# IP-MOT: Instance Prompt Learning for Cross-Domain Multi-Object Tracking


## Abstract

*Multi-Object Tracking (MOT) aims to associate multiple objects across video frames and is a challenging vision task due to inherent complexities in the tracking environment. Most existing approaches train and track within a single domain, resulting in a lack of cross-domain generalizability to data from other domains. While several works have introduced natural language representation to bridge the domain gap in visual tracking, these textual descriptions often provide too high-level a view and fail to distinguish various instances within the same class. In this paper, we address this limitation by developing IP-MOT, an end-to-end transformer model for MOT that operates without concrete textual descriptions. Our approach is underpinned by two key innovations: Firstly, leveraging a pre-trained vision-language model, we obtain instance-level pseudo textual descriptions via prompt-tuning, which are invariant across different tracking scenes; Secondly, we introduce a query-balanced strategy, augmented by knowledge distillation, to further boost the generalization capabilities of our model. Extensive experiments conducted on three widely used MOT benchmarks, including MOT17, MOT20, and DanceTrack, demonstrate that our approach not only achieves competitive performance on same-domain data compared to state-of-the-art models but also significantly improves the performance of query-based trackers by large margins for cross-domain inputs.*


## 1. Introduction

Multi-object Tracking is one of the fundamental visual tracking tasks [34, 35, 49], with applications ranging from human-computer interaction, surveillance, autonomous driving, etc. It aims at detecting the bounding box of the target and associating the same target across consecutive frames in a video sequence. Recent MOT approaches can be categorized into tracking-by-detection methods and joint detection and tracking-by-query methods. Tracking-by-detection methods have emerged as the dominant tracking paradigm for several years, powered by advances in deep learning and real-time object detectors [7, 13]. In this paradigm, a detector first identifies objects' bounding boxes within each single frame, followed by an association model that generates trajectories by linking these identified objects across subsequent frames. This process employs techniques such as motion-based tracking using the Kalman filter [40] and Re-identification (Re-ID) [2, 8] methods to ensure accurate object matching [1, 4, 6, 41, 48, 49, 51]. On the other hand, tracking-by-query methods have recently gained traction, offering a more holistic, end-to-end MOT approach. These query-based methods [5, 23, 36, 47] perform detection and tracking concurrently by leveraging the interplay and progressive decoding of detect and track queries within a Transformer framework.

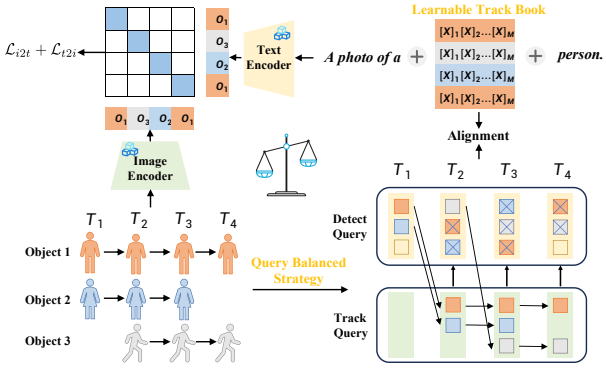



Figure 1. **IP-MOT**. We propose IP-MOT, which further improves the generalization ability of the model by using a online learnable TrackBook instead of a manually designed TrackBook to obtain a more fine-grained instance-level textual description. Meanwhile, a query balanced strategy (QBS) is also proposed to further improve the tracking and detection accuracy of IP-MOT for cross-domain and some-domain inputs.

While previous methods have shown significant performance in certain contexts, their predominant focus on homogenous domains constrains their versatility. This specialization results in limited applicability across diverse scenarios, creating a development bottleneck in the MOT field. Some works [16, 46] try to integrate natural language representations to enhance domain adaptability. However, the limited availability of detailed textual descriptions has restricted the improvement of these models' generalization abilities.

Based on the analysis above, in this paper, we focus

on leveraging natural language presentation by proposing a instance-level language-augmented Multi-Object Tracking method with Transformer, coined as **IP-MOT**. We use pre-trained vision-language models like CLIP [28] to introduce the natural language representation into MOT models, as illustrated in Figure 1. Unlike using a handcrafted textual descriptions, our model maintains a trainable TrackBook to generate instance-level textual descriptions for each tracked object through prompt-tuning, which contain invariant information of tracked targets across different tracking scenes. Afterward, we align output embedding with its corresponding stable textual representation through contrastive learning to improve the generalization ability. Besides, we apply triplet loss to produce a more distinguishable representation.

Although the MOTR [47] architecture is elegant, it suffers from the optimization conflict between detection and association critically, which finally results in poor detection precision. Therefore, to overcome the unfair label assignment problem between detect queries and track queries, we propose a query balanced strategy where the detect queries are responsible for detecting all appeared targets and extra duplication module is used to filter out the same target from detection results. This strategy not only refines the tracking accuracy of IP-MOT but also enriches the training dataset for textual description alignment, thereby boosting cross-domain generalization.

To evaluate the generalization performance of our model, we train our models on MOT17 and validate it on MOT20 dataset. We also evaluate our model on MOT17 and challenging DanceTrack dataset to show the performance for same-domian inputs. The experimental results reveal that our approach not only achieves competitive performance on same-domain data compared to state-of-the-art models but also significantly improves the performance of query-based trackers by large margins for cross-domain inputs. In addition, we perform extensive ablation studies to further demonstrate the effectiveness of our designs.

## 2. Related Work

Existing MOT algorithms can be divided into two mainstream approaches according to the paradigm of handling the detection and association, *i.e.*, the tracking-by-detection and tracking-by-query methods.

**Tracking-by-Detection** is a common practice in the MOT field, where object detection and data association are treated as separate modules. The methods [1, 4, 6, 41, 49] use an existing detector [11, 13, 30] and then integrate detections through a distinct motion tracker across consecutive frames, employing various techniques. SORT [4] initiated the use of the Kalman filter [40] for object tracking, associating each bounding box with the highest overlap through the Hungarian algorithm [15]. DeepSORT [41] enhanced this by incorporating both motion and deep appearance features, while StrongSORT [10] further integrated lightweight, appearance-free algorithms for detection and association. ByteTrack [49] addressed fragmented trajectories and missing detections by utilizing low-confidence detection similarities. P3AFormer [51] combined pixel-wise distribution architecture with Kalman filter to refine object association, and OC-SORT [6] amended the linear motion assumption within the Kalman Filter for superior adaptability to occlusion and non-linear motion.

**Tracking-by-Query methods.** In recent years, there have been several explorations into the one-stage paradigm, which combines object detection and data association into a single pipeline. Unlike the tracking-by-detection paradigm mentioned above, tracking-by-query methods apply the track query to decode the location of tracked objects progressively. Inspired by DETR-family [7, 54], most of these methods [23, 47] leverage the learnable object query to perform newborn object detection, while the track query localizes the position of tracked object. Techniques such as TrackFormer [23] and MOTR [47] perform simultaneous object detection and association using concatenated object and track queries. TransTrack [36] employs cyclical feature passing to aggregate embeddings, while MeMOT [5] encodes historical observations to preserve extensive spatio-temporal memory.

**Pre-trained Vision-Language Models**. Recently, the Pre-trained vision-language CLIP model [28] measures the similarity between images and text, mapping images and their corresponding textual descriptions into a shared embedding space that allows the model to perform various tasks, such as image segmentation [29], few-shot learning [38] and image caption [25]. In addition to the above applications, the pre-trained CLIP encoders are also applied to MOT, e.g., open-vocabulary tracking [16], language-guided tracking [46] and so on. However, to the best of our knowledge, we are the first to use prompt tuning to distill the knowledge contained in the CLIP and obtain instance-level pseudo textual descriptions that can be used to boost the generalization performance of visual MOT models.

**Domain Generalization for MOT.** Although the performance of the aforementioned methods is competitive, they do not perform consistently with that of the training domain. Some works try to bridge this gap by introducing natural language representation, such as LTrack [46] employing hand-crafted TrackBook to inject language information into MOT, OVTrack simply adopting a constant textual description. While previous methods alleviate the generalization problem in MOT, there is still room for improvement when it comes to textual description generation strategy. Inspired by the recent advances in natural language process (NLP), we use CLIP to automatically generate distinguishable instance-level textual description via prompt tuning. In

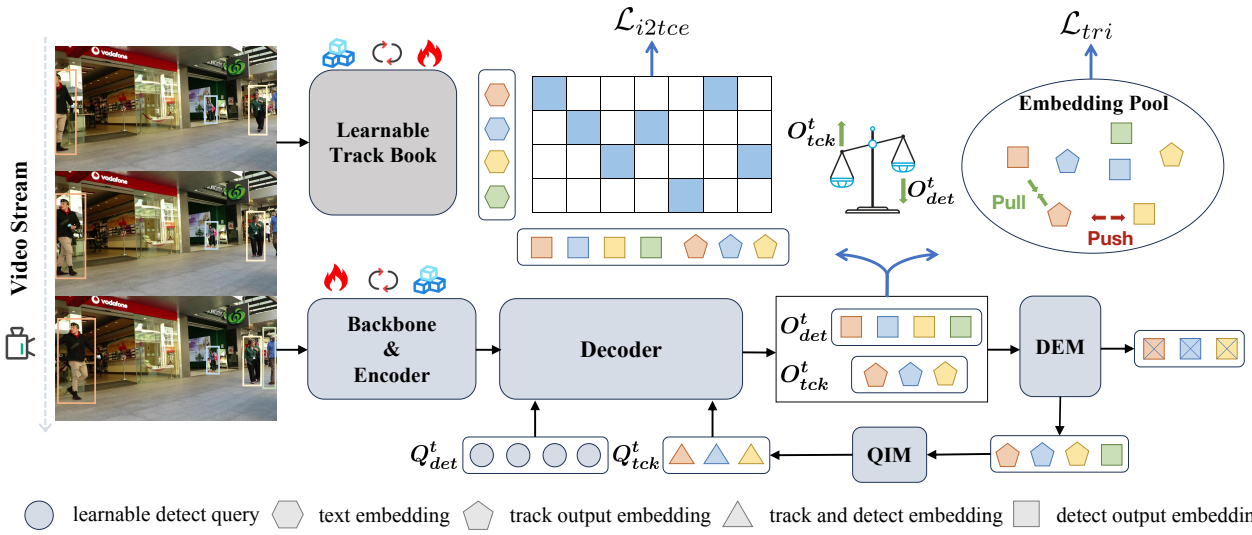



Figure 2. **The overall architecture of IP-MOT**. We use different colors to indicate different tracked targets, and the same color represents the same target. In each iteration, we first optimize our trainable TrackBook to obtain a instance-level textual description based on the target in a clip of video stream. Then, adopt a ResNet-50 [14] backbone and a Transformer [39] Encoder to learn a 2D representation of an input image. Afterward, the Decoder processes the detect query $Q_{det}$ and track $Q_{tck}$ , and generates the detect output embedding $O_{det}$ and track output embedding $O_{tck}$, respectively. Finally, we add output embedding into the clip-level embedding pool and align them with the corresponding frozen textual description presentation. Since the query balanced strategy (QBS) is used to alleviate unfair label assignment conflict, we designe a simple and elegant deduplication module (DEM) to duplicate detection results.

order to make better use of invariant information, we propose query balanced strategy to enhance the tracked object feature alignment over time for a more distinguishable and stable representation. Our comprehensive ablation studies further validate the effectiveness of our approach.

## 3. Methodology

### 3.1. Method Overview

We propose the **IP-MOT**, an instance-level language-augmented Transformer for multi-object tracking. Different from most existing methods [16, 46], which only explicitly utilize hand-crafted textual description, our core contribution involves constructing a TrackBook (in Section 3.2) that maintains learnable instance-level textual description for each tracked target. Additionally, we introduce a deduplication module (DEM) that effectively boosts the performance of IP-MOT via filtering out the redundant target from detection results.

As shown in Figure 2, we use a ResNet50 [14] backbone and a Transformer Encoder to produce the image feature of an input frame $I_t$. Afterward by querying the encoded image feature with $[Q^t_{det}, Q^t_{tck}]$, the Transformer Decoder produces the corresponding output $[O^t_{det}, O^t_{tck}]$. It is worth noting that in our paper, $Q^t_{det}$ is responsible for detecting all targets, while $Q^t_{tck}$ is responsible for detecting each tracked target. Then, we predict the classification confidence $c^t_i$, bounding box $b^t_i$, and instance embedding $e^t_i$ corresponding to the $i^{th}$ target from the output emdeddings. For simplicity, we skip the training of the trainable TrackBook and directly obtain the corresponding textual representation $p^t_i$. Finally, after aligning the text representation $p^t_i$ and the instance embedding $e^t_i$, we filter out the redundant target detected by $Q^t_{det}$ based on the deduplication confidence $d^t_i$, output of the DEM, to retain the newborn target. The details of our components will be elaborated in the following sections.

### 3.2. Trainable TrackBook

We first briefly review CLIP. It consists of two encoders, an image encoder $\mathcal{I}(\cdot)$ and a text encoder $\mathcal{T}(\cdot)$. The text encoder $\mathcal{T}(\cdot)$ and image encoder $\mathcal{I}(\cdot)$ are implemented as two transformers, which are used to generate a representation from a textual description and image respectively. Specifically, $i \in \{1...B\}$ denotes the index of the image-text pair within a batch. Let $img_i$ be the [CLS] token embedding of image feature, while $text_i$ is the corresponding [EOS] token embedding of text feature, then compute the similarity between $img_i$ and $text_i$:

$$s(V_i, T_i) = V_i \cdot T_i = g_V(img_i) \cdot g_T(text_i) \quad (1)$$

where $g_V(\cdot)$ and $g_T(\cdot)$ are linear layers projecting embedding into a shared embedding space. The image-to-text con-

trastive loss $\mathcal{L}_{i2t}$ is calculated as:

$$\mathcal{L}_{i2t}(i) = -\log \frac{\exp(s(V_i, T_i))}{\sum_{a=1}^{B} \exp(s(V_i, T_a))} \tag{2}$$

and the text-to-image contrastive loss $\mathcal{L}_{t2i}$:

$$\mathcal{L}_{t2i}(i) = -\log \frac{\exp(s(V_i, T_i))}{\sum_{a=1}^{B} \exp(s(V_a, T_i))} \tag{3}$$

where numerators in Eq. (2) and Eq. (3) are the similarities of two embeddings from matched pair, and the denominators are all similarities with respect to anchor $V_i$ or $T_i$.

We build trainable TrackBook by introducing ID-specific learnable tokens to learn ambiguous textual descriptions, which are independent for each object ID. Specifically, the text descriptions fed into $\mathcal{T}(\cdot)$ are designed as "A photo of a $[\mathrm{X}]_1[\mathrm{X}]_2[\mathrm{X}]_3...[\mathrm{X}]_\mathrm{M}$ person", where each $[\mathrm{X}]_\mathrm{m}(\mathrm{m} \in 1, ...\mathrm{M})$ is a learnable text token with the same dimension as word embedding. $\mathrm{M}$ indicates the number of learnable text tokens. During training phrase, we fix the parameters of $\mathcal{I}(\cdot)$ and $\mathcal{T}(\cdot)$, and only tokens $[\mathrm{X}]_\mathrm{m}$ are optimized.

Similar to CLIP, we use $\mathcal{L}_{i2t}$ and $\mathcal{L}_{t2i}$, but replace $text_i$ with $text_{o_i}$ in Eq. (1), since each object ID shares the same text description. Moreover, for $\mathcal{L}_{t2i}$, different images in a batch probably belong to the same person, so $T_{o_i}$ may have more than one positive, we change it to:

$$\mathcal{L}_{t2i}(o_i) = -\frac{1}{|P(o_i)|} \log \frac{\sum_{p \in P(o_i)} \exp(s(V_p, T_{o_i}))}{\sum_{a=1}^{B} \exp(s(V_a, T_{o_i}))} \tag{4}$$

where $P(o_i) = \{p \in 1...B : o_p = o_i\}$ is the set of indices of all positives for $T_{o_i}$ in the batch, and $|\cdot|$ is its cardinality. By minimizing the loss of $\mathcal{L}_{i2t}$ and $\mathcal{L}_{t2i}$, we can obtain textual description for each tracked object.

### 3.3. Query Balanced Strategy

Although the MOTR [47] architecture is elegant, it suffers from the optimization conflict between detection and association critically. During training, the number of label assignments of the track queries is several times that of the detect queries label assignment, and the insufficient training of the detect query eventually leads to poor detection accuracy. Therefore, to overcome the unfair label assignment problem between detect queries and track queries, we propose a query balanced strategy in which detect queries are responsible for detecting all appeared targets, so that the number of both queries is balanced during supervised training. Moreover, query balanced strategy can provide more training samples for invariant textual description alignment to further boost cross-domain generalization capability. This simple but effective strategy can alleviate aforementioned problem and improve the performance of the model for same-domain and cross-domain inputs.

### 3.4. Deduplication Module

Since the query balanced strategy is adopted, we design a deduplication module (DEM) to filter out the redundant targets in the detect query and leave only the newborn targets. DEM consists of a one-layer multi-head self-attention and a two-layer MLP, and then the deduplication confidence $d^t = [d^t_{det}, d^t_{tck}]$ is calculated as:

$$[d^t_{det}, d^t_{tck}] = \mathrm{MLP}(\mathrm{MHA}([O^t_{det}, O^t_{tck}])) \tag{5}$$

In order to ensure the end-to-end elegance of the approach, we have only made a minor change to the calculation of the tracking score $s^t$:

$$s^t = \mathrm{Sqrt}(\mathrm{Sigmoid}(c^t) \cdot \mathrm{Sigmoid}(d^t)) \tag{6}$$

where $c^t$ is classification confidence, $d^t$ is deduplication confidence.

### 3.5. Model Training

For each iteration in one epoch, we first optimize our trainable TrackBook by minimizing $\mathcal{L}_{t2i}$ and $\mathcal{L}_{i2t}$ loss. In order to better train the online TrackBook, we use distributed training operation to gather all training samples located on different nodes. Afterward, we freeze the TrackBook and optimize IP-MOT with extension of the collective average loss. Given a clip $V_\xi$ of $N$ frames as input, the results predicted by the model are denoted as $\widehat{P} = \{\hat{p}_i\}_{i=1}^N$, and the corresponding ground-truths are $P = \{p_i\}_{i=1}^N$. The collective average loss $\mathcal{L}_{clip}$ is computed based on $\widehat{P}$ and $P$. It consists of two parts, the tracking loss and detection loss. These two losses exactly share the same form. The difference is that the tracking loss is for localizing the targets that have been recognized in previous frames, and the detection loss is to tackle the newborn targets. Mathematically, The original collective average loss $\mathcal{L}_{clip}$ can be formulated as follows:

$$\mathcal{L}_{clip} = \frac{1}{T} \sum_{n=1}^{N} (\mathcal{L}(\widehat{P}^i_{tck}|_{q_t}, P^i_{tck}) + \mathcal{L}(\widehat{P}^i_{det}|_{q_d}, P^i_{det})) \tag{7}$$

where $\widehat{P}^i_{tck}|_{q_t}$, $P^i_{tck}$, $\widehat{P}^i_{det}|_{q_d}$, and $P^i_{det}$ are the association predictions, association labels, detection predictions, and detection labels, respectively. $T$ denotes the total number of the targets in the clip $V_\xi$ of $N$ frames. $\mathcal{L}(\cdot)$ is implemented similarly to the one in DETR, which is formulated as:

$$\mathcal{L}(\widehat{P}_i|_{q_i}, P_i) = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{l_1}\mathcal{L}_{l_1} + \lambda_{giou}\mathcal{L}_{giou} \tag{8}$$

$\mathcal{L}_{cls}$, $\mathcal{L}_{l_1}$, and $\mathcal{L}_{giou}$ are the focal loss for classification, $L_1$ loss for regressing width and height, and the common generalized IoU loss. $\lambda_{cls}$, $\lambda_{l_1}$, $\lambda_{giou}$ are three hyper-parameters.

We employ the triplet loss $\mathcal{L}_{tri}$ and image to text cross-entropy loss $\mathcal{L}_{i2tce}$ with label smoothing to extend collective average loss $\mathcal{L}_{clip}$ , they are calculated as:

$$\mathcal{L}_{tri}(i) = \max(\|e_i - P(e_i)\|_2) - \|e_i - N(e_i)\|_2 + \alpha, 0) \quad (9)$$

$$\mathcal{L}_{i2tce}(i) = \sum_{k=1}^{T} -q_k \log \frac{\exp(s(V_i, T_{o_k}))}{\sum_{o_a=1}^{T} \exp(s(V_i, T_{o_a}))} \quad (10)$$

here $q_k = (1-\epsilon)\delta_{k,y} + \epsilon/T$ denotes value in the target distribution, $\|e_i - P(e_i)\|_2$ and $\|e_i - N(e_i)\|_2$ are $l_2$ euclidean distance of positive pair and negative pair, while $\alpha$ is the margin of $\mathcal{L}_{tri}$. Eventually, The extended collective average loss $\mathcal{L}^*_{clip}$ can be formulated as follows:

$$\mathcal{L}^*_{clip} = \mathcal{L}_{clip} + \frac{1}{T}\sum_{n=1}^{T}(\lambda_{tri}\mathcal{L}_{tri} + \lambda_{i2tce}\mathcal{L}_{i2tce}) \quad (11)$$

# 4. Experiments

## 4.1. Datasets and Metrics

**Datasets**. We evaluate our method on multiple multi-object tracking datasets, including MOT17 [24], MOT20 [9] and DanceTrack [37]. MOT17 and MOT20 are used for pedestrian tracking, where targets mostly move linearly, while scenes in MOT20 are more crowded. To verify the domain generalization ability of models to unseen domains, we train our model on MOT17 and validate them on MOT20. Additionally, we assess the same-domain performance of IP-MOT on DanceTrack, a challenging dataset have a similar appearance, severe occlusion, and frequent crossovers with highly non-linear motion.

**Metric**. We mainly use the higher order (HOTA [21]) metrics and CLEAR-MOT metrics to evaluate our method. Specifically, HOTA metrics consist of higher order tracking accuracy (HOTA), association accuracy score (AssA) and detection accuracy score (DetA). CLEAR-MOT Metrics include IDF1 score (IDF1) [31], multiple object tracking accuracy (MOTA) [3] and identity switches (IDS). Among them, HOTA, AssA, and IDF1 are crucial metrics for comparing tracking performance, while MOTA, DetA are the pivotal metrics for comparing detection performance.

## 4.2. Implementation Details

We adopt the visual encoder $\mathcal{I}(\cdot)$ and the text encoder $\mathcal{T}(\cdot)$ from CLIP as the backbone to optimize trainable TrackBook. We choose the ViT-B/16, which contains 12 transformer layers with the hidden size of 768 dimensions to extract image feature. To match the output of the $\mathcal{T}(\cdot)$, the dimension of the image feature vector is reduced from 768 to 512 by a linear layer. we use the AdamW optimizer with a learning rate initialized at 3.5e-4 and decayed by a cosine schedule to optimize the learnable text tokens $[\mathrm{X}]_{\mathrm{m}}(\mathrm{m} \in 1, ...\mathrm{M})$ in TrackBook. For each iteration, we first resize the cropped target to 128x256 and train our learnable TrackBook. And then, we align IP-MOT with frozen instance-level textual description.

| **Methods** | DanceTrack (Same-domain) | | | | |
|---|---|---|---|---|---|
| | HOTA | MOTA | DetA | AssA | IDF1 |
| *with extra data/memory:* | | | | | |
| MeMOTR [12] | 68.5 | 89.9 | 80.5 | 58.4 | 71.2 |
| MOTRv2 [50] | 69.9 | 91.9 | 83.0 | 59.0 | 71.7 |
| *w/o extra data/memory:* | | | | | |
| QDTrack [27] | 45.7 | 83.0 | 72.1 | 29.2 | 44.8 |
| FairMOT [48] | 39.7 | 82.2 | 66.7 | 23.8 | 40.8 |
| TraDes [42] | 43.3 | 86.2 | 74.5 | 25.4 | 41.2 |
| SORT [4] | 47.9 | **91.8** | 72.0 | 31.2 | 50.8 |
| ByteTrack [49] | 47.3 | 89.5 | 71.6 | 31.4 | 52.5 |
| OC-SORT [6] | 54.6 | 89.6 | 80.4 | 40.2 | 54.6 |
| TransTrack [36] | 45.5 | 88.4 | 75.9 | 27.5 | 45.2 |
| MOTR [47] | 54.2 | 79.7 | 73.5 | 40.2 | 51.5 |
| CenterTrack [52] | 41.8 | 86.8 | 78.1 | 22.6 | 35.7 |
| GTR [53] | 48.0 | 84.7 | 72.5 | 31.9 | 50.3 |
| DiffusionTrack [22] | 52.4 | 89.3 | **82.2** | 33.5 | 47.5 |
| C-BIoU [44] | 60.6 | 91.6 | 81.3 | 45.4 | 61.6 |
| **IP-MOT*** (ours) | 59.5 | 84.5 | 76.0 | 46.7 | 60.4 |
| **IP-MOT** (ours) | **61.9** | 88.2 | 79.0 | **48.7** | **62.0** |

Table 1. Same-domain performance comparison to state-of-the-art approaches on the Dancetrack test set. The best results are shown in **bold**. IP-MOT* means the result based on standard Deformable-DETR [54].

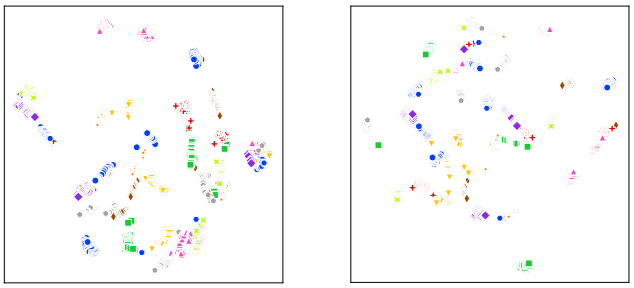
Following MeMOTR [12], we build IP-MOT based on DAB-Deformable-DETR [19], which is pre-trained on COCO and employs ResNet50 as backbone. To make fair comparison, we also provide the results of our model based on original Deformable-DETR [54] in Table 1. During the training process, the batch size is 1 and each batch contains a multi-frame video clip. The frames in each clip are selected from training videos with a random interval between 1 to 10. We use the AdamW [20] optimizer with the initial learning rate of 2.0e-4. Our models are conducted on PyTorch with 8 NVIDIA GeForce RTX 3090 with the some data augmentation strategy of MeMOTR, which includes random random flip and random crop. During the training process, $\lambda_{cls}$, $\lambda_{l_1}$, and $\lambda giou$ are set as 2, 5, and 2, while $\lambda_{tri}$, $\lambda_{i2tce}$, and the margin of $\alpha$ are set as 2, 4, and 0.3. IP-MOT is trained for totally 20 epochs and the learning rate decays by 10 at the $10^{th}$ epoch on DanceTrack dataset. For MOT17, we train our model on a joint train set with additional CrowdHuman val set [33] for totally 120 epochs and the learning rate decays by 10 at the $60^{th}$ epoch. For simplicity, we set score threshold $\tau = 0.5$ in our experiments.

| Methods | MOT17 (Same-domain) | | | | | |
|---|---|---|---|---|---|---|
| | MOTA | IDF1 | HOTA | AssA | DetA | IDS |
| *CNN based:* | | | | | | |
| GTR [53] | 75.3 | 71.5 | 59.1 | 57.0 | 61.6 | / |
| TubeTK [26] | 63.0 | 58.6 | / | / | / | 4137 |
| CenterTrack [52] | 67.8 | 64.7 | 52.2 | 51.0 | 53.8 | 3039 |
| ByteTrack [49] | 80.3 | 77.3 | 63.1 | 62.0 | 64.5 | 2196 |
| FairMOT [48] | 73.7 | 72.3 | 59.3 | 58.0 | 60.9 | 3303 |
| StrongSORT [10] | 79.6 | 79.5 | 64.4 | 64.4 | 64.6 | 1194 |
| OC-SORT [6] | 78.0 | 77.5 | 63.2 | 63.4 | 63.2 | 1950 |
| BoT-SORT [1] | 80.5 | 80.2 | 65.0 | 65.5 | 64.9 | 1212 |
| *Transformer based:* | | | | | | |
| TrackFormer [23] | 74.1 | 68.0 | 57.3 | 54.1 | 60.9 | 2829 |
| TransTrack [36] | **74.5** | 63.9 | 54.1 | 47.9 | **61.6** | 3663 |
| TransCenter [43] | 73.2 | 62.2 | 54.5 | 49.7 | 60.1 | 4614 |
| MeMOT [5] | 72.5 | 69.0 | 56.9 | 55.2 | / | 2724 |
| MOTR [47] | 71.9 | 68.4 | 57.2 | 55.8 | 58.9 | 2115 |
| LTrack [46] | 72.1 | 69.1 | 57.5 | 56.1 | 59.4 | 2100 |
| **IP-MOT** (ours) | 73.2 | **69.6** | **58.2** | **56.4** | 60.4 | **1896** |

Table 2. Same-domain performance comparison to state-of-the-art approaches on the MOT17 test set with the private detections. The best results are shown in **bold**.

| Methods | MOT20 (Cross-domain) | | | | | |
|---|---|---|---|---|---|---|
| | MOTA | IDF1 | HOTA | AssA | DetA | Data |
| CenterTrack [52] | 42.9 | 39.0 | 29.7 | 25.6 | 35.0 | CH+17 |
| FairMOT [48] | 57.6 | 53.8 | 41.9 | 35.9 | 49.7 | CH+17 |
| TraDeS [42] | 44.9 | 39.3 | 28.0 | 25.5 | 32.7 | CH+17 |
| CSTrack [18] | 49.6 | 44.9 | 33.9 | 29.8 | 38.8 | CH+17 |
| OMC [17] | 55.9 | 49.4 | 38.8 | 32.2 | 46.9 | CH+17 |
| MTrack [45] | 54.8 | 52.9 | 40.6 | 37.0 | 44.9 | CH+17 |
| TransTrack [36] | 58.1 | 44.8 | 35.8 | 27.3 | 47.3 | CH+17 |
| MOTR [47] | 54.2 | 56.0 | 43.1 | 42.3 | 43.9 | CH+17 |
| MeMOTR [12] | 55.7 | 56.4 | 43.1 | 41.8 | 44.7 | CH+17 |
| LTrack [46] | 57.4 | 60.4 | 46.2 | 43.8 | 48.2 | CH+17 |
| **IP-MOT** (ours) | **68.3** | **62.5** | **49.2** | **44.6** | **55.3** | CH+17 |

Table 3. Cross-domain performance comparison to state-of-the-art approaches on the MOT20 train set with the private detections. The best results are shown in **bold**. The used datasets of all methods are marked out in the column "Data" of this table (CH and 17 refer to CrowdHuman and MOT17, respectively). Notably, all methods are trained and validated in the same setting.



(a) MOTR (b) IP-MOT

Figure 3. **Visualization of track Output Embedding** $O_{tck}$ (the first 50 frames in sequence MOT20-02 on cross-domain benchmark) by using t-Distributed Stochastic Neighbor Embedding (t-SNE). Embeddings for different targets are marked in different colors and shapes. Our method (right) helps the model learn a more stable and distinguishable representation than MOTR (left) for the track output embedding. Corresponding tracking performance is shown in Table 7.

### 4.3. Same-domain State-of-the-art Comparison

**Comparison on the DanceTrack Dataset**. To evaluate IP-MOT under same-domain challenging non-linear object motion, we compare IP-MOT with the state-of-the-art methods on the DanceTrack test set. As shown in Table 1, IP-MOT* surpasses MOTR [47] by 7.7 (61.9 vs. 54.2) on HOTA, 6.5 (46.7 vs. 40.2) on AssA and 8.9 (60.4 vs. 51.5) on IDF1. Our method alleviates the unfair label assignment conflicts between detect and track queries by query balanced strategy. Therefore, IP-MOT* also has a significant improvement in detection accuracy, such as 4.8 (84.5 vs. 79.7) on MOTA and 2.5 (76.0 vs. 73.5) on DetA in addition to the improvement in tracking accuracy. Eventually, our method achieves 61.9 HOTA, 48.7 AssA, and 62.0 IDF1 without extra data or memory mechanism, showing promising potential by competitive performance compared with the state-of-the-art methods.

**Comparison on the MOT17 Dataset**. Query-based trackers suffer from serious overfitting problems in MOT17 since the number of training set in MOT17 is insufficient to train an end-to-end tracker. Therefore, our method only slightly improves the performance compared to original MOTR [47]. As illustrated in Table 2, IP-MOT obtains the metrics 58.2 HOTA, 56.4 AssA, and 69.6 IDF1 on MOT17. The competitive results indicate that IP-MOT can tackle same-domain tracking scenes well. Notably, IP-MOT also obtains competitive performance on the detection related metrics (60.4 DetA and 73.2 MOTA). Meanwhile, it only produces 1896 IDS, which is the lowest among all compared methods. The more continuous tracklets generated by IP-MOT demonstrate that our proposed learnable TrackBook can produce more distinguishable textual representations. Compared to the MOTR [47], the introduction of learnable text token and query balanced strategy consistently improves the detection (DetA) and association (AssA) accuracy by 1.4% and 2.6% correspondingly. These experimental results further validate the effectiveness of our designs.

### 4.4. Cross-domain State-of-the-art Comparison

**Comparison on the MOT20 Dataset**. We test our approach on the cross-domain evaluation benchmark pro-

| M | HOTA ↑ | MOTA ↑ | IDF1↑ | DetA ↑ | AssA ↑ |
|---|---|---|---|---|---|
| 2 | 31.4 | 38.5 | 42.7 | 29.8 | 33.8 |
| 4 | **33.1** | **40.9** | **45.7** | **31.0** | 36.0 |
| 6 | 32.0 | 38.0 | 44.6 | 28.8 | **36.7** |
| 8 | 31.0 | 37.9 | 41.7 | 29.7 | 32.8 |

Table 4. Ablation study of the length of learnable text token in TrackBook prompt, which is denoted as M.

| $L_{\mathcal{L}^{*}_{clip}}$ | HOTA ↑ | MOTA ↑ | IDF1↑ | DetA ↑ | AssA ↑ |
|---|---|---|---|---|---|
| 1 | 28.5 | 31.8 | 38.7 | 23.9 | 32.6 |
| 3 | 31.0 | 40.2 | 43.2 | 30.2 | 34.6 |
| 5 | **33.1** | **40.9** | **45.7** | **31.0** | **36.0** |

Table 5. Ablation experiments on the layers of the usage of new collective average loss in multi-layer auxiliary loss $\mathcal{L}_{clip*}$.

posed by LTrack [46]. As presented in Tab 3, the performance drop of IP-MOT is relatively small, while the performance of compared end-to-end methods drops significantly. Specifically, IP-MOT achieves 49.2 HOTA, 44.6 AssA and 62.5 IDF1, which significantly outperforms all compared methods by large margins. The results indicate that the generalization ability of IP-MOT to unseen domains is promising. In addition, IP-MOT surpasses previous SOTA method, LTrack, by 3.0 (49.2 vs. 46.2) on HOTA, 0.8 (44.6 vs. 43.8) on AssA and 2.1 (62.5 vs. 60.4) on IDF1, which further confirms the benefit of aliging with fine-grained instance-level textual description. We further prove our components' effectiveness in Section 4.5

### 4.5. Ablation Study

In this section, we study several components of our model through ablation studies. All the experiments are conducted on the cross-domain evaluation benchmark. To accelerate the ablation study process. We train our model on the MOT17 train-half set for totally 10 epochs and evaluate it on the MOT20 val-half set by TrackEval.

**Trainable TrackBook**. The prompt in TrackBook are designed as "A photo of a $[\mathrm{X}]_1[\mathrm{X}]_2[\mathrm{X}]_3...[\mathrm{X}]_{\mathrm{M}}$ person", where each $[\mathrm{X}]_{\mathrm{m}}(\mathrm{m} \in 1, ...\mathrm{M})$ is a learnable text token with the same dimension as word embedding and $\mathrm{M}$ indicates the length of learnable text tokens. The length of learnable text tokens $\mathrm{M}$ determines the semantic richness of the textual description, which in turn affects the performance of the IP-MOT. We experimentally search for a suitable length $\mathrm{M}$, as show in Table 4. Increasing $\mathrm{M}$ from 2 to 4 dramatically improves HOTA and AssA metrics by 5.4% and 6.5%, respectively. However, continuing to increase the length of learnable text tokens will cause semantic sparsity problems, thus slightly weakening the overall performance.

**New Collective Average Loss**. We extend original collective average loss $\mathcal{L}_{clip}$ by adding triplet loss $\mathcal{L}_{tri}$ and image to text cross entropy loss $\mathcal{L}_{i2tce}$ to get new collective average loss $\mathcal{L}_{clip*}$. We explore the effect of the layers of replacing the $\mathcal{L}_{clip}$ with $\mathcal{L}_{clip*}$ in multi-layer auxiliary loss on the performance of the model. As shown in Table 5, increasing the layers of $\mathcal{L}_{clip*}$ in multi-layer auxiliary loss can steadily improve the model's generalization performance. $\mathcal{L}_{i2tce}$ is responsible for aligning the target embedding with the corresponding text description representation to make the target association more stable. $\mathcal{L}_{tri}$ try to make target embedding more distinguishable by shortening the distance between same targets and alienating the distance between different targets. In Table 6 , our experimental results show that using either of these two losses can improve the generalization ability of the model, and the performance can be further enhanced by using them together. Therefore, the $\mathcal{L}_{clip*}$ loss helps the model learn a more stable and distinguishable representation, as visualized in Figure 3.

| $\mathcal{L}_{tri}$ | $\mathcal{L}_{i2tce}$ | HOTA ↑ | MOTA ↑ | IDF1 ↑ | AssA ↑ |
|---|---|---|---|---|---|
| - | - | 29.8 | 36.8 | 41.2 | 32.6 |
| ✓ | - | 32.2 | 39.8 | 44.6 | 35.0 |
| - | ✓ | 31.2 | 37.4 | 43.2 | 35.6 |
| ✓ | ✓ | **33.1** | **40.9** | **45.7** | **36.0** |

Table 6. Ablation study on the triplet loss $\mathcal{L}_{tri}$ and image to text cross entropy loss $\mathcal{L}_{i2tce}$.

| *Align* | *QBS* | HOTA ↑ | MOTA ↑ | IDF1 | AssA ↑ |
|---|---|---|---|---|---|
| *naïve* | | 24.6 | 24.9 | 30.3 | 28.7 |
| - | - | 29.8 | 36.8 | 41.2 | 32.6 |
| ✓ | - | 33.1 | 40.9 | 45.7 | 36.0 |
| - | ✓ | 29.6 | 36.8 | 39.8 | 31.7 |
| ✓ | ✓ | **36.9** | **48.8** | **50.2** | **36.7** |

Table 7. Ablation study on instance-level textual representation alignment (Align) and query balanced strategy (QBS). *naïve* meas the result based on standard Deformable-DETR [54].

**Query Balanced Strategy**. We perform the alignment by using the new collective average loss $\mathcal{L}_{clip*}$ and employ query balanced strategy (QBS) to further enhance the performance. In addition to explore the new collective average loss $\mathcal{L}_{clip*}$, we also ablate QBS in Table 7. It shows that by using the new collective average loss with QBS, our IP-MOT achieves much better performance (36.9 vs. 33.1 on HOTA), especially improving MOTA by 19.3%. However, without alignment, QBS produces worse performance (-3.4% IDF1 and -2.8% AssA). We explain that $\mathcal{L}_{clip*}$ and QBS can complement each other and ultimately achieve a better performance. Specifically, QBS can provide more training samples for clip-level embedding pool for alignment training, while alignment training can provide QBS

(a) target* #1 (b) target #1 (c) color #1 (d) texture #1 (e) target* #2 (f) target #2 (g) color #2 (h) texture #2

(i) target* #3 (j) target #3 (k) color #3 (l) texture #3 (m) target* #4 (n) target #4 (o) color #4 (p) texture #4

Figure 4. **Visualization of instance-level textual description**. Since the Stable Diffuion [32] and the CLIP [28] share a same text encoder, we can generate the corresponding image based on the instance-level textual description. Target* means the original target in MOT17 dataset, while target, color, and texture means corresponding synthetic image by replacing the last word in textual description with "person","color", and "texture", respectively.

with more stable and distinguishable out embedding to alleviate deduplication difficulties. Therefore, using them together can obtain enhanced performance.

### 4.6. Visualization

To better demonstrate the superiority of IP-MOT, we visualize some instance-level textual description from TrackBook. Leveraging the shared text encoder between the Stable Diffusion model [32] and CLIP [28], we generated corresponding images based on these detailed textual descriptions. Figure 4 displays various targets from the training set alongside their synthetic counterparts depicting person, color, and texture. These synthesized images accurately capture the original targets' attributes, including color, texture, and high-level semantic information like clothing, hats, bags, and gender. In contrast, LTrack [46], the previous cross-domain SOTA method, utilized a manually designed TrackBook, limiting the embedding's interpretability and semantic richness. Our method overcomes these constraints by efficiently generating textual descriptions with robust semantics, stability, and recognizability, thus enhancing the model's generalizability.

## 5. Conclusion

In this paper, we propose IP-MOT, an end-to-end instance-level language-augmented Transformer for multi-object tracking without concrete textual description. Our method builds a trainable TrackBook to obtain stable textual description for each tracked object and exploits this description to augment the representation of track embedding, thereby improving cross-domain association performance. Furthermore, through the use of a query balanced strategy, our model improves the detection accuracy, making various targets more distinguishable and stable. Consequently, IP-MOT not only exhibits competitive performance on same-domain MOT benchmarks, but also achieves the state-of-the-art performance on cross-domain MOT benchmarks. Comprehensive ablation experiments and visualizations substantiate the effectiveness of our components. We hope that future work will pay more attention on leveraging textual descriptions in multi-object tracking.

**Limitation**. While IP-MOT demonstrates excellent performance in cross-domain MOT benchmarks, it faces challenges in scenarios with multiple similar targets in the same frame. This difficulty stems from the limitations of distinguishing targets using only textual and semantic information. In such cases, appearance cues are unreliable, necessitating spatial priors or advanced post-processing for effective tracking.

## References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Botsort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 1, 2, 6

[2] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 941–951, 2019. 1

[3] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 5

[4] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 1, 2, 5

[5] 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 CVPR*, pages 8090–8100, 2022. 1, 2, 6

[6] Jinkun Cao, Xinshuo Weng, Rawal Khirodkar, Jiangmiao Pang, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. *arXiv preprint arXiv:2203.14360*, 2022. 1, 2, 5, 6

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the ECCV*, pages 213–229. Springer, 2020. 1, 2

[8] Long Chen, Haizhou Ai, Zijie Zhuang, and Chong Shang. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *2018 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2018. 1

[9] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020. 5

[10] Yunhao Du, Yang Song, Bo Yang, and Yanyun Zhao. Strongsort: Make deepsort great again. *arXiv preprint arXiv:2202.13514*, 2022. 2, 6

[11] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the ICCV*, pages 6569–6578, 2019. 2

[12] Ruopeng Gao and Limin Wang. Memotr: Long-term memory-augmented transformer for multi-object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9901–9910, 2023. 5, 6

[13] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 1, 2

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the CVPR*, pages 770–778. IEEE, 2016. 3

[15] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 2

[16] Siyuan Li, Tobias Fischer, Lei Ke, Henghui Ding, Martin Danelljan, and Fisher Yu. Ovtrack: Open-vocabulary multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5567–5577, 2023. 1, 2, 3

[17] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, and Weiming Hu. One more check: Making "fake background" be tracked again. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1546–1554, 2022. 6

[18] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, Shuyuan Zhu, and Weiming Hu. Rethinking the competition between detection and reid in multiobject tracking. *IEEE Transactions on Image Processing*, 31:3182–3196, 2022. 6

[19] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: Dynamic anchor boxes are better queries for DETR. In *International Conference on Learning Representations*, 2022. 5

[20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the ICLR*, 2018. 5

[21] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129:548–578, 2021. 5

[22] Run Luo, Zikai Song, Lintao Ma, Jinlin Wei, Wei Yang, and Min Yang. Diffusiontrack: Diffusion model for multi-object tracking. *arXiv preprint arXiv:2308.09905*, 2023. 5

[23] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the CVPR*, pages 8844–8854, 2022. 1, 2, 6

[24] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 5

[25] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 2

[26] Bo Pang, Yizhuo Li, Yifan Zhang, Muchen Li, and Cewu Lu. Tubetk: Adopting tubes to track multi-object in a one-step training model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6308–6318, 2020. 6

[27] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 164–173, 2021. 5

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 8

[29] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18082–18091, 2022. 2

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region

proposal networks. *Advances in neural information processing systems*, 28, 2015. 2

[31] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Proceedings of the ECCV*, pages 17–35. Springer, 2016. 5

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 8

[33] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 5

[34] Zikai Song, Junqing Yu, Yi-Ping Phoebe Chen, and Wei Yang. Transformer tracking with cyclic shifting window attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8791–8800, 2022. 1

[35] Zikai Song, Run Luo, Junqing Yu, Yi-Ping Phoebe Chen, and Wei Yang. Compact transformer tracker with correlative masked modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2023. 1

[36] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020. 1, 2, 5, 6

[37] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the CVPR*, pages 20993–21002, 2022. 5

[38] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021. 2

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 3

[40] Greg Welch, Gary Bishop, et al. An introduction to the kalman filter. 1995. 1, 2

[41] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 1, 2

[42] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment: An online multi-object tracker. In *Proceedings of the CVPR*, pages 12352–12361, 2021. 5, 6

[43] 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*, 2022. 6

[44] Fan Yang, Shigeyuki Odashima, Shoichi Masui, and Shan Jiang. Hard to track objects with irregular motions and similar appearances? make it easier by buffering the matching space. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4799–4808, 2023. 5

[45] En Yu, Zhuoling Li, and Shoudong Han. Towards discriminative representation: Multi-view trajectory contrastive learning for online multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8834–8843, 2022. 6

[46] En Yu, Songtao Liu, Zhuoling Li, Jinrong Yang, Zeming Li, Shoudong Han, and Wenbing Tao. Generalizing multiple object tracking to unseen domains by introducing natural language representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3304–3312, 2023. 1, 2, 3, 6, 7, 8

[47] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *Proceedings of the ECCV*, pages 659–675, 2022. 1, 2, 4, 5, 6

[48] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129:3069–3087, 2021. 1, 5, 6

[49] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Proceedings of the ECCV*, pages 1–21. Springer, 2022. 1, 2, 5, 6

[50] Yuang Zhang, Tiancai Wang, and Xiangyu Zhang. Motrv2: Bootstrapping end-to-end multi-object tracking by pretrained object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22056–22065, 2023. 5

[51] Zelin Zhao, Ze Wu, Yueqing Zhuang, Boxun Li, and Jiaya Jia. Tracking objects as pixel-wise distributions. In *Proceedings of the ECCV*, pages 76–94. Springer, 2022. 1, 2

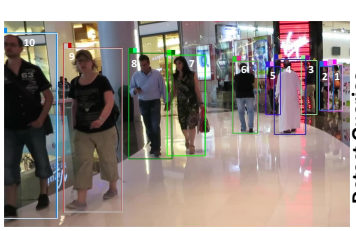
[52] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Proceedings of the ECCV*, pages 474–490. Springer, 2020. 5, 6

[53] Xingyi Zhou, Tianwei Yin, Vladlen Koltun, and Philipp Krähenbühl. Global tracking transformers. In *CVPR*, 2022. 5, 6

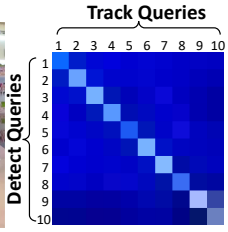
[54] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2, 5, 7

# IP-MOT: Instance Prompt Learning for Cross-Domain Multi-Object Tracking

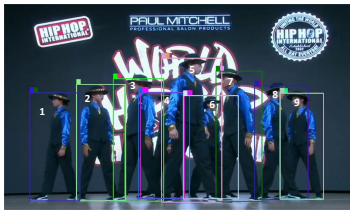
## Supplementary Material



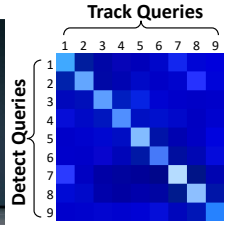
(a) Detect and track query prediction




(b) Query attention map



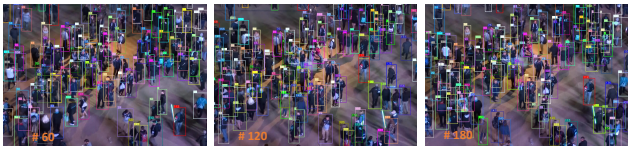
(c) Detect and track query prediction




(d) Query attention map

Figure 1. **Visualization** of 1a, 1c IP-MOT track query box prediction highly overlaps the detect query box prediction on the same-domain MOT17 and DanceTrack test set, and 1b, 1d the query self-attention map shows a clear exchange of information between the detect query and the corresponding track query of the same instance.


(a) "dancetrack0050" sequence on DanceTrack test set.




(b) "MOT20-02" sequence on MOT20 test set.

Figure 2. **Visualization of generalizability for cross-domain inputs**. We train our model on MOT17 datatset and visualize it for two cross-domain inputs.

## A. Pseudo-code of Optimization in IP-MOT

For each iteration in one epoch, we first optimize our trainable TrackBook. In order to better train the online TrackBook, we gather all training samples located on different nodes. Afterward, we freeze the TrackBook and optimize IP-MOT with extension of the collective average loss. The pseudo-code of optimization in IP-MOT is shown in follow-

**Algorithm 1:** Pseudo-code of Optimization in IP-MOT.

**Input:** A clip of video sequence `V` and corresponding ground-truth label `Y`; IP-MOT `IP-MOT`; deduplication module `DEM`; clip matcher `Match`; track score threshold $\tau$; detection score threshold $\tau_{det}$; trainable TrackBook `B`;
**Output:** $\emptyset$

Initialization: $\mathcal{P}_{crop}, \mathcal{P}_{emb} \leftarrow \emptyset$

`/* activate the frozen TrackBook */`
Activation: $\texttt{B} \leftarrow ACT(\texttt{B})$

**for** *frame* $(f_t, y_t)$ *in* $(\texttt{V}, \texttt{Y})$ **do**

`/* crop each object at current frame based on corresponding label */`
$\mathcal{C}_t \leftarrow \texttt{CROP}(f_t, y_t)$
$\mathcal{P}_{crop} \leftarrow \mathcal{C}_t \cup \mathcal{P}_{crop}$
**end**

`/* gather all object from distributed nodes and optimize TrackBook by minimizing` $\mathcal{L}_{i2t}$ `and` $\mathcal{L}_{t2i}$ `*/`
All-Gather: $\mathcal{P}_{crop} \leftarrow AllGather(\mathcal{P}_{crop})$
$Optimize(\mathcal{L}_{i2t}(\mathcal{P}_{crop}) + \mathcal{L}_{t2i}(\mathcal{P}_{crop}))$

`/* freeze the activated TrackBook */`
Frozen: $\texttt{B} \leftarrow Frozen(\texttt{B})$

Initialization: $\mathcal{Q}_{tck}, \mathcal{L}_{clip*} \leftarrow \emptyset, 0$
**for** *frame* $(f_t, y_t)$ *in* $(\texttt{V}, \texttt{Y})$ **do**

`/* initial detect queries and predict association results & detect output embedding & track output embedding */`
Initialization: $\mathcal{Q}_{det}$
$re_t, \mathcal{O}_{det}, \mathcal{O}_{tck} \leftarrow \texttt{IP-MOT}(\mathcal{Q}_{det} \cup \mathcal{Q}_{tck}, f_t)$

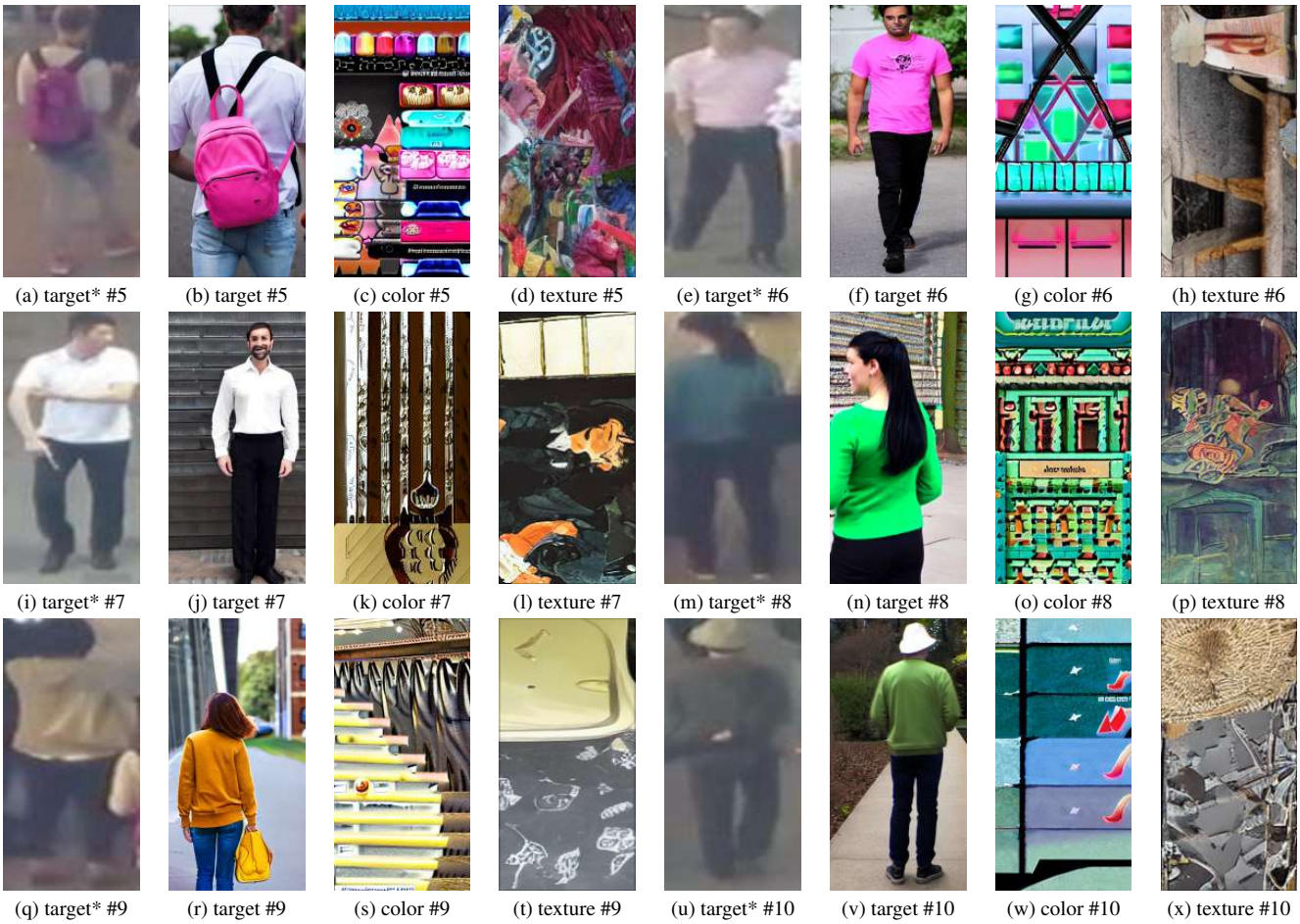
`/* add output embeddings into clip-level embedding pool */`
$\mathcal{P}_{emb} \leftarrow \mathcal{P}_{emb} \cup \mathcal{O}_{det} \cup \mathcal{O}_{tck}$

`/* filter out redundant detect output embedding and update track queries */`
$\mathcal{Q}_{tck} \leftarrow \texttt{DEM}(\mathcal{O}_{det}, \mathcal{O}_{tck})$

`/* calculate the original collective average loss and accumulate it to the new collective average loss */`
$\mathcal{L}_{clip} \leftarrow \texttt{Match}(re_t, y_t)$
$\mathcal{L}_{clip*} \leftarrow \mathcal{L}_{clip*} + \mathcal{L}_{clip}$
**end**

`/* calculate the final loss and optimize it */`
$\mathcal{L}_{clip*} \leftarrow \mathcal{L}_{clip*} + \mathcal{L}_{tri}(\mathcal{P}_{emb}) + \mathcal{L}_{i2tce}(\mathcal{P}_{emb}, \text{Text}(\texttt{B}, \texttt{Y}))$
$Optimize(\mathcal{L}_{clip*})$

Return: $\emptyset$



(a) target* #5 (b) target #5 (c) color #5 (d) texture #5 (e) target* #6 (f) target #6 (g) color #6 (h) texture #6

(i) target* #7 (j) target #7 (k) color #7 (l) texture #7 (m) target* #8 (n) target #8 (o) color #8 (p) texture #8

(q) target* #9 (r) target #9 (s) color #9 (t) texture #9 (u) target* #10 (v) target #10 (w) color #10 (x) texture #10

Figure 3. **More visualization of instance-level textual description**. In addition to the visualization in our paper,we also provide more visualization for trained TrackBook. Target* means the original target in MOT17 dataset, while target, color, and texture means corresponding synthetic image by replacing the last word in textual description with "person","color", and "texture", respectively.

ing Algorithm 1.

## B. Visualization of Deduplication Module

To better understand the interaction between detect queries and track queries, we visualize the query self-attention map in Figure 1. For the same instance, the detect query and the corresponding track query have high similarity, and there is a clear exchange of information between them, which verifies the effectiveness of our deduplication module.

## C. Visualization of Generalizability

We offer visualizations of prototypical challenging scenes in MOT, including the non-linear scene (Figure 2a) and very crowded scene (Figure 2b), to demonstrate the generalizability of the proposed IP-MOT. We observe that our approach has a strong discriminative ability for cross-domain object with severe non-linear motion and keeps high reliably associative ability in crowded scenes with dramatic occlusions.

## D. More Visualization of TrackBook

To better demonstrate the superiority of IP-MOT, we visualize more instance-level textual description from TrackBook. Figure 3 illustrates more different targets in the training set, along with the synthetic image of person, color, and texture generated based on their corresponding textual descriptions. As we can seen, most of the synthetic images accurately capture the original target's attributes, including color, texture, and high-level semantic information like clothing, hats, bags, and gender. Our promising method overcomes these constraints by efficiently generating text descriptions with robust semantics, stability, and recognizability, thereby enhancing the model's generalizability. In the future, we hope researchers will pay more attention to the insights which enhances the generalizability of model by introducing NLP presentation into multi-object tracking.