HFF-Tracker: A Hierarchical Fine-grained Fusion Tracker for Referring Multi-Object Tracking

Zeyong Zhao[1], Yanchao Hao[1*], Minghao Zhang[1], Qingbin Liu[1], Bo Li[1], Dianbo Sui[2], Shizhu He[3], Xi Chen[1*]

[1]Platform and Content Group, Tencent, Beijing 100080, China
[2]Harbin Institute of Technology, Harbin 150001, China
[3]The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences., Beijing 101400, China
zc_1413@163.com, {marshao, hellomzhang, qingbinliu, ryanbli, jasonxchen}@tencent.com, suidianbo@hit.edu.cn, shizhu.he@nlpr.ia.ac.cn

Abstract

Referring Multi-Object Tracking (RMOT) aims to track multiple objects based on a provided language expression. Although prior studies have sought to accomplish this by integrating an textual module into the multi-object tracker, these methods combine text and image features in a basic way, neglecting the importance of text features. In this study, we propose a Hierarchical Fine-grained text-image Fusion tracker, named HFF-Tracker, which can perform fine-grained fusion of pixel-level visual features and text features across various semantic levels. Specifically, we have devised a Hierarchical Multi-Modal Fusion (HMMF) module to merge text and image features at an early stage in a hierarchical and detailed manner. The Text-Guided Decoder (TGD) is designed to provide the query with prior semantic information during the decoding process. Additionally, we have crafted a Text-Guided Prediction Head (TGPH) that utilizes text information to enhance the performance of the prediction head. Furthermore, we have implemented an adaptive Look-Back training strategy to maximize the utilization of valuable labeled data. Extensive experiments on the Refer-KITTI dataset and the Refer-KITTI-V2 dataset demonstrate that our proposed HFF-Tracker outperforms other state-of-the-art methods with remarkable margins.

Introduction

Referring Multi-Object Tracking (RMOT) (Wu et al. 2023a) is a newly proposed task that addresses the limitations of traditional Multi-Object Tracking (MOT) tasks, particularly in terms of flexibility and generalization. It aims to detect and track all specific objects that match a given reference description frame by frame, while maintaining a unique identifier for each object. Although it is an emerging task, RMOT has gained widespread attention due to its potential advantages in various applications, including but not limited to autonomous driving (Liao, Li, and Ye 2024), human-computer interaction (Zhang et al. 2024), and animal surveys (Jiao et al. 2023).

Despite the swift advancement of RMOT algorithms over the past two years, current methods continue to grapple with several challenges. Primarily, the fusion of image and text features is frequently basic, suggesting that the information derived from text prompts is not being fully leveraged. This is crucial, as the data from text prompts is fundamental to the RMOT task and directly impacts the ultimate tracking results. Furthermore, insufficient analysis and utilization of RMOT datasets can lead to ineffective use of valuable labeled data during training. Consequently, the robustness and performance of the methods often fail to meet expectations.

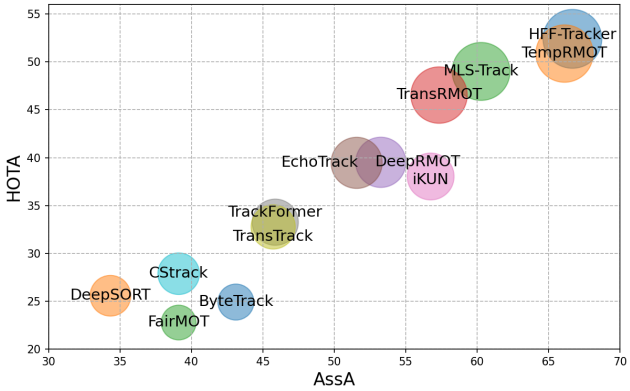


Figure 1: HOTA-AssA-DetA comparisons of different trackers on the Refer-KITTI, and the radius of circle is DetA. Our HFF-Tracker achieves 52.41% HOTA, 66.65% AssA and 41.29% DetA, outperforming all previous trackers.

Inspired by these challenges, we propose a Hierarchical Fine-grained text-image Fusion framework, HFF-Tracker, based on TransRMOT. This framework integrates text and visual information at the early fusion stage, queries, and prediction head. Specifically, we propose a Hierarchical Multi-Modal Fusion module (HMMF) for fine-grained fusion of pixel-level visual and text features across semantic levels, generating expressive visual-textual features. We also design a Text-Guild Decoder (TGD) that uses text features to guide query inputs, enhancing target identification and tracking. Furthermore, we integrate text semantic information into the visual features of queries in the prediction head's referring branch, named Text-Guide Predict Head (TGPH), improv-

*Corresponding authors

ing cross-modal data learning. We also introduce a Look-Back training strategy to enhance data utilization, first using a Look-Back-Hard approach to mine challenging examples, then a Look-Back-Remain method to leverage all annotated data, improving model accuracy and robustness. As a result, HFF-Tracker ranks 1st on Refer-KITTI, outperforming previous trackers, as shown in Figure 1.

Our key contributions are three-fold: (i) We introduce the HFF-Tracker framework, which efficiently integrates textual and visual features at pixel, word, and sentence levels using HMMF, TGD, and TGPH, enhancing RMOT performance; (ii) We propose a Look-Back training strategy, which improves model performance by adaptively identifying challenging examples and maximizing labeled data usage; (iii) We evaluate our method on Refer-KITTI and Refer-KITTI-V2 RMOT benchmarks, and our proposed HFF-Tracker outperforms previous state-of-the-art methods with large margins.

Related Work

RMOT Datasets

The introduction of numerous datasets has also significantly facilitated the advancement of referring multi-object tracking algorithms. Initially, Wu et al. introduce the RMOT task and construct Refer-KITTI dataset(Wu et al. 2023a), which laid the groundwork for the swift advancement of RMOT. The Refer-KITTI dataset, which is relabeled based on the public KITTI dataset (Geiger, Lenz, and Urtasun 2012), selects 18 videos for text description, resulting in a total of 818 textual expressions. To encompass a broad range of scenarios, GroOT (Nguyen et al. 2024) is developed using official videos and bounding box annotations from MOT17 (Milan et al. 2016), TAO (Dave et al. 2020), and MOT20 (Dendorfer et al. 2020). The construction of the NuPrompt (Wu et al. 2023b) dataset addresses the shortage of data in RMOT tasks within 3D scenes. iKUN (Du et al. 2024) further introduces Refer-Dance dataset, which extends upon common multi-object tracking dataset DanceTrack (Sun et al. 2022). Despite being a large dataset with 65 videos and 1,985 textual expressions, it has only 25 unique words, and 94.86% of expressions are null (not containing any targets). In order to reduce the cost of annotation, (Ma et al. 2024) develops the Refer-UE-City dataset. This dataset is created using Unreal Engine 5 (UE5) to construct a virtual world and simulate pedestrian and vehicular traffic through internal components. Recently, Zhang et al. enhances the Refer-KITTI dataset by incorporating 21 videos instead of the original 18 from KITTI, and by supplementing it with rich expressions. The number of expressions in Refer-KITTI subsequently increases from 895 to 9,758, leading to the creation of an improved dataset named Refer-KITTI-v2 (Zhang et al. 2024).

Significant RMOT advancements have been made through 2D scene-focused benchmarks like Refer-KITTI, GroOT, Refer-Dance, Refer-UE-City, and Refer-KITTI-V2. The 3D-focused NuPrompt dataset further broadens RMOT's potential in autonomous driving. As the first and most utilized RMOT dataset, we selected Refer-KITTI for benchmarking and ablation studies, facilitating comparison with other algorithms. We also compared performance with the state-of-the-art algorithm on the more challenging Refer-KITTI-V2 dataset.

RMOT Algorithms

The main challenge of referring multi-object tracking (RMOT) is modeling the semantic alignment of cross-modal sources, such as vision and language, and handling temporal issues like object occlusion. Recent state-of-the-art solutions, leveraging the Transformer's (Vaswani et al. 2017) flexibility and robust contextual comprehension, predominantly adopt the tracking-by-query paradigm. They integrate a visual-text fusion component into an existing MOT framework, enabling object tracking via textual descriptions. For instance, TransRMOT (Wu et al. 2023a) improves upon the end-to-end method MOTR (Zeng et al. 2022) by adding an early text-visual fusion module for cross-modal input. PromptTrack (Wu et al. 2023b) modifies the query-based method PF-Track (Pang et al. 2023) to accommodate prompt input and introduces a new prompt reasoning branch. MLS-Track (Ma et al. 2024) emphasizes cross-modal semantic understanding by integrating semantic information layer by layer. TempRMOT (Zhang et al. 2024) considers the temporal relationship between frames, introducing a query-based temporal enhancement module. Unlike these, iKUN (Du et al. 2024) follows a two-stage paradigm, first extracting object tracklets using an existing tracker, then selecting those matching the language expression.

Despite significant advancements in existing RMOT methods, they still have shortcomings. TransRMOT and TempRMOT only design a simple cross-attention module between the backbone and encoder, merging text and image features, but they do not fully utilize text prompt information. Although MLS-Track improves the text-image fusion strategy, it still operates in a coarse-grained fusion mode and cannot correctly align image-text features. iKUN, a two-stage method, allows for more thorough refinement and filtering of potential candidates. However, its architecture is typically complex, with higher computational demands. Moreover, the performance of iKUN heavily relies on the off-the-shelf tracker used in step 1, and they can only recognize categories that the tracker can identify, which deviates from the core objective of the RMOT task. Therefore, in this work, we propose a robust end-to-end RMOT algorithm, named HFF-Tracker, which performs fine-grained fusion of pixel-level visual features and text features across different semantic levels. Additionally, we design an adaptive Look-Back training strategy to further utilize valuable labeled data.

Method

Framework Overview

The overview of Hierarchical Fine-grained text-image Fusion framework (HFF-Tracker) is depicted in Figure 2. Building upon the TransRMOT, HFF-Tracker enhances its capabilities by introducing the Hierarchical Multi-Modal Fusion module, Text-Guide Decoder, and Text-Guide Predict Head. Additionally, drawing inspiration from Tem-

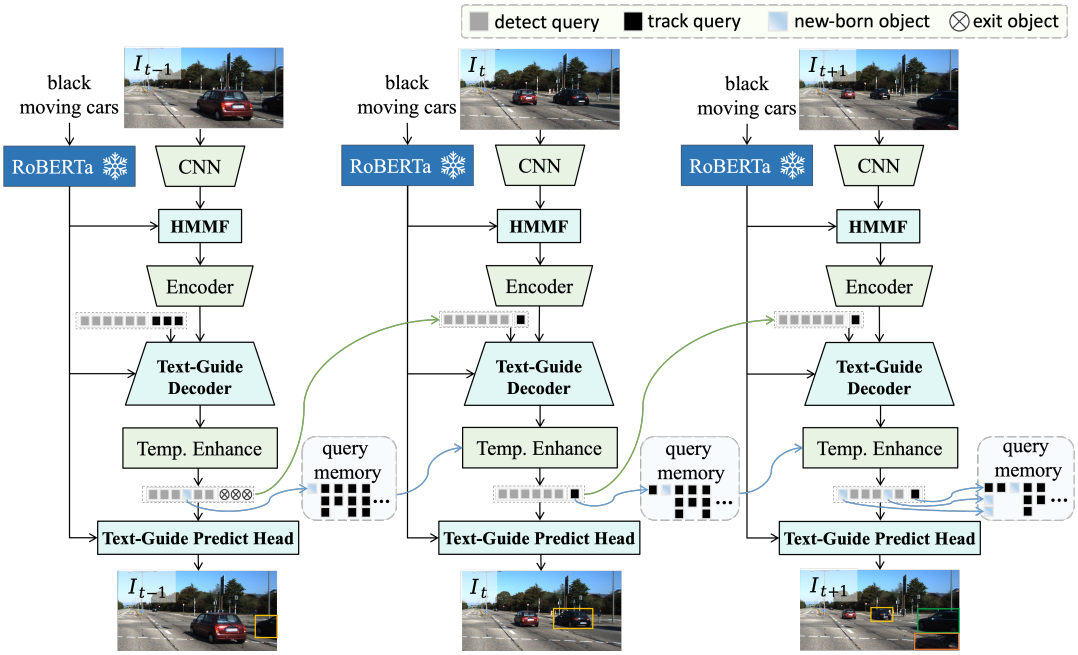


Figure 2: The overall architecture of Hierarchical Fine-grained text-image Fusion framework (HFF-Tracker).

pRMOT, we incorporate its Temporal Enhancement Module to bolster the model's ability to aggregate historical information across multiple frames. Furthermore, we have designed a Look-Back training strategy to optimize the utilization of datasets and to facilitate the learning of difficult examples.

Given a $(video, text)$ pair as input, the HFF-Tracker aims to ground all semantically matched objects within the video. The video sequence of length N and the referring text are denoted as $video = \{I_1, I_2, ..., I_N\}$ and $text$, respectively. At timestamp t, the t^{th} frame I_t is processed using a CNN backbone to extract feature maps F_t, which include 4 level feature maps, denoted as $F_t = \{F_t^1, F_t^2, F_t^3, F_t^4\}$. Simultaneously, a pre-trained linguistic model, RoBERTa (Liu et al. 2019), is employed to extract text features S. Subsequently, we utilize the HMMF module for fine-grained integration of word-level and sentence-level features for each layer of feature maps F_t and text features S, respectively. This process allows us to obtain highly expressive visual-text features. These visual-text features are then fed into an encoder for further interaction of image and text features, enhancing the visual-text features. Next, the learnable detect queries Q^D, track queries Q_{t-1}^T (from timestamp $t-1$), the enhanced visual-text features, and the text features S are entered into the Text-Guide Decoder. This step endows the queries with the ability to identify objects that match the description of the input $text$. Considering the temporal relationship between video frames, we employ the Temporal Enhancement Module to improve tracking queries across different frames. Finally, queries containing object information are input into the Text-Guide Predict Head to forecast the objects' trajectories in the current frame in accordance with the text description.

Hierarchical Multi-Modal Fusion Module

Cross-modal fusion of visual and textual features is pivotal for the RMOT task, given its direct influence on the model's performance. The prevalent approach is to map both types of features into a unified feature space, concatenate them, and subsequently feed them into an encoder. This process establishes dense connections via self-attention, as demonstrated by MDETR (Kamath et al. 2021). However, this method carries a substantial computational cost due to the high volume of tokens in images. To mitigate this computational complexity, TransRMOT employs a cross-attention mechanism before the encoder to merge visual and textual features. Despite its simplicity, this approach may not effectively amalgamate these two feature modalities and could potentially lead to information loss. To address above problems, we propose Hierarchical Multi-Modal Fusion (HMMF) module to fine-grained the visual and textual features both at word-level and sentence-level, as illustrated in Figure 3.

Given the textual features $S \in \mathcal{R}^{L\times D}$ and visual feature maps $F_t = \{F_t^1, F_t^2, F_t^3, F_t^4\}$ of frame I_t. Here, L represents the number of tokens and D is the feature dimension of word vectors. Also, $F_t^l \in \mathcal{R}^{C_l\times H_l\times W_l}$, where C_l, H_l, W_l represents the channel, height, and width, respectively. To align the channels between visual feature maps F_t and textual features S, each feature map layer F_t^l is processed through an independent Visual Project Layer (VPL), composed of a 1×1 convolution and group normalization. This process reduces its channel number to d, i.e., $F'^l_t \in \mathcal{R}^{d\times H_l\times W_l}$. Similarly, textual features S are projected into $S' \in \mathcal{R}^{L\times d}$ using a Text Project Layer (TPL), which consists of a fully-connected layer and layer normalization.

Subsequently, we partition textual features S' into word

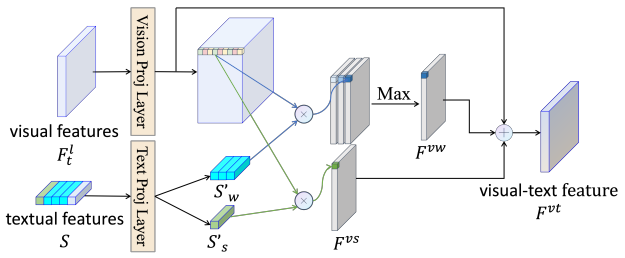


Figure 3: Diagram of the Hierarchical Multi-Modal Fusion (HMMF) Module, which can perform fine-grained fusion of pixel-level visual features and text features across different semantic levels.

features ${S'}_w \in \mathcal{R}^{(L-2)\times d}$ and a sentence feature ${S'}_s \in \mathcal{R}^{1\times d}$ based on tokens. It is worth noting that the first token signifies the semantic of the entire statement, the last token represents the end of the statement, and the remaining tokens correspond to the words of the input text, respectively.

Then, the word features ${S'}_w$ and sentence feature ${S'}_s$ are utilized for pixel-by-pixel interaction with the l^{th} level visual features ${F'}_t^l$, yielding visual-word feature $F^{vw} \in \mathcal{R}^{H_l\times W_l\times 1}$ and visual-sentence feature $F^{vs} \in \mathcal{R}^{H_l\times W_l\times 1}$. The specific formula is as follows:

$$ {F'}^{vw}(i,j,k) = (\sum_{c=1}^{d} {F'}_t^l(i,j,n)\times {S'}_w(k,c))\forall k \in [1, L-2], \tag{1} $$

$$ F^{vw}(i,j) = Max({F'}^{vw}(i,j,k)\forall k \in [1, L-2]), \tag{2} $$

$$ F^{vs}(i,j) = \sum_{c=1}^{d} {F'}_t^l(i,j,c) \times {S'}_s(c), \tag{3} $$

Where i and j denote the horizontal and vertical coordinates of feature map ${F'}_t^l$, respectively. Meanwhile, c represents the channel number, and k corresponds to the k^{th} word of the input text.

Finally, the F^{vw} and F^{vs} are added to ${F'}_t^l$ channel by channel as attention items, resulting in the visual-text feature $F^{vt} \in \mathcal{R}^{H_l\times W_l\times d}$:

$$ F^{vt}(i,j) = F^{vw}(i,j) + F^{vs}(i,j) + {F'}_t^l(i,j), \tag{4} $$

In this manner, we can efficiently and comprehensively aggregate the four levels of visual features $F_t = \{F_t^1, F_t^2, F_t^3, F_t^4\}$ and text features S at the word, sentence and pixel level. To further enhance the visual-text information, we flatten and concatenate the four levels if visual-text features into a 2D tensor, which is then fed into the encoder.

Text-Guide Decoder

Consistent with TransRMOT, the queries input into the decoder are divided into *detect queries* and *track queries*. The *detect query* refers to learnable queries that probe encoded features to yield instance embeddings, which further

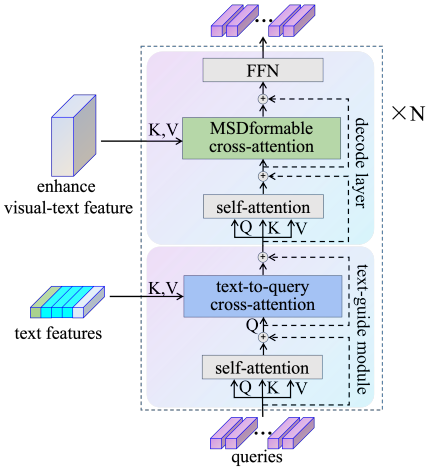


Figure 4: Schematic of the Text-Guide Decoder: Composed of N Pairs of Text-Guide Modules and Decoder Layers for Enhanced Query Generation.

produce instance boxes and classes. And the *track query* is the decoder embedding from the previous frame, which is updated for the current frame to track the same instance. The tracking process is shown in Figure 2.

As evident from the above discussion, the query plays a pivotal role in the entire architecture, acting as the carrier of the target in the trajectory and directly influencing the final prediction results. We believe that it is essential for the query to perceive semantic information before being fed into each decoder layer. Consequently, we propose a Text-Guide Decoder (TGD), which comprises N pairs of decoder layers and text-guide modules, as depicted in Figure 4.

Specifically, the decoder layer, akin to the one in Deformable DETR (Zhu et al. 2021), comprises a self-attention mechanism, MSDformable cross-attention, and a feed-forward neural network. The architecture of the text-guide module is quite simple, yet effective, consisting of a self-attention module and a text-to-query cross-attention module. The process initiates with self-attention processing on the input query, followed by normalization and the sequential application of residual connections. Subsequently, the result is used as the query for the cross-attention operation, with the text feature serving as both the key and value. This is then normalized, a residual connection is applied, and it is passed through a feed-forward neural network. In this way, we can not only imbue the *detect query* with text semantic information but also enhance the text semantic expression capability of the *track query*. Next, the query, processed by the N pairs of decoder layers and the text-guide module, is input into the Temporal Enhancement Module. This step serves to enhance the temporal consistency of the queries.

Text-Guide Predict Head

After processing through the Text-Guide Decoder and Temporal Enhancement Module, the learnable queries evolve into queries enriched with object-referring information, lo-

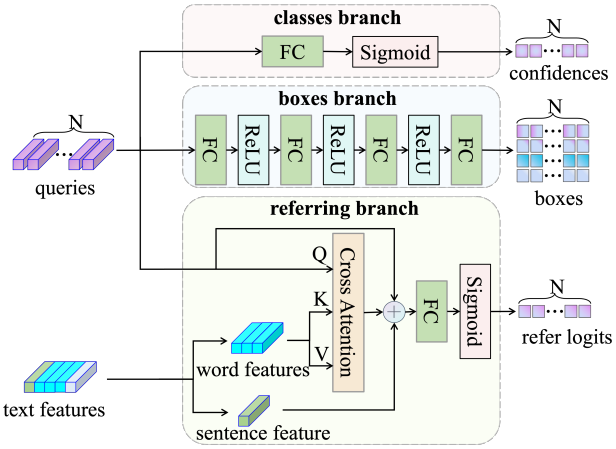


Figure 5: Illustration of Text-Guide Predict Head, which consists of classes, boxes and referring branches.

cation information, and category information. We have designed a Text-Guide Predict Head (TGPH) to generate object trajectories that align with the text description based on these queries. The TGPH comprises three branches: the classes branch, the boxes branch, and the referring branch, as illustrated in Figure 5.

The classes branch consists of a fully connected layer and a sigmoid activation function, outputting a confidence score that indicates whether a query represents a true object or an empty one. The boxes branch is a four-layer feed-forward network with ReLU activation, except for the last layer, which predicts the location and size of the boxes. As for the referring branch, the text feature is first split into word features and sentence features. Following this, the word features and queries are utilized for cross-attention operations. Subsequently, the sentence features, queries, and the outcomes of the cross-attention processing are combined through element-wise addition. The sum of these elements is then input into a fully connected layer and processed via a sigmoid activation function. This process generates a referent score, which indicates the probability of whether the instance matches the text description.

Look-Back Training Strategy

For end-to-end, transformer-based RMOT algorithms (like TransRMOT, MLS-Track, TempRMOT), the common training strategy involves using a sliding window to scan through the video, sampling a video clip of length L. An textual expression is then randomly chosen from the video's expression pool. This sampled video clip and expression are paired and fed into the model for training. However, this training strategy, while an improvement on the training strategy of the MOTR (Zeng et al. 2022) algorithm that only considers a full traversal of the video data, has certain limitations when applied to RMOT tasks. These limitations include: Firstly, the training pair (video clip, expression) may not always contain positive samples, as the object described by the textual expression often doesn't span the entire video. Thus, a randomly paired video clip and expression may lack positive samples corresponding to the text description. Second, there's a risk of overlooking certain expressions, as each video can have multiple expressions. For instance, Refer-KITTI has an average of 49.7 expressions per video. As a result, there are numerous (video clip, expression) pairs that may not be utilized for training throughout the entire training process. This could potentially lead to missed learning opportunities and a less robust model. Indeed, these two drawbacks highlight that the existing training strategy may overlook a large amount of valuable, annotated data during the model training process. This could potentially limit the effectiveness and accuracy of the trained model.

To address this issue, we propose a new training strategy called the Look-Back training strategy, which is based on the existing training approach. This strategy comprises two sub-strategies: Look-Back-Hard and Look-Back-Remain. These strategies are designed to make better use of the available annotated data, thereby improving the effectiveness of the training process and the performance of the resulting model.

Look-Back-Hard. We have observed that during the initial stages of model training, training samples with substantial losses often contain high-quality annotated data that aligns with the objectives of the text description. As a result, we have developed a Look-Back-Hard adaptive method to identify and utilize these high-loss training samples within a specified training epoch. Specifically, we define s and e as the epochs at which the Look-Back-Hard policy commences and concludes, respectively. Between the s and e epochs , we employ the existing training strategy to carry out standard training, recording the loss value of each training sample. After the epoch concludes, we select the samples with the top 30% of loss values and retrain on them. This method is not only straightforward but also highly effective at ensuring that valuable data isn't overlooked during the training process.

Look-Back-Remain. For situations where certain (video clip, expression) pairs are not utilized during training, we have designed a simple yet effective strategy named Look-Back-Remain. This strategy can be implemented after each epoch. Specifically, within each epoch, we keep a record of the (video clip, expression) pairs that are not used for training. At the end of the epoch, these recorded pairs are then utilized for additional training. In this work, we apply the Look-Back-Remain strategy solely during the final epoch of model training, enhancing the model's robustness and performance with only a minimal increase in training cost.

Experiment

Experiment Settings

Dataset and Metric. Refer-KITTI, being the first RMOT dataset, has garnered the most attention. Most RMOT algorithms have been experimentally validated using this dataset. We choose Refer-KITTI as our benchmark dataset to facilitate comparison with other algorithms and conduct ablation studies on it. Additionally, we carry out performance comparisons with the state-of-the-art (SOTA) algorithm on the more challenging Refer-KITTI-V2 dataset. Our pri-

Methods	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr	LocA
DeepSORT (Wojke, Bewley, and Paulus 2017)	25.59	19.76	34.31	26.38	36.93	39.55	61.05	71.34
FairMOT (Zhang et al. 2021)	22.78	14.43	39.11	16.44	45.48	43.05	71.65	74.77
TransTrack (Sun et al. 2021)	32.77	23.31	45.71	32.33	42.23	49.99	78.74	79.48
ByteTrack (Zhang et al. 2022)	24.95	15.50	43.11	18.25	43.48	48.64	70.72	73.90
CStrack (Liang et al. 2022)	27.91	20.65	39.10	33.76	32.61	43.12	71.82	79.51
TrackFormer (Meinhardt et al. 2022)	33.26	25.44	45.87	35.21	42.10	50.26	78.92	79.63
TransRMOT (Wu et al. 2023a) ♣	46.56	37.97	57.33	49.69	60.10	60.02	**89.67**	90.33
iKUN (Du et al. 2024) ♣	38.06	26.19	56.72	39.94	37.82	63.98	71.83	74.26
EchoTrack (Lin et al. 2024)	39.47	31.19	51.56	42.65	48.86	56.68	81.21	79.93
DeepRMOT (He et al. 2024)	39.55	30.12	53.23	41.91	47.47	58.47	82.16	80.49
MLS-Track (Ma et al. 2024)	49.05	40.03	60.25	**59.07**	54.18	65.12	88.12	-
TempRMOT (Zhang et al. 2024) ♣	50.91	39.30	66.09	52.01	60.24	71.20	87.59	**90.82**
HFF-Tracker (ours)	**52.41**	**41.29**	**66.65**	53.42	**62.89**	**71.48**	88.96	90.76

Table 1: Comparison with state-of-the-art methods on Refer-KITTI. ♣ indicates that the result was obtained by performing inference using the official open source code and weights after frame correction. The best results are highlighted in bold.

Methods	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr	LocA
FairMOT (Zhang et al. 2021)	22.53	15.80	32.82	20.60	37.03	36.21	71.94	78.28
ByteTrack (Zhang et al. 2022)	24.59	16.78	36.63	22.60	36.18	41.00	69.63	78.00
TransRMOT (Wu et al. 2023a)	31.00	19.40	49.68	36.41	28.97	54.59	82.29	89.82
iKUN (Du et al. 2024)	10.32	2.17	49.77	2.36	19.75	58.48	68.64	74.56
TempRMOT (Zhang et al. 2024) ♠	34.72	22.52	**53.64**	32.41	41.76	58.98	**83.16**	**90.38**
HFF-Tracker (ours)	**36.18**	**24.64**	53.27	**36.86**	**41.83**	**59.42**	81.40	89.77

Table 2: Comparison with state-of-the-art methods on Refer-KITTI-V2. ♠ indicates that the result was obtained by performing inference using the official open source code and weights. The best results are highlighted in bold.

mary metric is "Higher Order Tracking Accuracy" (HOTA) (Luiten et al. 2021), supplemented by an analysis of detection accuracy (DetA) and association accuracy (AssA).

Model Details. Following TransRMOT, we employ ResNet50 (He et al. 2016) as the CNN backbone for extracting visual features. We also use a text encoder derived from RoBERTa (Liu et al. 2019) to embed language prompts. The output feature dimension for both the Visual Project Layer and the Text Project Layer is set to d=256. In HFF-Tracker, we set the number of attention heads for multi-head attention to 8, and the output feature dimension to 256. The number of $detect\ query$ is set to 300.

Training and Testing. In accordance with the TransRMOT setup, the parameters of the Transformer-Encoder and Transformer-Decoder are initialized from the official Deformable DETR (Zhu et al. 2021). The parameters in the text encoder remain frozen during the training process, while all other parameters are randomly initialized. We employ the AdamW optimizer to train the HFF-Tracker, with a base learning rate of 1e-4. The learning rate of the backbone are set to 1e-5. For the Refer-KITTI dataset, the model is trained for 80 epochs, and for the Refer-KITTI-v2 dataset, it is trained for 70 epochs. The learning rate decays by a factor of 10 at the 40^{th} epoch. The Look-Back-Hard starts at the 5^{th} epoch and ends at the 40^{th} epoch, while the Look-Back-Remain only begins after the last epoch of training. The overall training process is conducted on 8 Nvidia V100 GPUs, with a batch size of 1. During testing, we set the class confident threshold $\beta_{cls} = 0.7$ and the reference threshold $\beta_{ref} = 0.5$ for Refer-KITTI dataset. For the Refer-KITTI-V2 dataset, we set and class threshold $\beta_{cls} = 0.6$ and reference threshold $\beta_{ref} = 0.4$. To ensure the repeatability of the experimental results, we set the random seed to 42 for the training stage and 2020 for the inference stage.

Benchmark Experiments

Refer-KITTI. Table 1 compares HFF-Tracker with mainstream RMOT methods on the Refer-KITTI. Each score is either sourced from previous studies or obtained by inference using the official open source code and weights. It is evident that the proposed HFF-Tracker outperforms all other algorithms, ranking first overall. Particularly, HFF-Tracker surpasses other methods by a large margin in term of HOTA, DetA and AssA, achieving scores of 52.41% in HOTA, 41.29% in DetA, and 55.9% in AssA. When compared to the second-best method, TempRMOT, our HFF-Tracker has increased the HOTA by 2.16%, DetA by 2.45 %, and AssA by 1.47%, establishing a new state of the art.

Refer-KITTI-V2. As shown in Table 2, we report results on Refer-KITTI-V2. Each score is either sourced from previous studies or obtained by inference using the official open source code and weights. HFF-Tracker also achieves the start-of-the-art performance on this dataset, surpassing the closest competitor TempRMOT by margins of 1.46% and 2.12% on HOTA and DetA, respectively. The results show the generalization and scalability of HFF-Tracker.

HMMF	TGD	TGPH	Temp.	LB	HOTA	DetA	AssA
					46.56	37.97	57.33
✓					47.53	39.30	57.67
✓	✓				47.99	39.89	57.90
✓	✓	✓			48.62	39.92	59.37
✓	✓	✓	✓		51.62	39.91	**66.90**
✓	✓	✓	✓	✓	**52.41**	**41.29**	66.65

Table 3: Ablation study on Refer-KITTI testing set.

LBH	LBR	HOTA	DetA	AssA
		51.62	39.91	**66.90**
✓		52.07	40.96	66.32
✓	✓	**52.41**	**41.29**	66.65

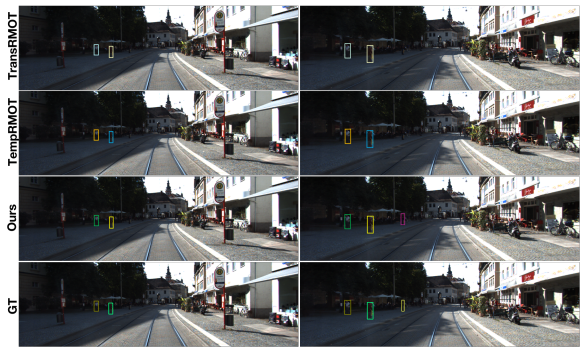
Table 4: Ablation study of applying Look-Back-Hard and Look-Back-Remain on Refer-KITTI testing dataset.

Ablation Study

Module Effectiveness. To demonstrate the effectiveness of the proposed modules, we perform an ablation study on Refer-KITTI dataset. Starting with a baseline of TransRMOT, we incrementally enhance it by adding the following modules: Hierarchical Multi-Modal Fusion (HMMF) module, Text-Guide Decoder (TGD), Text-Guide Predict Head (TGPH), Temporal Enhancement (Temp.) module, and Look-Back (LB) training strategy. Table 3 summarizes the path from the original TransRMOT to our proposed HFF-Tracker. The first row represents the original TransRMOT, without any enhancements. The incorporation HMMF module results in a 0.97% increase in HOTA, a 1.33% increase in DetA, and a 0.34% increase in AssA. Integration of the TGD and TGPH enhances HOTA by 1.09%, DetA by 0.62%, and AssA by 1.7%. The addition of the Temporal Enhancement module yields substantial improvement with HOTA increasing to 51.62%, AssA to 66.90%, although DetA slightly decreases to 39.91%. The inclusion of Look-Back training strategy achieves optimal performance for HFF-Tracker at a minimal additional training cost, resulting in HOTA reaching 52.41%, DetA increasing to 41.29%, and AssA achieving 66.65%.

Effectiveness of Look-Back. The Look-Back training strategy comprises two sub-strategies: Look-Back-Hard and Look-Back-Remain. We examine the impact of both Look-Back-Hard (LBH) and Look-Back-Remain (LBR) during the training phase on the model performance, using the Refer-KITTI. The results are presented in Table 4, employing the Look-Back-Hard (LBH) strategy enhances the model's performance, elevating the HOTA score from 51.62% to 52.07% and the DetA from 39.91% to 40.96%. Further application of the Look-Back-Remain (LBR) strategy results in additional performance improvements, with HOTA increasing to 52.41% and DetA to 41.29%. However, this comes with a slight trade-off in AssA, which sees a minor reduction of 0.35%.

(a) Expression: "cars in left".



(b) Expression: "pedestrian in the left".

Figure 6: Visualization of predictions from TransRMOT, TempRMOT, and our HFF-Tracker.

Qualitative Results

We present several tracking results in Figure 6. As shown, HFF-Tracker can accurately track objects and comprehend complex instructions related to position. For instance, when dealing with instructions such as "cars on the left" and "pedestrian on the left", both TempRMOT and TransRMOT lose track of the object, which is not an issue encountered with HFF-Tracker.

Conclusion

In this work, we introduce HFF-Tracker, a novel framework for referring multi-object tracking, performing fine-grained fusion of pixel-level visual and text features across various semantic levels. We propose a Hierarchical Multi-Modal Fusion (HMMF) module for early-stage, granular integration of text and image features. Our Text-Guided Decoder (TGD) provides the query with prior semantic information during decoding, while the Text-Guided Prediction Head (TGPH) leverages text information to enhance prediction performance. Through these three modules, we achieve fine-grained interaction of image and text features at different levels. Additionally, we implemente an adaptive Look-Back training strategy to optimize the use of valuable labeled data. Extensive experiments demonstrate that our method outperforms other state-of-the-art methods with remarkable margins on both the Refer-KITTI and Refer-KITTI-V2 datasets.

Acknowledgements

We would like to express our sincere gratitude to all those who have contributed to the completion of this paper. Special thanks to our colleagues and peers for their valuable feedback and insightful discussions. We also appreciate the support and encouragement from our families and friends throughout this research journey.

References

Dave, A.; Khurana, T.; Tokmakov, P.; Schmid, C.; and Ramanan, D. 2020. TAO: A Large-Scale Benchmark for Tracking Any Object. In *Computer Vision – ECCV 2020*, 436–454. Cham: Springer International Publishing. ISBN 978-3-030-58558-7.

Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; and Leal-Taixé, L. 2020. MOT20: A benchmark for multi object tracking in crowded scenes. arXiv:2003.09003.

Du, Y.; Lei, C.; Zhao, Z.; and Su, F. 2024. ikun: Speak to trackers without retraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19135–19144.

Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*.

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

He, W.; Jian, Y.; Lu, Y.; and Wang, H. 2024. Visual-Linguistic Representation Learning with Deep Cross-Modality Fusion for Referring Multi-Object Tracking. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6310–6314. IEEE.

Jiao, B.; Liu, L.; Gao, L.; Wu, R.; Lin, G.; WANG, P.; and Zhang, Y. 2023. Toward Re-Identifying Any Animal. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 40042–40053. Curran Associates, Inc.

Kamath, A.; Singh, M.; LeCun, Y.; Synnaeve, G.; Misra, I.; and Carion, N. 2021. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1780–1790.

Liang, C.; Zhang, Z.; Zhou, X.; Li, B.; Zhu, S.; and Hu, W. 2022. Rethinking the competition between detection and reid in multiobject tracking. *IEEE Transactions on Image Processing*, 31: 3182–3196.

Liao, G.; Li, J.; and Ye, X. 2024. VLM2Scene: Self-Supervised Image-Text-LiDAR Learning with Foundation Models for Autonomous Driving Scene Understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 3351–3359.

Lin, J.; Chen, J.; Peng, K.; He, X.; Li, Z.; Stiefelhagen, R.; and Yang, K. 2024. EchoTrack: Auditory Referring Multi-Object Tracking for Autonomous Driving. arXiv:2402.18302.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692.

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

Ma, Z.; Yang, S.; Cui, Z.; Zhao, Z.; Su, F.; Liu, D.; and Wang, J. 2024. MLS-Track: Multilevel Semantic Interaction in RMOT. arXiv:2404.12031.

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

Milan, A.; Leal-Taixe, L.; Reid, I.; Roth, S.; and Schindler, K. 2016. MOT16: A Benchmark for Multi-Object Tracking. arXiv:1603.00831.

Nguyen, P.; Quach, K. G.; Kitani, K.; and Luu, K. 2024. Type-to-track: Retrieve any object via prompt-based tracking. *Advances in Neural Information Processing Systems*, 36.

Pang, Z.; Li, J.; Tokmakov, P.; Chen, D.; Zagoruyko, S.; and Wang, Y.-X. 2023. Standing between past and future: Spatio-temporal modeling for multi-camera 3d multi-object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17928–17938.

Sun, P.; Cao, J.; Jiang, Y.; Yuan, Z.; Bai, S.; Kitani, K.; and Luo, P. 2022. DanceTrack: Multi-Object Tracking in Uniform Appearance and Diverse Motion. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sun, P.; Cao, J.; Jiang, Y.; Zhang, R.; Xie, E.; Yuan, Z.; Wang, C.; and Luo, P. 2021. TransTrack: Multiple Object Tracking with Transformer. arXiv:2012.15460.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. arXiv:1706.03762.

Wojke, N.; Bewley, A.; and Paulus, D. 2017. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, 3645–3649. IEEE.

Wu, D.; Han, W.; Wang, T.; Dong, X.; Zhang, X.; and Shen, J. 2023a. Referring Multi-Object Tracking. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wu, D.; Han, W.; Wang, T.; Liu, Y.; Zhang, X.; and Shen, J. 2023b. Language Prompt for Autonomous Driving. arXiv:2309.04379.

Zeng, F.; Dong, B.; Zhang, Y.; Wang, T.; Zhang, X.; and Wei, Y. 2022. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, 659–675. Springer.

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

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

Zhang, Y.; Wu, D.; Han, W.; and Dong, X. 2024. Bootstrapping Referring Multi-Object Tracking. arXiv:2406.05039.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2021. Deformable DETR: Deformable Transformers for End-to-End Object Detection. arXiv:2010.04159.