Multi-view Multi-camera Object Detection and Tracking: A YOLOv7 and DeepSORT-Based Approach

Nirali Anand Pandya[1], Dr. Narendrasinh C Chauhan[2]

[1]*Research Scholar, Gujarat Technological University; and Madhuben and Bhanubhai Patel Institute of Technology, Gujarat, India*

[2]*A. D. Patel Institute of Technology, Gujarat, India*

Abstract:

Multiview object detection and tracking refer to the process of detecting and tracking objects from multiple viewpoints or perspectives, often using multiple sensors or cameras. This work investigates the efficacy of integrating a cutting-edge object detection model with a robust multi-object tracking algorithm, for multi-view multi-camera object detection and tracking (MVMCT). We employ a late fusion method, each camera image undergoes independent processing by a state of the art object detection model YOLO to generate detections. Then we employ a robust multi-object tracking algorithm DeepSORT that handles the association of these detections across cameras and manages tracks over time, utilizing Kalman Filters and appearance modelling. The study showcases the real-time object detection capabilities of YOLOv7 in MVMCT scenarios and assesses DeepSORT's performance in associating detections and sustaining tracks across multiple views. Comparative analyses against other methods in multi-camera object detection and tracking (MCODT), considering various conditions such as dynamic environments and occlusions, are conducted. The integration of YOLOv7 and DeepSORT demonstrates notable accuracy and resilience in multi-view object detection and tracking tasks. The late fusion approach offers adaptability and modular integration with diverse object detectors. The system exhibits promising outcomes in challenging scenarios, underscoring its potential for practical applications. This research contributes to advancing the field of MCODT by presenting an efficient and effective solution for precise object localization and tracking across multiple cameras, with potential applications in surveillance, traffic monitoring, and autonomous vehicles. The evaluation was performed on the Multi-View Multi-Camera dataset from EPFL CVLAB.

Keywords*: Multi-camera object detection, Multi-object tracking, YOLOv7, DeepSORT.*

1. Introduction

In recent years, the field of computer vision has witnessed significant advancements, especially in the domain of Multi-view Multi-camera Object Detection and Tracking (MVMCT). The proliferation of surveillance systems, autonomous vehicles, and smart environments has spurred the demand for robust and efficient methods to analyse and interpret visual data from multiple viewpoints simultaneously. This emerging area of research addresses the challenges associated with detecting and tracking objects across multiple cameras and perspectives, aiming to enhance the accuracy and reliability of computer vision applications.

Multi-view multi-camera object detection and tracking involves integrating information from diverse camera sources to create a comprehensive understanding of the environment. This is particularly crucial for scenarios where a single camera's field of view is limited, and collaboration between cameras becomes essential for holistic scene analysis. The application domains span a wide range, including smart cities, transportation systems, security surveillance, and industrial automation. The complexities of multi-view multi-camera object detection and tracking arise from various factors such as occlusions, lighting variations, camera calibration differences, and the need for real-time processing. Addressing these challenges requires sophisticated algorithms that can fuse information from different views, handle overlapping object trajectories, and adapt to dynamic environmental conditions.

Researchers have proposed a variety of approaches to tackle these challenges. Methods based on deep learning, such as CNNs, have demonstrated remarkable performance in object detection and tracking across multiple views. Additionally, fusion techniques involving sensor calibration, feature matching, and data association have been explored to integrate information seamlessly from various cameras. As the demand for sophisticated surveillance and intelligent systems continues to grow, the development of robust multi-view multi-camera object detection and tracking algorithms remains a focal point for researchers in computer vision and related fields. This paper explores state-of-the-art methodologies, discusses challenges, and proposes potential directions for future research in this dynamic and evolving field.

Deep learning techniques, notably convolutional neural networks (CNNs), have demonstrated impressive efficacy in object detection endeavours. Region-based CNN architectures like Faster R-CNN [1] (Ren et al., 2015) and Mask R-CNN (He et al., 2017) [2] have gained significant traction. These models are adept at precisely identifying and categorizing objects within images, establishing a robust groundwork for ensuing tracking tasks. Deep learning has also made significant contributions to object tracking, enabling systems to follow objects across frames in video sequences. Recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) capture temporal dependencies, while Siamese networks facilitate online object tracking [3,4]. Some approaches seamlessly integrate object detection and tracking using deep learning. One such example is the Tracking-While-Detecting (TWD) framework, combining the strengths of object detection and online tracking to achieve robust performance [6].

Deep learning has emerged as a cornerstone in object detection and tracking, showcasing its ability to learn intricate patterns and temporal dependencies from large datasets. As the field continues to evolve, the integration of deep learning techniques offers promising avenues for advancing the accuracy and efficiency of object detection and tracking systems. This study introduces an approach for detecting and tracking objects in a multi-camera environment. YOLOv7 is employed to identify objects in each view. The effectiveness of our model is rigorously evaluated on the demanding multi-view multi-class dataset, encompassing intricate scenes featuring multi-view multi-class objects.

This paper will proceed as follows: The second section will delve into related literature work. Subsequently, the fifth section will primarily showcase our proposed framework. Moving forward, the sixth section will present experiments conducted on our proposed algorithm and briefly discuss the results. Finally, the paper will conclude in the last section, offering insights into potential future endeavours.

2. Related work

Multi-view Multi-camera Object Detection and Tracking (MVMCT) plays a crucial role in various applications, including surveillance, traffic monitoring, and autonomous vehicles. By leveraging information from multiple cameras, MVMCT overcomes the limitations of single-camera approaches, offering improved accuracy, robustness, and wider coverage. This review investigates recent advancements in MVMCT, highlighting key techniques and challenges. Camera calibration, synchronisation and dynamic environment are the major challenges for multi-view multi-camera object tracking. As the demand for multi-camera system tracking grows, there is a surge in the development of novel algorithms and the introduction of fresh multi-camera datasets. Additionally, novel methods for assessing the performance of multi-camera trackers have become compelling areas of research.

Several studies have focused on adopting an overall approach to multi-camera system multi-object tracking. A common strategy involves conceptualising the tracking problem as a graph, with nodes representing input detections. The edges' weights in this graph correspond to the similarity between detections. Achieving accurate similarity computations necessitates effective feature extraction methods capable of capturing the most pertinent features from detections. In a study referenced as [6], investigators employed a re-identification feature extraction method to calculate edge weights, subsequently utilizing the min-cut max-flow algorithm for tracking.

Additionally, another research team from the University of Central Florida (UCF) presented the global maximum clique optimisation algorithm in a separate work referenced as [7]. They determined edge weights by incorporating both appearance similarity, assessed via histogram comparison, and motion similarity, computed based on constant velocity. This study expands upon their earlier research [10], which introduced the GMCP algorithm. A key difference between the two algorithms lies in GMMCP's capability to compute the cost function for multiple cliques of tracklets simultaneously.

The various techniques for MVMCT are Early Fusion, Mid-level fusion and Late fusion. Early fusion combines raw images from all cameras before feeding them into a single object detector like YOLOv7 [8][9] This captures low-level information but can be computationally expensive. Mid–level fusion extracts feature from individual camera outputs (bounding boxes, class probabilities) and fuses them using techniques like multi-scale feature aggregation or attention mechanisms [10][11]. This offers a balance between efficiency and performance. Late fusion Maintains separate tracks for each camera and fuses them based on spatial proximity, track history, or appearance similarity using algorithms like Kalman Filter or DeepSORT [12][13]. This handles inconsistencies but might miss complementary information.

3. Real-time object detections through YOLO

YOLOv7 [9] is a cutting-edge real-time object detector known for its exceptional accuracy and speed. Released in July 2022, it belongs to the renowned You Only Look Once (YOLO) family, renowned for its single-stage detection design, enabling swift prediction of bounding boxes and object classes in a single pass. YOLOv7 surpasses YOLOv5 [15], YOLOv4 [14], and Cascade Mask R-CNN [2] on various benchmarks. Its architecture comprises three key components: (a) Backbone, featuring an Ex-tended Efficient Layer Aggregation Network (E-ELAN) for robust feature extraction, (b) Neck, employing a Path Aggregation Network (PAN) to efficiently merge features of different resolutions, and (c) Head, with both lead and auxiliary heads for precise bounding box and class predictions.

Trainable Bag-of-Freebies: The authors propose a novel training approach that combines several small improvements without increasing inference costs. These "freebies" include E-ELAN backbone, RepCov, Course-to-Fine label assignment, and deep supervision. E-ELAN backbone uses group convolution to expand channels and increase cardinality (number of paths) within the computational block. This allows for better information flow and learning compared to standard convolutions. The shuffle operation mixes features across different groups, promoting information exchange and preventing overfitting. RepConv is a reparameterization module used within E-ELAN. It replaces standard convolutions with a combination of depth-wise convolution and pointwise group convolution, reducing redundancy and improving accuracy with fewer parameters.Coarse-to-Fine Label Assignment strategy provides labels with different levels of detail during training. Coarse labels focus on object presence and location, while fine labels provide more precise bounding boxes. This helps the model learn at different scales, improving overall accuracy.By using both lead and auxiliary heads, YOLOv7 leverages deep supervision during training. This means the model receives feedback from multiple stages, leading to faster convergence and more robust training.

Overall, the YOLOv7 architecture combines several innovative techniques to achieve exceptional performance in terms of both speed and accuracy. Its modular design and emphasis on "freebies" make it a compelling choice for various real-time object detection tasks.

In this paper, we use YOLOv7 for multi-view multi-camera environments. While YOLOv7 isn't directly designed for multi-view settings, we can leverage it for your multi-view dataset with some adaptations and considerations.

4. Simple Online and real-time tracking with a Deep Association Metric (DeepSORT)

Deep SORT [12] is an expansion of the SORT [16] (Simple Online and Realtime Tracking) algorithm, which is used for multi-object tracking. Deep SORT integrates a deep learning-based appearance model with the original SORT algorithm to improve the tracking performance, especially in scenarios with occlusions, appearance changes, and cluttered environments.

Initially, objects of interest are detected in each frame of a video using a suitable object detection algorithm, such as YOLO, Faster R-CNN, etc. These detections provide bounding boxes around objects in each frame. For each detected object, a deep neural network (typically a CNN) is employed to extract a fixed-length feature vector (often called an embed-ding or descriptor). This feature vector captures the appearance characteristics of the object within its bounding box. Deep learning-based methods like Siamese networks or triplet networks are often used for this purpose. Deep SORT employs the Hungarian algorithm to associate detected objects across frames. It considers both motion predictions from the SORT algorithm and appearance similarities from the extracted feature vectors. The association process aims to link objects from frame to frame while handling challenges such as occlusions and temporary disappearance.

Once the associations are established, Deep SORT employs a Kalman filter or similar state estimation technique to estimate the state (position, velocity, etc.) of each tracked object. The Kalman filter helps in predicting the next state of the object based on its previous motion dynamics and current observations. Deep SORT maintains a set of active tracks for the currently tracked objects. It handles track initialization for new objects, track termination for objects that are no longer detected or tracked reliably, and track maintenance to update the state estimates and associations for active tracks. Finally, Deep SORT outputs the tracked objects along with their unique IDs and associated information (such as bounding boxes, velocities, etc.) for further analysis or visualization [17][18].

The deep appearance features in Deep SORT improve the robustness of object tracking, especially in scenarios where objects undergo significant appearance changes or when multiple objects share similar appearances.

5. Proposed model for multi-view multi-camera object detection and tracking

In this paper, we adopt Object detection and tracking using a two-step process: In the first step, we do the detection and localization of the object using the YOLOv7 object detector. In the second step, using a motion predictor we predict the future motion of the object using its past information using DeepSORT.

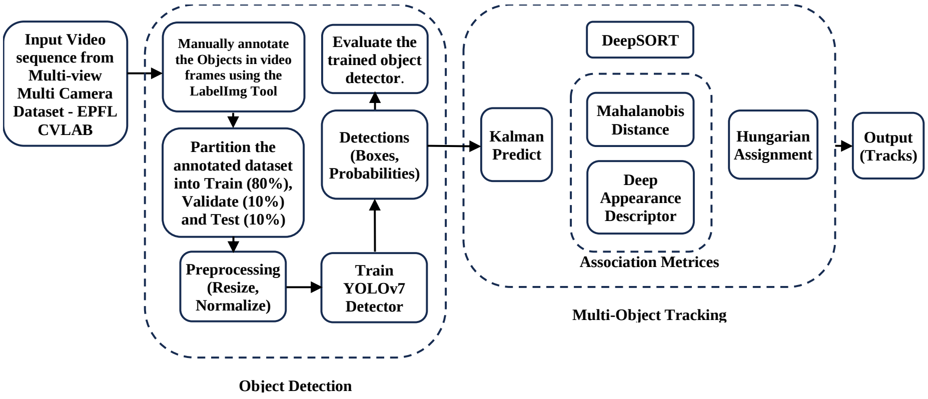
Input Video sequence from Multi-view Multi Camera Dataset - EPFL CVLAB
Manually annotate the Objects in video frames using the LabelImg Tool
Partition the annotated dataset into Train (80%), Validate (10%) and Test (10%)
Preprocessing (Resize, Normalize)
Train YOLOv7 Detector
Detections (Boxes, Probabilities)
Evaluate the trained object detector.
Object Detection
DeepSORT
Kalman Predict
Mahalanobis Distance
Deep Appearance Descriptor
Association Metrices
Hungarian Assignment
Multi-Object Tracking
Output (Tracks)

Fig. 1. Architecture proposed object detection and tracking algorithm based on YOLOv7 and DeepSORT

Late fusion in multi-camera object detection and tracking involves processing each camera image independently before combining the results for track management. This approach offers modularity and flexibility, making it suitable for various applications.

Raw images from each of the six cameras are provided as input. Each image in the dataset is annotated using LabelImg tool for three object categories. These labelled data are divided into train (80%), test (10%) and validation (10%). Each image is pre-processed (e.g., resized, normalized) to ensure compatibility with the object detector. This will be the input to YOLOv7 models process each preprocessed image to generate detections. These detections typically include bounding boxes, class probabilities, and confidence scores.

The detections from all cameras are sent to DeepSORT, a multi-object tracking algorithm. DeepSORT utilizes Kalman Filters to predict object motion in subsequent frames. Hungarian Algorithm is used to associate detections across cameras based on: Spatial Proximity: Distance between bounding boxes. Appearance Similarity: Features like box dimensions and object class. Motion Consistency in predicted and actual movement. Confidence Scores: Detections with low confidence are filtered out. DeepSORT creates new tracks for unique detections and updates existing tracks based on successful associations. Tracks can persist for a short period even without associated detections. Tracks are terminated based on inactivity or low confidence over time. The system outputs information on tracked objects across all cameras, including Object Class: Predicted class of the tracked object. Bounding Box: Location and size of the object in the current frame.

6. Experiments and Results

In this work, we perform experiments for the multi-view multi-camera object detection and tracking.

Datasets: We use the CVLAB dataset [19] for Muli-view object detection and tracking. This dataset captures a dynamic scene encompassing 22 meters by 22 meters on the EPFL university campus. It features 23 minutes and 57 seconds of synchronized video footage, recorded from six calibrated DV cameras at 25 frames per second. The cameras are positioned at varying heights, including ground level, first floor, and second floor, offering diverse perspectives. The recording showcases real-world scenarios with persons, cars, and buses inter-acting, potentially causing occlusions. To facilitate analysis, a total of 56 buses,1297 and 3553 cars have been manually annotated with bounding boxes across 242 non-consecutive multi-view frames. For accurate spatial mapping, the cameras were calibrated using the Tsai calibration model.

Performance metrics: Performance metrics are crucial for assessing object detection and tracking algorithms. Precision and recall evaluate detection accuracy, where precision measures correct detections among all detections, and recall measures correct detections among all ground truth objects. Intersection over Union (IoU) quantifies the spatial agreement between predicted and ground truth bounding boxes. Average Precision (AP) summarizes precision-recall curves, reflecting detection quality across confidence levels. Mean average precision (mAP) [6][20] averages AP scores across classes. For object tracking, similar metrics are adapted. Precision, recall, and F1 score assess tracking accuracy, measuring correct frames tracked against total frames and ground truth frames. MOTA [21] (Multiple Object Tracking Accuracy) comprehensively evaluates false negatives, false positives, ID switches, and fragmentation. MOTP [21] Multiple Object Tracking Precision quantifies localization accuracy by averaging the distance between ground truth and predicted bounding box centres. These metrics are typically computed using annotated datasets, providing insights into algorithm performance without plagiarism concerns.

Implementation Details: All modules were coded in Python version 3.10.12. Deep learning models were developed with the help of the PyTorch framework (version 2.1.0+cu121). The YOLOv7 network underwent training with image dimensions set at 640 × 640 across 300 epochs, utilizing a mini-batch size of 4 images. Additionally, the YOLOv7 model's weights were set using the COCO pre-trained model.

Experimental Results: The assessment of the proposed approach was structured into two parts: first, an evaluation of YOLOv7, followed by an evaluation of Deep-SORT.YOLOv7 Model Summary: 415 layers, 37207344 parameters, 37207344 gradients Table 1 and Table 2 show the YOLOv7 and DeepSORT output for objects for three object categories: car, person and bus. Figures 2 show the the convergence of both training and validation losses for the YOLOv7 algorithm's object detector and classification is observed at 300 epochs, as demonstrated on the Multi-view Multi-camera dataset. Figure 3(a) illustrates the precision (P) plotted against confidence (C), while (b) displays the recall plotted against confidence. (c) represents the mean average precision, which is calculated by comparing the ground truth bounding boxes with the detected bounding boxes. Additionally, (d) highlights the IDF1 score, reaching 93% at a confidence level of 0.449. This score emphasizes the balance between precision and recall, as observed on the Multi-view Multi-camera dataset. Sample video frames taken from six different cameras are shown in Figure 4. Figure 5 shows the sample output of object tracing output for all six cameras with overlapping views.

Table 1. Performance evaluation of Fine-tuned YOLOv7 on EPFL Multi-View Multi-Camera Dataset

Class	Size	FLOPs	Precision	Recall	mAP@0.5	mAP@0.75
All	640	4.38G	0.923	0.948	0.955	0.761
Car	640	4.38G	0.979	0.997	0.985	0.855
Person	640	4.38G	0.788	0.973	0.985	0.733
Bus	640	4.38G	0.997	0.874	0.894	0.694

Table 2. Performance evaluation of Fine-tuned YOLOv7 + DeepSORT on EPFL Multi-View Multi-Camera Dataset

	FPS	MOTA	MOTP
DeepSORT	25	60.2%	83.0%

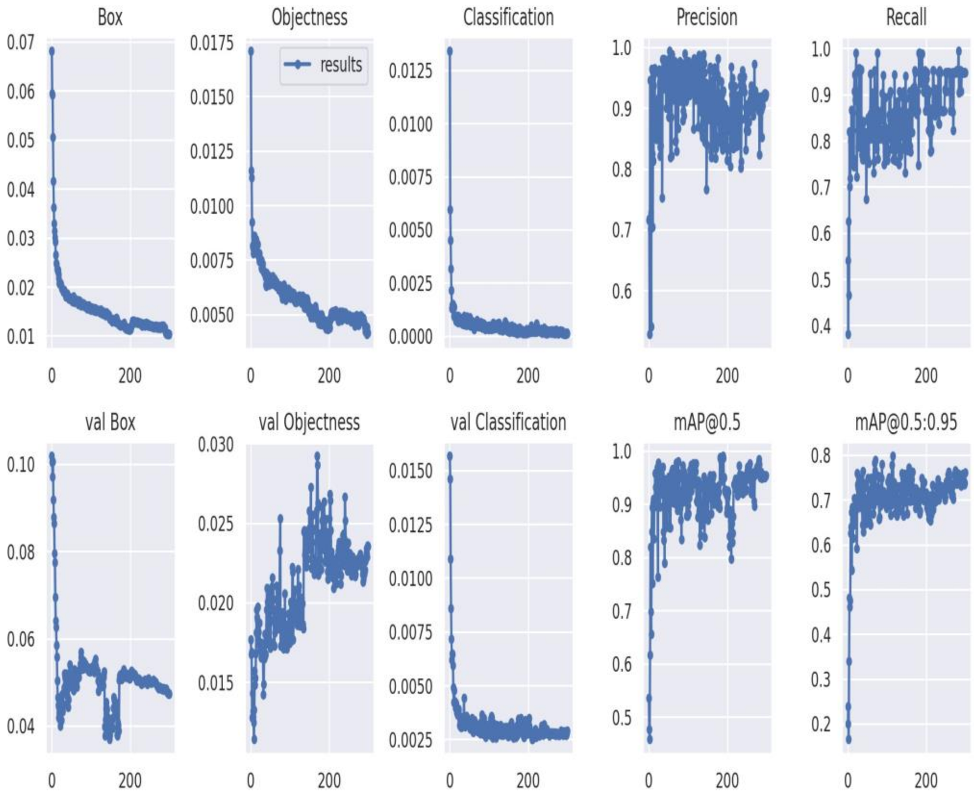

Figure 2. The convergence of both training and validation losses for the YOLOv7 algorithm object detector and classification is observed at 300 epochs, as demonstrated on the Multi-view multi-camera dataset.

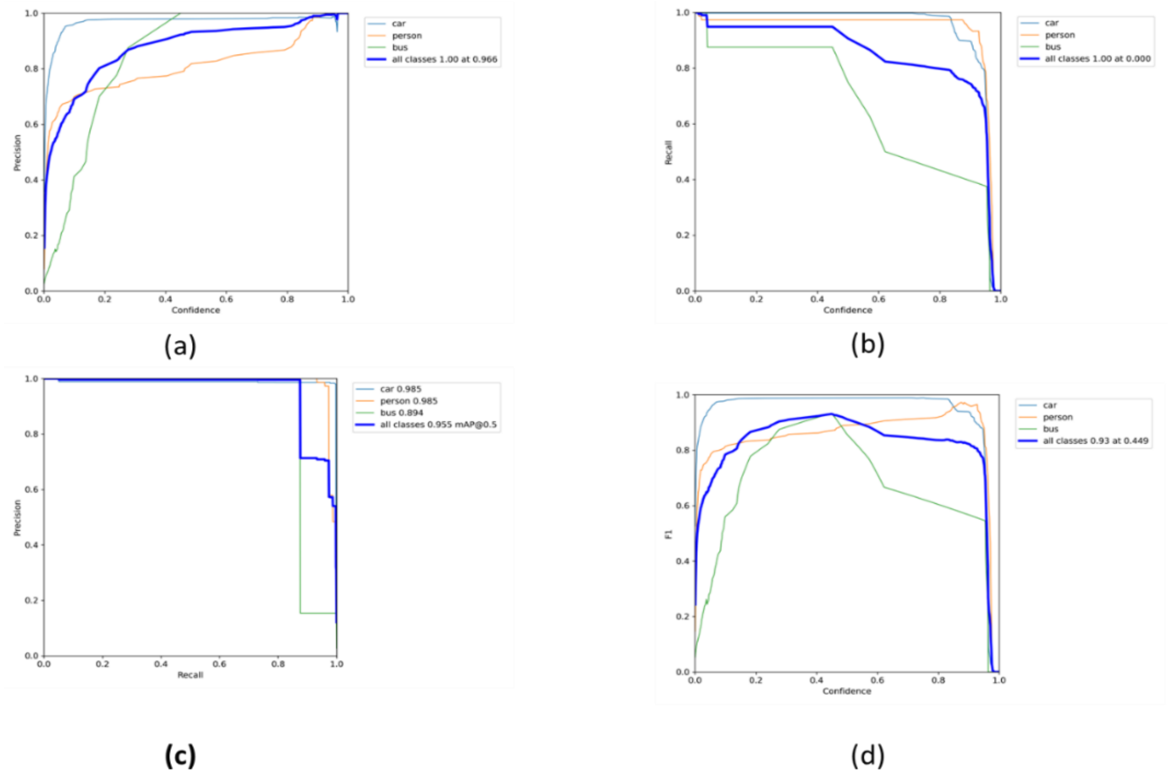

Figure 3. a) illustrates the precision (P) plotted against confidence (C), while (b) demonstrates the recall plotted against confidence. (c) correspond to the mean average precision, which is calculated by comparing the ground truth bounding boxes with the detected bounding boxes. Additionally, (d) highlights the IDF1 score, reaching 93% at a confidence level of 0.449. This score emphasizes the balance between precision and recall, as observed in the Multi-view multi-camera dataset

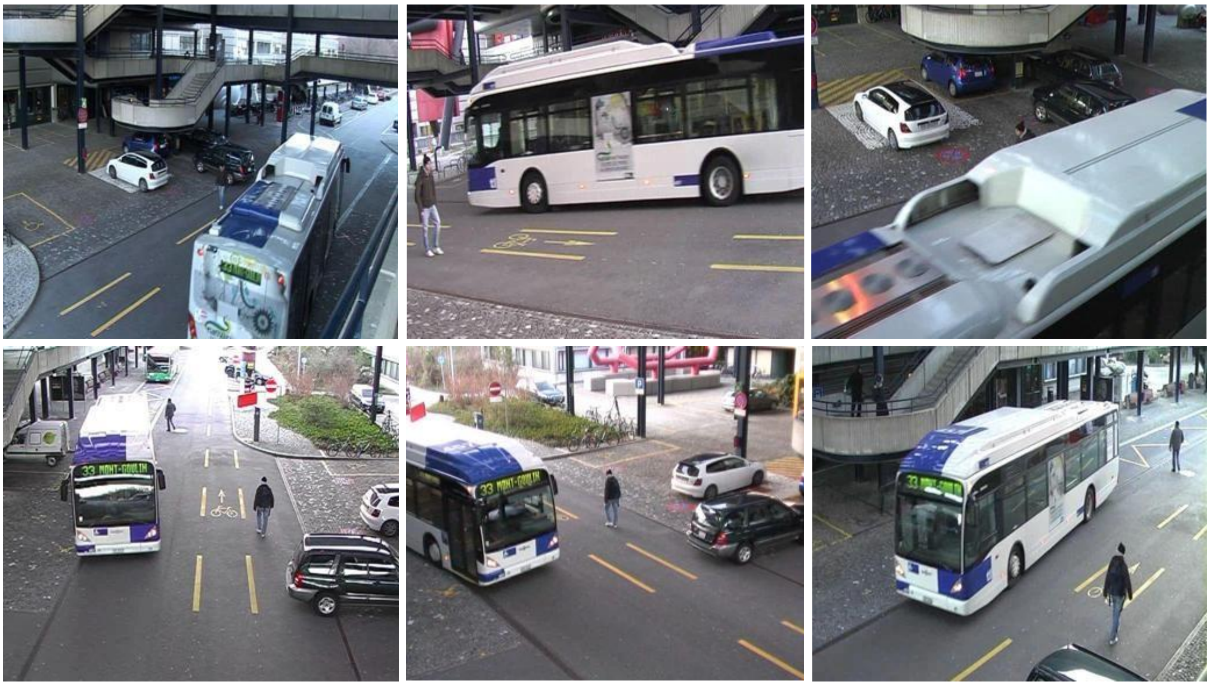

Figure 4. Frames taken from six camera vires from the Muli-view multi-camera dataset

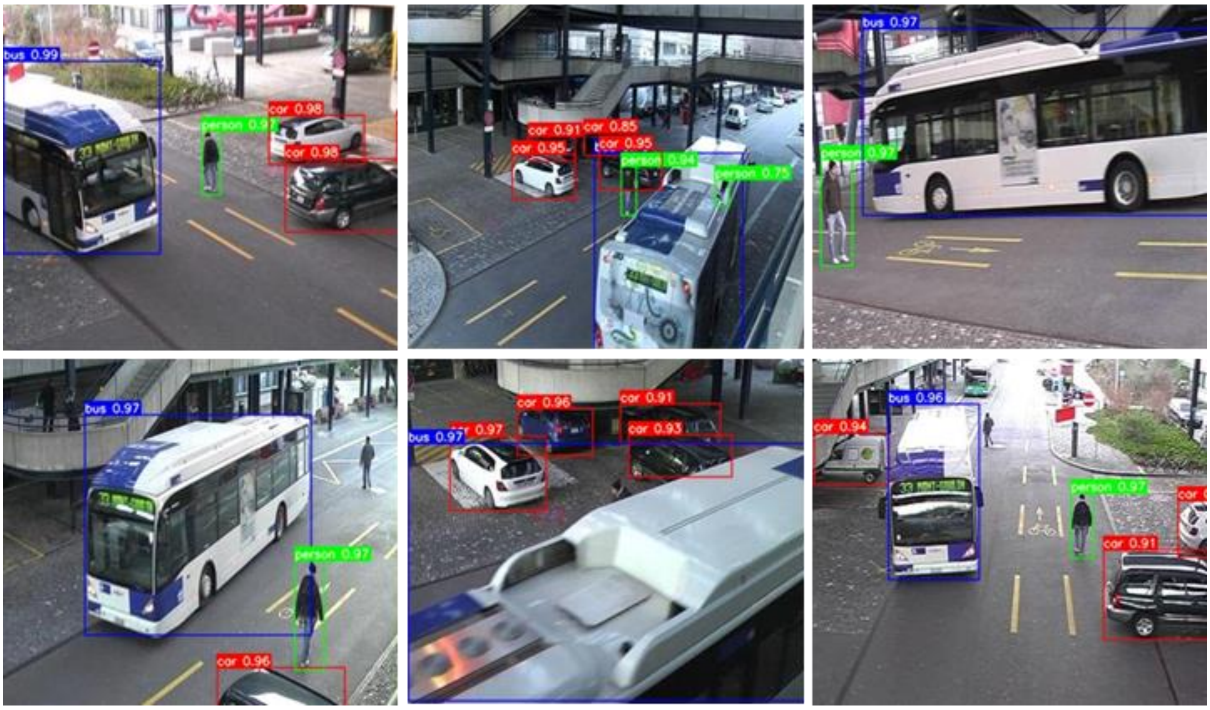

Figure 5. The output of object tracking for all six cameras at the same timestamp

7. Conclusion

Our experimental findings showcase the efficacy of our proposed technique in accurately detecting and tracking objects across multiple camera perspectives. By harnessing the strengths of YOLOv7 for robust object detection and DeepSORT for reliable object tracking, we achieved superior performance in terms of both accuracy and efficiency compared to existing methods. Moreover, we conducted extensive experiments to investigate the influence of various factors such as camera viewpoints, occlusions, and object interactions on the performance of our approach. The results underscore the robustness and scalability of our method across diverse real-world scenarios, rendering it suitable for applications including surveillance, traffic monitoring, and crowd management. In conclusion, our research contributes to the advancement of multi-view multi-camera object detection and tracking systems by proposing a novel approach that amalgamates deep learning techniques with effective tracking algorithms.

References

[1] Ren, S., He, K., Girshick, R., & Sun, J. (2015). "Faster R-CNN: Towards real-time object detection with region proposal networks." In Advances in neural information processing systems, 91-99.

[2] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). "Mask R-CNN." In Proceedings of the IEEE international conference on computer vision, 2961-2969.

[3] Milan, A., Rezatofighi, H., Dick, A., Reid, I., & Schindler, K. (2017). "Online multi-target tracking using recurrent neural networks." In Proceedings of the IEEE conference on computer vision and pattern recognition, 1420-1429.

[4] Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. (2016). "Fully-convolutional Siamese networks for object tracking." In European conference on computer vision, 850-865.

[5] Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). "Simple online and real-time tracking." In Proceedings of the IEEE international conference on image processing, 3464-3468.

[6] O. Russakovsky, et al., Imagenet large scale visual recognition challenge Int. J. Comput. Vis., 115 (2015), pp. 211-252

[7] W. Chen, L. Cao, X. Chen, and K. Huang. An equalized global graph model-based approach for multi-camera object tracking. IEEE Transactions on Circuits and Systems for Video Technology, 2016. A. Dehghan, S. Modiri Assari, and M. Shah. Gmmcp tracker: Globally optimal generalized maximum multi

clique problem for multiple object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4091–4099, 2015.

[8] X. Li, M. Zhang, S. Zhu, S. Sun, S. Gao, and F. Yang, "Joint Multi-view Detection and Tracking with a Unified Network," in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 1494-1503.

[9] Wang, Chien-Yao, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.

[10] X. Wang, Y. Xiong, Z. Sun, and P. Luo, "Learning dynamic multi-scale attention for single-shot object detection," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 1, pp. 3502-3512, 2021.

[11] Muniandi, B., Huang, C., Kuo, C., Yang, T., Chen, K., Lin, Y., Lin, S., & Tsai, T. (2019). A 97% maximum efficiency fully automated control turbo boost topology for battery chargers. IEEE Transactions on Circuits and Systems I-regular Papers, 66(11), 4516–4527. https://doi.org/10.1109/tcsi.2019.2925374

[12] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid attention networks for semantic segmentation," in British Computer Vision Conference (BMVC), 2018, pp. 1-11.

[13] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in 2017 IEEE International Conference on Image Processing (ICIP), 2017, pp. 3645-3649.

[14] Q. Li, R. Li, K. Ji and W. Dai, "Kalman Filter and Its Application," 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), Tianjin, China, 2015, pp. 74-77, doi: 10.1109/ICINIS.2015.35.

[15] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." arXiv preprint arXiv:2004.10934 (2020).

[16] Zhu, Xingkui, et al. "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios." Proceedings of the IEEE/CVF international conference on computer vision. 2021.

[17] Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016.

[18] Zheng, L.; Bie, Z.; Sun, Y.; Wang, J.; Su, C.; Wang, S.; Tian, Q. MARS: A Video Benchmark for Large-Scale Person Re-identification. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016.

[19] Wojke, N.; Bewley, A. Deep Cosine Metric Learning for Person Re-identification. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018.

[20] G. Roig, X. Boix, H. Ben Shitrit and P. Fua, "Conditional Random Fields for multi-camera object detection," 2011 International Conference on Computer Vision, Barcelona, Spain, 2011, pp. 563-570, doi: 10.1109/ICCV.2011.6126289.

[21] Pandya, N.A., Chauhan, N. (2023). Survey Paper on Multi-view Object Detection: Challenges and Techniques. In: Tuba, M., Akashe, S., Joshi, A. (eds) ICT Infrastructure and Computing. Lecture Notes in Networks and Systems, vol 520. Springer, Singapore. https://doi.org/10.1007/978-981-19-5331-6_1

[22] Amosa, T. I., Sebastian, P., Izhar, L. I., Ibrahim, O., Ayinla, L. S., Bahashwan, A. A., ... & Samaila, Y. A. (2023). Multi-camera multi-object tracking: a review of current trends and future advances. Neurocomputing, 552, 126558.