OPTIMIZING TRAFFIC MANAGEMENT IN DANANG: A COMPARATIVE STUDY OF MULTI-OBJECT TRACKING TECHNIQUES FOR REAL-TIME VEHICLE FLOW MONITORING

Hai T. Ton[1], Hung V. Nguyen[1], Hanh T. M. Tran[1], Tien V. Thai[1], Phong-Phu Le[2], Tung T. Huynh[1], Duy-Tuan Dao[1]*

[1]The University of Danang - University of Science and Technology, Vietnam
[2]National Cheng Kung University, Tainan, Taiwan

*Corresponding author: ddtuan@dut.udn.vn

(Received: January 19, 2024; Revised: February 21, 2024; Accepted: February 22, 2024)

Abstract - This study evaluates the effectiveness of various detection-based object-tracking algorithms to optimize accuracy and efficiency in traffic flow monitoring. Due to its high accuracy in detecting objects, YOLOv8 was chosen as the vehicle detector for this research, where precise and rapid vehicle detection was critical. Regarding object tracking, our focus centered on the evaluation of five prominent Multiple Object Tracking (MOT) algorithms, including BoTSORT, ByteTrack, DeepOCSORT, OCSORT, and StrongSORT. We introduce a comprehensive traffic urban dataset collected from intricate street networks in Danang City. Our experimental results show that the system has practical applicability in urban traffic monitoring. Notably, the best model achieves a detection accuracy of 0.721 on mAP@0.5-0.95, and the High Overlap Tracking Accuracy (HOTA) surpasses 72% for tracking performance across diverse traffic scenarios. This shows the applicability of MOT algorithms and provides a detailed view of traffic flow monitoring, especially in Danang City, Vietnam.

Key words - Traffic Flow Monitoring; Object Detection; YOLOv8; Multiple Object Tracking (MOT)

1. Introduction

In recent years, there has been a significant and rapid increase in the number of registered vehicles, encompassing both motorized and non-motorized types. According to data from the Vietnam Register, in 2022, the country hosts nearly 5 million cars and over 70 million motorcycles [1]. This surge underscores an escalating volume of traffic on the roads, which has profound implications for Vietnam's socio-economic development. It presents a dual-faceted scenario: an opportunity for growth in the transport sector and a challenge due to the substantial burden placed on traffic infrastructure and the complexities in traffic management and supervision.

For the implementation of an effective traffic monitoring system, it is imperative to gain an in-depth understanding of vehicular movement patterns and the dynamics of traffic flow on various routes. Traditional traffic monitoring methods, such as ultrasonic waves, radar, or infrared sensors, face numerous operational challenges. They are often costly and yield data that is not comprehensive. As such, there is a growing need for innovative approaches that can overcome these limitations and provide a more holistic and efficient solution to the evolving demands of traffic management.

Modern trends in automation and the increasing application of Artificial Intelligence (AI) in various life aspects have not excluded traffic management and monitoring. Several works proposed methods to support surveillance cameras installed across major routes and key intersections for violation detection [2], [3], [4]. These research, employing deep learning techniques, notably YOLOv3, for violation detection, demonstrates the potential of AI in traffic management. However, it also highlights key limitations, such as dependency on lighting conditions and challenges with high-speed traffic. An AI-powered video surveillance system can create an extensive traffic data repository, easily accessible for information like vehicle count, directions, waiting times, etc., through video devices integrated with image processing and analysis systems. Although several studies applied traffic monitoring technologies to Vietnam conditions [6], [7], [8], however; Vietnam's traffic characteristics include diverse terrain, unique vehicles, and a distinct traffic culture, such as a variety of motorized vehicles, from two-wheeled motorcycles, three-wheeled bikes, motorbikes, and cars to trucks. In contrast, our study expands on these foundations by employing the more advanced YOLOv8 model and a comprehensive evaluation of multiple object-tracking methods. Our approach not only addresses some of the mentioned limitations but also tailors the solution to the unique urban traffic context of Danang City. By offering a nuanced analysis of different MOT methods, we aim to provide a more robust and adaptable framework for traffic flow optimization, setting a new benchmark in the field.

As we explore optimizing traffic flow monitoring, it is instructive to consider the advancements and challenges in related domains. Our approach aligns with innovative research directions [9], particularly in employing the Region of Interest (ROI) for identifying and tracking vehicles. This dual-module strategy, encompassing object detection and monitoring, transcends the drawbacks of previous methodologies, as further elaborated in the subsequent sections. In this paper, we leverage deep learning techniques, a subset of machine learning focused on artificial neural networks, to address traffic monitoring challenges. Specifically, we have opted for the YOLOv8 model combined with multi-object tracking methods to enable real-time traffic flow monitoring and comprehensive analysis. The process is depicted in the following diagram:

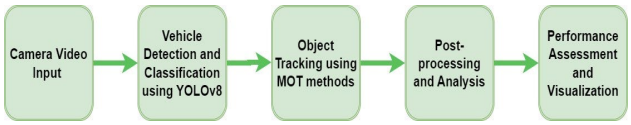

Figure 1. *Framework of the Traffic Flow Monitoring System*

The contribution of this research lies in evaluating and comparing the performance of various state-of-the-art multi-object tracking methods, with the primary objective of employing them for the detection, classification, and tracking of several types of traffic vehicles in a use case of Danang City. This is accomplished through the utilization of advanced image and video processing techniques, complemented by efficient detection and tracking methods, resulting in real-time traffic monitoring and detection methods. Specifically, as illustrated in Figure 1, the procedural framework includes five steps: (1) decomposing the input video sourced from the traffic camera system into frames; (2) detecting and classifying vehicles into predefined categories (car, motorcycle, bus, and truck) within these frames using the YOLOv8 model; (3) tracking and assigning unique IDs to monitor the information of vehicles presented in the video; (4) analyzing the traffic flow such as counting the number of vehicles; and (5) giving assessment and visualizing the representation of the tracked vehicles and traffic flow. As the experimental evaluation and results, the outcomes of our research are that we undertake a comprehensive analysis and comparison of various object-tracking methods for traffic flow monitoring.

2. Related Works

While existing studies in traffic vehicle monitoring offer valuable insights, limitations persist. In [5], the authors focused on the density estimation issue, which analyzed specific object characteristics to estimate quantity, yet it falled short in identifying precise vehicle trajectories, impacting accuracy. In [6], the authors integrated the vehicle detection and tracking method for improved accuracy, employing techniques like horizontal stripes for monitoring vehicles. These, however, these proposed methods suffered from the limitation in their application to complex intersections and specific road types. Additionally, methods utilizing all frames for object tracking faced challenges at wide camera angles, leading to detection, and tracking inaccuracies.

Building upon these methodologies, our study takes a significant leap forward. We build upon and push beyond the boundaries established by previous work, including [7]. The previous research, while insightful, was constrained by its specific context and the complexity of its algorithms. Our current research broadens this perspective, applying advanced multi-object tracking methods in a more diverse urban setting. By utilizing a robust dataset and algorithms fine-tuned for real-time application, we effectively address the challenges of data dependency and scalability. This methodological evolution not only surmounts earlier limitations but also enriches the comparative analysis with a broader spectrum of tracking methods, establishing new standards in urban traffic flow monitoring.

2.1. Object Detection

There are two primary branches of object detection methods: single-stage and two-stage, with the latter being more dominant in the field of object detection. Single-stage methods, such as You Only Look Once (YOLO) [10] and Single-shot Detectors (SSD) [11], approach object recognition as a regression problem where the coordinates of the bounding box and object classes are predicted directly. However, two-stage methods, like Region-based CNN (R-CNN) [12], utilize a searching approach, initially proposing regions of interest (RoI). These proposals are then sent for classification and bounding box regression. This method achieves higher accuracy than the single-stage approach but requires more processing time due to its multiple stages.

Recently, the YOLOv8 model, the latest iteration in the YOLO series, emerged as the gold standard in object detection. It stands out not only for its rapid detection capabilities but also for maintaining high accuracy. It demonstrates a higher mean Average Precision (mAP) compared to its predecessors with equivalent parameters, making it suitable for real-time processing of large volumes of image data. Therefore, this paper utilizes the YOLOv8 model for detecting traffic objects, ensuring effectiveness and accuracy in processing data from traffic cameras in Danang City.

2.2. Object Tracking

Object tracking has emerged as a crucial tool in deep learning and computer vision, enabling us to monitor the movements of objects across video sequences. By leveraging spatial and temporal information, object tracking assigns unique IDs to detected objects and follows their trajectories throughout a video. Object tracking can be categorized into two types: Single Object Tracking (SOT) and Multiple Object Tracking (MOT) [13]. MOT can identify and track multiple objects in a single frame and then assign and maintain IDs across different frames, making it ideal for applications like vehicle monitoring.

Although many studies and technological solutions have been successfully implemented in traffic monitoring abroad, adapting these to Vietnam's complex traffic context is not straightforward. Vietnamese traffic is characterized by diverse terrains, unique vehicles, and a distinct traffic culture. This diversity is evident in the range of motorized road vehicles, from two-wheeled and three-wheeled motorcycles to motorbikes, cars, and trucks.

In this context, the use of MOT becomes essential for monitoring traffic flow. There are several strong research and development methods, including ByteTrack [14], OCSORT [15], Deep-OCSORT [16], BoTSORT [17], and StrongSORT [18], however; each tracking method has its strengths and limitations. Notably, most previous studies have focused on evaluating the performance of these methods on pedestrian datasets like MOT16 [19], DanceTrack [20], etc. The ranking of methods varies across datasets. Therefore, this paper will focus on exploring and comparing the effectiveness of MOT methods on vehicle traffic in Danang City, Vietnam. The

goal is to provide a detailed and insightful view of the applicability of MOT methods under specific traffic conditions, hoping to expand their application in urban traffic monitoring.

3. The Proposed Methodology

When processing objects at a distance, the system may encounter challenges in recognition, leading to a low detection rate and adversely affecting the tracking process. To address this issue, we employ the Region of Interest (ROI) method to identify and track objects at specific locations precisely. This not only addresses recognition challenges but also minimizes errors in detecting objects.

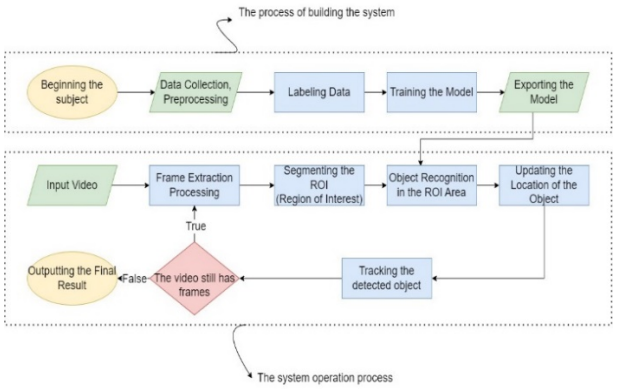

Figure 2. *Flowchart of the Traffic Vehicle Monitoring Algorithm*

Figure 2 illustrates the algorithm flowchart for detecting, recognizing, and tracking traffic vehicles. This process initiates with labeling, training the recognition and detection model, and exporting the model. Then, the model is utilized to detect and identify vehicles on testing input video. Subsequently, the process of counting vehicles in the video is carried out. The first step involves defining the Region of Interest (ROI). Each frame is then separated, and the process of identifying vehicles in each frame is iteratively performed. The positions of the vehicles are continuously updated in each frame. Upon a vehicle entering the ROI, the count variable is incremented according to the identified class. This iterative process continues until the end of the video.

3.1. Object Detection

3.1.1. Training YOLOv8

Figure 3 presents a comparison of the mean Average Precision (mAP) on the COCO dataset across various YOLOv8 variants [9], along with a comparison of the number of parameters in these variants against older versions of YOLO. In our study, we specifically employed YOLOv8n, the lightest variant of YOLOv8, designed to provide high frames per second (FPS) and suitable for real-time applications. To optimize performance, the data underwent pre-processing, resized to 640x640. Subsequently, we applied data augmentation techniques such as cutout and rotation to enrich the dataset.

The training process was executed on our dataset for 300 epochs with a batch size of 16 on a Google Colab V100 GPU with 51GB RAM. This powerful computing environment significantly contributed to expediting the training speed, thereby enhancing research efficiency.

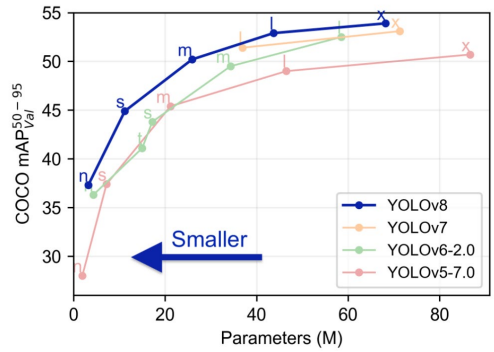

Figure 3. *Comparison of mAP (mean Average Precision) among YOLO models [9]*

3.1.2. Evaluation of Object Detection Models

In the problem of object detection, the evaluation of the model's accuracy and effectiveness was conducted using the following methods:

- **Intersection over Union (IoU):** IoU is a metric used to measure the overlap between the predicted bounding box and the actual (ground truth) bounding box. The IoU value is calculated by the ratio of the area of overlap between the two bounding boxes to their total area. When IoU is equal to 1, it indicates that the predicted bounding box perfectly matches the actual bounding box. The following image visually illustrates how IoU is calculated:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (1)$$

Regarding this:

+ "Area of Overlap" is the area of intersection between the predicted bounding box and the ground truth.

+ "Area of Union" includes the combined area of the predicted bounding box and the ground truth.

- A prediction is considered correct if the IoU between the predicted and the ground truth bounding boxes is greater than a pre-determined threshold value. Based on the IoU and this threshold, we can calculate the following metrics:

+ **True Positive (TP):** The model predicts that a bounding box exists at a specific location (Positive) and this is correct (True)

+ **False Positive (FP)**: The model predicts that a bounding box exists at a specific location (Positive) but this is incorrect (False).

+ **False Negative (FN)**: The model does not predict a bounding box at a specific location (Negative) and this is incorrect (False), meaning the actual bounding box does exist at that location.

- **Precision and Recall**: Precision measures the proportion of correct predictions made by the model out of all positive predictions, while Recall measures the proportion of correct predictions made by the model out of all actual positive cases in the data. These are calculated using the following two formulas:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2); \quad \text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- **Average Precision (AP)**: As discussed earlier, different threshold values result in varying precision. Therefore, AP is utilized to provide an objective measure.

$$\text{AP} = \int_{r=0}^{1} p(r)dr \quad (4)$$

- **Mean Average Precision (mAP)**: This is an extension of AP, calculated by averaging the AP across all object classes.

$$mAP = \frac{1}{k}\sum_{i}^{k} AP_i \quad (5)$$

In recent papers, researchers have employed mAP@0.5 with an IoU threshold of 0.5 for assessing models in simple detection scenarios, and mAP@0.5-0.95 with thresholds ranging from 0.5 to 0.95 in steps of 0.05 to provide a comprehensive view of the model's performance at various levels of detection difficulty. In this paper, we use both metrics.

3.2. Object Tracking

3.2.1. Tracking Techniques

The multiple object-tracking methods employed in our work are state-of-the-art (SOTA) and have demonstrated impressive performance across various benchmarks:

ByteTrack: Focuses on linking bounding boxes, including those with low confidence. This method has achieved top results (SOTA) on multiple datasets.

OCSORT: Improves the traditional Kalman filter to enhance tracking performance. It uses an object-centric approach to calculate hypothetical trajectories, helping to minimize cumulative error and effectively handle non-linear movements. OCSORT offers real-time performance and has achieved top results on several datasets.

Deep-OCSORT: An upgraded version of OCSORT, it utilizes deep appearance attributes, achieving top positions on MOT20 and second place on MOT17.

BoTSORT: Integrates information about movement, object characteristics, and camera motion with the Kalman filter to create a robust tracking method. It ranks among the top on MOT17 and MOT20 datasets.

StrongSORT [18]: Developed from DeepSORT, it improves detection, embedding, and linking to improve tracking efficiency.

In this study, we applied these five different tracking methods to the Danang City traffic dataset to provide the most comprehensive overview of this dataset.

3.2.2. Evaluation and Analysis of Tracking Methods

In recent years, the growing interest and investment in the autonomous vehicle industry have significantly propelled the development of the Multiple Object Tracking (MOT) research community. The autonomous vehicle industry demands systems capable of accurately and continuously tracking multiple objects, ranging from other vehicles to pedestrians. This enhancement not only improves the safety and efficiency of autonomous driving but also has applications in fields like security surveillance and traffic management.

The growth in MOT research has also led to the proposal of several new benchmarks. These benchmarks measure how well tracking systems perform by comparing the model's predicted outcomes (predictions) with actual tracks (ground truth), as illustrated in Figure 4.

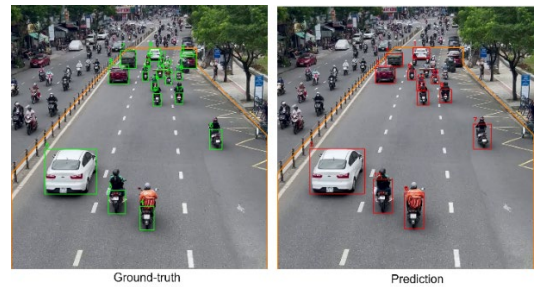

Figure 4. *Ground-truth and prediction*

Many tracking methods utilize the Multiple Object Tracking Accuracy (MOTA) metric for evaluation [21]. This metric measures the overall accuracy of both the tracking and detection processes. It accounts for the outputs from both tracking and detection. The MOTA metric reflects the number of errors from missed objects (*FN*), false positives (*FP*), and mismatches (*IDS*) in predictions.

$$MOTA = 1 - \frac{\sum_t FN_t + FP_t + IDS_t}{\sum_t GT_t} \quad (6)$$

The second metric employed is the Identification F1-Score (IDF1) [22], which reflects the accuracy of object association more than object detection. IDF1 calculates a one-to-one mapping between actual trajectories and predicted trajectories, using ID True Positive (IDTP), ID False Negative (IDFN), and ID False Positive (IDFP).

$$\text{IDF1} = \frac{|\text{IDTP}|}{|\text{IDTP}| + 0.5|\text{IDFN}| + 0.5|\text{IDFP}|} \quad (7)$$

A high IDF1 score estimates the number of unique objects within a scene rather than providing information about the ability to detect or accurately link them.

Recently, the Higher Order Tracking Accuracy (HOTA) metric, which is capable of evaluating all tracking aspects, was introduced [23]. It is designed to overcome many limitations of previous metrics like MOTA and IDF1. This metric assesses all aspects of tracking and comprises three main components.

Accurate Detection (DetA) – measures the accuracy of object existence compared to ground truth.

$$\text{Det A} = \text{Det-IoU} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|} \quad (8)$$

Association Accuracy (AssA) – this metric assesses the accuracy based on incorrect associations made by the model, such as assigning the same ID to two detections that have different ground truths.

$$\begin{aligned} \text{AssA} &= \frac{1}{|\text{TP}|}\sum_{c\in\text{TP}} \text{Ass-IoU}\ (c) \\ &= \frac{1}{|\text{TP}|}\sum_{c\in\text{TP}} \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + |\text{FNA}(c)| + |\text{FPA}(c)|} \end{aligned} \quad (9)$$

Localization Accuracy (LocA) – measures the precision of the detected object's location compared to its actual position.

$$\text{LocA} = \frac{1}{|\text{TP}|}\sum_{c\in\text{TP}} \text{Loc-IoU}\ (c) \quad (10)$$

Finally, HOTA is calculated using the following

formula:

$$\mathrm{HOTA}_\alpha = \sqrt{\frac{\sum_{c\in\{\mathrm{TP}_\alpha\}} \text{Ass-IoU}_\alpha(c)}{|\mathrm{TP}_\alpha|+|\mathrm{FN}_\alpha|+|\mathrm{FP}_\alpha|}} \quad (11)$$

$$\mathrm{HOTA} = \int_{0<\alpha\leq 1} \mathrm{HOTA}_\alpha \approx \frac{1}{19}\sum_{\substack{\alpha=0.05\\ \alpha+=0.05}}^{0.95} \mathrm{HOTA}_\alpha \quad (12)$$

4. Experimental Evaluation and Results

4.1. Dataset

Data collection from real scenarios plays an extremely crucial role in researching and analyzing object-tracking methods in Traffic Flow Monitoring. The data we collected is particularly diverse and rich, accurately reflecting the real-life traffic conditions in Danang. Here are some examples of our data with rain conditions and diverse angles (Figure 5):

- **Diverse camera angles**: We captured images from various perspectives, including front, rear, and diagonal angles, to provide a comprehensive view of traffic flow and vehicle movement patterns.
- **Varying weather conditions**: Data was collected under different weather conditions such as overcast, sunny with shadows, and even in rain, where road users often wear raincoats, creating unique challenges for tracking and analysis.
- **Density variation**: Data collection occurred in both low-traffic areas and traffic hotspots in Danang, such as the Dien Bien Phu and Nguyen Tri Phuong streets, reflecting a wide range of traffic scenarios and situations.

a. Rain b. Intersection

Figure 5. *Danang Traffic - Diverse Angles View*

Our data was categorized into four types of labels: motorcycles, cars, buses, and trucks. After the collection process, we obtained 50 videos with diverse traffic conditions. Out of these, we selected 43 videos for training the YOLOv8 model and 7 videos for evaluating our method. Before training data, these videos were labeled. Based on the 43 selected videos, we created around 1200 images and labeled over 22000 objects in this data. The results are summarized in Table 1.

Table 1. *Training and Testing Dataset of the System*

	Car	Motor-cycles	Bus	Truck
Training	6229	12266	332	695
Validation	552	1205	37	62
Testing	269	513	16	30
Total	***22206***			

Table 2. *Characteristics of the Videos in the Traffic Video Dataset in Danang*

Video	Descriptions			Size *(pixels)*	FPS	Length *Frames (Seconds)*	Bounding Boxes
	View	Weather	Traffic Density				
DN-1	Frontal rear	Overcast sky	High (14.9)	1920x1080	30	488 (00:16)	7281
DN-2	45-degree angle	Sunny sky	Medium (8.9)	1920x1080	30	574 (00:19)	5114
DN-3	45-degree angle	Rainy sky	Low (5.0)	1920x1080	30	490 (00:16)	2436
DN-4	45-degree angle	Overcast sky	Medium (8.7)	1920x1080	30	618 (00:20)	5372
DN-5	30-degree rear, intersection	Overcast sky	High (12.5)	1920x1080	30	608 (00:20)	7624
DN-6	Frontal	Overcast sky	High (15.6)	1080x1920	30	619 (00:20)	9666
DN-7	30-degree rear, two-way road	Rainy, Flooded	Low (3.6)	1920x1080	30	622 (00:25)	2266
Total						**4019 (136s)**	**39759**

To evaluate object tracking methods, we selected 7 manually annotated videos as mentioned in Table 2 as the ground truth. To ensure authenticity, we ensured that none of these videos had been used to train the YOLOv8 model.

4.2. Object Detection

To perform the training process, we utilized the Google Colab Pro platform with a Tesla V100 GPU card and 51GB of RAM.

Table 3. *Performance of pre-trained YOLOv8n and Trained YOLOv8n on Our Dataset*

Model	mAP@0.5	mAP@0.5-0.95
Pre-trained YOLOv8n (COCO weight)	0.595	0.377
YOLOv8n (Training on our dataset)	0.952	0.721

Table 3 compares the performance of two YOLOv8n models: a pre-trained model and a model trained on our dataset. The pre-trained YOLOv8n model, which was trained on the COCO dataset, a large and widely used dataset for object detection, achieved mAP@0.5 and mAP@0.5-0.95 scores of 0.595 and 0.377, respectively. It performed quite well when using an IoU threshold of 50%, but its performance significantly dropped when evaluated at higher IoU thresholds from 50% to 95%.

While the pre-trained YOLOv8n model with COCO weights demonstrates strong performance, a custom-trained model for our specific dataset takes object detection to another level. Starting with the same network architecture, we trained a YOLOv8n model from scratch using our data. This targeted approach paid off handsomely, as the model achieved a remarkable mAP@0.5 of 0.952 and a mAP@0.5-0.95 of 0.721.

This indicates that the model was well-optimized for the new dataset and could detect vehicles with high accuracy, even when applying higher IoU thresholds. The

YOLOv8n model trained on the new data outperforms the pre-trained model, both at the IoU threshold of 50% and in the wider range from 50% to 95%. This highlights the importance of training models on data that is specific to the intended context to achieve optimal performance.

This substantial improvement highlights the model's adeptness at recognizing vehicles, even with stricter IoU thresholds. Our custom-trained YOLOv8n outperforms the COCO-based model across both the standard 50% IoU and the wider range of 50% to 95%. This finding underscores a crucial point: for optimal performance, tailoring models to your specific data and context is paramount.

4.3. Object Tracking

Below are the test results with the following configuration: Intel Core i5 13400F CPU, NVIDIA GeForce RTX 3060 GPU, and 16 GB RAM. Tables 4, 5, 6, and 7 below show the performance of five multi-object tracking methods: ByteTrack, OCSORT, DeepOCSORT, BoTSORT, and StrongSORT, and applied to 7 selected videos from the dataset we collected. Specifically, Table 4 evaluates based on the HOTA index, Table 5 on MOTA, Table 6 on IDF1, and Table 7 assesses based on FPS. The highest performance on each video is marked as bold figures.

Analyzing the results from Tables 4, 5, and 6, we can observe that the BoTSORT and ByteTrack methods consistently rank at the top positions on most videos, indicating the highest performance. This suggests that both methods can track objects accurately and robustly. However, there is variability in the metrics across different videos, indicating differences in the conditions and characteristics of each video.

Table 4. *HOTA Score Results for the Traffic Video Dataset in Danang*

	Byte-track	OC-SORT	Deep-OCSORT	BoT-SORT	Strong-SORT
DN-1	80.29	80.50	80.63	**81.70**	78.72
DN-2	68.36	67.14	65.93	66.62	**70.30**
DN-3	67.43	72.31	**72.78**	64.33	68.90
DN-4	**71.56**	71.43	71.07	70.98	70.14
DN-5	62.20	62.57	61.82	**63.46**	57.52
DN-6	**86.72**	84.00	84.65	85.75	82.51
DN-7	67.00	61.61	58.70	**67.75**	59.77
Com-bined	**74.77**	73.96	73.79	74.66	72.30

Table 5. *MOTA Score Results for the Traffic Video Dataset in Danang*

	Byte-track	OC-SORT	Deep-OCSORT	BoT-SORT	Strong-SORT
DN-1	79.96	79.32	80.46	**80.64**	76.05
DN-2	**66.77**	63.10	62.83	60.89	66.00
DN-3	66.13	66.13	65.35	66.05	**67.20**
DN-4	72.86	72.82	72.23	70.14	**72.99**
DN-5	63.50	**64.21**	62.34	63.08	62.33
DN-6	82.59	**82.84**	83.11	82.60	81.33
DN-7	**65.36**	61.12	62.05	65.23	63.06
Com-bined	**73.11**	72.47	72.27	72.02	71.71

Table 6. *IDF1 Score Results for the Traffic Video Dataset in Danang*

	Byte-track	OC-SORT	Deep-OCSORT	BoT-SORT	Strong-SORT
DN-1	89.60	88.48	88.40	**90.35**	86.15
DN-2	73.86	72.03	71.53	75.15	**77.60**
DN-3	73.00	77.49	**78.83**	67.17	72.72
DN-4	**78.88**	78.63	78.25	78.35	75.27
DN-5	74.82	73.53	72.53	**76.71**	67.98
DN-6	**92.00**	87.44	88.30	91.38	86.38
DN-7	76.53	66.70	67.94	**78.34**	67.58
Com-bined	82.61	80.51	80.58	**82.86**	80.65

Table 7. *Frame per Second Comparison on Trafic Video Dataset*

	Byte-track	OC-SORT	Deep-OCSORT	BoT-SORT	Strong-SORT
DN-1	**14.72**	14.69	10.84	9.97	4.06
DN-2	16.18	**16.37**	12.69	12.70	9.08
DN-3	**17.74**	17.40	12.81	12.56	11.29
DN-4	15.71	**16.04**	12.21	11.78	7.32
DN-5	16.01	**16.18**	11.60	11.41	5.91
DN-6	14.76	**15.54**	10.84	10.42	4.69
DN-7	**17.52**	16.97	13.58	12.98	11.45
Com-bined	16.09	16.17	12.08	11.69	7.69

Table 7 provides an in-depth look at the performance of object-tracking methods across each video. Notably, ByteTrack and OCSORT consistently hold the top positions in processing speed (FPS). In contrast, BoTSORT and DeepOCSORT show lower performance in terms of FPS, and StrongSORT consistently ranks at the bottom in terms of FPS on all videos.

To gain a better understanding of these analyses, let us examine the graphs below, which depict the specific results of each method after aggregating data from 7 different videos. This will help us gain a clearer understanding of the performance differences among the methods and elucidate their specific strengths and weaknesses in real-world scenarios.

Table 8. *Comparison of Algorithms on Different Traffic Densities (Metrics: H - HOTA, M - MOTA, I - IDF1)*

		Byte-track	OC-SORT	Deep-OCSORT	BoT-SORT	Strong-SORT
Low	**H**	67.25	**67.39**	66.41	66.02	64.64
	M	**65.76**	63.72	63.76	65.65	65.21
	I	**74.70**	72.36	73.64	72.51	70.24
Mediu	**H**	**70.15**	69.52	68.78	69.06	69.36
	M	**69.89**	68.08	67.64	65.63	69.54
	I	76.58	75.60	75.19	**76.90**	74.49
High	**H**	77.75	76.77	76.90	**78.16**	74.31
	M	75.89	**76.01**	75.88	75.96	73.87
	I	86.26	83.74	83.79	**86.84**	80.96

Based on Table 8, we can see that under the condition of low traffic density, the research results show that the Byte-track algorithm brings high stable performance on all evaluation indicators, with high IDF1 score, reflecting the

ability to accurately identify and continuously track objects. This is especially important for maintaining the ID of each object over time in a low traffic density environment. OC-SORT, although not completely outperforming Byte-track, shows a slight improvement in the HOTA indicator, showing a better ability to balance target detection and tracking in low density. This is a sign that this algorithm handles well situations with little overlap and little confusion.

When the traffic density is medium, the results from Table 8 show that the Byte-track algorithm has a slight improvement in the HOTA indicator while maintaining a good IDF1 score, which proves the algorithm's strong ability to handle situations with increased overlap. OC-SORT, when placed in a condition of increased density, shows a slight decrease in all evaluation indicators. This proves that the algorithm may have difficulty handling a larger number of objects, as well as the increased overlap between objects. BoT-SORT, on the other hand, shows a significant increase in performance, with high HOTA and IDF1 scores, showing that it can handle well in medium traffic density conditions.

Finally, under high traffic density conditions, BoT-SORT shows its suitability in this condition, leading in both HOTA and IDF1 indicators, highlighting its superior ability to track in crowded traffic environments. This particular effectiveness indicates that BoT-SORT may be specifically designed to deal with the challenges of high density, where maintaining identity and accurately tracking objects becomes more difficult.

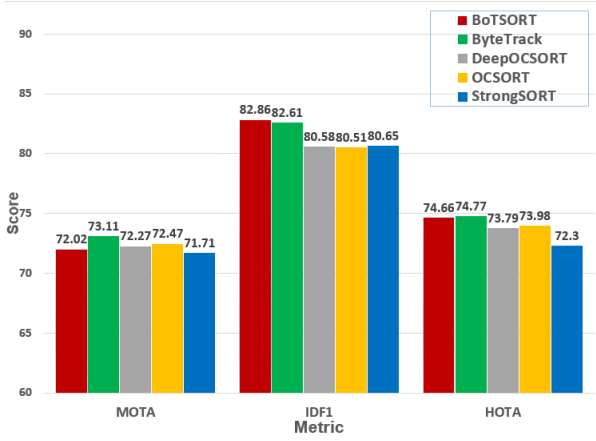

Figure 6. *Comparison of Evaluation Metrics for Object Tracking Methods*

Figure 6 summarizes the average performances of different methods with different metrics (Tables 4, 5, 6 and 7). ByteTrack and OCSORT have shown excellence in the MOTA index, with scores of 73.11 and 72.47, respectively. The slight difference between these two methods reflects an equivalent performance in accurately maintaining object identity and position across video frames, demonstrating effective error minimization. More notably, in the evaluation based on the IDF1 index, BoTSORT and ByteTrack emerge as the top-performing methods, achieving impressive scores of 82.86 and 82.61, respectively. This indicates that both methods possess outstanding capabilities in accurately maintaining object identity across multiple consecutive frames, a crucial factor reflecting strong data association throughout the tracking process.

When observing the HOTA index, we also find that BoTSORT and ByteTrack continue to lead the rankings with scores of 74.66 and 74.77, respectively. The prominence of these two methods in the HOTA index further reinforces the evidence of their ability to accurately determine object location and identity while maintaining this identity consistently and steadily across frames.

While the metrics assess tracking quality, FPS evaluates computational performance, and in this case, OCSORT and ByteTrack lead, indicating that they can process frames faster than other methods. Although BoTSORT appears to be the method with the highest tracking quality performance, ByteTrack and OCSORT can be excellent choices when balancing quality and processing speed. This can be particularly crucial when handling real-time videos or when there are requirements for fast processing. StrongSORT, while not the most outstanding, still maintains a reliable performance across all metrics.

5. Conclusion

Our study undertakes a comprehensive analysis and comparison of various object-tracking methods for traffic flow monitoring. Leveraging the YOLOv8 object detection model, trained on a diverse dataset encompassing motorcycles, cars, buses, and trucks within the traffic landscape of Danang City, this research meticulously assesses the performance of five distinct object-tracking methods. Analysis highlights BoTSORT and ByteTrack as the top-performing methods, demonstrating their superior ability to accurately maintain object identities across consecutive frames. This strong performance underscores the effectiveness of their data association strategies for traffic monitoring scenarios. Our findings contribute valuable insights for developers and practitioners, aiding in the selection of the most suitable object-tracking methods tailored to specific traffic monitoring requirements.

In the scope of further development, we propose solutions to enhance object recognition capabilities, including the application of automatic Region of Interest (ROI) detection methods and integration with Optical Character Recognition (OCR) technology to read and process information from license plates. This aims to support traffic monitoring and violation processing efforts.

Acknowledgment. This research is funded by The Murata Science Foundation and The University of Danang - University of Science and Technology, code number of Project: T2022-02-03MSF.

REFERENCES

[1] V. Nam, "Summary of number of vehicles delivered nationwide", *http://www.vr.org.vn*, 2022. [Online]. Available: http://www.vr.org.vn/thong-ke/Pages/tong-hop-so-lieu-phuong-tien-giao-thong-trong-ca-nuoc.aspx [Accessed August 09, 2023].

[2] V. Mandal and Y. Adu-Gyamfi, "Object detection and tracking algorithms for vehicle counting: a comparative analysis", *Journal of Big Data Analytics in Transportation*, vol. 2, pp. 251-261, 2020.

[3] S. Kumar, P. Sharma, and N. Pal, "Object tracking and counting in a zone using YOLOv4, DeepSORT and TensorFlow", *in Proceeding of 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, 2021, pp. 1017-1022.

[4] H. H. Hung, N. V. Phu, and N. Tuong, "Red Light and Wrong Parking Violation Detection System Based on Deep Learning", *The University of Danang - Journal of Science and Technology*, vol. 18, no. 5, 2020, pp. 101-105.

[5] H. Yang and S. Qu, "Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition*", Wiley Online Library*, vol. 12, no. 1, pp. 75-85, 2018.

[6] S. Li, F. Chang, C. Liu, and N. Li, "Vehicle counting and traffic flow parameter estimation for dense traffic scenes*", Wiley Online Library*, vol. 14, no. 12, pp. 1517-1523, 2020.

[7] D. P. Mien, T. T. Vu, and N. V. Si, "Estimating Traffic Density in Uncertain Environment: A Case Study of Danang, Vietnam", *The University of Danang - Journal of Science and Technology*, vol. 21, no. 6.2, pp. 33-38, 2023.

[8] K. T. Minh, Q. V. Dinh, T. D. Nguyen, and T. N. Nhut, "Vehicle Counting on Vietnamese Street", *in Proceeding 2023 IEEE Statistical Signal Processing Workshop (SSP),* IEEE, 2023, pp. 160--164.

[9] S. Neamah and A. Karim, "Real-time Traffic Monitoring System Based on Deep Learning and YOLOv8", *Aro-the scientific journal of koya university*, vol. 11, no. 2, pp. 137-150, 2023.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi "You only look once: Unified, real-time object detection", *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Elsevier, 2016, pp. 779-788.

[11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector", *in Proceeding Computer Vision--ECCV 2016: 14th European Conference, Amsterdam, The Netherland*s, October 11--14, 2016, Proceedings, Part I 14, Springer, 2016, pp. 21-37.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik "Rich feature hierarchies for accurate object detection and semantic segmentation", *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580-587.

[13] A. Yilmaz, O. Javed, and M. Shah "Object tracking: A survey", *ACM computing surveys (CSUR)*, vol. 38, no. 2, pp. 13-es, 2006.

[14] Y. Zhang *et al.,* "Bytetrack: Multi-object tracking by associating every detection box", *in Proceeding of European Conference on Computer Vision*, Springer, 2022, pp. 1-21.

[15] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani "Observation-centric sort: Rethinking sort for robust multi-object tracking", *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9686-9696.

[16] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani, "Deep OC-Sort: Multi-Pedestrian Tracking by Adaptive Re-Identification", *2023 in Proceeding of IEEE International Conference on Image Processing (ICIP),* Kuala Lumpur, Malaysia, 2023, pp. 3025-3029, doi: 10.1109/ICIP49359.2023. 10222576.

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

[18] Y. Du *et al.,* "StrongSORT: Make DeepSORT Great Again", *in IEEE Transactions on Multimedia*, vol. 25, pp. 8725-8737, 2023, doi: 10.1109/TMM.2023.3240881.

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

[20] P. Sun, J. Cao, Y. Jiang, Z. Yuan, S. Bai, K. Kitani, and P. Luo, "Da Aharon track: Multi-object tracking in uniform appearance and diverse motion", *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 20993-21002.

[21] W. Kang, C. Xie, J. Yao, L. Xuan, and G. Liu, "Online Multiple Object Tracking with Recurrent Neural Networks and Appearance Model", *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering, and Informatics (CISP-BMEI),* Chengdu, China, 2020, pp. 34-38, doi: 10.1109/CISP-BMEI51763.2020. 9263623.

[22] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi "Performance measures and a data set for multi-target, multi-camera tracking", *in European conference on computer vision*, 2016, pp. 17-35.

[23] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe, "Hota: A higher order metric for evaluating multi-object tracking*", International Journal of Computer Vision*, vol. 129, pp. 548-578, 2021.