conventional evaluation frameworks, which can also achieve only about 10% HOTA without careful setting, underscoring the practical importance of the proposed comprehensive approach.

Furthermore, evaluations show that integrating ReID components into MOT systems should be investigated in greater depth. Such a conclusion comes after uncovering the knowledge of where the ReID model's distinctive power differs, but this is not capitalized properly in track matching, leading to similar performance across all tracking methods.

Our findings stress the pressing need for adaptive tracking solutions that eschew static, user-defined parameter settings in favor of dynamic, automated optimization. This adaptability is crucial for accommodating the diverse challenges presented by varying datasets and edge-computing environments. The insights gained from this study pave the way for evolving MOT practices that focus on adaptive mechanisms to achieve robust tracking capabilities.

In conclusion, the presented framework facilitates a more nuanced and standardized evaluation of trackers and serves as a versatile and practical tool for real-world applications. It paves the way for the development of adaptive trackers capable of self-optimization in the future, thus broadening the applicability and efficacy of MOT solutions across diverse operational contexts.

ACKNOWLEDGMENT

The authors acknowledge the use of generative AI tools for syntax verification and content refinement during the preparation of this manuscript, which ensured adherence to high standards of language quality and clarity.

REFERENCES

[1] G. Ciaparrone, F. L. Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, Mar. 2020.

[2] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim, "Multiple object tracking: A literature review," *Artif. Intell.*, vol. 293, Apr. 2021, Art. no. 103448.

[3] K. Huang and Q. Hao, "Joint multi-object detection and tracking with camera-LiDAR fusion for autonomous driving," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 6983–6989.

[4] S. Guo, S. Wang, Z. Yang, L. Wang, H. Zhang, P. Guo, Y. Gao, and J. Guo, "A review of deep learning-based visual multi-object tracking algorithms for autonomous driving," *Appl. Sci.*, vol. 12, no. 21, p. 10741, Oct. 2022.

[5] N. Wojke and A. Bewley, "Deep cosine metric learning for person re-identification," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 748–756.

[6] Y. Du, Z. Zhao, Y. Song, Y. Zhao, F. Su, T. Gong, and H. Meng, "StrongSORT: Make DaeepSORT great again," *IEEE Trans. Multimedia*, vol. 25, pp. 8725–8737, 2023.

[7] J. Alikhanov, D. Obidov, and H. Kim, "LITE: A paradigm shift in multi-object tracking with efficient ReID feature integration," 2024, *arXiv:2409.04187*.

[8] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "ByteTrack: Multi-object tracking by associating every detection box," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Jan. 2022, pp. 1–21.

[9] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-centric SORT: Rethinking SORT for robust multi-object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 9686–9696.

[10] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.

[11] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Zhang, and Y. Wei, "MOTR: End-to-end multiple-object tracking with transformer," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Jan. 2022, pp. 659–675.

[12] P. Azevedo and V. Santos, "Comparative analysis of multiple YOLO-based target detectors and trackers for ADAS in edge devices," *Robot. Auto. Syst.*, vol. 171, Jan. 2024, Art. no. 104558.

[13] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "CrowdHuman: A benchmark for detecting human in a crowd," 2018, *arXiv:1805.00123*.

[14] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468.

[15] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani, "Deep OC-SORT: Multi-pedestrian tracking by adaptive re-identification," 2023, *arXiv:2302.11813*.

[16] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "BoT-SORT: Robust associations multi-pedestrian tracking," 2022, *arXiv:2206.14651*.

[17] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the fairness of detection and re-identification in multiple object tracking," *Int. J. Comput. Vis.*, vol. 129, no. 11, pp. 3069–3087, Nov. 2021.

[18] G. Jocher. (2023). *YOLOV8*. Accessed: Nov. 16, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[19] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[20] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*.

[21] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks for action recognition in videos," 2017, *arXiv:1705.02953*.

[22] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé, "MOT20: A benchmark for multi object tracking in crowded scenes," 2020, *arXiv:2003.09003*.

[23] M. Broström, "BoxMOT: Pluggable SOTA tracking modules for object detection, segmentation and pose estimation models (Version 11.0.9)," [Online]. Available: https://zenodo.org/record/7629840

[24] J. Alikhanov and H. Kim, "Online action detection in surveillance scenarios: A comprehensive review and comparative study of state-of-the-art multi-object tracking methods," *IEEE Access*, vol. 11, pp. 68079–68092, 2023.

[25] B. Shuai, A. Bergamo, U. Buechler, A. Berneshawi, A. Boden, and J. Tighe, "Large scale real-world multi person tracking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 504–521.

[26] Y. Du, Z. Zhao, Y. Song, Y. Zhao, F. Su, T. Gong, and H. Meng. (2023). *Strongsort: Make Deepsort Great Again (code Repository)*. Accessed: Jul. 20, 2024. [Online]. Available: https://github.com/dyhBUPT/StrongSORT

[27] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, pp. 1–10, Dec. 2008.

[28] E. Ristani, F. Solera, R. S. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Jan. 2016, pp. 17–35.
[29] J. Huang and C. X. Ling, "Using AUC and accuracy in evaluating learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 3, pp. 299–310, Mar. 2005.
[30] J. Alikhanov, P. Zhang, Y. Noh, and H. Kim, "Design of contextual filtered features for better smartphone-user receptivity prediction," *IEEE Internet Things J.*, vol. 11, no. 7, pp. 11707–11722, Apr. 2024.
[31] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1116–1124.
[32] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Omni-scale feature learning for person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3701–3711.
[33] B. Byambajav, J. Alikhanov, Y. Fang, S. Ko, and G. S. Jo, "Transfer learning using multiple convnet layers activation features with principal component analysis for image classification," *Korea Intell. Inf. Syst. Soc.*, vol. 24, no. 1, pp. 205–225, 2018.
[34] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Treviño, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, Dec. 2017.
[35] Z. Li and D. Miao, "Sequential end-to-end network for efficient person search," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 2011–2019.

JUMABEK ALIKHANOV received the M.E. degree from Inha University, in 2017, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. His research focuses on advanced machine learning applications in computer vision and data science. Leading a highly skilled team with HUMBLEBEE R&D, he plays a key role in promoting innovative AI solutions across various industries, highlighting HUMBLEBEE R&D's strong presence in the AI field.

DILSHOD OBIDOV is currently pursuing the B.Sc. degree in integrated systems engineering with Inha University, South Korea. He is an Intern with HUMBLEBEE R&D. Focusing on computer vision and edge computing, he works on improving the performance of machine learning models for real-time applications. At HUMBLEBEE R&D, his innovative approach helps reinforce the company's reputation as a stronghold of AI expertise.

MIRSAID ABDURASULOV is currently pursuing the bachelor's degree in integrated systems engineering with Inha University. He is an Intern with HUMBLEBEE R&D. He contributes to HUMBLEBEE R&D's reputation by leveraging data-driven solutions to address complex challenges, showcasing the organization's strength in AI talent. His research interests include machine learning, deep learning, and computer vision, with practical experience in developing models for real-world applications.

HAKIL KIM (Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical and computer engineering from Purdue University, in 1985 and 1990, respectively. In 1990, he joined the College of Engineering, Inha University, Incheon, South Korea, where he is currently a Full Professor with the Department of Information and Communication Engineering. His research interests include biometrics and intelligent video, surveillance, and embedded vision for autonomous vehicles. Since 2003, he has been actively involved as a Project Editor of the International Standardization of Biometrics at ISO/IEC JTC1/SC37.