Dual-Path Temporal Decoder for End-to-End Multi-Object Tracking

Hyunseop Kim[1*] **Juheon Jeong**[1*] **Hanul Kim**[2] **Yeong Jun Koh**[1†]
[1]Chungnam National University [2]Seoul National University of Science and Technology
hyunseop95@gmail.com, jjh990427@gmail.com, hukim@seoultech.ac.kr, yjkoh@cnu.ac.kr

Abstract

We present a novel end-to-end transformer-based framework for Multiple Object Tracking (MOT) that advances temporal modeling and identity preservation. Despite recent progress in transformer-based MOT, existing methods still struggle to maintain consistent object identities across frames, especially under occlusions, appearance changes, or detection failures. We propose a dual-path temporal decoder that explicitly separates appearance adaptation and identity preservation. The appearance-adaptive decoder dynamically updates query features using current frame information, while the identity-preserving decoder freezes query features and reuses historical sampling offsets to maintain long-term temporal consistency. To further enhance stability, we introduce a confidence-guided update suppression strategy that retains previously reliable features when predictions are unreliable. Extensive experiments on MOT benchmarks demonstrate that our approach achieves state-of-the-art performance across major tracking metrics, with significant gains in association accuracy and identity consistency. Our results demonstrate the importance of decoupling dynamic appearance modeling from static identity cues, and provide a scalable foundation for robust tracking in complex scenarios. Code is available at github.com/altkddhfcjs/DualTemporalMOT

1 Introduction

Multi-object tracking (MOT) aims to consistently estimate the spatial locations and identities of multiple objects across a video sequence. As a fundamental task in computer vision, MOT is essential for a broad range of real-world scenarios, including autonomous driving [35], surveillance [29, 43], sports analytics [9, 32], and crowd analysis [23], where consistent spatio-temporal tracking is required. Early MOT studies typically have followed the tracking-by-detection paradigm [1, 3, 17, 31, 42]. These methods first detect objects in each frame and then perform association steps across frames. These methods perform association steps based on the spatial proximity between objects in consecutive frames, measured by Interaction of Unions (IoU) [3, 6] or appearance similarity using ReID embeddings [1, 24, 30, 39]. These methods have benefited from rapid advances in object detection, but they remain limited in complex scenarios involving occlusion, similar appearance, and non-linear motions.

To address these limitations, recent approaches [12, 18, 36, 40] have adopted transformer-based MOT. These methods unify the DETR [7]-based detector and tracker in an end-to-end manner. In these models, track queries are propagated from the previous frame and used to associate and track objects over time. However, transformer-based MOT still faces a key challenge in maintaining the

[*]Equal contributions.
[†]Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

temporal consistency of track query features. Although these methods refine query features over multiple decoder layers, erroneous updates can accumulate and degrade feature consistency over time, ultimately resulting in identity switches.

In this work, we propose a transformer-based MOT framework with a novel dual-path temporal decoder that explicitly addresses this issue. Each decoder layer consists of two parallel components: an appearance-adaptive decoder layer that refines query features using current frame information and an identity-preserving decoder layer that maintains temporal consistency by reusing fixed query features and historical sampling offsets from the previous frame. The identity-preserving decoder layer reuses fixed query features and historical sampling offsets to enhance temporal stability and reduce sensitivity to abrupt changes in feature representation. To further enhance robustness, we introduce a confidence-guided update suppression strategy during inference, which retains the states of low-confidence track queries instead of updating them. This mechanism alleviates identity drift and improves tracking reliability in challenging scenarios such as occlusion and detection failure. Together, these designs enable the model to balance adaptability and stability, leading to more accurate and consistent multi-object tracking across long temporal spans.

The proposed MOT achieves the state-of-the-art performance on the DanceTrack [27] and SportsMOT [9] benchmarks, demonstrating strong association ability in challenging scenarios involving diverse objects with similar appearances.

In summary, the contributions are as follows:

- We propose a dual-path temporal decoder that disentangles appearance adaptation and identity preservation. The appearance-adaptive decoder layer dynamically refines query features using current-frame information, while the identity-preserving decoder layer keeps track queries fixed and reuses historical sampling offsets to maintain temporal consistency.
- We introduce a confidence-guided update suppression strategy that prevents unreliable updates under low-confidence conditions, thereby stabilizing identity association in the presence of occlusions and detection noise.
- Our method achieves new state-of-the-art performance on the DanceTrack [27] and SportsMOT [9] benchmarks, demonstrating strong improvements in both tracking accuracy and identity preservation.

2 Related Work

Tracking-by-Detection. Tracking-by-detection remains a dominant paradigm in MOT, where per-frame object detections are temporally associated to form object trajectories. SORT [3] employs the Kalman filter and the Hungarian algorithm to associate bounding boxes based on IoU. DeepSORT [31] enhances identity stability by combining the Kalman filter with appearance-based ReID embeddings. JDE [30], FairMOT [39], and Unicorn [33] jointly optimize detection and ReID features to learn discriminative representations and achieve consistent identity preservation. Recent efforts have extended this framework by improving detection quality and robustness against association errors. BoT-SORT [1] introduces appearance fusion, motion compensation, and the improved Kalman filter to strengthen real-time performance. Transformer-based approaches such as TransMOT [8] and GTR [42] model long-range spatiotemporal dependencies to enable more structured data association. OC-SORT [6] replaces heuristic motion models with learnable predictors for improved motion estimation. Extensions such as GHOST [26] and StrongSORT [10] focus on practical deployment by addressing domain shift, embedding refinement, and inference efficiency. More recently, DeconfuseTrack [16] formulates association as a multi-stage decision problem to reduce ID switches. DiffMOT [20] formulates data association as the denoising diffusion process to jointly predict object trajectories over time.

Transformer-based MOT. Recently, the transformer-based MOT has emerged as a promising direction in end-to-end MOT. Trackformer [22] and MOTR [36], both built upon on the DETR [7] architecture, perform joint detection and tracking by propagating track queries across frames within the decoder. MeMOT [5] extends this idea by incorporating both short- and long-term memories into track embeddings, improving robustness to occlusions and appearance changes. Subsequent works have explored improvements in temporal consistency, identity preservation, and scalability. MOTRv2 [40] enhances detection recalls by integrating external YOLOX [13] detections into the

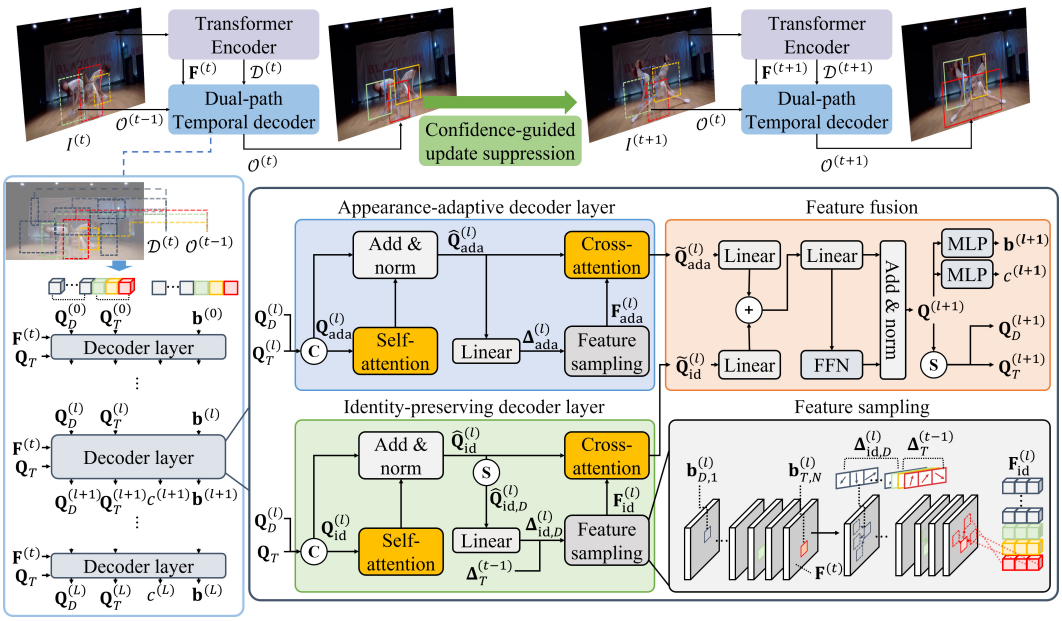

Figure 1: An overview of the proposed MOT framework. The dual-path temporal decoder consists of an appearance-adaptive decoder layer that refines query features using the current frame, and an identity-preserving decoder layer that maintains temporal consistency by freezing track query features and reusing historical sampling offsets. During inference, a confidence-guided update suppression strategy is applied to prevent unreliable feature updates. Ⓒ denotes concatenation and Ⓢ denotes feature splitting for track and candidate queries.

MOTR framework. MeMOTR [12] further strengthens identity stability under occlusion by directly injecting long-term memory into the transformer backbone. ColTrack [18] maintains identity consistency by applying self-attention between the current query and past queries along the same object's trajectory. MOTIP [11] explicitly decouples detection and association into two learnable modules, offering a more interpretable and flexible query interaction framework. However, these methods often suffer from inconsistent query updates, where inaccurate feature refinement across frames can degrade identity representations and lead to frequent identity switches.

3 Method

We aim to compose the set of tracked objects $\mathcal{O}^{(t)} = \{o_1^{(t)}, \ldots, o_N^{(t)}\}$ in each video frame $I^{(t)}$, where N is the number of tracked objects. Each tracked object $o_n^{(t)}$ includes a bounding box $\mathbf{b}_{T,n}^{(t)} \in \mathbb{R}^4$, a confidence score $c_{T,n}^{(t)} \in \mathbb{R}^1$, a query feature $\mathbf{q}_{T,n}^{(t)} \in \mathbb{R}^C$, and sampling offsets $\mathbf{\Delta}_{T,n}^{(t)} \in \mathbb{R}^{K\times 2}$, where C denotes the feature dimension and K is the number of sampling points. For each frame $I^{(t)}$, we extract a feature map $\mathbf{F}^{(t)} \in \mathbb{R}^{H\times W\times C}$, where H and W denote height and width, and a set of M detection candidates $\mathcal{D}^{(t)} = \{d_1^{(t)}, \ldots, d_M^{(t)}\}$ using a DINO [37]-based detector. Each detection candidate $d_m^{(t)}$ contains an anchor box $\mathbf{b}_{D,m}^{(t)} \in \mathbb{R}^4$ and a query feature $\mathbf{q}_{D,m}^{(t)} \in \mathbb{R}^C$. We aim to predict updated bounding boxes and confidence scores for tracked objects and detection candidates at the current frame t based on thier previous states $\mathcal{O}^{(t-1)}$ and current observations $\mathcal{D}^{(t)}$.

Figure 1 illustrates the structure of the proposed MOT framework, which integrates a dual-path temporal decoder consisting of two complementary decoding layers. An appearance-adaptive decoder layer updates query features using the current frame, enhancing localization accuracy and adaptability to appearance changes. In parallel, an identity-preserving decoder layer maintains the original track queries and reuses historical sampling offsets, thereby preserving temporal consistency and alleviating identity drift. In addition, we introduce a confidence-guided update suppression strategy, which preserves previously reliable query features by preventing updates when current predictions are unreliable.

3.1 Dual-Path temporal decoder

The dual-path temporal decoder consists of stacked layers, each including two parallel branches: an appearance-adaptive decoder layer and an identity-preserving decoder layer. The appearance-adaptive decoder layer dynamically updates query features through deformable attention and sampling, while the identity-preserving path maintains temporal identity consistency by keeping track query features fixed and reusing their historical sampling offsets. For each decoder layer l, outputs from both branches are fused to refine the object representation and estimate bounding boxes and confidence scores.

Appearance-adaptive decoder layer. The appearance-adaptive decoder layers iteratively refine query features across L decoder layers. For the initial decoder layer ($l = 0$), the track query features $\mathbf{Q}_T^{(0)} \in \mathbb{R}^{N\times C}$ are initialized from the propagated queries of the previous frame, where n-th row corresponds to a track query feature $\mathbf{q}_{T,n}^{(t-1)}$, while candidate query features $\mathbf{Q}_D^{(0)} \in \mathbb{R}^{M\times C}$ are initialized from the detection candidates, where m-th row corresponds to a candidate query feature $\mathbf{q}_{D,m}^{(t)}$. For subsequent decoder layers, both track and candidate query features, denoted as $\mathbf{Q}_T^{(l)}$ and $\mathbf{Q}_D^{(l)}$, respectively, are obtained as outputs from the previous decoder layer.

For each layer l, we concatenate these query features to form the combined query matrix $\mathbf{Q}_{\text{ada}}^{(l)} = [\mathbf{Q}_T^{(l)}; \mathbf{Q}_D^{(l)}]$. We then apply a multi-head self-attention, followed by residual connection and layer normalization to compute refined features

$$\hat{\mathbf{Q}}_{\text{ada}}^{(l)} = \text{LN}\left(\mathbf{Q}_{\text{ada}}^{(l)} + \text{SelfAttn}(\mathbf{Q}_{\text{ada}}^{(l)})\right). \tag{1}$$

We then predict the sampling offsets by applying a linear projection over the features $\hat{\mathbf{Q}}_{\text{ada}}^{(l)}$, *i.e.* $\mathbf{\Delta}_{\text{ada}}^{(l)} = \text{Linear}(\hat{\mathbf{Q}}_{\text{ada}}^{(l)}) \in \mathbb{R}^{(N+M)\times K\times 2}$, where K is the number of sampling points. These offsets guide the deformable attention to extract features $\mathbf{F}_{\text{ada}}^{(l)} \in \mathbb{R}^{(N+M)\times K\times C}$ from the image feature map $\mathbf{F}^{(t)}$, inspired by the deformable attention mechanism in deformable DETR [44]. For the feature sampling process, the bounding box of the track query at the initial decoder layer $\mathbf{b}_{T,n}^{(0)}$ is initialized from the estimated bounding box at the previous frame, $\mathbf{b}_{T,n}^{(t-1)}$. Similarly, the bounding box of each candidate query $\mathbf{b}_{D,m}^{(0)}$ is initialized from the detection candidate's bounding box $\mathbf{b}_{D,m}^{(t)}$ at the current frame. Unlike the previous transformer-based MOT methods [18, 22, 40] that compute attention weights only from the query features, our approach aggregates the sampled image features $\mathbf{F}_{\text{ada}}^{(l)}$ based on affinity between query and sampled image features. Thus, a cross-attention layer is adopted to obtain the final enhanced query features $\mathbf{Q}_{\text{ada}}^{(l)}$ at layer l:

$$\tilde{\mathbf{Q}}_{\text{ada}}^{(l)} = \text{CrossAttn}\left(\hat{\mathbf{Q}}_{\text{ada}}^{(l)}, \mathbf{F}_{\text{ada}}^{(l)}\right) \tag{2}$$

where $\hat{\mathbf{Q}}_{\text{ada}}^{(l)}$ serves as query, while $\mathbf{F}_{\text{ada}}^{(l)}$ serves as key and value in the cross-attention.

Identity-preserving decoder layer. The identity-preserving decoder layer shares the same structure as the appearance-adaptive decoder, but handles track queries differently. By keeping track queries fixed across decoder layers and reusing historical sampling offsets, it prevents the injection of unstable evidence from the current frame, thereby preserving identity-specific features and ensuring temporal consistency across frames. Specifically, the track queries $\mathbf{Q}_T \in \mathbb{R}^{N\times C}$ are initialized from the propagated queries of the previous frame as in the adaptive decoder layer, but remain fixed across decoder layers to preserve temporal identity. For each layer l, the static track queries and candidate queries $\mathbf{Q}_D^{(l)} \in \mathbb{R}^{M\times C}$ are concatenated to form $\mathbf{Q}_{\text{id}}^{(l)} = [\mathbf{Q}_T; \mathbf{Q}_D^{(l)}]$. We then apply a multi-head self-attention, followed by residual connection and layer normalization to compute refined features

$$\hat{\mathbf{Q}}_{\text{id}}^{(l)} = \text{LN}\left(\mathbf{Q}_{\text{id}}^{(l)} + \text{SelfAttn}(\mathbf{Q}_{\text{id}}^{(l)})\right). \tag{3}$$

We split these refined features $\hat{\mathbf{Q}}_{\text{id}}^{(l)}$ into track and candidate components, denoted as $\hat{\mathbf{Q}}_{\text{id},T}^{(l)}$ and $\hat{\mathbf{Q}}_{\text{id},D}^{(l)}$, respectively.

Sampling offsets for candidate queries are obtained by applying a linear projection to $\hat{\mathbf{Q}}_{\text{id},D}^{(l)}$, *i.e.*, $\mathbf{\Delta}_{\text{id},D}^{(l)} = \text{Linear}(\hat{\mathbf{Q}}_{\text{id},D}^{(l)}) \in \mathbb{R}^{M\times K\times 2}$. In contrast, the offsets for track queries are reused from their historical offsets $\mathbf{\Delta}_T^{(t-1)}$ in the previous frame $t-1$ based on the observation that the same object typically exhibits similar sampling offset patterns across adjacent frames. Reusing historical offsets promotes spatial stability in attention, maintains alignment with persistent object regions, and suppresses transient noise arising from frame-specific variations. Although the track query features $\mathbf{Q}_T$ themselves do not change across decoder layers, their corresponding bounding boxes $\{\mathbf{b}_{T,1}^{(l)}, \ldots, \mathbf{b}_{T,N}^{(l)}\}$ are iteratively refined at each layer. Deformable attention uses the updated bounding box centers as reference positions, sampling image features $\mathbf{F}_{\text{id}}^{(l)} \in \mathbb{R}^{(N+M)\times K\times C}$ from the image feature map $\mathbf{F}^{(t)}$ using $\mathbf{\Delta}_T^{(t-1)}$ and $\mathbf{\Delta}_{\text{id},D}^{(l)}$. A cross-attention layer is then applied to obtain enhanced query features at layer l:

$$\tilde{\mathbf{Q}}_{\text{id}}^{(l)} = \text{CrossAttn}\left(\hat{\mathbf{Q}}_{\text{id}}^{(l)}, \mathbf{F}_{\text{id}}^{(l)}\right), \tag{4}$$

where queries are $\hat{\mathbf{Q}}_{\text{id}}^{(l)}$, and keys and values are sampled features $\mathbf{F}_{\text{id}}^{(l)}$.

Feature fusion and state prediction. For each decoder layer l, we fuse the outputs from the appearance-adaptive and identity-preserving decoder layers to obtain refined representations that incorporate both dynamic appearance changes and stable identity information. Specifically, given the enhanced query features from the appearance-adaptive path $\tilde{\mathbf{Q}}_{\text{ada}}^{(l)}$ and from the identity-preserving path $\tilde{\mathbf{Q}}_{\text{id}}^{(l)}$, we combine the corresponding track and candidate features from each path separately using linear projections, resulting in the fused track and candidate query features as follows:

$$\mathbf{Q}^{(l+1)} = \text{Linear}\left(\text{Linear}(\tilde{\mathbf{Q}}_{\text{ada}}^{(l)}) + \text{Linear}(\tilde{\mathbf{Q}}_{\text{id}}^{(l)})\right). \tag{5}$$

The fused features are further refined using a feed-forward network (FFN) with a residual connection and layer normalization. Then, $\mathbf{Q}^{(l+1)}$ is split into $\tilde{\mathbf{Q}}_T^{(l+1)}$ and $\tilde{\mathbf{Q}}_D^{(l+1)}$, which are used for the next layer $l+1$. Here, $\tilde{\mathbf{Q}}_D^{(l+1)}$ is used for the appearance-adaptive and identity-preserving decoder layers, while $\tilde{\mathbf{Q}}_T^{(l+1)}$ is used for the appearance-adaptive layer only.

The refined fused query features $\mathbf{Q}^{(l+1)}$ are further used to predict bounding boxes and corresponding confidence scores for each object. Specifically, the bounding boxes $\mathbf{b}^{(l+1)} \in \mathbb{R}^{(N+M)\times 4}$ and confidence scores $c^{(l+1)} \in \mathbb{R}^{(N+M)\times 1}$ are obtained by passing $\mathbf{Q}^{(l+1)}$ through two separate multi-layer perceptrons (MLPs), each consisting of two linear layers with an activation function in between. These predictions serve as the updated object states at decoder layer $l+1$, facilitating accurate tracking and identification of objects across frames. The predicted bounding boxes and confidence scores are subsequently split into track and candidate components ($\mathbf{b}_T^{(l+1)}, c_T^{(l+1)}$) and ($\mathbf{b}_D^{(l+1)}, c_D^{(l+1)}$), respectively, for further sampling processing in subsequent decoding layers.

At the last layer L, the final tracking decisions for frame t are determined based on confidence thresholding. Tracked objects whose predicted confidence scores $c_T^{(L)}$ are larger than a threshold α are regarded as active tracks, otherwise, they are marked as lost. Similarly, candidate objects with confidence scores $c_D^{(L)} > \alpha$ are determined as new tracks. Tracks that remain lost for more than τ consecutive frames are terminated. Finally, sampling offsets $\mathbf{\Delta}_T^{(t)}$ are extracted from the entries in $\Delta_{\text{ada}}^{(L)}$, corresponding to both the track queries and the selected candidate queries, and are reused in the identity-preserving decoder at frame $t+1$.

3.2 Confidence-guided update suppression

While the confidence score determines whether an object is considered tracked or lost, conventional transformer-based tracking frameworks [18, 22, 36] continue to update query features of all track objects at every frame, regardless of their confidence. That is, even after an object is marked as lost, its query feature continues to be updated through self-attention and cross-attention. As a result, noisy observations from unreliable predictions are incorporated into the query representation. This can be problematic, since the updated query feature is subsequently used to predict the object's bounding box

and confidence score in the next frame. As a result, this unconditional update strategy accumulates noisy query features over time, which can cause identity drift or misassociation with nearby objects.

To address this issue, we propose a confidence-guided update suppression strategy. For objects with low confidence, we do not update their query features unless their predicted confidence exceeds a predefined threshold β. Instead, we preserve their previously reliable representations from earlier frames where their confidence was sufficiently high. This selective update mechanism suppresses the accumulation of noisy predictions from uncertain objects, thereby preserving clean identity embeddings and improving long-term tracking stability.

3.3 Training

We train our model in an end-to-end manner using a bipartite matching objective [7], as in previous transformer-based MOT frameworks [18, 36, 40]. Given the predicted tracked objects $\mathcal{O}^{(t)}$ and ground-truth set, we first compute the optimal assignment using Hungarian matching. The training loss is then computed over matched pairs using a weighted sum of a classification loss, a bounding box regression loss, and a generalized IoU loss, following standard practice.

4 Experiments

4.1 Datasets & Metrics

DanceTrack [27]. It is a multi-human tracking dataset in dancing scenes with similar uniform appearance and diverse motion, requiring strong association under occlusion and ambiguity. DanceTrack contains 40, 25, and 35 videos for training, validation, and test sets.

SportsMOT [9]. It is a recently released multi-object tracking dataset that focuses on athlete tracking in fast-paced sports such as soccer, basketball, and volleyball. The SportsMOT dataset presents significant challenges for motion modeling due to frequent acceleration and abrupt direction changes in these scenes. The dataset consists of 45, 45, and 150 sports sequences for training, validation, and test sets, respectively.

MOT17 [23]. It is a widely used pedestrian tracking dataset. MOT17 mainly contains massive pedestrians with simple and linear motions. It contains 7 training sequences and 7 test sequences. The sequences contain 500-1500 frames, recorded and annotated at 25-30 FPS.

Metrics. We assess the performance of the proposed method based on diverse MOT metrics for comparisons with other methods. Higher order tracking accuracy (HOTA) [19] is used as the primary metric, as it provides a balanced assessment of detection and association performance. To further dissect this trade-off, we report detection accuracy (DetA) and association accuracy (AssA), which decompose HOTA into its constituent factors. We also include ID F1 score (IDF1) [25], which measures the alignment between predicted and ground-truth identities, and multi-object tracking accuracy (MOTA) [2], a conventional metric that emphasizes detection errors, including false positives, false negatives, and ID switches.

4.2 Implementation Details

MOT Network. The proposed framework is built on DINO [37] that uses ResNet-50 [14] backbone and transformer-based encoder. We select the top-$M = 300$ detection candidates from the encoder in DINO as anchor boxes and extract a candidate query feature $\mathbf{q}_{D,m}^{(t)}$ for each candidate by combining its learnable query embedding and positional embedding, following [37]. We set the number of dual-path temporal decoder layers to $L = 6$, a feature dimension to $C = 256$, and sampling points to $K = 256$. The confidence threshold α, the suppression threshold β, and τ are set to 0.6, 0.4, and 60, respectively.

Training. As in the prior works [18, 22, 36], we perform a two-stage training strategy. In the first stage, the object detector is trained for 40 epochs. In the second stage, the backbone and encoder are frozen, and only the dual-path temporal decoder is trained. The input images are resized to a resolution of 1440×800. The proposed MOT framework employs multi-scale training, Mosaic [4], and MixUp [15] for data augmentation. We use the AdamW optimizer with a learning rate of 1×10^{-4}

Table 1: Quantitative comparison on the DanceTrack [27] test set. The performance *with validation data* presents that the validation set is also included during training. The best results are boldfaced.

Methods	HOTA	DetA	AssA	MOTA	IDF1
w/o valid data:					
CenterTrack [41]	41.8	78.1	22.6	86.8	35.7
TransTrack [28]	45.5	75.9	27.5	88.4	45.2
ByteTrack [38]	47.7	71.0	32.1	89.6	53.9
QDTrack [24]	54.2	80.1	36.8	87.7	50.4
MOTR [36]	54.2	73.5	40.2	79.7	51.5
OC-SORT [6]	55.1	80.3	38.3	92.0	54.6
DiffMOT [20]	62.3	82.5	47.2	**92.8**	63.0
MeMOTR [12]	68.5	80.5	58.4	89.9	71.2
CO-MOT [34]	69.4	82.1	58.9	91.2	71.9
MOTRv2 [40]	69.9	83.0	59.0	91.9	71.7
MOTIP [11]	72.0	81.8	63.5	91.9	76.8
ColTrack [18]	72.6	-	62.3	92.1	74.0
Ours	**74.1**	**83.9**	**65.6**	92.5	**78.6**
with valid data:					
MOTRv2 [40]	73.4	83.7	64.4	92.1	76.0
ColTrack [18]	75.3	-	66.9	92.2	77.3
Ours	**76.2**	**85.0**	**68.3**	**92.5**	**79.9**

Table 2: Quantitative comparison on the SportMOT [9] test set. The best results are boldfaced.

Methods	HOTA	DetA	AssA	MOTA	IDF1
QDTrack [24]	60.4	77.5	47.2	90.1	62.3
CenterTrack [41]	62.3	82.1	48.0	90.8	60.0
ByteTrack [38]	62.8	77.1	51.2	94.1	69.8
TrackFormer [22]	63.3	66.0	61.1	74.1	72.4
BoT-SORT [1]	68.7	84.4	55.9	**94.5**	70.0
MeMOTR [12]	68.8	82.0	57.8	90.2	69.9
TransTrack [28]	68.9	82.7	57.5	92.6	71.5
ColTrack [18]	71.5	80.5	63.6	89.4	74.6
OC-SORT [6]	71.9	**86.4**	59.8	**94.5**	72.2
DiffMOT [20]	72.1	86.0	60.5	**94.5**	72.8
MOTIP [11]	72.6	83.5	63.2	92.4	77.1
Ours	**73.9**	82.2	**66.6**	91.5	**78.7**

and a weight decay of 1×10^{-4}. The learning rate is decayed by a factor of 0.1 during the final 15 training epochs. The model is trained for 45, 45 and 65 epochs on DanceTrack [27], SportsMOT [9] and MOT17 [23], respectively. All experiments are conducted on 8 NVIDIA RTX 4090 Ti GPUs with a batch size of 1, where each batch consists of a 4-frame video clip.

4.3 Benchmark Evaluation

DanceTrack. Table 1 shows the comparison of the proposed method with the existing methods on the test set in DanceTrack [27]. The proposed MOT achieves a HOTA score of 74.1 and achieves state-of-the-art performance across all metrics. In particular, compared to the previous best method ColTrack [18], it exhibits significant improvements in association accuracy, with AssA increasing from 62.3 to 65.6 and IDF1 increasing from 74.0 to 78.6. These results demonstrate the effectiveness of our temporal modeling in maintaining consistent object identities over time. Even when compared to prior works [18, 40] trained on both training and validation sets, our model consistently outperforms all competitors, demonstrating the effectiveness and robustness of our approach.

Table 3: Quantitative comparison on MOT17 [23] test set. The best results are boldfaced.

Methods	**HOTA**	**DetA**	**AssA**	**MOTA**	**IDF1**
Heuristic:					
OC-SORT [6]	63.2	-	63.2	78.0	77.5
ByteTrack [38]	63.1	64.5	62.0	80.3	77.3
BoT-SORT [1]	64.6	-	-	80.6	79.5
MixSort-OC [9]	63.4	63.8	63.2	78.9	77.8
MixSort-Byte [9]	64.0	64.1	64.2	79.3	78.7
Deep OC-SORT [21]	64.9	-	**65.9**	79.4	**80.6**
DeconfuseTrack [16]	**64.9**	**65.0**	65.1	**80.4**	**80.6**
End-to-end:					
MOTR [36]	57.8	60.3	55.7	73.4	68.6
MeMOTR [36]	56.9	58.9	55.8	72.5	69.0
TransTrack [28]	54.1	61.6	47.9	74.5	63.9
MOTRv2 [40]	**62.0**	**63.8**	60.6	78.6	75.0
TrackFormer [22]	-	-	-	74.1	68.0
ColTrack [18]	61.0	-	-	**78.8**	73.9
MOTIP [11]	59.3	62.0	57.0	75.3	71.3
Ours	61.5	60.8	**62.5**	73.8	**75.1**

Table 4: Ablation studies for the identity-preserving decoder layer (IDL) on the DanceTrack [27] validation set. The best results are boldfaced.

Method	HOTA	DetA	AssA	MOTA	IDF1
without IDL	66.7	76.1	56.3	87.0	69.5
IDL with varying offsets $\mathbf{\Delta}_{\mathrm{id},T}^{(l)}$	67.5	77.5	59.0	87.1	73.2
IDL with static historical offsets $\mathbf{\Delta}_{T}^{(t-1)}$	**69.1**	**77.8**	**61.6**	**87.5**	**74.9**

SportsMOT. Table 2 lists the performance on the SportMOT [32] test set. The proposed MOT achieves 73.9 HOTA, surpassing the previous state-of-the-art MOTIP [11] by margins of 1.3. Also, ours significantly improves the IDF1 score by 5.9 over DiffMOT [20], demonstrating superior association accuracy. DiffMOT, as a tracking-by-detection method reliant on pretrained detectors [13], excels on detection-centric metrics (*e.g.* MOTA, DetA) but underperforms on association-focused metrics such as IDF1.

MOT17. Table 3 presents the results on the MOT17 [23] test set. The proposed method achieves the best AssA and IDF1 performance among end-to-end approaches. It indicates that the proposed method maintains stable associations and is robust against ID switches. Despite our lower detection accuracy (MOTA 73.8) than ColTrack (MOTA 78.8) and MOTIP (MOTA 75.3), our stronger association capability enables higher HOTA and lower IDF1 than them, narrowing the gap to heuristic-augmented pipelines. Compared to MOTRv2, which uses heuristic post-processing, our method achieves superior AssA and comparable IDF1 while remaining fully end-to-end.

4.4 Ablation Study

We conduct ablation studies on the validation set of DanceTrack [27] to evaluate the effectiveness of the proposed components, including the identity-preserving decoder layer and the confidence-guided update suppression. In addition, We analyze the performance under various detection settings to further validate the robustness of the proposed framework.

Identity-preserving decoder layer. Table 4 reports the ablation study on the identity-preserving decoder layer (IDL). We evaluate three model variants to analyze its impact. First, we remove the IDL from the dual-path temporal decoder, reducing it to a single-path structure that consists of the appearance-adaptive decoder only. Second, we include the IDL but replace the historical sampling

Table 5: Ablation studies for the confidence-guided update suppression on DanceTrack [27] validation set. The best results are boldfaced.

Method	β	HOTA	DetA	AssA	MOTA	IDF1
without confidence-guided update suppression	-	67.9	**78.1**	59.2	**87.6**	72.9
with confidence-guided update suppression	0.2	68.2	77.6	60.2	87.3	73.8
	0.4	**69.1**	77.8	**61.6**	87.5	**74.9**
	0.6	68.7	77.9	60.7	**87.6**	74.2

Table 6: Comparison of the proposed method with other methods using various detectors on the DanceTrack [27] validation set. The best results are boldfaced.

Detector	mAP	Tracker	HOTA	DetA	AssA	MOTA	IDF1
YOLOX	72.1	MOTRv2 [40]	64.5	**78.7**	53.0	–	–
		Ours	**67.9**	77.2	**60.0**	**87.2**	**73.3**
Deformable DETR	63.7	MOTIP [11]	62.2	75.3	51.5	85.2	64.8
		Ours	**66.4**	**77.1**	**57.3**	85.9	**70.6**
DINO	73.1	ColTrack [18]	61.9	–	–	86.5	61.6
		Ours	**69.1**	**77.8**	**61.6**	**87.5**	**74.9**

offsets $\mathbf{\Delta}_T^{(t-1)}$ with predicted offsets $\mathbf{\Delta}_{\text{id},T}^{(l)}$, obtained by applying a linear projection to $\hat{\mathbf{Q}}_{\text{id},T}^{(l)}$. Finally, the full model includes the IDL with historical sampling offsets.

We observe that incorporating the identity-preserving decoder layer leads to consistent performance improvements over using only the appearance-adaptive decoder. Specifically, the variant with fixed track query features $\mathbf{Q}_T$ achieves notable gains, improving HOTA from 66.7 to 67.5 and IDF1 from 69.5 to 73.2. These results indicate that maintaining stable query features across decoder layers strengthens identity association and reduces drift. Furthermore, augmenting this with historical sampling offsets $\mathbf{\Delta}_T^{(t-1)}$ yields the best performance, achieving 69.1 HOTA and 74.9 IDF1. This underscores the importance of both feature consistency and temporally coherent attention for robust identity preservation over time.

Confidence-guided update suppression. Table 5 shows an ablation study to evaluate the impact of the confidence-guided update suppression strategy. Without this suppression, the model achieves 67.9 HOTA and 72.9 IDF1. In contrast, enabling the strategy consistently improves performance. Setting the threshold to $\beta = 0.4$ yields the best results, improving HOTA by 1.2 and IDF1 by 2.0, while also increasing AssA from 59.2 to 61.6. These results indicate that selectively retaining previously reliable query features under low-confidence conditions stabilizes identity association and reduces identity drift.

Analysis under various object detectors. To validate the generalization capability of the proposed dual-path decoder, we reproduced experiments on the DanceTrack validation using the same detectors adopted by prior methods. Table 6 shows the comparison of our model with previous transformer-based MOT under three detectors: YOLOX [13], Deformable DETR [44], and DINO [37], which are used in MOTRv2 [40], MOTIP [11], and ColTrack [18], respectively. The proposed method outperforms other transformer-based MOT methods for all detectors with significant HOTA improvements. These results demonstrate that the proposed dual-path decoder and stable query propagation consistently enhance association accuracy across all detector settings.

Tracking results and sampling locations according to IDL. Figure 2 illustrates a layer-wise comparison of tracking results and sampling locations from the appearance-adaptive decoder layer (ADL), with and without the identity-preserving decoder layer (IDL). For each setting, we visualize the predicted bounding boxes and the top 50 sampling points (based on attention weights) for multiple decoder layers ($l = 1, 2, 5, 6$) at the current frame t. Without IDL, the model fails to preserve the identity of the target object (green box) from frame $t - 1$, resulting in a misaligned bounding box

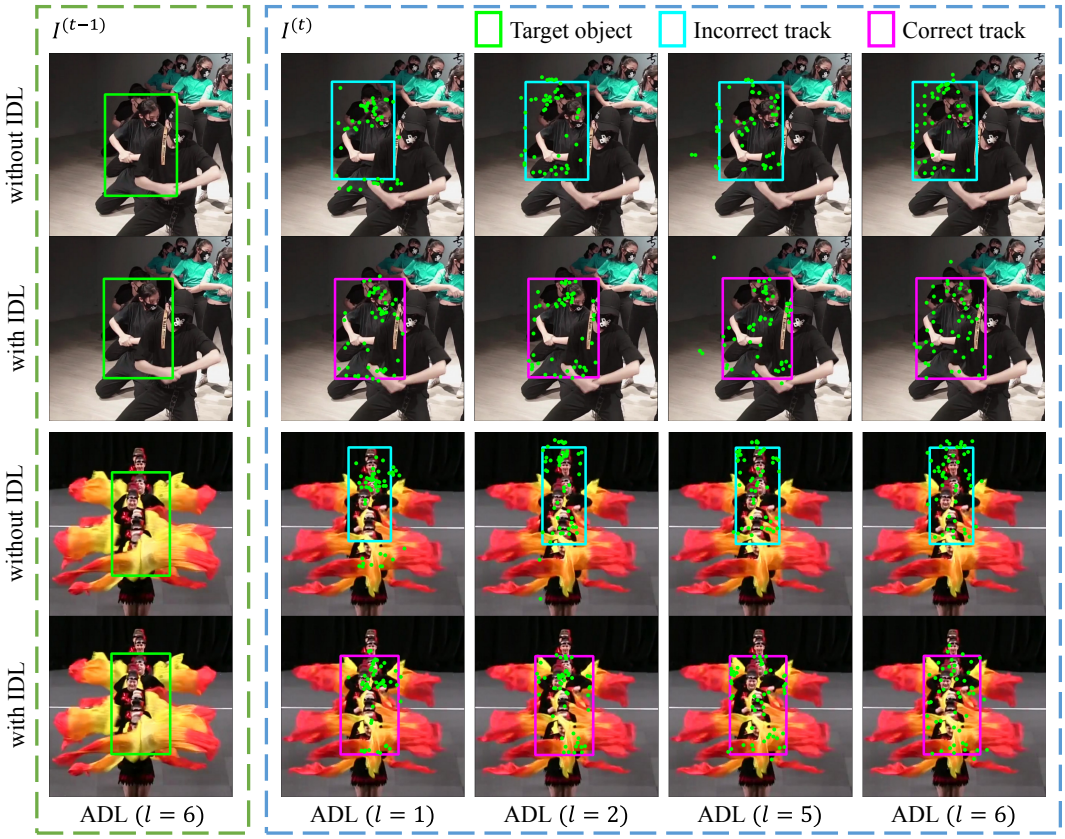

Figure 2: Visualization of tracking results and sampling locations in the appearance-adaptive decoder layer (ADL), with and without the identity-preserving decoder layer (IDL). The green box indicates the target object from frame $t-1$. The cyan box represents an incorrectly tracked result, corresponding to an identity switch (IDSW), while the magenta box denotes the correctly tracked object. Green dots indicate the top 50 sampling locations with the highest attention weights.

at frame t that corresponds to a different object (cyan box). The associated sampling points also shift toward this incorrect object, yielding a drift in query attention. In contrast, with IDL, the model consistently tracks the correct target object (magenta box) across layers, and the sampling points remain localized around the intended target. These observations demonstrate that IDL, by reusing static queries from the previous frame, enhances temporal stability and guides the decoder to maintain focus on the correct object during the iterative processes.

5 Conclusions

We introduced a transformer-based MOT framework that explicitly addresses the challenges of maintaining temporal consistency and robust identity association in complex scenes. At the core of our approach is a dual-path temporal decoder that decouples appearance adaptation from identity preservation, enabling the model to refine object representations while safeguarding identity-specific information from frame-specific noise. Additionally, we proposed a confidence-guided update suppression mechanism that further stabilizes tracking by selectively retaining reliable features under unreliable predictions. Through extensive experiments on DanceTrack and SportsMOT, our method consistently outperforms existing approaches in both detection and association metrics, establishing new state-of-the-art results on two benchmarks. The significant improvements in IDF1 and HOTA demonstrate the effectiveness of our temporal modeling strategy. We believe this work provides a strong foundation for further advancements in end-to-end multi-object tracking and highlights the importance of disentangling temporal dynamics and identity stability in transformer-based architectures.

Acknowledgements. This work was partly by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (No. RS-2024-00397293, No. RS-2024-00352566, No. RS-2025-00559165), and partly by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2022-00155857, Artificial Intelligence Convergence Innovation Human Resources Development (Chungnam National University))

References

[1] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022.

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

[3] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. Ieee, 2016.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[5] J. Cai, M. Xu, W. Li, Y. Xiong, W. Xia, Z. Tu, and S. Soatto. Memot: Multi-object tracking with memory. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8090–8100, 2022.

[6] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9686–9696, 2023.

[7] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-toend object detection with transformers. in eccv. *Springer*, 1(2):4, 2020.

[8] P. Chu, J. Wang, Q. You, H. Ling, and Z. Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. In *Proceedings of the IEEE/CVF Winter Conference on applications of computer vision*, pages 4870–4880, 2023.

[9] Y. Cui, C. Zeng, X. Zhao, Y. Yang, G. Wu, and L. Wang. Sportsmot: A large multi-object tracking dataset in multiple sports scenes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9921–9931, 2023.

[10] Y. Du, Z. Zhao, Y. Song, Y. Zhao, F. Su, T. Gong, and H. Meng. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 25:8725–8737, 2023.

[11] R. Gao, J. Qi, and L. Wang. Multiple object tracking as id prediction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27883–27893, 2025.

[12] R. Gao and L. 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.

[13] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Y. N. D. D. L.-P. Hongyi Zhang, Moustapha Cisse. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations*, 2018.

[16] C. Huang, S. Han, M. He, W. Zheng, and Y. Wei. Deconfusetrack: Dealing with confusion for multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19290–19299, 2024.

[17] H. Kim, H.-J. Lee, Y. Lee, J. Lee, H. Kim, and Y. J. Koh. Grae-3dmot: Geometry relation-aware encoder for online 3d multi-object tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11697–11706, 2025.

[18] Y. Liu, J. Wu, and Y. Fu. Collaborative tracking learning for frame-rate-insensitive multi-object tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9964–9973, 2023.

[19] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129:548–578, 2021.

[20] W. Lv, Y. Huang, N. Zhang, R.-S. Lin, M. Han, and D. Zeng. Diffmot: A real-time diffusion-based multiple object tracker with non-linear prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19321–19330, 2024.

[21] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani. Deep oc-sort: Multi-pedestrian tracking by adaptive re-identification. In *2023 IEEE International conference on image processing (ICIP)*, pages 3025–3029. IEEE, 2023.

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

[23] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[24] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, and F. 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.

[25] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision*, pages 17–35. Springer, 2016.

[26] J. Seidenschwarz, G. Brasó, V. C. Serrano, I. Elezi, and L. Leal-Taixé. Simple cues lead to a strong multi-object tracker. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13813–13823, 2023.

[27] P. Sun, J. Cao, Y. Jiang, Z. Yuan, S. Bai, K. Kitani, and P. Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20993–21002, 2022.

[28] P. Sun, J. Cao, Y. Jiang, R. Zhang, E. Xie, Z. Yuan, C. Wang, and P. Luo. Transtrack: Multiple object tracking with transformer. arxiv 2020. *arXiv preprint arXiv:2012.15460*, 2, 2012.

[29] Z. Tang, M. Naphade, M.-Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, and J.-N. Hwang. Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8797–8806, 2019.

[30] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang. Towards real-time multi-object tracking. In *European conference on computer vision*, pages 107–122. Springer, 2020.

[31] N. Wojke, A. Bewley, and D. 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.

[32] T. Wu, R. He, G. Wu, and L. Wang. Sportshhi: A dataset for human-human interaction detection in sports videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18537–18546, 2024.

[33] B. Yan, Y. Jiang, P. Sun, D. Wang, Z. Yuan, P. Luo, and H. Lu. Towards grand unification of object tracking. In *European conference on computer vision*, pages 733–751. Springer, 2022.

[34] F. Yan, W. Luo, Y. Zhong, Y. Gan, and L. Ma. Bridging the gap between end-to-end and non-end-to-end multi-object tracking. *arXiv preprint arXiv:2305.12724*, 2023.

[35] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[36] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Zhang, and Y. Wei. Motr: End-to-end multiple-object tracking with transformer. In *European conference on computer vision*, pages 659–675. Springer, 2022.

[37] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

[38] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European conference on computer vision*, pages 1–21. Springer, 2022.

[39] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International journal of computer vision*, 129:3069–3087, 2021.

[40] Y. Zhang, T. Wang, and X. 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.

[41] X. Zhou, V. Koltun, and P. Krähenbühl. Tracking objects as points. In *European conference on computer vision*, pages 474–490. Springer, 2020.

[42] X. Zhou, T. Yin, V. Koltun, and P. Krähenbühl. Global tracking transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8771–8780, 2022.

[43] P. Zhu, L. Wen, X. Bian, H. Ling, and Q. Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018.

[44] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. arxiv 2020. *arXiv preprint arXiv:2010.04159*, 3, 2010.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction of this paper clearly identify the challenges in existing MOT methods and appropriately present the proposed solutions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are explicitly stated in Section 5, Conclusion and Limitations.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We conducted both quantitative and qualitative ablation studies for all assumptions.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The experimental environment and training configurations are described in Section 4.2 Implementation Details of the main text.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code and result files have been made publicly available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental environment and training configurations are described in Section 4.2 Implementation Details of the main text.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We conducted experiments on multiple datasets and reported the results from the official evaluation servers.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: It is specified in Section 4.2 Implementation Details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research was conducted in accordance with the Code of Ethics, using publicly available datasets.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: It is appropriately described in the Introduction section.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not use any data or models that pose a high risk of misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The assets used in the paper are properly cited with appropriate references.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: All newly introduced assets, including the model code and checkpoints, are released with accompanying documentation covering training details, usage instructions, license information, and limitations. This ensures clarity and reproducibility.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This work does not involve human subjects or crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This work does not involve human subjects or crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.