Improving Multi-Target Multi-Camera Tracking by Track Refinement and Completion

Andreas Specker[1,2,3] Lucas Florin[2,3] Mickael Cormier[1,2,3] Jürgen Beyerer[2,1,3]
[1]Karlsruhe Institute of Technology [2]Fraunhofer IOSB [3]Fraunhofer Center for Machine Learning
{andreas.specker,lucas.florin,mickael.cormier,juergen.beyerer}@iosb.fraunhofer.de

Abstract

Multi-camera tracking of vehicles on a city-wide level is a core component of modern traffic monitoring systems. For this task, single-camera tracking failures are the most common causes of errors concerning automatic multi-target multi-camera tracking systems. To address these problems, we propose several modules that aim at improving single-camera tracklets, e.g., *appearance-based tracklet splitting, single-camera clustering, and track completion. After these track refinement steps, hierarchical clustering is used to associate the enhanced single-camera tracklets. During this stage, we leverage vehicle re-identification features as well as prior knowledge about the scene's topology. Last, the proposed track completion strategy is adopted for the cross-camera association task to obtain the final multi-camera tracks. Our method proves itself competitive: With it, we achieved 4th place in track 1 of the 2022 AI City Challenge.*

1. Introduction

Multi-Target Multi-Camera Tracking (MTMCT) systems aim at tracking multiple targets, in our case vehicles, as they move through a scene captured by numerous cameras. This means localizing and tracking targets in each camera feed and identifying the other instances of the same target in the other camera feeds. Applications include traffic flow analysis and traffic signal time planning.

An MTMCT system consists of two core components: First, a single-camera tracking pipeline that localizes all relevant objects in each video frame and connects these detections across time into tracklets. Second, an inter-camera association module matches tracks belonging to the exact vehicle across different cameras. In a complex real-world traffic scene, the distance and orientation of the objects w.r.t. to the camera vary enormously between cameras. Also, different cameras have different technical characteristics. These properties make both sub-tasks of MTMCT especially challenging in this context. To solve these problems, most of the Single-Camera Tracking (SCT) methods follow the *tracking-by-detection* paradigm [2–4, 49, 51, 58, 60]: First, a set of detections is generated for each video frame independently. Afterward, these detections are linked together to form tracks based on a similarity metric. Usually, this similarity metric considers visual features extracted by a re-identification (re-ID) model together with position information.

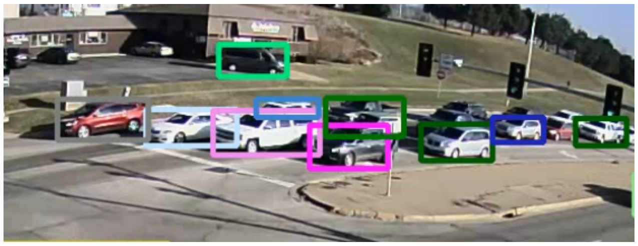


Figure 1. **Challenges of tracking vehicles within a camera view –** Tracking vehicles in real-world scenarios is challenging due to heavy occlusions, *e.g.*, when vehicles wait at a traffic light. Some vehicles are not detected while they are occluded which often leads to track fragmentation or identity switches.

This has been proven to be a powerful approach in ideal environments where targets are visible in their entirety. However, in real-world environments, this is often not the case: Vehicles occlude each other, particularly in crowded scenes (see Fig. 1) at traffic lights or when vehicles overtake each other. As a result, single-camera tracklets are divided into multiple fragments or the tracklet switches from one vehicle to another. To solve this, we develop a track refinement module consisting of several mechanisms to further improve single-camera tracklets obtained by the JDE tracker [57]. Besides filtering methods such as background filtering and track filtering, splitting approaches are leveraged to reduce the number of identity switches. For instance, we propose to employ K-Means clustering to split tracklets in which multiple vehicles occur based on the visual appearance. Visual information is also used to reconnect multiple track fragments belonging to the same vehicle and therefore tackle the problem of fragmentation. Moreover, a so-called track completion component finalizes tracklets based on the knowledge that cars cannot suddenly appear or disappear in the middle of the camera view.

Following many works from related literature [25,28,44,47], visual features are the core component of our Multi-Camera Tracking (MCT) approach since they efficiently reidentify vehicles across cameras. We use a background subtraction model similar to [47] to handle occlusions from static objects. This model removes detections occluded by static objects such as traffic lights. It also discards tracks located entirely in the background, such as parked vehicles. Besides visual features, we also consider the structure of the scene. Our scene model includes information about the topology of the traffic camera network as well as temporal information. This way we prevent implausible matches of tracks across cameras. Finally, we also adopt the proposed track completion approach to the multi-camera tracking task.

Our main contributions can be summarized as follows:

- We develop a robust MTMCT system that leverages topological and temporal information and is easily extendable.
- We address the primary error source, *i.e.*, single-camera tracking errors caused by occlusions, through our track refinement module.
- We propose an explicit track completion mechanism that improves single-camera tracking results and is also applicable and beneficial for the cross-camera tracking task.

2. Related Work

2.1. Vehicle Detection

Vehicle detection is a domain-specific sub-task of object detection, which is often associated with autonomous driving and smart cities. In recent years the contribution of several large datasets for vehicle detection and tracking [6, 10, 11, 17, 67] has facilitated the adoption of object detection architectures such as SSD [29], YOLO [5, 40], and Faster R-CNN [42] to the vehicle detection task. In fact, several challenges [35, 36, 67] have been proposed, which often resulted in the adoption of variants of the YOLO-based models [18,30,54,55] which offer a favorable trade-off between accuracy and computational efficiency. While adopting a large offline ensemble of detectors in such challenges is a widely used approach [47], the winner of track 3 of the 2021 AI City Challenge [28] used a single YOLOv5 [23] detector. Therefore, we follow this practice and adopt YOLOv5 as our sole detector in this work.

2.2. Vehicle Re-identification

Vehicle re-ID has attracted increasing attention in the computer vision community in the context of intelligent transportation systems. Although re-ID has been researched for a long time, the vehicle re-ID task is still extremely challenging. Similar to person re-ID, high intra-class and small inter-class variances are prominent problems in vehicle re-ID due to different camera perspectives and similar vehicle appearances [22]. Several earlier works attempted to design invariant features for vehicles in different scenes. In [16], a descriptor with a global feature invariant to affine transformations and global illumination changes is designed. In [12], the Local Binary Patterns (LBP) and Local Variance (VAR) are applied to local grid cells of the image for extremely low-resolution vehicle re-ID. More recent works apply deep learning and achieve competitive results by learning global features using a bag of tricks [32,33,64]. An Identity Unrelated Information Decoupling paradigm is proposed in [31] to learn invariant features of the vehicle with the same ID in different scenes using camera perspective and background information as two kinds of identity-unrelated information. Global feature learning does not rely on prior knowledge such as a specific structure, *i.e.*, a body structure. Therefore, methods from the widely active field of person re-ID may be adapted to vehicle re-ID. While complex methods aim at making use of the particular structure of the domain with attention mechanisms [8, 65] or using auxiliary high-level semantic attributes [27, 46], similar concepts are available for the task of vehicle re-ID as well [9, 14, 24]. In this work, we rely on a global feature learning approach for our vehicle re-ID component, as in [19,28,36,47].

2.3. Single-camera Tracking

Most SCT approaches use the tracking-by-detection paradigm, *i.e.*, the task is divided into a detection step using an efficient detector followed by an association step, in which detections of the same targets are matched based on a similarity measure [2–4, 41, 45, 49, 51, 58–61, 63]. Most methods combine position and motion information [2, 3], and additionally re-ID features [4, 49, 51, 58] or other cues such as pose information [51, 60]. While these approaches are highly effective, the temporal context available in videos is often considered less. In [38], the TPM algorithm is proposed, which efficiently combines multiple short subtrajectories into a long trajectory and, using trajectory context, mitigates missing detections. TNT [56] uses a graph-based model to incorporate temporal and appearance information for tracking simultaneously.

Nonetheless, a recent line of works tightens the link between detection and tracking by extending object detectors to trackers [1, 66], incorporating tracking results as prior knowledge for detection [15, 66] or 3D CNNs to de-

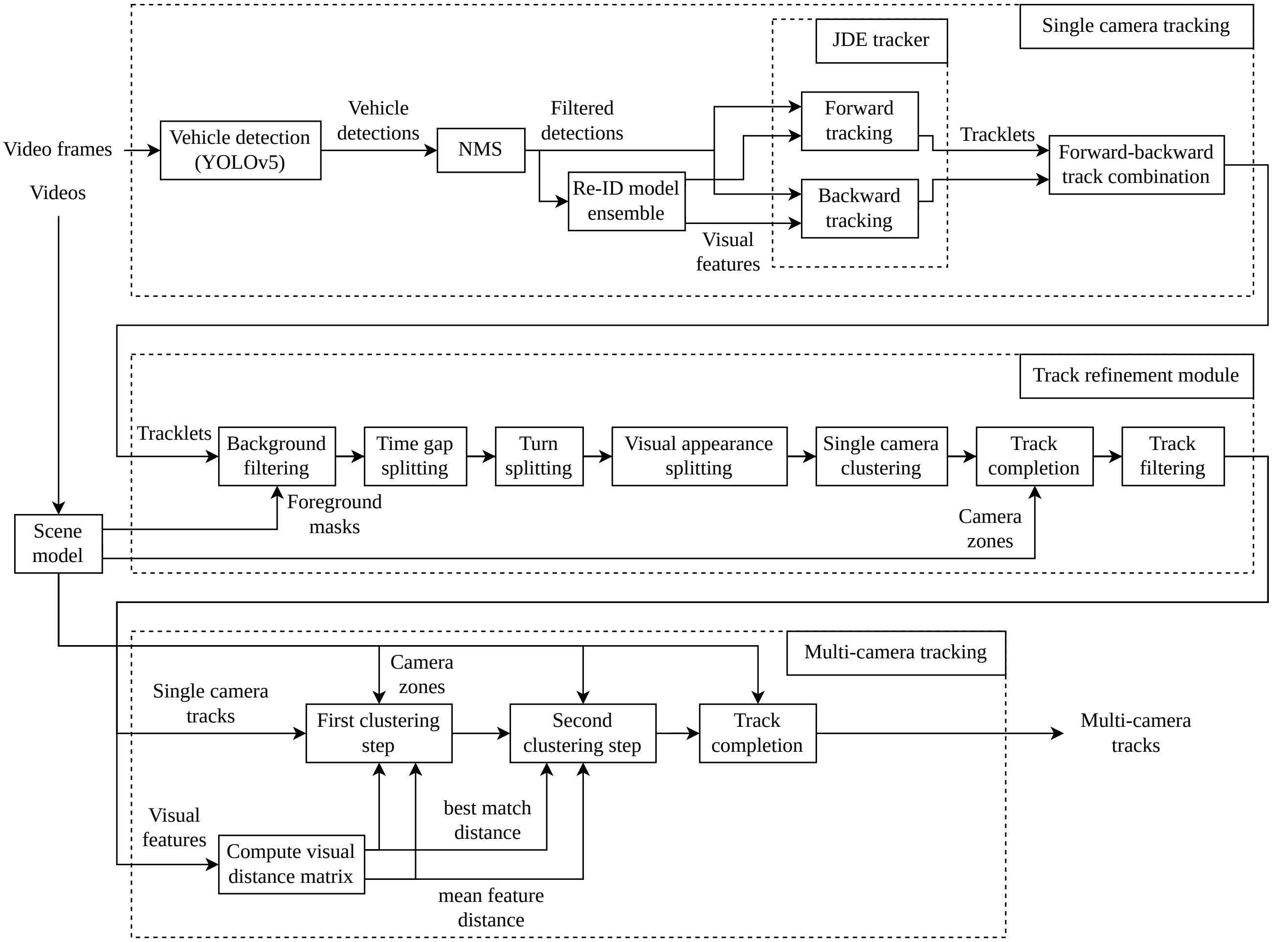


Figure 2. **System diagram of the proposed MTMCT system –** First, we generate tracklets by looking at vehicle detections in each video feed. Then, these tracklets are filtered, split, clustered and completed, resulting in high quality single-camera tracks. Finally, these tracks are joined across cameras into multi-camera tracks.

tect tracklets [37]. TrackFormer [34] leverages recent advances in vision transformers to address occluded, missing, or noisy detection. In addition, TransTrack [50] further improves performance by combining the object query from DETR [7] and a track query.

2.4. Multi-camera Tracking

Typical MTMCT pipelines include detection, MTSCT, and cross-camera clustering [21, 25, 44, 53]. Recent works [20, 21, 25, 39] use additional external information about the camera setup to improve their pipeline. In [21, 28, 47] the scene topology is used to prevent infeasible cross-camera transitions, camera adjacency in [28, 39, 47], the movement directions are used in [20] to determine the plausibility of camera transitions while camera-specific areas are defined in [21] to determine the possibility of tracks appearing in multiple cameras. Specker *et al.* [47] propose an occlusion-aware approach to discard obstacle-occluded bounding boxes and overlapping tracks. Last year's winners of AI City Challenge [35] especially focused on spatio-temporal information and traffic rules [28, 30, 63]. Inspired by these recent trends, we adapt a clustering approach to cluster tracks from adjacent cameras and leverage information about the scene's topology.

3. Methods

3.1. Overview

Our multi-target multi-camera vehicle tracking system with its components is depicted in Figure 2. In general, it consists of three main parts.

The first one is the single-camera tracking stage which takes videos as input and generates a set of single-camera tracklets for each input video. Specifically, vehicles are detected, and appearance features are extracted using a vehicle re-identification approach. Subsequently, a single-camera tracker is applied in order to group detections into single-camera tracklets.

The second part is the Track Refinement Module (TRM) which contains several post-processing steps that aim at reducing the number of identity switches and fragmentations.

Finally, multi-camera tracking is performed by associating the single-camera tracklets provided by the previous stages. Two rounds of clustering are applied, followed by our track completion module to finalize tracks that must have predecessors or successors since they start or end in the middle of camera views. All the modules and components are described in detail in the following sections.

3.2. Detection

To be able to track vehicles' routes within a camera network accurately, it is essential to detect them reliably. The detection stage should not miss vehicles since subsequent tracking, and filtering stages can suppress false positives or multiple detections for the same instance. In contrast, it is more challenging to interpolate missing bounding boxes afterward. We employ the YOLOv5 [23] single-stage detector due to its good trade-off between speed and accuracy. It is applied to each video frame to obtain the bounding boxes of the occurring vehicles and a confidence score that expresses the detector's certainty. To avoid false negatives, *i.e.*, missing vehicles due to the reasons mentioned above, we rely on a low confidence threshold for detections of 0.1. Moreover, we found that using the off-the-shelf YOLOv5x6 model pre-trained on the COCO [26] dataset is sufficient to achieve promising performance and detect all vehicles. Training on external data or fine-tuning the model on the challenge dataset is not required. Since the COCO [26] dataset considers a variety of different classes which are not relevant to the multi-camera tracking task, classes such as person, etc., are discarded, and only detections for different types of vehicles are kept. Using a low detection threshold leads to many double detections, so non-maximum suppression (NMS) is applied directly after detection as a first filtering step. Strongly overlapping bounding boxes measured by the intersection-over-union (IoU) in the video frames are filtered to reduce the number of bounding boxes to one per vehicle instance. Subsequently, the bounding boxes for each frame in the videos are forwarded to a vehicle re-ID network to extract meaningful appearance descriptors.

3.3. Vehicle Re-identification

Vehicle re-ID is the task of extracting meaningful feature representations of vehicles' visual appearances to assess the similarity between different vehicles based on a distance measure. This pipeline stage constitutes an integral part of single-camera tracking and multi-camera association. Many approaches and best practices can be transferred from the widely studied topic of person re-ID.

However, vehicle re-ID raises additional challenges since multiple, almost identical cars from the same make, model, and color may appear in the scene. As a result, models have to be able to distinguish vehicles based on small-scale visual features, *e.g.*, scratches, dirt, or special equipment. Data augmentation and synthetic data are essential since the challenge dataset is limited concerning the number of cars and trucks and therefore lacks diversity. Similar to the strong baseline for re-ID [32], we train different models using real-world data, synthetic data, as well as image data that was transferred from the synthetic to the real-world domain using a generative adversarial network [13]. This procedure leads to diverse appearance representations which generalize well when used as an ensemble. In general, we rely on a global approach which means that we aim at learning one global feature vector instead of extracting multiple embeddings for different parts of the vehicles. Such approaches are lightweight, do not tend to overfit, and deliver robust results.

In detail, we train ResNet-101 IBN-A and ResNeXt-101 IBN-A models with an input image size of 384×384. A fully-connected classification layer is appended to these backbone networks with as many output neurons as instances in the training dataset. During inference, *i.e.*, feature extraction, the classification layer is omitted. As done in many works, the stride parameter of the last pooling layer is set to 1 to keep fine-grained details. Analogous to current state-of-the-art approaches to re-identification, a combination of the cross-entropy classification loss function and the metric learning triplet loss function is used. While the former aims to identify the vehicles, the latter helps to learn features being close in embedding space for samples of the same class and far for similar vehicles originating from different classes. We extract 2048-dimensional feature vectors for each bounding box extracted in the previous detection stage. Bounding boxes and corresponding appearance representations serve as input for the subsequent single-camera tracking.

3.4. Single-camera Tracking

In this processing step, detections within the video frames are combined to form so-called tracklets. Tracklets represent the spatial-temporal trajectories of vehicles crossing a single camera view. Our method uses the JDE tracker [57] that associates detections to tracklets building on motion tracking using a Kalman filter and visual similarity based on the extracted re-identification embeddings. The tracker outputs a set of tracklets containing the corresponding bounding boxes and feature vectors for each time step the tracklet is visible. Situations in which vehicles lower their velocity and come to a stop at traffic lights are a significant challenge in the dataset. The motion estimates of the Kalman filter get worse, and due to heavy occlusions, many tracks get fragmented. To reduce the negative impact of such situations, we perform single-camera tracking in both temporal directions [48]. First, the video is processed from start to end and then from end to start. By searching for the best overlap between tracklets from the forward and backward tracking and keeping the larger one, robustness against fragmentation is greatly increased.

3.5. Single-camera Track Refinement

Although the JDE single-camera tracker delivers primarily promising results, some systematic error sources exist. On the one hand, many tracks are fragmented due to occlusions caused by other vehicles or obstacles. On the other hand, identity switches occur, especially in scenes when cars stop at traffic lights or overtake each other. We have developed and combined several methods to post-process and greatly enhance single-camera tracklets in our TRM. Each of them is explained in the following.

Background filtering Analogous to the work of Specker et al. [47], we filter false positive detections based on a background-foreground segmentation model. First, foreground regions are determined by computing areas that do not change during the video. Detections or even entire tracklets are discarded if they overlap with the static background by more than 50%.

Time gap splitting Time gaps between associated detections in single-camera tracklets may indicate that the vehicle got lost and re-identified after, *i.e.*, it was occluded. Since this may lead to identity switches, we examine such time gaps within single-camera tracklets. If the gap is sufficiently large and the re-ID features representing the visual appearance change too much, tracklets will be split and divided into two new ones.

Turn splitting Another common source of error is identity switches that occur when a vehicle leaves a camera view and a second one enters it nearby. In some cases, the tracklet of the leaving vehicle is not finished, but instead, it is resumed by the entering one. This problem is straightforward to remedy by examining the direction of travel and splitting the tracklet if it is suddenly reversed. In contrast to, *e.g.*, people, it is improbable that a vehicle performs such a maneuver due to traffic rules.

Visual appearance splitting To further reduce the number of identity switches, we propose a visual appearance splitting mechanism. For this, K-Means clustering is applied to the embeddings of a tracklet to group them into two clusters. If two different vehicles appear within the same tracklet, the respective detections should be assigned to different clusters. Subsequently, the distance between the cluster centers is leveraged to assess whether the tracklet shows multiple vehicles. If the Cosine distance between the cluster centers exceeds 0.68, the tracklet will be split at the point after which all detections were assigned to the same cluster. The high threshold ensures that correct tracklets are not divided.

Single-camera clustering After splitting tracklets to correct identity switches, we use two rounds of Agglomerative Clustering to merge fragments showing the same vehicle. To do this, a distance matrix of size $N_c \times N_c$ is constructed where N_c equals the number of single-camera tracklets found in camera c. Each element at position (x, y) of the matrix corresponds to the Cosine distance between the mean feature vectors of two tracklets T_x and T_y. In the first round, no constraints are applied to combine tracklets that overlap in time and position. This leads to great improvement, especially in waiting situations with heavy occlusions near traffic lights. A low distance threshold of 0.1 is used for the first clustering to avoid false-positive combinations. Afterward, tracklet clusters are merged. If some of the tracklets overlap in time, detections of the longer tracklet are kept, and detections of the shorter one are discarded. Before the second clustering, the distance matrix is re-computed using the newly created tracklets. In contrast to the first round, constraints w.r.t. time overlap, time gaps between fragments, and direction of travel are applied. This allows leveraging a higher distance threshold of 0.3 to reduce the number of fragments further.

Single-camera track completion We propose a track completion module since vehicles follow strict traffic rules and are unlikely to disappear in the middle of the camera view suddenly. As long as vehicles do not reach the frame boundaries or the video ends, there must be a successor tracklet. We aim at connecting single-camera tracklets based on these prior assumptions. For each tracklet, the following actions are performed:

1. Check if the tracklet is already finished, *i.e.*, enters and leaves the camera or starts/ends with the start or end of the video.
2. If not, search for possible predecessors and successors based on direction, time gap, distance, and visual similarity.
3. Merge tracklets with the best matching predecessor and/or successor.

After the track completion refinement step, most single-camera tracking errors are corrected, and the number of identity switches and fragmentations is greatly lowered. We conclude the refinement pipeline by applying several filtering methods. This includes omitting short tracklets with less than five detections, tracklets that cannot appear in another camera, as well as static tracks that do not change position.

3.6. Cross-camera Association

After the single-camera tracklets have been finalized, the next step is the association of tracklets from different cameras that show the identical vehicle to obtain multi-camera tracks.

Multi-camera clustering Many works [25, 28, 47] regarding multi-camera tracking solve the task by hierarchical clustering, so we also rely on this approach. Like many other works [21,28,47], we make use of so-called zones. In detail, we leverage four different zones: one for each possible direction vehicles can come from or go to, respectively. This scene model is helpful to constrain impossible transitions of vehicles between cameras. Inspired by [28], two rounds of clustering are applied. In the first one, tracklets are clustered separately for each possible transition between the cameras, and in the second one, all tracks from adjacent cameras are clustered. In contrast to [28], we modify the distance metric and do not solely rely on the Cosine distance between the tracks' mean features for clustering. [47] indicates that it is beneficial to consider the distance between the most similar detection pair between two tracks in addition to the distance of mean features. So, we build the distance matrix by multiplying both aforementioned distances. The idea is that tracklets may be merged when either the visual appearances of vehicles across whole single-camera tracklets are very similar or when there is one strong agreement between detections of the tracklets. We set distance values between tracks that cannot belong to the same vehicle due to invalid zone transitions or impossible transition durations to infinity. This limits the search space, and thus false-positive associations are avoided. After the second clustering step, tracklets within the same clusters are merged to form multi-camera tracks.

Multi-camera track completion Similar to the single-camera track refinement stage, resulting multi-camera tracks are post-processed by our track completion module. For instance, some tracks may not be matched by the previous clustering step since they do not start or end in transition zones. The track completion algorithm is identical to the single-camera version, but instead of searching for candidates within the camera, the search space is composed of tracks from adjacent cameras.

Approach	IDF1	IDP	IDR
Baseline	76.42	78.80	74.19
+ TRM	82.07	86.08	78.41
+ multiplying best match distance	83.12	86.45	80.04
+ multi-camera track completion	**83.48**	**86.74**	**80.46**

Table 1. **Ablation Study –** Comparison of the influence of different modules on the overall multi-camera tracking performance. The proposed track refinement module leads to the most significant improvement. The use of the combined distance and of the multi-camera track completion component brings lower but still significant improvements.

4. Evaluation

Experimental results are presented in this section. After briefly introducing the CityFlowV2 [52] dataset and evaluation metrics, we provide an ablation study, qualitative results, and the final challenge ranking.

4.1. Datasets

We used the two datasets allowed for track 1 of the AI City Challenge 2022: the real-world CityFlowV2 [52] dataset and the synthetic VehicleX [62] dataset.

CityFlowV2 This dataset is a benchmark for city-scale MTMCT. The training and validation sets consist of a high number of video feeds covering intersections of a road network of a U.S. city. Different city areas are represented, such as residential areas and highways. Some intersections are covered by multiple overlapping video feeds. However, the test set only covers intersections along a single stretch of a highway, with only one camera for each intersection. This makes the tracking task somewhat simpler, but the training and validation sets can hardly be used to predict tracking performance on the test set.

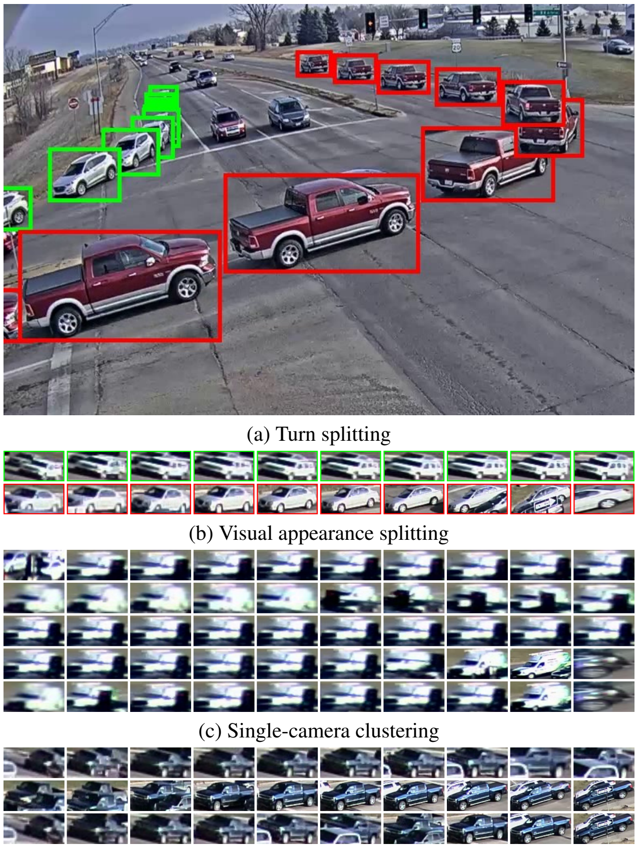
VehicleX The VehicleX dataset is a synthetic dataset built from rendered 3D models of vehicles. The synthetic images are placed on background images taken from a real-world dataset. It can be used to augment a smaller, real-world dataset such as CityFlowV2.

4.2. Evaluation Metrics

The challenge ranking is based on the IDF1 score. This is a comprehensive metric that measures both SCT and MCT performance. In our ablation study, we also provide the IDP (identity precision) and IDR (identity recall) metrics [43].

4.3. Results & Discussion

This section provides qualitative and quantitative results for different stages of our pipeline.



(a) Turn splitting

(b) Visual appearance splitting

(c) Single-camera clustering

(d) Single-camera track completion

Figure 3. **Track refinement module results –** Qualitative evaluation of our TRM. Figure 3a visualizes a case where the tracklet transitions from a vehicle leaving the scene to one driving into the scene at the frame boundary. By analyzing the direction of travel, our TRM separates the tracklets. In Figure 3b, each row of images stands for one tracklet after the split. One can observe that our clustering approach is able to determine and correct the identity switch. In Figures 3c and 3d, the last row visualizes the merged tracklet and the rows above visualize separate track fragments. The resulting tracklets are greatly enhanced by combining several sub-tracklets showing the same vehicle.

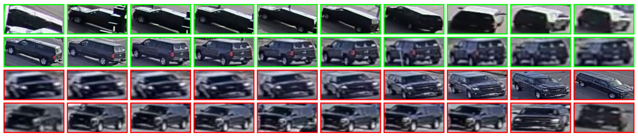


Figure 4. **Cross-camera track completion –** Qualitative results of our multi-camera track completion component. It is capable of connection track fragments even if the visual appearance differs due to varying lighting conditions and viewing angles.

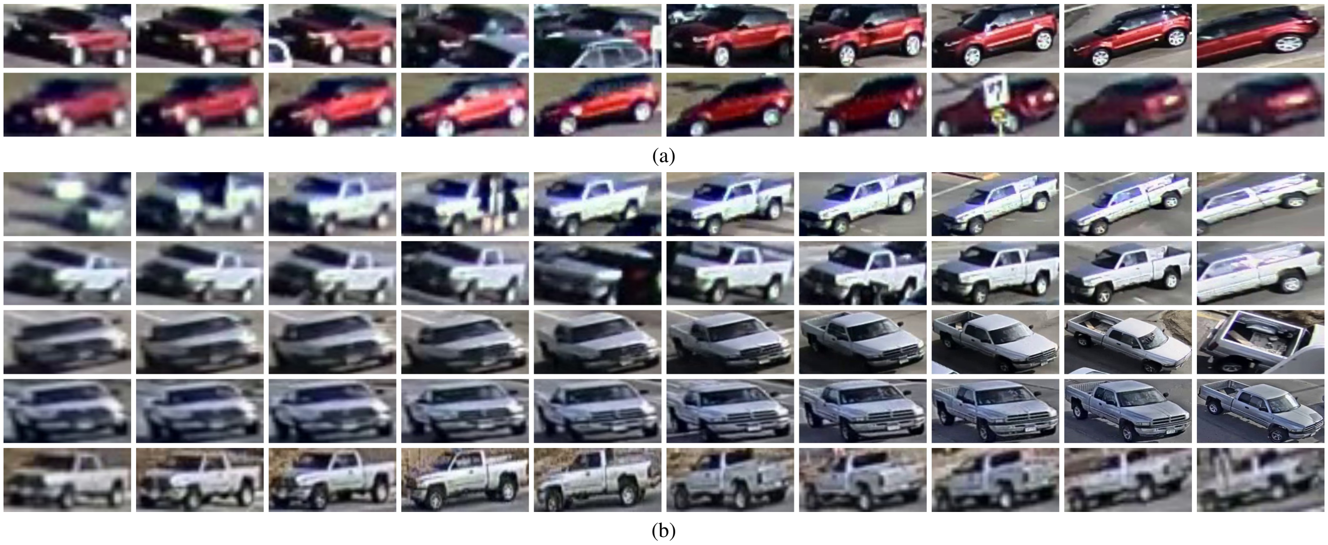
Rank	Team ID	IDF1	Rank	Team ID	IDF1
1	28	84.86	11	114	81.27
2	59	84.37	12	57	80.95
3	37	83.71	13	5	79.55
4	**Ours**	**83.48**	14	18	78.79
5	70	82.51	15	38	75.53
6	36	82.18	16	49	74.57
7	15	81.71	17	109	72.62
8	118	81.66	18	4	72.55
9	110	81.40	19	141	62.12
10	94	81.29	20	16	60.94

Table 2. **Challenge results –** Challenge results on the official test set.

TRM Figure 3 visualizes qualitative examples for selected components of the proposed TRM. One can observe that the turn splitting component (see Figure 3a) is able to detect sudden changes of direction and split the tracklets accordingly. Furthermore, the proposed clustering approach to correct identity switches is working as expected, as shown in Figure 3b. Detections belonging to different vehicles are assigned to different clusters and are subsequently divided into separate tracklets. To reduce the number of fragmented tracklets, we use single-camera clustering (see Figure 3c) and the introduced track completion module (see Figure 3d). The single-camera clustering example shows a partly occluded vehicle that appears mainly in the background with a small size. As a result, the single-camera tracklet is divided into multiple fragments. Our clustering approach is capable of reuniting the fragments based on visual similarity. The single-camera tracklet completion module handles cases where the visual similarity is lower, *e.g.*, due to different distances from the camera. The sample pictured in Figure 3d visualizes such a case. It shows that using the information that a tracklet is not completed yet and thus a successor must exist, tracklets can be merged accurately based on movement information.

Multi-camera track completion Regarding multi-camera track completion, we present an example of a vehicle performing a U-turn and driving back the way it was coming from in Figure 4. Different colors stand for other multi-camera tracks before the module is applied. Due to different lighting conditions and viewing angles, the tracks were not merged during the cross-camera clustering. Leveraging movement information and less strict constraints allows the connection of the track fragments and, therefore, the improvement of the resulting multi-camera tracks.

Ablation Table 1 presents the impact of the TRM module, the multiplied distances, and the multi-camera track completion on the multi-camera tracking accuracy. The results



(a)

(b)

Figure 5. **Qualitative Multi-camera Tracking Results –** Each row shows a single-camera tracklet from a different camera. The upper track shows the effectiveness of our TRM. The lower track shows that with our approach, even with variation in viewing angles and lighting conditions, vehicles can be tracked through multiple cameras.

indicate that single-camera track refinement is the most crucial component for increasing performance. Single-camera tracklets constitute the base of the whole multi-camera tracking pipeline, and errors in this stage impede robust cross-camera association. Moreover, the results show that post-processing resulting multi-camera tracks using prior knowledge about traffic rules further enhances the results.

Challenge results Table 2 compares our approach with the other challenge participants. We achieved fourth place with an IDF1 score of 83.48%.

Final results Last but not least, we give some final qualitative results of our tracking approach in Figure 5. Each row represents a single-camera sub-tracklet from the multi-camera track. The first example shows a track that benefits from our single-camera clustering and track completion. The car is almost entirely overlapped by another vehicle in the top row, leading to fragmentations after the single-camera tracking stage. Our track refinement strategy corrects the error before cross-camera clustering. The example shown in Figure 5b proves the capability of our approach to robustly track the routes of vehicles across multiple cameras and thus allow applications such as automatic traffic monitoring.

5. Conclusion

In this work, we have proposed a multi-target multi-camera vehicle tracking system that focuses on improving single-camera tracklets to enhance the overall performance. In addition, the featured track completion strategy is also applied to the cross-camera association task. Experimental validation proved the effectiveness of the proposed approaches. In summary, the tracking system achieves an IDF1 score of 83.48%, which corresponds to the fourth position in track 1 of the AI City Challenge 2022.

References

[1] P. Bergmann, T. Meinhardt, and L. Leal-Taixé. Tracking without bells and whistles. In *Int. Conf. Comput. Vis.*, pages 941–951, 2019. 2

[2] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and realtime tracking. In *IEEE Int. Conf. Image Process.*, pages 3464–3468, 2016. 1, 2

[3] E. Bochinski, V. Eiselein, and T. Sikora. High-speed tracking-by-detection without using image information. In *IEEE Int. Conf. Adv. Video Sign. Surv.*, 2017. 1, 2

[4] E. Bochinski, T. Senst, and T. Sikora. Extending iou based multi-object tracking by visual information. In *IEEE Int. Conf. Adv. Video Sign. Surv.*, 2018. 1, 2

[5] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 2

[6] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 2

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3

[8] T. Chen, S. Ding, J. Xie, Y. Yuan, W. Chen, Y. Yang, Z. Ren, and Z. Wang. Abd-net: Attentive but diverse person re-

identification. In *Int. Conf. Comput. Vis.*, pages 8351–8361, 2019. 2

[9] T.-S. Chen, C.-T. Liu, C.-W. Wu, and S.-Y. Chien. Orientation-aware vehicle re-identification with semantics-guided part attention network. In *Eur. Conf. Comput. Vis.*, pages 330–346, 2020. 2

[10] Yukyung Choi, Namil Kim, Soonmin Hwang, Kibaek Park, Jae Shin Yoon, Kyounghwan An, and In So Kweon. Kaist multi-spectral day/night data set for autonomous and assisted driving. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):934–948, 2018. 2

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2

[12] Mickael Cormier, Lars Wilko Sommer, and Michael Teutsch. Low resolution vehicle re-identification based on appearance features for wide area motion imagery. In *2016 IEEE Winter Applications of Computer Vision Workshops (WACVW)*, pages 1–7. IEEE, 2016. 2

[13] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 4

[14] V. Eckstein, A. Schumann, and A. Specker. Large scale vehicle re-identification by knowledge transfer from simulated data and temporal attention. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 616–617, 2020. 2

[15] C. Feichtenhofer, A. Pinz, and A. Zisserman. Detect to track and track to detect. In *Int. Conf. Comput. Vis.*, pages 3057–3065, 2017. 2

[16] Andres Frias-Velazquez, Peter Van Hese, Aleksandra Pižurica, and Wilfried Philips. Split-and-match: A bayesian framework for vehicle re-identification in road tunnels. *Engineering Applications of Artificial Intelligence*, 45:220–233, 2015. 2

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[18] Synh Viet-Uyen Ha, Nhat Minh Chung, Tien-Cuong Nguyen, and Hung Ngoc Phan. Tiny-pirate: A tiny model with parallelized intelligence for real-time analysis as a traffic counter. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4119–4128, 2021. 2

[19] S. He, H. Luo, W. Chen, M. Zhang, Y. Zhang, F. Wang, H. Li, and W. Jiang. Multi-domain learning and identity mining for vehicle re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 582–583, 2020. 2

[20] Y. He, J. Han, W. Yu, X. Hong, X. Wei, and Y. Gong. City-scale multi-camera vehicle tracking by semantic attribute parsing and cross-camera tracklet matching. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 576–577, 2020. 3

[21] H.-M. Hsu, T.-W. Huang, G. Wang, J. Cai, Z. Lei, and J.-N. Hwang. Multi-camera tracking of vehicles based on deep features re-id and trajectory-based camera link models. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 416–424, 2019. 3, 6

[22] Yi Jin, Chenning Li, Yidong Li, Peixi Peng, and George A Giannopoulos. Model latent views with multi-center metric learning for vehicle re-identification. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1919–1931, 2021. 2

[23] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, TaoXie, Jiacong Fang, imyhxy, Kalen Michael, Lorna, Abhiram V, Diego Montes, Jebastin Nadar, Laughing, tkianai, yxNONG, Piotr Skalski, Zhiqiang Wang, Adam Hogan, Cristi Fati, Lorenzo Mammana, AlexWang1900, Deep Patel, Ding Yiwei, Felix You, Jan Hajek, Laurentiu Diaconu, and Mai Thanh Minh. ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference, Feb. 2022. 2, 4

[24] P. Khorramshahi, N. Peri, J.-c. Chen, and R. Chellappa. The devil is in the details: Self-supervised attention for vehicle re-identification. In *Eur. Conf. Comput. Vis.*, pages 369–386, 2020. 2

[25] P. Köhl, A. Specker, A. Schumann, and J. Beyerer. The mta dataset for multi-target multi-camera pedestrian tracking by weighted distance aggregation. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 1042–1043, 2020. 2, 3, 6

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4

[27] Y. Lin, L. Zheng, Z. Zheng, Y. Wu, Z. Hu, C. Yan, and Y. Yang. Improving person re-identification by attribute and identity learning. *Pattern Recognition*, 95:151–161, 2019. 2

[28] Chong Liu, Yuqi Zhang, Hao Luo, Jiasheng Tang, Weihua Chen, Xianzhe Xu, Fan Wang, Hao Li, and Yi-Dong Shen. City-scale multi-camera vehicle tracking guided by crossroad zones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4129–4137, 2021. 2, 3, 6

[29] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, et al. Ssd: Single shot multibox detector. In *ECCV*, 2016. 2

[30] Jincheng Lu, Meng Xia, Xu Gao, Xipeng Yang, Tianran Tao, Hao Meng, Wei Zhang, Xiao Tan, Yifeng Shi, Guanbin Li, et al. Robust and online vehicle counting at crowded intersections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4002–4008, 2021. 2, 3

[31] Zefeng Lu, Ronghao Lin, Xulei Lou, Lifeng Zheng, and Haifeng Hu. Identity-unrelated information decoupling model for vehicle re-identification. *IEEE Transactions on Intelligent Transportation Systems*, 2022. 2

[32] Hao Luo, Weihua Chen, Xu Xianzhe, Gu Jianyang, Yuqi Zhang, Chong Liu, Jiang Qiyi, Shuting He, Fan Wang, and Hao Li. An empirical study of vehicle re-identification on the ai city challenge. In *Proc. CVPR Workshops*, 2021. 2, 4

[33] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2019. 2

[34] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. *arXiv preprint arXiv:2101.02702*, 2021. 3

[35] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Yue Yao, Liang Zheng, Pranamesh Chakraborty, Christian E. Lopez, Anuj Sharma, Qi Feng, Vitaly Ablavsky, and Stan Sclaroff. The 5th ai city challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021. 2, 3

[36] M. Naphade, S. Wang, D. C. Anastasiu, Z. Tang, M.-C. Chang, X. Yang, L. Zheng, A. Sharma, R. Chellappa, and P. Chakraborty. The 4th ai city challenge. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 626–627, 2020. 2

[37] B. Pang, Y. Li, Y. Zhang, M. Li, and C. Lu. Tubetk: Adopting tubes to track multi-object in a one-step training model. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6308–6318, 2020. 3

[38] Jinlong Peng, Tao Wang, Weiyao Lin, Jian Wang, John See, Shilei Wen, and Erui Ding. Tpm: Multiple object tracking with tracklet-plane matching. *Pattern Recognition*, 107:107480, 2020. 2

[39] Y. Qian, L. Yu, W. Liu, and A. G. Hauptmann. Electricity: An efficient multi-camera vehicle tracking system for intelligent city. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 588–589, 2020. 3

[40] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[41] Pengfei Ren, Kang Lu, Yu Yang, Yun Yang, Guangze Sun, Wei Wang, Gang Wang, Junliang Cao, Zhifeng Zhao, and Wei Liu. Multi-camera vehicle tracking system based on spatial-temporal filtering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4213–4219, 2021. 2

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 2

[43] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Eur. Conf. Comput. Vis.*, pages 17–35, 2016. 6

[44] E. Ristani and C. Tomasi. Features for multi-target multi-camera tracking and re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6036–6046, 2018. 2, 3

[45] Kyujin Shim, Sungjoon Yoon, Kangwook Ko, and Changick Kim. Multi-target multi-camera vehicle tracking for city-scale traffic management. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4193–4200, 2021. 2

[46] A. Specker, A. Schumann, and J. Beyerer. A multitask model for person re-identification and attribute recognition using semantic regions. In *Art. Intell. and Mach. Learn. in Def. Appl.*, 2020. 2

[47] Andreas Specker, Daniel Stadler, Lucas Florin, and Jurgen Beyerer. An occlusion-aware multi-target multi-camera tracking system. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 4173–4182, 2021. 2, 3, 5, 6

[48] D. Stadler and J. Beyerer. Improving multiple pedestrian tracking by track management and occlusion handling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 5

[49] D. Stadler, L. W. Sommer, and J. Beyerer. Pas tracker: Position-, appearance- and size-aware multi-object tracking in drone videos. In *Eur. Conf. Comput. Vis. Worksh.*, pages 604–620, 2020. 1, 2

[50] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020. 3

[51] S. Tang, M. Andriluka, B. Andres, and B. Schiele. Multiple people tracking by lifted multicut and person re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3701–3710, 2017. 1, 2

[52] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8797–8806, 2019. 6

[53] Y. T. Tesfaye, E. Zemene, A. Prati, M. Pelillo, and M. Shah. Multi-target tracking in multiple non-overlapping cameras using constrained dominant sets. *arXiv:1706.06196*, 2017. 3

[54] Duong Nguyen-Ngoc Tran, Long Hoang Pham, Huy-Hung Nguyen, Tai Huu-Phuong Tran, Hyung-Joon Jeon, and Jae Wook Jeon. A region-and-trajectory movement matching for multiple turn-counts at road intersection on edge device. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4087–4094, 2021. 2

[55] Vu-Hoang Tran, Le-Hoai-Hieu Dang, Chinh-Nghiep Nguyen, Ngoc-Hoang-Lam Le, Khanh-Phong Bui, Lam-Truong Dam, Quang-Thang Le, and Dinh-Hiep Huynh. Real-time and robust system for counting movement-specific vehicle at crowded intersections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4228–4235, 2021. 2

[56] Gaoang Wang, Yizhou Wang, Haotian Zhang, Renshu Gu, and Jenq-Neng Hwang. Exploit the connectivity: Multi-object tracking with trackletnet. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 482–490, 2019. 2

[57] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang. Towards real-time multi-object tracking. In *Eur. Conf. Comput. Vis.*, pages 107–122, 2020. 1, 4

[58] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *IEEE Int. Conf. Image Process.*, pages 3645–3649, 2017. 1, 2

[59] Minghu Wu, Yeqiang Qian, Chunxiang Wang, and Ming Yang. A multi-camera vehicle tracking system based on city-scale vehicle re-id and spatial-temporal information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4077–4086, 2021. 2

[60] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. In *Eur. Conf. Comput. Vis.*, pages 472–487, 2018. 1, 2

[61] Kai-Siang Yang, Yu-Kai Chen, Tsai-Shien Chen, Chih-Ting Liu, and Shao-Yi Chien. Tracklet-refined multi-camera tracking based on balanced cross-domain re-identification for vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3983–3992, 2021. 2

[62] Y. Yao, L. Zheng, X. Yang, M. Naphade, and T. Gedeon. Simulating content consistent vehicle datasets with attribute descent. In *Eur. Conf. Comput. Vis.*, pages 775–791, 2020. 6

[63] Jin Ye, Xipeng Yang, Shuai Kang, Yue He, Weiming Zhang, Leping Huang, Minyue Jiang, Wei Zhang, Yifeng Shi, Meng Xia, et al. A robust mtmc tracking system for ai-city challenge 2021. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4044–4053, 2021. 2, 3

[64] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021. 2

[65] Z. Zhang, C. Lan, W. Zeng, X. Jin, and Z. Chen. Relation-aware global attention for person re-identification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3186–3195, 2020. 2

[66] X. Zhou, V. Koltun, and P. Krähenbühl. Tracking objects as points. In *Eur. Conf. Comput. Vis.*, pages 474–490, 2020. 2

[67] Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Heng Fan, Qinghua Hu, and Haibin Ling. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. 2