Applied Computer Science, vol. 20, no. 1, pp. 72–84
doi: 10.35784/acs-2024-05

Submitted: 2024-01-09 | Revised: 2024-03-18 | Accepted: 2024-03-23

Keywords: object detection, tracking by detection, pedestrian tracking, YOLOv8, deep SORT

Ghania ZIDANI [0000-0002-1338-3296]*, *Djalal DJARAH* **, *Abdslam BENMAKHLOUF* **, *Laid KHETTACHE* **

OPTIMIZING PEDESTRIAN TRACKING FOR ROBUST PERCEPTION WITH YOLOV8 AND DEEPSORT

Abstract

Multi-object tracking is a crucial aspect of perception in the area of computer vision, widely used in autonomous driving, behavior recognition, and other areas. The complex and dynamic nature of environments, the ever-changing visual features of people, and the frequent appearance of occlusion interactions all impose limitations on the efficacy of existing pedestrian tracking algorithms. This results in suboptimal tracking precision and stability. As a solution, this article proposes an integrated detector-tracker framework for pedestrian tracking. The framework includes a pedestrian object detector that utilizes the YOLOv8 network, which is regarded as the latest state-of-the-art detector, that has been established. This detector provides an ideal detection base to address limitations. Through the combination of YOLOv8 and the DeepSort tracking algorithm, we have improved the ability to track pedestrians in dynamic scenarios. After conducting experiments on publicly available datasets such as MOT17 and MOT20, a clear improvement in accuracy and consistency was demonstrated, with MOTA scores of 63.82 and 58.95, and HOTA scores of 43.15 and 41.36, respectively. Our research highlights the significance of optimizing object detection to unleash the potential of tracking for critical applications like autonomous driving.

1. INTRODUCTION

Multi-object tracking (MOT) refers to the detection and identification of the trajectories of multiple targets in a video sequence, such as pedestrians, vehicles, animals, drones, etc. (Yu et al., 2016; Xu et al., 2019; Ciaparrone et al., 2020). Different targets are assigned unique identifiers to enable trajectory prediction, accurate search and other subsequent processing. Multi-object tracking is an important technology in the field of computer vision, widely used in autonomous driving, intelligent video surveillance, behavior recognition and other applications (Kamal et al., 2020; Ess et al., 2010).

Multi-object tracking not only faces the challenges of occlusion, deformation, motion blur, crowds, scale and illumination changes already present in simple object tracking, but

* University of Mostefa Ben Boulaid, Department of Pharmacy, Algeria, g.zidani@univ-batna2.dz
** University of Kasdi Merbah, Department of Electrical Engineering, Algeria

also complex problems such as trajectory initiation and termination, and mutual interference between similar targets (Behrendt et al., 2017). Multi-object tracking therefore remains a challenging area of research in image processing, attracting the sustained interest of many researchers (Bewley et al., 2016; Bochinski et al., 2017; Zhang et al., 2022; Zeng et al., 2022). Visual object tracking has developed particularly over the last ten years. Initial classical methods such as particle filters (Okuma et al., 2004) and Mean shift (Cheng, 1995) were of limited accuracy and focused on tracking simple objects, struggling to meet the demands of complex scenes. In recent years, the rapid development of deep learning has improved the accuracy of these methods.

Upon examining current methods, it is evident that tracking pedestrians in intricate settings remains a present-day research challenge. The frequent obstructions encountered during tracking hinder the precise location of objects. Furthermore, visually similar targets complicate the preservation of their unique identifiers. Interactions between objects may also lead to deviations in the tracking frame.

To overcome these limitations, the authors suggest an integrated detector-tracker framework that is specifically designed for pedestrian tracking in autonomous vehicles. The key contributions of this paper are as follows:

- The utilization of YOLOv8 as a detector is crucial for real-time detection due to its faster speed compared to its predecessors YOLOv5x/m/l/s, while still maintaining a high level of precision.
- To evaluate the effectiveness of multi-object tracking algorithms by detection, namely SORT and Deep-SORT, a performance assessment is conducted.
- In order to evaluate the robustness of our method in handling occlusions and its capability to track pedestrians in real-time scenarios involving obstructions caused by objects or other individuals, we carried out experiments using the MOT17 and MOT20 public benchmarks.

The current paper is organized as follows: Section 2 provides a review of similar works, Section 3 presents a detailed explanation of the fundamental algorithms used, Section 4 presents the results and discussion, and Section 5 concludes the discussion.

2. RELATED WORK

The extensive research and development work that accompanies deep learning applications has generated a significant amount of interest. Deep learning-based methods have been applied to various tasks, including object detection and tracking (Girshick et al., 2014; Wojke et al., 2017).

This project involves establishing mechanisms for object detection and tracking, and as such, the current documentation on this topic is addressed in this context. Initially, the object detection process based on machine learning can be classified into two categories, You Only Look Once (YOLO) (Redmon et al., 2016) and SSD (single shot multibox detector) (Liu et al., 2016) Both approaches treat detection as a regression problem and operate as single-stage networks.

Conversely, algorithms such as Region-Based CNN (RCNN) (Abbas & Singh, 2018) first establish the region of interest before classifying it. The R-CNN model implements a selective search algorithm to determine the number of candidates for object-bounding

regions, which are then used as features in a CNN acting as a feature extractor. To determine the presence of an object in a given region, an SVM (Support Vector Machine) is employed using the extracted features. Even though the R-CNN model is proficient in detecting objects in different scenarios, it demands an extensive period of training and has a restricted detection speed (Mao et al., 2019).

Object tracking, as per (Luo et al., 2021), can be classified into two categories: single object tracking (SOT) and multiple object tracking (MOT). The tracking of multiple objects (MOT) is an active field of research in computer vision. Recently, two main approaches have been the focus of research: tracking by detection (Bergmann et al., 2019; Pang et al., 2020; Peng et al., 2020; Wang et al., 2020;) and joint detection and tracking (Munjal et al., 2020; Feng et al., 2023; Wang et al., 2021). Joint detection and tracking methods detect and track objects within a single model, utilizing visual appearance to locate objects within images.

(De Rosa & Papa, 2022) are frequently used in this approach. When MOT uses tracking by detection, the framework process is as follows: First, the algorithm identifies objects in each video sequence frame, separates them using bounding boxes, and finds all objects in the frame. The problem is then transformed into a correlation problem between previous images and objects in the current image. It construct a similarity matrix using indicators such as intersection and union (IoU) and appearance features, and then use algorithms such as Hungarian (Kuhn, 1955) algorithm for analysis. The efficiency of this tracking algorithm depends on the performance of its object detection network.

The most widely used recognition network is the YOLO series, which includes YOLOv5 (Zhang et al., 2022) and YOLOv8 (Jocher et al., 2023). The most commonly used tracking algorithms are MOTDT (Chen et al., 2018), SORT (Bewley et al., 2016) and DeepSORT (Wojke et al. 2017). SORT uses Kalman (1960) to predict the location of a target and then associates the result of this prediction with detections from an object detection network, such as YOLO, using the Hungarian matching algorithm. However, due to the variability of target movements and frequent occlusions in real-world scenarios, SORT can lead to a large number of identity changes. For this reason, the author implemented features such as cascade matching and developed DeepSORT, which offers better performance. The techniques of object tracking have been employed in a wide range of fields, including the tracking of pedestrians (Sun et al., 2021).

3. METHODOLOGY

The purpose of this task is to meticulously track pedestrians in a video, which requires assigning unique identifiers to individuals and corresponding tracks that remain consistent throughout the entire sequence of tracks. By achieving a flawless tracking result, the authors can ensure that the movements of pedestrians are precisely tracked and analyzed.

The algorithm's workflow is illustrated in (Fig. 1). First, the algorithm identifies the individuals in each frame of the video sequence, separates them using bounding boxes, and finds all the individuals in the frame. Each individual within every image is tracked using the DeepSORT algorithm, which is an extension of the Simple Online Realtime Tracking algorithm (SORT). DeepSORT employs appearance descriptors to minimize identity changes, thus enhancing tracking efficiency. In cases that involve predicting temporal or time series data, we utilize the Kalman filtering algorithm.

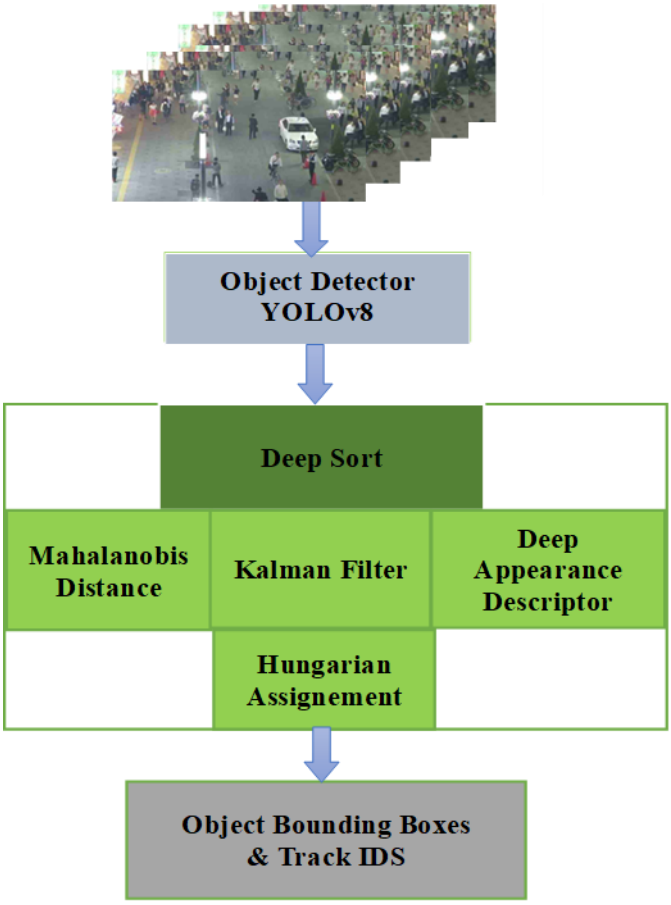


Fig. 1. Flowchart of online tracking-by-detection with YOLOv8 and DeepSort

3.1. Object detection algorithms

After the publication of "You Only Look Once: Unified, Real-Time Object Detection, the YOLOv1 object detection algorithm gained popularity due to its straightforward approach and high speed, as well as its comparatively high mean average precision (mAP) at the time. YOLO's main innovation was formulating object detection as a single-pass regression, using a single neural network to predict both bounding boxes and associated class probabilities.

3.1.1. YOLOv8

The newest version of the YOLO object detection model is YOLOv8. Despite sharing the same architecture as its predecessors (Fig. 2), this latest version features numerous enhancements over previous iterations of YOLO. These include a brand-new neural network architecture that employs both a feature pyramid network (FPN) and a path aggregation network (PAN), as well as a novel labeling tool that simplifies the annotation process. The labeling tool offers multiple useful features such as automatic labeling, labeling shortcuts,

and customizable keyboard shortcuts. The combination of these features makes image annotation for model training an easier process. The FPN method gradually decreases the image's spatial resolution while simultaneously increasing its feature channels, producing characteristic maps that detect objects at various scales and resolutions. On the other hand, the PAN architecture aggregates features through skip connections between different levels of the network. This approach is effective in capturing characteristics at different scales and resolutions, which is crucial for accurately detecting various objects. These techniques have been cited in reference (Treven et al., 2023).

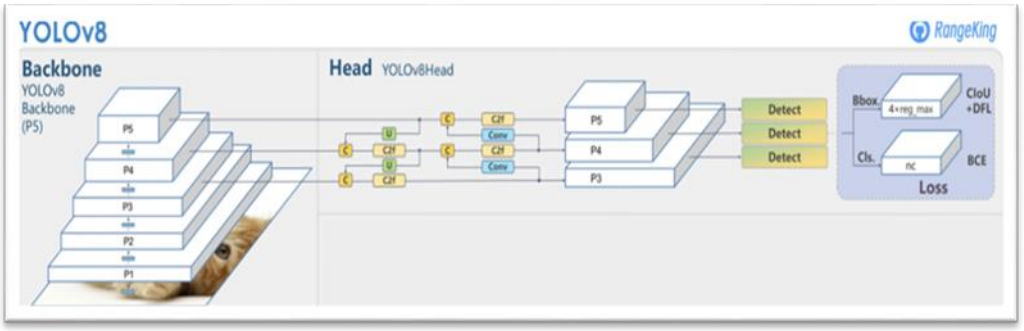
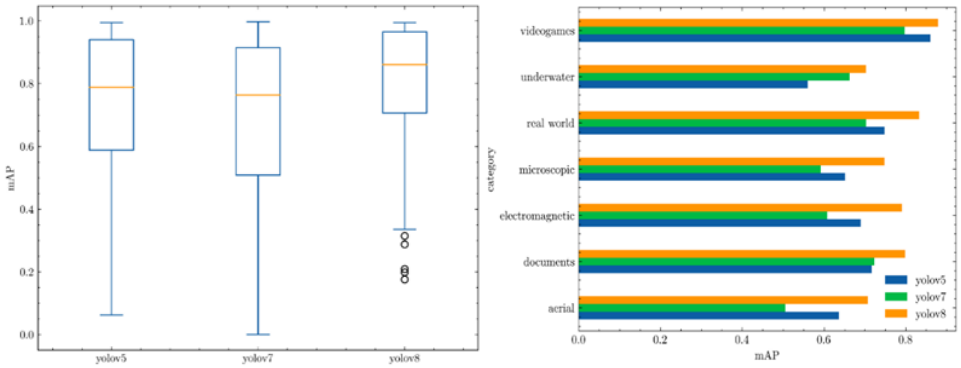


Fig. 2. YOLOv8 Architecture (Solawetz & Francesco, 2023)

3.1.2. YOLOv8 vs YOLOv5

The rationale behind comparing YOLOv8 and YOLOv5, rather than other iterations of YOLO, stems from their closely matched performance and metrics. Nevertheless, YOLOv8 outshines YOLOv5 in numerous aspects, particularly in showcasing a heightened mAP, as evidenced in Figure 3a. This advancement also implies that YOLOv8 features fewer outliers, as demonstrated by Figure 3b. Moreover, it is evident that YOLOv8 yields comparable, if not superior, outcomes to those of YOLOv5. Regarded as a linear assignment dilemma, this represents a pivotal stage in resolving the tracking problem.



(a) YOLOs average *mAP@.50* against RF100 categories **(b) YOLOs mAP@.50 against RF100**

Fig. 3. YOLOv8 vs Previous Versions (Solawetz & Francesco, 2023)

3.2. Objects tracker

3.2.1. Sort

To keep track of objects in video footage in real-time, the Simple Online and Real-Time Tracking (SORT) method is frequently used. This method combines a Kalman filter with a Murken assignment algorithm to determine object positions and velocities and then link them throughout frames. By decreasing the video measurement noise to predict the object's positions, the Kalman filter is useful, while the assignment algorithm solves the association issue. SORT is effective in computer vision applications such as surveillance, autonomous driving, and robotics because it can handle complicated scenarios like occlusion and changes in appearance. It is well-known for its precision and efficiency.

3.2.2. Deep-SORT

DeepSORT is an enhancement to the SORT algorithm, designed for object tracking and tracing. To improve tracking in difficult scenarios, it incorporates appearance measures based on pedestrian features, as well as a cascade matching module. The feature extraction network is essential for the accuracy and quality of pedestrian target appearance information. Comprising two primary components, DeepSORT entails the branches of appearance descriptors and motion prediction via the Kalman filter. Spatio-temporal dissimilarity is measured by the Mahalanobis distance, while the cosine distance evaluates appearance similarity. These distances are used for cascade matching, which associates trajectories. In track management, tracks are updated, initialized and deleted.

3.2.3. Data association

The management of data associations (Korepanova et al., 2020; Vijaymeena & Kavitha, 2016) is an integral part of computer vision object tracking, whether it is within the SORT algorithm or in the second step of the Deep-SORT algorithm. This management can be viewed as a problem of linear assignment, the crucial step of solving the tracking problem becomes apparent. Typically, this problem is formulated using a cost matrix, which offers several approaches for creating these matrices using bounding box metrics. These methods are at the core of developing precise and efficient object tracking systems. A comprehensive understanding of these techniques is imperative in the context of designing computer vision systems.

3.3. Evaluation netrics for multiple targets tracking

Given the intricate nature of multi-object tracking, a single standard falls short in grasping its inherent efficiency. Thus, a comprehensive quantitative evaluation of dedicated algorithms requires multiple indicators capturing diverse perspectives. In this context, the Multi-Object Tracking Accuracy (MOTA) Index notably gauges object relevance in video sequences. MOTA (Kasturi et al., 2009), is pivotal in assessing multi-object tracking algorithm efficacy.

$$\text{MOTA} = 1 - \frac{\sum_{t=1}^{|t|} (FN_t + FP_t + IDs_t)}{\sum_{t=1}^{|t|} GT_t} \quad (1)$$

where:

Ground Truth (GT), False Positive (FP), ID switch (IDs), and False Negative (FN).

An important indicator Objects Identification Accuracy (IDF1):

$$IDF1 = \frac{2|IDTP|}{2|IDTP| + |IDFP| + |IDFN|} \quad (2)$$

Distinct from MOTA, IDTP, IDFP, and IDF1 in Equation (2) correspond to true positives, false positives, and false negatives tied to object identity. These convey a nuanced insight into the multi-object tracking algorithm's prowess in identity preservation. While often encompassing operational performance, these measures acknowledge inherent biases in MOTA and IDF1 (Ristani et al., 2016), favoring detection precision. Designers of MOTchallenge introduced innovative metrics like Higher Order Tracking Accuracy (HOTA) to counterbalance. HOTA (Luiten et al., 2021) stands as a pivotal measure among their tailored evaluations for video-based multi-object tracking algorithms.

$$\text{HOTA}_\alpha = \sqrt{\frac{\sum_c A(c)}{|TP| + |FN| + |FP|}} \quad (3)$$

$$A(c) = \frac{|TPA(c)|}{|TPA(c)| + |FNA(c)| + |FPA(c)|} \quad (4)$$

When presented with a pair of complete trajectories, whether they are observed and predicted or not, the True Positive Association (TPA(c)) refers to the accurately predicted segment of the trajectory. False Negative Association (FNA(c)) pertains to the recorded authentic path that was not predicted. Conversely, False Positive Association (FPA(c)) occurs when a negative trajectory is wrongly predicted as a positive trajectory by the model.

4. RESULTS AND DISCUSSION

This document outlines an integrated framework for detection and tracking, with the experiment divided into two distinct groups. The first group combines the YOLOv5 detector with the SORT and DeepSORT trackers to assess the impact of the MOT algorithm. The second group also utilizes the YOLOv8 detector in conjunction with the same SORT and DeepSORT trackers to evaluate the efficacy of the MOT algorithm.

The hardware environment for this experiment consists of an NVIDIA GeForce RTX 3050 graphics card with 8 GB of video memory, an Intel Core i3-12100K processor clocked at 3.30 GHz and 16 GB of RAM memory.

The evaluation was conducted on all combinations of detectors and trackers for sequences 02, 04, 05, 10, 11, and 13 of the MOT17 challenge (Milan et al., 2016), as well as sequences 01, 03, and 05 of the MOT20 challenge (Dendorfer et al., 2020). The parameters of SORT or DeepSORT methods in each combination were meticulously standardized to ensure a fair comparison.

Tab. 1. Tracking results on the mot17 challenge. Comparing the tracking performance of yolov8-sort and yolov5(s/m/l/x)-sort

	MOTA↑	IDF1↑	HOTA↑	MT↑%	ML↓%	FP↓	FN↓	IDs↓
YOLOv5s	39.52	53.42	29.22	18.23	38.22	6655	47689	596
YOLOv5m	39.25	52.02	29.03	18.55	34.17	6719	47895	612
YOLOv5l	41.05	53.81	30.11	21.21	31.22	6708	45654	526
YOLOv5x	41.95	54.35	30.65	22.12	31.12	6632	45658	569
YOLOv8	**44.82**	**56.56**	**32.90**	**22.36**	**30.19**	**5538**	**43258**	**496**

Tab. 2. Tracking results on the mot17 challenge. Comparing the tracking performance of yolov8-deepsort and yolov5(s/m/l/x)-deepsort

	MOTA↑	IDF1↑	HOTA↑	MT↑%	ML↓%	FP↓	FN↓	IDs↓
YOLOv5s	54.70	60.42	35.72	21.58	37.22	6632	45758	556
YOLOv5m	54.36	60.02	35.15	20.75	33.62	6687	46192	569
YOLOv5l	57.32	66.55	38.43	24.98	31.41	6508	43325	491
YOLOv5x	58.18	67.35	39.56	25.09	30.35	5932	42788	455
YOLOv8	**63.82**	**72.56**	**43.15**	**27.36**	**29.02**	**5238**	**40125**	**403**

The results obtained provide compelling evidence that the improvements made to the detector in this document are indeed effective. The combinations of detectors and trackers are thoroughly assessed on the widely recognized MOT17 and MOT20 datasets. Every result from these exhaustive experiments is thoughtfully presented in Tables 1, 2, 3, and 4.

The first chart, Tab.1, demonstrates that the YOLOv8 detector plays an important role in the proposed tracking algorithm, resulting in the most favorable outcomes in terms of MOTA, IDF1, HOTA, and IDS. In comparison to the algorithm that employs the YOLOv5x detector, MOTA increases by 2.7%, IDF1 increases by 2.21%, HOTA increases by 2.25%, and the number of IDS decreases by 100. The second chart, Tab 2, shows that the proposed tracking algorithm, YOLOv8_DeepSORT, achieves the best results in terms of MOTA, IDF1, HOTA, and IDS. Compared to the base algorithm YOLOv8_SORT (as seen in Tab 1). When comparing SORT and DeepSORT algorithms across all Tables 1, 2, 3, and 4, it is evident that the MOTA, IDF1, and HOTA of SORT without feature extraction networks is significantly lower than that of the DeepSORT pedestrian tracking algorithm that employs such networks. Furthermore, the number of IDS is much higher in the SORT algorithm compared to that of DeepSORT. It is clear that the inclusion of feature extraction networks has a considerable impact on the enhancement of tracking.

Tab. 3. Tracking results on the mot20 challenge. Comparing the tracking performance of yolov8-sort and yolov5(s/m/l/x)-sort

	MOTA↑	IDF1↑	HOTA↑	MT↑%	ML↓%	FP↓	FN↓	IDs↓
YOLOv5s	36.76	50.14	28.35	17.20	37.14	7189	48965	718
YOLOv5m	36.52	49.02	27.42	16.28	39.02	7712	49542	775
YOLOv5l	39.12	51.09	29.74	21.13	35.03	6989	46212	613
YOLOv5x	40.78	52.05	30.07	21.56	34.28	6896	45899	589
YOLOv8	**42.98**	**54.32**	**31.98**	**22.02**	**31.26**	**5956**	**43985**	**501**

Tab. 4. Tracking results on the mot20 challenge. Comparing the tracking performance of yolov8-sort and yolov5(s/m/l/x)-DeepSORT

	MOTA↑	IDF1↑	HOTA↑	MT↑%	ML↓%	FP↓	FN↓	IDs↓
YOLOv5s	52.64	57.97	34.39	20.82	36.09	6985	47552	625
YOLOv5m	50.95	56.02	33.52	20.36	37.32	7082	48561	642
YOLOv5l	54.81	63.84	36.59	23.67	33.21	6130	45252	522
YOLOv5x	55.98	65.97	38.76	23.96	32.20	5987	44563	496
YOLOv8	**58.95**	**71.32**	**41.36**	**26.23**	**29.11**	**5396**	**42255**	**410**

The comparison of YOLOv8-DeepSORT's performance on the MOT17 and MOT20 databases yields significant insights. This examination highlights the algorithm's robustness and versatility in different contexts, including variations in lighting, object movements, and crowd density. The MOT17 database demonstrates superior performance with noteworthy increases in MOTA, IDF1, and HOTA by 4.87, 1.24, and 1.79 respectively compared to MOT20. This study underscores the impact of each database's unique characteristics on the algorithm's efficacy and practical applications in real-world tracking scenarios.

Based on Tables 1, 2, 3, and 4, it is evident that YOLOv8-DeepSORT outperforms YOLOv5s/m/l/x-DeepSORT in terms of MOTA, IDF1, and HOTA, indicating a superior tracking accuracy. Overall, YOLOv8-DeepSORT has significantly enhanced tracking precision compared to YOLOv5s/m/l/x-DeepSORT.

5. QUALITATIVE ANALYSIS

Figure 3 exemplifies the proficiency of the proposed algorithm in managing intricate traffic situations, showcasing the tracking. The detector-tracker integrated framework, as evidenced by qualitative test outcomes on real-world data, is capable of executing multi-pedestrian target tracking and preserving robustness, even in demanding environments. This underscores the scientific rigor and practical applicability of the proposed system.

Figure 4a: The lighting in this image appears to be that of an ideal daytime sun, greatly facilitating object detection and tracking. The level of congestion is moderate, with several pedestrians present. The size of the objects varies, with both close and distant pedestrians. Partial occlusions can be observed, caused by vegetation, buildings, and other pedestrians.

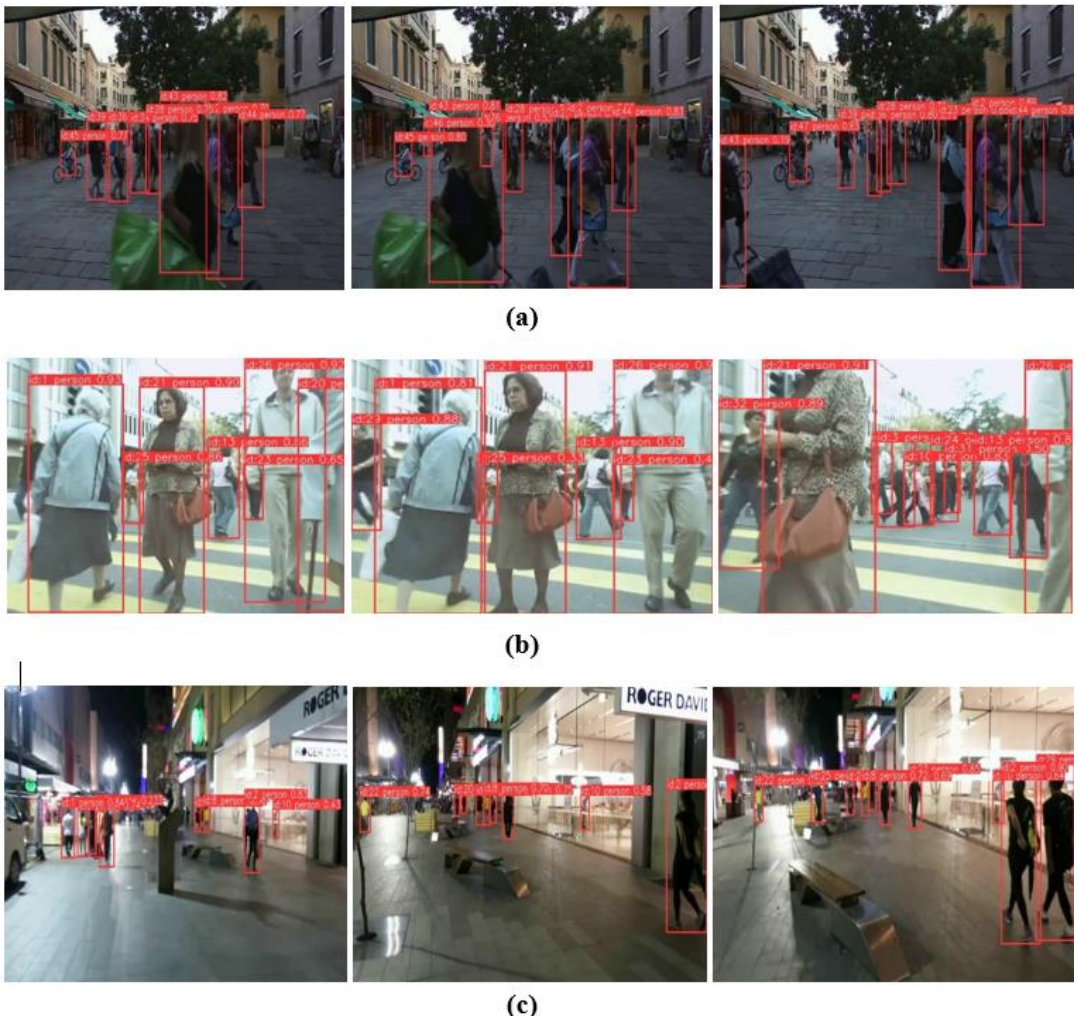


Fig. 4. Tracking results for various datasets

Figure 4b: The lighting conditions in this image appear to be cloudy, which could potentially decrease the contrast. The level of clutter is moderate, with several pedestrians to track. The size of objects varies. There are occasional occlusions caused by pedestrians and urban elements.

Figure 4c: The brightness of this image seems low, indicating a nighttime scene or public lighting. A significant crowd can be observed, with numerous pedestrians moving in various directions. The objects present vary in size. Frequent obstructions are visible, caused by vehicles, vegetation, and urban elements.

In summary, these images demonstrate the performance of the proposed algorithm under a variety of complex traffic conditions, with different levels of illumination, object sizes, clutter and occlusions. Further analysis of these aspects is essential to fully evaluate the effectiveness of the system.

6. CONCLUSION AND OUTLOOK

The problem of tracking pedestrians in complex traffic environments is a challenging one, but this article presents a practical and effective solution. Our approach involves the integration of YOLOv8 into DeepSORT, resulting in an innovative and efficient multi-object tracking algorithm called YOLOv8-DeepSORT. The authors conducted experiments to determine the optimal detector (including YOLOv5s/l/m/x and YOLOv8) and multi-object tracking (SORT and Deep SORT) combinations. The quantitative analysis of public datasets MOT17 and MOT20 revealed that YOLOv8-DeepSORT outperformed the other combinations in terms of tracking accuracy, as measured by evaluation metrics such as MOTA, HOTA, and IDF1. Despite the more challenging target tracking and complex scenes presented in MOT20, the integrated pedestrian detection and tracking framework proved to be more robust in such environments in summary, the algorithm presented in this article offers an efficient method for tracking pedestrians in complex scenes.

The incorporation of advanced sensors such as lidars and radars in autonomous vehicles offers a promising perspective for training a custom object detection and tracking model. By adapting the model, which is based on YOLOv8, to the unique characteristics of these sensors and autonomous driving scenarios, its performance could be significantly improved.

REFERENCES

Abbas, S. M., & Singh, S. (2018). Region-based object detection and classification using faster R-CNN. *4th International Conference on Computational Intelligence & Communication Technology (CICT)* (pp. 1-6). IEEE. https://doi.org/10.1109/ciact.2018.8480413

Behrendt, K., Novak, L., & Botros, R. (2017). A deep learning approach to traffic lights: Detection, tracking, and classification. *IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1370-1377). IEEE. https:/doi.org/10.1109/ICRA.2017.7989163

Bergmann, P., Meinhardt, T., & Leal-Taixe, L. (2019). Tracking without bells and whistles. *ArXiv, abs/1903.05625*. https://doi.org/10.48550/arXiv.1903.05625

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple online and realtime tracking. *IEEE International Conference on Image Processing (ICIP)* (pp. 3464-3468). IEEE. https://doi.org/10.1109/ICIP.2016.7533003

Bochinski, E., Eiselein, V., & Sikora, T. (2017). Highspeed tracking-by-detection without using image information. *14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)* (pp. 1-6). IEEE. https:/doi.org/10.1109/AVSS.2017.8078516

Chen, L., Ai, H., Zhuang, Z., & Shang, C. (2018). Real-time multiple people tracking with deeply learned candidate selection and person re-identification. *IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICME.2018.8486597

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *17*(8), 790-799. https://doi.org/10.1109/34.400568

Ciaparrone, G., Sánchez, F. L., Tabik, S., Troiano, L., Tagliaferri, R., & Herrera, F. (2020). Deep learning in video multi-object tracking: A survey. *Neurocomputing*, *381*, 61–88. https://doi.org/10.1016/j.neucom.2019.11.023

De Rosa, G. H., & Papa, J. P. (2022). Learning to weight similarity measures with Siamese networks: A case study on optimum-path forest. In *Optimum-Path Forest* (pp. 155–173). Elsevier. https://doi.org/10.1016/B978-0-12-822688-9.00015-3

Ess, A., Schindler, K., Leibe, B., & Van Gool, L. (2010). Object detection and tracking for autonomous navigation in dynamic environments. *The International Journal of Robotics Research*, *29*(14), 1707-1725. https://doi.org/10.1177/0278364910365417

Feng, W., Bai, L., Yao, Y., Gan, W., Wu, W., & Ouyang, W. (2023). Similarity- and quality-guided relation learning for joint detection and tracking. *IEEE Transactions on Multimedia*, *26*, 1267-1280. https://doi.org/10.1109/tmm.2023.3279670

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *ArXiv*, *abs*/1311.2524. https://doi.org/10.48550/arXiv.1311.2524

Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics. Retrieved February, 2, 2024 from https://github.com/ultralytics/ultralytics

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, *82*(1), 35-45. https://doi.org/10.1115/1.3662552

Kamal, R., Chemmanam, A. J., Jose, B., Mathews, S., & Varghese, E. (2020). Construction safety surveillance using machine learning. *International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-6). IEEE. https:/doi.org/10.1109/ISNCC49221.2020.9297198

Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., Boonstra, M., Korzhova, V., & Zhang, J. (2009). Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(2), 319-336. https://doi.org/10.1109/TPAMI.2008.57

Korepanova, A. A., Oliseenko, V. D., & Abramov, M. V. (2020). Applicability of similarity coefficients in social circle matching. *2020 XXIII International Conference on Soft Computing and Measurements (SCM)* (pp. 41-43). IEEE. https://doi.org/10.1109/SCM50615.2020.9198782

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, *2*(1-2), 83-97. https://doi.org/10.1002/nav.3800020109

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multiBox detector. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision – ECCV 2016* (Vol. 9905, pp. 21–37). Springer International Publishing. https://doi.org/10.1007/978-3-319-46448-0_2

Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., & Leibe, B. (2021). HOTA: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, *129*, 548-578. https://doi.org/10.1007/s11263-020-01416-9

Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., & Kim, T. K. (2021). Multiple object tracking: A literature review. *Artificial Intelligence*, *293*, 103448. https://doi.org/10.1016/j.artint.2020.103448

Mao, Q. C., Sun, H. M., Liu, Y. B., & Jia, R. S. (2019). Mini-YOLOv3: Real-time object detector for embedded applications. *IEEE Access*, *7*, 133529–133538. https://doi.org/10.1109/ACCESS.2019.2941547

Munjal, B., Aftab, A. R., Amin, S., Brandlmaier, M. D., Tombari, F., & Galasso, F. (2020). Joint detection and tracking in videos with identification features. *Image and Vision Computing*, *100*, 103932. https://doi.org/10.1016/j.imavis.2020.103932

Okuma, K., Taleghani, A., De Freitas, N., Little, J. J., & Lowe, D. G. (2004). A boosted particle filter: Multitarget detection and tracking. In T. Pajdla & J. Matas (Eds.), *Computer Vision—ECCV 2004* (Vol. 3021, pp. 28–39). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-24670-1_3

Pang, B., Li, Y., Zhang, Y., Li, M., & Lu, C. (2020). TubeTK: Adopting tubes to track multi-object in a one-step training model. *ArXiv*, *abs/2006.05683*. https://doi.org/10.48550/arXiv.2006.05683

Peng, J., Wang, C., Wan, F., Wu, Y., Wang, Y., Tai, Y., Wang, C., Li, J., Huang, F., & Fu, Y. (2020). Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer Vision – ECCV 2020* (Vol. 12349, pp. 145–161). Springer International Publishing. https://doi.org/10.1007/978-3-030-58548-8_9

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *ArXiv*, *abs/1506.02640*. https://doi.org/10.48550/arXiv.1506.02640

Ristani, E., Solera, F., Zou, R., Cucchiara, R., & Tomasi, C. (2016). Performance measures and a data set for multi-target, multi-camera tracking. *ArXiv*, *abs/1609.01775*. https://doi.org/10.48550/arXiv.1609.01775

Solawetz, J., & Francesco. (2023, January 11). What is yolov8? The ultimate guide. https://blog.roboflow.com/whats-new-in-yolov8/

Sun, Z., Chen, J., Chao, L., Ruan, W., & Mukherjee, M. (2021). A survey of multiple pedestrian tracking based on tracking-by-detection framework. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(5), 1819-1833. https://doi.org/10.1109/TCSVT.2020.3009717

Treven, J. R., & Cordova-Esparaza, D. M., Romero-González, J. A. (2023). A Comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning. & Knowledge. Extraction*, *5*(4), 1680-1716. https://doi.org/10.3390/make5040083

Vijaymeena, M., & Kavitha, K. (2016). A survey on similarity measures in text mining. *Machine Learning Applications: An International Journal*, *3*(1), 19-28.

Wang, Y., Kitani, K., & Weng, X. (2021). Joint object detection and multi-object tracking with graph neural networks. *2021 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 13708-13715). https://doi.org/10.1109/icra48506.2021.9561110

Wang, Z., Zheng, L., Liu, Y., Li, Y., & Wang, S. (2020). Towards Real-Time Multi-Object Tracking. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer Vision – ECCV 2020* (Vol. 12356, pp. 107–122). Springer International Publishing. https://doi.org/10.1007/978-3-030-58621-8_7

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

Xu, Y., Ošep, A., Ban, Y., Horaud, R., Leal-Taixé, L., & Alameda-Pineda, X. (2019). How to train your deep multi-object tracker. *ArXiv, abs/1906.06618*. https://doi.org/10.48550/arxiv.1906.06618

Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., & Yan, J. (2016). POI: Multiple object tracking with high performance detection and appearance feature. *ArXiv, abs/1610.06136*. https://doi.org/10.48550/arxiv.1610.06136

Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X., & Wei, Y. (2022). Motr: End-to-end multiple object tracking with transformer. *ArXiv, abs/2105.03247*. https://doi.org/10.48550/arXiv.2105.03247

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., & Wang, X. (2022). Bytetrack: Multi-object tracking by associating every detection box. *ArXiv, abs/2110.06864*. https://doi.org/10.48550/arXiv.2110.06864