An Occlusion-aware Multi-target Multi-camera Tracking System

*Andreas Specker[1,2,3] *Daniel Stadler[1,2,3] Lucas Florin[2,3] Jürgen Beyerer[2,1,3]
[1]Karlsruhe Institute of Technology [2]Fraunhofer IOSB [3]Fraunhofer Center for Machine Learning
{andreas.specker,daniel.stadler,lucas.florin,juergen.beyerer}@iosb.fraunhofer.de

Abstract

Multi-camera tracking of vehicles on a city-scale level is a crucial task for efficient traffic monitoring. Most of the errors made by such multi-target multi-camera tracking systems arise due to tracking failures or misleading visual information of detection boxes under occlusion. Therefore, we propose an occlusion-aware approach that leverages temporal information from tracks to improve the single-camera tracking performance by an occlusion handling strategy and additional modules to filter false detections. For the multi-camera tracking, we discard obstacle-occluded detection boxes by a background filtering technique and boxes overlapping with other targets using the available track information to improve the quality of extracted visual features. Furthermore, topological and temporal constraints are incorporated to simplify the re-identification task in the multi-camera clustering. We give detailed insights into our method with ablative experiments and show its competitiveness on the CityFlowV2 dataset, where we achieve promising results ranking 4th in Track 3 of the 2021 AI City Challenge.

1. Introduction

Multi-target multi-camera tracking (MTMCT) is an important component in many applications related to transportation or infrastructure, for example, traffic flow analysis or traffic signal time planning. The MTMCT task demands the localization and identification of multiple targets throughout a set of videos captured by different cameras. MTMCT systems consist of at least two components: A single-camera tracker that localizes all relevant objects in each frame of a video and assigns a unique ID to every instance, and a re-identification (re-ID) model which decides whether two tracks from different cameras belong to the same target. Both sub-tasks are especially challenging in the traffic context because of a large variance in object appearances due to different orientations and distances w.r.t. the camera as well as variations in the camera characteristics at distinct locations. To solve these problems, most of the single-camera tracking (SCT) methods follow the *tracking-by-detection* paradigm [2, 3, 4, 35, 37, 40, 41], where first, a set of detections is generated for each frame of a video independently and afterwards, the detections are linked to tracks based on a similarity measure. That similarity often considers visual features extracted by a re-ID model in addition to position information. Whereas this approach works well in situations where all targets are clearly visible, the representational ability of the extracted features degrades under occlusion since the re-ID model can get confused by overlapping nearby targets [34]. This makes tracking in occlusion situations very challenging.

For this reason, we aim with our SCT approach at focusing more on temporal information derivable from tracks and object relations instead of relying on visual features. Inspired by [34], we apply an occlusion handling strategy, that uses the concept of *occluded* and *occluding* tracks to allow a logic-based re-ID when an occlusion is about to end, for instance, at the end of an overtaking maneuver. Furthermore, the information of track positions is leveraged in order to remove false detections in regions with dense object distributions, where it is unlikely for new objects to appear [35]. One problem we have frequently observed for non-moving vehicles waiting in a turning lane is that this situation is susceptible for identity-switches (IDSWs) with bypassing vehicles. To prevent those tracking errors, we propose additional constraints for the matching of detections to *static* tracks, which we declare based on a velocity threshold.

For the multi-camera tracking (MCT) approach, considering occlusion information is also of high importance, as visual features extracted from occluded detection boxes are not expressive. To cope with occlusions from static obstacles, for example, traffic signs, we propose generating a foreground mask with a background subtraction model to filter both tracks that appear completely in the background and discard features of track boxes with severe background overlaps. As a result, our approach solely relies on features of frames in which the vehicle is clearly visible. With the same motivation, we use the overlap information available from tracks to discard track boxes, that have overlaps with

*These authors contributed equally.

boxes of other targets, for the feature extraction. To further support our re-ID model in the multi-camera clustering, we incorporate a scene model that uses the topology of the traffic network and temporal constraints to inhibit impossible matches [17].

In summary, the main contributions of our work are as follows:

- A sophisticated MTMCT system that focuses on an improved performance under occlusion and leverages topological and temporal constraints is proposed.
- Besides other SCT extensions to cope with occlusions and false detections adapted from related work, we introduce a module which prevents tracking errors by verifying detection assignments to static tracks.
- We propose to use a background subtraction method and the overlap information from tracks to remove boxes with misleading visual information from the feature extraction in order to enhance re-ID performance.

2. Related Work

2.1. Single-camera Tracking

The predominant number of SCT approaches divide the task into a detection step followed by an association step, in which detections of the same targets are linked on the basis of a similarity measure [2, 3, 4, 35, 37, 40, 41]. While most methods incorporate at least position and motion information [2, 3], some also utilize visual information extracted by convolutional neural networks (CNNs) designed for the task of re-ID [4, 35, 37, 40] or other cues like pose information are used [37, 41]. While this procedure allows a high flexibility and the approaches still achieve state-of-the-art performance, the temporal context available in videos is mostly ignored. In contrast, some recent methods integrate detection and tracking more tightly extending the object detector to a tracker [1, 46], applying 3D CNNs to directly detect tracklets [27], or using tracking results as prior knowledge for the detection model [12, 46]. We also follow this trend leveraging temporal information from tracks in our occlusion handling strategy, an approach to filter false detections, and the verification of assigned detections to static tracks.

2.2. Vehicle Re-identification

Although there has been research in the field of vehicle re-ID in recent years, there is a more extensive literature on the related problem of person re-ID. For person re-ID, there have been many complex approaches that aim to exploit the particular structure of the domain, such as aggregating local features [36, 45], exploiting attention mechanisms [8, 44], or using auxiliary high-level semantic attributes [23, 33]. Similar complex concepts are exploited for the task of vehicle re-ID as well [9, 11, 18]. However, several works [24, 43] show that state-of-the-art performance can be achieved simply by learning global features using a bag of tricks. Since global feature learning does not rely on the specific structure of the person re-ID problem (such as body parts), it can be easily adapted for vehicle re-ID. This was done in [15], which achieved the third place in the re-ID Track of the last edition of the AI City Challenge [25]. Due to the promising results of this work, we also rely on a global feature learning approach for our vehicle re-ID component.

2.3. Multi-camera Tracking

In general, related literature indicates that clustering approaches are well suited to tackle the task of MTMCT [17, 20, 31, 39]. Moreover, recent works [16, 17, 20, 28] show that the use of external information about the camera setup is beneficial. For example, both the top two teams of the last edition of the AI City Challenge exploited the scene topology to prevent infeasible cross-camera transitions. While in [28], only camera adjacency is used, He *et al*. [16] also leverage the movement directions to determine which camera transitions are plausible. In [17], camera-specific zones are defined to decide which tracks can appear in multiple cameras. Based on this, we decide to develop a clustering approach for the multi-camera component of our tracking system that utilizes a scene model to improve the matching of single-camera tracks.

3. Proposed MTMCT System

3.1. Overview

Before we describe the components of our MTMCT system in more detail, we give an overview in Fig. 1. At each time step t, the generated detections $\mathcal{D}_t$ are associated with the propagated tracks $\tilde{\mathcal{T}}_t$, after applying a motion model to the tracks from the previous time step $\mathcal{T}_{t-1}$. Instead of directly initializing new tracks from remaining detections $\mathcal{D}_t^r$, we use our occlusion handling strategy to re-identify occluded tracks $\tilde{\mathcal{T}}_t^r$ when they get visible again. After that, we incorporate a module that removes false detections in regions with dense object distributions using past track information. Furthermore, a method to identify and remove false detections by checking the match of detections to non-moving vehicles, which we find to be susceptible for IDSWs when being overtaken by other vehicles, is proposed. The resulting tracks $\mathcal{T}_t$ are made up of active tracks $\mathcal{T}_t^a$, propagated inactive tracks $\tilde{\mathcal{T}}_t^i$, and new tracks $\mathcal{T}_t^n$.

After the generation of tracks in each camera of a multi-camera setting, the following pipeline is applied. As a preprocessing step, a scene model is built which includes foreground masks for all cameras of a sequence generated by a background subtraction technique. After removing short

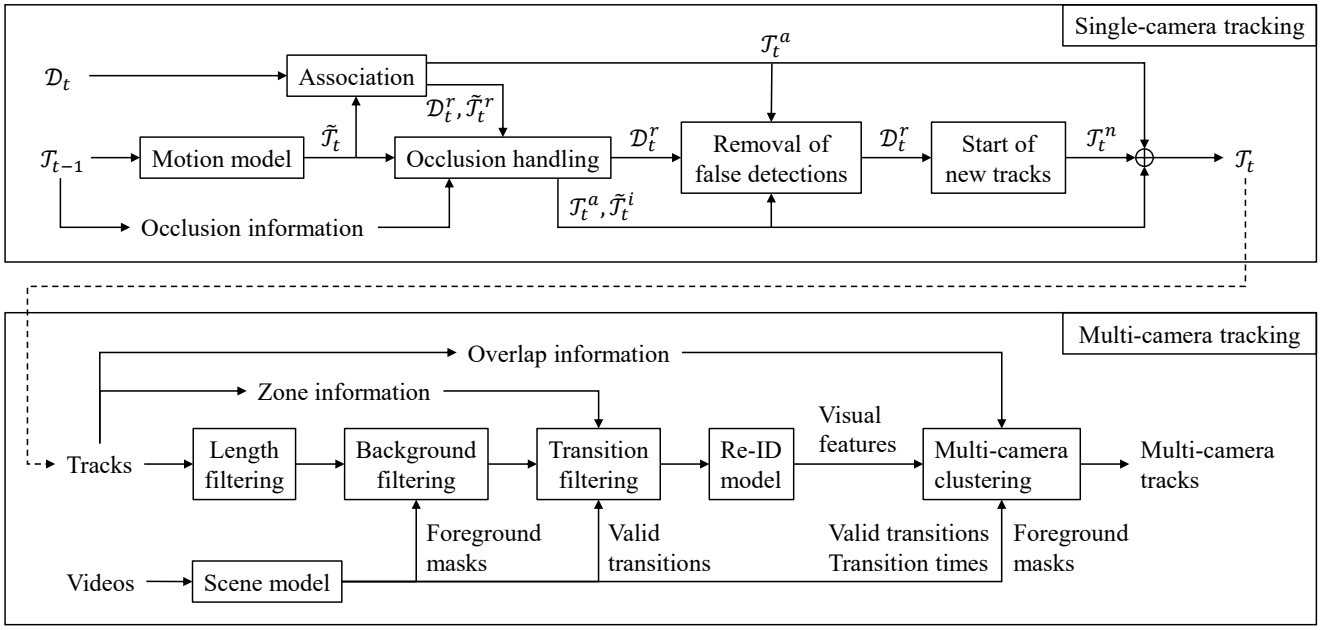

Figure 1: Scheme of the proposed MTMCT system. Besides a sophisticated track management, the special features of our SCT include an occlusion handling strategy and a module to remove false detections. Our MCT also includes occlusion-aware modules to cope with overlapping tracks or detection boxes that are occluded by static obstacles like traffic signs.

tracks that are mostly clutter tracks, the foreground masks are used to further remove irrelevant tracks from parking vehicles or false background detections like traffic signs. Besides the foreground masks, the scene model contains manually defined lists of valid transitions between cameras using topological information of the camera network and, in addition, both lower and upper bounds for the possible transition times of vehicles driving from one camera to another one. The valid transitions allow to filter tracks which can not appear in multiple cameras within a limited period of time. For the remaining tracks, visual features, that are the basis for the subsequent cross-camera clustering, are extracted by a re-ID model. In addition to the constraints coming from the scene model, the overlap information of tracks with nearby targets is considered in our multi-camera clustering approach, that finally yields the multi-camera tracks.

3.2. Single-camera Tracking

For our single-camera vehicle tracker, we follow the *tracking-by-detection* paradigm. The separate treatment of detection and subsequent association of detections to tracks allows the usage of an arbitrary object detection model that can be chosen dependent on the specific task. Furthermore, regarding detection and association separately provides a high flexibility in the design of the track management which is very important for tracking in scenarios with severe object-object occlusions as present in videos of traffic cameras. Especially at intersections, where multiple vehicles can overlap with each other because of the oblique view, we identify and treat two major challenges in SCT – unreliable visual information in occluded detection boxes and an increased risk for false detections under occlusion.

Basic algorithm. As occluded detection boxes include parts of nearby vehicles, an extraction of expressive visual features is very difficult. Hence, we do not use visual features in the assignment of detections to tracks but use only position, size, and motion information. For that, we utilize the Intersection over Union (IoU) as similarity measure between propagated track boxes and detection boxes, whereby a constant velocity assumption is made to predict the position of track boxes. In detail, the velocity vector of a track is computed by averaging the displacements of bounding box centers over the last n time steps. The matching problem is solved with the Hungarian method [19] and a minimum overlap criterion o_{min} is applied. To allow a motion-based re-ID after occlusion, we do not terminate tracks immediately when no detection can be assigned but keep tracks for a maximum of i time steps as *inactive*. For an increased robustness against false detections, our track management includes a concept of track initialization. A so-called *candidate* track only turns *active* if a detection can be assigned to it in m consecutive time steps, otherwise the track is deleted. Note that active tracks are preferred both over inactive tracks and candidate tracks in the association step.

Occlusion handling. We develop several approaches that aim at improving the association accuracy under occlusion,

as most of the tracking errors occur in such situations. Our first extension is an occlusion handling strategy that is inspired by the concept of *occluded* and *occluding* tracks introduced in [34]. Whenever an active track turns inactive, it is checked whether there exists another active track with high overlap, *i.e.*, larger than the non-maximum suppression (NMS) threshold o_{NMS}, and if this is the case, the two tracks are considered as *occlusion pair*. When later, an unassigned detection overlaps with both of the (propagated) track boxes of an occlusion pair, the detection is matched with the occluded track. Note that the difference of the proposed occlusion handling to the motion-based re-ID of the basic algorithm lies in the omission of the minimum overlap criterion o_{min} for occlusion track pairs. With this strategy, it is possible to leverage the interaction information of occluding and occluded vehicles to successfully re-identify vehicles even if the similarity constraint is not fulfilled, for instance, because of an inaccurate motion estimation.

Filtering of false detections. As a second extension in our SCT pipeline, we adopt a filtering mechanism from [35] to suppress false detections in regions with dense object distributions. For unassigned detections, the intersection of the detection box with the union of all track boxes is calculated. If this value exceeds the threshold o_{max}, the detection is deleted, arguing that it is unlikely for new objects to occur at positions where other objects are already present.

Check of assignments to static tracks. Besides false detections in crowded regions like intersections, we observe another type of detection error that often occurs when a vehicle bypasses a non-moving vehicle waiting in a turning lane. Instead of providing two correct detections, the object detector sometimes generates one detection that covers both vehicles which in turn can cause IDSWs. To avoid this, we propose a new module that checks the assignments of detections to *static* tracks. We term a track static if its normalized velocity w.r.t. the track size falls below a threshold v_{static} for m consecutive time steps. While a track is deemed static, we add two further constraints for assigning a detection to it. The relation of the aspect ratio of the detection box w.r.t. the track box must not exceed a_{max} and the ratio of box sizes must not exceed s_{max}, arguing that the dimensions of non-moving vehicles can not change on images of static cameras. With these additional constraints, we are able to identify and remove false detections that would erroneously be assigned to static tracks and potentially cause IDSWs.

Tracking backwards. As last extension, we take over the *tracking in both temporal directions* from [34]. By running our tracker on the videos one time in the forward direction, one time in the backward direction, and applying a post-processing merging mechanism of the two generated sets of tracks, the overall performance can be further improved.

3.3. Vehicle Re-identification

For the task of vehicle re-ID, we rely on a global feature learning model similar to [15]. In detail, we employ the ResNet-101 architecture [14] with IBN-A bottlenecks [26] followed by a batch normalization layer, and a fully connected identity classification layer. Furthermore, we remove the last downsampling operation by setting the stride parameter of the last convolutional layer to 1 [36]. The spatial resolution of the resulting feature map increases and, thus, the features represent more fine-grained information, which is especially important if vehicles are small. Analogous to many other works, our networks are trained using a combination of the triplet and the identity loss function.

To further enhance the accuracy, we use an ensemble of multiple re-ID networks. The use of an ensemble is only beneficial if the models provide complementary information. To achieve this, we use varying data augmentation methods or make small adjustments to the network architecture. In detail, we train one network with random rotation data augmentation (RR) and one with generalized mean (GeM) pooling [43] in addition to the baseline approach. RR pre-processes images by rotating them with a probability of 30%. The rotation angle is randomly chosen between 0 and 30 degrees, either clockwise or counter-clockwise. GeM pooling for re-ID tasks was proposed because standard global pooling methods such as maximum or average pooling are not well suited to capture discriminative features. On the one hand, average pooling is beneficial because it tends to focus on the whole visual appearance of a vehicle but on the other hand, fine-grained local information is necessary for distinguishing between cars of the same make and model which look almost the same. GeM pooling handles this problem by a learnable parameter which balances the trade-off between both pooling operations. We therefore replace the global average pooling layer by the GeM pooling operation in our experiments.

Similar to [15], we adopt a two-stage training strategy. In the first stage, we pre-train the models using a mix of real-world and simulated data from the VehicleX dataset [42]. Subsequently, we fine-tune the models solely based on real-world data. During the fine-tuning step, we use images from the test set with pseudo labels in addition to the training data. We generate these pseudo labels by using our pre-trained models for feature extraction and conducting our MCT clustering approach (see Section 3.4) for the first 250 iterations. Afterwards, we assign a separate label to each of the clustered multi-camera tracks and use the detections of every 10th frame as training images. Since our clustering approach focuses on the most visually similar tracklets, the assumption can be made that merges of single-camera tracklets are correct during early iterations. As a result, meaningful pseudo labels are obtained.

3.4. Multi-camera Clustering

In this section, we describe our MCT pipeline, which incorporates the re-ID component described in Section 3.3. In general, our approach consists of a filtering step, followed by a hierarchical clustering which is based on the vehicle re-ID, and constraints derived from a scene model. In the following, these components are thoroughly introduced.

Scene model. Our scene model is inspired by [17], where zones are defined for every camera in order to assign each track a trajectory that is used as prior knowledge, in which camera the vehicle can appear next. Furthermore, temporal constraints are applied to inhibit impossible assignments. For a detailed description of the adopted *camera link models*, refer to [17]. In addition, we propose to incorporate foreground masks, that are used to filter background detections (see next section), into our scene model. Another difference to [17] is that we allow assigning tracks which have only one valid start or end zone so that also track fragments which lie in a valid start or end zone can be merged.

Filtering. Before clustering, we filter irrelevant tracks, *i.e.*, emerged from parking vehicles or false detections, to reduce the risk for incorrectly merging tracks across multiple cameras while at the same time lowering the computational effort required for the clustering step. A total of three filter methods are applied – background filtering, removal of short tracks, and filtering of tracks that can not appear in another camera. Background filtering aims at removing false positive tracks. This is particularly important as we follow [32], which has shown that it is beneficial to focus on a high recall in the detection phase to avoid the risk for missing meaningful detections and subsequently filter the detections. But in contrast to this work, we propose to filter complete tracks instead of single detections. Examples of such false positives are parked vehicles or traffic signs misclassified as vehicles by the detection model. Both types of errors have in common that their position and pixel values do not change much during the video. Thus, we propose to filter the aforementioned false positives by leveraging the MOG2 foreground-background segmentation method [47, 48]. Foreground masks are computed separately for each camera view by feeding all frames to the background subtraction model, assigning the maximum foreground value to each pixel, and subsequently performing morphological opening to reduce noise. Finally, the tracks whose bounding boxes overlap with the foreground by less than 90% are removed. Moreover, we do not consider single-camera tracks with less than 10 detections. Such tracks mainly consist of false detections or only represent a small track fragment emerged, for example, from an IDSW. At last, we filter tracks that can not appear in another camera due to topological or time constraints with our scene model because only tracks occurring in multiple cameras are relevant for the MTMCT task. Vehicles that do not appear in any of the multi-camera zones are not driving in the direction of another camera and are therefore omitted.

Re-ID distance calculation. To determine the visual similarity between two tracks, we compute the Euclidean distance between the L2-normalized mean features of the track. However, this procedure is prone to errors since overtaking vehicles, low resolution detections in the background, and boxes that mainly show background clutter add noise to the features. Therefore, we propose to exclude detections that overlap with the background (using foreground masks) and detections that overlap with other vehicles (using track information) by more than 20% from the mean feature computation. This makes the re-ID features more expressive so that many wrong merges are avoided.

Hierarchical clustering. As the core component for merging single-camera tracks into multi-camera tracks, we employ an iterative hierarchical clustering method since works from literature [17, 20] show promising results using such an approach. We enhance these works by introducing additional constraints, *e.g.*, the best match constraint (see details below). The visually most similar tracks according to the distance between the vehicles' re-ID features are merged iteratively until the minimal distance exceeds a threshold t_{reid}. To avoid false multi-camera merges, we formulate the following constraints that must be fulfilled for two tracks to be combined – the *no time overlap* constraint, the *valid transition* constraint, the *transition time* constraint, and the *best match* constraint. The first constraint leverages that single-camera tracks which belong to the same vehicle can not occur in different cameras at the same time. Therefore, solely tracks that do not overlap in time are candidates for merging. The valid transition constraint results from the zone information of the scene model. A vehicle leaving a camera view in a specific direction can only appear in a small subset of zones in the adjacent camera. As a result, we only cluster tracks if there is such a valid transition between the two merge candidates. The transition time constraint leverages that the travel between two cameras takes at least a minimum and at most a maximum amount of time (based on our scene model) since speeding and stopping are prohibited. The best match constraint assumes that there must be at least one good visual similar match between detections of two tracklets belonging to the same vehicle. We assure this by computing the minimum distance between all re-ID embedding pairs of merge candidates and applying a threshold t_{bm}. In cases where the majority of track boxes is very small, these skew the average features. This is solved by looking at the best match distance, which is usually found between higher resolution detections. The best match distance is applied as a constraint and is not used to select cluster candidates because it is vulnerable to tracks with IDSWs.

4. Experiments

4.1. Datasets

For Track 3 of the AI City Challenge 2021, the usage of two datasets is allowed – the real-world CityFlowV2 dataset [38] and the synthetic VehicleX dataset [42].

CityFlowV2. The CityFlowV2 dataset is the new version of the CityFlow dataset, a benchmark for city-scale MTMCT. It consists of a large number of high-resolution camera feeds from different intersections in a U.S. city and covers a variety of scenarios such as city streets, residential areas, and highways. In the training and validation scenes, there are multiple cameras per intersection leading to overlapping field of views. In the test cameras, however, there is only one camera per intersection. Therefore, the validation set is useful to evaluate the re-ID models but it is no reliable performance predictor for our MTMCT system on the test scene [25]. In the challenge, the performance is evaluated using the IDF1 score [30], which measures within-camera tracking accuracy and cross-camera ID consistency.

VehicleX. The VehicleX dataset is a synthetic dataset for vehicle re-ID. Each vehicle of the dataset consists of a 3D model for which rendering parameters such as viewpoint and lighting can be varied. These models are rendered using the Unreal Engine and pasted on background images taken from the CityFlow dataset. Afterwards, the synthetic images are adapted to the domain of the CityFlow dataset.

4.2. Vehicle Detection and Single-camera Tracking

Detection model. The performance of the overall MTMCT system heavily depends on the quality of the detections. Since no online requirement has to be fulfilled, we build an ensemble of four state-of-the-art object detection models, in particular Cascade Mask R-CNN [5], HTC [6], GFL [21], and DetectoRS [29]. Training a detector on the CityFlowV2 dataset is not straightforward because only multi-camera vehicles are annotated, *i.e.*, no annotation boxes are available for vehicles that only appear in a single camera. Therefore, we use models trained on the large COCO dataset [22] from the MMDetection toolbox [7] to generate the detections for our single-camera tracker. The detection boxes from the four models are combined with an adapted NMS, where we base the suppression not on the detection score but keep the boxes with higher y2-value, as these boxes usually correspond to the visible objects due to the oblique view of the cameras. The overlap threshold o_{NMS} is empirically set to 0.5 and only detection boxes with a minimum score of 0.3 are considered in the SCT.

Single-camera tracking. To evaluate our SCT approach and the proposed additional modules, we run several experiments with different configurations on the CityFlowV2 dataset. A meaningful quantitative evaluation of both the detection and the SCT performance can be hardly obtained because of the missing annotations for vehicles which do not appear in multiple cameras. Therefore, we first tune our algorithms qualitatively on the validation set to determine the best settings, which are listed in Tab. 1. Afterwards, we apply our tracker both in the basic version and with the proposed extensions on the test set to prove the generalization ability of our framework.

Table 1: Parameters of our single-camera tracker.

n	o_{min}	i	m	o_{max}	v_{static}	a_{max}	s_{max}
10	0.3	20	3	0.5	0.02	2	1.5

A qualitative example of our occlusion handling strategy is depicted in Fig. 2. Two accelerating vehicles are shown, whereby one overtakes the other and occludes it. When the occluded vehicle is visible again, the occlusion handling enables the re-ID of the occluded vehicle. Without the occlusion handling strategy, the occluded track can not be continued because the motion estimation is not accurate enough.

In Fig. 3, a typical scene where multiple vehicles are waiting at an intersection is shown. In these dense areas, the risk for false detections is high. With the filtering module that utilizes the temporal information available in tracks, some wrong detections can be identified and removed so that they do not introduce tracking errors.

The benefits of the proposed module to check the assignments of detections to static tracks can be seen in Fig. 4. If a vehicle bypasses a non-moving vehicle and the distance to the camera is high, the detector often can not distinguish the two overlapping objects. This can lead to an IDSW as assigning this detection is ambiguous. In the proposed approach, we take advantage of the fact that the dimensions of a non-moving vehicle on images of static traffic cameras can not change and prevent the assignment of false detections to static tracks which in turn avoids tracking errors.

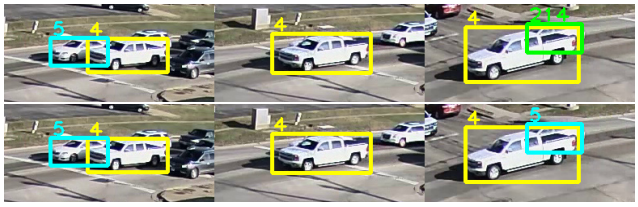

Figure 2: Successful re-ID of an occluded track with our occlusion handling (bottom) where the basic SCT fails (top).

Figure 3: With the approach to filter false detections (right), the start of a wrong track with ID 512 (left) is prevented.

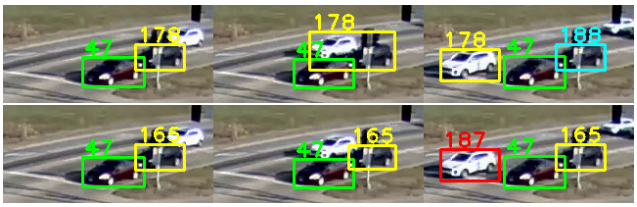

Figure 4: Prevention of an IDSW (178/188) by our method (bottom) in comparison with the basic SCT algorithm (top).

4.3. Vehicle Re-identification

Dataset. For the training of our re-ID models, we use the re-ID split of the CityFlowV2 dataset in conjunction with the provided VehicleX engine images. Analogous to [15], we use all real-world images plus the images of the first 100 vehicle IDs of the synthetic dataset which adds 14,536 images to our training set. For validation and comparison of different approaches, we construct a validation set from the four cameras included in scene 2 of the MTMCT split. It consists of every 10th track detection and includes a total of 2,368 images from 145 different vehicle IDs. The first frames of vehicle tracks in a camera view are selected as query leading to 450 query and 1,918 gallery images. To avoid overfitting, images of vehicles occurring in our validation set are excluded from training during our experiments. The final models were trained using all IDs from training and validation splits. We fine-tune our models with training and validation data as well as test data annotated with pseudo labels. In total, we have assigned 158 different vehicle IDs to 3,589 cropped vehicle bounding boxes.

Training parameters. We initialize our models with weights pre-trained on ImageNet [10]. First, we train our models with real-world and simulated data for 70 epochs with a learning rate of 0.01 and a warmup schedule, which increases the learning rate linearly during the first 10 epochs starting at 0.0001. The learning rate is multiplied by a factor of 0.1 in epoch 40. Fine-tuning is conducted with a reduced base learning rate of 0.001 for 10 epochs.

Table 2: Re-ID results on our validation set.

Method	mAP	R-1
Baseline ResNet-50	48.7	54.8
Baseline ResNet-50 IBN-A	48.5	56.2
Baseline ResNet-101 IBN-A	52.8	57.5
Baseline + GeM	52.9	57.3
Baseline + RR	53.5	58.4
Fine-tuned w/ Test	51.9	56.6
Fine-tuned w/ Test + GeM	49.9	54.8
Fine-tuned w/ Test + RR	52.5	58.2
Ensemble	**58.0**	**63.1**

Figure 5: Qualitative results of our re-ID component. Mainly good results are achieved. Typical error cases are vehicles of the same make that share a similar appearance.

Results. We provide quantitative results for vehicle re-ID on our custom validation set in Tab. 2. The mean Average Precision (mAP) and the rank-1 accuracy (R-1) serve as evaluation metrics. Note that ResNet-101 is applied as backbone unless otherwise stated. One can observe that the ResNet models with IBN-A bottlenecks outperform the vanilla version in terms of R-1, which is the more important measure in our case. That is because our clustering method considers the minimum distance between two tracks in each iteration, which corresponds to the R-1 result. Furthermore, the results show that the use of the deeper ResNet-101 model is beneficial. In numbers, the ResNet-101 IBN-A achieves a 4.3 points higher mAP and a 1.3 points increase in R-1 compared to the ResNet-50 IBN-A network. Replacing the global average pooling layer by GeM pooling scores similar as the baseline. In contrast, using RR as an additional data augmentation improves vehicle re-ID performance from 52.8% to 53.5% mAP and from 57.5% to 58.4% in R-1. Fine-tuning the network with test data does not lead to an improvement on our validation set but greatly improves the re-ID accuracy on the test set from the challenge. The reasons for that are different characteristics of the validation scene 2 and the unlabeled test data provided for the challenge. Analogous to the pre-training results, the model trained with RR achieves the highest scores, followed by the baseline, and the GeM pooling approach. Our ensemble further improves both metrics resulting in the best performance of 58.0% in mAP and 63.1% in R-1.

Besides, we provide qualitative results of our vehicle re-ID component in Fig. 5. The results indicate that our ensemble performs well even if query and gallery vehicles are displayed from different views. Errors such as in the last row arise from similar visual appearance. Unlike persons, vehicles of the same make or model share a very similar visual appearance and can only be distinguished based on a few cues, such as dirt or small differences in car equipment.

4.4. Multi-camera Clustering

In this section, we discuss the results of our cross-camera clustering approach. Since test set annotations are not publicly available, we evaluate the components on scene 2 of the dataset's validation split. Moreover, it is impossible to evaluate all components and constraints on this valida-

Table 3: Influence of different components of our MCT approach evaluated on scene 2 of the validation set.

Approach	IDF1	IDP	IDR
Dataset Baseline	23.24	13.95	69.45
Our SCT	34.41	22.07	**78.14**
+ BG filtering	43.98	30.60	78.10
+ Exclude BG boxes	45.82	32.67	76.68
+ Exclude overlapping boxes	**48.17**	**34.82**	78.12

tion set because it includes, *e.g.*, overlapping camera views. Therefore, we only apply the best match constraint and use a single re-ID model in the following. We determine thresholds empirically and set t_{reid} to 0.5 and t_{bm} to 0.4.

Tab. 3 highlights the influences of several components on the resulting MTMCT performance. Note that parameters and thus the results might not be optimal since some components are omitted. Nevertheless, the results clearly show that our SCT approach outperforms the baseline provided with the dataset (Mask R-CNN [13] + DeepSORT [40]) by a large margin. The additional filtering of background (BG) tracks leads to a further strong increase in IDF1. With respect to the IDR metric, the performance stays the same. This indicates that indeed incorrect background tracks are filtered. Excluding background boxes and overlapping boxes from vehicle re-ID further improves the tracking performance. In total, the baseline approach is outperformed through the proposed optimizations by 25 points in IDF1.

The final challenge results are given in Tab. 4. Our MTMCT system achieves the fourth place with an IDF1 score of 69.10%.

Finally, we present two qualitative MCT results on the test set in Fig. 6. Each row shows evenly sampled detections from a single-camera track. In both examples, the cars were tracked correctly across four different camera views. A look at the second row of Fig. 6a shows that our re-ID component is able to combine single-camera tracks from

Table 4: Challenge results on the official test set.

Rank	Team ID	IDF1	Rank	Team ID	IDF1
1	75	80.95	11	3	29.74
2	29	77.87	12	45	29.08
3	7	76.51	13	110	25.68
4	**Ours**	**69.10**	14	60	25.26
5	42	62.38	15	82	22.85
6	27	57.63	16	67	20.38
7	15	56.54	17	11	19.24
8	48	55.34	18	123	13.43
9	79	54.58	19	61	11.57
10	112	54.52	20	129	5.58

(a)

(b)

Figure 6: Final MTMCT results on the official test set of the AI City Challenge 2021. Each row represents one camera view. Vehicles are correctly tracked although lighting conditions vary strongly and overlapping situations occur.

different cameras, even if the lighting conditions are very different. This can be explained by the fine-tuning stage that allows the model to learn dataset-specific characteristics. Moreover, the second example in Fig. 6b proves the benefit of excluding overlapping boxes from vehicle re-ID. The single-camera track in the last row exhibits some heavy occlusions induced by an overtaking car. Nevertheless, the track follows the target vehicle and does not switch to the occluding car. In addition, the single-camera track is assigned to the correct multi-camera track since the boxes that overlap with other vehicles are not used for re-ID, but instead features from images like the first one in the last row.

5. Conclusion

In this paper, we propose an occlusion-aware MTMCT system that ranks 4th in Track 3 of the 2021 AI City Challenge. For SCT, we develop several modules utilizing temporal information derivable from tracks with the focus on handling occlusions, suppressing false detections, and verifying assignments to non-moving vehicles. Regarding MCT, we propose a hierarchical cross-camera clustering based on vehicle re-ID features, which leverages a scene model, topological and temporal constraints, and a background filtering component. To decrease the negative influence of overlapping vehicles, we improve the re-ID by excluding boxes in the background or with occlusion.

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] 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.

[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.

[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.

[5] Z. Cai and N. Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(5):1483–1498, 2021.

[6] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. Hybrid task cascade for instance segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4969–4978, 2019.

[7] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. MMDetection open mmlab detection toolbox and benchmark. *arXiv:1906.07155*, 2019.

[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.

[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.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255, 2009.

[11] 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.

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

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2961–2969, 2017.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] H. W. Kuhn and Bryn Yaw. The hungarian method for the assignment problem. *Naval Res. Logist. Quart.*, pages 83–97, 1955.

[20] 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.

[21] X. Li, W. Wang, L. Wu, S. Chen, X. Hu, J. Li, J. Tang, and J. Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *Adv. Neural Inform. Process. Syst.*, pages 21002–21012, 2020.

[22] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755, 2014.

[23] 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.

[24] 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.

[25] 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.

[26] X. Pan, P. Luo, J. Shi, and X. Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Eur. Conf. Comput. Vis.*, pages 464–479, 2018.

[27] 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.

[28] 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.

[29] S. Qiao, L.-C. Chen, and A. Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv:2006.02334*, 2020.

[30] 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.

[31] 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.

[32] A. Schumann, A. Specker, and J. Beyerer. Attribute-based person retrieval and search in video sequences. In *IEEE Int. Conf. Adv. Video Sign. Surv.*, 2018.

[33] 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.

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

[35] 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.

[36] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *Eur. Conf. Comput. Vis.*, pages 480–496, 2018.

[37] 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.

[38] 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.

[39] 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.

[40] 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.

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

[42] 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.

[43] 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.

[44] 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.

[45] L. Zhao, X. Li, Y. Zhuang, and J. Wang. Deeply-learned part-aligned representations for person re-identification. In *Int. Conf. Comput. Vis.*, pages 3219–3228, 2017.

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

[47] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Int. Conf. Pattern Recog.*, pages 28–31, 2004.

[48] Z. Zivkovic and F. Van Der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognit. Lett.*, 27(7):773–780, 2006.