Multi-camera People Tracking With Mixture of Realistic and Synthetic Knowledge

Quang Qui-Vinh Nguyen[1,2], Huy Dinh-Anh Le[1,2], Truc Thi-Thanh Chau[1,2], Duc Trung Luu[1,2], Nhat Minh Chung[1,2], Synh Viet-Uyen Ha[1,2,*]

[1] School of Computer Science and Engineering, International University, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam

Abstract

This paper presents a solution for Track 1 of the AI City Challenge 2023, which involves Multi-Camera People Tracking in indoor scenarios. The proposed framework comprises four modules: Vehicle detection, ReID feature extraction, single-camera multi-target tracking (SCMT), single-camera matching, and multi-camera matching. A significant contribution of our approach is the introduction of ID switch detection and ID switch splitting using the Gaussian mixture model, which efficiently addresses the problem of tracklets with ID switches. Furthermore, our system performs well in matching both synthetic and real data. The proposed R-matching algorithm performs exceptionally well in real scenarios despite being trained on synthetic data. Experimental results on the public test set of 2023 AI City Challenge Track 1 demonstrate the efficacy of the proposed approach, achieving an IDF1 of 94.17% and securing 2nd position on the leaderboard. Codes will be available at https://github.com/nguyenquivinhquang/Multi-camera-People-Tracking-With-Mixture-of-Realistic-and-Synthetic-Knowledge

1. Introduction

Multi-Target Multi-Camera (MTMC) tracking approaches pose significant challenges as they require solving multiple computer vision problems, including person detection, single-camera multi-target tracking, and person re-identification. These challenges involve addressing variations in camera resolution, distance, view angle, non-overlapping camera views, crowded areas, and changes in illumination. Furthermore, due to the limited availability of labeled datasets, multi people multi-camera tracking remains a challenge.

The 7th AI City Challenge workshop's AIC23 [34] Track 1 dataset aims to address these challenges by combining real and synthetic data to track people across multiple cameras. However, the dataset's indoor setting poses additional challenges as people can occur multiple times under each camera, which is not typically the case in urban environments. Additionally, the dataset's training data is entirely synthetic, making it challenging to train models with discriminative features for real data. A desired solution ought to carefully address not only the domain gap between synthetic test data and training data but also generalize on the real-life test data with suitable constraints.

In this paper, we present a new MCMT tracking system designed to track multiple people across different cameras in an indoor scenario. Existing multi-object multi-camera tracking systems [15], [17], [6], and [43] only track objects until they exit the camera's field of view, making it difficult to track them if they re-enter later. In contrast, our framework can track people even if they exit and re-enter the camera's field of view. Additionally, our framework detects when a tracklet has two different user IDs and adjusts the tracklet to improve the results. Most importantly, our proposed solution can be applied on both real and synthetic test settings, despite having been trained on only synthetic data.

In order to accomplish the above, our proposed system presents several technical contributions, most notably including: (a) Our detector and single-camera tracking module can detect and track people effectively in both synthetic and real data. Additionally, we have incorporated a Gaussian Mixture Model (GMM) to alleviate the problems of ID switches. (b) Our single-camera matching and multi-camera matching modules perform well on both synthetic and real data. For real data, we introduced an R-matching algorithm to tackle the domain gap from the synthetic training set, significantly improving the IDF1 scores by 8.46%. (c) Our system's performance has been evaluated in the Multi-camera people tracking track of the 2023 AI City

*Corresponding author. Email: hvusynh@hcmiu.edu.vn

Challenge, where we achieved second place.

2. Related Works

Various designs for an MTMC tracking system have been proposed over the recent years, as summarised by Naphade et.al. [33] [32]. Authors have typically followed the aforementioned processes: (1) Object Detection, (2) Multi-target Single-camera Tracking, (3) Appearance Feature Extraction, and (4) Cross-Camera Tracklet Matching. The performance of a particular design apparently correlates with how well authors can develop contrastive models for extracting appearance features and constrain the data domain's search space [29] [55] [47].

2.1. Object Detection

An object detection model is essential in determining vehicle positions throughout a camera image. Many state-of-the-art models have been proposed that include single-shot detectors such as YOLOv4 [1], YOLOv5 [10], YOLOv6 [24], YOLOv7 [44] CenterNet [9], and EfficientDet [42] to directly output object positions alongside their classes, and two-shot detectors Mask-RCNN [11] and Cascade-RCNN [3] that rely on the generation of bounding box priors before classifying them. In the MTMC vehicle tracking literature, authors [29] [47] have leveraged pre-trained models' generalization capabilities without training on the test set with good results.

2.2. Multi-Target Single-Camera Tracking

The aim of tracking is to identify the trajectory, i.e. a set of bounding boxes, of each ID using the detection result. Most recent works on MOT are categorized into two methods: online and offline, which differ in the way of observation processing. Online approaches [20] [58] [59] [48] [56] [39] [46] utilize the bounding boxes on the current frame to extend the existing trajectories, which results in a short processing time, but lower accuracy. On the other hand, offline approaches [40] [36] [52] [23] [16] [41] [5] [50] [2] [51] gain higher accuracy due to optimizing the solution for linking all of the bounding boxes in the video with different trajectories. In order to improve the performance on switching ID and missing tracking occluded objects cases, DeepSORT [46], an online tracking framework, uses a CNN model that is trained on a large-scale person ReID dataset to learn a deep association metric which combines motion and appearance information. the Yang et al. [53] modified DeepSORT to deal with occlusion and applied forward and backward tracking in time. MedianFlow [22], an offline tracking framework, also performed forward and backward tracking in time and then compared the two trajectories to detect the tracking failures based on the assumption that the tracklet is independent of the direction of time flow. Li et al. [25] implemented the modified version of MedianFlow to tackle the cases that the object moves too fast or moves with a rapidly direction-changing trajectory. As this version is combined with Efficient convolution operators to evaluate the correlation or similarity of two signals, the ID switches could be reduced.

2.3. Image-based Re-Identification

Vision-based re-identification refers to recognizing the same object across different images or videos captured from different cameras, which take a critical role in MTMC tracking problems. The challenging problem is that depending on each camera and point in time, lighting conditions, occlusions, pose variations, and camera viewpoint could not be invariant, which results in the visual difference of a target vehicle. The state-of-the-art techniques in image-based reID can be categorized into two main types: feature-based and deep learning-based methods. Feature-based methods [35] [28] [7] extract handcrafted features, i.e the features containing the information from images such as color, texture, and shape information, and use them for matching. Regarding deep learning-based methods, many approaches [12] [19] [49] gain great performance by implementing CNN-based models to learn robust feature representations. However, CNN-based methods still have limitations as they focus on a local neighborhood and lose information due to downsampling operations. Thus, recent approaches [13] [8] deployed Transformers to improve the result by the ability to capture global relations. For example, TransReID [13] encodes an image as a series of patches and establishes a transformer-based strong baseline with a few essential enhancements. Besides, to tackle the low-diverse and limited dataset, many GAN-based ReID works [4] [62] [30] [61] [27] have been proposed.

2.4. Cross-Camera Tracklet Matching

For MTMC tracking, an appropriate method to associate tracklets across cameras is indispensable. Many approaches [57] [37] [26] [21] [18] [38] principally utilize the embedding feature vectors of IDs to compute the appearance similarity, then evaluate the result to match the tracklets. However, relying solely on appearance features may be prone to ID switches. To enhance the performance, the majority of works [14] [6] [25] [53] [54] [60] combine different constraints such as camera topology, temporal information, motion rules, etc.

In this work, we improve performance by first performing single-camera matching and then using the results to enhance multi-camera matching rather than matching multiple cameras directly.

3. Methodology

This section introduces our proposed frameworks, which consist of five components, namely, person detection, single-camera tracking, single-camera matching, and multi-camera matching. An overview of our framework is presented in Figure 1.

3.1. Person Detection-based Continuous Tracking

Our first major component is designed to track people in a single camera robustly. Motivated by the problems of occlusion (e.g. two people overlapping each other) and intermittent tracking (e.g. temporarily missing detections), we aim to robustly tackle ID switches and maintain consistent track trajectories with a detection-based tracking framework. Hence, we propose a 2-stage approach via (1) scene-conditioned person detection and (2) continuous tracking with a mixture model.

3.1.1 Scene-conditioned Person Detection

Person detection plays a crucial role in initiating Multi-camera people tracking. Our approach utilizes YOLOv5, a state-of-the-art network, specifically the pre-trained YOLOv5x6 model on the COCO dataset for object detection. Our approach seeks to address a major challenge: the domain gap between the synthetic dataset and the pre-trained COCO dataset used to train the model. To address this, we propose to develop meta-data for denoting scenes of real and synthetic scenes and apply an appropriate detection model:

On real-life scenes: We rely solely on the pre-trained model to achieve optimal performance. This is because the pre-trained model has already been trained on the COCO dataset to capture the features of a real person and the limited domain gap between real-life scenes in certain settings.

On synthetic scenes: We fine-tuned our model using the synthetic animated people dataset using the NVIDIA Omniverse Platform, particularly adapting Yolov5x6 to a similar test domain. By leveraging pre-trained patterns in the model, the model is tuned for person detection on the synthetic domain's properties.

Regardless of the scene, if an input frame I_t captured at a specific time step t, we can extract a set of m detections denoted as $\mathbf{D}_t := \{\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3, ..., \mathbf{d}_m\}$. Each detection $\mathbf{d}_i$ can be represented by a tuple $\mathbf{d}_i := (x_i, y_i, w_i, h_i, c_i, t)$, which represents the i-th bounding box coordinates with the centre at x and y, and size of width w, height h; finally c_i represents the confidence score of the bounding box, which we used to ensure that $c_i > c_\epsilon$. In our case, $c_\epsilon = 0.5$.

3.1.2 Continuous Person Tracking on Mixture Model

Our Single-Camera Multi-Target Tracking system utilizes mostly reliable detection results, and we use features extracted by models in subsection 3.2 to associate targets across video frames using the tracking-by-detection paradigm. For our case, we adapted the DeepSORT tracker as our baseline method and implemented advanced techniques as mentioned in [53], such as occlusion handling, interpolating missing detections with Kalman-Filter-predicted boxes, and forward-backwards tracking. However, since it is common for individuals to walk in unpredictable patterns such as straight lines, stopping, turning around, and continuing on, or for one person to stop and another to pass closely by. We propose a Gaussian Mixture Model-based approach for splitting ID-switching tracklets. In particular, suppose a tracklet $\mathbf{t}$ is denoted by $\mathbf{t} := \{(\mathbf{d}_1, \mathbf{f}_1), (\mathbf{d}_2, \mathbf{f}_2), ..., (\mathbf{d}_k, \mathbf{f}_k)\}$, where there are k detections for this tracklet and $\mathbf{f}_i = f(\mathbf{d}_i)$ denotes the extracted feature vector of the i-th detection.

Since each tracklet should belong to only one identity, the distribution of the features should follow a Gaussian distribution, with a single mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$ that captures the overall characteristics of the tracklet. Therefore, we propose an ID switch detection approach to identify the tracklet where ID switching occurs. Then, we utilize ID Switch splitting to split this tracklet into two separate tracklets by modelling each tracklet's feature set in a Gaussian Mixture Model. In particular, given a feature set, we model it as a Gaussian Mixture Model through Expectation Maximization, i.e. minimizing the likelihood:

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{j=1}^{k} \boldsymbol{\pi}_j \mathcal{N}(\mathbf{f}_i | \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}) \tag{1}$$

where $\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ are the mixture model parameters, with $\mathcal{N}(\mathbf{f}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ as the Gaussian probability density function with mean $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$. $\boldsymbol{\pi}_j$ is the mixture coefficient for the j-th Gaussian component.

ID switch detection: With $n = 2$, we constructed two Gaussian distributions on $\mathbf{t}$. Next, we compute the cosine distance between the mean points of these two distributions. If the distance is found to be less than a pre-defined threshold (which we set to 0.4), we can infer that an identity switch has occurred.

ID switch splitting: Let $\boldsymbol{\theta}_1 = (\boldsymbol{\pi}_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\boldsymbol{\theta}_2 = (\boldsymbol{\pi}_2, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ be the two Gaussian distributions obtained from GMM, to split the tracklet, we use the longest subsequence of consecutive bounding boxes in the higher-weighted Gaussian distribution, starting from the first box. We assign these boxes to the first person and the remaining to the other. Specifically, if $\boldsymbol{\pi}_1 > \boldsymbol{\pi}_2$, let s and e be the start and end indices of the longest subsequence of consecutive bounding boxes that are clustered to $\boldsymbol{\theta}_1$, then we assign the bounding boxes in $\{(\mathbf{d}_s, \mathbf{f}_s), (\mathbf{d}_{s+1}, \mathbf{f}_{s+1}), ..., (\mathbf{d}_e, \mathbf{f}_e)\}$ to the first person and the remaining boxes to the other person, and the same otherwise when $\boldsymbol{\pi}_1 \leq \boldsymbol{\pi}_2$. This splitting strategy alleviates ID switches and improves tracking performance.

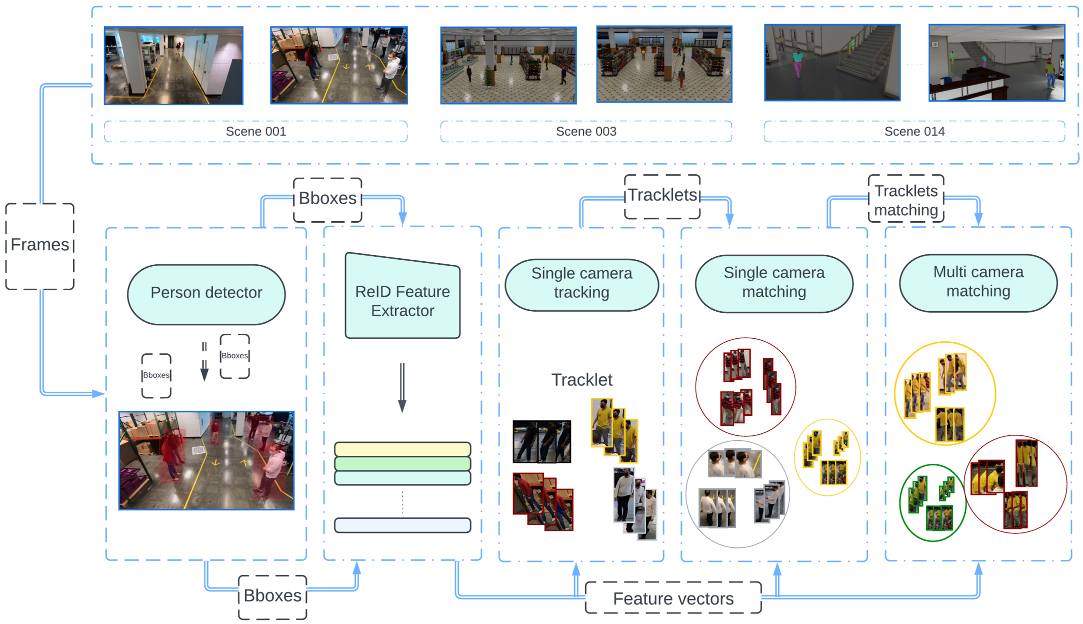


Figure 1. An overview of our system. Our proposed framework consists of five main components: person detection, single-camera tracking, single-camera matching, multi-camera matching, and ReID feature extraction.

3.2. Person Re-identification

3.2.1 Feature Extractor

Re-identification is a critical component of Multi-Camera Multi-Target (MCMT) tracking, as it relies on obtaining each individual's reliable and discriminative appearance features. To achieve this, we explore two types of deep feature extraction methods: (1) transformer-based and (2) CNN-based for our feature extraction model. For transformer-based methods, we use TransReID, while for CNN-based methods, we employ a range of models such as ResNet [12], ResNeXt [49], and HRNet [45]. After that, we apply bags of tricks from [31], which achieves state-of-the-art results in the re-identification field, to train our ReID model. The input image was resized to 256×128 in the training and feature extraction stages. We apply some data augmentation for the preprocessed input data, such as random horizontal flip, random erasing and random padding.

For the feature extraction stage, we generate a global feature with the dim of 2048 before batch normalization neck as the final output of the input image. We simply concatenate the feature extract from each model above for the ensemble ReID feature. As mentioned previously, we refer to our feature extractor as $f(\cdot)$, which represents either our use of TransReID, ResNet50, NesNeXt101, HRNet or an ensemble of them.

3.2.2 Objective Losses

We jointly used (1) an ID loss function using cross-entropy loss with label smoothing, and (2) a contrastive loss function using triplet loss for the optimization.

Regarding ID loss, the probability that person image $\mathbf{x}$ corresponds to person i is denoted as $p(i|\mathbf{x})$. Let the true person ID being represented by y, the cross-entropy loss with label smoothing is defined as:

$$L_{\text{ID}} = \mathbb{E}_{\mathbf{x},y}\left[\sum_{i=1}^{N} -q(i|y)\log\left[p(i|\mathbf{x})\right]\right] \tag{2}$$

such that $q(i|y)$ is the smoothed label distribution:

$$q(i|y) = \begin{cases} 1 - \frac{N-1}{N}\varepsilon & \text{if } i = y \\ \varepsilon/N & \text{otherwise} \end{cases} \tag{3}$$

where N denotes the number of persons in the training set. At the same time, ε is a small positive constant that regulates the smoothing level applied to the label distribution. This smoothing technique prevents the model from overfitting to person IDs of the training set.

Regarding Triplet loss, the goal is to minimize the distance between an anchor sample x^a and a positive sample x^p while maximizing the distance between the anchor and a

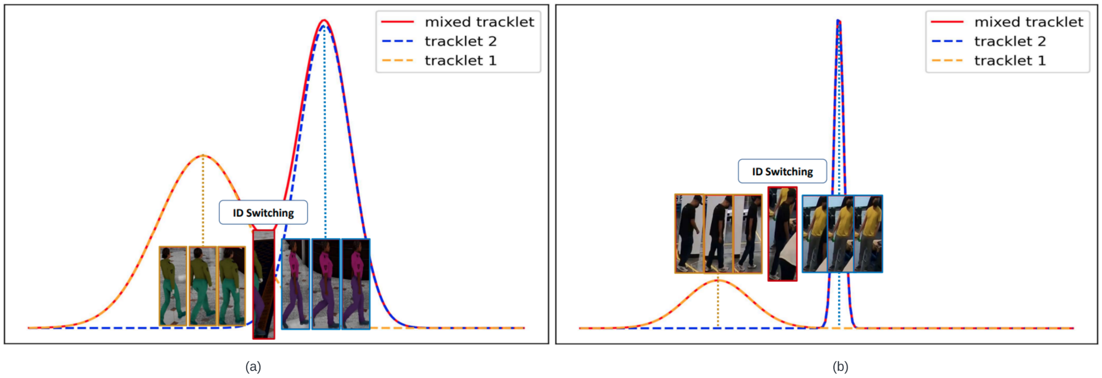


Figure 2. Examples of ID switching scenarios that can occur in tracklets from simplified GMM viewpoints.

negative sample x^n. This is achieved by setting a margin m and optimizing the following loss function for N samples:

$$L_{tri} = \sum_{i=1}^{N} \max\left(m + d\left(\mathbf{f}_i^a, \mathbf{f}_i^n\right) - d\left(\mathbf{f}_i^a, \mathbf{f}_i^p\right), 0\right) \tag{4}$$

Here, $\mathbf{f}^a, \mathbf{f}^p, \mathbf{f}^n$ denote the feature embeddings of the anchor, positive, and negative samples, respectively, and $d(\cdot)$ calculates the distance between two feature embeddings. Hence, t he combined loss function that we used is:

$$L_{reid} := L_{\text{ID}} + L_{tri} \tag{5}$$

3.3. Single-camera Tracklet Matching

In dealing with inconsistent tracking trajectories within a camera, e.g. a person moving out of the view and back again, we propose a single-camera matching approach to cluster-generated tracklets in terms of appearance feature. The purpose of single-camera tracklet matching is to generate groups of tracklets originating from the same person id, within the same camera.

To match the people under the camera, we find the group of tracklet belonging to one person by clustering the tracklets together based on their similarities. Suppose the function $f(\cdot)$ when applied on a tracklet will extract the average feature vector across all features, then the similarity of two tracklets $\mathbf{t}_i$ and tracklet $\mathbf{t}_j$ is determined by the cosine distance formula:

$$\text{dist}\left(\mathbf{t}_i, \mathbf{t}_j\right) = 1 - \frac{f\left(\mathbf{t}_i\right) \cdot f\left(\mathbf{t}_j\right)}{\left\|f\left(\mathbf{t}_i\right)\right\| \left\|f\left(\mathbf{t}_j\right)\right\|} \tag{6}$$

For all pairs of tracklets, we can then generate a single-camera distance matrix of all tracklets,

Because the tracklets represent the short trajectory of a person's motion over time. The longer tracklet, the more informative feature can have. Moreover, shorter tracklets are more likely to be affected by occlusions, which occur when another object partially or completely blocks the object. In such cases, the tracklet may not capture the full motion or appearance of the object, leading to incomplete or inaccurate feature information. This can also contribute to the noisy feature of short tracklets. Therefore, the feature information of short tracklets will not be as obvious as those of tracklets with a length greater than α. We denote the set of removed short tracklets as $\mathbf{R}$ Once the short tracklets have been removed, the remaining cluster group contains sufficient information to act as an anchor for matching another tracklet. After temporarily removing tracklet noise, the obvious tracklets matrix distance will be calculated:

$$D := \begin{bmatrix} \text{dist}\left(\mathbf{t}_1, \mathbf{t}_1\right) & \cdots & \text{dist}\left(\mathbf{t}_1, \mathbf{t}_N\right) \\ \vdots & \ddots & \vdots \\ \text{dist}\left(\mathbf{t}_N, \mathbf{t}_1\right) & \cdots & \text{dist}\left(\mathbf{t}_N, \mathbf{t}_N\right) \end{bmatrix} \tag{7}$$

Furthermore, if the sequence in terms of time steps between two tracklets $\mathbf{t}_i, \mathbf{t}_j$ intersects, then in $D_{i,j} \leftarrow \infty$, tracklets will then be clustered together based on the distance matrix D.

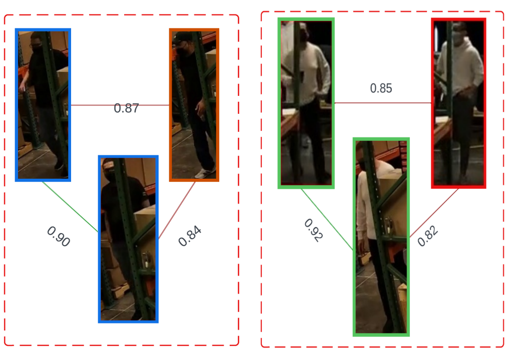
In synthetic scenes, our approach has a high discriminative ability to distinguish between the features of different people, the accuracy of clustering using traditional algorithms on this dataset is significantly high. Hence, the clustering approach can perform robustly. For synthetic scenes, the set $\mathbf{R}$ of short tracklets are simply discarded.

In real-life scenes, however, the feature model's ability to discriminate between individuals with different features for inter-class persons is unclear, as it is trained on the synthetic person dataset. Figure 3b depicts the similarity score between two individuals wearing white shirts is 0.85. Hence, during clustering, inter-class persons may be incorrectly grouped together. To address this issue, we propose an algorithm that can overcome this issue. First, we select the frames with the highest number of people. Then, we construct a graph for the individuals in each frame, with vertices representing people and edges determined by their cosine similarity. Then, we select the frame with the smallest sum of edges as the initial cluster. Finally, we employ the algorithm depicted in algorithm 1 to merge tracklets that fit into the same cluster.

After determining the group of clusters, denoted by $\mathbf{C}_v$ as the v-th set of different tracklets, the feature of each cluster will calculate based on the following:

$$f(\mathbf{C}_v) = \frac{1}{|\mathbf{C}_v|} \sum_{\mathbf{t}_i \in \mathbf{C}_v} f(\mathbf{t}_i) \tag{8}$$

We treat each cluster $\mathbf{C}_v$ as a gallery list and all the temporarily removed tracklets (mentioned above) as a query list $\mathbf{R}$. Then we simply solve a ReID problem with the method outlined in [31] to find the best matches for each sample in $\mathbf{R}$ and each cluster $\mathbf{C}_v$. This enables us to group both long and short tracklets in one cluster from the same person in a single camera. We refer to this extension for real-life scenes as **R-matching**.



(a) (b)

Figure 3. High inter-class similarity. The red line indicates that two individuals have different IDs in both figures, while the green line represents a match. In Figure (a), there are different individuals who happen to be wearing similar clothing. If similarity scores were used for matching, it would result in an incorrect match. The same issue occurs in Figure (b).

3.4. Multi-camera Tracklet Matching

The idea of multi-camera tracklet matching is to match the person's ID across multiple cameras. Here, the person id in each camera was represented by the cluster in that camera. We propose using the aforementioned cluster feature calculations to perform Agglomerative Clustering. Therefore, the distance between two clusters was defined by:

$$\mathrm{dist}\left(\mathbf{C}_v, \mathbf{C}_u\right) = 1 - \frac{f\left(\mathbf{C}_v\right) \cdot f\left(\mathbf{C}_u\right)}{\left\| f\left(\mathbf{C}_v\right)\right\| \left\| f\left(\mathbf{C}_u\right)\right\|} \tag{9}$$

Figure 4. Single-camera matching. The resulting clusters, depicted as circles in the figure, serve as the gallery for the remaining noisy tracklets to match against.

Algorithm 1 R-Matching

function MATCHING($tracklet\_list$, $cluster\_init$, τ)
    $uncertain \leftarrow \{\}$
    for $track$ in $tracklet\_list$ **do**
        $valid\_clusters \leftarrow \{\}$
        for $cluster$ in $clusters\_init$ **do**
            if $\neg time\_intersect(cluster, track)$ **then**
                append $cluster$ to $valid\_cluster$
        if $valid\_cluster.empty()$ **then**
            append $track$ to $uncertain$
        else ▷ using nearest cosine distance
            $nearest \leftarrow get\_nearest(track, cluster\_init)$
            $2nd\_nearest \leftarrow get\_2nd\_nearest(track, cluster\_init)$
            if $err(nearest, 2nd\_nearest) < \tau$ **then**
                append $track$ to $uncertain$
            else
                merge $track$ to $cluster\_init[nearest]$
        return $clusters\_init, uncertain$

It follows that the distance matrix of $N \times N$ pairwise cluster appearance distances is:

$$S := \begin{bmatrix} \mathrm{dist}\left(\mathbf{C}_1, \mathbf{C}_1\right) & \cdots & \mathrm{dist}\left(\mathbf{C}_1, \mathbf{C}_N\right) \\ \vdots & \ddots & \vdots \\ \mathrm{dist}\left(\mathbf{C}_N, \mathbf{C}_1\right) & \cdots & \mathrm{dist}\left(\mathbf{C}_N, \mathbf{C}_N\right) \end{bmatrix} \tag{10}$$

Persons in the same camera cannot be in the same group after clustering. Therefore, we redefine their distance for pairs of people with the same camera. For each cluster C_i and C_j that come from the same camera, we redefine its distance on matrix $S_{i,j} = \infty$. This ensures that clusters from the same camera are not grouped together during clustering. Finally, we use agglomerative clustering on the updated distance matrix S to group these clusters.

4. Experiments

4.1. Dataset

The dataset used in this challenge comprises both real and synthetic data, totaling 2.607.781 frames across 22 different scenarios. Specifically, ten scenes are dedicated to training, with 4.375.736 bounding boxes and 71 unique person IDs. Similarly, ten scenes are reserved for validation, consisting of 1.950.917 bounding boxes and 35 unique person IDs. In terms of testing, there are two types of data to consider. The first one is Scene 001, which includes 388.671 frames of real data. The second type is synthetic data, which accounts for 648.360 frames.

4.2. Evaluation Metrics

The F1 score of people identity (IDF1) is used to assess how well multi-camera people tracking performs. IDF1 calculates the proportion of correctly identified detection in relation to the average number of ground-truth and computed detection. The AI City Challenge evaluation system will present IDF1, IDP, IDR, Precision (detection), and Recall (detection).

4.3. Implementation Details

We use Pytorch for our main framework. The experiments are performed on one Quadro RTX 6000 with 24GB.

Re-Identification In our ReID experiments, we tested both Transformer-based and CNN-based models. Specifically, we employed the TransReID base and TransReID with Jigsaw Patch Module (JPM) and Side Information Embeddings (SIE) from [13]. For our CNN-based models, we used ResNet50, HRNetW48, ResNet101, and ResNeXt101_ibn_a. Our optimization strategy used Stochastic Gradient Descent (SGD) with a base learning rate of 1e-4 and the Cosine Annealing scheduler. During training, we use a batch size of 96 and 4 IDs per batch.

Detection We fine-tune our model on the training data for one epoch. In the inference stage, we use the fine-tuned model for synthetic data. For the S001, which is real data, we only use the pre-trained model on the COCO dataset.

4.4. Parameter choosing

Our proposed system considered and optimized several factors to achieve high performance in single-camera and multi-camera matching. These included the selection of an appropriate number of clusters and the identification of a reliable zone. Additionally, we carefully chose the appropriate ReID model to extract features for accurate matching.

4.4.1 Reliable region selection

We pre-defined the regions that offer comprehensive information about the person's appearance. These regions should capture the complete body parts without any occlusions from static objects, thereby preventing situations where only a single body part is visible. Subsequently, we consider a tracklet reliable if it occurs more than θ times within these predefined regions. During the feature calculation step of the reliable tracklet, bounding boxes outside these regions are temporarily removed.

4.4.2 Number of Clusters Choosing

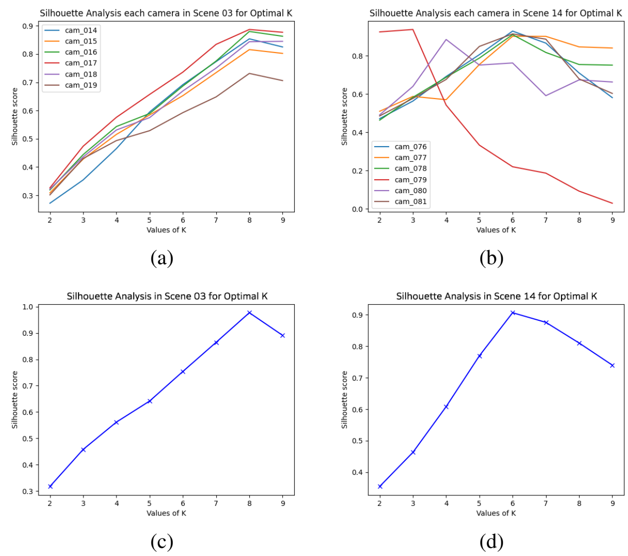


Figure 5. Silhouette score

For choosing the number of clusters in single-camera matching and multi-camera matching, we use the Silhouette Analysis to determine it. Silhouette analysis involves computing the silhouette coefficient for each data point in each cluster, which measures how similar that point is to other points in its cluster compared to points in neighboring clusters. The brute force approach is used to cluster the data into a range of cluster numbers to determine the optimal number of clusters. Then, the silhouette coefficient is computed for each clustering, and the number of clusters with the highest average silhouette coefficient is chosen as the optimal number of clusters. The figures 5a, and 5b show the average silhouette coefficient in each camera, and the figures 5c and 5d show silhouette coefficient score in multi-camera in one scene.

4.4.3 Choosing ReID model strategy

In order to choose the most effective ReID model strategy, we face a challenge in determining whether a model that performs well on synthetic validation data will generalize well to real data (Scene 001) in the test set. To address this, we propose a heuristic rule for selecting the best model based on performance on S001. Specifically, we use each ReID model to extract feature vectors and perform single-camera matching to create clusters, which are then used for multi-camera matching to obtain the final results. It is important to note that clusters from the same camera cannot be in the same group in the final results. Violation of this indicates that the model did not provide a sufficiently discriminative feature for single-camera matching, leading to the merging of different people's tracklets into one group, and consequently negatively impacting the feature in the multi-camera matching step. Any ReID model that fails to satisfy this rule will not be used during inference.

4.5. Experiments Results

4.5.1 Re-identification

To evaluate how well ReID features perform, we only use the person id from Scene 017 and split it into a query set and a gallery set. We then determined the mean Average Precision (mAP) as the evaluation metric. The results in the table 1 demonstrate that TransReID models have taken the top two ranks for extracting features from the synthetic dataset. HRNetW48 has demonstrated the lowest mAP among all tested models. However, after applying the validation method outlined in section 4.4.2, we discovered that only TransReID models and HRNetW48 performed well on real data (S001). Consequently, we only utilized TransReID models and HRNetW48 for feature extraction and ensembling in S001. We determined that the TransReID base produced satisfactory accuracy for synthetic data and was adequate to perform optimally on the data.

Model	mAP
TransReID + JPM + SIE	95.92
TransReID	95.76
ResNet101	94.43
ResNext101_ibn_a	94.34
ResNet50	94.00
HRNetW48	92.08

Table 1. The ablation study for ReID feature extraction

4.5.2 Ablation Study

The performance of individual components in our proposed system was evaluated through an ablation study. Table 2 shows the results. Compared with the baseline, the ID Switch helps us increase 0.5%. The proposed R-matching method improved the IDF1 score by 8.46%. Finally, the ensemble method, which combines features from TransReID, HRNetw48, and TransReID with JPM and SIE, resulted in the highest IDF1 score of 0.9417.

Method	IDF1	IDP	IDR
Baseline	0.8253	0.8568	0.7961
+ ID Switch	0.8322	0.8633	0.8032
+ R-matching	0.9168	0.9192	0.9145
+ Ensemble	0.9417	0.9393	0.9441

Table 2. The ablation study of combination of components

4.5.3 Comparison with other teams

Table 3 the evaluation of our proposed system in Track 1 of AI City Challenge 2023. Our system obtained an IDF1 score of 94.17%, which secured the second position among more than 25 teams worldwide.

5. Conclusions

In this paper, a solution for Multi-Camera People Tracking in indoor scenarios is proposed for Track 1 of the AI City Challenge 2023. The proposed framework has four modules, and the introduction of ID switch detection and id switch splitting efficiently addresses the problem of tracklets with ID switches. The system performs well in matching both synthetic and real data, with the r-matching algorithm performing exceptionally well in real scenarios despite being trained on synthetic data. Experimental results on the public test set of 2023 AI City Challenge Track 1 demonstrate the efficacy of the proposed approach, achieving an IDF1 of 94.17% and securing 2nd position on the leaderboard.

Rank	Team ID	Team Name	IDF1
1	6	UWIPL_ETRI	0.9536
2	**9**	**HCMIU-CVIP (ours)**	**0.9417**
3	41	AILab	0.9331
4	51	hust432	0.9207
5	113	FraunhoferIOSB	0.9284

Table 3. Final results on Track 1 test set.

6. Acknowledgement

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number DS2022-28-04. We would like to express our heartfelt appreciation to Ho Chi Minh City International University—Vietnam National University (HCMIU-VNU) for facilitating our efforts. Additionally, we would like to express our heartfelt appreciation to all of our colleagues for their contributions, which considerably aided in the revision of the manuscript.

References

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

[2] William Brendel, Mohamed R. Amer, and Sinisa Todorovic. Multiobject tracking as maximum weight independent set. *CVPR 2011*, pages 1273–1280, 2011. 2

[3] 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 (CVPR)*, June 2018. 2

[4] Wei-Ting Chen, I-Hsiang Chen, Chih-Yuan Yeh, Hao-Hsiang Yang, Jian-Jiun Ding, and Sy-Yen Kuo. Sjdl-vehicle: Semi-supervised joint defogging learning for foggy vehicle re-identification. In *AAAI Conference on Artificial Intelligence*, 2022. 2

[5] Wongun Choi and Silvio Savarese. Multiple target tracking in world coordinate with single, minimally calibrated camera. In *European Conference on Computer Vision*, 2010. 2

[6] Nhat Minh Chung, Huy Dinh-Anh Le, Vuong Ai Nguyen, Quang Qui-Vinh Nguyen, Thong Duy Nguyen, Tin Trung Thai, and Synh Viet-Uyen Ha. Multi-camera multi-vehicle tracking with domain generalization and contextual constraints. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3326–3336, 2022. 1, 2

[7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 1:886–893 vol. 1, 2005. 2

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020. 2

[9] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[10] Glenn Jocher et. al. ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support, Oct. 2021. 2

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):386–397, 2020. 2

[12] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015. 2, 4

[13] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. *CoRR*, abs/2102.04378, 2021. 2, 7

[14] Yuhang He, Jie Han, Wentao Yu, Xiaopeng Hong, Xing Wei, and Yihong Gong. City-scale multi-camera vehicle tracking by semantic attribute parsing and cross-camera tracklet matching. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2456–2465, 2020. 2

[15] Zhiqun He, Yu Lei, Shuai Bai, and Wei Wu. Multi-camera vehicle tracking with powerful visual features and spatial-temporal cue. In *CVPR Workshops*, 2019. 1

[16] João F. Henriques, Rui Caseiro, and Jorge P. Batista. Globally optimal solution to multi-object tracking with merged measurements. *2011 International Conference on Computer Vision*, pages 2470–2477, 2011. 2

[17] Yunzhong Hou, Heming Du, and Liang Zheng. A locality aware city-scale multi-camera vehicle tracking system. In *CVPR Workshops*, 2019. 1

[18] Yunzhong Hou, Liang Zheng, Zhongdao Wang, and Shengjin Wang. Locality aware appearance metric for multi-target multi-camera tracking. *ArXiv*, abs/1911.12037, 2019. 2

[19] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:2011–2023, 2017. 2

[20] Weiming Hu, Xi Li, Wenhan Luo, Xiaoqin Zhang, Stephen J. Maybank, and Zhongfei Zhang. Single and multiple object tracking using log-euclidean riemannian subspace and block-division appearance model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:2420–2440, 2012. 2

[21] Omar Javed, Khurram Shafique, and Mubarak Shah. Appearance modeling for tracking in multiple non-overlapping cameras. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2:26–33 vol. 2, 2005. 2

[22] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Forward-backward error: Automatic detection of tracking failures. *2010 20th International Conference on Pattern Recognition*, pages 2756–2759, 2010. 2

[23] Cheng-Hao Kuo, Chang Huang, and Ramakant Nevatia. Multi-target tracking by on-line learned discriminative appearance models. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 685–692, 2010. 2

[24] Chuyin Li, Lu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, L. Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, Yiduo Li, Bo Zhang, Yufei Liang, Linyuan Zhou, Xiaoming Xu, Xiangxiang Chu, Xiaoming Wei, and Xiaolin Wei. Yolov6: A single-stage object detection framework for industrial applications. *ArXiv*, abs/2209.02976, 2022. 2

[25] Fei Li, Zhen Hai Wang, Ding Nie, Shiyi Zhang, Xingqun Jiang, Xingxing Zhao, and Peng Hu. Multi-camera vehicle tracking system for ai city challenge 2022. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3264–3272, 2022. 2

[26] Hongli Li, Yongsheng Dong, and Xuelong Li. Online association by continuous-discrete appearance similarity measurement for multi-object tracking. *Neurocomputing*, 487:86–98, 2022. 2

[27] Hongchao Li, Xianmin Lin, Aihua Zheng, Chenglong Li, Bin Luo, Ran He, and Amir Hussain. Attributes guided feature learning for vehicle re-identification. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6:1211–1221, 2019. 2

[28] Tony Lindeberg. Scale invariant feature transform. *Scholarpedia*, 7:10491, 2012. 2

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

[30] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Ling yu Duan. Embedding adversarial learning for vehicle re-identification. *IEEE Transactions on Image Processing*, 28:3794–3807, 2019. 2

[31] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and A strong baseline for deep person re-identification. *CoRR*, abs/1903.07071, 2019. 4, 6

[32] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Yue Yao, Liang Zheng, Pranamesh Chakraborty, Christian E. Lopez, Anuj Sharma, Qi Feng, Vitaly Ablavsky, and Stan Sclaroff. The 5th ai city challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4263–4273, June 2021. 2

[33] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Xiaodong Yang, Liang Zheng, Anuj Sharma, Rama Chellappa, and Pranamesh Chakraborty. The 4th ai city challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 2

[34] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Sanjita Prajapati, Alice Li, Shangru Li, Krishna Kunadharaju, Shenxin Jiang, and Rama Chellappa. The 7th AI City Challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2023. 1

[35] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:971–987, 2002. 2

[36] Zhen Qin and Christian R. Shelton. Improving multi-target tracking via social grouping. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1972–1978, 2012. 2

[37] Cuong Le Quoc and Moncef Hidane. Appearance features for online multiple camera multiple target tracking. *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 2020. 2

[38] Jamie Sherrah, Dmitri Kamenetsky, and Tony Scoleri. Evaluation of similarity measures for appearance-based multi-camera matching. *2011 Fifth ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–6, 2011. 2

[39] Francesco Solera, Simone Calderara, and Rita Cucchiara. Learning to divide and conquer for online multi-target tracking. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4373–4381, 2015. 2

[40] Bi Song, Ting-Yueh Jeng, Elliot Staudt, and Amit K. Roy-Chowdhury. A stochastic graph evolution framework for robust multi-target tracking. In *European Conference on Computer Vision*, 2010. 2

[41] Daisuke Sugimura, Kris Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. Using individuality to track individuals: Clustering individual trajectories in crowds using local appearance and frequency trait. *2009 IEEE 12th International Conference on Computer Vision*, pages 1467–1474, 2009. 2

[42] Mingxing Tan, Ruoming Pang, and Quoc V. Le. Efficientdet: Scalable and efficient object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10778–10787, 2020. 2

[43] Duong Nguyen-Ngoc Tran, Long Hoang Pham, Hyung-Joon Jeon, Huy-Hung Nguyen, Hyung-Min Jeon, Tai Huu-Phuong Tran, and Jae Wook Jeon. A robust traffic-aware city-scale multi-camera vehicle tracking of vehicles. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3149–3158, 2022. 1

[44] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *ArXiv*, abs/2207.02696, 2022. 2

[45] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *CoRR*, abs/1908.07919, 2019. 4

[46] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017. 2

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

[48] Yu Xiang, Alexandre Alahi, and Silvio Savarese. Learning to track: Online multi-object tracking by decision making. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4705–4713, 2015. 2

[49] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5987–5995, 2016. 2, 4

[50] Bo Yang, Chang Huang, and Ramakant Nevatia. Learning affinities and dependencies for multi-target tracking using a crf model. *CVPR 2011*, pages 1233–1240, 2011. 2

[51] Bo Yang and Ramakant Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust ap-

pearance models. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1918–1925, 2012. 2
[52] Bo Yang and Ramakant Nevatia. An online learned crf model for multi-target tracking. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2034–2041, 2012. 2
[53] Xipeng Yang, Jinxing Ye, Jincheng Lu, Chenting Gong, Minyue Jiang, Xiangru Lin, Wei Zhang, Xiao Tan, Yingying Li, Xiaoqing Ye, and Errui Ding. Box-grained reranking matching for multi-camera multi-target tracking. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3095–3105, 2022. 2, 3
[54] Hui Yao, Zhizhao Duan, Zhen Xie, Jingbo Chen, Xi Wu, Duo Xu, and Yutao Gao. City-scale multi-camera vehicle tracking based on space-time-appearance features. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3309–3317, 2022. 2
[55] Jin Ye, Xipeng Yang, Shuai Kang, Yue He, Weiming Zhang, Leping Huang, Minyue Jiang, Wei Zhang, Yifeng Shi, Meng Xia, and Xiao Tan. A robust mtmc tracking system for ai-city challenge 2021. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4044–4053, June 2021. 2
[56] Ju Hong Yoon, Chang-Ryeol Lee, Ming-Hsuan Yang, and Kuk jin Yoon. Online multi-object tracking via structural constraint event aggregation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1392–1400, 2016. 2
[57] Sisi You, Hantao Yao, and Changsheng Xu. Multi-target multi-camera tracking with optical-based pose association. *IEEE Transactions on Circuits and Systems for Video Technology*, 31:3105–3117, 2021. 2
[58] Jianming Zhang, Liliana Lo Presti, and Stan Sclaroff. Online multi-person tracking by tracker hierarchy. *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, pages 379–385, 2012. 2
[59] Lu Zhang and Laurens van der Maaten. Preserving structure in model-free tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:756–769, 2014. 2
[60] Chuyang Zhao, Haobo Chen, Wenyuan Zhang, Junru Chen, Sipeng Zhang, Yadong Li, and Boxun Li. Symmetric network with spatial relationship modeling for natural language-based vehicle retrieval. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3225–3232, 2022. 2
[61] Yi Zhou and Ling Shao. Cross-view gan based vehicle generation for re-identification. In *British Machine Vision Conference*, 2017. 2
[62] Yi Zhou and Ling Shao. Viewpoint-aware attentive multi-view inference for vehicle re-identification. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6489–6498, 2018. 2