The MTA Dataset for Multi Target Multi Camera Pedestrian Tracking by Weighted Distance Aggregation

Philipp Köhl[1,2] Andreas Specker[3,1] Arne Schumann[1,2] Jürgen Beyerer[1,3]
[1]Fraunhofer IOSB, Karlsruhe, Germany [2]Fraunhofer Center for Machine Learning
[3]Vision and Fusion Lab, Institute for Anthropomatics and Robotics,
Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
{philipp.koehl, andreas.specker, arne.schumann, juergen.beyerer}@iosb.fraunhofer.de

Abstract

Existing multi target multi camera tracking (MTMCT) datasets are small in terms of the number of identities and video length. The creation of new real world datasets is hard as privacy has to be guaranteed and the labeling is tedious. Therefore in the scope of this work a mod for GTA V to record a MTMCT dataset has been developed and used to record a simulated MTMCT dataset called Multi Camera Track Auto (MTA). The MTA dataset contains over 2,800 person identities, 6 cameras and a video length of over 100 minutes per camera. Additionally a MTMCT system has been implemented to provide a baseline for the created dataset. The system's pipeline consists of stages for person detection, person re-identification, single camera multi target tracking, track distance calculation, and track association. The track distance calculation comprises a weighted aggregation of the following distances: a single camera time constraint, a multi camera time constraint using overlapping camera areas, an appearance feature distance, a homography matching with pairwise camera homographies, and a linear prediction based on the velocity and the time difference of tracks. When using all partial distances, we were able to surpass the results of state-of-the-art single camera trackers by +13% IDF1 score. The MTA dataset, code, and baselines are available at `github.com/schuar-iosb/mta-dataset`.

1. Introduction

Multi target multi camera (MTMC) tracking, i.e. tracking many persons across a network of possibly non-overlapping cameras, is an important element of modern security, sports analysis, or retail systems. The development of MTMC tracking approaches is a complex matter, as they involve a number of tasks, which are themselves challenging computer vision problems, namely: person detection, single camera multi target tracking, and person re-identification. All these components are impacted by challenges, such as variation in resolution or camera distance, variation in view angle, non-overlapping camera views, occlusion in crowded areas, or illumination changes. However, a more fundamental challenge in the development of MTMC tracking methods is the lack of suitable datasets. In order to allow for MTMC development and assessment, a dataset needs to provide imagery with these challenging characteristics, as well as comprehensive ground truth, particularly consistent IDs across all cameras. Such data is not only difficult to annotate but its collection comes with the risk of violating current or future data protection rights.

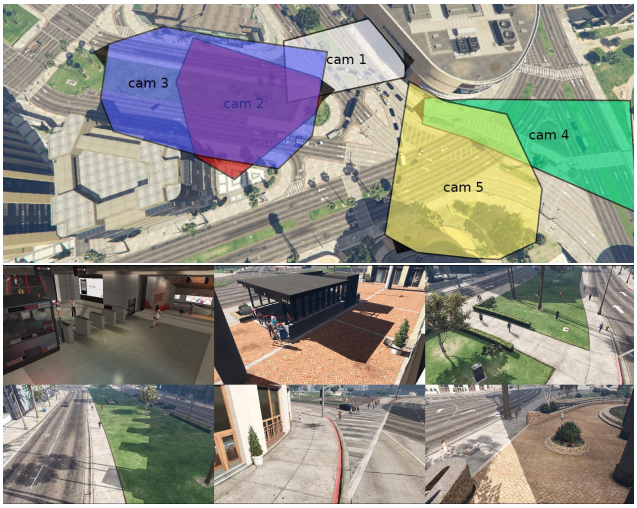

Figure 1. Camera view arrangement and sample images of the MTA dataset.

In this work we address the lack of suitable MTMC datasets by creating a new large scale simulated dataset, which depicts an urban scene recorded by six cameras as shown in Figure 1. The dataset is recorded in a small section of the Grand Theft Auto 5 (GTA) virtual world, which offers a high degree of realism and detail. We have taken care to include overlapping and non-overlapping cameras,

night- and daytime, indoor and outdoor regions, and varying degrees of crowdedness. Our dataset consists of six camera views recording over 2,800 different persons for a combined video length of 10 hours. This makes it by far the largest existing MTMC dataset available and allows for a comprehensive evaluation of MTMC tracking approaches under a variety of conditions. Based on this dataset we conduct baseline evaluations of several established state-of-the-art methods in person detection, single camera tracking and re-identification. We combine the best suited models into our own modular MTMC tracking system, which relies on calculation and aggregation of a number of relevant distance measures to associate tracks across the camera network. Our work makes the following contributions: i) We create the currently largest MTMC dataset for evaluation of person detection, re-identification, pose estimation, single- and multi-camera tracking. ii) The dataset, as well as the code for its creation, will be made available to the community. iii) We propose our own modular MTMC tracking system based on aggregation of several track distance measures as a baseline result on this dataset.

2. Related Work

The development and evaluation of MTMC tracking requires data with very specific properties. Datasets must include temporally synchronized videos from a number of cameras, some of which should be non-overlapping. Particularly the requirement that persons must be annotated with consistent identities across all camera views results in high annotation cost. Most existing tracking datasets focus on the multi target aspect but do not provide synchronized multi camera data [27, 14]. However, some datasets exist that are suitable for MTMC tracking, see Table 1. Most of these datasets have the drawback of either being very short in duration [18, 15, 13, 17], having a very low video resolution [1, 7] or not providing consistent person IDs [8]. Closest to our dataset in idea is the MOCAT dataset [2], which comprises three scenes recorded by up to 10 cameras and 100 objects. This dataset was created using Garry's Mod, which is a physics-based sandbox game [32]. However, the small size of the dataset makes it unsuitable to train large deep learning models and the degree of visual variation is limited. Closest in scope to our dataset is the popular real world DukeMTMC dataset [30]. Unfortunately, this dataset was recently withdrawn due to privacy and consent issues with the depicted pedestrians. Our dataset aims to provide a larger basis for training and evaluation than DukeMTMC while alleviating privacy concerns by relying on high quality simulated data.

Systems for multi target multi camera tracking often consist of components which solve several subproblems in order to combine them as a solution for entire multi camera tracking problem. Such subproblems are person detection, appearance description, single camera tracking, feature calculation for data association and data association [31]. We will limit our discussion of related methodology to the main task, i.e., multi camera multi target tracking by cross-camera data association.

Features for data association In order to perform data association, it is often necessary to calculate features, distances or restrictions between detections, tracklets or tracks of different cameras. Zhang et al. [38] use a time restriction which leverages the fact that persons can only be visible at the same time in cameras with overlap. Depending on the available information about the cameras of the used dataset, some approaches like the one from Bredereck et al. [5] transform track positions from multiple cameras with the help of camera parameters into a 3D coordinate system to support the data association. Another popular feature that is being used by Ristani et al. [31] is linear motion information of people. In this work we rely on a combination of these three methods, i.e., time restriction, position projection through homographies learned from data, and linear motion projection.

Data association procedure The task of forming groups of detections, tracklets (short tracks) or tracks to constitute identities is often called data association in this context. Some approaches like, e.g., [7] only look at time consecutive data points, whereas others consider all pairwise datapoints [12]. It is more computationally costly to look at pairwise datapoints, but more accurate [31]. The actual association procedures differ as well. Cao et al. [7] create a graph and solve a min-cost flow network problem. The approach from Zhang et al. [38] uses hierarchical clustering to cluster tracklets in order to solve the data association problem. Ristani et al. [31] utilizes correlation clustering [4] of detections to tackle the data association problem. A benefit of correlation clustering is that it is not necessary to provide a number of clusters in advance. Tang et al. [33] solves the minimum cost lifted multicut problem to perform multi person tracking in a single camera. Similar to [38], we apply hierarchical clustering to group single camera tracks as well.

3. The MTA Dataset

Our proposed dataset relies on high quality imagery from the video game GTA V and is created with three key motivations in mind: i) Providing a much needed comprehensive basis with high fidelity annotations for evaluation of multi person multi camera tracking in the first place, ii) alleviating privacy concerns with depicted subjects, and iii) while direct image processing tasks suffer from a *reality gap* when transferred from synthetic training data to real world test data, this is less true for MCMT tracking, as these methods often do not directly rely on the imagery. We thus contend that methods and design choices developed on our dataset

for MCMT tracking can more directly be transferred to real world data. Following [16], we call our dataset *Multi Camera Track Auto (MTA)*.

Scene Design Unlike existing multi camera tracking datasets (see Table 1), our recorded scene offers a variety of different conditions, ranging from day to night and from sunny to rainy periods. Additionally, cameras mounted indoor and outdoor are included, making the dataset unique in its diversity of scenes and range of conditions. Figure 1 provides an overview of the scene with camera positions and footprints. Please note that only five out of six cameras are shown, the final camera is located inside a metro station. Moreover, one can see that some camera footprints overlap while others are separated.

Dataset Creation For creation of the MTA dataset, we created a GTA plugin based on the one provided by Fabbri et al. [16]. First of all, a mechanism was required which allows to record multiple cameras synchronously. The key challenge in creating a camera network dataset is the requirement of multiple synchronized cameras. GTA is a single player game and does not support this. As a workaround we record camera frames one after the other by changing camera positions and rotations between shots. A slight time offset occurs, as the game proceeds by one frame each time the camera position is changed. We reduce this offset to $\frac{1}{41}$ of a second by activation slow motion mode. Our plugin also takes full control of all persons in the scene. This ensures that no persons with the same appearance but different assigned IDs occur. Paths to be walked are defined by a network graph which is used by our plugin to randomly spawn varying amounts of people with different assigned walking speeds and paths starting from different locations. Our plugin also handles several technical issues, such as avoiding collisions of pedestrians and cars or persons erroneously marked as being visible through walls.

Dataset	IDs	Minutes	Cams	Resolution
Fleuret et al. [18]	≤ 7	3,6,4	≤ 4	320x240
Passageway [1]	4	20	4	320x240
Issia Soccer [15]	25	2	6	1920x1080
Apidis Basket. [13]	12	1	7	1600x1200
PETS2009 [17]	30	1	8	768x576
NLPR MCT [7]	≤ 235	20,20,5,25	≤ 5	320x240
Dana36 [28]	24	N/A	36	2048x1536
USC [21]	146	25	3	852x480
CamNeT [37]	50	30	8	640x480
MOCAT [2]	≤ 100	2,2,2	≤ 10	1920x1080
WILDTRACK [8]	N/A	60	7	1920x1080
Duke (deleted) [30]	2,834	85	8	1920x1080
MTA (ours)	2,840	102	6	1920x1080

Table 1. Comparison of different multi target tracking datasets. Multiple values for duration indicate sets of separate sequences.

MTA Characteristics Table 1 compares our MTA dataset to further widely-used multi target tracking datasets. The MTA dataset consists of 102 minutes of video from each camera at 41 fps, 37,324,348 bounding boxes and 2,840 tracks. Our originally generated data is cut into two equal pieces for training and testing. A small gap of time is discarded to avoid the same person being present in train and test set. As a result, the training set consists of videos with a duration of 00:50:29 per camera and the test set 00:51:59. The dataset includes a wide range of different bounding box sizes, from very small ones with less than 50 pixels in height to large ones with more than 350 pixels. The distribution of bounding box sizes is visualized in Figure 2. The dataset thus represents a realistic scenario since surveillance cameras usually cover large areas and, as a result, a variety of different sizes of persons occur depending on the distance from the camera. Similarly, the distance from camera and pathing can cause a variety of track lengths, see Figure 2. We provide a variety of automatically generated annotations for the MTA dataset. An overview is given in Table 2. Annotations include general information about time, camera position, person identities and positions, as well as 22 body keypoints for each person in 2D and 3D.

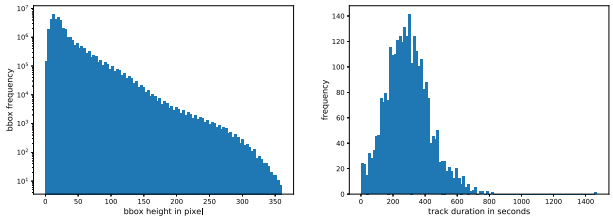

Figure 2. Track duration distribution over the MTA dataset.

Person	Camera
Unique ID	Sync. Frame Number
Position (2D,3D)	Sync. Game Time of Day
Bounding Box (2D)	3D Position
Orientation (Yaw)	3D Rotation
22 Body Joints (2D,3D)	Field of View

Table 2. Types of annotations in the MTA dataset.

MTA Tasks The MTA dataset natively lends itself to evaluation of pedestrian detection, pose estimation, single camera tracking, person re-identification, and, of course, multi camera tracking. We define several subsets to standardize the evaluation of individual tasks. For person detection and pose estimation we select one image per second, i.e., every 41^{st} frame. Our subset for person re-identification, termed MTA-ReID, is created by selecting one image every 5 seconds (205 frames) and cropping individual persons. The larger interval avoids near-identical person images in the subset. We randomly choose 20% of

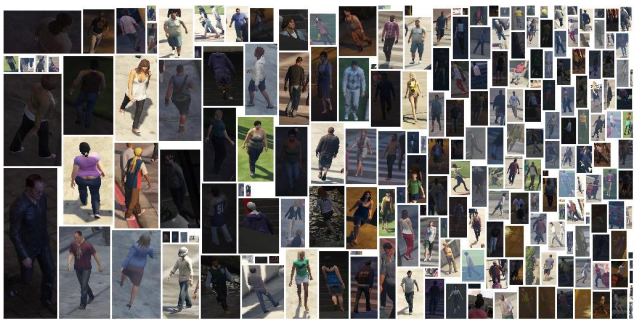

Figure 3. Example images of the MTA-ReID dataset. The relative dimensions of the cropped images are preserved.

the cropped test images for queries and leave the remainder as gallery. Samples of MTA-ReID are visualized in Figure 3. While there are a few re-identification datasets of similar size available, MTA-ReID uniquely offers the possibility to evaluate methods across weather conditions, at night or low light, across indoor and outdoor areas, and at significant differences in image resolution. Note that MTA-ReID also contains distractor images comprised of cropped areas with single or few body joints or test set identities that are not part of the randomly chosen query set.

4. MTMCT by Weighted Distance Aggregation

Our MTMC tracking approach comprises all necessary steps for MTMC tracking from person detection to connecting person tracks from different cameras. The framework has a modular design so that each individual component can be exchanged or adapted to suit the requirements of application in a surveillance scenario. It was designed for offline use which means that the focus is on forensic evaluation of collected video mass data and not on the real-time tracking of persons. However, only small alterations like a track management would be necessary to transform the approach into an online MTMC tracking method. Core components such as the weighted distance aggregation can be used directly. Figure 4 presents an overview of the proposed system. In general, it consists of 5 main components. First, a person detection module predicts person bounding boxes for which person appearance features are computed. Bounding boxes along with corresponding person embeddings are then forwarded to the single camera tracking stage, which calculates and outputs tracklets for each camera view separately. Resulting tracklets are then passed to the core of the framework which is track comparison by computing a set of different feature distances between tracks. Subsequently, tracklets are merged based on the weighted aggregation of track distances using a hierarchical clustering approach. In the following, each component of our framework will be presented in detail.

4.1. Person detection

Since tracking-by-detection approaches are used in the single camera tracking component, it is necessary to detect persons in every frame of every video in advance. A lot of object detectors already exist in literature which are able to recognize pedestrians very robustly and accurately. Which is why the focus was not on this stage of the framework, but instead state-of-the-art detectors are applied.

4.2. Person re-identification

Person re-identification is the problem of finding a person based on its appearance. This means that an image of a person of interest serves as query for retrieving further images showing the same identity from a large number of gallery images. Similar problems arise in the context of multi camera tracking of persons. For instance, single camera tracks can break off due to occlusions or persons can leave a scene and reappear later. If this is the case, person appearances from completed tracks can be used to compare it to new ones in order to reassign persons identities. To make appearances of persons in images comparable, person images are embedded into a feature space using CNNs. Via distance computation in this learned embedding space, the similarity of persons can be determined.

4.3. Single camera tracking

The idea behind our framework is to compute single camera tracklets and connect them within and across camera views in order to get multi camera tracks. The single camera tracking component takes person detections and, if necessary, person embeddings as input and provides single camera tracklets as output that are clustered in the next step in order to combine tracklets of the same persons. A variety of tracking-by-detection single camera trackers exist in literature and are suitable for the framework, ranging from simple intersection over union (IoU) approaches to more sophisticated ones which additionally leverage embeddings extracted from person images, e.g. DeepSORT [34].

4.4. Track association via clustering

The goal of track association is to create groups of tracklets that belong together which means that they come from the same person. Note that our clustering approach not only focuses on combining tracklets across camera views but in addition is able to correct tracking errors resulting from the single camera step. Also within one camera there might be multiple tracks of one person that can be grouped together. E.g. interrupted tracks resulting from single camera tracking errors or occlusions or people leaving a camera view and coming back later. To be able to form a cluster of a person's tracks, a distance metric is required. This metric must meet the requirement that the distance between tracks of the same

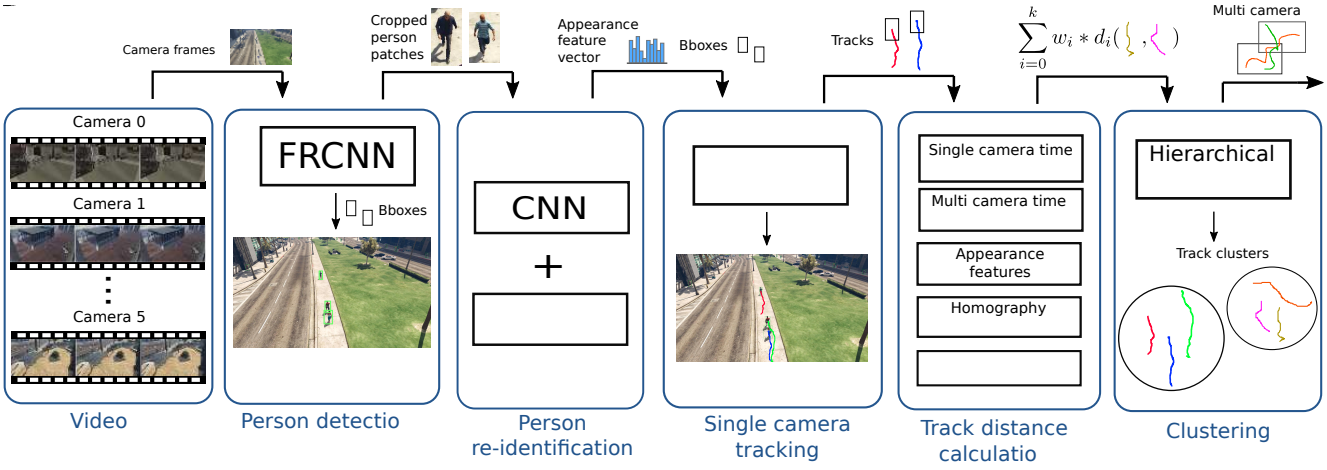

Figure 4. An overview of our proposed MTMC tracking framework.

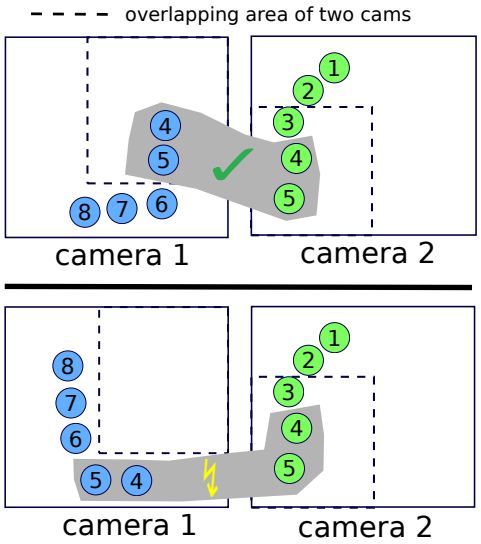

Figure 5. Illustration of the multi camera time constraint.

person are significantly smaller than between tracks showing different people. As a further main contribution of this work, several partial distances were developed in order to fulfill this requirement. In the following, the concept of the partial distances and constraints are explained. The overall distance is calculated by weighted aggregation of 5 individual distances.

Single camera time constraint The single camera time constraint leverages the fact that one person cannot appear in multiple tracks of the same camera at the same time. Therefore, tracks are not connected if they harm this constraint.

Multi camera time constraint The multi camera time constraint exploits that it is impossible for persons to be visible in the views of two non-overlapping cameras at the same time. Again, tracks will only be clustered if this is not the case. An example is visualized in Figure 5. In order to apply this constraint, it is necessary to determine the overlapping areas. If training data is available, this can be done by using ground truth annotations. For every combination of cameras, positions of person ids that appear in both cameras simultaneously can be used to compute convex hulls around this positions to estimate overlapping areas.

Homography matching distance The homography matching distance uses the fact that a person walking through overlapping camera view areas produces tracks in both cameras at the same time. These tracks are obviously not connected by the single camera tracker, although they actually belong to the same track. In order to get an indication of their affiliation, the precise location information of track positions visible in both cameras is used. First, it is checked if bounding box centers are within an overlapping area. This is done by using the convex hulls already computed for the multi camera time constraint. If that is the case the centers of the bounding boxes are transformed from one camera to the other. Projected track positions are counted as match if the transformed position is close to the track detection in this camera at the corresponding time. We transform track positions from one camera view to another by using a linear transformation which we call homography in this context. The homography is computed using point correspondences between cameras, in this case positions of people in overlapping areas from the training data. Based on the RANSAC algorithm [22], homographies for all camera combinations with overlapping areas are determined. In doing so, the centers of the bounding box with larger heights are always transformed to the camera with the smaller bounding box. After calculating all transformations and matches for two tracks, the matching score is defined as the portion of transformation matches. As this score is a similarity measure and can't be calculated for all tracks, it is subtracted from the total distance. In Figure 6 an example for the distance calculation is presented.

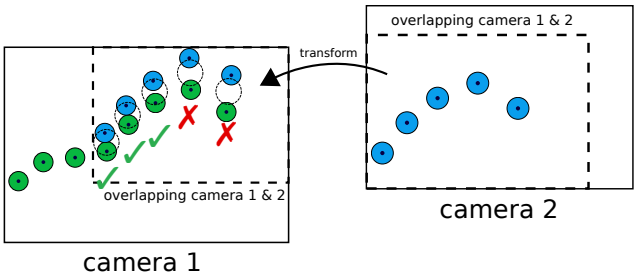

Figure 6. Illustration of the homography matching distance calculation.

Linear prediction discount A big problem with track-

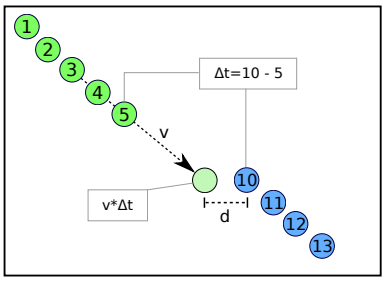

Figure 7. Illustration of the linear prediction.

ing are track interruptions, which can be caused e.g. by occlusions or missing detections. To correct such errors and to connect corresponding tracklets, we exploit the principle that persons often walk along a straight line with a constant velocity. That means it can be estimated where a person might be after a certain time period using a linear prediction model. With these assumptions, the probability of a track being the continuation of another track is estimated based on the time difference of both tracks and the estimated velocity of tracks. As the value which indicates the described property can only be calculated if both input tracks come from the same camera, which is not the case for all track pairs, it has been constructed as a discount. Assuming that two tracks exist: t_o which is an older track and t_y which is a younger track for which this discount should be calculated. A tail with the length n_{tail} is taken from the end of the older track. $n_{tail} = 40$ was chosen because at a frame rate of 41 FPS it corresponds to one second. For such a short time period the assumption of linear motion of pedestrians is valid. An estimated velocity $\vec{v_o}$ of the older track can be calculated by using the position and frame number of the first and last track position of the tail. Let $t_o[-1]$ be the last position of the older track and let Δt be the frame number (time) distance between t_o and t_y, then the predicted position p can be calculated as $p = t_o[-1] + \vec{v_o} * \Delta t$. An example of the calculation is depicted in Figure 7. The distance which indicates if the predicted position p explains the start of the younger track can be calculated via an euclidean distance divided by the bounding box height of the last position of the older track:
$dist_m(t_y[0], p) = \|t_y[0] - p\| / bbox\_height(t_o[-1])$. Dividing by the bounding box height leads to a normalization regarding the different distances resulting from the perspective of the camera view.

As it makes no sense to assume a younger track as the continuation of a older track if the distance $dist_m(t_y[0], p)$ is too large, 0 will be returned if a maximum link distance l_{max} has been exceeded. $l_{max} = v_{mean} * f_{max}$, with v_{mean} as mean velocity of all persons over all cameras and a maximum number of frames $f_{max} = 500$. This maximum number of frames was chosen because after this time it can be assumed for the most people that they have changed direction.

The final discount is then calculated based on Equation 1.

$$d_{pred} = \begin{cases} 0 & dist_m(t_y[0], p) > l_{max} \\ -(1 - \frac{dist_m(t_y[0], p)}{l_{max}}) & \text{otherwise} \end{cases} \quad (1)$$

The range of the discount is $d_{pred} \in [-1, 0]$.

Appearance feature distance The appearance feature distance is based on the CNN embedding described in Section 4.2. Feature vectors are extracted for all bounding boxes of the tracks. Due to variations in illuminations or pose of persons in tracks, the person's track appearance is represented by the mean feature vector over the entire track. The appearance feature distance between two tracks is then calculated by applying the cosine distance metric.

4.5. Hierarchical clustering

As clustering method to group tracks of persons, agglomerative hierarchical clustering is used with an adapted single linkage. The operating principle is that at the beginning of the clustering every track has its own cluster and then always the two clusters with the smallest distance will be merged until a distance threshold or a desired number of clusters has been reached [19]. In detail the clustering works as follows. At the beginning all distances of pairwise track combinations have to be calculated which leads to a space complexity of $O(\frac{n^2}{2})$ for the clustering algorithm. Subsequently, every cluster distance will be inserted into a priority queue based on a heap data structure. Afterwards, the cluster pair associated with the smallest distance at the top of the priority queue will be removed and the contained clusters will be merged together. For this new cluster all distances to other clusters have to be calculated and inserted into the priority queue. There are different so called linkage methods to calculate that distance, like single linkage, complete linkage or average linkage. In this work an adapted single linkage is used. Single linkage normally would search the minimum distance of all track pairs

from a cluster pair with one track out of each cluster. What is changed is, that if one track pair of the clusters has a distance of infinity, infinity is returned instead of the minimum distance. A distance would be infinity if a constraint like the one which avoids time overlaps in a single camera is violated. That means the change of the linkage procedure helps to comply with constraints. After computing the new distances, the next two clusters with the smallest distance will be removed from the priority queue and will be merged. This strategy is repeated until there are only distances greater than a threshold left. We use a fixed threshold of 1 as a stop criterion. As weights are searched for the partial distances and discounts, it is not necessary to determine a threshold, as the weights adapt to this threshold during optimization.

5. Evaluation

In the following, experimental results on the MTA dataset are presented. This includes baseline results for e.g. person detection and person re-identification as well as a thorough evaluation of our MTMC tracking with weighted distance aggregation.

5.1. Person Detection

Implementation & evaluation strategy As an implementation of the different detection approaches the detection toolbox from Chen et al. [9] was used and approaches were evaluated using the *cocoapi* [11]. For all approaches the values of the default toolbox configuration files were used. Just the input image size was increased to 1920x1080 pixels. The depicted values of average recall (AR) and average precision (AP) result from an evaluation mode were an average of evaluation values resulting from different IoU (intersection over union) thresholds is calculated. The IoU threshold denotes the percentage of overlap which is required between calculated detection and groundtruth detection that a true positive is counted. The used thresholds range from 0.5 to 0.95 with a step size of 0.05, which means that 10 thresholds were used to calculate the average. Note that bounding boxes of all sizes were incorporated into the calculation. A maximum count of 100 detections per frame was used to calculate the shown scores.

Results We rely on R-CNN approaches because they show good performance in detecting small and overlapping objects compared to, e.g., SSD architectures [25]. Since people in the back areas of surveillance camera recordings are usually very small, such detectors suit our needs very well. The first row in Table 3 shows the MTA evaluation results for Faster R-CNN [29] with ResNet-50 (RN-50) as backbone trained on the COCO dataset [24]. Looking at the first two rows of Table 3, one can observe that training detectors on COCO or similar datasets has the problem that only a few small people are included in the training data. Therefore, a noticeable performance gap exists to detectors that have been trained on the MTA dataset. When using ResNext-101 (RNX-101) [35] as a backbone for Faster R-CNN instead of ResNet-50 [20] an improvement by +2.8 AR percentage points could be observed. The best results with 69.5% AR and 67.0% AP were achieved when using Cascade R-CNN with ResNext-101 [6].

	Approach	Trained on	AR	AP
RN-50	Faster R-CNN [29]	COCO	14.6	11.3
	Faster R-CNN [29]	MTA	64.8	61.6
RNX-101	Faster R-CNN [29]	MTA	67.6	64.9
	Cascade R-CNN [6]	MTA	**69.5**	**67.0**
	RetinaNet [23]	MTA	67.3	62.8

Table 3. Person detection evaluation results on the MTA dataset.

5.2. Person re-identification

In this section the evaluation results for person re-identification will be discussed.

Parameters & implementation We use the official implementations and standard parameters of state-of-the-art approaches to provide baseline results on the MTA-ReID dataset. Note that the results for our tracking method presented in Section 5.3 were achieved using an extended version of the MTA-ReID dataset.

Results Person re-identification results for three state-of-the-art approaches are shown in Table 5.2. In line with results on other datasets, the ABD-Net[10] achieves the best mAP of 30.5%, followed by the AGW [36] approach. The Strong Baseline approach from Luo et al. [26] resulted in 25.8% mAP. For all methods random erasing data augmentation (RE) led to a performance degradation because additional data augmentation is not necessary due to the large number of images in the MTA-ReID dataset. To demonstrate the benefit and practical relevance of the synthetic data, Table 5.2 provides results when approaches were trained on the artificial MTA data and evaluated on the real-world Market-1501 [39] dataset. For comparison, the first row shows the mAP score of 25.5% for the Strong Baseline [26] approach trained on DukeMTMC-reID [30]. It is observable the each of the approaches trained on the MTA data outperforms this significantly. In this case, AGW leads to the best mAP score of 33.7%. So it can be stated that the domain gap between the synthetic MTA dataset and real-world images is similar or even smaller than the domain gap between datasets recorded from the real world. Regarding cross-domain person re-identification, the most complex ABD-Net leads to the worst performance because it tends to overfit the training data and thus the learned information is too specific for domain transfer. Note that since the Market-1501 and DukeMTMC-reID dataset only

Approach	mAP	R-1	R-5	R-10
Strong Baseline [26]	25.8	50.5	69.7	74.9
Strong Baseline [26] + RE	22.4	46.6	66.1	72.0
AGW [36]	27.7	53.8	71.7	76.5
AGW [36] + RE	26.0	50.8	69.7	75.0
ABD [10]	**30.5**	**56.6**	72.4	76.8
ABD [10] + RE	30.3	56.5	**73.0**	**77.0**

Table 4. Person re-identification results on our MTA dataset.

Approach	Trained on	mAP	R-1	R-5	R-10
Strong Baseline* [26]	Duke	25.5	54.3	–	–
Strong Baseline [26]	MTA-ReID	31.1	60.7	75.7	80.8
AGW [36]		**33.7**	**64.0**	**77.4**	**82.6**
ABD [10]		27.6	54.4	71.0	78.3

Table 5. Cross-domain person re-identification results on Market-1501 dataset. Note that we restricted the MTA-ReID training data to images with a height of more than 65 pixels for fair comparison. (* result was taken from literature)

include a fixed image size, we restricted our training data to images with a height of more than 65 pixels for fair comparison.

5.3. Tracking

Implementation & evaluation strategy In surveillance scenarios, the core task is to track people across multiple cameras for as long as possible. Identity metrics [30] best reflect this requirement and are therefore used below as evaluation metrics. The evaluation scores which are presented were achieved by dividing the MTA test set into 10 parts with an approximate length of 5 min each with subsequent tracking and evaluation on every part as otherwise the evaluation time would have been extremely long. The depicted scores denote the mean over all ten parts. To calculate the identity metrics it is necessary to find matches of computed tracks and ground truth tracks that minimize the overall number of detection misses (FP and FN). This problem can be formulated as a minimum weight bipartite graph matching problem. In this work, the IDF1 metric was selected as the target metric, which was used to measure an improvement in the tracking procedure. Due to practical considerations regarding computation time, we decided to rely on Faster R-CNN and Strong baseline as standard configurations for person detection and re-identification.

Single camera tracking results Results on the MTA dataset for different single camera tracking approaches are presented in Table 6. The depicted values represent the mean over the 6 cameras. The DeepSORT [34] approach greatly outperforms the IoU tracker. One main reason for that is the use of person embeddings in order to not only consider detections but instead leverage appearance features of pedestrians.

Tracker	IDF1	IDP	IDR	IDs
IoU [3]	38.1	40.9	35.8	2370.3
DeepSORT [34]	**42.0**	**45.1**	**39.6**	**1797.8**

Table 6. Tracking results for two different single camera trackers.

Configuration	IDF1	IDP	IDR	IDs
None	17.3	19.2	15.7	11535.3
All	**30.1**	33.6	**27.3**	7107.5
w/o Single Camera Time Constraint	26.8	**41.9**	24.3	6869.5
w/o Multi Camera Time Constraint	26.0	28.9	23.6	**6693.0**
w/o Homography	28.5	31.8	25.9	6781.3
w/o Linear Prediction	29.7	33.0	26.9	7726.1

Table 7. MTMC tracking results and ablation study to determine the influence of distances on the Weighted Distance Aggregation approach.

Multi camera tracking results Table 7 presents the final multi camera tracking results as well as an ablation study. Results without person embeddings are not shown since we use this as basis distance for the clustering. If all distances and constraints are used an improvement from 17.3% to 30.1% in IDF1 is achieved. Leaving out the time constraints leads to the largest drop in performance. The reason for this is that if these constraints are harmed, tracks to be merge cannot belong to the same person and thus always lead to tracking errors. In contrast, the difference between 'All' distances and without the linear prediction distance is low because this only improves the results from the single camera trackers and therefore leads to improvements of only a few tracks. The influence of the homographies lies in between, because on the one hand they can only be applied to overlapping areas. On the other hand, the camera footprints overlap mainly in areas that are in the focus of one camera, but far from a second one. As a result, stable detections are available in one camera that can be transformed to the second camera for which only a few small detections are available.

6. Conclusion

In summary, we provide a new and currently largest synthetic dataset for development and evaluation of multi camera multi person tracking methods. Our dataset contains a diversity of weather conditions, times of day, indoor and outdoor scenes and is not susceptible to privacy claims. We provide useful baseline results for person detection, re-identification, single camera tracking and our own MTMC tracking method. While we achieve reasonable accuracies, the results also show the challenging nature of the dataset and room for improvement remains. We hope that the data and reference results will spark further activities in the field.

References

[1] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua. Multiple object tracking using k-shortest paths optimization. *IEEE transactions on pattern analysis and machine intelligence*, 33(9):1806–1819, 2011.

[2] Erik Bochinski, Volker Eiselein, and Tomas Sikora. Training a convolutional neural network for multi-class object detection using solely virtual world data. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 278–285. IEEE, 2016.

[3] Erik Bochinski, Volker Eiselein, and Thomas Sikora. High-speed tracking-by-detection without using image information. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[4] Francesco Bonchi, David Garcia-Soriano, and Edo Liberty. Correlation clustering: from theory to practice. 2014.

[5] Michael Bredereck, Xiaoyan Jiang, Marco Körner, and Joachim Denzler. Data association for multi-object tracking-by-detection in multi-camera networks. In *2012 Sixth International Conference on Distributed Smart Cameras (ICDSC)*, pages 1–6. IEEE, 2012.

[6] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[7] Lijun Cao, Weihua Chen, Xiaotang Chen, Shuai Zheng, and Kaiqi Huang. An equalised global graphical model-based approach for multi-camera object tracking. *arXiv preprint arXiv:1502.03532*, 2015.

[8] Tatjana Chavdarova, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Louis Lettry, Pascal Fua, Luc Van Gool, and François Fleuret. The wildtrack multi-camera person dataset. *arXiv preprint arXiv:1707.09299*, 2017.

[9] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[10] Tianlong Chen, Shaojin Ding, Jingyi Xie, Ye Yuan, Wuyang Chen, Yang Yang, Zhou Ren, and Zhangyang Wang. Abd-net: Attentive but diverse person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8351–8361, 2019.

[11] cocodataset. cocodataset/cocoapi: Coco api - dataset @ http://cocodataset.org/. https://github.com/cocodataset/cocoapi. (Accessed on 03/26/2020).

[12] Abir Das, Anirban Chakraborty, and Amit K Roy-Chowdhury. Consistent re-identification in a camera network. In *European conference on computer vision*, pages 330–345. Springer, 2014.

[13] Christophe De Vleeschouwer, Fan Chen, Damien Delannay, Christophe Parisot, Christophe Chaudy, Eric Martrou, Andrea Cavallaro, et al. Distributed video acquisition and annotation for sport-event summarization. *NEM summit*, 8, 2008.

[14] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixe. Cvpr19 tracking and detection challenge: How crowded can it get? *arXiv preprint arXiv:1906.04567*, 2019.

[15] Tiziana D'Orazio, Marco Leo, Nicola Mosca, Paolo Spagnolo, and Pier Luigi Mazzeo. A semi-automatic system for ground truth generation of soccer video sequences. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 559–564. IEEE, 2009.

[16] Matteo Fabbri, Fabio Lanzi, Simone Calderara, Andrea Palazzi, Roberto Vezzani, and Rita Cucchiara. Learning to detect and track visible and occluded body joints in a virtual world. In *European Conference on Computer Vision (ECCV)*, 2018.

[17] J Ferryman and A Shahrokni. An overview of the pets 2009 challenge. 2009.

[18] Francois Fleuret, Jerome Berclaz, Richard Lengagne, and Pascal Fua. Multicamera people tracking with a probabilistic occupancy map. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):267–282, 2007.

[19] G. Gan, C. Ma, and J. Wu. *Data Clustering: Theory, Algorithms, and Applications*. ASA-SIAM Series on Statistics and Applied Probability. Society for Industrial and Applied Mathematics, 2007.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[21] Cheng-Hao Kuo, Chang Huang, and Ram Nevatia. Inter-camera association of multi-target tracks by on-line learned appearance affinity models. In *European Conference on Computer Vision*, pages 383–396. Springer, 2010.

[22] S. Li, C. Liu, and Y. Wang. *Pattern Recognition: 6th Chinese Conference, CCPR 2014, Changsha, China, November 17-19, 2014. Proceedings*. Number Teil 1 in Communications in Computer and Information Science. Springer Berlin Heidelberg, 2014.

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[24] 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.

[25] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[26] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[27] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[28] Janez Per, Vildana Sulic Kenk, Matej Kristan, Stanislav Kovacic, et al. Dana36: A multi-camera image dataset for object identification in surveillance scenarios. In *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, pages 64–69. IEEE, 2012.

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[30] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer, 2016.

[31] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6036–6046, 2018.

[32] Facepunch Studios. Garry's mod. https://gmod.facepunch.com/. (Accessed on 01/31/2020).

[33] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3539–3548, 2017.

[34] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649. IEEE, 2017.

[35] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[36] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. Deep learning for person re-identification: A survey and outlook. *arXiv preprint arXiv:2001.04193*, 2020.

[37] Shu Zhang, Elliot Staudt, Tim Faltemier, and Amit K Roy-Chowdhury. A camera network tracking (camnet) dataset and performance baseline. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 365–372. IEEE, 2015.

[38] Zhimeng Zhang, Jianan Wu, Xuan Zhang, and Chi Zhang. Multi-target, multi-camera tracking by hierarchical clustering: Recent progress on dukemtmc project. *arXiv preprint arXiv:1712.09531*, 2017.

[39] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015.