

## Referring Multi-Object Tracking

Dongming Wu<sup>1\*†</sup>, Wencheng Han<sup>2\*</sup>, Tiancai Wang<sup>3</sup>, Xingping Dong<sup>4</sup>, Xiangyu Zhang<sup>3,5</sup>, Jianbing Shen<sup>2†</sup>  
<sup>1</sup> Beijing Institute of Technology, <sup>2</sup> SKL-IOTSC, CIS, University of Macau, <sup>3</sup> MEGVII Technology,  
<sup>4</sup> School of Computer Science, Wuhan University, <sup>5</sup> Beijing Academy of Artificial Intelligence  
 {wudongming97, wenchenghan, shenjiangbingcg}@gmail.com, wangtiancai@megvii.com

### Abstract

Existing referring understanding tasks tend to involve the detection of a single text-referred object. In this paper, we propose a new and general referring understanding task, termed referring multi-object tracking (RMOT). Its core idea is to employ a language expression as a semantic cue to guide the prediction of multi-object tracking. To the best of our knowledge, it is the first work to achieve an arbitrary number of referent object predictions in videos. To push forward RMOT, we construct one benchmark with scalable expressions based on KITTI, named Refer-KITTI. Specifically, it provides 18 videos with 818 expressions, and each expression in a video is annotated with an average of 10.7 objects. Further, we develop a transformer-based architecture TransRMOT to tackle the new task in an online manner, which achieves impressive detection performance and outperforms other counterparts. The Refer-KITTI dataset and the code are released at <https://referringmot.github.io>.

### 1. Introduction

Recently, referring understanding [5, 17, 33, 55], integrating natural language processing into scene perception, has raised great attention in computer vision community. It aims to localize regions of interest in images or videos under the instruction of human language, which has many applications, such as video editing and autonomous driving. For referring understanding, several significant benchmarks have been published. Flickr30k [53], ReferIt [15], and RefCOCO+/+g [55] have greatly encouraged the development of image-based referring tasks. More datasets (e.g., Lingual OTB99 [19], Cityscapes-Ref [42], Talk2Car [5], Refer-DAVIS<sub>17</sub> [17], and Refer-Youtube-VOS [38]) are further

\*Equal contribution. †Corresponding author: Jianbing Shen. This work was supported in part by the FDCT grant SKL-IOTSC(UM)-2021-2023, the Grant MYRG-CRG2022-00013-IOTSC-ICI, and the Start-up Research Grant (SRG) of University of Macau (SRG2022-00023-IOTSC). ‡The work is done during the internship at MEGVII Technology.

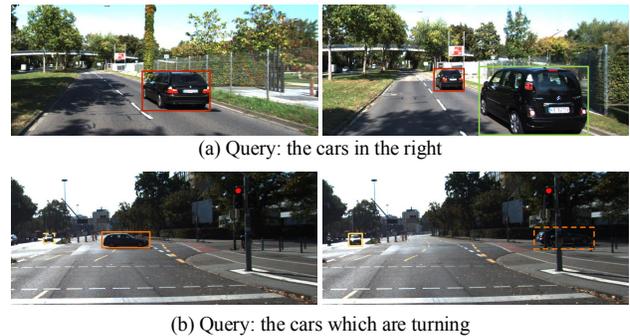


Figure 1. **Representative examples from RMOT.** The expression query can refer to multiple objects of interest (a), and captures the short-term status with accurate labels (b).

proposed to cover the application in videos.

Despite these advanced progress, previous benchmarks have two typical limitations. **First**, each expression tends to correspond to only one target. However, many objects have the same semantics in an open world, i.e., one single expression could refer to multiple objects. From this side, existing datasets lack flexible simulation on the multi-object scenarios, causing referring understanding tasks far from satisfactory. **Second**, the given expression may only describe part of frames for the video referring task, making the correspondence inaccurate. For example, given the expression ‘the car which is turning’, we have to predict the overall trajectory even if the car has finished the turning action. Obviously, a single expression cannot cover all short-term status of one target. Overall, existing datasets fail to provide an accurate evaluation under the situations of multiple referent targets and temporal status variances.

To address these problems, we propose a novel video understanding task guided by the language description, named referring multi-object tracking (RMOT). Given a language expression as a reference, it targets to ground all semantically matched objects in a video. Unlike previous tasks, our proposed RMOT is much closer to the real environment, as each expression can involve multiple objects. For instance, the expression query ‘the cars in the right’ corresponds to one object at the 20<sup>th</sup> frame but two objects at the



Figure 2. **More examples of Refer-KITTI.** It provides high-diversity scenes and high-quality annotations referred to by expressions.

40<sup>th</sup> frame (see Fig. 1 (a)). The phenomenon indicates that RMOT focuses more on finding the matched targets so that the referent number can be flexibly changed. In addition, the temporal status variances are also considered in RMOT. As shown in Fig. 1(b), the given example shows the cars can be detected only when they start the turning action, and the tracking will be ended if they finish the activity.

To speed up the development of RMOT, we construct a new benchmark, *i.e.*, Refer-KITTI, concerning the traffic scenes. It is developed from the public KITTI [9] dataset. Compared to existing referring understanding datasets, it has three distinguishing characteristics: **i)** High flexibility with referent objects. The number of objects described by each expression range from 0 to 105, with 10.7 on average. **ii)** High temporal dynamics. The temporal status of targets covers a longer time with more frames (varying in 0~400 frames), and the temporal variance of targets is accurately captured using our labeling tool. **iii)** Low labeling cost with identification spread. We provide an effortless tool to annotate a target tracklet using only two clicks.

Although RMOT has a more flexible referring setting, it brings additional challenges: multi-object prediction and cross-frame association. Towards this end, we propose an end-to-end differentiable framework for RMOT. Our model builds upon the recent DETR framework [3], enhanced by powerful cross-modal reasoning and cross-frame conjunction. It has an encoder-decoder architecture. Specifically, we design an early-fusion module in the encoder to densely integrate visual and linguistic features, followed by a stack of deformable attention layers for further refining the cross-modal representations. In the decoder, query-based embeddings interact with the cross-modal features to predict referent boxes. To track multi-objects, similar to MOTR [57], we decouple the object queries into track query for tracking objects of previous frames and detect query for predicting the bounding boxes of new-born objects.

In summary, our contributions are three-fold. **First**, we propose a new task for referring multi-objects, called re-

Dataset	Video	Images	Instances per-expression	Temporal ratio per-expression
RefCOCO [55]	-	26,711	1	1
RefCOCO+ [55]	-	19,992	1	1
RefCOCOg [55]	-	26,711	1	1
Talk2Car [5]	✓	9,217	1	-
VID-Sentence [4]	✓	59,238	1	1
Refer-DAVIS <sub>17</sub> [17]	✓	4,219	1	1
Refer-YV [38]	✓	93,869	1	1
Refer-KITTI	✓	6,650	10.7	0.49

Table 1. **Comparison of Refer-KITTI with existing datasets.** Refer-YV is short for Refer-Youtube-VOS. The temporal ratio represents the average ratio of referent frames covering the entire video sequence. ‘-’ means unavailable.

ferring multi-object tracking (RMOT). It tackles limitations in the existing referring understanding tasks and provides multi-object and temporally status-variant circumstances. **Second**, we formulate a new benchmark, Refer-KITTI, to help the community to explore this new field in depth. As far as we know, it is the first dataset specializing in an arbitrary number of object predictions. **Third**, we propose an end-to-end framework built upon Transformer, termed as TransRMOT. With powerful cross-modal learning, it provides impressive RMOT performance on Refer-KITTI compared to hand-crafted RMOT methods.

## 2. Related Work

**Referring Understanding Datasets.** Many advanced datasets have greatly contributed to the progress of referring understanding. Pioneering attempts (*e.g.*, Flickr30k [53], ReferIt [15], RefCOCO+/g [55]) propose to employ a succinct yet unambiguous language expression to ground corresponding visual region within an image. However, these datasets are fully image-based and do not fit well with common and practical video scenes. Therefore, more efforts have been devoted to video-based benchmarks in recent years, such as Lingual OTB99 [19], Cityscapes-Ref [42], VID-Sentence [4], and Talk2Car [5]. In ad-



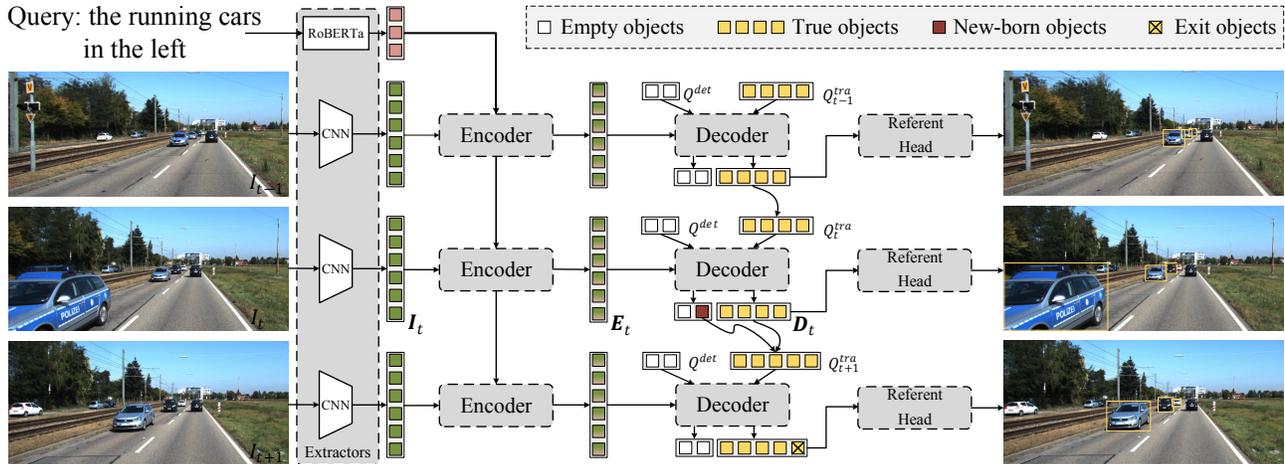


Figure 6. **The overall architecture of TransRMOT.** It is an online cross-modal tracker and includes four essential parts: feature extractors, cross-modal encoder, decoder, and referent head. The feature extractors embed the input video and the corresponding language query into feature maps/vectors. The cross-modal encoder models comprehensive visual-linguistic representation via efficient fusion. The decoder takes the visual-linguistic features, detect queries and the track queries as inputs and updates the representation of queries. The updated queries are further used to predict the referred objects by the referent head.

can automatically propagate these labels to the intermediate frames according to their identification number. Finally, the labeled information (*i.e.*, frame ID, object ID, and box coordinates) and the corresponding expression are saved for training and testing. The annotation procedure produces lower human costs than the frame-by-frame manner.

**Dataset Split.** KITTI provides 21 high-resolution and long-temporal videos, but we abandon three over-complex videos and use the remaining 18 videos to formulate Refer-KITTI. We create a total of 818 expressions for Refer-KITTI using our labeling tool. The word cloud of the expressions is shown in Fig. 4 (a). Refer-KITTI is randomly split into separate *train*, and *test* sets, yielding a unique split consisting of 15 training videos and 3 testing videos.

### 3.2. Dataset Features and Statistics

To offer deeper insights into Refer-KITTI, we next discuss the discriminative features and descriptive statistics.

**High Flexibility with Referent Objects.** Different from previous datasets that contain just one referent object for each language expression, RMOT is designed to involve an arbitrary number of predicted objects in videos. Quantitatively, the expressions of Refer-KITTI mostly describe 0-25 objects, and the maximum number can be up to 105. On average, each expression in a video corresponds to **10.7** objects. The per-expression object number distribution is shown in Fig. 4 (b). These statistics are more representative of high-flexibility applications with referent objects.

**High Temporal Dynamic.** Another real-world complexity is reflected in the temporal dimension of referent objects. Fig. 5 (a) shows the length distribution of frames per expression. Most expressions of Refer-KITTI cover 0-400 frames,

while the longest sequence has more than 600 frames. Additionally, we show the per-expression temporal ratio covering the entire video in Fig. 5 (b). It indicates many referent objects enter or exit from visible scenes. The long time and undetermined ratio bring an additional challenge compared to existing works, *i.e.*, *cross-frame object association*.

### 3.3. Evaluation Metrics

We adopt Higher Order Tracking Accuracy (HOTA) [27] as standard metrics to evaluate the new benchmark. Its core idea is calculating the similarity between the predicted and ground-truth tracklet. Unlike MOT using HOTA to evaluate all visible objects, when those non-referent yet visible objects are predicted, they are viewed as false positives in our evaluation. As the HOTA score is obtained by combining Detection Accuracy (DetA) and Association Accuracy (AssA), *i.e.*,  $HOTA = \sqrt{DetA \cdot AssA}$ , it performs a great balance between measuring frame-level detection and temporal association performance. Here, DetA defines the detection IoU score, and AssA is the association IoU score.

## 4. Method

### 4.1. Network Architecture

The overall pipeline of our method is illustrated in Fig. 6. Taking the video stream as well as a language query as inputs, the goal is to output the track boxes of the corresponding query. Similar to MOTR [57], our model mainly follows the Deformable DETR [61], and we make several modifications on it to adapt the cross-modal inputs. It consists of four key components: feature extractors, cross-modal encoder, decoder and referent head. The feature extractor first

produces visual and linguistic features for the raw video and text. Then, the cross-modal encoder fuses the features of two modalities. Next, the decoder is used to update the representation of object queries. Finally, the referent head predicts the target sequences based on the predicted classification, bounding box and referent scores.

**Feature Extractor.** Given a  $T$ -frame video, a CNN backbone model is used to extract the frame-wise pyramid feature maps, e.g., the  $t^{\text{th}}$  frame for features  $I_t^l \in \mathbb{R}^{C_l \times H_l \times W_l}$ , where  $C_l, H_l, W_l$  represents the channel, height, width of the  $l^{\text{th}}$  level feature map, respectively. At the same time, we employ a pre-trained linguistic model to embed the text with  $L$  words into 2D vectors  $S \in \mathbb{R}^{L \times D}$ , where  $D$  is the feature dimension of word vectors.

**Cross-modal Encoder.** The cross-modal encoder is responsible for accepting the visual and linguistic features and fusing them. The common strategy is to concatenate two kinds of features and feed them into the encoder to model dense connections via self-attention, like MDETR [14]. However, the computation cost of self-attention is enormous due to the large token number of images. To address this problem, we propose an early-fusion module to integrate the visual and linguistic features before deformable encoder layers. Our early-fusion module is illustrated in Fig. 7.

Specifically, given the  $l^{\text{th}}$  level feature maps  $I_t^l$ , we use a  $1 \times 1$  convolution to reduce its channel number to  $d = 256$ , and flatten it into a 2D tensor  $I_t^l \in \mathbb{R}^{H_l W_l \times d}$ . To keep the same channels with visual features, the linguistic features are projected into  $S \in \mathbb{R}^{L \times d}$  using a fully-connected layer. Three independent full-connected layers transform the visual and linguistic features as  $Q, K$ , and  $V$ :

$$\begin{aligned} Q &= W_q(I_t^l + P^V) \in \mathbb{R}^{H_l W_l \times d}, \\ K &= W_k(S + P^L) \in \mathbb{R}^{L \times d}, \\ V &= W_v S \in \mathbb{R}^{L \times d}, \end{aligned} \quad (1)$$

where  $W$ s are weights.  $P^V$  and  $P^L$  are position embedding of visual and linguistic features following [3, 43]. We make matrix product on  $K$  and  $V$ , and use the generated similarity matrix to weight linguistic features, i.e.,  $(QK^T / \sqrt{d})V$ . Here,  $d$  is the feature dimension. The original visual features are then added with the vision-conditioned linguistic features to produce the fused features  $\hat{I}_t^l$ :

$$\hat{I}_t^l = \frac{QK^T}{\sqrt{d}}V + I_t^l \in \mathbb{R}^{H_l W_l \times d}. \quad (2)$$

After fusing two modalities, a stack of deformable encoder layers is used to promote cross-modal interaction:

$$E_t^l = \text{DeformEnc}(\hat{I}_t^l) \in \mathbb{R}^{H_l W_l \times d}, \quad (3)$$

where  $E_t^l$  is encoded cross-modal embedding, which will facilitate referring prediction in the following decoder.

**Decoder.** The original decoder in the DETR framework uses learnable queries to probe encoded features for yielding instance embedding, further producing instance boxes

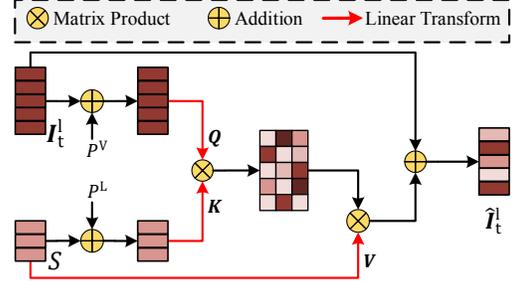


Figure 7. **The early-fusion module in our cross-modal encoder.** They achieve efficient visual-linguistic fusion followed by a stack of deformable encoder layers for further interaction.

and classes. To associate objects between adjacent frames, we make full use of the decoder embedding from the last frame, which is updated as *track query* of the current frame to track the same instance. For new-born objects in the current frame, we adapt the original query from DETR, named *detect query*. The tracking process is shown in Fig. 6.

Formally, let  $D_{t-1} \in \mathbb{R}^{N_{t-1} \times d}$  denote the decoder embedding from the  $(t-1)^{\text{th}}$  frame, which is further transformed into *track query* of the  $t^{\text{th}}$  frame, i.e.,  $Q_t^{tra} \in \mathbb{R}^{N'_{t-1} \times d}$ , using self-attention and feed-forward network (FFN). Note that part of the  $N_{t-1}$  decoder embeddings correspond to empty or exit objects, so we filter out them and only keep  $N'_{t-1}$  true embeddings to generate *track query*  $Q_t^{tra}$  in terms of their class score. Let  $Q_t^{det} \in \mathbb{R}^{N \times d}$  denote *detect query*, which is randomly initialized for detecting new-born objects. In practice, the two kinds of queries are concatenated together and fed into the decoder to learn target representation  $D_t$ :

$$D_t = \text{Decoder}(E_t^l, \text{concat}(Q_t^{det}, Q_t^{tra})) \in \mathbb{R}^{N_t \times d}, \quad (4)$$

where the number of output embedding is  $N_t = N'_{t-1} + N$ , including track objects and detect objects.

**Referent Head.** After a set of decoder layers, we add a referent head on top of the decoder. The referent head includes class, box and referring branches. The class branch is a linear projection, which outputs a binary probability that indicates whether the output embedding represents a true or empty object. The box branch is a 3-layer feed-forward network with ReLU activation except for the last layer. It predicts the box location of all visible instances. Another linear projection acts as the referring branch to produce referent scores with binary values. It refers to the likelihood of whether the instance matches the expression.

## 4.2. Instance Matching and Loss

To train the model, we decouple the final loss as *track loss* and *detect loss*. As described before, our method predicts a flexible-size set of  $N_t$  predictions for the  $t^{\text{th}}$  frame, including  $N'_{t-1}$  *tracking objects* and  $N$  *detection objects*. The  $N'_{t-1}$  tracking objects and their ground-truth are one-to-one matched, while the detection objects are set predictions

Method	HOTA ( $\Delta$ HOTA)	DetA	AssA	DetRe	DetPr	AssRe	AssPr	LocA
FairMOT [59]	22.78( $\pm$ 0.87)	14.43	39.11	16.44	45.48	43.05	71.65	74.77
DeepSORT [45]	25.59( $\pm$ 0.79)	19.76	34.31	26.38	36.93	39.55	61.05	71.34
ByteTrack [58]	24.95( $\pm$ 0.84)	15.50	43.11	18.25	43.48	48.64	70.72	73.90
CStrack [21]	27.91( $\pm$ 0.73)	20.65	39.10	33.76	32.61	43.12	71.82	79.51
TransTrack [40]	32.77( $\pm$ 0.68)	23.31	45.71	32.33	42.23	49.99	78.74	79.48
TrackFormer [31]	33.26( $\pm$ 0.65)	25.44	45.87	35.21	42.19	50.26	78.92	79.63
TransRMOT (Ours)	<b>35.54</b> ( $\pm$ 0.71)	<b>28.25</b>	<b>46.25</b>	<b>39.22</b>	<b>45.94</b>	<b>50.69</b>	<b>80.67</b>	<b>79.79</b>

Table 2. **Quantitative results on Refer-KITTI.** HOTA scores are reported, and  $\Delta$ HOTA presents score variance over three runnings.

(i.e., the number of predictions  $N$  is larger than the number of true new-born objects).

Therefore, we first calculate the track loss using tracking prediction set  $\{\mathbf{c}_{t,i}^{tra}, \mathbf{b}_{t,i}^{tra}, \mathbf{r}_{t,i}^{tra}\}_{i=1}^{N'_{t-1}}$  and the ground-truth set  $\{\hat{\mathbf{c}}_{t,i}^{tra}, \hat{\mathbf{b}}_{t,i}^{tra}, \hat{\mathbf{r}}_{t,i}^{tra}\}_{i=1}^{N'_{t-1}}$  directly. Here,  $\mathbf{c}_{t,i}^{tra} \in \mathbb{R}^1$  is a probability scalar indicating whether this object is visible in the current frame.  $\mathbf{b}_{t,i}^{tra} \in \mathbb{R}^4$  is a normalized vector that represents the center coordinates and relative height and width of the predicted box.  $\mathbf{r}_{t,i}^{tra} \in \mathbb{R}^1$  is a referring probability between the instance and the language description. The track loss  $\mathcal{L}_t^{tra}$  is obtained via one-to-one computation:

$$\mathcal{L}_t^{tra} = \sum_{i=1}^{N'_{t-1}} [\lambda_{cls} \mathcal{L}_{cls}(\mathbf{c}_{t,i}^{tra}, \hat{\mathbf{c}}_{t,i}^{tra}) + \mathcal{L}_{box}(\mathbf{b}_{t,i}^{tra}, \hat{\mathbf{b}}_{t,i}^{tra}) + \lambda_{ref} \mathcal{L}_{ref}(\mathbf{r}_{t,i}^{tra}, \hat{\mathbf{r}}_{t,i}^{tra})], \quad (5)$$

where  $\mathcal{L}_{box}$  weights the L1 loss  $\mathcal{L}_{L_1}$  and the generalized IoU loss  $\mathcal{L}_{giou}$  [35], i.e.,  $\mathcal{L}_{box} = \lambda_{L_1} \mathcal{L}_{L_1} + \lambda_{giou} \mathcal{L}_{giou}$ .  $\mathcal{L}_{cls}$  and  $\mathcal{L}_{ref}$  are the focal loss [23].  $\lambda_{L_1}$ ,  $\lambda_{giou}$ ,  $\lambda_{cls}$ , and  $\lambda_{ref}$  are the corresponding weight coefficients.

Next, for detection objects, we need to find a bipartite graph matching which of the predicted objects fits the true new-born objects. Let  $\mathbf{y}_t^{det} = \{\mathbf{c}_{t,i}^{det}, \mathbf{b}_{t,i}^{det}, \mathbf{r}_{t,i}^{det}\}_{i=1}^N$  denote detection set, and  $\hat{\mathbf{y}}_t^{det}$  denote the new-born ground-truth. Then we search for a permutation of  $N$  predictions  $\delta \in P_n$  by minimizing matching cost:

$$\hat{\delta} = \arg \min_{\delta \in P_n} \mathcal{L}_{match}(\mathbf{y}_{t,\delta(i)}^{det}, \hat{\mathbf{y}}_t^{det}), \quad (6)$$

where  $\mathcal{L}_{match} = \mathcal{L}_{box} + \lambda_{cls} \mathcal{L}_{cls}$ . After obtaining the best permutation  $\hat{\delta}$  with the lowest matching cost, we use it as a new index of predictions  $\{\mathbf{y}_{t,\hat{\delta}(i)}^{det}\}_{i=1}^N$  to compute the detect loss with ground-truth set  $\hat{\mathbf{y}}_t^{det}$ , as similar to Eq. 5:

$$\mathcal{L}_t^{det} = \sum_{i=1}^N [\lambda_{cls} \mathcal{L}_{cls} + \mathbb{1} \mathcal{L}_{box} + \mathbb{1} \lambda_{ref} \mathcal{L}_{ref}], \quad (7)$$

where  $\mathbb{1}$  refers to  $\mathbb{1}_{\{\hat{\mathbf{c}}_{t,i}^{det} \neq \emptyset\}}$ . Eventually, the final loss  $\mathcal{L}^{final}$  is the summation of track loss and detect loss:

$$\mathcal{L}^{final} = \sum_{t=1}^T (\mathcal{L}_t^{tra} + \mathcal{L}_t^{det}). \quad (8)$$

As the first frame has no previous frames, its track query is set to empty. In other words, we only use the detect query to predict all new objects in the first frame.

## 5. Experiments

### 5.1. Experimental Setup

**Model Details.** We adopt visual backbone ResNet-50 [10] and text encoder RoBERTa [26] in our TransRMOT. Similar to Deformable DETR [61], the last three stage features  $\{\mathbf{I}_t^3, \mathbf{I}_t^4, \mathbf{I}_t^5\}$  from the visual backbone are used for further cross-modal learning. Besides, the lowest resolution feature map  $\mathbf{I}_t^6$  is added via a  $3 \times 3$  convolution with spatial stride 2 on the  $\mathbf{I}_t^5$ . Each of the multi-scale feature maps is independently performed the cross-modal fusion. After that, deformable attention in the encoder and decoder integrates the multi-scale features. The architecture and number of the encoder and decoder layer follow the setting of [61]. The number of *detect query* is set as  $N = 300$ .

**Training.** The parameters in the cross-modal module are randomly initialized for training, while the parameters in the text encoder are frozen during training. The remained parameters are initialized from the official Deformable DETR weights [61] pre-trained on the COCO dataset [24]. Random crop is used for data augmentation. The shortest side ranges from 800 to 1536 for multi-scale learning. Moreover, object erasing and inserting are added to simulate object exit and entrance following [57]. The loss coefficients are set as  $\lambda_{cls} = 5$ ,  $\lambda_{L_1} = 2$ ,  $\lambda_{giou} = 2$ ,  $\lambda_{ref} = 2$ . AdamW optimizer is employed to train TransRMOT with base learning rate of  $1e^{-4}$ . The learning rates of the backbone are set to  $1e^{-5}$ . The model is trained for 100 epochs, and the learning rate decays by a factor of 10 at the 50<sup>th</sup> epoch. The overall training is deployed on 8 Nvidia 2080Ti GPUs with batch size of 1.

**Testing.** TransRMOT is able to handle the arbitrary length of videos without post-process. At the  $t^{th}$  frame, it produces  $N_t$  instance embeddings, each corresponding to true or empty objects. We choose these embeddings whose class score exceeds 0.7 to yield true object boxes. Further, the final referent objects are determined from these true objects by a referring threshold  $\beta_{ref} = 0.4$ .

### 5.2. Quantitative Results

On top of Refer-KITTI, we examine the proposed TransRMOT and several competitors in Table 2. Most previous approaches in referring understanding tasks are designed for

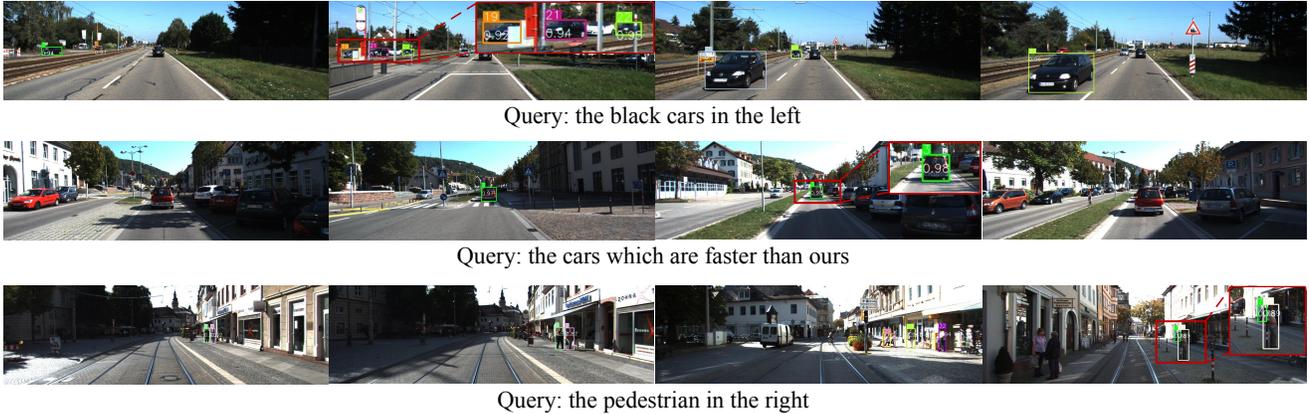


Figure 8. **Qualitative examples on Refer-KITTI.** TransRMOT successfully predicts referent objects according to the given expression.



Figure 9. **Qualitative comparison between all visible objects (top) and the referent objects (bottom).** TransRMOT can capture all visible objects and highlight the referent ones. Please zoom in on the figures for more details.

single-object scenarios, which fail to predict boxes of multiple referent objects. Therefore, we construct a series of CNN-based competitors by integrating our cross-modal fusion module into the detection part of multi-object tracking models, such as FairMOT [59], DeepSORT [45], ByteTrack [58], and CStrack [21]. These competitors follow a *tracking-by-detection* paradigm and employ independent trackers to associate each referent box. More model details can be found in *supplementary materials*. From Table 2, we can see that our TransRMOT outperforms other CNN counterparts by a large margin.

In addition, we compare with Transformer-based works, such as TransTrack [40] and TrackFormer [31], by adding the cross-modal learning parts. As shown in Table 2, both two methods perform worse than our TransRMOT across all metrics but achieve better scores than CNN-based models. Overall, these experiments indicate the model priority of our proposed TransRMOT. Moreover, the stability of Refer-KITTI is also evaluated by training all models three times with different seeds. The slight performance variance (*i.e.*,  $\Delta\text{HOTA} < 0.87$ ) shows great benchmark stability.

### 5.3. Qualitative Results

We visualize several typical referent results in Fig. 8. As seen, TransRMOT is able to detect and track targets accurately under various challenging situations, including multiple objects, status variance, and varying object numbers.

Besides, we provide a qualitative comparison between all predicted objects and the referent objects from TransRMOT in Fig. 9. As observed, all visible objects in the video are detected, and the referent objects are also highlighted based on the given expression query. More qualitative results can be found in *supplementary materials*.

### 5.4. Ablation Study

To study the effect of core components in our model, we conduct extensive ablation studies on Refer-KITTI.

**Cross-modal Fusion.** As described before, the separate early-fusion module is used to model cross-modal representation. To explore its effect, we remove this module to formulate a new model without expressions as input, which will predict all visible objects. As shown in Table 3 (a), the lack of our cross-modal fusion causes a large performance degradation under all metrics (*e.g.*, HOTA: 34.29→17.01, DetA: 28.25→19.05, AssA: 46.25→15.26). We also investigate two different variants. As depicted in Table 3 (a), the first type is to concatenate and input visual and linguistic features into an encoder, as identical with MDETR [14]. The outputted visual part is split and fed into the decoder. The second type (*i.e.*, language as query) sums up sentence-level language embedding with decoder query to probe the corresponding objects. Our method achieves better results than both, demonstrating the effectiveness of our early-fusion cross-modal module.

Fusion Way	HOTA	DetA	AssA	DetRe	DetPr
w/o fusion	17.01	19.05	15.26	22.00	56.07
Concatenation	28.61	22.48	37.69	27.16	51.16
Language as query	33.29	26.15	43.80	35.94	44.83
Ours	<b>35.54</b>	<b>28.25</b>	<b>46.25</b>	<b>39.22</b>	<b>45.94</b>

(a) Comparison on cross-modal fusion way.

Linguistic Extractor	HOTA	DetA	AssA	DetRe	DetPr
FastText [32]	32.39	23.40	46.03	37.03	36.11
Glove [34]	32.45	23.71	46.18	39.02	35.03
Distill-BERT [37]	33.56	26.60	44.01	38.21	43.09
BERT [6]	35.28	28.14	45.73	38.32	47.07
RoBERTa [26]	<b>35.54</b>	<b>28.25</b>	<b>46.25</b>	<b>39.22</b>	<b>45.94</b>

(c) Comparison on linguistic extractor.

Association Way	HOTA	DetA	AssA	AssRe	AssPr
w/o track query	-	26.95	-	-	-
SORT [1]	31.63	24.32	41.50	46.97	79.85
ByteTrack [58]	32.12	24.40	42.33	50.55	75.07
Ours	<b>35.54</b>	<b>28.25</b>	<b>46.25</b>	<b>50.69</b>	<b>80.67</b>

(b) Comparison on cross-frame association way.

Referring threshold	HOTA	DetA	AssA	DetRe	DetPr
0.2	35.10	26.98	<b>47.15</b>	<b>41.77</b>	39.91
0.3	35.07	27.82	45.63	41.74	41.82
0.4	<b>35.54</b>	<b>28.25</b>	46.25	39.22	45.94
0.5	34.73	26.45	47.11	34.16	49.20
0.6	31.09	23.18	43.27	27.47	<b>54.07</b>
0.7	31.63	23.20	44.66	27.75	52.84

(d) Comparison on referring threshold  $\beta_{ref}$ .

Table 3. Ablation studies of different components in TransRMOT. HOTA scores are reported, and ‘-’ means unavailable.

**Cross-frame Association.** It is also of interest to analyze the impact of cross-frame association using the track query. Removing track query (*i.e.*, w/o track query) causes TransRMOT to be a fully image-based model. Table 3 (b) shows that it results in numerous IDs and unavailable associations in terms of metrics AssA, AssRe and AssPr. We associate the referent boxes predicted from the image-based model using state-of-art IoU-matching methods SORT [1] and ByteTrack [58]. Despite achieving association, they have lower HOTA scores than our track query. These experiments approve the necessity of our decoupled query.

**Linguistic Extractor.** Next, we study different linguistic extractors, including the widely-used Transformer-based text encoders (*e.g.*, BERT [6] and Distill-BERT [37]) and the simple word embedding methods (*e.g.*, Glove [34] and FastText [32]). As shown in Table 3 (c), these Transformer-based encoders achieve comparable performance in comparison to the RoBERTa [26], while the simple embedding methods are insufficient in our cross-modal learning.

**Referring Threshold.** At last, we investigate the effect of referring threshold  $\beta_{ref}$ . As reported in Table 3 (d), the HOTA score is marginal at around 0.2~0.5 and begins to have a slight reduction when  $\beta_{ref}$  gets larger. Overall, the referring performance is robust to the varying referring threshold. In this work, we choose  $\beta_{ref} = 0.4$  as default.

### 5.5. Generalization Analysis

As reported in previous works [13, 14, 56], language descriptions have a significant advantage in recognition generalization. Even if a new expression does not exist in the dataset, the referring understanding model can reason the referent objects by learning existing language knowledge. To verify this point, we employ a new expression, ‘the left persons in black’, to test TransRMOT. Although Refer-KITTI contains some succinct expressions, *e.g.*, ‘the left persons’ and ‘the persons in black’, the new expression is not included in the whole dataset. In Fig. 10, TransRMOT can correctly recognize the referent objects, showing

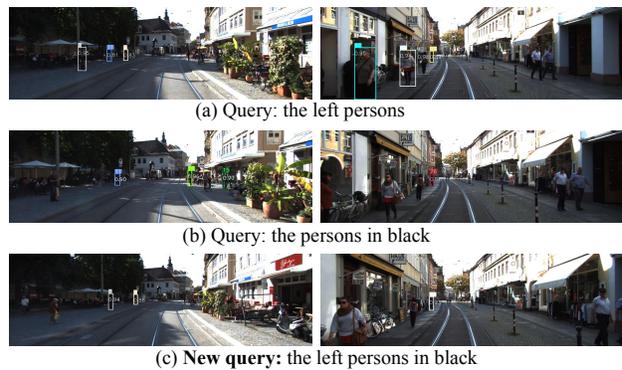


Figure 10. Generalization analysis of TransRMOT. The new query (c) is not included in dataset, but our model can still infer the referent objects according to the existing knowledge (a) (b).

the powerful generalization ability of TransRMOT.

## 6. Conclusion

In this paper, we proposed a novel referring understanding task, called Referring Multi-Object Tracking (RMOT). It addressed the single-object limitation of referring understanding tasks and provided a more flexible multi-object setting. Additionally, it leveraged the essential temporal status variant into referring understanding. Both two new settings make RMOT more general, which is appropriate for evaluating real-world requirements. To promote RMOT, we developed a new benchmark, named Refer-KITTI. The benchmark provided high flexibility with referent objects and high temporal dynamics but yielded low labeling costs. Furthermore, we proposed a Transformer-based method, TransRMOT, to tackle the new task. The framework is fully end-to-end optimized during training, and predicts referent objects frame by frame. We validated TransRMOT on Refer-KITTI, and it achieved state-of-art performance.

## References

- [1] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *ICIP*, 2016. 8
- [2] Adam Botach, Evgenii Zheltonozhskii, and Chaim Baskin. End-to-end referring video object segmentation with multi-modal transformers. In *CVPR*, 2022. 3
- [3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2, 5
- [4] Zhenfang Chen, Lin Ma, Wenhan Luo, and Kwan-Yee K Wong. Weakly-supervised spatio-temporally grounding natural sentence in video. In *ACL*, 2019. 2
- [5] Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Luc Van Gool, and Marie-Francine Moens. Talk2car: Taking control of your self-driving car. *arXiv preprint arXiv:1909.10838*, 2019. 1, 2
- [6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 8
- [7] Zihan Ding, Tianrui Hui, Junshi Huang, Xiaoming Wei, Jizhong Han, and Si Liu. Language-bridged spatial-temporal interaction for referring video object segmentation. In *CVPR*, 2022. 3
- [8] Kirill Gavriluk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *CVPR*, 2018. 3
- [9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012. 2, 3
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6
- [11] Shaofei Huang, Tianrui Hui, Si Liu, Guanbin Li, Yunchao Wei, Jizhong Han, Luoqi Liu, and Bo Li. Referring image segmentation via cross-modal progressive comprehension. In *CVPR*, 2020. 3
- [12] Tianrui Hui, Shaofei Huang, Si Liu, Zihan Ding, Guanbin Li, Wenguan Wang, Jizhong Han, and Fei Wang. Collaborative spatial-temporal modeling for language-queried video actor segmentation. In *CVPR*, 2021. 3
- [13] Dat Huynh, Jason Kuen, Zhe Lin, Jiuxiang Gu, and Ehsan Elhamifar. Open-vocabulary instance segmentation via robust cross-modal pseudo-labeling. In *CVPR*, 2022. 8
- [14] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *ICCV*, 2021. 3, 5, 7, 8
- [15] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 1, 2
- [16] Ilker Kesen, Ozan Arkan Can, Erkut Erdem, Aykut Erdem, and Deniz Yüret. Modulating bottom-up and top-down visual processing via language-conditional filters. In *CVPR*, 2022. 3
- [17] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *ACCV*, 2018. 1, 2, 3
- [18] Muchen Li and Leonid Sigal. Referring transformer: A one-step approach to multi-task visual grounding. *NeurIPS*, 2021. 3
- [19] Zhenyang Li, Ran Tao, Efstratios Gavves, Cees GM Snoek, and Arnold WM Smeulders. Tracking by natural language specification. In *CVPR*, 2017. 1, 2
- [20] Chen Liang, Wenguan Wang, Tianfei Zhou, Jiaxu Miao, Yawei Luo, and Yi Yang. Local-global context aware transformer for language-guided video segmentation. *IEEE TPAMI*, 2023. 3
- [21] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, Shuyuan Zhu, and Weiming Hu. Rethinking the competition between detection and reid in multiobject tracking. *IEEE TIP*, 2022. 6, 7
- [22] Yue Liao, Si Liu, Guanbin Li, Fei Wang, Yanjie Chen, Chen Qian, and Bo Li. A real-time cross-modality correlation filtering method for referring expression comprehension. In *CVPR*, 2020. 3
- [23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 6
- [24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 6
- [25] Daqing Liu, Hanwang Zhang, Feng Wu, and Zheng-Jun Zha. Learning to assemble neural module tree networks for visual grounding. In *ICCV*, 2019. 3
- [26] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 6, 8
- [27] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *IJCV*, 2020. 4
- [28] Gen Luo, Yiyi Zhou, Xiaoshuai Sun, Liujuan Cao, Chenglin Wu, Cheng Deng, and Rongrong Ji. Multi-task collaborative network for joint referring expression comprehension and segmentation. In *CVPR*, 2020. 3
- [29] Ruotian Luo and Gregory Shakhnarovich. Comprehension-guided referring expressions. In *CVPR*, 2017. 3
- [30] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 3
- [31] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *CVPR*, 2022. 6, 7
- [32] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Fuhrsch, and Armand Joulin. Advances in pre-training distributed word representations. In *LREC*, 2018. 8
- [33] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *ECCV*, 2016. 1, 3
- [34] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In

- EMNLP*, 2014. 8
- [35] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. 6
- [36] Nivedita Rufus, Unni Krishnan R Nair, K Madhava Krishna, and Vineet Gandhi. Cosine meets softmax: A tough-to-beat baseline for visual grounding. In *ECCV*, 2020. 3
- [37] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. 8
- [38] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *ECCV*, 2020. 1, 2, 3
- [39] Sijie Song, Xudong Lin, Jiaying Liu, Zongming Guo, and Shih-Fu Chang. Co-grounding networks with semantic attention for referring expression comprehension in videos. In *CVPR*, 2021. 3
- [40] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020. 6, 7
- [41] Reuben Tan, Bryan Plummer, Kate Saenko, Hailin Jin, and Bryan Russell. Look at what i'm doing: Self-supervised spatial grounding of narrations in instructional videos. *NeurIPS*, 2021. 3
- [42] Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. Object referring in videos with language and human gaze. In *CVPR*, 2018. 1, 2
- [43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 5
- [44] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *CVPR*, 2019. 3
- [45] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, 2017. 6, 7
- [46] Dongming Wu, Xingping Dong, Ling Shao, and Jianbing Shen. Multi-level representation learning with semantic alignment for referring video object segmentation. In *CVPR*, 2022. 3
- [47] Jiannan Wu, Yi Jiang, Peize Sun, Zehuan Yuan, and Ping Luo. Language as queries for referring video object segmentation. In *CVPR*, 2022. 3
- [48] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Tubedetr: Spatio-temporal video grounding with transformers. In *CVPR*, 2022. 3
- [49] Sibe Yang, Guanbin Li, and Yizhou Yu. Cross-modal relationship inference for grounding referring expressions. In *CVPR*, 2019. 3
- [50] Sibe Yang, Guanbin Li, and Yizhou Yu. Graph-structured referring expression reasoning in the wild. In *CVPR*, 2020. 3
- [51] Sibe Yang, Meng Xia, Guanbin Li, Hong-Yu Zhou, and Yizhou Yu. Bottom-up shift and reasoning for referring image segmentation. In *CVPR*, 2021. 3
- [52] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. In *CVPR*, 2019. 3
- [53] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2014. 1, 2
- [54] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018. 3
- [55] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, 2016. 1, 2
- [56] Alireza Zareian, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. Open-vocabulary object detection using captions. In *CVPR*, 2021. 8
- [57] Fangao Zeng, Bin Dong, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *ECCV*, 2021. 2, 4, 6
- [58] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *ECCV*, 2022. 6, 7, 8
- [59] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *IJCV*, 2021. 6, 7
- [60] Tianfei Zhou, Fatih Porikli, David J Crandall, Luc Van Gool, and Wenguan Wang. A survey on deep learning technique for video segmentation. *IEEE TPAMI*, 2022. 3
- [61] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020. 4, 6