Research Article

Rashad N. Razak* and Hadeel N. Abdullah

Improving multi-object detection and tracking with deep learning, DeepSORT, and frame cancellation techniques

https://doi.org/10.1515/eng-2024-0056
received March 26, 2024; accepted May 22, 2024

Abstract: Multi-object detection and tracking is a crucial and extensively researched field in image processing and computer vision. It involves predicting complete tracklets for many objects in a video clip concurrently. This article uses the frame cancellation technique to reduce the computation time required for deep learning and DeepSORT (for any version of the YOLO detector) coupled with DeepSORT algorithm techniques. This novel technique implements a different number of frame cancellations, starting from one frame and continuing until nine frame cancellations, tabling the result of each frame cancellation against the overall system performance for each frame cancellation. The proposed method worked very well; there was a small drop in the average tracking accuracy after the third frame rate cancellation, but the execution time was much faster.

Keywords: Kalman filter, multi-object detection, multi-object tracking, YOLO5 deep learning, data association metric

1 Introduction

Automatically identifying multiple objects in a video and accurately representing them as a set of trajectories is a challenging task known as multi-object detection and tracking (MODT). This issue is of significant importance in computer vision, with practical applications in various areas such as CCTV security cameras, autonomous vehicles, and robot systems equipped with security cameras [1,2]. While our main focus of this study is on pedestrian tracking in video footage, problems with detecting people include (A) changing positions and directions, (B) different clothing styles, (C) different points of view, (D) changing lighting, (E) occlusion, (F) pedestrians of different sizes, and (G) different characteristics of motion, such as silent walking, running, or jumping. Or jump, our approach can easily be adapted to handle various types of objects using a general object detector in deep learning and employing DeepSORT methods [3–5]. Traditionally, MODT activities have mainly relied on the tracking-by-detection paradigm. This approach involves initially detecting objects using an object detector and then applying an object tracking method to establish connections between objects across consecutive frames [6]. Most proposed approaches incorporate a Kalman filter (KF) as a motion module to predict the position of objects of interest in the current frame. In contrast, the emergence of deep learning-based neural networks has led to innovative approaches in object vision-related tasks, including object categorisation, recognition, and tracking. A previous study [7] compared the inference efficiency of several frameworks and suggested an efficient method that optimises the network while utilising only approximately 30% of the hardware capacity compared to other methods, making it suitable for real-time applications. Wu *et al.* [8] aimed to develop an MODT prediction method capable of estimating the potential locations of occluded objects. This method used the velocity and position of objects in previous frames to speculate where occluded objects might be located, considering their visibility in earlier frames. Additionally, their proposed technique employed an efficient version of YOLO version 4 (YOLOv4)-tiny to generate detections, which accelerated the tracking process and enhanced robustness. Incorporating YOLOv4-tiny resulted in a significant increase in tracking speed. Park *et al.* [9] mentioned multiple studies, providing a comprehensive history of MODT over the past few decades, exploring the latest developments in the field, and highlighting promising avenues for future research. While looking at the study from the past 3 years, Li *et al.* [10] mainly talked about the MOT strategies for continuous optimisation in terms of the growth of object detection at each step. This article also talks about the benchmark datasets

*** Corresponding author: Rashad N. Razak,** Electrical Engineering Department, University of Technology, Baghdad, Iraq; Chief Engineering Work at Ministry of Industrial and Minerals, Baghdad, Iraq, e-mail: eee.20.05@grad.uotechnology.edu.iq
Hadeel N. Abdullah: Electrical Engineering Department, University of Technology, Baghdad, Iraq, e-mail: hadeel.n.abdullah@uotechnology.edu.iq

that are widely used and how they can be used in MOT. YOLOv4, a one-stage deep learning detector, is used to generate bounding boxes containing object classes, locations, and confidence values [11]. These bounding boxes are then processed through a simple online and real-time tracking (SORT) system using the deep association metric (DeepSORT) tracker to monitor the movement of targets. The detector's architecture has been improved by incorporating attention mechanisms and reducing parameters, aiming for accurate object detection with minimal graphics processing unit (GPU) memory usage, particularly in scenarios where objects are small or obscured. The effectiveness of cutting-edge networks such as DarkNet has been demonstrated by previous researchers [12], showcasing successful object detection and tracking on a dataset featuring urban objects. While various solutions have been proposed to address the challenge of simultaneously tracking multiple objects, recent breakthroughs in deep learning-based object identification (ID) approaches have led to a focus on detection-based tracking within the domain. Once all detection hypotheses from video data are collected, the tracking problem becomes one of data association – linking similar detections to form a coherent trajectory [12]. The application of this technology in the development of land-based defence weapons has presented challenges in countering these threats. As a result, researchers have pioneered a combat model that deviates progressively from traditional algorithms [13–15].

All MODTs (MODT have YOLO3, YOLO4, YOLO5, YOLO7, and YOLO8 deep learning detector or other version) fully depend on the video frames, which let algorithm need a high cost and complex hardware to implement the MODT algorithm in real time. Every frame needs computation size from the graphics processing unit and central processing unit processor during algorithm calculation especially when there are many objects, have zero or low speed, which they waste the overall systems execution time but the object have the same position or short distance change in object position, so the frame cancellation technique will help the system to overcome this problem because the algorithm will only depend on the KF estimation during frame cancellation period which have less computation time compared with the normal case. More simple explanation if we have a 30 or 25 s video frame rate record or online show the human movement, all know the humans have zero speed when stop or low speed when walk and low or medium speed when run with all these cases (1, 2, … and 9) frame cancellation will be possible to estimate the human movement trajectory within the given video frame rate. From this, we conclude that the frame cancellation technique will be useful for MODT to reduce the execution time and keep the accuracy for low rate cancellation or gain more time but for less accuracy when increasing the frame rate cancellation.

While more advanced tracking-by-detection algorithms have been proposed in recent years, it is evident that certain aspects of the established framework require further refinement for effective implementation in robotic systems. Specifically, we observe that the quality of detections significantly influences detector results, and challenges arise when objects are in crowded scenes or are partially occluded. To address this, we propose a novel approach involving frame cancellation within the multiple object tracking (MOT) framework, a method that, to our knowledge, has not been explored before. This study makes the following contributions to the tracking-by-detection framework. First, to reduce the execution time required for a deep learning-based detector, we introduce frame cancellation within the MOT framework. To our knowledge, we are the first to explore this method, which involves cancelling frames and relying on the KF to predict and estimate object locations. Second, unlike previous versions of MOT that heavily relied on GPU processors, we propose allowing the KF to work during frame cancellation, rather than the deep learning detector. This algorithm helps enhance reliability in complex tracking scenes. Four modes of operation have been implemented based on the type of cancellation. The first mode does not involve frame cancellation; the second mode is applied only when the deep learning detector experiences cancellation; the third mode operates after frame cancellation in the convolutional neural network (CNN) used for feature extraction in the DeepSORT tracker; the fourth and final mode is employed after frame cancellation is implemented in both the deep learning detector and DeepSORT tracker. The simulation demonstrates that, with the help of this innovative model, the importance of accuracy constraints can be reduced after the third frame cancellation in tracking problems. This is achieved while still maintaining state-of-the-art (SOTA) performance, supported by a reliable affinity measure calculated using the KF. Additionally, the execution time shows significant improvement, especially in overall frame rate cancellation. We evaluate the effectiveness of the proposed algorithm on the MOT16 datasets [16] and compare our method with the YOLO5+DeepSORT approaches. The main highlight done in the new algorithm can now implement the MODT with less execution time and keep the accuracy the same when compared with other MODTs (MODT have YOLO5s, YOLO5m, YOLO5L, YOLO7, or YOLO8 detector) for the first and second frame cancellation rate (FCR). The remainder of this article is organised as follows. Section 2 explores the related work of this research. Section 3 provides a detailed theory for the methodology used in the frame cancellation algorithm. In Section 4, a thorough analysis of the proposed algorithm will be conducted, addressing each of the four modes of operation. Section 5 is dedicated to presenting the experiment results, along with tests conducted for the proposed algorithm in different modes of operation. We will

compare these results with those of existing algorithms to evaluate performance, effectiveness, and efficiency. Subsequently, a detailed discussion of these results will be provided. In the final part of this section, we will present our conclusions. Additionally, we will explain the proposed algorithm and discuss further work needed to study and analyze it, with the aim of enhancing the value, robustness, and suitability of the MOT with frame cancellation technique for robotic systems.

2 Related work

2.1 Object detection model

In recent years, CNN-based object identification models have become increasingly preferred in both academic and industrial settings due to their remarkable resilience and efficient performance [17–19]. Object detection methods are typically categorised into two types: one-stage and two-stage object detectors, based on whether they employ a region-based CNN (R-CNN). Specifically, the two-stage object detector requires a specific region and its performance is limited by the generation network component, which affects its operating speed. An extension of R-CNN, known as fast R-CNN [20], addressed this limitation by incorporating an area of interest pooling layer. This layer enables the mapping of feature maps from candidate regions of varying sizes to fixed-size feature maps. Another framework, faster R-CNN [20], utilises a region proposal network based on CNNs to process an image feature map and generate potential regions of interest. In contrast, the one-stage object detector eliminates the area generation network component, resulting in a generally faster method. However, it often exhibits significantly lower accuracy compared to the two-stage detector. Lin *et al.* proposed RetinaNet [21] as an enhancement to the single-stage object detection technique, leveraging the feature pyramid network [22] to achieve improved performance. The YOLO series serves as a prominent example of a single-stage method. YOLOv3 [4], the third iteration, was designed to enhance efficiency and precision through multiscale feature detection, multilabel task integration, and anchor box clustering. Building on YOLOv3, YOLOv4 [23] incorporated the cross-stage partially connected Darknet (CSPDarknet) [24] and the PANet [25] to enhance the overall performance of the model. YOLOX, a novel YOLO network, was introduced in 2021. Nevertheless, the rapid evolution of these detectors has markedly improved their ability to detect and identify objects [26]. YOLOv7 [27] was predominantly built upon the foundations of YOLOv5 [28], encompassing the overall network architecture, configuration file parameters, and the procedures for training, inference, and validation. Notably, the YOLOv7 model introduced the extended efficient layer aggregation network module into its network architecture. This innovative approach utilises convolutional operations to expand the feature space, employs a feature shuffle operation, and ultimately combines output feature maps from different convolutional layers. The aim is to enhance the network's capability to extract a more comprehensive range of picture features.

2.2 Data association with Kalman tracking algorithm

Data association is a computational method used to track objects within a given dataset. This process involves computing similarity measures between trajectories and detection boxes and then matching these entities based on their computed similarities. Feature models and similarity metrics are important components in the field of data association. Among the models used, the motion model plays a key role in predicting the spatial coordinates of objects within video frames. The proposed method uses a predictive approach to track bounding boxes in the current frame by leveraging information from the previous frame. By establishing correspondences between detected boxes in the current frame and predicted boxes from the previous frame, smooth and uninterrupted tracking of objects is achieved. The appearance model focuses on capturing the distinctive characteristics of objects. This ensures that features of the same object in different frames show more significant similarity compared to features of other objects. To measure the degree of similarity between detection and tracking boxes, similarity metrics are used. Commonly used measures include the intersection over union (IoU) metric, Mahalanobis distance, and cosine distance [29]. The SORT algorithm [30] used the faster R-CNN object detector for object detection. This algorithm employs the KF prediction technique to forecast and update motion trajectories for tracking bounding boxes. The integration of these models and metrics contributes to the robust and effective tracking of objects in dynamic datasets. In addition, the IoU metric serves as the matching criterion. The real-time DeepSORT technique, a deep simplified object tracking method built upon the SORT algorithm, introduced several enhancements. It incorporates a cascade matching step before IoU matching, integrates deep appearance features, and extracts these features as an embedded layer using a

reidentification (Re-ID) network [31]. This approach holds promise for mitigating occlusion issues to some extent and reducing identity-flipping occurrences. Further enhancement to the DeepSORT method came from the MODT algorithm [32], which introduced a trajectory scoring mechanism to bolster system reliability, increasing as the trajectory length grows. The joint detection and embedding (JDE) method [33] was improved by incorporating the MODT approach. A key advantage is the integration of detection and embedding networks during the feature extraction phase, striking a balance between computational efficiency and precision. FairMOT [34] enhanced the anchor-free approach for object detection by leveraging the JDE algorithm. It tackled scale invariance issues through the introduction of multilayer feature aggregation. The proposed ByteTrack method [35] presented a straightforward yet effective data association approach. It efficiently distinguishes between high-scoring and low-scoring boxes, allowing the identification of more genuine objects from the latter category. ByteTrack has achieved SOTA performance on the Multiple Object Tracking 2020 (MOT20) dataset [36]. These tracking algorithms collectively underscore the current research trajectory, which predominantly focuses on developing improved data association techniques. It is crucial to emphasise the significance of the object detection module in the tracking by detection TBD algorithm, requiring equal consideration for both the detection capacity and the speed of detection techniques.

2.3 Improvements in MODT

Abdulghafoor *et al.* [15] incorporated techniques that combine principal component analysis with deep learning networks. This integration aimed to maximise the benefits of both approaches, resulting in the development of a real-time intelligent identification and tracking system. In pursuit of creating a practical video surveillance system that efficiently identifies moving people while utilising minimal resources, this study, as detailed in the study by Kim *et al.* [37], adopted a strategy that combined background removal with CNNs. This straightforward framework was designed for detecting and identifying moving objects in outdoor CCTV video footage. Additionally, Alikhanov and Kim [38] examined SOTA MODT trackers. The goal was to fill a gap by providing a comprehensive analysis of their performance in surveillance scenarios. The study aimed to identify the trackers most suitable for an online action detection pipeline. Introducing the Kalman-intersection-over-union (KIOU) tracker in Chen and Shao [39], the article proposed a novel method for multi-object tracking in movies. This approach combined a KF with IoU-based track association techniques. Furthermore, [40] enhanced the components of YOLOv3 and suggested a new military target detection method, denoted as YOLO-G. The study employed a military target dataset featuring armed individuals wielding various weapons. This dataset serves as a testing ground for evaluating various object detection algorithms. When addressing the challenges of occlusion and homogeneous appearance, researchers proposed a depth-enhanced tracking-by-detection framework [40]. This innovative approach utilised a semantic matching strategy and a scene-aware affinity measurement method. Furthermore, to expand the evaluation scope of indoor tracking systems, they introduced a dedicated dataset. Shifting focus, Natarajan *et al.* [41] introduced a vision-based formation control method for unmanned aerial vehicle (UAV) swarms, eliminating the need for an external positioning device. The hierarchical architecture adopts a modified leader–follower strategy, where follower UAVs calculate control inputs to achieve the desired swarm formation. The application of modern deep learning techniques, including YOLOv7 and DeepSORT, enables UAV localisation through vision. In the study by Silano and Iannelli [42], a hybrid visual geometry group 19+ bidirectional long short-term memory network was introduced. This research aimed to identify animals and generate alerts based on their activity. The development of the Swin transformer neck-YOLOX (STN-YOLOX) algorithm as the object detection module and the G-Byte data association method as the tracking module led to the creation of a novel MOT algorithm, known as STN-Track [43]. Testing on UAVDT and VisDrone MOT datasets demonstrated the effectiveness of the STN-Track framework, with SMS-enabled notifications promptly sent to the neighbourhood forest office for quick responses. In another work [44], the primary goal was to present a simulation strategy tailored for a specific UAV task: the optical recognition and tracking of randomly moving objects. All research entirely depends on all video frame if they have new information to update the detector or not, which produce a time delay and computation consumption in case there is no information update or low rate object change, especially for pedestrian object. So our proposed algorithm shows the effectiveness of frame cancellation on MODT system performance related to a different number of cancellation. Liu *et al.* [45] tried to solve these MOT problems by suggesting a new MOT method for cars that are moving in traffic. The author's tracker looks at the vehicle tracks as a single 3D instance of a spatiotemporal route and uses deep learning to figure out how the vehicles are moving from the 3D instances. The GM-YOLO network was developed in this work [46] so that it can give multi-tracker high-quality

detections. The backbone is made up of a coordinate attention mechanism and a weighted bidirectional feature pyramid network structure. The effective receptive field for each feature point is described as a Gaussian distribution. This letter [47] tries to solve the problem of improving detection accuracy and lowering uncertainty by using the different points of view of many agents to create a framework for uncertainty transmission called MOT-CUP. This framework starts by figuring out how unsure collaborative object detection (COD) is. It does this through direct modelling and conformal prediction. Jain *et al.* [48] suggested a new deep fused learning optimised deep fused learning (ODFL) model that can be used to find and follow objects more effectively in video security systems.

3 Methodology

The methodology used in this study employs a frame cancellation approach to evaluate its impact on the overall performance of the MODT system. We utilise one of the latest versions of MODT, namely, YOLOv5+DeepSORT. The proposed algorithm operates sequentially in both the normal and cancellation phases. Figure 1 illustrates the primary algorithm sequence timeline, including the processing methods. Normal phase: this phase begins when the first two frames are introduced to the system. The algorithm locates and calculates the position, speed, and dimensions of the bounding box for each object within these initial frames. Following the completion of the cancellation phase, the normal phase is applied to the subsequent two frames, and the cycle repeats. During the normal phase, a deep learning detector (*e.g.* YOLOv5 or a later version) is used to provide detection results to DeepSORT. This initiates object tracking and determines crucial parameters such as IDs, position, velocity, height, and width of the bounding box for each object.

Cancellation phase: the cancellation phase starts after the normal phase concludes for a specified number of cancellations, denoted as N-frames. In this phase, object parameters positions, speeds, and aspect ratio of the enclosed box required during the normal phase are calculated using the KF, which operates recursively. The algorithm then selects the optimal value between the estimated location and the CNN appearance, leveraging metrics such as Euclidean distance, cosine similarity, and the Hungarian algorithm. Selected parameter values for each object are synchronised at each N-frame cancellation and composited for the corresponding N-frame. Figure 2 illustrates the block diagram of the proposed algorithm. The algorithm initially processes the first two video frames to initialise the system, detecting and tracking objects, and determining key parameters such as position, velocity, width, and height of the bounding box. Subsequent frames are cancelled based on the frame rate cancellation value. Predicted and estimated values (i.e. position, velocity, width, and height of the bounding box) for each detected object are then calculated using the KF. After the frame cancellation phase, the algorithm resumes regular operation, similar to the first two frames. Figure 3 provides a visual representation of the frame sequence throughout the normal and cancellation. The study examines the effects of N-frame cancellation at different rates, ranging from 1 to 9 frames. The purpose of this analysis is to evaluate how frame cancellation impacts the overall performance of the deep learning with DeepSORT algorithm. The KF equations, as described in previous studies [49,50], are used in both the normal and cancellation phases, and their details are summarised in Table 1. All equations remain consistent throughout the phases, except for the observation

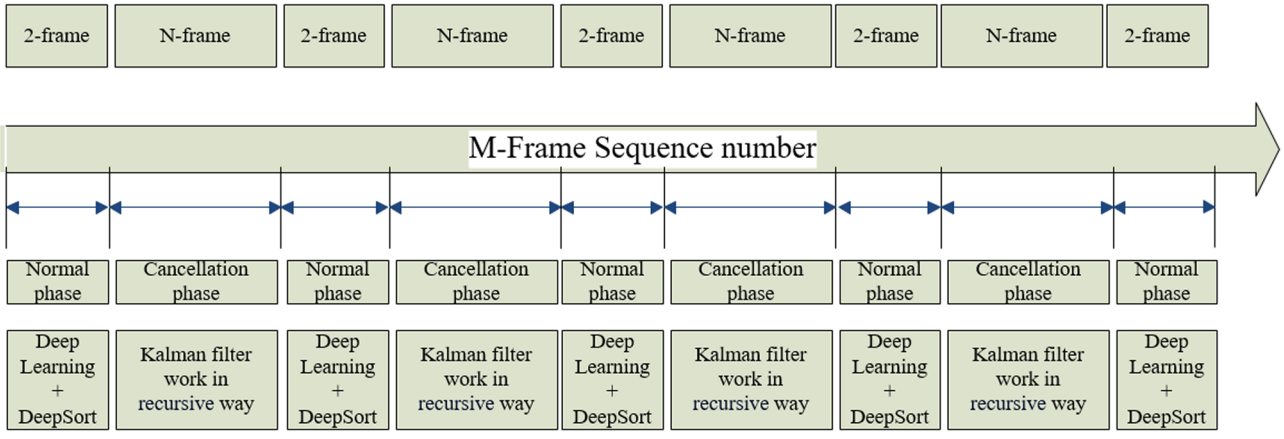

Figure 1: Time sequence for frame cancellation algorithm with processing methods.

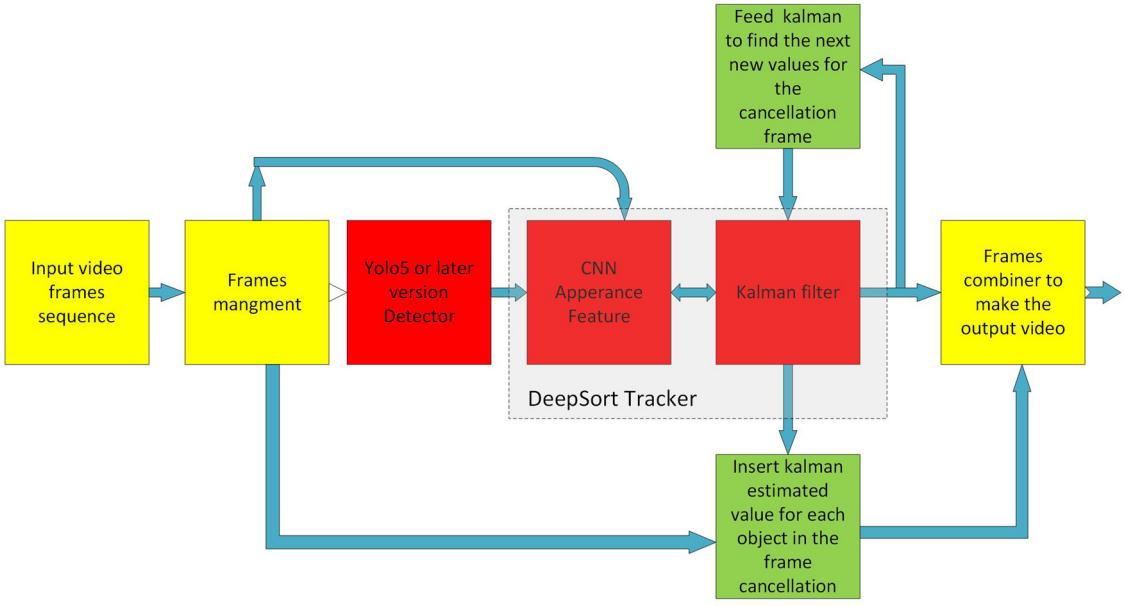

Figure 2: Illustration of the block diagram of the frame cancellation technique implemented on YOLO5 with the DeepSORT algorithm.

measurement equation, which is excluded during the cancellation phase. During this phase, only predicted values are utilised until the cancellation phase is completed. This decision is based on the proximity of the predicted values to the observation values, particularly when observation values are not available.

3.1 Detector

Our prototype framework utilises one of the versions of the YOLO detector (YOLOv5) for deep learning, leveraging artificial intelligence. However, the choice of YOLO detector

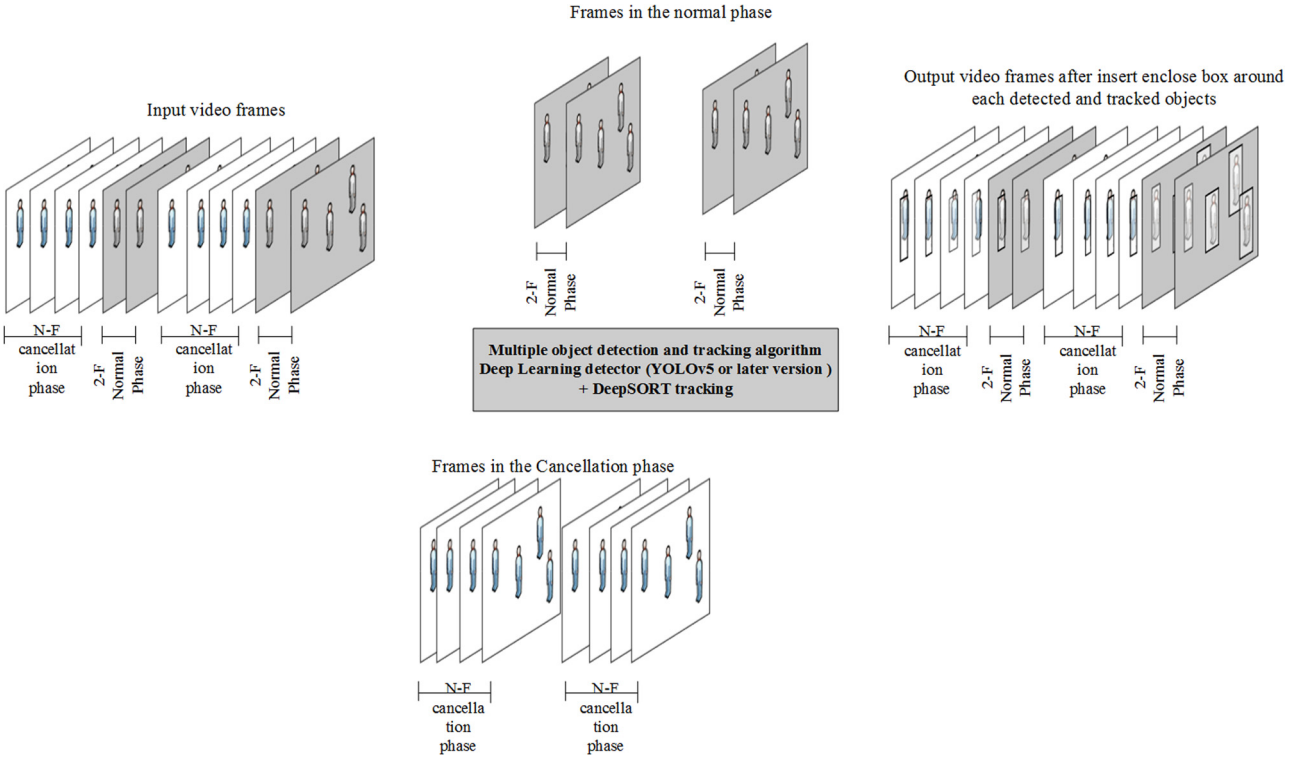

Figure 3: Frame sequence during the normal and cancellation phases.

Table 1: KF equations are used in the normal and cancellation phase

Normal phase KF equation	Cancellation phase KF equation	Description
$\hat{X}_{n+1,n} = F\hat{X}_{n,n} + GU_n$	$\hat{X}_{n+1,n} = F\hat{X}_{n,n} + GU_n$	Predictor equation
$P_{n+1,n} = FP_{n,n}F^T + Q$	$P_{n+1,n} = FP_{n,n}F^T + Q$	Prediction covariance equation
$\hat{X}_{n,n} = \hat{X}_{n,n-1} + K_n(Z_n - H\hat{X}_{n,n-1})$	$\hat{X}_{n,n} = \hat{X}_{n,n-1} + K_n(Z_n - H\hat{X}_{n,n-1})$	Estimated equation
	$P_{n,n} = (I - K_nH)P_{n,n-1}(I - K_nH)^T + K_nR_nK_n^T$	
	$P_{n,n} = (I - K_nH)P_{n,n-1}(I - K_nH)^T + K_nR_nK_n^T$	Corrector equation
$K_n = P_{n,n-1}H^T(HP_{n,n-1}H^T + R_n)^{-1}$	$K_n = P_{n,n-1}H^T(HP_{n,n-1}H^T + R_n)^{-1}$	Kalman gain equation
$Z_n = HX_n$	No measurement is available, using the predicted values instead of the measurement	Measurement equation

X = state vector, Z = output vector, F = state transition matrix, U = input variable, G = control matrix, P = estimated uncertainty, Q = process noise uncertainty, R = measurement uncertainty, H = observation matrix, K = Kalman gain, and n = discrete-time index.

version can be flexible, depending on considerations such as accuracy, complexity, and execution time [51].

3.2 DeepSORT

SORT, a tracking method introduced by previous studies [30,31], is designed for MOT tasks. SORT streamlines time-consuming processes, enhancing task efficiency. By employing CNN-based object trackers, SORT achieves accurate object identification despite its simplicity.

DeepSORT, an extension aiming to minimise ID changes, incorporates additional information into the tracking methodology outlined in SORT [30,31]. Distinguishing itself from SORT, DeepSORT employs multiple techniques to detect objects already under tracking. Specifically, DeepSORT employs two distinct distance measures – Mahalanobis distance [30] and cosine distance between appearance descriptors – to assess new detections against tracked objects. Each bounding box undergoes processing through a CNN trained on a person reidentification dataset. Similar to SORT, DeepSORT employs a KF to determine the state of tracked objects. Consequently, DeepSORT combines both SORT and CNN appearance functionalities. For the implementation of the frame cancellation technique in MODT, we integrated a YOLO detector and DeepSORT tracking. The frame cancellation technique operates in four distinct modes, as elaborated in Section 3.3.

3.3 Mode operation

- **Normal mode (NM)**: no cancellation occurs.
- **Detector cancellation frame mode (DCFM)**: only the frame from the YOLO detector is cancelled.
- **DeepSORT cancellation frame mode (DSCFM)**: the frames from both the detector and DeepSORT blocks are cancelled simultaneously.
- **Detector and DeepSORT cancellation frame mode (D&DSCFM)**: the frame from the detector and deep sort blocks are cancelled at the same time.

Table 2 illustrates the phase operation sequence along with the equations required for each mode.

3.4 Data association metric in the frame cancellation technique

During the normal phase operation, the data association metrics consist of the original equations for Mahalanobis distance and cosine metric, respectively [20]:

Table 2: Phase sequence and equation required in each mode

Cancellation mode type	Phase operation	Equations are given in Table 1
NM	Normal phase	Normal phase KF equations
DCFM	Normal and cancellation phases	Normal phase and cancellation phase KF equation
DSCFM	Normal phase	Normal phase KF equations
D&DSCFM	Normal and cancellation phases	Normal phase and cancellation phase KF equation

$$d^{(1)}(i,j) = (d_j - y_i)^T S_i^{(-1)}(d_j - y_i), \quad (1)$$

$$d^{(2)}(i,j) = \min[1 - r_j^T r_k^{(i)} | r_k^{(i)} \in R_i], \quad (2)$$

where $d^{(1)}(i,j)$ is the Mahalanobis distance, d_i represents the incoming new measurement, y_i denotes the predicted Kalman states, $d^{(2)}(i,j)$ is the cosine distance, r_j is the appearance descriptor, and r_k is the gallery.

These two measures work together to create a comprehensive data association metric. The Mahalanobis distance provides valuable information for short-term predictions in the context of DeepSORT frame cancellation, considering possible object locations based on motion. On the other hand, the cosine distance considers appearance information, aiding in identity recovery after long-term occlusions during detector frame cancellation. The two measures are combined using a weighted sum $c_{i,j}$ to formulate the association problem in the frame cancellation technique:

$$C_{i,j} = \lambda * d^{(1)}(i,j) + (1 - \lambda) * d^{(2)}(i,j). \quad (3)$$

During the cancellation phase, three possible scenarios arise. Detector cancellation frame mode: in this mode, the Mahalanobis distance equals zero due to the absence of measurement values. The prediction value of the KF is used instead of the absent measurement value. This choice is made because the prediction is the nearest available value, resulting in a Mahalanobis distance of zero, as per Equation (1). Consequently, the gate region depends on the long-term value of the cosine metric of the appearances, as specified in (2). DeepSORT cancellation frame mode: In cancellation mode, the Mahalanobis distance remains the same as described in Equation (1). However, the cosine metric relies on the last latch gallery set obtained in the normal phase to determine the appearances, as outlined in Equation (2). Detector and DeepSORT cancellation frame mode: Mahalanobis distance is zero in this mode, and DeepSORT relies on the last latch gallery set obtained from the normal phase to identify the appearances, following the guidance in Equation (2).

4 Proposed algorithm

In this study, we used one of the latest versions of MODT, specifically YOLOv5+DeepSORT, as the foundation for a framework that incorporates frame cancellation, deep learning (detector), and DeepSORT (tracker) techniques. The investigation focused on understanding the impact and improvement of these methods on the overall system performance of MODT. The findings from this research have practical applications in the development of various applications and robotic systems to enhance the performance of MODT algorithms.

4.1 NM

In NM operation, the FCR is set to zero for both the YOLO detector and the DeepSORT tracker. The MODT system algorithm functions as in the original setup without any alterations.

4.2 Detector cancellation frame mode

- The normal phase begins by sending the first and second frames to a deep learning detector (YOLOv5 or another detector version) to determine the object's position in the frame.
- DeepSORT resumes estimating and predicting object parameters, selecting the best values between the detector and Kalman estimation using the Hungarian algorithm.
- The cancellation phase is initiated by cancelling subsequent frames from passing to the detector, depending on the FCR.
- The object's main parameters (position, speed, width, and height of the enclosed box) are estimated based on the KF values.
- Data association metrics are computed as described in Section 3.4.
- The KF updates the main object parameters until the end of the frame cancellation sequence.
- If the video does not conclude, the normal phase is reinitiated and resumes operation; otherwise, the program is stopped.

4.3 DeepSORT cancellation frame mode

The normal phase begins by sending the first and second frames to a deep learning detector to determine the object's position in the frame.

- DeepSORT resumes estimating and predicting object parameters, selecting the best values between the detector and Kalman estimation using the Hungarian algorithm.
- The cancellation phase is initiated by cancelling subsequent frames from the pass to the CNN appearance block in DeepSORT only, based on the FCR value. The CNN relies on the last values obtained in the normal phase

Table 3: Video specification in the Mot16 dataset used to test the proposed algorithm

Name	FPS	Resolution	Length	Density	Camera	Viewpoint	Conditions
MOT16-02	30	1,920 × 1,080	600 (00:20)	29.7	Static	Medium	Cloudy
MOT16-04	30	1,920 × 1,080	1,050 (00:35)	45.3	Static	High	Night
MOT16-05	14	640 × 480	837 (01:00)	8.1	Moving	Medium	Sunny
MOT16-09	30	1,920 × 1,080	525 (00:18)	10.0	Static	Low	Indoor
MOT16-10	30	1,920 × 1,080	654 (00:22)	18.8	Moving	Medium	Night
MOT16-11	30	1,920 × 1,080	900 (00:30)	10.2	Moving	Medium	Indoor
MOT16-13	25	1,920 × 1,080	750 (00:30)	15.3	Moving	High	Sunny
Average	27		760 (00:30.7)	19.62			

and updates the weight when the cancellation phase concludes, after which the normal phase resumes its operation.
- Data association metrics are computed as described in Section 3.4.
- The detector and other MODT blocks function as in the normal phase.
- If the video does not conclude, return to the normal phase; otherwise, stop the program.

4.4 Detector and DeepSORT cancellation frame mode

- Initiate the normal phase by sending the first and second frames to a deep learning detector to determine the positions of the objects in the frame.
- DeepSORT will then continue to estimate and predict object parameters, selecting the best values between the detector and Kalman estimation using the Hungarian algorithm.
- In the cancellation phase, proceed by cancelling subsequent frames from the pass to the detector and CNN appearance in DeepSORT, depending on the FCR value.
- Compute the data association metric according to Section 3.4.
- Estimate the main parameters of the object (position, speed, width, and height of the enclosed box) depending on the values of the KF. The KF will continuously update the main object parameters until the end of the frame cancellation number.
- If the video does not conclude, the program is reinitiated, and the normal phase resumes operation. Otherwise, the program is stopped.

5 Results and discussion

The programme discussed in this study was implemented using Python on MSI Crosshair 15 laptops featuring 11th generation Intel Core i7 processors clocked at 2.3 GHz, equipped with 16 GB of RAM. The performance of the

Table 4: Performance evaluation of normal mode and detection cancellation frame mode on the MOT16 dataset

FCR	IDF1	FP	FN	IDs	MOTA	MOTP	MT (%)	ML
None	52.39	5,375	60,882	432	39.6	80.85	15.45	39.65
1	50.49	7,488	63,009	399	35.8	80.1	17.21	41.01
2	49.19	6,566	61,370	834	37.8	77.4	17.79	39.46
3	48.99	6,912	62,065	892	36.8	76.7	16.05	41.39
4	48.79	7,738	62,789	1024	35.2	76	13.93	42.55
5	48.09	10,772	62,724	1190	32.2	75	12.57	42.36
6	47.49	8,927	64,052	1101	32.9	74.5	11.41	43.71
7	46.49	10,049	64,698	1129	31.3	73.8	9.67	44.68
8	47.99	10,408	65,198	1149	30.5	73.4	7.74	46.62
9	45.09	10,832	66,150	1211	29.2	72.6	6.77	46.23

FCR = frame cancellation rate. IDF1 = the identification metric (IDF1 score). IDF1 = (IDTP)/(2 * IDTP + IDFP + IDFN). GT = the number of ground truth trajectories. MT = most tracked (number of most tracked trajectories). ML = (number of most lost trajectory): The number of trajectories with less than 20 FP = false positive. FN = false negative: Total number of false negatives among all frames. IDs = ID switch number, indicating the number of ID jumps. MOTA = multi-object tracking accuracy: A metric that reflects the tracking accuracy. It has integrated consideration of FN, FP, and IDs. MOTA = $(1 \ \sum_t(\mathrm{FN}t + \mathrm{FP}t + \mathrm{IDS}t))/\sum_t(\mathrm{GT}t)$. MOTP = multi-object tracking precision: A metric that reflects the tracking precision.

Table 5: Execution time table for normal mode and detector cancellation frame mode, YOLO5 time, DeepSORT time, and total detector and DeepSORT time

FCR	AF	ATPT/s	AYT/F	ADST/F	ATD&DS/F
0	760	166.2	0.0808	0.089	0.1698
1	760	159.4	0.0557	0.108	0.1637
2	760	150.98	0.0421	0.113	0.1551
3	760	143.08	0.0335	0.113	0.1465
4	760	136.625	0.0281	0.113	0.1411
5	760	137.81	0.0242	0.117	0.1412
6	760	132.37	0.0208	0.114	0.1348
7	760	130.75	0.0187	0.115	0.1337
8	760	130.17	0.0172	0.117	0.1342
9	760	129.48	0.0155	0.118	0.1335

FCR = frame cancellation rate, AF = average frame, ATPT/s = average total programmed time/s, AYT/F = average YOLO5-time/frame, ADST/F = average deep-SORT time/frame, ATD&DS/F = average time (detection + DeepSORT)/frame.

algorithm was evaluated across four modes of operation using standard MOT metric evaluation. Previous research, specifically studies [51,52], established the CLEAR MOT measures, which are widely adopted for evaluation purposes. The leaderboards in the MOT challenges are determined based on a combination of metrics, including mostly tracked objects (MT), mostly lost objects (ML), IDF1, and the false-positive (FP) rate. It is important to note that the FP rate accounts for items incorrectly identified, contributing to erroneously missed objects, commonly referred to as false negatives.

5.1 Dataset

The dataset employed for algorithm evaluation was MOT16 as given in the study of Milan *et al.* [16], and the specifications of the videos used for testing are detailed in Table 3.

5.2 Normal mode and detector cancellation frame mode results

In the NM and DCFM, the YOLO5 detector and DeepSORT tracker were utilised, incorporating the frame cancellation phase as outlined in Section 3. The results for these modes, tested across various (FCR values on the MOT16 dataset, are presented in Table 4. The outcomes were derived by executing the algorithm on seven videos from the MOT16 dataset (02, 04, 05, 09, 10, 11, and 13) and computing average values. This approach aimed to ensure stability by considering multiple frames from diverse videos, each presenting distinct challenges. The evaluation revealed a minimal decline in MOTA accuracy for all FCRs in this mode of operation, maintaining MOTP for the initial FCR and

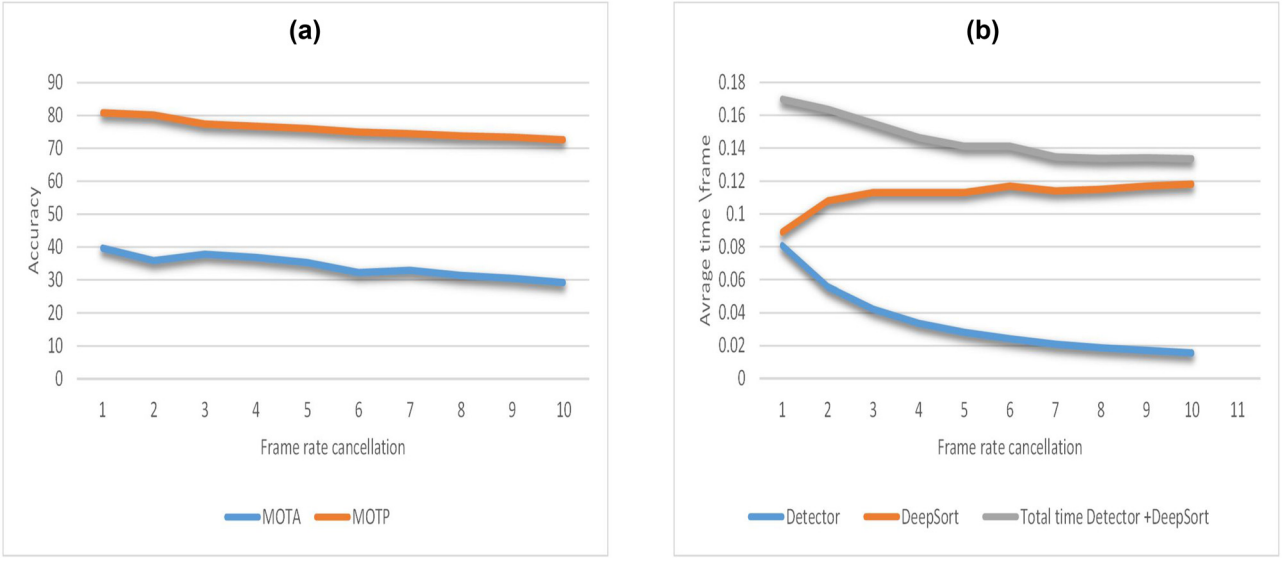

Figure 4: Normal mode and detector cancellation frame mode: (a) accuracy vs FCR (red for multiple object tracking precision MOTP, blue for multiple object tracking accuracy MOTA); (b) average time per frame vs FCR (blue for detection (red for DeepSORT, grey for total detection and DeepSORT average time).

Figure 5: Videos output for different frame cancellation rates for normal mode and detector cancellation frame mode (a: Without cancellation, b, c, d, e, and f for 2, 4, 6, 8, and 9 frame cancellation rates, respectively) using Dataset MOT16: A Benchmark for Multi-Object Tracking [16].

Figure 6: Video outputs for different frame cancellation rates for normal mode and detector cancellation frame mode (a: Without cancellation, b, c, d, e, and f for 2, 4, 6, 8 and 9 frame cancellation rates, respectively) using Dataset MOT16: A Benchmark for Multi-Object Tracking [16].

Table 6: Performance evaluation of normal mode and DeepSORT cancellation frame mode on the MOT16 dataset

FCR	IDF1	FP	FN	IDs	MOTA	MOTP	MT (%)	ML (%)
None	52.39	5,375	60,882	432	39.6	80.85	15.45	39.65
1	55.09	2,020	64,191	413	40.8	83.45	17.21	41.01
2	54.59	2,200	64,407	425	39.9	83.15	16.63	41.20
3	51.99	2,354	64,845	566	38.6	82.55	13.93	41.39
4	51.29	3,081	65,863	594	37	82.15	11.61	43.71
5	50.99	4,102	66,763	724	35.1	81.45	9.28	45.45
6	51.29	5,080	67,504	806	33.5	80.85	8.32	47.58
7	49.39	5,728	68,487	828	32	80.35	5.80	48.74
8	49.19	6,661	69,345	832	30.4	79.85	5.03	48.55
9	48.19	7,659	70,013	848	29.8	77.05	4.06	48.55

Table 7: Execution time table for normal mode and DeepSORT cancellation frame mode, YOLO5-time, DeepSORT time, and total detection and DeepSORT time

FCR	AF	ATPT/s	AYT/F	ADST/F	ATD&DS/F
None	760	166.2	0.0808	0.089	0.1698
1	760	140.299	0.081	0.053	0.134
2	760	131.194	0.082	0.041	0.123
3	760	115.477	0.077	0.031	0.108
4	760	110.781	0.076	0.027	0.103
5	760	122.18	0.085	0.029	0.114
6	760	118.931	0.083	0.026	0.109
7	760	119.402	0.084	0.025	0.109
8	760	117.984	0.083	0.023	0.106
9	760	114.937	0.083	0.022	0.105

subsequently decreasing for subsequent FCR values. Notably, Table 5 highlights a significant enhancement in execution time. The table provides the average execution times for NM and DCFM, encompassing the average times for the YOLO5 Detector, DeepSORT Tracker, and the overall average time for the detector and tracker.

These averages were computed from the seven videos, each varying in frame count (with an average of 760 frames across all videos) and considering different frame cancellation values (ranging from 0 to 9) for evaluation

Figure 4(a) illustrates the relationship between accuracy and frame rate cancellations for NM and DCFM. No significant improvement in accuracy is observed with an increase in FCR for this mode of operation. In Figure 4(b),

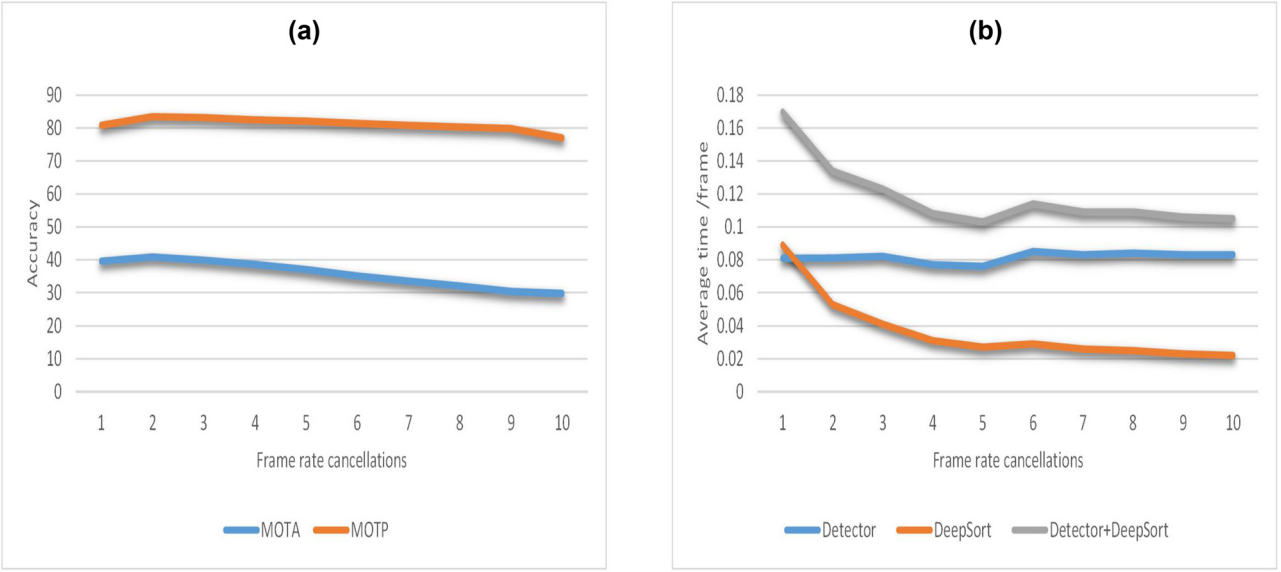

Figure 7: Normal mode and DeepSORT cancellation frame mode: (a) accuracy vs FCR (red for multiple object tracking precision MOTP, blue for multiple object tracking accuracy MOTA and (b) average time/frame vs FCR (blue for detection, red for DeepSORT, grey for total detection and DeepSORT time).

Table 8: Performance of detector and DeepSORT cancellation frame mode with normal mode tested on the MOT16 dataset

FCR	IDF1	FP	FN	IDs	MOTA	MOTP	MT (%)	ML (%)
None	52.39	5,375	60,882	432	39.6	80.85	15.45	39.65
1	55.49	2,829	61,581	419	41.3	81.55	18.96	41.20
2	55.89	3,193	61,714	594	40.7	81.45	18.18	39.26
3	52.69	3,444	62,426	755	39.7	81.15	17.21	41.39
4	52.99	4,071	63,021	891	38.5	80.45	14.12	41.97
5	51.39	4,563	63,553	936	37.5	79.85	12.57	42.55
6	53.49	5,056	63,962	986	36.6	79.15	12.19	43.91
7	51.79	5,579	64,669	984	35.5	78.45	11.22	44.49
8	48.59	14,461	63,447	1,179	28.4	77.65	9.28	44.10
9	50.69	6,809	65,778	1,093	33.3	77.15	7.35	44.87

Table 9: Execution timetable for detector and DeepSORT cancellation frame mode with normal mode: YOLO5-time/frame, DeepSORT time/frame, and total detection and DeepSORT time/frame

FCR	AF	ATPT/s	AYT/F	ADST/F	ATD&DS/F
Non	760	166.2	0.0808	0.089	0.1698
1	760	125.36	0.055	0.053	0.108
2	760	104.38	0.042	0.041	0.083
3	760	90.28	0.033	0.031	0.064
4	760	81.06	0.028	0.027	0.055
5	760	73.28	0.024	0.029	0.053
6	760	68.29	0.020	0.026	0.046
7	760	63.65	0.018	0.025	0.043
8	760	59.53	0.017	0.023	0.04
9	760	57.26	0.015	0.022	0.037

the relationship between the average time per frame and FCR for NM and DCFM is shown. There is a notable improvement in the average detection time for the first five frames of cancellation and the total algorithm time when increasing the FCR. Subsequently, there is consistent improvement for the remaining frame rate cancellations. However, there is no noticeable improvement for the DeepSORT tracker during the increase in FCR for this mode of operation. Figures 5 and 6 display the video output of the proposed NM and DCFM at different frame rate cancellations for videos of MOT16 dataset [16], ranging from zero to nine in two-frame increment steps for the video sequence. In the subsequent study modes, we will focus more on tables and graphic results than video pictures, as they accurately depict the details of the results.

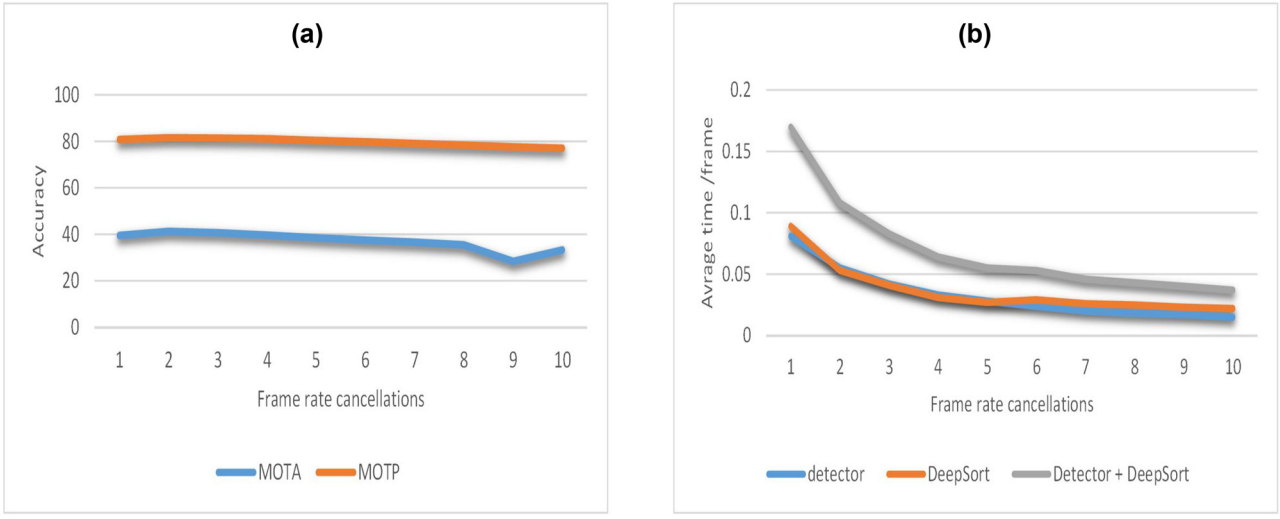

Figure 8: Normal mode and detector and DeepSORT cancellation frame mode: (a) accuracy vs FCR (red for multiple object tracking precision MOTP, blue for multiple object tracking accuracy MOTA) and (b) average time/frame vs FCR (blue for detection, red for DeepSORT, grey for total detection and DeepSORT time).

5.3 Normal mode and DeepSORT cancellation frame mode results

Table 6 presents the results of the first and third modes of operation, namely, the NM and DSCFM techniques, for different FCR tests on the MOT16 dataset. The results were calculated by running seven videos. It indicates a slight improvement in MOTA for the first FCR and consistency for the second FCR, followed by a performance decrease for the subsequent FCRs as the cancellation rate increases. The average time, calculated from seven videos with varying frame numbers (averaging 760 frames), demonstrates a significant improvement in execution time, as shown in Table 7. The table displays the execution timetable for both NM and DSCFM, the average time for the YOLO5-detector, the average time for the DeepSORT tracker, and the total average time for detection and tracker.

Figure 7(a) illustrates the relationship between accuracy and FCR for NM and DSCFM. It shows a noticeable decrease in MOTA accuracy for the first FCR, remaining relatively constant for the next two FCR values, and then experiencing a performance decline for subsequent FCR increases. Figure 7(b) illustrates the relationship between average time per frame and FCR for NM and DSCFM. A significant improvement is observed for the average time of DeepSORT and the total algorithm time with increasing FCR. However, the detector shows no noticeable improvement during this mode.

5.4 Normal mode and detector and DeepSORT cancellation frame mode results

Table 8 shows the results of the first and fourth modes of operation: NM and detector and D&DSCFM for various tests on the dataset. The results were obtained by analysing seven videos, revealing a slight improvement in MOTA accuracy for the first three FCR values and a subsequent decline in performance for higher FCR values. However, Table 9 clearly shows a significant improvement in execution time, presenting the execution timetable for NM and DSCFM.

Figure 8(a) illustrates the correlation between accuracy and frame rate cancellations for NM and D&DSCFM. It demonstrates a minimal improvement in MOTA accuracy for the first three FCR values, followed by a decline for higher FCR values. Figure 8(b) depicts the relationship between average time per frame and FCR for NM and D&DSCFM. It exhibits a significant improvement in the average time for detector, DeepSORT, and the overall average algorithm time as FCR increases.

5.5 Comparison and study results

To compare the proposed algorithm with other algorithms, we select the first FCR from each mode: DCFM, DSCFM, and D&DSCFM. These modes exhibit the best accuracy and execution time performance. While the remaining FCRs show less accuracy improvement, they demonstrate significant enhancements in execution time. For more in-depth comparisons, the aforementioned tables provide additional information for different FCRs. Table 10 shows the tracking results for the MOT16 challenge [53], comparing the tracking performance of YOLOv7-DeepSORT and YOLOv5(S/M/L)-DeepSORT with the frame cancellation technique modes DCFM, DSCFM, and D&DSCFM. DCFM does not affect system accuracy as FCR increases, but it significantly improves the detector and the overall system execution time. Also, the switch ID increases with FCR. Several general features can be observed from the frame cancellation technique in different modes. The NM does not affect the overall system performance.

Table 10: Tracking results on the MOT16 challenge, comparing the tracking performance of YOLOv7-DeepSORT and YOLOv5 (small/medium/large) (S/M/L)-DeepSORT, detector cancellation frame mode DCFM, DeepSORT cancellation frame mode DSCFM, and detector and DeepSORT cancellation frame mode D&DSCFM

Model	MOTA	MOTP	IDF1	IDs	ML (%)	MT (%)	FP	FN
YOLOv5s	39.60	80.85	52.39	432	39.65	15.45	5,375	60,882
YOLOv5L	40.77	81.96	52.43	547	31.92	20.70	7,853	56,990
YOLOv7	40.82	82.01	53.65	514	32.11	20.12	7,940	57,434
D&DSCFM	41.3	81.55	55.49	419	41.20	18.96	2,829	61,581
DCFM	35.8	80.1	50.49	399	41.01	17.21	7,488	63,009
DSCFM	40.8	83.45	55.09	413	41.01	17.21	2,020	64,191

The DCFM and the DSCFM have a minimal impact on system accuracy for the first and second FCR, but they result in significant improvements for DeepSORT and the overall system execution time. Additionally, the switch ID increases with FCR. The last mode, D&DSCFM, shows the most significant improvement in execution time with minimal accuracy improvement for the first FCR. Specifically, for successful implementation of the frame rate cancellation technique, consideration should be given to the camera's condition and environment. The MOT16-05 video, with its low frame rate and resolution, has an impact on the algorithm's performance compared to other videos. The tracking system based on the KF characterises and models moving objects without accounting for the effect of camera movement during video recording. Therefore, modeling the camera movement to obtain its parameters, specifying individual KFs for camera movement to track the camera motion parameters, and using them as composited errors for each object KF can significantly reduce the total error produced from videos with a moving camera. Finally, cloud and night vision videos also affect algorithm performance compared to other sunny static videos. Therefore, cameras with good night vision contribute to the detector's ability to identify all objects within the camera's field of view, simplifying and improving the overall algorithm performance.

6 Conclusion

This study shows a brand-new unified framework that aims to greatly shorten the time it takes to run a program while requiring less from the GPU processor and keeping the main features of MODT. Using the frame cancellation method, the suggested structure works as a variable in the MODT algorithm, shortening the runtime and using the GPU best. A four-mode operation test was used to analyse the MOT16 dataset fully, which led to creating tables and graphs showing key important factors during implementation. One big benefit of our method is that it can reduce MODT complexity and processing time. The runtime is a lot shorter than it was with the original versions of the deep learning and DeepSORT methods. For the first and second frame rate cancellations for D&DSCFM, we obtained 25% and 37% time gains, respectively. The accuracy stayed the same and improved compared to the original YOLO5 algorithm with DeepSORT. You can learn more about the frame cancellation method using different types of YOLO detectors (YOLO7, YOLO8, and YOLO9) along with DeepSORT for tracking. As part of this investigation, metric values will be looked at, and complexity, accuracy, and completion time will be compared. The objective is to find the best version that can be used in robotics and single-board control systems. The frame cancellation method could also be combined with other algorithms, as explained in Section 2.3. This could lead to better algorithms that are better suited to certain uses.

Acknowledgement: The authors are grateful for the reviewer's valuable comments that improved the manuscript.

Funding information: Authors state no funding involved.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and consented to its submission to the journal, reviewed all the results and approved the final version of the manuscript. Conceptualization and methodology: RNR and HNA; software: RNR; validation: RNR and HNA; formal analysis: RNR; investigation: RNR; resources, RNR; data curation: RNR; writing – original draft preparation: RNR; writing – review and editing: HNA; visualization: HNA; supervision: HNA; project administration: HNA.

Conflict of interest: Authors state no conflict of interest.

Data availability statement: Datasets analyzed during the current study are available in the following website links: https://motchallenge.net/data/.

References

[1] Abdulghafoor NH, Abdullah HN. Enhancement performance of multiple objects detection and tracking for realtime and online applications. Int J Intel Eng Syst. 2020;13(6):533–45.
[2] Abdullah HN, Abdulghafoor NH. Automatic objects detection and tracking using FPCP, Blob analysis and Kalman filter. Eng Tech J. 2020;38(2):246–54.
[3] Wu W, Liu H, Li L, Long Y, Wang X, Wang Z, et al. Application of local fully Convolutional Neural Network combined with YOLO v5 algorithm in small target detection of remote sensing image. PLoS One. 2021;16(10):e0259283.
[4] Redmon J, Farhadi A. Yolov3: An incremental improvement. 2018. arXiv:1804.02767.
[5] Rohan A, Rabah M, Kim SH. Convolutional neural network-based real-time object detection and tracking for parrot AR drone 2. IEEE Access. 2019;7:69575–84.
[6] Pereira R, Carvalho G, Garrote L, Nunes UJ. Sort and deep-SORT based multi-object tracking for mobile robotics: Evaluation with new data association metrics. Appl Sci. 2022;12(3):1319.
[7] Hussain J, Prathap BR, Sharma A. An improved and efficient YOLOv4 method for object detection in video streaming. In: Data

Science and Security: Proceedings of IDSCS 2022. Springer; 2022. p. 305–16.
[8] Wu H, Du C, Ji Z, Gao M, He Z. SORT-YM: An algorithm of multi-object tracking with YOLOv4-tiny and motion prediction. Electronics. 2021;10(18):2319.
[9] Park Y, Dang LM, Lee S, Han D, Moon H. Multiple object tracking in deep learning approaches: A survey. Electronics. 2021;10(19):2406.
[10] Li S, Cao Y, Xie X. A review of detection-related multiple object tracking in recent times. In: 2024 26th International Conference on Advanced Communications Technology (ICACT). IEEE; 2024. p. 122–30.
[11] Zheng D. Use of improved deep learning and DeepSORT for vehicle estimation. MSc Thesis, KTH, School of Electrical Engineering and Computer Science (EECS), Swedish. 2022.
[12] Aradhya HR. Object detection and tracking using deep learning and artificial intelligence for video surveillance applications. Int J Adv Comput Sci Appl. 2019;10(12):517–30.
[13] Abdulghafoor NH, Abdullah HN. Object detection with simultaneous denoising using low-rank and total variation models. 2nd International congress on human – computer interaction, optimization and robotic application (HORA2020), Ankara, Turkey. IEEE; 2020. p. 1–10.
[14] Abdullah HN, Abdulghafoor NH. Objects detection and tracking using fast principle component purist and kalman filter. Int J Electr Comput Eng (2088-8708). 2020;10(2):1317–26.
[15] Abdulghafoor NH, Abdullah HN. A novel real-time multiple objects detection and tracking framework for different challenges. Alexandr Eng J. 2022;61(12):9637–47.
[16] Milan A, Leal-Taixé L, Reid I, Roth S, Schindler K. MOT16: A benchmark for multi-object tracking. 2016.
[17] Rasti B, Hong D, Hang R, Ghamisi P, Kang X, Chanussot J, et al. Feature extraction for hyperspectral imagery: The evolution from shallow to deep: overview and toolbox. IEEE Geosci Remote Sens Magazine. 2020;8(4):60–88.
[18] Fasana C, Pasini S, Milani F, Fraternali P. Weakly supervised object detection for remote sensing images: A survey. Remote Sensing. 2022;14(21):5362.
[19] Wang X, Xu H, Yuan L, Dai W, Wen X. A remote-sensing scene-image classification method based on deep multiple-instance learning with a residual dense attention ConvNet. Remote Sensing. 2022;14(20):5095.
[20] Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell. 2015;28:1137–49.
[21] Ross TY, Dollar G. Focal loss for dense object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 2980–8. https://arxiv.org/pdf/1708.02002.
[22] Zhao Y, Han R, Rao Y. A new feature pyramid network for object detection. In: 2019 International Conference on Virtual Reality and Intelligent Systems (ICVRIS). IEEE; 2019. p. 428–31.
[23] Zhang Y, He S, Wa S, Zong Z, Liu Y. Using generative module and pruning inference for the fast and accurate detection of apple flower in natural environments. Information. 2021;12(12):495.
[24] Wang CY, Liao HYM, Wu YH, Chen PY, Hsieh JW, Yeh IH. CSPNet: A new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops; 2020. p. 390–1.
[25] Liu S, Qi L, Qin H, Shi J, Jia J. Path aggregation network for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2018. p. 8759–68.
[26] Geeee Z, Liu S, Wang F, Li Z, Sun J. Yolox: Exceeding yolo series in 2021. 2021.
[27] Wang CY, Bochkovskiy A, Liao HYM. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023. p. 7464–75.
[28] Xu S, Guo Z, Liu Y, Fan J, Liu X. An improved lightweight yolov5 model based on attention mechanism for face mask detection. In: International Conference on Artificial Neural Networks. Springer; 2022. p. 531–43.
[29] Rezatofighi SH, Milan A, Zhang Z, Shi Q, Dick A, Reid I. Joint probabilistic data association revisited. In: Proceedings of the IEEE International Conference on Computer Vision; 2015. p. 3047–55.
[30] Bewley A, Ge Z, Ott L, Ramos F, Upcroft B. Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP). IEEE; 2016. p. 3464–8.
[31] Wojke N, Bewley A, Paulus D. Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP). IEEE; 2017. p. 3645–9.
[32] Chen L, Ai H, Zhuang Z, Shang C. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In: 2018 IEEE International Conference on Multimedia and expo (ICME). IEEE; 2018. p. 1–6.
[33] Wang Z, Zheng L, Liu Y, Li Y, Wang S. Towards real-time multi-object tracking. In: European Conference on Computer Vision. Springer; 2020. p. 107–22. https://link.springer.com/chapter/10.1007/978-3-030-58621-8_7.
[34] Zhang Y, Wang C, Wang X, Zeng W, Liu W. Fairmot: On the fairness of detection and re-identification in multiple object tracking. Int J Comp Vision. 2021;129:3069–87.
[35] Zhang Y, Sun P, Jiang Y, Yu D, Weng F, Yuan Z, et al. Bytetrack: Multi-object tracking by associating every detection box. In: European Conference on Computer Vision. Springer; 2022. p. 1–21.
[36] Dendorfer P, Rezatofighi H, Milan A, Shi J, Cremers D, Reid I, et al. Mot20: A benchmark for multi object tracking in crowded scenes. 2020.
[37] Kim C, Lee J, Han T, Kim YM. A hybrid framework combining background subtraction and deep neural networks for rapid person detection. J Big Data. 2018;5:1–24.
[38] Alikhanov J, Kim H. Online action detection in surveillance scenarios: a comprehensive review and comparative study of state-of-the-art multi-object tracking methods. IEEE Access. 2023;11:68079–92.
[39] Chen S, Shao C. Efficient online tracking-by-detection with Kalman filter. IEEE Access. 2021;9:147570–8.
[40] Ma L, Meng D, Huang X, Zhao S. Vision-based formation control for an outdoor UAV swarm with hierarchical architecture. IEEE Access. 2023;11:75134–51.
[41] Natarajan B, Elakkiya R, Bhuvaneswari R, Saleem K, Chaudhary D, Samsudeen SH. Creating alert messages based on wild animal activity detection using hybrid deep neural networks. IEEE Access. 2023;11:67308–21.
[42] Silano G, Iannelli L. MAT-fly: an educational platform for simulating unmanned aerial vehicles aimed to detect and track moving objects. IEEE Access. 2021;9:39333–43.

[43] Xu X, Feng Z, Cao C, Yu C, Li M, Wu Z, et al. STN-track: multiobject tracking of unmanned aerial vehicles by swin transformer neck and new data association method. IEEE J Selected Topics Appl Earth Observ Remote Sensing. 2022;15:8734–43.
[44] Yousif YM, Mukbil A, Müller JP. Offlinemot: A python package for multiple objects detection and tracking from bird view stationary drone videos. J Open Source Softw. 2022;7(74):4099.
[45] Liu L, Song X, Song H, Sun S, Han XF, Akhtar N, et al. Yolo-3DMM for simultaneous multiple object detection and tracking in traffic scenarios. IEEE Transactions on Intelligent Transportation Systems. 2024.
[46] Yuan Y, Wu Y, Zhao L, Chen H, Zhang Y. Multiple object detection and tracking from drone videos based on GM-YOLO and multi-tracker. Image Vision Comput. 2024;143:104951.
[47] Su S, Han S, Li Y, Zhang Z, Feng C, Ding C, et al. Collaborative multi-object tracking with conformal uncertainty propagation. IEEE Robotics Automat Lett. 2024;9:3323–30.
[48] Jain DK, Zhao X, Gan C, Shukla PK, Jain A, Sharma S. Fusion-driven deep feature network for enhanced object detection and tracking in video surveillance systems. Inform Fusion. 2024;109:102429.
[49] Youngjoo K, Hyochoong B. Introduction to Kalman filter and its applications. In: Govaers F, editors. Introduction and implementation of Kalman filter. London: Intechopen; 2018.
[50] Julier SJ, Uhlmann JK. Unscented filtering and nonlinear estimation. Proc IEEE. 2004;92(3):401–22.
[51] Jung HK, Choi GS. Improved yolov5: efficient object detection using drone images under various conditions. Appl Sci. 2022;12(14):7255.
[52] Bernardin K, Stiefelhagen R. Evaluating multiple object tracking performance: the clear mot metrics. EURASIP J Image Video Process. 2008;2008:1–10.
[53] Yang F, Zhang X, Liu B. Video object tracking based on YOLOv7 and DeepSORT. 2022.