MC-GTA: A Synthetic Benchmark for Multi-Camera Vehicle Tracking

Luca Ciampi[1[0000−0002−6985−0439]], Nicola Messina[1[0000−0003−3011−2487]], Gaetano Emanuele Valenti[2], Giuseppe Amato[1[0000−0003−0171−4315]], Fabrizio Falchi[1[0000−0001−6258−5313]], and Claudio Gennaro[1[0000−0002−3715−149X]]

[1] Institute of Information Science and Technologies, ISTI-CNR
Via G. Moruzzi 1, 56124 Pisa, Italy
name.surname@isti.cnr.it
[2] Department of Information Engineering, University of Pisa
Via G. Caruso 16, 56122 Pisa, Italy

Abstract. Multi-camera vehicle tracking (MCVT) aims to trace multiple vehicles among videos gathered from overlapping and non-overlapping city cameras. It is beneficial for city-scale traffic analysis and management as well as for security. However, developing MCVT systems is tricky, and their real-world applicability is dampened by the lack of data for training and testing computer vision deep learning-based solutions. Indeed, creating new annotated datasets is cumbersome as it requires great human effort and often has to face privacy concerns. To alleviate this problem, we introduce MC-GTA - Multi Camera Grand Tracking Auto, a synthetic collection of images gathered from the virtual world provided by the highly-realistic Grand Theft Auto 5 (GTA) video game. Our dataset has been recorded from several cameras recording urban scenes at various crossroads. The annotations, consisting of bounding boxes localizing the vehicles with associated unique IDs consistent across the video sources, have been automatically generated by interacting with the game engine. To assess this simulated scenario, we conduct a performance evaluation using an MCVT SOTA approach, showing that it can be a valuable benchmark that mitigates the need for real-world data. The MC-GTA dataset and the code for creating new ad-hoc custom scenarios are available at https://github.com/GaetanoV10/GT5-Vehicle-BB.

Keywords: Multi-Camera Vehicle Tracking · Multi-Target Multi-Camera Tracking · Synthetic Data · Deep Learning · Computer Vision.

1 Introduction

Intelligent transportation systems (ITS) constitute an essential pillar of modern smart cities, playing a crucial role in traffic management, urban areas planning, pollution reduction, and, in general, improving urban mobility and sustainability. In particular, automated video analysis is emerging as one of the more attractive ITS smart applications, aided by the ubiquity of city-camera networks and the recently astonishing progress in computer vision.

In this context, multi-target multi-camera tracking (MTMCT) is essential for such applications since it provides crucial information for scene understanding. Specifically, it aims at tracking objects over large areas in multiple, possibly non-overlapping, surveillance camera networks. Among its branches, multi-camera vehicle tracking (MCVT) is beneficial for city-scale traffic analysis and management and for tasks in modern security, e.g., tracing a felonious vehicle between different cameras in a city. However, designing and developing MCVT techniques is tricky since several challenging computer vision tasks are involved – object detection, single-camera multiple object tracking, and object re-identification. A more critical challenge is the lack of suitable datasets for training and testing the deep learning models on which SOTA computer vision solutions rely. Indeed, to enable MCVT development and assessment, data needs to include a comprehensive ground truth covering heterogeneous scenarios, different illumination and weather conditions, large variations in camera distance, resolution, and view angle, as well as provide consistent vehicle IDs across all the cameras. Such datasets require a tremendous human effort for data acquisition and curation, and often it is not even possible to collect them due to violations of data protection rights.

In this paper, we tackle the data scarcity problem affecting the MCVT task by introducing and making freely available MC-GTA - Multi Camera Grand Tracking Auto, a collection of synthetic images gathered from the virtual world provided by the highly-realistic Grand Theft Auto 5 (GTA) video game. Our dataset has been collected by several overlapping and non-overlapping cameras recording urban scenes located at various crossroads, as shown in Figure 1. Annotations are automatically generated by interacting with the game engine and consist of bounding boxes localizing the vehicles with associated unique IDs that remain consistent across all the cameras, thus making our dataset suitable for training/testing deep learning-based MCVT models. Furthermore, we provide the code[3] needed for easily creating and recording new ad-hoc scenarios resembling real-world scenes that practitioners wish to simulate, where it is possible to vary not only camera locations but also other factors of interest, such as weather conditions and time of the day. This simulated environment has been exploited as a benchmark where we conduct a baseline performance evaluation using a SOTA deep learning-based approach for MCVT. The results show that our simulator can be a helpful and versatile tool for assessing MCVT techniques.

By summarizing, we list below the main contributions of this paper:

- we propose MC-GTA - Multi Camera Grand Tracking Auto, a new synthetic benchmark suitable for training/testing deep learning-based multi-camera vehicle tracking techniques, collected by exploiting the highly-realistic Grand Theft Auto 5 (GTA) video game;
- we release the code for designing and implementing new ad-hoc scenarios where practitioners have control of several factors such as camera locations and weather conditions;
- we conduct a performance evaluation using a SOTA MCVT model and our dataset as a testing ground;

[3] https://github.com/GaetanoV10/GT5-Vehicle-BB

– results show that our simulated scenarios can be a valuable tool for measuring the performances of MCVT techniques in a controlled environment, mitigating the need for new real-world annotated data.

2 Related Work

In this section, we report some works present in the literature that are relevant to our. Specifically, we focus on methods suitable for the Multi-Camera Vehicle Tracking task. Then we describe some of the most influential synthetic datasets exploited for tackling the data scarcity problem.

2.1 Multi-Camera Vehicle Tracking

Multi-camera vehicle tracking (MCVT) [28, 24] is usually tackled using a combination of different computer vision techniques ranging from object detection, single-camera multi-object tracking, and object re-identification.

Object Detection. Object detection is one of the fundamental tasks of computer vision aiming at localizing instances of semantic objects belonging to several classes, such as people, bikes, or vehicles, in digital images and videos. Current approaches rely on deep learning, leveraging different approaches that can be classified as (i) anchor-based that rely on anchors, i.e., prior bounding boxes with various scales and aspect ratios, either directly regressing from pixels to bounding boxes (such as YOLO family [25, 16, 29] and RetinaNet [19] algorithm) or by refining a bunch of region of interest computed in a preliminary step (such as Faster R-CNN [26] and Mask R-CNN [15]); (ii) anchor-free that rely on predicting key points, such as corners or center points, instead of using anchor boxes and their inherent limitations (such as CenterNet [32] and YOLOX [14]); (iii) transformer-based that rely on the recently introduced attention modules in processing image feature maps (such as DEtection TRansformer (DETR) [4] and one of its evolution Deformable DETR [33]).

Single-camera Multi-Object Tracking. Multi-object tracking (MOT) aims to trace multiple targets in video frames. Popular implementations of MOT algorithms are SORT [3], which uses object detection and Kalman filtering, and DeepSORT [30], which improves SORT by adding a feature similarity matching strategy. Another notable architecture is Towards-Realtime-MOT [18], which unifies detection and feature into a single model. More recently, architectures relying on transformers, such as TrackFormer [23], are also becoming available.

Object Re-identification. Object re-identification (ReID) is usually considered a retrieval task that aims at matching targets in different scenes. Previously, most works addressed person ReID; on the other hand, vehicle ReID is even more challenging since the same vehicle models have the same appearance. Therefore, to enhance accuracy, some methods resorted to multiple information formats like license plates, vehicle color information, time and space metadata, etc [21, 22, 20, 31].

2.2 Synthetic Datasets

Synthetic datasets have recently received considerable interest since they represent an appealing solution to mitigate the data scarcity problem. Usually, data is gathered by creating ad-hoc scenarios using simulators based on the Unreal or Unity graphical engines; some examples of these collections of images are [10, 9, 27, 12, 13], suitable for autonomous driving and pedestrian and tracking. Furthermore, another option is to use the Grand Theft Auto 5 (GTA) video game, which exhibits a higher level of realism and variability among scenarios; some notable existing works present in the literature are Joint Track Auto (JTA) [11] for pedestrian pose estimation and tracking, CrowdVisorPPE [2] for personal protective equipment detection, Virtual World Fallen People (VWFP) [5] for fallen people detection, Grand Traffic Auto (GTA) [8] for vehicle segmentation and counting, and Virtual Pedestrian Dataset (ViPeD) [6, 1] for pedestrian detection and tracking. In this work, we fill the gap determined by the lack of a synthetic dataset collected from the GTA5 video game suitable for the multi-camera vehicle tracking task.

3 The MC-GTA - Multi Camera Grand Tracking Auto Dataset

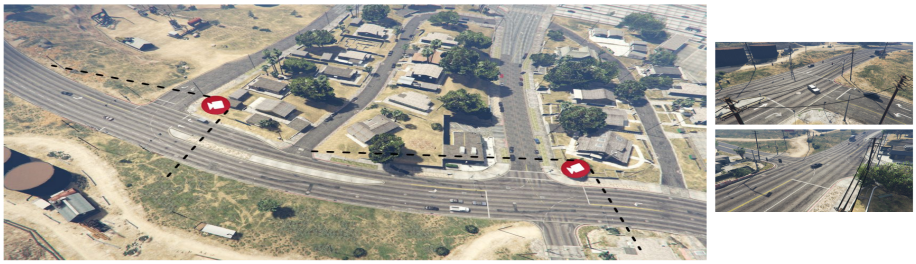
In this section, we deeply describe our Multi Camera Grand Tracking Auto (MC-GTA) dataset, providing details about the procedure employed for acquiring and processing data and illustrating some statistics.

3.1 Overview

Our proposed dataset, which we called MC-GTA - Multi Camera Grand Tracking Auto, includes high-quality imagery collected from the virtual world provided by the highly-realistic Grand Theft Auto 5 (GTA) video game. Specifically, it has been gathered from two different simulated urban scenarios, where several overlapping and non-overlapping cameras have been placed at various crossroads: (i) the first scenario consists of six cameras divided into three overlapping camera-pairs located in three consecutive crossroads (Figure 1a); (ii) the second scenario includes two non-overlapping cameras placed in two consecutive crossroads (Figure 1b).

Annotations are automatically gathered by interacting with the game engine and rely on bounding boxes localizing the vehicles together with associated IDs that remain unique and consistent among the cameras belonging to a specific scenario. In Table 1, we report some statistics from each acquired scenario. Specifically, each scenario was recorded with a final frame rate of around 12 FPS. The first scenario lasts 79 seconds with 70 unique vehicles, while the second scenario lasts 253 seconds and contains 109 different vehicles. The detailed statistics on the number of cameras traversed by each vehicle are reported in Figure 2b. Specifically, most of the vehicles are captured by two cameras in both environments. This is most likely in scenario #1, where every crossing is captured by

(a) Scenario #1

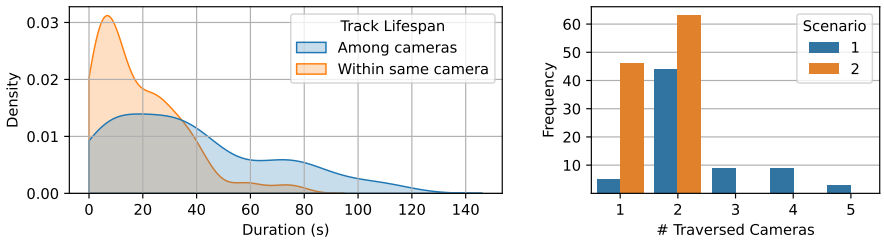


(b) Scenario #2

Fig. 1: The considered scenarios of our MC-GTA dataset: (a) the first one includes three pairs of overlapping cameras located at three crossroads; (b) the second one includes two non-overlapping cameras placed at two crossroads.

two cameras. In this case, the few vehicles captured by only one camera are the ones that are leaving the intersection when the video is started. In Figure 2a, we can better appreciate the distribution of tracks' lifespan, either within single cameras (for standard single-camera vehicle tracking) or across cameras for MCVT.

The core reason that motivated the creation of this dataset is the need to have a well-labeled benchmark resembling with high fidelity the real world, helpful for assessing the performance of multi-camera multi-vehicle tracking models in a simulated and controlled environment. Since labels are generated through an automated procedure, MC-GTA helps in mitigating and contrasting the data scarcity problem, also considering that practitioners can create new custom ad-hoc scenarios by using the provided freely available code. Furthermore, since data come from a virtual environment, our dataset can alleviate possible privacy concerns with the depicted subjects. Secondly, MC-GTA can also be exploited as training data for the supervised learning of deep learning models, given that the limitation represented by the domain gap between synthetic training data and real-world test data is less pronounced in MTMCT [17], or can eventually be contrasted with domain adaptation techniques [6, 8, 7].



(a) Distribution of track lifespan for both single-camera tracks and tracks transiting within multiple cameras.

(b) Distribution of the number of cameras traversed by each vehicle for both scenarios.

Fig. 2: MC-GTA dataset statistics.

Table 1: MC-GTA dataset statistics for each scenario.

Feature	Scenario 1	Scenario 2
Num Cameras	6	2
Video duration	79s	253s
FPS	12.6	11.8
Unique vehicles	70	109
Average Vehicles per Camera	5.3±2.5	7.7±4.0
Track lengths	28.0±21.6s	43.9±32.1s

3.2 Dataset Creation

For the creation of the MC-GTA dataset, we created a GTA plugin using the Script Hook V library[4] to interface with the GTAV environment, similarly to [17, 11]. The same plugin can be used by practitioners to create new ad-hoc scenarios. Specifically, the plugin implements an interface with a set of functionalities for creating, deleting, and navigating through the network of cameras in the desired scenarios. Compared to [17, 11], we added four more parameters to have finer control over the virtual environments: (i) *TimeOfDay*, which allows practitioners to select the time of the day; (ii) *MaxDistanceFromCamera*, determining the maximum distance at which framed vehicles should be annotated; (iii) *PerCameraNumFrames*, which sets the number of frames to be recorded per camera; (iv) *WeatherCondition*, to set the weather conditions.

The framed vehicle collection and annotation algorithm, shown in detail in Algorithm 1, consists of an iterative procedure that queries the video game's engine to obtain the vehicles present at a given time in the views of a network of cameras. Since GTA is a single-player video game, it does not support the synchronization of multiple cameras; therefore, we exploited a workaround similar to the one used in [17], recording camera frames one after the other by changing

[4] http://www.dev-c.com/gtav/scripthookv/

Algorithm 1 Recording and Annotating algorithm

```
Set TimeOfDay ← Get from configuration file
Set WeatherCondition ← Get from configuration file
Set PerCameraNumFrames ← Get from configuration file
Set MaxDistanceFromCamera ← Get from configuration file
Set Cameras ← Get from configuration file
Set SlowMotionSpeed ← Get from configuration file
FrameID = 0
while FrameID < PerCameraNumFrames do
   for each Camera in Cameras do
      CameraID ← Get ID of the current camera
      Set Camera as the main camera
      Teleport player to Camera coordinates
      Vehicles ← Get vehicles in the scene
      for each Vehicle in Vehicles do
         VehicleDistance ← Distance between camera and vehicle
         if VehicleDistance < MaxDistanceFromCamera then
            if Vehicle not occluded and Vehicle engine is on then
               LicensePlate ← Get vehicle license plate
               BoundingBox3D ← Get vehicle 3D bounding box coordinates
               BoundingBox2D ← Project 3D bounding box to 2D screen
               Save [FrameID, CameraID, LicensePlate,
               BoundingBox2D, BoundingBox3D] to file
```

camera positions and angles between shots. The drawback of this strategy is that a slight offset-time occurs each time the camera position is changed. To reduce this offset to only a few milliseconds, we activated the slow-motion mode by setting a playback speed, arguing that a few milliseconds delay is negligible in this applicative scenario.

Once all vehicles are extracted from a single camera view, they are filtered based on several criteria. These include vehicle-to-camera distance, which cannot be greater than the *MaxDistance*, and vehicles not present in the camera's field of view or occluded by buildings. Vehicles with their engine off are also excluded, given that fixed vehicles can never travel between different cameras.

In contrast to the ray-tracing methodology used by [17] to obtain vehicle bounding boxes, we simply projected the 3D bounding box coordinates of the vehicles into the camera frame. While this technique creates not tight-fitting 2D bounding boxes around vehicles, it (i) reduces computational complexity and enables scaling data acquisition to many cameras to possibly handle many recording hours, and (ii) mitigates the variability of the bounding-box shape, which in the case of ray-tracing tends to depend on the positioning of the rigging bones that changes across different vehicle classes. Furthermore, we consider as the unique ID that must remain consistent among the cameras the vehicle license plate. We provide the final format of the saved annotations concerning each collected frame in Table 2.

Table 2: Final MC-GTA dataset annotation format.

Metadata	**Description**
Timestamp	Time information about current frame
FrameID	Index frame to establish temporal order
LicensePlate	Vehicle unique license plate
$\text{Center}_{x,y}$	Coordinates of the *i-th* vehicle's 2D bounding box center
Width	Width of the *i-th* vehicle's 2D bounding box
Height	Height of the *i-th* vehicle's 2D bounding box
$\text{Vehicle}_{x,y,z}$	Coordinates of the *i-th* vehicle's 3D bounding box

4 Experimental Evaluation

In this section, we perform an assessment of our MC-GTA dataset, exploiting a SOTA MCVT approach[5] and our synthetic data as a testing benchmark. The adopted methodology was one of the top solutions proposed in the City-Scale Multi-Camera Vehicle Tracking track of the AI City 2022 Challenge[6], a popular competition about applying AI to several ITS tasks. It uses the YOLOv5 object detector [16], three ReID models for vehicle detection and feature extraction, the ByteTrack deep learning-based algorithm [31] for MOT, and a post-processing procedure that exploits geometrical and temporal information.

More in detail, we performed the experimental evaluation only over the second scenario of our MC-GTA dataset, since it resembles the real-world scenario addressed in the AI City 2022 Challenge. In order to exploit the above-mentioned MCVT approach and to be compliant with it, we manually defined a set of Regions Of Exclusion (ROE), i.e., we filtered out vehicle tracklets ending in some pre-defined regions, and we exploited temporal filters (TFs), i.e., we filtered out tracklets vehicles that are traced from temporal intervals not coherent with predefined temporal intervals computed on the basis of camera distances.

We measured the performance by using two golden standard MTMCT evaluators, i.e., the MOTA - Multi-Object Tracking Accuracy, the MOTP - Multi-Object Tracking Precision, and the IDF1, defined as:

$$MOTA = 1 - \frac{|FN| + |FP| + |IDSW|}{|GT_{Dets}|} \tag{1}$$

$$MOTP = \frac{1}{|TP|} \sum_{TP} S \tag{2}$$

$$IDF1 = \frac{|IDTP|}{|IDTP| + 0.5 \times |IDFN| + 0.5 \times |IDFP|} \tag{3}$$

where FN are False Negatives, FP are False Positives, TP are True Positives, IDSW are ID Switches, GT_{Dets} are the ground truth detections, IDTP is Identity True Positives, IDFP is Identity False Positives, IDFN is Identity False

[5] https://github.com/royukira/AIC22_Track1_MTMC_ID10
[6] https://www.aicitychallenge.org/

Table 3: The obtained results with and without the Region Of Exclusions (ROEs) and Temporal Filters (TFs).

	MOTA ↑	**MOTP** ↑	**IDF1** ↑	**IDP** ↑	**IDR** ↑	**FN** ↓	**IDSW** ↓
No ROEs, No TFs	71.11	24.58	76.33	84.58	69.54	11,267	27
2^{nd} Cam ROEs, TFs	73.14	24.58	77.51	84.58	71.52	9,925	27
ROEs, TFs	73.50	24.67	78.55	84.65	73.27	9,107	26

Negatives, and S is a similarity function (in this case the IoU) used to consider matches with a value greater than a threshold. For a finer analysis we also computed the IDP and the IDR defined as $IDP = \frac{|IDTP|}{|IDTP|+|IDFP|}$ and $IDR = \frac{|IDTP|}{|IDTP|+|IDFN|}$. We report the obtained results in Table 3, where we also show an ablation study of the considered methodology with and without the ROEs and the temporal filters used for the post-processing procedure.

As we can see, with the best setting we obtained an IDF1 score of 78.55, a value comparable with the ones obtained in the AI City 2022 Challenge, demonstrating that our synthetic benchmark can be a valuable tool for testing MCVT models. In particular, the same methodology applied to the scenarios of the AI City 2022 Challenge obtained an IDF1 score of 81.7. Furthermore, it is worth noting that the inclusion of the ROEs and the temporal filters implies an improvement in performance, as expected. Specifically, we obtained a reduction of 19% concerning the number of FNs and, consequently, a boost in the IDR of 4%, a slight increment in the IDF1 and MOTA metrics, and, finally, a small improvement in the number of IDSW.

5 Conclusion

In this paper, we introduced MC-GTA, a synthetic, freely available dataset gathered from the popular GTA5 video game. Gathering data from virtual worlds is an appealing solution contrasting the data scarcity problem since annotations are automatically generated by interacting with the graphical engine, thus considerably reducing human effort. Specifically, we collected images and labels from several overlapping and non-overlapping cameras located at various crossroads of simulated urban scenarios; labels correspond to bounding boxes localizing the vehicles present in the scenes, together with associated unique IDs persistent among the different video sources. Thus, MC-GTA is suitable for training/testing deep learning models performing multi-camera vehicle tracking, a challenging task extremely helpful for city-scale traffic analysis and security but whose real-world applicability is often constrained by the lack of data. Furthermore, future practitioners can use the code we released to create new ad-hoc scenarios resembling specific custom scenes they want to simulate. To assess our new dataset, we performed a performance evaluation exploiting a SOTA MCVT deep learning methodology using MC-GTA as a testing ground; the obtained

results showed that it can be a valuable benchmark that alleviates the need for real-world data. We hope the data and reference results will spark further activities in the field.

Acknowledgements

Supported by: MOST – Sustainable Mobility National Research Center, funded by the European Union Next-GenerationEU (Piano Nazionale di Ripresa E Resilienza (PNRR) – Missione 4 Componente 2, Investimento 1.4 – D.D. 1033 17/06/2022, CN00000023); AI4Media – A European Excellence Centre for Media, Society, and Democracy (EC, H2020 No. 951911); SUN – Social and hUman ceNtered XR (EC, Horizon Europe No. 101092612).

References

1. Amato, G., Ciampi, L., Falchi, F., Gennaro, C., Messina, N.: Learning pedestrian detection from virtual worlds. In: Lecture Notes in Computer Science, pp. 302–312. Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-30642-7_27
2. Benedetto, M.D., Carrara, F., Ciampi, L., Falchi, F., Gennaro, C., Amato, G.: An embedded toolset for human activity monitoring in critical environments. Expert Systems with Applications **199**, 117125 (aug 2022). https://doi.org/10.1016/j.eswa.2022.117125
3. 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 (sep 2016). https://doi.org/10.1109/icip.2016.7533003
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Computer Vision – ECCV 2020, pp. 213–229. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-58452-8_13
5. Carrara, F., Pasco, L., Gennaro, C., Falchi, F.: Learning to detect fallen people in virtual worlds. In: International Conference on Content-based Multimedia Indexing. ACM (sep 2022). https://doi.org/10.1145/3549555.3549573
6. Ciampi, L., Messina, N., Falchi, F., Gennaro, C., Amato, G.: Virtual to real adaptation of pedestrian detectors. Sensors **20**(18), 5250 (sep 2020). https://doi.org/10.3390/s20185250
7. Ciampi., L., Santiago., C., Costeira., J., Falchi., F., Gennaro., C., Amato., G.: Unsupervised domain adaptation for video violence detection in the wild. In: Proceedings of the 3rd International Conference on Image Processing and Vision Engineering - IMPROVE,. pp. 37–46. INSTICC, SciTePress (2023). https://doi.org/10.5220/0011965300003497
8. Ciampi, L., Santiago, C., Costeira, J., Gennaro, C., Amato, G.: Domain adaptation for traffic density estimation. In: Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. SCITEPRESS - Science and Technology Publications (2021). https://doi.org/10.5220/0010303401850195
9. Deschaud, J.: KITTI-CARLA: a kitti-like dataset generated by CARLA simulator. CoRR **abs/2109.00892** (2021)

10. Dosovitskiy, A., Ros, G., Codevilla, F., López, A.M., Koltun, V.: CARLA: an open urban driving simulator. In: 1st Annual Conference on Robot Learning, CoRL 2017, Mountain View, California, USA, November 13-15, 2017, Proceedings. Proceedings of Machine Learning Research, vol. 78, pp. 1–16. PMLR (2017)
11. Fabbri, M., Lanzi, F., Calderara, S., Palazzi, A., Vezzani, R., Cucchiara, R.: Learning to detect and track visible and occluded body joints in a virtual world. In: Computer Vision – ECCV 2018, pp. 450–466. Springer International Publishing (2018). https://doi.org/10.1007/978-3-030-01225-0_27
12. Foszner, P., Szczęsna, A., Ciampi, L., Messina, N., Cygan, A., Bizoń, B., Cogiel, M., Golba, D., Macioszek, E., Staniszewski, M.: CrowdSim2: An open synthetic benchmark for object detectors. In: Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. SCITEPRESS - Science and Technology Publications (2023). https://doi.org/10.5220/0011692500003417
13. Foszner, P., Szczęsna, A., Ciampi, L., Messina, N., Cygan, A., Bizoń, B., Cogiel, M., Golba, D., Macioszek, E., Staniszewski, M.: Development of a realistic crowd simulation environment for fine-grained validation of people tracking methods. In: Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. SCITEPRESS - Science and Technology Publications (2023). https://doi.org/10.5220/0011691500003417
14. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: YOLOX: exceeding YOLO series in 2021. arXiv preprint arXiv:2107.08430 (2021)
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. In: IEEE International Conference on Computer Vision, ICCV 2017. pp. 2980–2988. IEEE Computer Society (2017). https://doi.org/10.1109/ICCV.2017.322
16. Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., NanoCode012, Kwon, Y., Michael, K., TaoXie, Fang, J., imyhxy, Lorna, Yifu, Z., Wong, C., V, A., Montes, D., Wang, Z., Fati, C., Nadar, J., Laughing, UnglvKitDe, Sonck, V., tkianai, yxNONG, Skalski, P., Hogan, A., Nair, D., Strobel, M., Jain, M.: ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation (Nov 2022). https://doi.org/10.5281/zenodo.7347926
17. Kohl, P., Specker, A., Schumann, A., Beyerer, J.: The MTA dataset for multi target multi camera pedestrian tracking by weighted distance aggregation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE (jun 2020). https://doi.org/10.1109/cvprw50498.2020.00529
18. Li, Y., Hilton, A., Illingworth, J.: Towards reliable real-time multiview tracking. In: Proceedings 2001 IEEE Workshop on Multi-Object Tracking. IEEE Comput. Soc. https://doi.org/10.1109/mot.2001.937980
19. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **42**(2), 318–327 (2020). https://doi.org/10.1109/TPAMI.2018.2858826
20. Liu, C., Zhang, Y., Luo, H., Tang, J., Chen, W., Xu, X., Wang, F., Li, H., Shen, Y.D.: City-scale multi-camera vehicle tracking guided by crossroad zones. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE (jun 2021). https://doi.org/10.1109/cvprw53098.2021.00466
21. Liu, H., Tian, Y., Wang, Y., Pang, L., Huang, T.: Deep relative distance learning: Tell the difference between similar vehicles. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (jun 2016). https://doi.org/10.1109/cvpr.2016.238

22. Liu, X., Liu, W., Mei, T., Ma, H.: PROVID: Progressive and multimodal vehicle reidentification for large-scale urban surveillance. IEEE Transactions on Multimedia **20**(3), 645–658 (mar 2018). https://doi.org/10.1109/tmm.2017.2751966
23. Meinhardt, T., Kirillov, A., Leal-Taixe, L., Feichtenhofer, C.: TrackFormer: Multi-object tracking with transformers. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (jun 2022). https://doi.org/10.1109/cvpr52688.2022.00864
24. Qian, Y., Yu, L., Liu, W., Hauptmann, A.G.: Electricity: An efficient multi-camera vehicle tracking system for intelligent city. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 588–589 (2020)
25. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
26. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(6), 1137–1149 (jun 2017). https://doi.org/10.1109/tpami.2016.2577031
27. Staniszewski, M., Foszner, P., Kostorz, K., Michalczuk, A., Wereszczyński, K., Cogiel, M., Golba, D., Wojciechowski, K., Polański, A.: Application of crowd simulations in the evaluation of tracking algorithms. Sensors **20**(17), 4960 (sep 2020). https://doi.org/10.3390/s20174960
28. Tan, X., Wang, Z., Jiang, M., Yang, X., Wang, J., Gao, Y., Su, X., Ye, X., Yuan, Y., He, D., et al.: Multi-camera vehicle tracking and re-identification based on visual and spatial-temporal features. In: CVPR Workshops. pp. 275–284 (2019)
29. Wang, C., Bochkovskiy, A., Liao, H.M.: Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. CoRR **abs/2207.02696** (2022). https://doi.org/10.48550/arXiv.2207.02696
30. 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 (sep 2017). https://doi.org/10.1109/icip.2017.8296962
31. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X.: ByteTrack: Multi-object tracking by associating every detection box. In: Lecture Notes in Computer Science, pp. 1–21. Springer Nature Switzerland (2022). https://doi.org/10.1007/978-3-031-20047-2_1
32. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019)
33. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable DETR: deformable transformers for end-to-end object detection. In: 9th International Conference on Learning Representations, ICLR 2021. OpenReview.net (2021)