Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.1596008

Open Access Journal
e-ISSN: 2619 – 8991

Research Article
Volume 8 - Issue 2: 363-370 / March 2025

A COMPARATIVE ASSESSMENT ON THE NOVEL LONG-TERM REAL-TIME SINGLE OBJECT TRACKING TECHNIQUES USING YOLO-NAS AND YOLO11

Cevahir PARLAK[1]*

[1]*Fenerbahçe University, Faculty of Engineering, Department of Computer Engineering, 34758, İstanbul, Türkiye*

Abstract: This study sheds light on the daunting task of single-object tracking using state-of-the-art BoT-SORT, DeepSORT, SORT, and ByteTrack tracking algorithms with YOLO-NAS and YOLO11 object detectors. Object tracking is a step further of object detection and tries to detect the movement of objects in video files and it has enormous range of real-world application fields. Object tracking also assigns unique identifiers to each tracked object and tries to maintain the identity throughout the entire sequence. Current models can achieve great success in object tracking, however there are still too many obstacles and challenges lying ahead to resolve. YOLO-NAS and YOLO11 are the latest and most used object detection models. YOLO can be combined with different tracking methods such as ByteTrack, BoT-SORT, SORT, and DeepSORT for object tracking. The advantage of YOLO is its extremely fast implementation compared to the other methods. When accompanied by specialized tracking algorithms, YOLO achieves the best scores in object tracking. This study focuses on the implementation of YOLO-NAS and YOLO11 in tracking and results demonstrate that YOLO11 is more accurate and stable with BoT-SORT, however, it is faster using ByteTrack method.

Keywords: Single-Object tracking, YOLO, BoT-SORT, ByteTrack, SORT, DeepSORT

***Corresponding author:** Fenerbahçe University, Faculty of Engineering, Department of Computer Engineering, 34758, İstanbul, Türkiye
E mail: cevahir.parlak@fbu.edu.tr (C. PARLAK)
Cevahir PARLAK https://orcid.org/0000-0002-5500-7379

Received: December 04, 2024
Accepted: January 17, 2025
Published: March 15, 2025

Cite as: Parlak C. 2025. A comparative assessment on the novel long-term real-time single object tracking techniques using YOLO-NAS and YOLO11. BSJ Eng Sci, 8(2): 363-370.

1. Introduction

Object tracking is a very formidable, energetic, and vitally important computer vision field which combines the detection of an object and pursuing its traces across multiple video frames. Broadly speaking, SOT (stands for Single-Object Tracking) together with Multi-Object Tracking (abbreviated as MOT) constitute two major branches of object tracking. SOT technology deals with single objects even if there can be other objects in the environment. MOT tries to detect all different objects, assigns unique identities to each of them, and furthermore tracks them by their identities across the entire sequence. Object tracking harbors two main ideas: Detection of certain objects and Tracking of the Detected Objects with their unique identities. In the first stage of object tracking a detection algorithm is used to identify specified objects and in the second phase, specified objects are pursued by another algorithm using their unique identities. Object tracking becomes increasingly popular and finds a broad range of application areas such as traffic surveillance, security, robotics, sports analytics, automatic driving assistance systems, and medical diagnosis.

Many methods and algorithms have been put forward to advance object tracking in computer vision. Tracking can be done using Kalman filters, Meanshift (Fukunaga and Hostetler, 1975), deep learning-based methods, and particle filters. Optical flow (Black and Anandan, 1993) and Kalman filters are the most popular methods of object tracking. Optical flow tracks the movement of pixels whereas Kalman filter pursues the movement of specified object. Boosting (Grabner et al., 2006) is an Adaboost-based (Freund and Schapire, 1996) old algorithm and quite slow, susceptible to noise and obstacles, and does not stop when the object is lost. Boosting can be used in simple and low-resource cases. Multiple Instance Learning (MIL) (Babenko et al., 2009) method runs on the positive and negative samples and noise-robust, however cannot stop when object is lost. Kernelized Correlation Filters (KCF) (Henriques et al., 2014) is a Boosting and MIL combined method. It is fast, it can stop tracking when the object is lost, however it is difficult to restart the tracking. Tracking, Learning, Detection (TLD) (Kalal et al., 2012) splits tracking operation into tracking, learning, and detecting phases. It is good at object scaling and overlapping, however, it has rather unpredictable behavior and confuses the similar objects instead of the intended object. MedianFlow (Kalal et al., 2010) uses Lucas-Kanade algorithm by tracking the object in forward and backward time directions to calculate the errors of these paths. Median-flow is vulnerable to high-speed objects. Generic Object Tracking

This work is licensed (CC BY-NC 4.0) under Creative Commons Attribution 4.0 International License

Using Regression Networks (GoTurn) (Held et al, 2016) uses convolutional neural networks. The previous and current frames are used as the network input and current location of the object is predicted. It is noise-robust, but accuracy may depend on the dataset it was trained on and therefore prone to overfitting. It may lose the object and track another one at high-speed tracking applications. Minimum Output Sum of Squared Error (MOSSE) (Bolme et al., 2010) tracker uses adaptive correlations of Fourier transformation. It is useful in high-speed tracking but may continue tracking even if the object is totally lost from the screen. Channel and Spatial Reliability Tracker (CSRT) (Lukežič et al., 2017) leverages spatial reliability maps to obtain a wider area for its search and can pursue non-rectangular-shaped objects. It uses Colornames and HOG attributes which derive the oriented gradients and obtain their histograms by (Dalal and Triggs, 2005). It can achieve higher accuracy, but it is slow and also it can produce unexpected results if the object disappears from the frame. Meanshift and CamShift (Continuously Adaptive Meanshift) (Bradski, 1998) are based on the same methods which use color histograms. Meanshift cannot evaluate rotations, translations, and scales whereas CAMShift can. Optical flow tracks pixel movements in time using a vector field. It is based on spatial smoothness and constancy brightness. It can be implemented using Shi-Tomasi, Lucas-Kanade-Tomasi (sparse) or Fanerb*ä*ck (dense) methods. Siamese networks (Bertinetto et al, 2016) have two identical networks to unearth the relevant features of objects to pinpoint the location and appearance of the objects. These feature vectors are compared using some distance methods such as cosine or Euclidean distances. After determining the objects in the first frame, extracted features are used to find the best likely matches in the next frames. Siamese networks need only a single image to track an image enabling fast online learning. They can be robust against translation and occlusion or other light changes. They are also fast enough to be used in real-time tracking. StrongSORT (Du et al., 2023) is proposed to resolve the missing detection and missing association problems of previous tracking methods. StrongSORT also incorporates Gaussian-smoothed interpolation (GSI) and appearance-free link algorithms to alleviate the missing detection problem and to balance the speed-accuracy trade-off. MCITrack (Kang et al., 2024) uses a mamba layer and cross-attention layer as principal components to further exploit the contextual information inside the video streams. Experiments on the LASOT dataset demonstrated strong performance and achieved 76.6% AUC. Another newly introduced tracking algorithm called LoRAT (Lin et al., 2025) facilitates a large ViT (Vision Transformer) which decomposes positional embeddings as shared spatial embeddings and independent embeddings. LoRAT is highly inspired by the success of PEFT (Parameter-Efficient Fine-Tuning) in the transformers and adapts it with a special multilayer perceptron architecture. Thus, obtains less computational complexity while boosting performance. Other newly introduced methods are Simple Online and Realtime Tracking also known as SORT (Bewley et al., 2016), Simple Online and Realtime Tracking with a Deep Association Metric also referred to as DeepSORT (Wojke et al., 2017), ByteTrack (Zhang et al., 2022), and Bag of Tricks for SORT aka BoT-SORT (Aharon et al., 2022) are evaluated in this paper and discussed in the next section.

There are numerous issues that need to be overcome in object tracking including object reidentification, occlusion, and interlacing handling. Object reidentification is the process of reidentifying the object after it disappeared for a while and assigning the same identity to the object. Occlusion handling involves tracking while the object is covered by other objects partially or completely. Interlacing occurs when different objects are identified inside the same bounding box intertwined with one another or distinct parts of the same object are identified as different objects. Fast moving objects, dynamic ambience (complex background, light changes etc.), and detection of similar objects are other difficulties in object tracking applications.

Deep learning techniques and object detection is investigated by Tan et al (Tan et al., 2021). A comprehensive review of object tracking techniques, datasets, and metrics can be found in (Kadam et al., 2024). Soleimanitaleb and Keyvanrad (Soleimanitaleb and Keyvanrad, 2022) surveyed the single object tracking methods, metrics, and datasets.

Şimşek and Tekbaş (Şimşek and Tekbaş, 2024) proposed a YOLO8-based DeepSORT approach utilizing heatmaps to efficiently and adaptively analyze the in-store behaviors of customers. Instead of tracking the full bounding boxes of customers they used only the bounding boxes of feet of customers. Their study achieved 89.16% F1-score.

Havuç et al. (Havuç et al., 2021) proposed a YOLO-based ping-pong playing robotic arm tracking the ping-pong ball. They created their own table-tennis dataset by recording their tennis matches and obtaining videos from YouTube. They included 21000 images in their datasets. They used a specialized camera hardware to detect the fast-moving ping-pong balls. They trained their model with 80% of the data and 20% is used as test data. In their experiments, they used YOLO-tiny model for fast implementation of detecting the fast-moving ball. They showed that YOLO-tiny can successfully track the ping-pong ball and respond accurately to the ball movements.

Atalı and Eyüboğlu (Atalı and Eyüboğlu, 2022) studied tracking on the colorful circular objects with varying diameters using CIE (Commission Internationale de l'éclairage) color format by a mobile robot and compare the results with the HSV color mode. A robot with a constant speed pursuing the detected object from a certain distance using ROS (Robot Operating System) system. They concluded that CIE color provides better results from the noisy images compared to HSV (Hue Saturation Value) color coding system, but HSV is better in response time and image capturing.

2. Materials and Methods

This section summarizes the single object dataset of this study, YOLO11, YOLO-NAS, SORT, DeepSORT, ByteTrack, and BoT-SORT methods. YOLO11 and YOLO-NAS are used for detection part whereas SORT, DeepSORT, ByteTrack, and BoT-SORT are used for tracking the detected objects and assigning unique identities.

2.1. LASOT Dataset

There are too many datasets related to object tracking and, in this study, 3 different image sequences are used from the freely available Large-Scale Single Object Tracking image dataset by Fan and others (LASOT) (Fan et al., 2018). LASOT is an image dataset intended for long-term single object detection and tracking; however, some of the images contain many other objects. The difference in single object datasets is that the camera mostly focuses on a specified target object making it easier to detect. LASOT contains 1550 image sequences with 85 categories and more than 3.87 million frames and various object types including ground-truth bounding-box, visual, and lingual information. In this study, cat, car, and airplane images are selected. The videos that were used in this study almost always contain single objects. The Cat video has 2651 frames, the Car video has 3401 frames, and the Airplane video contains 1567 frames totaling 7619 frames combined. The resolutions are $640x360$, $480x360$, and $1280x720$ for the Car, the Cat, and the Airplane videos, respectively. All videos of this study have been recorded with 25 frame per second video speed. Another newly introduced and more comprehensive dataset is the SOTVerse (Hu et al., 2024) dataset which can be used for more advanced tracking tasks. LASOT is primarily designed for long-term tracking whereas SOTVerse is designed for short-term tracking, however, SOTVerse is gradually removing some of its short-term and single camera constraints.

2.2. YOLO11

Currently, YOLO11 (Jocher and Qiu, 2024) is the latest YOLO product for object detection and tracking. It is fast, efficient, and accurate compared to the previous YOLO versions. YOLO11 comes with novel training methods model architecture to enable different machine vision tasks such as object detection, instance segmentation, pose/keypoint estimation, oriented bounding box object detection (OBB), multithreaded tracking and as well as object classification. It introduces many different models such as nano, small, medium, large, and xlarge models. It has enhanced, stronger neck and backbone design to improve the feature extraction stages. Its architectural design is optimized for training processes to achieve faster implementations. It can reach higher accuracies with smaller number of parameters and less complex structures. YOLO11 also is adaptable to different environments and cloud platforms.

2.3. YOLO-NAS

YOLO-NAS (Neural Architecture Search) (Aharon et al., 2021) is another groundbreaking YOLO model for object detection. YOLO-NAS facilitates AutoNAC (Automated Neural Architecture Construction) to resolve the limitations of older YOLO models. YOLO-NAS uses selective quantization and quantization-aware blocks. AutoNAC is an advanced model optimization technology to obtain the best trade-off between latency, memory consumption, throughput, and accuracy on a specific hardware. It delivers small, medium, and large models. YOLO models may have problems when dealing with small objects or objects that are too close to each other, however, this is no concern for our study since we deal with single object tracking. Contrary to R-CNN family, YOLO is a one-stage detector which allows it to process images faster and also has a very good tradeoff between speed and accuracy which make it very popular and suitable for real-time object detection and tracking applications.

2.4. SORT Tracking Algorithm

SORT algorithm uses Kalman filters for real-time object tracking. İt is quite fast and easy to implement. SORT needs an object detection algorithm such as YOLO or faster R-CNN, uses Kalman filter for object movement estimation, and applies Hungarian Algorithm to associate the objects with the previous frames. SORT has difficulties with occlusions and does not take the visual features of objects while tracking.

2.5. DeepSORT Tracking Algorithm

DeepSORT is an improved version of SORT method to overcome its limitations. DeepSORT uses a deep learning based Reidentification algorithm by evaluating the visual features of tracked objects. DeepSORT algorithm creates a feature vector containing visual features of objects which are extracted with a deep learning model. DeepSORT includes this information with the tracking algorithm to compare them to the objects in the next frames. DeepSORT performs better in occlusion cases and can reidentify objects even after they disappear from the scene, or they are occluded. However, it is costlier than SORT and slower.

2.6. ByteTrack Tracking Algorithm

ByteTrack tracks both the objects with high and low confidence scores. ByteTrack does not ignore low confidence scores instead it evaluates them temporarily making it more powerful for occlusions and dense scenes. ByteTrack unifies the objects with low confidence scores to improve the tracking predictions.

2.7. BoT-SORT Tracking Algorithm

BoT-SORT tries to improve the capabilities of SORT by adding visual-matching techniques. It employs deep learning-based techniques to obtain and uncover the visual features of objects. It adds to the ability to work with the low confidence scores borrowed from ByteTrack algorithm. It uses an advanced Re-Identification method. It is very strong in dense and dynamic fast-moving scenes and also more stable than ByteTrack. BoT-SORT also includes CMC (Camera Motion Compensation) method to compensate for possible camera movements.

As summary, SORT is most suitable for real-time fast applications, DeepSORT and BoT-SORT is used for occlusions and reidentifications for high precision task, ByteTrack is suitable for high and low confidence tasks and fast implementations.

Another difficult and intriguing aspect of object tracking is the performance evaluation of trackers. Various elements and factors need to be taken into consideration to precisely assess the performance of tracking methods including detection confidence scores, missing objects etc. Single object detection metrics are precision, accuracy, Center Location Error, robustness for occlusion and reidentification. For multi object detection, Multi Object Tracking Accuracy (MOTA), Fragmentation (Frag), False Positives, Higher Order Tracking Accuracy (HOTA), False Negatives, Track Completeness (TC), and Multi Object Tracking Precision (MOTP) metrics can be recruited. Frame per second (FPS), latency, Identity Switches (IDS), and accuracy vs speed trade-off are metrics for both single and multi-object tracking.

In this study, 4 different experiments are conducted using YOLO-NAS and YOLO11. YOLO-NAS is used together with SORT and DeepSORT, on the other hand, YOLO11 is used with BoT-SORT and ByteTrack algorithms. For all experiments, the minimum confidence score is set to 0.35, minimum IoU (Intersection of Union) threshold is fixed at 0.5. This study also compares the speed of the trackers. Experiments used YOLO-NAS-medium and YOLO11-medium models pretrained in the COCO (Lin et al., 2014) complex and diverse image dataset with Python 3.10 (Rossum, 2007) and Torch 2.5.0 (Paszke et al., 2019) environment. COCO is a diverse and extensive image dataset contrived for object classification, segmentation, instance segmentation, object tracking, oriented bounding box tracking applications. COCO contains 330,000 mostly annotated pictures, 1.5 million sample objects, 80 different classes, and 250,000 people key points.

3. Results and Discussions

In this part of this manuscript, the outputs of experiments are delineated and presented. The evaluations of the models are run according to the FPS and IDS (identity switches) metrics. Speed is an important key facet in object detection and tracking, particularly for real-time online implementations of the applications. Actually, defining a perfect metric for tracking is an extremely complicated and intimidating issue due to the variety and diversity of the applications. For instance, speed is usually evaluated by frames processed in a second, however, frames in a video sequence can be vastly different from one another. Some frames may have lots of different objects whereas some of them may contain single or no objects to detect at all. In this study, experimentations are done with single object frames, therefore, FPS is a suitable metric for speed evaluations. FPS and IDS are defined as follows (equations 1 and 2):

$$FPS = \frac{FC}{T} \tag{1}$$

$$IDS = \#\{id \mid id_{k+1} \neq id_k, k = 1, 2, 3, \ldots, \mathrm{FC}\} \tag{2}$$

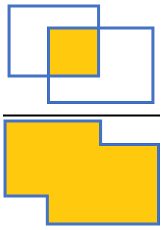
where FC denotes and indicates the total number of frames and T represents total elapsed time for tracking.

The total number of frames term points to the frame images available in the video sequence. Total time is the time spent on tracking and also includes the time spent on detection.

IDS denotes the number of times the algorithm assigns a new tracking identity to the tracked object. Note that tracking identity is different than the class identity. A perfect tracker should follow the same object throughout the entire sequence with the same tracking identity number and object class identity number. Tracking identity may change when objects are lost and then they reappear or are occluded or are covered by some other objects partially or completely. In these cases, trackers need to assign a new identity to the same objects assuming it is a new object. This study also includes incorrect class detections to evaluate the algorithms because as stated earlier on this text, tracking is composed of detection and then tracking. In some cases, the detection algorithm may fail but the tracking algorithm can correctly track the object bounding box as depicted in Figure 1 a and b. In our experiments, this happens mostly in the Cat video. In Figure 1 a, the tracking employing YOLO-NAS and DeepSORT is shown and in Figure 1 b, the tracking employing YOLO11 and BoT-SORT is depicted. As can be seen BoT-SORT is able to maintain the same identity for the tracked cat while DeepSORT needs to assign a new identity. Detectors are confusing the cat with dog most of the time, however, trackers can still be able to track the related bounding box correctly with the same tracking identity. In the Cat video, however, there is great confusion between cat and dog classes. In the Airplane video, the airplane class sometimes is seen as bird, kite, or even person class as illustrated in Figure 2 a and b.

Another important metric is the confidence score which is highly related to precision for the evaluation of tracking methods. Confidence score is the detection correctness probability of the object. The scores are evaluated at different Intersection of Union (IoU) thresholds (e.g., 0.5, 0.6, …, 0.9, 0.95) which means that the label will be evaluated if and only if the confidence of detected object is over the required threshold value. IoU metric is governed by the formula (equation 3):

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} = \quad (3)$$



The Confidence Score denotes the average precision for all given thresholds as follows (equation 4):

$$Conf = mean\ Precision = \frac{1}{\#thresholds}\sum_{t}\frac{tp}{tp + fp + fn} \quad (4)$$

YOLO models provide two types of confidence score as outputs. The first one is the box confidence and typically relates to the probability of how certain an object inside a bounding box is the interested class and it is multiplied by IoU as denoted by the following formula (equation 5):

$$Conf_{box} = P(object) \times IoU \quad (5)$$

The second YOLO confidence score is the class confidence score which expresses the probability of how certain the detected object belongs to a certain class. This computation multiplies the conditional probability of the particular class with IoU and objectness score (the probability that there is an object inside the box) and the final confidence score is defined as below (equation 6) (Redmon, 2016; Kim and Cho, 2021):

$$Conf_{class_i} = P(class_i) \times IoU \quad (6)$$

In this study, confidence scores are also included to further evaluate the tracking processes. Because detectors can detect incorrect classes with extraordinarily strong confidence scores. As seen in Figure 2, the cat can be detected as dog, bird, or even cow with strong confidences. Airplane can be detected as kite, bird, or even person with strong prediction scores. Therefore, high confidence scores do not mean correct detections and tracking. Particularly in the Cat video, detectors give remarkably high confidence scores to the dog class incorrectly. The Car video is an amazingly easy task for detectors and trackers. In the Car video, there is no class confusion, and the car is detected and tracked nearly perfectly with the exact box boundaries throughout the entire video as shown in Figure 3. There are some frames with no detection and tracking at all as depicted in Figure 4. One of the major intriguing obstacles of object detection and tracking techniques is the interlacing of the objects inside one another as illustrated in Figure 5. Detectors and trackers can intertwine different objects inside the same area, or some part of the same object can be detected as another new object.

The Car video results are presented in Table 1 and as stated, the car video is the easiest task for all trackers. Trackers can track the car from start to end of the video without any error either on tracking identity or on class identity. On the other hand, The Cat video is the hardest challenge for all detectors and trackers as pointed by the results in Table 2. In the Cat video, there are too many incorrect class predictions, particularly between cat and dog classes. SORT and DeepSORT incur many IDS changes whereas BoT-SORT and ByteTrack have no or very little IDS changes. Finally, in Table 3, the results of the Airplane video are tabulated. This video is in-between the Cat video and the Car video in terms of difficulty level. The number of incorrect class detections and IDS are fairly lower than the Cat video.

When we investigate these tables, we can conclude that in terms of tracking accuracy and precision BoT-SORT is the winner with 0 IDS value for all experiments. But speed is the factor where ByteTrack shines. DeepSORT and SORT are significantly weak compared to BoT-SORT and ByteTrack in FPS and exclusively in IDS measures. As seen from the following Tables, YOLO11 performs better and faster tracking than YOLO-NAS, however, YOLO-NAS obtains higher confidence scores in detection part of the tracking. Note that confidence score is averaged over all frames for the object interested. As can be seen from these tables, the Car video has the highest confidence score, lowest IDS, and zero incorrect class assignments. We should also note that both YOLO-NAS and YOLO11 are pre-trained on COCO dataset with 80 classes.

4. Conclusions and Future Works

This manuscript evaluates the performance of YOLO-NAS with SORT and DeepSORT tracking, and YOLO11 with BoT-SORT and ByteTrack methods in long-term, real-time, single-object tracking experiments. Results show that object tracking can achieve outstanding jobs, however, there are too many obstacles that need to be overcome. ByteTrack outperforms all others in speed whereas BoT-SORT shows its strength in IDS measure. In the Car video, all tracking algorithms exhibit near-perfect detection and tracking without losing the tracking identity and maintaining very high confidence scores all the way through the video. In terms of number of incorrect classes, on the Cat video, detection of YOLO-NAS is more accurate than YOLO11, on the Airplane video, YOLO11 detects better than YOLO-NAS. However, YOLO-NAS provides better confidence scores in all videos compared to YOLO11. In all experiments, BoT-SORT and ByteTrack outperform SORT and DeepSORT both in IDS and FPS metrics, BoT-SORT and ByteTrack demonstrate equally similar performances. ByteTrack is faster than BoT-SORT, however BoT-SORT is more precise and accurate than ByteTrack in maintaining IDS metric. In the future works, more advanced tracking methods such as MCITrack and LoRAT which include Vision Transformers for object detection can be evaluated in larger datasets and different tracking modalities.

Table 1. Experimental results of object tracking algorithms in the Car video

	#Incorrect Classes	FPS	IDS	Confidence
YOLO11+BoT-SORT	0	4.51	0	93.37
YOLO11+ByteTrack	0	6.22	0	93.37
YOLO-NAS+DeepSORT	0	0.89	0	97.52
YOLO-NAS+SORT	0	0.94	0	97.52

Table 2. Experimental results of object tracking algorithms in the Cat video

	#Incorrect Classes	FPS	IDS	Confidence
YOLO11+BoT-SORT	1318	4.90	0	64.84
YOLO11+ByteTrack	1320	6.38	2	64.86
YOLO-NAS+DeepSORT	906	0.98	14	69.28
YOLO-NAS+SORT	946	0.98	81	69.28

Table 3. Experimental results of object tracking algorithms in the Airplane video

	#Incorrect Classes	FPS	IDS	Confidence
YOLO11+BoT-SORT	152	4.82	0	78.35
YOLO11+ByteTrack	151	5.88	3	78.35
YOLO-NAS+DeepSORT	195	0.95	3	86.37
YOLO-NAS+SORT	195	0.94	6	86.37

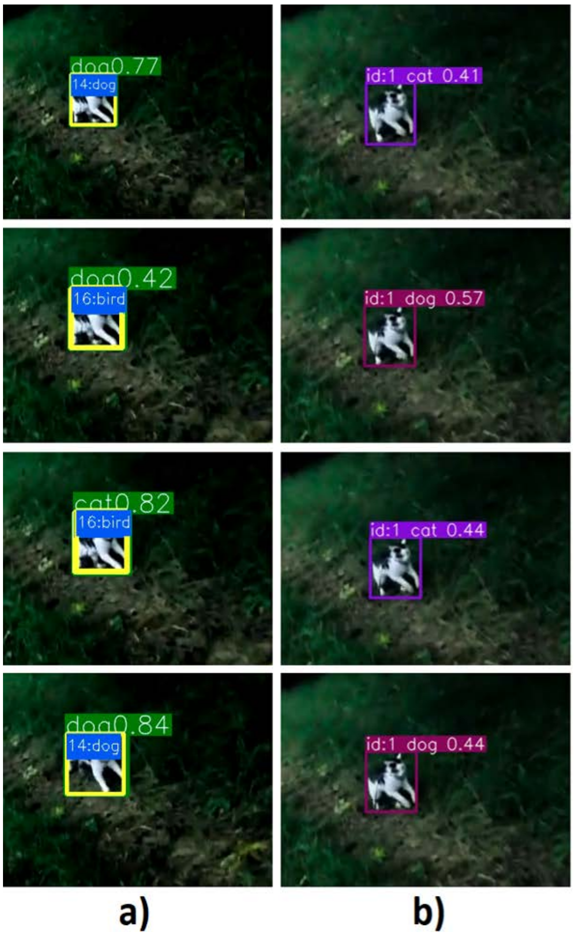


Figure 1. In a) YOLO-NAS and DeepSORT, in b) YOLO11 and BoT-SORT on the Cat video. DeepSORT assigns a new id to the cat when a class change occurs. BoT-SORT can maintain the same id even if an incorrect class detection occurs.

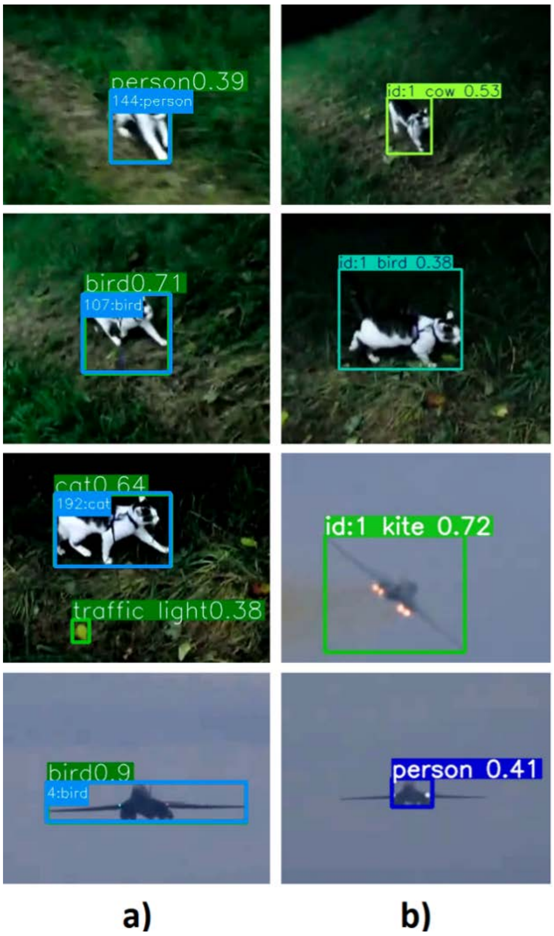


Figure 2. In a) YOLO-NAS and DeepSORT, in b) YOLO11 and BoT-SORT on the Cat video and the Airplane video. The bounding box calculations are remarkably successful and accurate, the tracking is also quite good, however there are too many incorrect class detections both in YOLO-NAS and YOLO11.

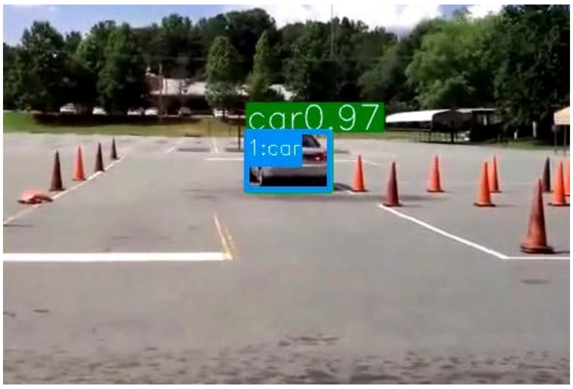


Figure 3. The Car video is the easiest challenge for all detectors and trackers. The car is always detected and tracked perfectly by all algorithms. It is detected as car with very strong confidence scores, and it never loses its tracking identity throughout the entire video.

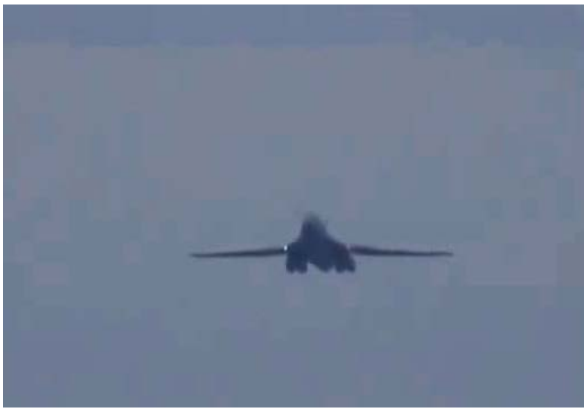


Figure 4. Object detectors and trackers sometimes fail to produce an output even though the object is quite clear on the scene.

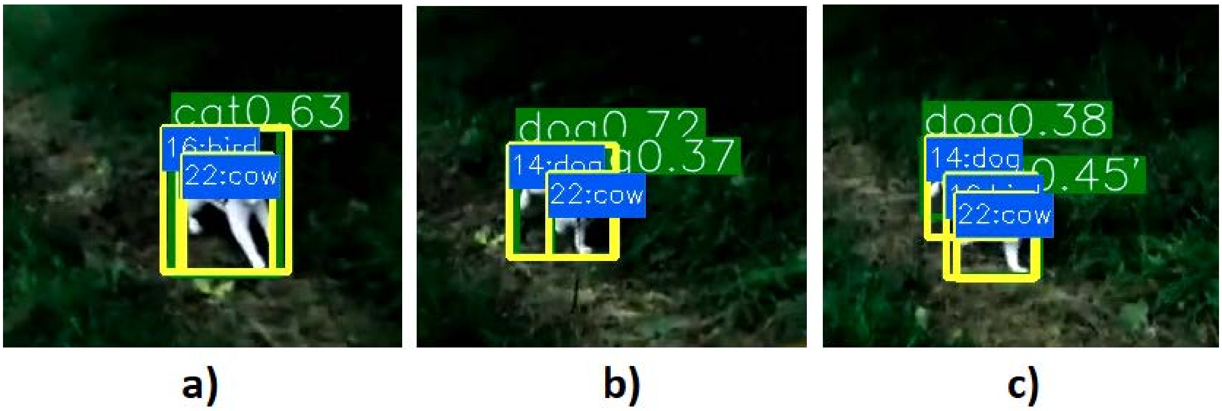


Figure 5. Interlacing different objects is one of the major problems in object detection and tracking. In a) a cow inside a bird, in b) a cow inside a dog, in c) a cow inside a bird inside a dog (YOLO-NAS+DeepSORT)

Author Contributions

The percentages of the author' contributions are presented below. The author reviewed and approved the final version of the manuscript.

	C.P.
C	100
D	100
S	100
DCP	100
DAI	100
L	100
W	100
CR	100
SR	100
PM	100
FA	100

C=Concept, D= design, S= supervision, DCP= data collection and/or processing, DAI= data analysis and/or interpretation, L= literature search, W= writing, CR= critical review, SR= submission and revision, PM= project management, FA= funding acquisition.

Conflict of Interest

The author declares that there is no conflict of interest in this study.

Ethical Consideration

Ethics committee approval was not required for this study because there was no study on animals or humans.

Acknowledgements

There is no financial support for this article.

References

Aharon N, Orfaig R, Bobrovsky BZ. 2022. BoT-SORT: Robust associations multi-pedestrian tracking. arXiv, 2206: 14651.

Aharon S, Dupont L, Masad O, Yurkova K, Fridman L, Lkdci, Khvedchenya E, Rubin R, Bagrov N, Tymchenko B, Keren T, Zhilko A, Deci E. 2021. Supergradients. Github Repository, URL: https://github.com/Deci-AI/super-gradients (accessed date: December 4, 2024).

Atalı G, Eyüboğlu M. 2022. A study on object detection and tracking of a mobile robot using CIE L*a*b* color space. Düzce Uni J Sci Technol, 10(5): 77-90.

Babenko B, Yang MH, Belongie S. 2009. Visual tracking with online multiple instance learning. In: Proceedings of 2009 IEEE Conference on Computer Vision and Pattern Recognition, June 20-25, Miami FL, USA, p: 983-990.

Bertinetto L, Valmadre J, Henriques JF, Vedaldi A, Torr PHS. 2016. Fully-Convolutional Siamese networks for object tracking. In: Proceedings of Computer Vision–ECCV 2016 Workshops: Proceedings, Springer International Publishing Part II 14, October 8-10 and 15-16, Amsterdam, the Netherlands, p: 850-865.

Bewley A, Ge Z, Ott L, Ramos F, Upcroft B. 2016. Simple online and realtime tracking. In: Proceedings of 2016 IEEE International Conference on Image Processing (ICIP), September 20-25, Phoenix AZ, USA, p: 3464-3468.

Black MJ, Anandan P. 1993. A framework for the robust estimation of optical flow. In: Proceedings of 1993 4th International Conference on Computer Vision, May 11-14, Berlin, Germany, p: 231-236.

Bolme DS, Beveridge JR, Draper BA, Lui YM. 2010. Visual object tracking using adaptive correlation filters. In: Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 13-18, San Francisco CA, USA, p: 2544-2550.

Bradski GR. 1998. Computer vision face tracking for use in a perceptual user interface. Intel Technol J, 2.

Dalal N, Triggs B. 2005. Histograms of oriented gradients for human detection. In: Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) 1, June 20-25, San Diego CA, USA, p: 886-893.

Du Y, Zhao Z, Song Y, Zhao Y, Su F, Gong T, Meng H. 2023. Strongsort: Make deepsort great again. IEEE Trans Multimedia, 25: 8725-8737.

Fan H, Bai H, Lin L, Yang F, Chu P, Deng G, ... Ling H. 2021. LaSOT: A high-quality large-scale single object tracking benchmark. Int J Comput Vis, 129: 439-461.

Freund Y, Schapire RE. 1996. Experiments with a new boosting algorithm. In: Proceedings of 13th International Conference on Machine Learning, July 3-6, Bari, Italy, p: 148-156.

Fukunaga K, Hostetler LD. 1975. The Estimation of the gradient of a density function, with applications in pattern-recognition. IERE Trans Inf Theory, 21(1): 32-40.

Grabner H, Grabner M, Bischof H. 2006. Real-time tracking via on-line boosting. In: Proceedings of the British Machine Vision Conference 2006, British Machine Vision Association BMVA, September 4-7, Edinburg, UK, p: 47–56.

Havuç E, Alpak Ş, Çakırel G, Baran MK. 2021. Ping-pong ball tracking through deep learning, Eur J Sci Technol, 27: 629-635.

Held D, Thrun S, Savarese S. 2016. Learning to track at 100 fps with deep regression networks. In: Proceedings of Computer Vision–ECCV 2016: 14th European Conference Part I 14 Springer International Publishing, October 11–14, Amsterdam, the Netherlands, p: 749-765.

Henriques JF, Caseiro R, Martins P, Batista J. 2014. High-speed tracking with kernelized correlation filters. IEEE Trans Pattern Anal Mach Intell, 37(3): 583-596.

Hu S, Zhao X, Huang K. 2024. SOTVerse: A user-defined task space of single object tracking. Int J Comput Vis, 132(3): 872-930.

Jocher G, Qiu J. 2024. Ultralytics YOLO11. URL: https://github.com/ultralytics/ultralytics, (accessed date: December 4, 2024).5

Kadam P, Fang G, Zou JJ. 2024. Object tracking using computer vision: A review. Comput, 13(6): 136.

Kalal Z, Mikolajczyk K, Matas J. 2010. Forward-backward error: Automatic detection of tracking failures. In: Proceedings of 2010 20th International Conference on Pattern Recognition, August 23–26, Istanbul, Türkiye, p: 2756-2759.

Kalal Z, Mikolajczyk K, Matas J. 2012. Tracking-learning-detection. IEEE Trans Pattern Anal Mach Intell, 34(7): 1409-1422.

Kang, B., Chen, X., Lai, S., Liu, Y., Liu, Y., & Wang, D. 2024. Exploring enhanced contextual information for video-level object tracking. arXiv preprint arXiv:2412.11023.

Kim J, Cho J. 2021. A set of single YOLO modalities to detect occluded entities via viewpoint conversion. Appl Sci, 11(13): 6016.

Lin L, Fan H, Zhang Z, Wang Y, Xu Y, Ling H. 2025. Tracking meets LoRA: Faster training, larger model, stronger performance. In: Proceedings of European Conference on Computer Vision, Jan 15-16, London, UK, Springer, Cham, p: 300-318.

Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. 2014. Microsoft COCO: Common objects in context. In: Proceedings of Computer Vision–ECCV 2014: 13th European Conference, Proceedings, Part V 13, Springer International Publishing, September 6-12, Zurich, Switzerland, p: 740-755.

Lukezic A, Vojir T, Cehovin ZL, Matas J, Kristan M. 2017. Discriminative correlation filter with channel and spatial reliability. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, Honolulu HI, USA, p: 6309-6318.

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Kopf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai J, Chintala S. 2019. PyTorch: An imperative style, high-performance deep learning library. In: Proceedings of Advances in Neural Information Processing Systems 32, December 8-14, Vancouver, Canada, p: 8026–8037.

Redmon J. 2016. You Only Look Once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, Las Vegas NV, USA, p: 779-788.

Rossum GV. 2007. Python programming language. In: Proceedings of USENIX Annual Technical Conference 41:1, June 17-22, Santa Clara CA, USA, p: 1-36.

Soleimanitaleb Z, Keyvanrad MA. 2022. Single object tracking: A survey of methods, datasets, and evaluation metrics. arXiv preprint arXiv:2201.13066.

Şimşek M, Tekbaş MK. 2024. Heatmap creation with YOLO-Deep SORT system customized for in-store customer behavior analysis. Commun Fac Sci Univ Ank Series A2-A3 Phys Sci and Eng, 66(1): 118-131.

Tan FG, Yüksel AS, Aydemir E, Ersoy M. 2021. A review on object detection and tracking with deep learning techniques. Eur J Sci Technol, 25: 159-171.

Wojke N, Bewley A, Paulus D. 2017. Simple online and realtime tracking with a deep association metric. In: Proceedings of 2017 IEEE International Conference on Image Processing (ICIP), September 17-20, Beijing, China, pp: 3645-3649.

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. In: Proceedings of European Conference on Computer Vision, Cham: Springer Nature Switzerland, October 23–27, Tel Aviv, Israel, p: 1-21.