Multi-view Crowd Tracking Transformer with View-Ground Interactions Under Large Real-World Scenes

Qi Zhang[1] Jixuan Chen[1] Kaiyi Zhang[1] Xinquan Yu[1]
Antoni B. Chan[2] Hui Huang[1*]
[1]College of Computer Science and Software Engineering, Shenzhen University, China
[2]Department of Computer Science, City University of Hong Kong, China
{qi.zhang.opt, chen.jixuanstu, zhangky1999, xinquanyu2619}@gmail.com,
abchan@cityu.edu.hk, hhzhiyan@gmail.com

Abstract

Multi-view crowd tracking estimates each person's tracking trajectories on the ground of the scene. Recent research works mainly rely on CNNs-based multi-view crowd tracking architectures, and most of them are evaluated and compared on relatively small datasets, such as Wildtrack and MultiviewX. Since these two datasets are collected in small scenes and only contain tens of frames in the evaluation stage, it is difficult for the current methods to be applied to real-world applications where scene size and occlusion are more complicated. In this paper, we propose a Transformer-based multi-view crowd tracking model, MVTrackTrans, which adopts interactions between camera views and the ground plane for enhanced multi-view tracking performance. Besides, for better evaluation, we collect and label two large real-world multi-view tracking datasets, MVCrowdTrack and CityTrack, which contain a much larger scene size over a longer time period. Compared with existing methods on the two large and new datasets, the proposed MVTrackTrans model achieves better performance, demonstrating the advantages of the model design in dealing with large scenes. We believe the proposed datasets and model will push the frontiers of the task to more practical scenarios, and the datasets and code are available at: https://github.com/zqyq/MVTrackTrans.

1. Introduction

Multi-view crowd tracking estimates each person's tracking trajectories for a time period on the ground of the scene by accumulating the information from multiple synchronized and calibrated cameras. It can be applied to many applications, such as crowd management [23], public transportation [43], autonomous driving [18], *etc*.

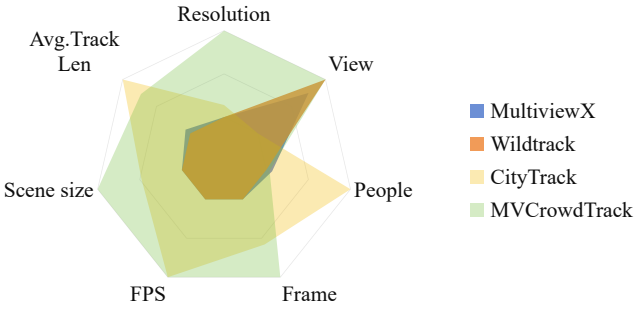

Figure 1. Comparison of the proposed MVCrowdTrack and CityTrack datasets with existing multi-view tracking datasets MultiviewX and Wildtrack. Our proposed datasets are superior in various aspects, featuring larger scene coverage, longer average track lengths, and a greater number of people, providing a more comprehensive benchmark for multi-view crowd tracking.

Most of the recent works [29, 30] evaluate their methods and compare with others on relatively small datasets, such as Wildtrack [4] and MultiviewX [13], which are recorded on small scenes and only consist of hundreds of frames in total. Thus, the existing works may not be well applied to real-world scenarios, since real-world multi-view crowd tracking may happen on large scenes with a large crowd and severe occlusions, and over a long time period. Therefore, to study the multi-view crowd tracking task under more difficult real scenes is in demand in the area.

In this paper, we first address the dataset and evaluation issues by collecting and labeling two large real-world multi-view crowd tracking datasets: MVCrowdTrack and CityTrack. MVCrowdTrack is a newly collected dataset captured in a campus environment, containing 4,122 frames from seven synchronized camera views. CityTrack is constructed based on the existing CityStreet dataset [41], which originally contains 500 multi-view frames sampled at 2 fps. We re-sampled the videos at 4 fps and re-annotated them for

* Corresponding author

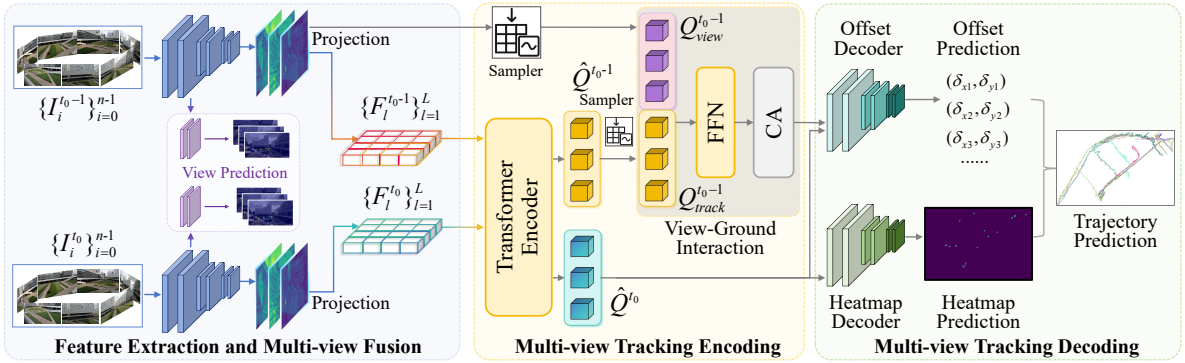

Figure 2. The overall pipeline of MVTrackTrans consists of Feature Extraction and Multi-view Fusion, Multi-view Tracking Encoding, and Multi-view Tracking Decoding. Consecutive multi-view frames are processed by a shared feature extractor and projected onto the ground plane to obtain fused ground features. These features are then encoded by a Transformer Encoder to generate track queries from the previous timestep and current frame queries from the current one. A View-Ground Interaction module is proposed to refine track queries by combining feature queries from both camera views and the ground plane. Finally, the Offset Decoder predicts temporal offsets, while the ground Heatmap Decoder outputs the current location detection results, which are combined together to obtain the predicted trajectories.

multi-view tracking, resulting in 2,588 frames with denser temporal continuity. As shown in Figure 1, compared to datasets used in the state-of-the-art methods [29, 30], *e.g.*, Wildtrack and MultiviewX, the proposed datasets contain larger scenes, more frames with longer video duration, and more number of trajectories (CityTrack) with longer average trajectory length. Overall, the proposed datasets are more suitable for studying the multi-view crowd tracking task under difficult real-world scenes.

Besides, it is noticed that most SOTAs adopt CNNs-based architecture for the multi-view crowd tracking task, and seldom explore more advanced architectures such as transformers. In contrast, we propose a Transformer-based multi-view crowd tracking model, denoted MVTrackTrans, which performs the multi-view crowd tracking on the ground with transformers. Furthermore, MVTrackTrans adopts interactions between camera views and the ground plane for better multi-view tracking. As illustrated in Figure 2, the whole pipeline of the proposed MVTrackTrans model consists of three steps: (1) **Feature extraction and Multi-view Fusion**: Each camera view's features of the two neighboring frames are extracted with a shared feature extractor and then projected to the ground plane for multi-view fusion to obtain the ground representation. (2) **Multi-view Tracking Encoding**: To capture temporal dynamics, the ground representations from the previous and current frames are encoded by a deformable encoder. From the previous-frame ground representation, a set of discrete track queries is sampled to represent the tracked entities in the ground space. A *view-ground interaction* module is proposed, where these queries interact with the previous-frame multi-view detection features via cross-attention, enabling cross-view and temporal information exchange. (3) **Multi-view Tracking Decoding**: The refined queries are then fused with the current frame's ground representation to jointly decode motion offsets and detection heatmaps of the crowd, from which the final tracking results are obtained.

As far as we know, this is the first time that multi-view crowd tracking is put forward to much larger real-world scenes with more crowds and longer time periods, and a transformer-based multi-view crowd tracking model is presented. We believe the proposed datasets and MVTrackFormer model will push the frontiers of the area to more practical scenarios. Our main contributions are as follows.

- We propose two large real-world datasets, MVCrowdTrack and CityTrack, for multi-view crowd tracking. Compared to existing datasets widely used in the area, the proposed two datasets are much more difficult and more suitable for real-world research, which shall advance the area to more practical applications.
- We propose a transformer-based multi-view crowd tracking model, MVTrackTrans, which uses transformer architecture for multi-view crowd tracking instead of CNNs models to deal with the demand for more complicated spatial and temporal association among the crowds in the scene. Besides, MVTrackTrans adopts extra view-ground interactions for better multi-view tracking performance.
- The experiments validate that the proposed MVTrackTrans model achieves better multi-view crowd tracking performance than comparison methods on the two large real-world datasets, which demonstrates its superiority for the task under complicated scenarios.

2. Related Work

We first review existing multi-view crowd-tracking methods and transformer-based single-view multi-object tracking methods. Then, we also review the related datasets for multi-view crowd-tracking.

Multi-view Crowd Tracking. Multi-view crowd tracking

aims to track individuals across multiple cameras. A number of works have been proposed to achieve robust and temporally consistent tracking in multi-view settings. To maintain motion continuity across time, MVFlow [9] models ground-plane motion flow for occlusion-robust tracking. However, it restricts each person's movement to a single discrete grid per timestep, which limits its ability to model continuous and realistic human motion dynamics. To learn a more discriminative representation for each individual, EarlyBird [30] builds upon standard BEV-based multi-view detection architectures [13, 42], integrates an additional ReID module to enhance identity distinction, and leverages a Kalman Filter [15] for temporal association across frames. TrackTacular [29] improves on EarlyBird by lifting the BEV representation into 3D space and stacking world representations from adjacent frames to extract temporal information for regressing motion offsets of people. Different from previous methods that rely solely on view features or BEV features, DepthTrack [31] leverages clustered point clouds projected from depth maps to assist BEV-based tracking. By incorporating richer geometric cues such as shape and orientation, the point clouds enhance the model's ability to distinguish and re-identify individuals more accurately. Aiming to improve tracking performance in long multi-view videos, MCBLT [35] employs a Hierarchical Graph Neural Network to perform people association in the BEV space across multiple temporal scales. Each GNN layer focuses on short- and mid-term dependencies, significantly enhancing robustness under long-term occlusions. MVTrajecter [37] proposed a framework that utilizes motion and appearance costs across multiple past timestamps, instead of relying solely on the nearest one, to improve trajectory association. By considering information from several previous frames, it achieves more robust tracking performance.

However, most existing approaches rely on CNN-based architectures and rarely explore transformer-based models for multi-view crowd tracking. To address this, MCTR [24] proposed a transformer-based framework that iteratively updates global tracking embeddings by interacting with detections from all camera views and introduced a probabilistic association strategy to ensure consistent matching across views and time. *In contrast, while MCTR performs tracking in the original camera views, our MVTrackTrans operates directly in the BEV space and incorporates a View-Ground Interaction module to further enhance the performance.*

Transformer-based Single-view MOT. Multi-object tracking (MOT) [3, 5, 10, 17, 20, 21, 27, 45] typically uses monocular cameras to follow moving targets. Recent transformer-based approaches have shown substantial gains in detection accuracy, temporal association, and robustness under challenging conditions. For instance, GTR [46] proposed a global trajectory query mechanism to associate detections across the entire video sequence. MeMOT [2] developed a large spatio-temporal memory module to store and update object embeddings along trajectories, thereby achieving more robust and consistent tracking performance. P3AFormer [44] proposed detecting and tracking objects at the pixel level, which is particularly beneficial for handling small objects. Similarly, TransCenter [36] also leverages dense feature representations for tracking and introduces carefully designed detection and tracking queries to reduce the computational cost of attention over such detailed maps. Modeling object motion is crucial in MOT. STDFormer [14] introduced a purely motion-driven transformer tracking framework that jointly models linear and exponential motion patterns, enabling effective handling of both simple and complex object dynamics. To address low tracking performance caused by missed or incorrect detections, BUSCA [32] proposed a method to recover lost detections under occlusions by generating potential proposals and formulating the object-proposal association as a multi-choice question-answering task using a decision transformer. TGFormer [40] addresses occlusions from a different perspective by introducing a group of queries for each object, enabling better capture of appearance variations under varying levels of occlusion. LA-MOTR [34] proposed a novel learnable association module to iteratively refine the association across multiple attention layers by incorporating relative spatial cues and enabling bidirectional feature interaction between detections and tracklets. *Although there are many transformer-based methods for single-view multi-object tracking, transformer architectures for* multi-view *crowd tracking remain largely unexplored. Our MVTrackTrans aims to fill this gap.*

Multi-view Crowd Tracking Datasets. Existing multi-view crowd tracking datasets primarily focus on small- to medium-scale areas with relatively short sequences. Table 1 summarizes the key statistics of representative datasets, including resolution, number of views, people, frames, frame rate, and scene size. Wildtrack [4, 5] is a real-world dataset captured by seven synchronized and calibrated cameras covering an area of 36×12 m. The camera resolution is 1920×1080 pixels at 2 fps, and the total sequence length is 400 frames. MultiviewX [13] is a synthetic dataset designed as a virtual replica of Wildtrack. It contains six virtual cameras covering an area of 25×16 m. The camera resolution, frame rate (2 fps), and sequence length (400 frames) are consistent with Wildtrack. GMVD [33] is also a synthetic multi-view dataset with multiple scenes. Nevertheless, both the scene size and the crowd density are relatively small, remaining largely similar to MultiViewX. While Wildtrack and MultiviewX are widely adopted in prior multi-view detection [8, 12, 13, 26, 28, 42] and tracking [29, 30, 37] studies, their scene coverage is limited, and the video sequences are relatively short.

To address these limitations, we introduce two newly collected large-scale real-world multi-view tracking datasets: MVCrowdTrack and CityTrack. As shown in Figure 1 and Table 1, MVCrowdTrack and CityTrack are collected in large real-world scenes and provide substantially larger spatial coverage, significantly longer sequences, and larger numbers of crowds than existing benchmarks, enabling more comprehensive evaluation of multi-view tracking methods for real-world applications.

3. Multi-view Crowd Tracking Transformer

The multi-view crowd tracking task aims to estimate all individuals' trajectories on the ground plane of a scene captured by multiple synchronized and calibrated cameras. In this section, we present our Multi-view Crowd Tracking Transformer framework (as shown in Figure 2), which unifies ground-plane reasoning and view-level temporal modeling for robust human localization and tracking. The overall architecture consists of three main stages:

(1) *Feature Extraction and Multi-view Fusion*: The multi-view images are first fed into a ResNet backbone to extract multi-level features from the first three stages. The extracted single-view features are then projected into the ground plane using the calibrated camera parameters, producing fused ground features after multi-view fusion.

(2) *Multi-view Tracking Encoding*: To incorporate temporal information, fused ground features from both the previous frame and the current frame are first processed by the deformable encoder to enhance their spatial representation. Discrete track queries are then sampled from specific positions within the encoded ground feature map of the previous frame, representing the tracked entities in the ground-plane space. Afterward, a View-Ground Interaction module is introduced, where the track queries and the multi-view queries are fused via a cross-attention mechanism.

(3) *Multi-view Tracking Decoding*: Subsequently, the refined track queries and the current-frame ground queries are jointly decoded to estimate the motion offsets of tracked entities in the current frame. Finally, the current-frame ground queries directly regress the crowd location heatmap. By combining the heatmap with the predicted offsets, the final tracking results are obtained. See stage details as follows.

3.1. Feature Extraction and Multi-view Fusion

We first extract multi-scale single-view features from each camera view using a ResNet backbone [11]. Let the input multi-view images at timestamp t_0 be $\{I_i^{t_0}\}_{i=0}^{n-1}$, where i denotes the camera view index and n is the number of cameras. The multi-scale single-view features are extracted as: $\{F_{view_i}^{t_0}\}_{l=1}^{L} = \{F(I_i^{t_0})\}_{l=1}^{L}, \quad i = 0, 1, \ldots, n-1$, where $l = 1 : L$ indexes the feature scale and L denotes the number of scales. Each camera then independently passes its multi-scale features through a feature pyramid network (FPN) to produce single-scale view features $\hat{F}_{\text{view}_i}^{t_0}$ for subsequent camera view detection and view feature sampling.

Instead of projecting features to a fixed-height plane, we adopt a multi-height bilinear sampling approach [1]:

$$\left(u_n, v_n, 1\right)^T = K[R|T]\left(x_n, y_n, z_n, 1\right)^T. \tag{1}$$

where each 3D voxel pulls information from the corresponding single-view features. For a voxel with coordinates (x_n, y_n, z_n), its eight vertices are projected to all image planes via the camera projection matrix $K[R|T]$.

The voxel features are sampled from the image features $\hat{F}_{\text{view}_i}^{t_0}$ of all camera views and aggregated across views. This ensures that each voxel receives informative features, making the method robust for long-range perception and crowded scenes. Finally, the voxel features are collapsed along the height axis and fused across views using convolution, resulting in the multi-scale *ground feature* $\{F_l^{t_0}\}_{l=1}^{L}$ of the current frame. In the following stages, the ground features of timestamps t_0 and t_0-1 are input into transformer networks for further processing.

3.2. Multi-view Tracking Encoding

We denote the multi-scale ground features at the previous and current timestamps as $\{F_l^{t_0-1}\}_{l=1}^{L}$ and $\{F_l^{t_0}\}_{l=1}^{L}$. Each set of features is independently processed by the same Transformer Encoder, which is implemented using a Multi-scale Deformable Attention Module, to aggregate information across multiple scales:

$$\hat{Q}^{t_0-1} = \text{TransformerEncoder}(\{F_l^{t_0-1}\}_{l=1}^{L}), \tag{2}$$

$$\hat{Q}^{t_0} = \text{TransformerEncoder}(\{F_l^{t_0}\}_{l=1}^{L}). \tag{3}$$

After obtaining the two-frame queries, we perform discrete sampling on the previous-frame query $\hat{Q}^{t_0-1}$ at the past detection locations (x, y) on the ground plane to construct the track queries: $Q_{\text{track}}^{t_0-1} = \text{SampleQueries}(\hat{Q}^{t_0-1}, (x, y))$.

View-Ground Interaction. As in Figure 3, to better represent each tracked person, we integrate complementary information from both the ground and camera views. For

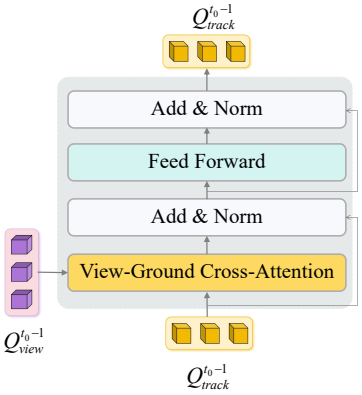

Figure 3. The view-ground interaction module details.

each camera, we sample a set of view-specific queries from the corresponding view detection features, and then concatenate them across all cameras to form the *view queries*:

$$Q_{\text{view}}^{t_0-1} = \text{Concat}\big(Q_{\text{view},0}^{t_0-1}, Q_{\text{view},1}^{t_0-1}, \ldots, Q_{\text{view},n-1}^{t_0-1}\big), \quad (4)$$

where $Q_{\text{view},i}^{t_0-1}$ denotes the queries sampled from the i-th camera view's feature map $\hat{F}_{\text{view}_i}^{t_0}$ in the previous frame. The track queries ($Q_{\text{track}}^{t_0-1}$) and view queries ($Q_{\text{view}}^{t_0-1}$) from the previous frame are first processed by independent feed-forward networks (FFN) to refine their embeddings. Then, we adopt a cross-attention mechanism where the track queries serve as the queries Q, and the view queries sampled from multi-cameras act as the keys K and values V:

$$Q_{\text{track}}^{t_0-1} = \text{CrossAttn}\big(\text{FFN}(Q_{\text{track}}^{t_0-1}),\ \text{FFN}(Q_{\text{view}}^{t_0-1})\big). \quad (5)$$

This design allows each track queries to aggregate visual features from all camera views corresponding to the same tracked individual, addressing the limitation that discrete sampling from the previous frame's ground representation may not fully capture the appearance of the person.

3.3. Multi-view Tracking Decoding

The decoding stage consists of two parallel branches: a heatmap decoder and an offset decoder.

Offset Decoder. The offset decoder adopts a standard Multi-scale Deformable Attention (MSDA) structure to model temporal correspondence between consecutive Ground frames. Specifically, given the previous-frame track queries $Q_{\text{track}}^{t_0-1}$, the corresponding detection locations (x^{t_0-1}, y^{t_0-1}), and the current Ground features $\hat{Q}^{t_0}$, the temporal interaction is formulated as:

$$\hat{Q}_{\text{track}}^{t_0} = \text{MSDA}\big(Q_{\text{track}}^{t_0-1},\ \hat{Q}^{t_0},\ (x^{t_0-1}, y^{t_0-1})\big), \quad (6)$$

where (x^{t_0-1}, y^{t_0-1}) serve as reference points to guide the deformable sampling. The refined track queries are then passed through a lightweight MLP head to predict the motion offsets on the ground plane:

$$O^{t_0} = \text{MLPHead}(\hat{Q}_{\text{track}}^{t_0}) = \big[\delta x, \delta y\big]^T. \quad (7)$$

Heatmap Decoder. The Heatmap Decoder integrates the multi-scale Ground features of the current frame to predict human centers. It first employs a Feature Pyramid Network (FPN) to upsample and fuse $\hat{Q}^{t_0}$ into the highest spatial resolution. The fused representation is then passed through a convolutional regression head to predict the final crowd heatmap on the ground: $H^{t_0} = \text{ConvHead}(\text{FPN}(\hat{Q}^{t_0}))$. Together, these two decoders jointly predict the crowd locations and their temporal displacements, enabling continuous multi-view tracking in dynamic environments.

3.4. Model Training and Loss

The proposed model is trained by jointly optimizing a heatmap classification loss for both ground and image domains, and a regression loss for the motion offset on the ground plane. Following the uncertainty weighting strategy, we introduce two learnable parameters to adaptively balance the center and tracking branches during training.

Heatmap Loss. To supervise the center prediction, we construct the ground-truth heatmap H^* by placing Gaussian responses at each object center. Given the predicted heatmap H, we apply the focal loss [19]:

$$\mathcal{L}_{\text{ground}} = \text{FocalLoss}(H, H^*). \quad (8)$$

An additional image-level supervision term $\mathcal{L}_{\text{img}}$ is introduced using the same formulation to predict human center heatmaps in views.

Offset Regression Loss. The offset decoder predicts the displacement of each tracked object center between consecutive frames. Given the predicted offset $O = [\delta x, \delta y]$ and its ground truth O^*, we employ an ℓ_1 loss:

$$\mathcal{L}_{\text{track}} = \frac{1}{K}\sum_{x,y} \|O_{xy} - O_{xy}^*\|_1, \quad \text{if } C_{xy}^* = 1. \quad (9)$$

This loss is only applied to valid center locations, ensuring sparse supervision over active tracks.

Total loss. To adaptively balance the losses of different branches, we follow the uncertainty weighting strategy [16]:

$$\mathcal{L}_{\text{all}} = 10e^{-\sigma_c}\mathcal{L}_{\text{ground}} + e^{-\sigma_t}\mathcal{L}_{\text{track}} + \mathcal{L}_{\text{img}} + \sigma_c + \sigma_t, \quad (10)$$

where σ_c and σ_t are learnable uncertainty parameters for the center and tracking branches, respectively. This formulation allows the network to automatically calibrate the relative contribution of each branch during training.

4. Experiments and Results

4.1. Datasets

MVCrowdTrack Dataset. To advance the multi-view crowd tracking task to more complicated conditions, we first collect and label a large real-world multi-view crowd tracking dataset, MVCrowdTrack, which is collected on a large campus with a size of 120 m $\times$ 80 m. The scene of MVCrowdTrack is covered with 7 synchronized cameras together, with an image resolution of 5312×2988 , and lasts for 18 minutes. The frame rate of videos is 60, and we label 4 frames per second, resulting in a total of 4122 multi-view frames. In total, the dataset contains 342 people's trajectories with an average track length of 176 frames. In the experiments, 80% of the data (3297 frames) are used for training, and the remaining 20% (825 frames) are used for testing. We label the people in each view with bounding boxes,

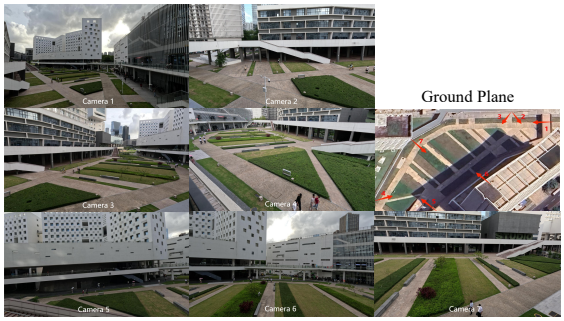

Figure 4. Multi-camera views and corresponding ground-plane layout in the MVCrowdTrack dataset.

Table 1. Comparison of multi-view pedestrian tracking datasets.

Dataset	Resolution	View	People	Frame	FPS	Size (m^2)	Avg. Track Len.
MultiviewX	1920×1080	6	360	400	2	25×16	44
Wildtrack	1920×1080	**7**	313	400	2	36×12	30
CityTrack	2704×1520	3	**950**	2588	**4**	64×76	**228**
MVCrowdTrack	**5312×2988**	**7**	342	**4122**	**4**	**120×80**	176

and make sure the same person is assigned a consistent ID across the time period. The foot points of each person are projected from all camera views onto the ground plane with camera calibrations (both extrinsic and intrinsic), and then the projected points from different views of the same person are averaged as the ground-truth location on the ground. The ground-plane map resolution is 1200×800, and 1 pixel is 0.1m in the real world. See an example in Figure 4.

CityTrack Dataset. We have also labeled an existing multi-view crowd dataset called CityStreet [41] for the multi-view tracking task. CityTrack contains 2,588 frames from the CityStreet dataset videos, annotated at 4 fps, and ensures that the IDs of the crowds are consistent for tracking tasks. It contains trajectories with an average track length of 228 frames. The first 1948 frames are used for training, and the rest are used for testing. The ground-plane resolution is 640×768, and 1 pixel represents 0.1 m in the real world.

Compared to **Wildtrack** and **MultiviewX** in Table 1, the newly proposed **MVCrowdTrack** and **CityTrack** datasets have larger scene scales, higher spatial resolutions, denser frame annotations, and significantly longer average trajectory lengths. Specifically, MVCrowdTrack covers a wider area with higher-resolution cameras for a longer period, while CityTrack also provides denser temporal annotations and many more crowds with consistent IDs. These characteristics make them more suitable for evaluating multi-view crowd tracking approaches in real-world and complex scenarios, where existing small-scale datasets may not fully reflect the challenges of practical applications.

4.2. Experiment Settings

Comparison methods. We have compared with the state-of-the-art multi-view crowd tracking methods, such as Earlybird [30], MVFlow [9], and TrackTacular [29] on the proposed MVCrowdTrack and CityTrack datasets. The code provided for each method is adopted from their papers, and they are trained and evaluated with the same settings as our model. We have tried to run ReST [6] on the new datasets, but it failed. For other methods like MVTrajecter [37], MVTr [38] or MCBLT [35], since no codes are provided, we cannot compare with them on the two new datasets. We also compare our model with other existing methods, REMP [7], MCBLT [35], MVTr [38] on MultiviewX and Wildtrack. Their metrics on MultiviewX and Wildtrack are adopted from their papers.

Implementation details. Our framework follows the ResNet18 [11] feature extractor and the Transformer encoder–decoder architecture presented in Deformable DETR [47]. During training, all input images are resized to 1280 × 720 pixels. We train for 50 epochs on all datasets, including MVCrowdTrack and CityTrack. The initial learning rate is 0.01. All experiments are conducted on 4 NVIDIA RTX 4090 GPUs with a batch size of 1.

Evaluation metrics. All tracking metrics are evaluated on the ground plane to ensure consistent spatial alignment. We adopt the same standard Multiple Object Tracking (MOT) metrics as the latest SOTAs [29, 30] along with identity-aware measures. The distance threshold for positive association is set to $r = 2\,\text{m}$ for the larger-scale MVCrowdTrack and CityTrack datasets, and $r = 1\,\text{m}$ on the Wildtrack and MultiviewX datasets. *The main evaluation metrics are Multiple Object Tracking Accuracy (MOTA) and IDF1*, which jointly consider missed detections, false positives, and identity switches. We further report Mostly Tracked (MT) and Mostly Lost (ML), representing the proportion of trajectories that are successfully tracked for more than 80% or less than 20% of their lifespan, respectively, relative to the total number of unique pedestrians in the test set.

4.3. Experiment Results

We show the multi-view tracking performance on MVCrowdTrack and CityTrack in Table 2. On **MVCrowdTrack**, our proposed method MVTrackTrans achieves the best performance across all methods. EarlyBird is a typical CNNs-based multi-view crowd tracking method supervised with similar heatmaps to ours. But it is much worse compared to our method on MVCrowdTrack, demonstrating the transformer architecture's advantages on large and complicated scenes. MVFlow achieves the worst performance, where the possible reason is that it uses a weakly-supervised human motions for tracking, resulting in much lower metrics for long-time tracking on MVCrowdTrack. TrackTacular uses better historical information compared to EarlyBird. It achieves the second-best IDF1, MOTA, MT, and ML metrics, but it is still worse than our method. Generally, our method is the best among all methods, proving that the transformer model's superiority on the large and long-time multi-view crowd tracking tasks.

Table 2. Comparison of the multi-view crowd tracking performance on the larger datasets MVCrowdTrack and CityTrack using 5 metrics. The proposed method ranks the best among all methods according to the metrics on the two datasets. We use MOTA and IDF1 as main metrics. **Bold** font indicates the best metric, and underline font indicates the second-best.

Dataset	MVCrowdTrack					CityTrack				
Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
EarlyBird [30]	54.56	30.46	53.84	24.48	14.22	48.85	21.83	32.15	17.33	13.9
MVFlow [9]	49.82	**46.79**	44.06	22.22	37.04	38.19	6.94	27.89	8.92	24.88
TrackTacular [29]	62.86	29.23	58.71	40.81	10.20	43.37	**23.23**	32.49	20.43	**12.38**
MVTrackTrans (Ours)	**63.87**	40.59	**59.06**	**42.85**	**8.16**	**55.39**	22.71	**34.41**	**25.07**	12.69

Table 3. Tracking performance on CityTrack for different variants of our proposed MVTrackTrans model.

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
Baseline	54.92	**22.83**	34.11	27.86	13.00
+ View Prediction Branch	53.17	20.68	32.65	**30.34**	**10.52**
++ ViewInteraction (Ours)	**55.39**	22.71	**34.41**	25.07	12.69

On **CityTrack**, the proposed method also achieves the best performance according to MOTA, IDF1, and MT metrics, and the second best MOTP and ML. Compared to MultiviewX and Wildtrack, CityTrack is a dataset with larger sizes, more crowds, and more severe occlusions. Our method achieves the best results, revealing that our method can handle these challenges better than existing methods. EarlyBird is the second according to MOTA, showing that it is quite a stable architecture across different datasets, but it is behind our transformer model, due to which our model can be better adapted to more complicated scenarios. Similarly, MVFlow performs the worst on CityTrack for similar reasons as on MVCrowdTrack. TrackTacular achieves the second IDF1, but much lower MOTA than ours, which suggests that TrackTacular cannot perform detection well on complicated datasets.

Overall, the proposed MVTrackTrans method achieves the best performance among all methods on the two large real-world datasets. The reason is that we adopt a transformer-based architecture for the large and complicated scenes, which provides stronger spatial (multi-view) and temporal fusion for the tracking task. In addition, the proposed view-ground interaction module further improves the tracking results (see the ablation study in Sec. 4.4). We also show the **visualization** results of predicted trajectories on the MVCrowdTrack and CityTrack datasets in Figure 5. It concludes that our method can accurately track more people compared to other methods as seen in the red boxes. Especially, as shown in the red box of the results on MVCrowdTrack, after a long time period of tracking, our method could still track more people compared to EarlyBird and TrackTacular.

4.4. Ablation Study

Architecture ablation study. We have conducted an ablation study on the model architecture on the CityTrack dataset: 'Baseline', '+View Prediction Branch', '++ViewInteractions (Ours)', in Table 3. 'Baseline' means no 2D camera view heatmap prediction branch or supervision is used in the model training; '+View Prediction Branch' means adding the 2D heatmap prediction branch to the Baseline model, but without the proposed view-ground interaction module; '++ViewInteractions (Ours)' means further adding our proposed view-ground interaction module to the model. From Table 3, we conclude that simply adding the 2D camera-view branch in the model does not improve the model performance due to the competition in the 2D and ground-plane task training. And with our view-ground interaction module further, the model can achieve better tracking performance, especially according to MOTA and IDF1 metrics. The reason is that the proposed view-ground interaction module fuses both the camera view and ground information, which helps to achieve more stable tracking for a long time.

Table 4. Ablation study on the view-ground interaction module, which is conducted on the CityTrack dataset.

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
SelfAtt	55.38	**23.71**	33.64	**27.86**	**12.38**
CrossAtt (Ours)	**55.39**	22.71	**34.41**	25.07	12.69

Table 5. Ablation study on the supervision manner: Use sparse queries with direct coordinate regression loss, or use dense pixel representations with heatmap prediction loss (ours). The experiment is conducted on the CityTrack dataset

Training	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
Coordinate regression	40.71	13.88	31.45	9.28	20.12
Heatmap regression (Ours)	**55.39**	**22.71**	**34.41**	**25.07**	**12.69**

View-ground interaction ablation study. We conduct an ablation study on the view-ground interaction module in Table 4. We compare different ways of implementing the module: 'SelfAtt', and 'CrossAtt (Ours)'. 'SelfAtt' means the camera view queries are fused with the ground queries with a self attention mechanism; 'CrossAtt (Ours)' the camera view queries are fused with the ground queries with a cross attention mechanism. As in Table 4, the best performance is achieved by using cross attention, in terms of MOTA and IDF1. The reason is that the cross attention provides a more thorough fusion of the camera view features and ground features, compared to self attention, resulting in enhanced performance.

Training method ablation study. We also conduct an ablation study on the model training in Table 5. We compare

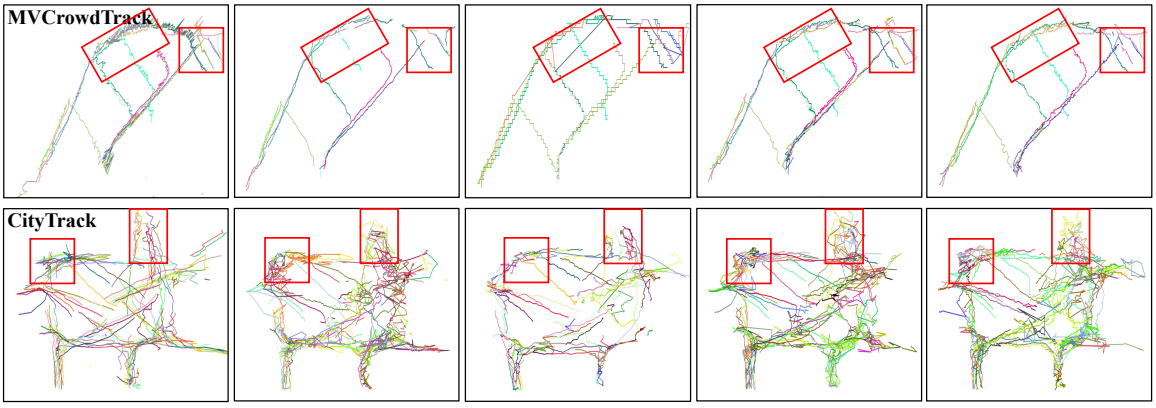

Figure 5. The predicted trajectory visualizations. Our method can accurately track more people for a long time (see red boxes).

Table 6. Comparison with previous methods on Wildtrack. For MCBLT, we report the results using its original detector.

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
KSP-DO [4]	69.6	61.5	73.2	28.7	25.1
KSP-DO-ptrack [4]	72.2	60.3	78.4	42.1	14.6
GLMB-YOLOv3 [25]	69.7	73.2	74.3	79.5	21.6
GLMB-DO [25]	70.1	63.1	72.5	93.6	22.8
DMCT [39]	72.8	79.1	77.8	61.0	4.9
DMCT Stack [39]	74.6	78.9	81.9	65.9	4.9
ReST† [6]	81.6	81.8	85.7	79.4	4.9
MCBLT[35]	87.5	**94.3**	93.4	90.2	**2.4**
REMP† [7]	88.5	86.8	–	–	–
EarlyBird [30]	89.5	86.6	92.3	78.0	4.9
MVFlow [9]	91.3	–	93.5	–	–
TrackTacular [29]	91.8	85.4	95.3	87.8	4.9
MVTr [38]	92.3	92.7	93.1	**95.1**	4.9
MVTrajecter [37]	**94.3**	93.0	96.5	90.2	4.9
MVTrackTrans (Ours)	91.2	86.9	94.1	82.9	4.9
MVTrackTrans (Ours)†	93.6	86.7	**96.7**	85.4	4.9

Table 7. Comparison with previous methods on MultiviewX.

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓
REMP† [7]	81.0	85.8	–	–	–
EarlyBird [30]	88.4	86.2	82.4	82.9	1.3
TrackTacular [29]	92.4	80.1	85.6	92.1	2.6
MVTr [38]	91.4	**95.0**	82.9	96.1	**0.0**
MVTrajecter [37]	**92.8**	**95.0**	85.8	**97.4**	**0.0**
MVTrackTrans (Ours)	89.8	90.2	72.1	85.5	6.6
MVTrackTrans (Ours)†	90.2	83.6	**86.3**	88.2	3.9

different ways of training our proposed transformer-based multi-view crowd tracking model: coordinate regression and heatmap regression. The first use spare queries and direct coordinate ground-truth as supervision, as in 2D transformer tracking method [22]; The second one uses dense pixel representations with heatmap prediction loss (Ours) for supervision, as in [36]. As in Table 5, the model trained with heatmap regression is much better than using the direct coordinate regression. The reason might be that in the multi-view crowd tracking task, the multi-view projection step stretches the features on the ground, which causes extra difficulties for accurate tracking. And the dense heatmap supervision can better guide the model training to reject these noises, and thus better performance is achieved.

Performance on small datasets Wildtrack and MultiviewX. We have also conducted experiments on small datasets, Wildtrack and MultiviewX, as shown in Table 6 and 7, respectively. The proposed model outperforms several methods on Wildtrack, such as ReST, MCBLT, EarlyBird, and REMP. While it achieves comparable performance on Wildtrack as SOTAs TrackTacular, MVTr, or MVFlow, though lower than MVTrajecter. It also achieves comparable performance on MultiviewX as EarlyBird, but lower performance than MVTrajecter, TrackTacular and MVTr. Additionally, MVTrackTrans† denotes a variant of MVTrackTrans with two-frame temporal feature fusion and Kalman filter to improve temporal consistency, achieving performance comparable to existing SOTAs. Overall, MVTrackTrans also achieves good results on small datasets.

5. Conclusion

In this paper, we aim to advance the current study for the multi-view crowd tracking task to more challenging and practical scenarios. Thus, first, we propose to collect a large multi-view tracking dataset that contains a much larger scene size with a long time period, and label an existing large real-world multi-view crowd dataset for the task. Besides, instead of exploring pure CNN-based models as previous research in the area, we propose a transformer-based multi-view crowd tracking model, *MVTrackTrans*, which adopts interactions between camera views and the ground plane for better multi-view tracking. Compared with existing methods on the two large real-world datasets, the proposed MVTrackTrans model achieves much better performance. We believe the proposed datasets and model will advance the multi-camera-based multi-object tracking task to more practical scenarios.

Acknowledgments

This work was supported in part by Guangdong Science and Technology Program (2024B0101050004), NSFC (62202312), ICFCRT (W2441020), Shenzhen Science and Technology Program (KJZD20240903100022028, KQTD20210811090044003), Scientific Foundation for Youth Scholars and Scientific Development Funds from Shenzhen University.

References

[1] AdamW.Harley, ZhaoyuanFang, JieLi, RaresAmbrus, and KaterinaFragkiadaki. Simple-bev: What really matters for multi-sensor bev perception? In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 4

[2] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8090–8100, 2022. 3

[3] Tzoulio Chamiti, Leandro Di Bella, Adrian Munteanu, and Nikos Deligiannis. Refergpt: Towards zero-shot referring multi-object tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3849–3858, 2025. 3

[4] Tatjana Chavdarova, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Timur Bagautdinov, Louis Lettry, Pascal Fua, Luc Van Gool, and François Fleuret. Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5030–5039, 2018. 1, 3, 8

[5] Sijia Chen, En Yu, and Wenbing Tao. Cross-view referring multi-object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2204–2211, 2025. 3

[6] Cheng-Che Cheng, Min-Xuan Qiu, Chen-Kuo Chiang, and Shang-Hong Lai. Rest: A reconfigurable spatial-temporal graph model for multi-camera multi-object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10051–10060, 2023. 6, 8

[7] Kosta Dakic, Kanchana Thilakarathna, Rodrigo N. Calheiros, and Teng Joon Lim. Resource-efficient multiview perception: Integrating semantic masking with masked autoencoders. In *2025 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 145–151, 2025. 6, 8

[8] Amir Etefaghi Daryani, M. Usman Maqbool Bhutta, Byron Hernandez, and Henry Medeiros. Camuvid: Calibration-free multi-view detection. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1220–1229, 2025. 3

[9] Martin Engilberge, Weizhe Liu, and Pascal Fua. Multi-view tracking using weakly supervised human motion prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1582–1592, 2023. 3, 6, 7, 8

[10] Ruopeng Gao, Ji Qi, and Limin Wang. Multiple object tracking as id prediction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27883–27893, 2025. 3

[11] 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. 4, 6

[12] Yunzhong Hou and Liang Zheng. Multiview detection with shadow transformer (and view-coherent data augmentation). In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1673–1682, 2021. 3

[13] Yunzhong Hou, Liang Zheng, and Stephen Gould. Multiview detection with feature perspective transformation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 1–18. Springer, 2020. 1, 3

[14] Mengjie Hu, Xiaotong Zhu, Haotian Wang, Shixiang Cao, Chun Liu, and Qing Song. Stdformer: Spatial-temporal motion transformer for multiple object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(11): 6571–6594, 2023. 3

[15] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960. 3

[16] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 7482–7491, 2018. 5

[17] Yunhao Li, Xiaoqiong Liu, Luke Liu, Heng Fan, and Libo Zhang. Lamot: Language-guided multi-object tracking. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6816–6822. IEEE, 2025. 3

[18] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1

[19] 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 (ICCV)*, pages 2980–2988, 2017. 5

[20] Zelin Liu, Xinggang Wang, Cheng Wang, Wenyu Liu, and Xiang Bai. Sparsetrack: Multi-object tracking by performing scene decomposition based on pseudo-depth. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025. 3

[21] Kai Luo, Hao Shi, Sheng Wu, Fei Teng, Mengfei Duan, Chang Huang, Yuhang Wang, Kaiwei Wang, and Kailun Yang. Omnidirectional multi-object tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21959–21969, 2025. 3

[22] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8844–8854, 2022. 8

[23] Hong Mo, Xiong Zhang, Jianchao Tan, Cheng Yang, Qiong Gu, Bo Hang, and Wenqi Ren. Countformer: Multi-view crowd counting transformer. In *Computer Vision – ECCV 2024*, pages 20–40, Cham, 2025. Springer Nature Switzerland. 1

[24] Alexandru Niculescu-Mizil, Deep Patel, and Iain Melvin. Mctr: Multi camera tracking transformer. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 816–826, 2025. 3

[25] Jonah Ong, Ba-Tuong Vo, Ba-Ngu Vo, Du Yong Kim, and Sven Nordholm. A bayesian filter for multi-view 3d multi-object tracking with occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2246–2263, 2020. 8

[26] Rui Qiu, Ming Xu, Yuyao Yan, Jeremy S Smith, and Xi Yang. 3d random occlusion and multi-layer projection for deep multi-camera pedestrian localization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, pages 695–710. Springer, 2022. 3

[27] Kyujin Shim, Kangwook Ko, Yujin Yang, and Changick Kim. Focusing on tracks for online multi-object tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11687–11696, 2025. 3

[28] Liangchen Song, Jialian Wu, Ming Yang, Qian Zhang, Yuan Li, and Junsong Yuan. Stacked homography transformations for multi-view pedestrian detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6049–6057, 2021. 3

[29] Torben Teepe, Philipp Wolters, Johannes Gilg, Fabian Herzog, and Gerhard Rigoll. Lifting multi-view detection and tracking to the bird's eye view. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 667–676, 2024. 1, 2, 3, 6, 7, 8

[30] Torben Teepe, Philipp Wolters, Johannes Gilg, Fabian Herzog, and Gerhard Rigoll. Earlybird: Early-fusion for multi-view tracking in the bird's eye view. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 102–111, 2024. 1, 2, 3, 6, 7, 8

[31] Tai Huu-Phuong Tran, Duong Nguyen-Ngoc Tran, Ngoc Doan-Minh Huynh, Chi Dai Tran, Long Hoang Pham, Quoc Pham-Nam Ho, Huy-Hung Nguyen, Duong Khac Vu, Hyung-Min Jeon, Hyung-Joon Jeon, Son Hong Phan, Trinh Le Ba Khanh, and Jae Wook Jeon. Depthtrack: Cluster meets bev for multi-camera multi-target 3d tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 5289–5298, 2025. 3

[32] Lorenzo Vaquero, Yihong Xu, Xavier Alameda-Pineda, Víctor M. Brea, and Manuel Mucientes. Lost and found: Overcoming detector failures in online multi-object tracking. In *European Conf. Comput. Vis. (ECCV)*, pages 448–466. Springer, 2024. 3

[33] Jeet Vora, Swetanjal Dutta, Kanishk Jain, Shyamgopal Karthik, and Vineet Gandhi. Bringing generalization to deep multi-view pedestrian detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 110–119, 2023. 3

[34] Peng Wang, Yongcai Wang, Hualong Cao, Wang Chen, and Deying Li. La-motr: End-to-end multi-object tracking by learnable association. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12438–12448, 2025. 3

[35] Yizhou Wang, Tim Meinhardt, Orcun Cetintas, Cheng-Yen Yang, Sameer Pusegaonkar, Benjamin Missaoui, Sujit Biswas, Zheng Tang, and Laura Leal-Taixe. Mcblt: Multi-camera multi-object 3d tracking in long videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 5245–5254, 2025. 3, 6, 8

[36] Yihong Xu, Yutong Ban, Guillaume Delorme, Chuang Gan, Daniela Rus, and Xavier Alameda-Pineda. Transcenter: Transformers with dense representations for multiple-object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):7820–7835, 2023. 3, 8

[37] Taiga Yamane, Ryo Masumura, Satoshi Suzuki, and Shota Orihashi. Mvtrajecter: Multi-view pedestrian tracking with trajectory motion cost and trajectory appearance cost. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13270–13280, 2025. 3, 6, 8

[38] Yihan Yang, Ming Xu, Jason F Ralph, Yuchen Ling, and Xiaonan Pan. An end-to-end tracking framework via multi-view and temporal feature aggregation. *Computer Vision and Image Understanding*, 249:104203, 2024. 6, 8

[39] Quanzeng You and Hao Jiang. Real-time 3d deep multi-camera tracking. *arXiv preprint arXiv:2003.11753*, 2020. 8

[40] Rui Zeng, Yuanzhou Huang, and Songwei Pei. Tgformer: Transformer with track query group for multi-object tracking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(9):9824–9832, 2025. 3

[41] Qi Zhang and Antoni B Chan. Wide-area crowd counting via ground-plane density maps and multi-view fusion cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8297–8306, 2019. 1, 6

[42] Qi Zhang, Kaiyi Zhang, Antoni B Chan, and Hui Huang. Mahalanobis distance-based multi-view optimal transport for multi-view crowd localization. In *European Conference on Computer Vision*, pages 19–36. Springer, 2024. 3

[43] Juanjuan Zhao, Liutao Zhang, Jiexia Ye, and Chengzhong Xu. Mdlf: A multi-view-based deep learning framework for individual trip destination prediction in public transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13316–13329, 2021. 1

[44] Zelin Zhao, Ze Wu, Yueqing Zhuang, Boxun Li, and Jiaya Jia. Tracking objects as pixel-wise distributions. In *European Conference on Computer Vision*, pages 76–94. Springer, 2022. 3

[45] Zeyong Zhao, Yanchao Hao, Minghao Zhang, Qingbin Liu, Bo Li, Dianbo Sui, Shizhu He, and Xi Chen. Hff-tracker: A hierarchical fine-grained fusion tracker for referring multi-object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10528–10536, 2025. 3

[46] Xingyi Zhou, Tianwei Yin, Vladlen Koltun, and Philipp Krähenbühl. Global tracking transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8771–8780, 2022. 3

[47] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 6