**Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control**
**Journal homepage: http://kinetik.umm.ac.id**
**ISSN: 2503-2267**
**Vol. 11, No. 2, May, Pp. 403-414**


# LITE-BoostTrack: A hybrid real-time multi-object tracking architecture for resource-constrained environments

**Ruri Suko Basuki[1,2], Adhitya Nugraha[*,1,2], Ardytha Luthfiarta[1,2], Ika Novita Dewi[1,2], Allifian Ilham Febriyana[2], Michael Surya Adi Prasaja[2], Dzawil Uqul[2]**
Research Center for Intelligent Distributed Surveillance and Security (IDSS), Indonesia[1]
Faculty of Computer Science, Universitas Dian Nuswantoro, Indonesia[2]

**Article Info**
**Keywords:**
Multi-Object Tracking (MOT), BoostTrack, LITE Architecture, Real-time Tracking, Edge Computing

**Article history:**
Received: August 26, 2025
Accepted: January 13, 2026
Published: May 01, 2026

**Cite:**
R. S. Basuki, "LITE-BoostTrack: A Hybrid Real-Time Multi-Object Tracking Architecture for Resource-Constrained Environments", *KINETIK*, vol. 11, no. 2, May 2026.
https://doi.org/10.22219/kinetik.v11i2.2478

*Corresponding author.
Adhitya Nugraha
E-mail address:
adhitya@dsn.dinus.ac.id

**Abstract**
*Multi-object tracking (MOT) is a fundamental task in computer vision that underpins applications such as intelligent surveillance, autonomous driving, and crowd analysis. The primary challenge in MOT lies in maintaining identity consistency under frequent occlusions while ensuring real-time performance on resource-constrained devices. This study proposes LITE-BoostTrack, a hybrid tracking framework that combines the confidence-based association mechanism of BoostTrack with the lightweight embedding strategy of the Lightweight Integrated Tracking and Embedding (LITE) architecture. The proposed model extracts appearance descriptors directly from the internal feature maps of the YOLOv8 detector, thereby eliminating the need for an external re-identification network. This design significantly reduces computational complexity while preserving reliable identity association. Experiments were conducted on the MOT20 benchmark using standard MOT evaluation metrics, including HOTA, MOTA, IDF1, IDSW, and FPS, to assess both tracking accuracy and runtime efficiency. The results show that LITE-BoostTrack achieves a HOTA of 27.31 and IDF1 of 37.48, outperforming LITE-BoT-SORT (HOTA 25.73, IDF1 33.88), while reducing identity switches by 37% (2,939 vs. 4,674) and maintaining real-time performance at 13.22 FPS. These outcomes demonstrate that substantial efficiency gains can be achieved through detector-level feature integration without introducing additional deep embedding modules. Although occasional failures still occur under severe occlusion, LITE-BoostTrack provides a balanced and practical solution that effectively combines accuracy and efficiency for real-time multi-object tracking in edge-computing and embedded vision systems.*

## 1. Introduction

Multi-object tracking (MOT) is a crucial component of many computer vision applications, including intelligent surveillance systems, autonomous vehicles [1][2], smart city analytics [3], and crowd monitoring [4][5]. The main objective of MOT is to detect and track multiple objects consistently across video frames while maintaining their unique identities. The major challenge lies in achieving accurate real-time tracking in dense crowds with frequent occlusions, dynamic motion, and significant appearance variations [6][7][8][9]. Most modern MOT systems adopt a tracking-by-detection approach, which separates object detection from identity association. Although this approach offers flexibility, many high-performance trackers still rely on deep learning–based appearance embedding modules. Such dependence increases computational load and limits real-time deployment on devices with restricted processing resources [5][10][11][12][13][14].

The performance of MOT algorithms is typically evaluated using standard benchmarks provided by the MOTChallenge community. Among these datasets, MOT20 [15] is specifically designed to test tracking systems in dense crowd scenarios with heavy occlusion. In response to the challenges represented by MOT20, several tracking architectures have been developed that apply different strategies to improve accuracy and robustness.

The MOT20 dataset itself represents real-world surveillance environments, capturing crowded pedestrian scenes recorded in public areas using fixed IoT-based cameras. These sequences exhibit dense occlusion, varying illumination, and complex motion dynamics, which closely resemble real deployment conditions in autonomous driving systems, UAV-based crowd monitoring, and smart city surveillance networks. Similar real-world use cases have been explored in recent studies on vehicle-mounted tracking systems [1], pedestrian detection for autonomous vehicles [2], and attention-based smart city surveillance frameworks [3], confirming that the MOT20 benchmark effectively reflects the operational challenges faced by practical multi-object tracking systems.

One of the early milestones in multi-object tracking was SORT (Simple Online and Real-time Tracking) [16]. It provided a lightweight and efficient tracker by combining a Kalman filter for position prediction with Intersection-over-Union (IoU) for data association. Its main strength was high processing speed, which made it suitable for real-time applications. However, its accuracy decreased sharply in crowded scenes because it lacked appearance modeling, leading to frequent identity switches.

To address this limitation, DeepSORT [17] introduced a convolutional neural network-based appearance embedding module. The inclusion of visual features improved identity preservation under occlusion and positional variation. This gain in robustness, however, came with a high computational cost, which limited its efficiency on devices with restricted resources.

The next refinement, StrongSORT [18], extended the DeepSORT framework by adding Gaussian process interpolation to enhance trajectory prediction when objects temporarily disappeared. It also applied an independent visual linking mechanism to improve long-term associations. These changes stabilized the tracking process but still depended heavily on deep visual embeddings, which may fail under heavy occlusion or drastic appearance changes.

In parallel, some studies focused more on motion modeling. OC-SORT (Occlusion-aware SORT) improved the original SORT by refining motion estimation, making it more resilient to short-term occlusion. Deep OC-SORT [19] further combined adaptive appearance embedding, context-aware updates, and Camera Motion Compensation (CMC) to enhance robustness, especially when the camera was moving.

Another line of work, BoT-SORT [20], proposed a richer association strategy by combining Re-ID embeddings, IoU distance, and Mahalanobis distance, while also applying CMC to correct global shifts caused by camera motion. This method achieved higher accuracy in dense and dynamic scenes, as shown on the MOT20 benchmark. Despite these advances, most trackers still depend on deep Re-ID modules that impose substantial computational costs. This limitation highlights the need for lighter yet accurate alternatives such as BoostTrack, which aims to balance efficiency and robustness in real-world tracking scenarios.

To overcome these limitations, BoostTrack [21] was introduced as a tracking-by-detection system built upon the SORT framework with additional lightweight plug-and-play modules. Unlike previous methods that rely heavily on deep visual embeddings, BoostTrack improves the reliability of similarity measures and detection confidence without adding significant computational overhead. Two key mechanisms, illustrated in Figure 1, form the core of this design: (1) a confidence-based adjustment of association weights, and (2) the integration of supplementary metrics such as Mahalanobis distance and shape similarity to reduce ambiguity in crowded or noisy environments. These strategies allow BoostTrack to minimize identity switches while maintaining real-time processing speed, achieving competitive performance on standard MOT benchmarks, including challenging datasets such as MOT20.

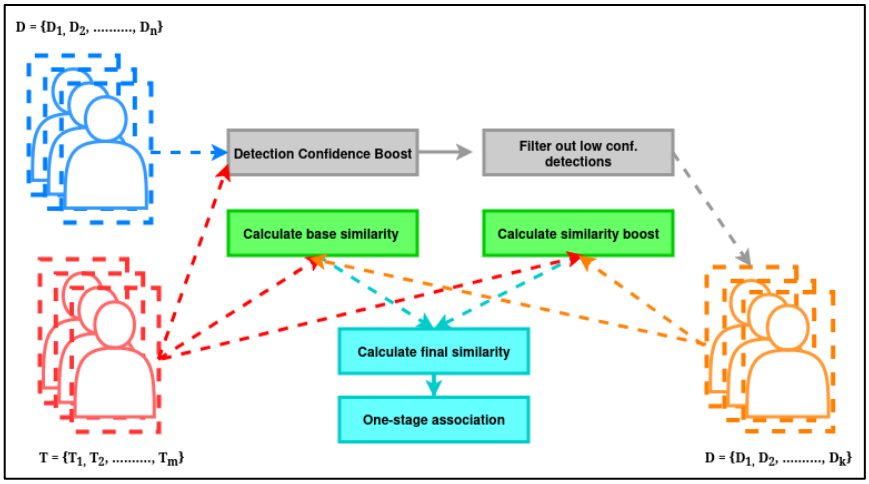



*Figure 1. Workflow of the BoostTrack Algorithm [20]*

While BoostTrack offers high computational efficiency, its design remains modular, with detection and association executed as separate stages. As a result, it has not fully exploited the potential of integrating visual features directly from the detector. Moreover, despite its competitive benchmark results, BoostTrack has not yet been evaluated in resource-constrained edge environments.

Recent research in lightweight multi-object tracking (MOT) has focused on balancing accuracy and efficiency by simplifying feature extraction and association modules. Ye et al. [9] proposed a Lightweight Deep Appearance Embedding (LDAE) model using compact convolutional layers to replace heavy Re-ID networks, while Wan et al. [21] employed a transformer-based lightweight framework with Swin-T to enhance contextual representation at a reduced

parameter cost. Li et al. [22] introduced FDBTrack, combining a fast OSNet backbone with hierarchical exponential moving averaging to stabilize feature updates, and Karthikeyan et al. [23] developed LightMOT, an anchor-free MobileNet-based design achieving real-time inference on dense-crowd datasets. These studies typically evaluate performance on publicly available MOT17 or MOT20 training subsets, following the common protocol for reproducible lightweight MOT benchmarking.

Although these approaches demonstrate substantial efficiency gains, they still encounter difficulties in maintaining long-term identity consistency under severe occlusion and high crowd density. LDAE and LightMOT, for instance, rely on shallow convolutional features that limit discriminative power, while transformer-based methods such as FDBTrack improve context modeling at the expense of increased computational cost.

To alleviate these limitations, recent works have sought to further reduce dependency on external embedding modules. One such development is the Lightweight Integrated Tracking and Embedding (LITE) framework [24], which replaces standalone appearance extractors with internal features from real-time detectors such as YOLOv8. This architecture simplifies the tracking pipeline, significantly increases inference speed, and represents a shift from accuracy maximization toward deployable real-time MOT solutions, as illustrated in Figure 2.

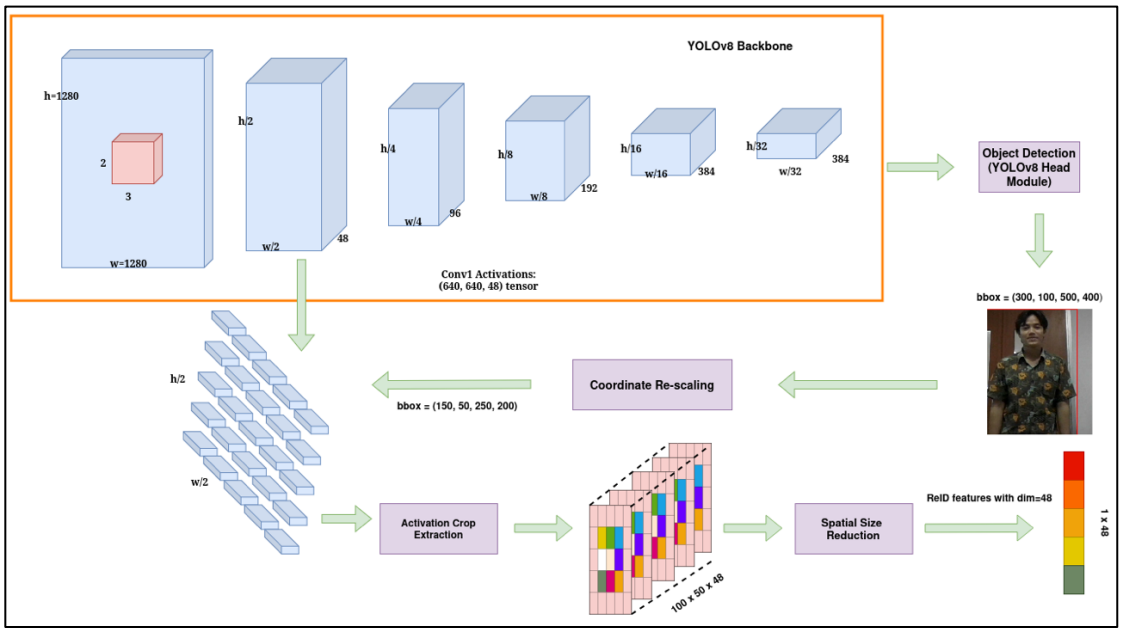



*Figure 2. Efficient Re-ID Feature Extraction Via the LITE Paradigm [24]*

The LITE framework has been successfully integrated into established trackers such as DeepSORT and BoT-SORT [6]. In LITE-DeepSORT, removing the external Re-ID module nearly doubles processing speed while maintaining competitive accuracy, whereas LITE-BoT-SORT achieves faster inference without sacrificing tracking stability in occluded scenes. These findings confirm that internal detector features can serve as effective embeddings for online identity association, enabling lightweight and efficient tracking on limited hardware.

However, despite its remarkable speed advantage, the original LITE framework still relies on shallow intermediate features from early convolutional layers, which limits its ability to preserve identity consistency in highly crowded MOT20 scenes [25]. Meanwhile, heavy hybrid trackers such as BoT-SORT and BoostTrack achieve strong results with HOTA scores around 50 but operate at only 5–6 FPS due to complex appearance extraction and motion refinement modules. This contrast reveals a persistent gap between highly accurate but computationally expensive methods and lightweight yet less robust designs.

To bridge this gap, this study introduces LITE-BoostTrack, a hybrid tracking architecture that integrates LITE's efficient embedding extraction with BoostTrack's motion-boosting and adaptive association mechanisms. By combining the efficiency of LITE with the robustness of BoostTrack, the proposed model achieves real-time multi-object tracking while maintaining strong identity consistency under crowded and occluded conditions. Its effectiveness is validated on the MOT20 benchmark using standard MOT evaluation metrics, including HOTA, MOTA, IDF1, IDSW, and FPS. The main contribution of this work lies in demonstrating that substantial runtime improvements can be achieved without compromising tracking accuracy, thereby enabling practical deployment in resource-constrained, real-time environments.

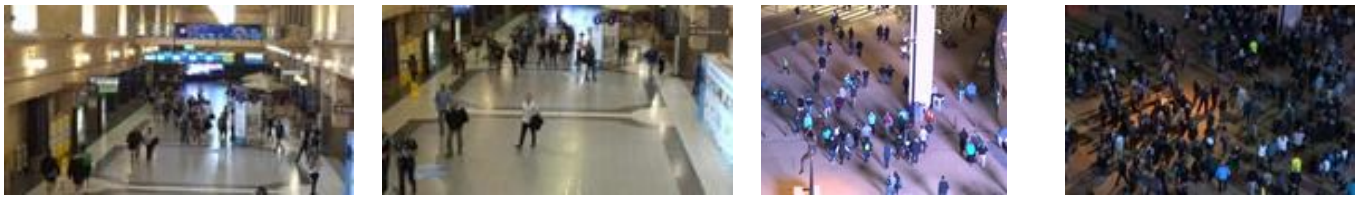
## 2. Research Method

This study aims to develop a real-time multi-object tracking system based on the BoostTrack framework, optimized for operation on resource-constrained devices. To achieve this objective, the research process is divided into three main stages: (1) integrating the BoostTrack architecture within the LITE evaluation framework, (2) defining the experimental configuration, and (3) evaluating performance using the MOT20 benchmark dataset. In addition, comparative experiments are conducted with several baseline trackers to assess both tracking accuracy and computational efficiency.

### 2.1 Dataset

The experiments in this study used the MOT20 dataset, a well-known public benchmark for multi-object tracking in crowded scenes. The dataset contains eight full-HD (1920 × 1080) video sequences that capture dense pedestrian environments with frequent occlusion and heavy crowd movement. These conditions make MOT20 a realistic and challenging benchmark for evaluating real-time tracking systems on resource-limited devices.

Among the eight MOT20 sequences, four (MOT20-01, MOT20-02, MOT20-03, and MOT20-05) include public ground-truth annotations and are therefore used in this work. The remaining four sequences belong to the official test set, whose annotations are not publicly released and can be accessed only through submission to the MOTChallenge evaluation server. Following the standard experimental protocol adopted in recent multi-object tracking research, studies such as Alikhanov et al. [6], [25] have also relied solely on the MOT20 training subset due to the unavailability of test annotations. For this reason, all evaluations in this study are conducted on the training subset. Figure 3 shows examples from these four training sequences, illustrating the diversity of crowd density, lighting conditions, and motion patterns captured in MOT20.



(a) (b) (c) (d)

*Figure 3. Data Training MOT20*

The selected subset consists of approximately 13,400 annotated frames, with an average of nearly 100 tracked objects per frame, reflecting a very high crowd density. These sequences capture diverse conditions in illumination, occlusion, and motion, providing sufficient variability for reliable evaluation. Preliminary inspection of the hidden test set (based on public MOT statistics) indicates a comparable level of density and scene complexity, suggesting that the chosen subset remains representative for performance analysis.

### 2.2 Baseline Architecture

This study adopts BoostTrack as the baseline tracking framework, which integrates one-stage association with confidence management to improve identity preservation. To evaluate the effectiveness of the proposed LITE-BoostTrack model, several widely used baseline trackers are selected for comparison, including DeepSORT, StrongSORT, BoT-SORT, and Deep OC-SORT. These models employ different strategies for object association, ranging from appearance-based embedding to motion modeling and global interpolation.

In addition to the original versions, lightweight variants based on the LITE framework are also included in the evaluation, namely LITE-DeepSORT, LITE-StrongSORT, LITE-BoT-SORT, and LITE-Deep OC-SORT. In these models, the computationally expensive deep Re-ID module is removed and replaced with lightweight appearance features directly extracted from the detection backbone. This modification is designed to reduce computational overhead while preserving acceptable levels of tracking accuracy.

### 2.3 LITE Architecture Integration: LITE-BoostTrack

The LITE (Lightweight Integrated Tracking and Embedding) architecture, proposed by Alikhanov et al. [24], was introduced as an efficient alternative to conventional multi-object tracking systems that depend on deep re-identification modules. Traditional tracking pipelines usually perform detection, feature embedding, and association in separate steps, which increases latency and computational cost. LITE simplifies this process by embedding appearance representation directly into the detector, so that features are produced in the same stage as detection. This integration removes the need for a separate network and enables real-time performance on devices with limited computing power.

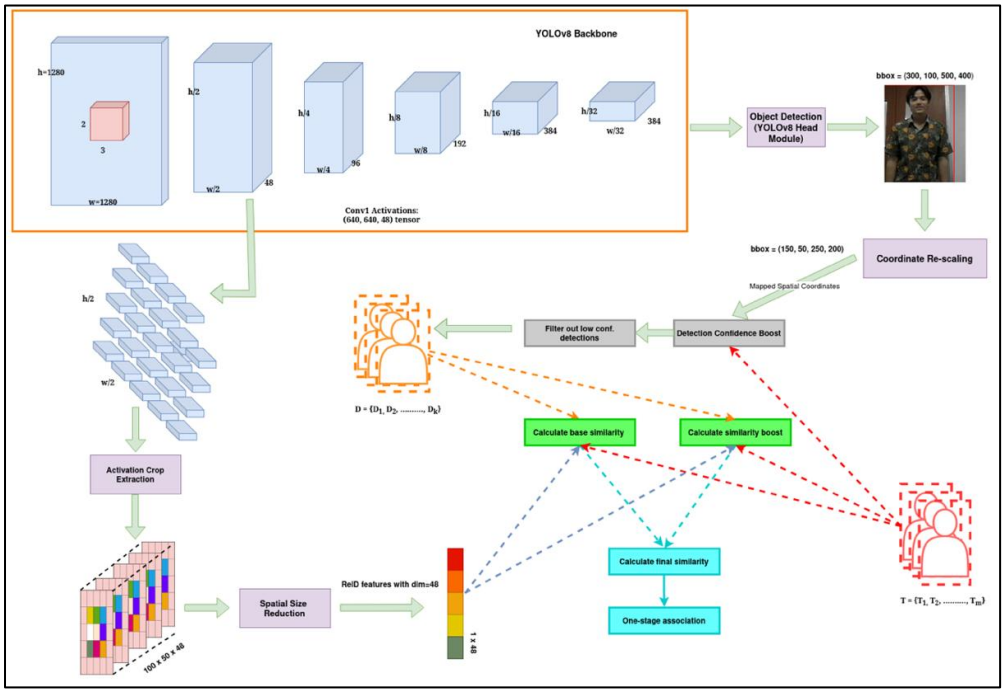



*Figure 4. Integration of Lightweight Embedding in the LITE Architecture*

As illustrated in Figure 4, the proposed LITE-BoostTrack architecture unifies the efficient embedding extraction of the Lightweight Integrated Tracking and Embedding (LITE) paradigm with the motion-boosting and adaptive association modules of BoostTrack. The workflow consists of five core stages: feature extraction, spatial alignment and confidence boosting, multi-metric affinity computation, global assignment optimization (one-stage association), and state updating. Each stage contributes to maintaining identity consistency and computational efficiency during real-time tracking.

**a. Integrated Feature Extraction (LITE Paradigm).**
Derived from the LITE framework, this stage employs the YOLOv8 backbone as a single feature extractor for both detection and identity embedding tasks. As the input image (1280 × 1280) passes through the convolutional layers, the network produces activation tensors at multiple spatial resolutions (e.g., Conv1 with 48 channels). Instead of relying on an external re-identification (Re-ID) network, activation crop extraction is performed directly on these intermediate feature maps. Through spatial size reduction using global average pooling, each crop is converted into a compact 1 × 48-dimensional descriptor representing the appearance identity of each detected object. This zero-cost Re-ID extraction, adopted from the LITE paradigm, eliminates redundant forward passes and ensures high inference efficiency.

**b. Spatial Alignment and Confidence Enhancement.**
This module, adapted from BoostTrack, refines detections using a dual-path mechanism. The YOLOv8 detection head outputs bounding boxes with class probabilities and confidence scores, which are rescaled from image coordinates to the activation map scale to ensure feature alignment. A detection confidence boosting process evaluates spatial consistency between predicted trajectories (from a Kalman filter) and current detections. Detections that are geometrically consistent but have low confidence are selectively boosted and retained, while irrelevant ones are filtered out. This procedure enhances detection reliability and mitigates false negatives in crowded or partially occluded environments.

**c. Multi-Metric Affinity Computation (Hybrid Integration).**
This hybrid stage combines LITE's detector-level embeddings with the multi-metric association strategy of BoostTrack. The base similarity integrates Intersection over Union (IoU) for geometric overlap and cosine similarity from LITE's appearance embeddings. This matrix is further refined through BoostTrack's similarity boosting, which introduces three auxiliary metrics: Mahalanobis distance (motion uncertainty), shape similarity (morphological consistency), and Re-ID fusion (temporal visual coherence). By integrating LITE's efficient visual features into BoostTrack's affinity framework, this stage produces a discriminative and computationally efficient representation suitable for dense-crowd tracking.

**d. Global Assignment Optimization (One-Stage Association).**
Following BoostTrack's design principle, the final cost matrix is optimized through a global one-stage association. Unlike multi-stage methods that separately process motion and appearance cues, this unified procedure jointly resolves all detections and tracklets within a single bipartite matching (Hungarian) optimization step. This global optimization minimizes the overall association cost between detection candidates (*D*) and active tracklets (*T*), ensuring consistent identity linkage even under complex occlusion scenarios.

**e. State Update and Momentum Embedding Refinement.**
The state update mechanism is also adapted from BoostTrack, where matched trajectories are updated and unmatched detections initiate new tracklets. Each active tracklet's embedding is refined using momentum-based feature averaging, blending new and past descriptors to create a stable representation that adapts to illumination changes and partial occlusions. This update strategy reduces identity fragmentation and maintains long-term tracking stability.

Overall, the proposed LITE-BoostTrack architecture inherits the lightweight embedding efficiency of LITE and the robust motion-driven association of BoostTrack, effectively bridging the gap between high-speed and high-accuracy tracking frameworks. The resulting system simplifies the tracking pipeline, removes external Re-ID dependencies, and achieves real-time deployment capabilities even in dense, occluded, and resource-constrained environments.

### 2.4 Experimental Configuration and Procedure

All tracking models in this study were implemented using PyTorch and executed on a workstation with the following hardware specifications: Intel Core i7 processor, 8 GB RAM, and an NVIDIA GeForce RTX 4060 GPU. To ensure experimental consistency, each model was run with its original default configuration as provided by the respective developers. No parameter adjustments were applied, including the detection confidence threshold, maximum track age, or data association threshold. This strategy was adopted to guarantee a fair and unbiased comparison across models without the influence of manual optimization. The LITE system implementation was adapted from the official repository (https://github.com/Jumabek/LITE) and further modified to integrate with multiple multi-object tracking models, including BoostTrack, DeepSORT, StrongSORT, BoT-SORT, and Deep OC-SORT.

### 2.5 Evaluation Metrics

The evaluation process was conducted using the TrackEval toolkit, which provides standard implementations for various multi-object tracking metrics. This toolkit is widely used in the MOT community to ensure objective and reproducible evaluations. Evaluations were conducted under consistent runtime conditions and datasets across models to ensure fair performance comparisons. The metrics used include:

a. Higher Order Tracking Accuracy (HOTA) provides a comprehensive evaluation of tracking performance by integrating two key components: detection accuracy (DetA) and association accuracy (AssA). The metric is computed as the geometric mean of these two components, as defined in Equation 1:

$$HOTA = \sqrt{DetA.AssA} \tag{1}$$

b. Multiple Object Tracking Accuracy (MOTA) measures the overall tracking performance by penalizing three types of errors: false positives (FP), false negatives (FN), and identity switches (IDSW). The formulation of MOTA is presented in Equation 2:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{2}$$

where $GT_t$ denotes the number of ground-truth objects at time $t$.

c. The Identification F1 (IDF1) evaluates the accuracy of identity preservation over time. It is defined as the harmonic mean of identity precision and identity recall, based on identity-level matches, as shown in Equation 3:

d.

$$IDF1 = 1 - \frac{2\,.IDTP}{2.IDTP + IDFP + IDFN} \tag{3}$$

where IDTP, IDFP, and IDFN denote true positive, false positive, and false negative identities, respectively.

e. Identity Switching (IDSW) quantifies the frequency with which tracked objects are reassigned to new identities, indicating fragmentation in identity continuity. The metric is computed as the total number of identity-switching events accumulated over the entire sequence, as expressed in Equation 4:

$$ID\ Sw = \sum_{t} Switches_t \tag{4}$$

f. Frames Per Second (FPS) evaluates the runtime efficiency of a tracking system. It is defined as the total number of frames processed divided by the total processing time in seconds, as formulated in Equation 5:

$$FPS = \frac{Total\ Frames}{Total\ Processing\ Time\ (s)} \tag{5}$$

## 3. Results and Discussion

This chapter presents the evaluation results of the proposed LITE-BoostTrack model using the MOT20 dataset. The experiments were designed to assess two key aspects of performance: tracking accuracy, measured by HOTA, MOTA, IDF1, and IDSW, and computational efficiency, measured by frames per second (FPS). To demonstrate the effectiveness of the proposed approach, the results are compared with several established multi-object tracking baselines, including DeepSORT, StrongSORT, OC-SORT, and BoT-SORT, along with their corresponding LITE variants. In addition, comparative and ablation analyses are conducted to evaluate the specific contribution of integrating the LITE embedding mechanism into the BoostTrack framework. These analyses highlight how the proposed hybrid design balances accuracy and runtime efficiency under dense-crowd conditions.

### 3.1 Experimental Results

Experiments were conducted on four sequences from the MOT20 training subset (MOT20-01, MOT20-02, MOT20-03, and MOT20-05). All models were executed using the original default configurations provided by their respective developers to ensure fair and consistent comparisons. These results serve as the basis for subsequent analyses, both in terms of comparative evaluation among trackers and in assessing the contribution of LITE integration into BoostTrack.

Table 1 presents a comparison of tracking performance among several multi-object tracking models evaluated on the MOT20 training subset. The models include DeepSORT, StrongSORT, OC-SORT, BoT-SORT, and BoostTrack, as well as their respective LITE-based variants.

*Table 1. Performance Comparison of Multi-Object Tracking Models on the MOT20 Dataset*

| Tracker | HOTA | MOTA | IDF1 | IDSW | FPS (AVERAGE) |
|---|---|---|---|---|---|
| Deep SORT | 24,539 | 29,325 | 32,133 | 5129 | 2,925 |
| Strong SORT | 27,375 | 29,457 | 37,421 | 2840 | 3.17 |
| OC-SORT In | 25,375 | 25,777 | 34,287 | 2160 | 4.45 |
| BoT-SORT | 25,745 | 28,467 | 33,896 | 4702 | 5.3 |
| BoostTrack | 27,457 | 28,979 | 37,724 | 2755 | 6.65 |
| LITE-DeepSORT | 25,022 | 29,474 | 32,801 | 4431 | 5,225 |
| LITE-StrongSORT | 25,043 | 29,597 | 32,825 | 4855 | 6.5 |
| LITE-Sort OC In | 25,324 | 25,776 | 34,233 | 2176 | 12,525 |
| LITE-BoT-SORT | 25,734 | 28,476 | 33,881 | 4674 | 15,175 |
| **LITE-BoostTrack (Proposed)** | **27,316** | **28,966** | **37,485** | **2939** | **13,225** |

BoostTrack demonstrates strong performance across accuracy metrics, achieving the highest IDF1 (37.724) and HOTA (27.457) scores among the baseline methods. However, this improvement comes with a moderate computational cost, as its average frame rate is only 6.65 FPS. In contrast, traditional models such as DeepSORT and BoT-SORT perform more slowly while offering slightly lower accuracy. OC-SORT achieves relatively balanced results but remains less stable in dense-crowd sequences, as indicated by a higher number of identity switches (IDSW).

The introduction of the LITE architecture significantly improves runtime efficiency across all trackers. For example, LITE-StrongSORT and LITE-BoT-SORT achieve more than double the frame rates of their original versions while maintaining comparable accuracy. This finding confirms that the lightweight embedding head reduces redundant computations and enhances overall efficiency without compromising association quality.

Among all evaluated variants, the proposed LITE-BoostTrack achieves the most balanced performance. It records an IDF1 score of 37.485 and a HOTA of 27.316, which are only slightly lower than those of the full BoostTrack

model. At the same time, it increases the processing speed from 6.65 FPS to 13.225 FPS, demonstrating a substantial improvement in computational efficiency. The trade-off between accuracy and runtime remains minimal, indicating that the embedded feature extraction mechanism is sufficient to preserve reliable identity matching.

These results show that integrating LITE into BoostTrack successfully resolves one of the main limitations of the original method, namely the computational redundancy caused by the use of an external embedding process. The combined design maintains the strong association capability of BoostTrack while greatly enhancing runtime efficiency, highlighting the advantage of performing feature extraction directly within the detector backbone.

### 3.2 Comparative Analysis

#### 3.2.1 Baseline vs. BoostTrack

The results in Table 1 show clear differences between the groups of tracking models. Baseline methods such as SORT, BoT-SORT, and OC-SORT achieve relatively high frame rates but limited accuracy in metrics such as HOTA and IDF1. This finding indicates that, although computationally efficient, baseline trackers are less reliable in maintaining identity consistency when facing occlusion or appearance variation.

BoostTrack demonstrates significant improvements in accuracy, reaching a HOTA of 27.457 and an IDF1 of 37.724, which are among the highest of all evaluated models. This improvement reflects the effectiveness of its confidence scaling mechanism and the integration of multiple metrics that reduce association errors. However, its runtime speed remains limited at only 6.65 FPS, making it less suitable for real-time use in resource-constrained environments.

#### 3.2.2 LITE Variants

The LITE-based trackers, including LITE-BoT-SORT and LITE-OC-SORT, show a clear advantage in efficiency, achieving substantially higher frame rates. For example, LITE-BoT-SORT reaches 15.175 FPS, the highest among all evaluated models. This performance improvement results primarily from the removal of the external Re-ID module and the direct use of internal embeddings extracted from YOLOv8. However, the gain in efficiency is accompanied by a slight reduction in identity preservation, as reflected in lower IDF1 scores compared to BoostTrack.

#### 3.2.3 LITE-BoostTrack

Combining both design philosophies leads to LITE-BoostTrack, which achieves a balanced trade-off between accuracy and efficiency. The model records a HOTA of 27.316 and an IDF1 of 37.485, values that are nearly identical to those of the standard BoostTrack, while increasing the runtime to 13.225 FPS, which is more than double the speed of the original version. These results confirm that LITE-BoostTrack maintains strong identity consistency while significantly improving computational efficiency. Consequently, it represents an effective and practical solution for real-time applications where both accuracy and processing speed are essential.

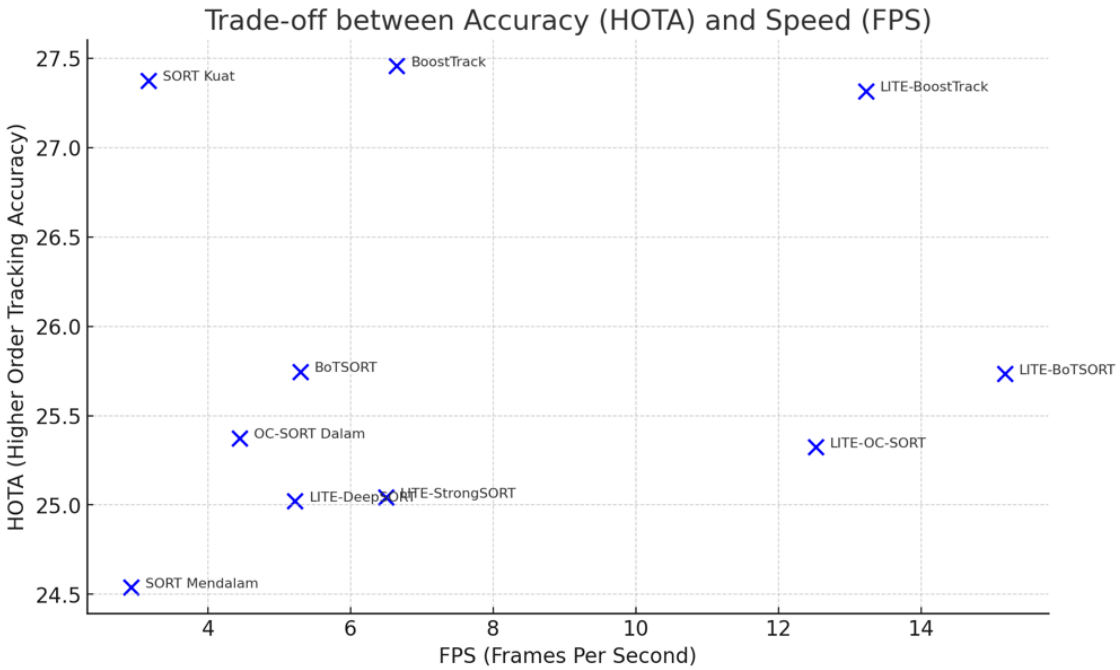



*Figure 5. Accuracy Versus Efficiency Comparison of MOT Models on MOT20*

Figure 5 shows how each model performs in terms of tracking accuracy and processing speed. The horizontal axis represents frames per second (FPS), which indicates how many video frames the model can process each second. A higher FPS means the model runs faster and is more suitable for real-time applications. The vertical axis represents Higher Order Tracking Accuracy (HOTA), which reflects how accurately the model can detect and follow multiple objects over time. A higher HOTA value means the model makes fewer identity or tracking errors.

As shown in the figure, simpler baseline models such as SORT and BoT-SORT achieve higher FPS, meaning they run quickly, but their HOTA values are relatively low. This indicates that these models often lose track of objects when occlusions or overlapping occur. BoostTrack achieves much higher accuracy, with the highest HOTA among all models, but it operates more slowly and therefore is less efficient for real-time processing on limited devices.

The LITE-based trackers improve speed by reducing unnecessary computations. For instance, LITE-BoT-SORT reaches 15.18 FPS, showing that removing the external Re-ID process allows the model to process frames faster. However, this increase in speed slightly reduces accuracy. The proposed LITE-BoostTrack achieves both high accuracy (HOTA 27.32) and high speed (13.23 FPS), showing that it can track objects reliably while still running efficiently. This balance between speed and accuracy makes LITE-BoostTrack well suited for real-time visual tracking tasks such as surveillance or UAV-based monitoring.

### 3.3 Ablation Study

To evaluate the specific contribution of integrating the LITE architecture into the BoostTrack framework, an ablation study was conducted using three representative benchmark models: BoT-SORT, BoostTrack and LITE-BoT-SORT. The comparison results are presented in Figure 5 and Table 2, focusing on three principal metrics, namely HOTA, IDF1, and FPS. All experiments were performed on the MOT20 training subset, and the evaluation metrics were calculated using the TrackEval framework to ensure consistent and fair comparison among the models.

*Table 2. Comparative Results of Real-Time Multi-Object Tracking Frameworks on the MOT20 Training Subset*

| Reference (Year) | Tracker / Variant | HOTA | MOTA | IDF1 | IDSW | FPS (Average) |
|---|---|---|---|---|---|---|
| Alikhanov et al. (2025) [6] | BoTSORT (Baseline) | 26.3 | 28,9 | 33,6 | 4801 | 7,5 |
| Alikhanov et al. (2025) [6] | LITE-BoTSORT | 25,73 | 28,47 | 33,88 | 4674 | 15,18 |
| This Study | LITE-BoostTrack | 27,31 | 28,96 | 37,48 | 2939 | 13,22 |

Table 2 summarizes the comparison results among three multi-object tracking frameworks that share the same evaluation protocol using the MOT20 training subset. The compared methods include the original BoT-SORT, the lightweight variant LITE-BoT-SORT from Alikhanov et al. [6], and the proposed LITE-BoostTrack model. This comparison aims to evaluate how well each method balances tracking accuracy, identity consistency, and processing efficiency under crowded and challenging scenes.

The results clearly show that LITE-BoostTrack achieves the most balanced performance among the three. It records a HOTA score of 27.31 and an IDF1 score of 37.48, outperforming LITE-BoT-SORT, which achieves 25.73 and 33.88, respectively. Although the numerical gap may seem modest, these improvements represent stronger identity association and more stable tracking performance across frames. In multi-object tracking, even small increases in IDF1 or HOTA are considered meaningful because they indicate fewer mismatched identities and more consistent trajectory links.

The improvement is also reflected in the number of identity switches (IDSW), which decreases significantly from 4,674 in LITE-BoT-SORT to 2,939 in LITE-BoostTrack. This reduction, reaching nearly 37%, demonstrates that the proposed system is more capable of maintaining consistent object identities even in dense scenes where occlusion frequently occurs. The improvement arises mainly from the integration of BoostTrack's motion-boosting and adaptive association strategy, which complements LITE's lightweight feature extraction mechanism.

From an efficiency perspective, LITE-BoostTrack operates at 13.22 FPS, slightly lower than LITE-BoT-SORT (15.18 FPS) but still almost twice as fast as the original BoT-SORT (7.5 FPS). This minor drop in frame rate is acceptable considering the substantial gain in identity stability and tracking accuracy. In practice, this means that LITE-BoostTrack remains well within real-time operational limits while providing more reliable tracking outputs, especially on embedded or edge-computing platforms where computational resources are limited.

Overall, these findings confirm that the proposed LITE-BoostTrack successfully combines the strengths of two complementary frameworks: the high-speed embedding efficiency of LITE and the robust motion-driven association mechanism of BoostTrack. The result is a hybrid system that achieves both real-time performance and improved tracking quality. Such a balanced design is essential for deploying multi-object tracking systems in practical applications like crowd surveillance, traffic monitoring, and autonomous navigation, where both accuracy and speed are equally important.

Despite these improvements, occasional tracking failures still occur under extreme occlusion and close-proximity interactions, where multiple pedestrians overlap or reappear after long-term disappearance. In such cases, partial feature suppression may lead to temporary identity switches or missed re-associations. These observations highlight that, while LITE-BoostTrack effectively reduces identity fragmentation compared to LITE-BoT-SORT, maintaining full robustness under severe visual occlusion remains an open challenge for future refinement.

## 4. Conclusion

This study proposed LITE-BoostTrack, a hybrid multi-object tracking framework that integrates LITE's efficient embedding extraction with BoostTrack's motion-boosting and adaptive association mechanisms. The model was designed to address the trade-off between accuracy and efficiency observed in previous frameworks, where heavy trackers such as BoT-SORT and BoostTrack achieved high accuracy at the cost of runtime speed, while lightweight methods like LITE and LITE-BoT-SORT provided faster inference but lower identity stability.

Experimental evaluations on the MOT20 benchmark demonstrate that LITE-BoostTrack achieves a balanced performance with a HOTA score of 27.31, IDF1 of 37.48, and FPS of 13.22, outperforming LITE-BoT-SORT (HOTA 25.73, IDF1 33.88, FPS 15.18) while remaining nearly twice as fast as the original BoT-SORT (7.5 FPS). The number of identity switches (IDSW) also decreased significantly from 4,674 to 2,939, representing a 37% reduction, which indicates stronger long-term identity association. The consistency of these results across repeated evaluations and comparisons with established trackers further validates the reliability and robustness of the proposed approach.

Despite these improvements, occasional failures still occur under severe occlusion and close-proximity interactions, where overlapping pedestrians may cause temporary identity switches or missed re-associations. Future research will focus on enhancing feature embedding resilience and adaptive re-identification to achieve higher robustness in dense-crowd environments.

Overall, the findings confirm that LITE-BoostTrack effectively bridges the gap between accuracy-oriented and efficiency-oriented MOT frameworks, delivering a practical and real-time solution for applications such as crowd surveillance, traffic monitoring, and autonomous navigation.

## Acknowledgement

This work was supported by the Institute for Research and Community Service (LPPM) of Dian Nuswantoro University under Grant No. 005/A.38-04/UDN-09/I/2025.

## References

[1] M. A. Altaf and M. Y. Kim, "Multiple object detection and tracking in autonomous vehicles: A survey on enhanced affinity computation and its multimodal applications," Aug. 01, 2025, *Korean Institute of Communications and Information Sciences*. https://doi.org/10.1016/j.icte.2025.06.005

[2] H. Wang, L. Jin, Y. He, Z. Huo, G. Wang, and X. Sun, "Detector–Tracker Integration Framework for Autonomous Vehicles Pedestrian Tracking," *Remote Sens. (Basel).*, vol. 15, no. 8, Apr. 2023. https://doi.org/10.3390/rs15082088

[3] X. Zhou, Y. Jia, C. Bai, H. Zhu, and S. Chan, "Multi-object tracking based on attention networks for Smart City system," *Sustainable Energy Technologies and Assessments*, vol. 52, Aug. 2022. https://doi.org/10.1016/j.seta.2022.102216

[4] M. Elshahawy, A. O. Aseeri, S. El-Sappagh, H. Soliman, M. Elmogy, and M. Abu-Elkheir, "Identification and Classification of Crowd Activities," *Computers, Materials and Continua*, vol. 72, no. 1, pp. 815–832, 2022. https://doi.org/10.32604/cmc.2022.023852

[5] J. Yan, S. Du, and Y. Wang, "Multi-Pedestrian Tracking in Crowded Scenes by Modeling Movement Behavior and Optimizing Kalman Filter," *IEEE Access*, vol. 10, pp. 118512–118521, 2022. https://doi.org/10.1109/ACCESS.2022.3220635

[6] J. Alikhanov, D. Obidov, M. Abdurasulov, and H. Kim, "Practical Evaluation Framework for Real-Time Multi-Object Tracking: Achieving Optimal and Realistic Performance," *IEEE Access*, vol. 13, pp. 34768–34788, 2025. https://doi.org/10.1109/ACCESS.2025.3541177

[7] P. Zhang, D. Wang, and H. Lu, "Multi-modal visual tracking: Review and experimental comparison," Apr. 01, 2024, *Tsinghua University*. https://doi.org/10.1007/s41095-023-0345-5

[8] S. Honarparvar, Z. B. Ashena, S. Saeedi, and S. Liang, "A Systematic Review of Event-Matching Methods for Complex Event Detection in Video Streams," Nov. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. https://doi.org/10.3390/s24227238

[9] Y. Li, Y. Liu, C. Zhou, D. Xu, and W. Tao, "A lightweight scheme of deep appearance extraction for robust online multi-object tracking," *Visual Computer*, vol. 40, no. 3, pp. 2049–2065, Mar. 2024. https://doi.org/10.1007/s00371-023-02901-2

[10] L. Ye, W. Li, L. Zheng, and Y. Zeng, "Lightweight and Deep Appearance Embedding for Multiple Object Tracking," *IET Computer Vision*, vol. 16, no. 6, pp. 489–503, Sep. 2022. https://doi.org/10.1049/cvi2.12106

[11] Z. Wan and W. Wu, "A robust approach to deformed pedestrian tracking with multi-trajectory prediction," *Cluster Comput.*, vol. 28, no. 5, Oct. 2025. https://doi.org/10.1007/s10586-024-05059-1

[12] V. M. Scarrica, C. Panariello, A. Ferone, and A. Staiano, "A hybrid approach to real-time multi-target tracking," Jun. 01, 2024, *Springer Science and Business Media Deutschland GmbH*. https://doi.org/10.1007/s00521-024-09799-4

[13] H. Li *et al.*, "Multi-object tracking via deep feature fusion and association analysis," *Eng. Appl. Artif. Intell.*, vol. 124, Sep. 2023. https://doi.org/10.1016/j.engappai.2023.106527

[14] K. Sriram and K. Purushotham, "Multiple object tracking using space-time adaptive correlation tracking," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 3, pp. 1805–1815, 2023. https://doi.org/10.11591/ijeecs.v32.i3.pp1805-1815

[15] P. Dendorfer *et al.*, "MOT20: A benchmark for multi object tracking in crowded scenes," Mar. 2020. https://doi.org/10.48550/arXiv.2003.09003

[16] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, Sep. 2017, pp. 3645–3649. https://doi.org/10.1109/ICIP.2017.8296962

[17] Y. Du *et al.*, "StrongSORT: Make DeepSORT Great Again," *IEEE Trans. Multimedia*, vol. 25, pp. 8725–8737, Feb. 2023. https://doi.org/10.1109/TMM.2023.3240881

[18] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2023, pp. 9686–9696. https://doi.org/10.1109/CVPR52729.2023.00934

[19] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani, "Deep OC-Sort: Multi-Pedestrian Tracking by Adaptive Re-Identification," in *Proceedings - International Conference on Image Processing, ICIP*, IEEE Computer Society, 2023, pp. 3025–3029. https://doi.org/10.1109/ICIP49359.2023.10222576

[20] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "BoT-SORT: Robust Associations Multi-Pedestrian Tracking," Jul. 2022. http://arxiv.org/abs/2206.14651

[21] V. D. Stanojevic and B. T. Todorovic, "BoostTrack: boosting the similarity measure and detection confidence for improved multiple object tracking," *Mach. Vis. Appl.*, vol. 35, no. 3, May 2024. https://doi.org/10.1007/s00138-024-01531-5

[22] Q. Wan *et al.*, "A transformer-based lightweight method for multiple-object tracking," *IET Image Process.*, vol. 18, no. 9, pp. 2329–2345, Jul. 2024. https://doi.org/10.1049/ipr2.13099

[23] S. Li, H. Ren, X. Xie, and Y. Cao, "A Review of Multi-Object Tracking in Recent Times," Jan. 01, 2025, *John Wiley and Sons Inc*. https://doi.org/10.1049/cvi2.70010

[24] P. Karthikeyan, Y. H. Liu, and P. A. Hsiung, "LightMOT: Lightweight and anchor-free solution for tracking multiple objects in dense populations," *Future Generation Computer Systems*, vol. 166, May 2025. https://doi.org/10.1016/j.future.2024.107690

[25] J. Alikhanov, D. Obidov, and H. Kim, "LITE: A Paradigm Shift in Multi-object Tracking with Efficient ReID Feature Integration," vol. 15293, M. Mahmud, M. Doborjeh, K. Wong, A. C. S. Leung, Z. Doborjeh, and M. Tanveer, Eds., in Lecture Notes in Computer Science, vol. 15293., Singapore: Springer Nature Singapore, 2025, pp. 92–106. https://doi.org/10.1007/978-981-96-6596-9_7