2024 IEEE International Conference on Robotics and Automation (ICRA)
May 13-17, 2024. Yokohama, Japan

2024 IEEE International Conference on Robotics and Automation (ICRA) | 979-8-3503-8457-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICRA57147.2024.10610979

Multi-modal 3D Human Tracking for Robots in Complex Environment with Siamese Point-Video Transformer

Shuo Xin[1], Zhen Zhang[1], Mengmeng Wang[1],
Xiaojun Hou[1], Yaowei Guo[1], Xiao Kang[2], Liang Liu[1]*, Yong Liu[1]*

Abstract— **Tracking a specific person in 3D scene is gaining momentum due to its numerous applications in robotics. Currently, most 3D trackers focus on driving scenarios with neglected jitter and uncomplicated surroundings, which results in their severe degeneration in complex environments, especially on jolting robot platforms (only 20-60% success rate). To improve the accuracy, a Point-Video-based Transformer Tracking model (PVTrack) is presented for robots. It is the first multi-modal 3D human tracking work that incorporates point clouds together with RGB videos to achieve information complementarity. Moreover, PVTrack proposes the Siamese Point-Video Transformer for feature aggregation to overcome dynamic environments, which captures more target-aware information through the hierarchical attention mechanism adaptively. Considering the violent shaking on robots and rugged terrains, a lateral Human-ware Proposal Network is designed together with an Anti-shake Proposal Compensation module. It alleviates the disturbance caused by complex scenes as well as the particularity of the robot platform. Experiments show that our method achieves state-of-the-art performance on both KITTI/Waymo datasets and a quadruped robot for various indoor and outdoor scenes.**

I. INTRODUCTION

3D human tracking is to distinguish an arbitrary person from consecutive frames, with not only the position but also 3D size and heading angle of the target. Along the process of robotics and automation, it has been an essential building block to many advanced applications in robotic fields, involving intelligent warehousing, human-machine collaboration, surveillance, and so on [1, 2]. For example, when accomplishing freight services, robot needs to follow a specific worker along the way. Here a core step that links to the accuracy of all subsequent modules is to frame out the human as accurately as possible, which considerably facilitates the robot in follow-up actions, as shown in Fig. 1.

Prior arts on 3D tracking [3–6] mainly follow the Siamese paradigm, which calculates the similarity between a canonical target template and searching area according to the geometric matching and have achieved fruitful accuracy on open-source datasets like KITTI [7] and Nuscenes [8]. Whilst, there are still three crucial bottlenecks that remain notable. (1) Severer degeneration in complex environments. (2) Low accuracy for the human category. (3) Incapacity to violent shaking when migrating to robotic platform.

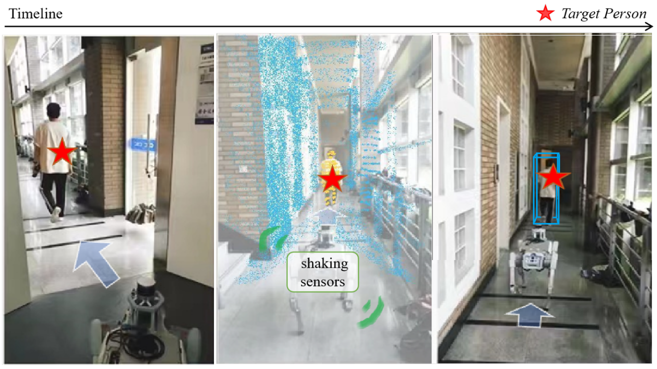

Fig. 1. An example scenario (left) of target-person tracking tasks on a quadruped robot. Even if the platform shakes violently, our system is capable of integrating multi-modalities from points and video frames (middle) to generate 3D bounding boxes of the specific person frame by frame (right).

The first challenge is mainly caused by disturbances in complex scenarios like illumination, distortion, and occlusion, which lead to huge accuracy fluctuations in different tracking scenarios. We note that almost all state-of-the-art 3D trackers are point-cloud based but the single sensor will inevitably limit the potential of tracking algorithms. Due to the lack of vital appearance in pure 3D trackers, they always neglect comprehensive enough indications so frequently miss targets. To address the problem, we combine the geometric and appearance features of point clouds and video frames together to build a unified 3D tracker.

Secondly, low tracking accuracy in the human category is still a long-standing problem. Recent studies have focused on generic tracking tasks like cars and trunks while extreme deterioration for humans. The size of the human target in the point cloud space is relatively small and it is easy to deform due to the non-rigid body, making even the latest paper M^2-Track gain only 61.5% success rate in KITTI pedestrians. For higher accuracy, we focus on the human-specific characteristic by designing a well-designed Human-ware Proposal Network and inheriting the efficient Attention mechanism from Transformer. Our modification boosts the precision by around 10% significantly.

For the third challenge, the robustness of current 3D trackers is unsatisfactory once the application scenario is changed to a physical robotic platform, such as a quadruped robot trotting and jumping. There are also a number of techniques to track objects with mobile robots [9–14], while most of them are designed for 2D tracking lacking depth measurements. In this paper, we developed an Anti-shake Proposal Compensation Module which capable of robotic vibrations. It is worth noting that we chose the quadruped

[1] Authors are with the Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou, China.

[2] Author is with China North Vehicle Research Institute, Beijing, China.

* Liang Liu and Yong Liu are the corresponding authors: (Email: leonliuz@zju.edu.cn; yongliu@iipc.zju.edu.cn).

robot as the representative bumpy platform.

To inspire research on this topic, our paper proposes an efficient 3D human tracking model based on the Siamese Point-Video Transformer. We well-designed the multi-sensor fusion on a robotic platform, which not only meets the practical requirements of tracking humans in various scenarios but also significantly reduces the missing rate for dynamic moving robotics. Additionally, we newly constructed a 2D+3D dataset on a quadruped robot, 'PVT3D', which contains 30 video sequences with dense 3D point clouds and more than 20 challenging scenarios indoors and outdoors.

Our contributions are three-fold:

- A multi-modal 3D human tracking framework is proposed that jointly integrates point clouds with RGB video through the feature alignment and fusion operator, yielding a great tolerance to complex environments.
- A novel Siamese Point-Video Transformer is proposed in a longer-contextual perspective, enabling a more global separation between target and background to further improve tracking accuracy.
- A goal-conditioned Anti-shake Proposal Compensation Module is well-designed with a Human-ware Proposal Network to overcome the violent shaking of robots and the following target missing.
- We successfully migrated our method from two open-source datasets KITTI/Waymo to the physical quadruped robot platform while running in real-time.

II. Related Works

A. 3D Single Object Tracking

LiDAR-only. As a brand-new task emerged in recent years, LiDAR-based 3D single object tracking (SOT) is mainly deep-learning-based. The pioneering SC3D [15] first proposed a shape completion strategy focuses on extending the 2D to the 3D tracker, but it can't achieve end-to-end training. To improve it, Siamese-like paradigms [16–18] are carried out unprecedentedly. The derived articles focus on either improving the point-wise correlation matchers by feature enhancement [19] or designing more robust prediction decoders [20, 21] with sophisticated structures. During the past few years, the success of vision Transformer [22–24] and point Transformer [25, 26] stimulated numerous attempts [4, 6, 27–29] to embed the Transformers into the 3D tracker designing. However, due to the lack of color and texture of the point set, the LiDAR-only tracking misses target when facing disturbances like occlusion, light change, and so on.

Multi-Modality. Plenty of studies attempted to merge visual contexts with other sensors in the 2D object detection and tracking fields [30–35]. However, the multisensor application in 3D SOT is still at the primary stage. Wang et al. [13] first fused RGB with ultrasonic data through the extended Kalman filter. Other papers [36–40] tried RGB-D methods to complement the cardinal lacking information. As far as we know, the Point-Video-based design for 3D tracking is few with only [41, 42]. What's more, they are not human-specific, so this paper comes just in time.

B. Human Tracking on robot

During the past few years, there are also several on-robot methods for tracking people [2, 11–13, 43]. Zhang et al. [2] first complete a human-following task for quadruped robots. Lin et al. [44] followed a moving target tracking system on quadrotors with Visual-Inertial Localization. However, most of them rely on visual tracking and lack depth information, which limits them to estimating three-dimensional direction and spatial distance between humans and robots.

III. Methodology

We model the 3D human tracking problem as a bottom-up learning problem for a specific person. As shown in Fig. 2, the proposed Point-Video Transformer Tracking (abbreviated as PVTrack) mainly consists of three parts: (1) Multi-Modal feature extraction and fusion, (2) Siamese Point-Video Transformer, and (3) 3D Human-ware Proposal and Verification. To handle the problem of incomplete target, PVTrack utilizes the Siamese two-branch pipeline to embed the target template (initialized in the first target bounding box and updated with the previous frame's prediction) to enhance the object features in the search area. In this formulation, given a dynamic 3D sequence of T frame point clouds $P = \{p_i\}_{i=1}^{T}$ and a 3D target bounding box (3DBBox) $\mathbf{b}_1$ as template in the first frame, our goal is to localize the same target $\mathbf{b}_i = \{\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{z}_i, \boldsymbol{w}_i, \boldsymbol{h}_i, \boldsymbol{l}_i, \boldsymbol{\theta}_i\}_{i=1}^{T}$ in the search area for the sequential $T-1$ frames. The whole pipeline is end-to-end trained with only a single stage.

A. Multi-Modal Feature Extraction and Fusion

Since LiDAR points and RGB videos have different view representations for the same scene, our goal for this section is to achieve multi-modality alignment and fusion while realizing the full complementarity of two inputs. Assuming that the initial input point cloud is $P \in \mathbb{R}^{N\times 3}$ (a point position sequence with N points), and the RGB image of a certain frame of video is $V \in \mathbb{R}^{H\times W\times 3}$. After inputting point clouds and video frames into the network, most Siamese trackers will directly use local descriptor networks like PointNet++ [45] as their feature extractors. However, the traditional backbone is time-consuming and computationally huge. To improve, we replaced them with two lightweight Transformer modules, which greatly reduces computational complexity while acquiring discriminative features. Specifically, the modified Point-MAE [46] is used to extract point features, and the MobileFormer [47] is used to extract video frame ones. The two networks separately output $F^p = \{f_i^p\}_{i=1}^{K} \in \mathbb{R}^{K\times C_1}$ and $F^v = \{f_{(i,j)}^v\} \in R^{H'\times W'\times C_2}$ as point feature and video frame feature, where K denotes the number of point feature groups after point down-sampling and $H' \times W'$ denotes the dimension of feature map. C_1 and C_2 denote the number of LiDAR channels and RGB channels, respectively.

Point-Video Feature Alignment and Fusion. As mentioned above, the point feature and video frame feature differ substantially in dimension, and the two modalities are not aligned optimally for a unified fusion. One intuitive solution

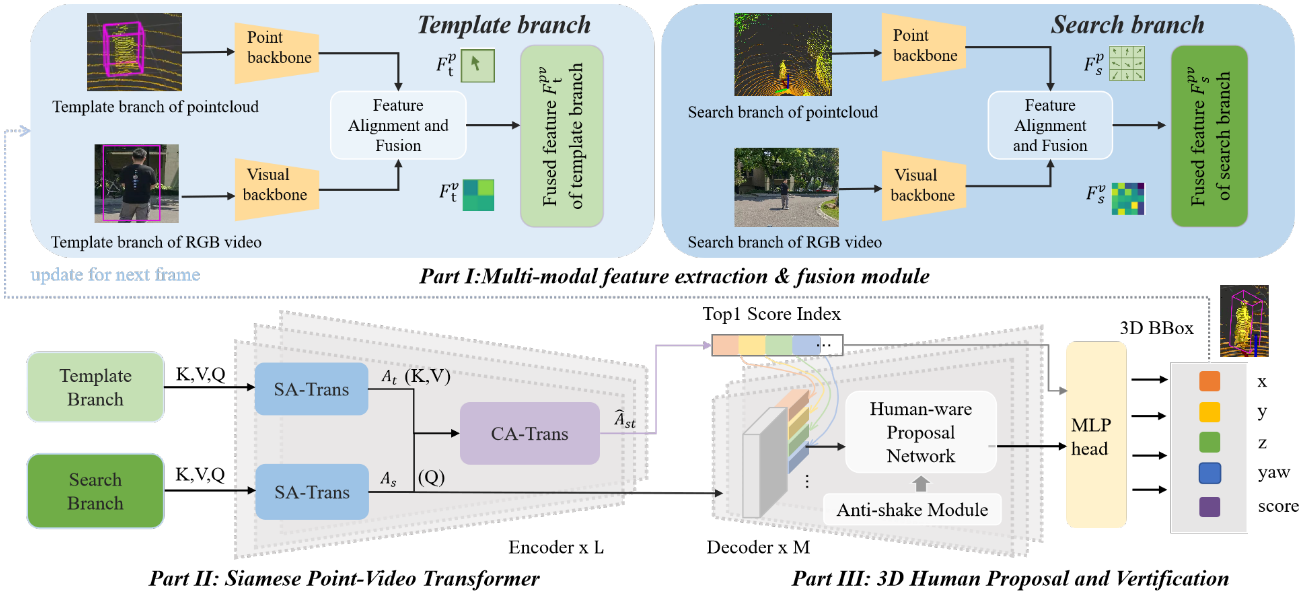
Template branch
Point backbone
Template branch of pointcloud
Feature Alignment and Fusion
Visual backbone
Template branch of RGB video
Fused feature F_t^{pv} of template branch
Search branch
Point backbone
Search branch of pointcloud
Feature Alignment and Fusion
Visual backbone
Search branch of RGB video
Fused feature F_s^{pv} of search branch
update for next frame
Part I:Multi-modal feature extraction & fusion module
Template Branch
K,V,Q
SA-Trans
A_t (K,V)
CA-Trans
$\widehat{A}_{st}$
Search Branch
K,V,Q
SA-Trans
A_s
(Q)
Encoder x L
Top1 Score Index
Human-ware Proposal Network
Anti-shake Module
Decoder x M
MLP head
3D BBox
x
y
z
yaw
score
Part II: Siamese Point-Video Transformer
Part III: 3D Human Proposal and Vertification

Fig. 2. The overall framework of the proposed PVTrack. First, modified PointMAE and MobileFormer are separately used as backbones to extract multi-modal features, and the feature alignment and fusion operator is proposed to better integrate them. Then we propose a Siamese Point-Video Transformer to embed the template branch into the search area by similarity-based matching. With the augmented score indices, the third part uses a human-aware proposal network to quickly output the final tracking results. The K,Q,V is short for Key, Query and Value.

is to artificially add a new dimension by repeating the point feature multiple times until it is consistent with video frame features. However, this approach loses a significant amount of information on the spatial correspondence between the two modalities. In our approach, we propose the Point-Video Feature Alignment and Fusion operation (PVAF).

First, a general farthest point sampling (FPS) is applied to output K point groups in total with corresponding center points $\{c_i\}_{i=1}^{K}$. Since the center point and group are not uniformly dispersed in space, PVAF then interpolates the point feature $\{f_i^p\}_{i=1}^{K}$ back to each point $\{p_i'\}_{i=1}^{N}$ in the original point clouds utilizing inverse distance weight w_j. The process can be described as:

$$p_i' = \sum_{j=1}^{K} w_j f_j^p, w_j = \frac{\frac{1}{\|p_i - c_j\|_2 + \epsilon}}{\sum_{k=1}^{K} \sum_{t=1}^{N} \frac{1}{\|p_t - c_k\|_2 + \epsilon}} \tag{1}$$

Next, PVAF translates the interpolated $p' \in \mathbb{R}^{N \times C_1}$ into 2D coordinates $\hat{p} \in \mathbb{N}^{H' \times W' \times C_3}$ based on LiDAR-camera settings where C_3 denotes the number of projected point feature channels. All the feature patch $\hat{p}_{(i,j)}$ in 2D plane generate a projected feature map $f^p \in \mathbb{R}^{H' \times W' \times C_3}$ that has the same dimension as the video-wise feature f^v. Finally, we utilize the multi-layer perception (MLP) $\chi(.)$ to extract interaction information and concatenate two modality features as the point-video-fused feature denoted $F^{pv} = \{f_{(i,j)}^{pv}\} \in \mathbb{R}^{M \times C}$, where $M = H' \times W'$and $C = C_2 + C_3$.

$$f_{(i,j)}^{pv} = \chi^p(f_{(i,j)}^p) \oplus \chi^v(f_{(i,j)}^v) \tag{2}$$

So far, the problem of modality misalignment has been solved. With the dedicated combination, PVTrack can catch more target-ware valuable tracking details in a joint manner. For example, when two people cross each other, only through a bunch of point sets is difficult to distinguish which one is the target person since they are very close to each other in the point cloud view. At this time, through the additional RGB channel, the color of two persons' clothing might be a great auxiliary in locating the target, leading to more accurate tracking results.

B. Siamese Point-Video Transformer

Another major breakthrough of PVTrack is the hierarchical attention mechanism from Transformer, allowing encoders to concentrate more on target-aware feature aggregation in a global view. As shown in Fig. 3, PVTrack significantly alleviates the inability of general convolutions when capturing key data (e.g. shape information) through obtaining long-range interactivity. Under the seq2seq framework, the hierarchical encoder consists of two main modules: Self-Attention Transformer (SA-Trans) and Cross-Attention Transformer (CA-Trans).

SA-Trans. This module is used to thoroughly fuse the adjacent features of point clouds and videos that were physically stitched in the previous section together. On each encoder layer, SA-Trans uses linear projection layers to transform the input feature vectors ("Query", "Key", "Value") and calculate the dot products. Subsequently, the attention map is normalized with a Softmax operation and outputs the augmented Z. The proposed process can be formulated as:

$$\mathrm{SA(Q, K, V)} = \phi(\mathrm{Q} - \mathrm{softmax(Z)} \cdot (W_v)\mathrm{V}) \tag{3}$$

where ϕ represents the linear layer and ReLU operation applied to the output features, the attention matrix Z is obtained by input vectors Q, K and linear projections W_k, W_q:

$$\mathrm{Z} = \bar{\mathrm{Q}} \cdot \bar{\mathrm{K}}^T = \frac{W_q\mathrm{Q}}{\|W_q\mathrm{Q}\|_2} \cdot \frac{W_k\mathrm{K}}{\|W_k\mathrm{K}\|_2} \tag{4}$$

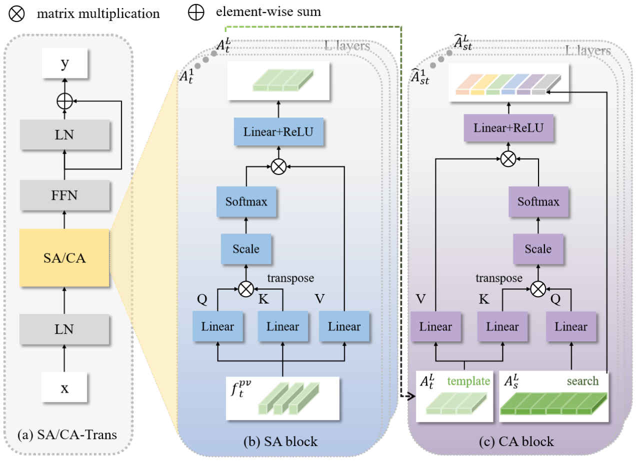

Fig. 3. Illustration of the SA-Trans and CA-Trans module. “LN” is short for Layer Normalization and “FFN” is short for Feed Forward Network. In SA block, we gain the template-branch attention maps $\{A_t^i\}_{i=1}^L$ from the template branch featuref_t^{pv}. The same goes for the search branch. In CA block, we gain the augmented attention map $\{\hat{A}_{st}^i\}_{i=1}^L$ that integrates two branches.

To avoid the dominance of a few feature channels with large magnitudes, we then utilize an L2-normalization together with a feed-forward network. Using shared weights, the point-video-fused features of search branch f_s^{pv} and of template branchf_t^{pv} are projected onto the same latent space, yielding search-branch attention maps A_s and template-branch ones A_t:

$$A_s = \text{SA}(f_s^{pv}, f_s^{pv}, f_s^{pv}), A_t = \text{SA}(f_t^{pv}, f_t^{pv}, f_t^{pv}) \qquad (5)$$

CA-Trans. The network architecture of CA-Trans follows the same pattern as SA-Trans, except for differing input heads and the addition of residual processing to the output. By learning the similarity between the template and search area, CA-Trans embeds the template as “Key” and “Value” into the “Query” search area to generate the refined attention map $\hat{A}_{st}$ to more accurately pinpoint the likely target in the search region. The multi-layers is indeed an iterative process that learns coarse-to-fine cross-correlation attention between the two branches, which are indicated as follows:

$$\hat{A}_{st} = \text{CA}(A_s, A_t, A_t) \qquad (6)$$

C. 3D Human Proposal and Vertification

In this part, we construct a 3D Human-ware Proposal Network (HPN) with human-specific priors as decoders. Note that the final encoder generates refined $\hat{A}_{st}^L$, the i column representing the most similar template feature to the i-th search feature. So it's indeed a top score indices. Using the index, we concatenate the template with the corresponding search ones in $\hat{A}_s^L$, yielding an intermediate representation of size $M \times (C + C)$. After feeding it into HPN, the coarse proposals are generated through the P2B-like network [28], including potential center generation and clustering. Moreover, we add some essential human-specific priors to further correct them. That is, HPN filters proposals by the size requirements (width: 10-40cm, height: 1.0-2.0m, length: 20-60cm), and the oversized ones will be immediately discarded. This process significantly speeds up the module and iteratively update the target sequence through multi-layers. However, some targets are still missed as a result of the robot platform's violent shaking.

TABLE I

THE ANTI-SHAKE PROPOSAL COMPENSATION MODULE ON ROBOT

```
Input: current frame $I_t$, response threshold $\Omega$, 3DBBox of previous 10 frames $\mathbf{X}\{x, y, z\}, \mathbf{bb}\{w, h, l\}$ with response score $\mathbf{R}\{r\}$
1. Initialization: When the target is missing, make $n = 0$
2. Calculate target speed: $S = max\{5, \sum_{i=t-9}^{t-1} \|\mathbf{X}_i - \mathbf{X}_{i-1}\|_2\}$
3. while(max response $r^n < \Omega \times mean(r^i)$) do:
      add $W^n = n * S + 2 * bb(1)$
      add $H^n = n * S + 2 * bb(2)$
      add $L^n = n * S + 2 * bb(3)$
      extract points in $I_t^{W^n \times L^n \times H^n} \subset I_t$ around center$\mathbf{X}_{i-1}$
      find the max response $r^t$ in $I_t^{W^n \times L^n \times H^n}$ as new $r^n$
      turn to the next frame: $n = n + 1$
   end while
4: Output 3D proposals of the missing frames and resume tracking
```

Anti-shake Proposal Compensation Module. To better overcome the specificity of the quadruped robot platform, we further introduce an Anti-shake Proposal Compensation assisted with stabilization mechanism to fix the missing proposals. It follows two principles: (1) people have limited speed so they are around spatial locations previously observed, and (2) the possible proposal region generally expands at a velocity proportional to the average velocity. Using the robot's integrated speed-measuring module, we can get the target velocity of the previous frame along with its next move trends. Once the tracking object disappears due to violent platform shaking or rugged terrain, the prediction algorithm will be triggered by adaptively expanding the candidate area as Table. I shown. While keeping up with the approximate position of the object until reappears, the algorithm also reduces the computational duplication of analyzing irrelevant spatial regions.

Finally, with M proposals generated above, we utilize MLP head to select the highest-score one as the final tracking result. The output prediction of t-th frame will be used as the template for the subsequent consecutive frame.

D. Loss functions

The whole sequence with T frames is calculated by ground-truths $Y = \{y^t\}_{t=1}^T$ and the prediction from PVTrack $\hat{Y} = \{\hat{y}^t\}_{t=1}^T$ using the matching variance ω.

$$\mathcal{L}^{total}(\hat{Y}|_\omega, Y) = \frac{\sum_{t=1}^T \mathcal{L}(\hat{y}^t|_\omega, y^t)}{T} \qquad (7)$$

For the t-th frame, $\mathcal{L}$ consists of the binary cross-entropy loss $\mathcal{L}_{cls}$ for confidence prediction to distinguish the foreground and the Smooth-L1 loss $\mathcal{L}_{reg}$ for 3DBBox regression.

$$\mathcal{L}(\hat{y}^t|_{\omega^i}, y^t) = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{reg} \sum \mathcal{L}_{reg}(\hat{\mathbf{b}}, \mathbf{b}) \qquad (8)$$

where $\mathbf{b}$ consists of $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{w}, \boldsymbol{h}, \boldsymbol{l}, \boldsymbol{\theta})$ for the target person that are calculated by $\Delta \boldsymbol{x} = \frac{\hat{\boldsymbol{x}} - \boldsymbol{x}}{d}, \Delta \boldsymbol{y} = \frac{\hat{\boldsymbol{y}} - \boldsymbol{y}}{d}, \Delta \boldsymbol{z} = \frac{\hat{\boldsymbol{z}} - \boldsymbol{z}}{h}, \Delta \boldsymbol{\theta} = \sin(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})$, and $d = \sqrt{\boldsymbol{w}^2 + \boldsymbol{l}^2}$. λ_{cls} and λ_{reg} are the corresponding weight coefficients.

IV. Experiments

A. Experimental Setup

In the experimental part, We validate the efficiency of PVTrack on the quadruped robot platform JueYing X20, illustrated in Fig.4. The robot is equipped with an onboard computing device, NVIDIA Jetson Xavier NX (21 TOPS, 16GB), a bottom LiDAR (RoboSense, 16-beam), a mounted LiDAR(Hesai PandarQT, 64-beam), and a forward-facing camera (Intel RealSense D455), publishing point cloud measurements at 10 Hz and video frames at 30 Hz, respectively.

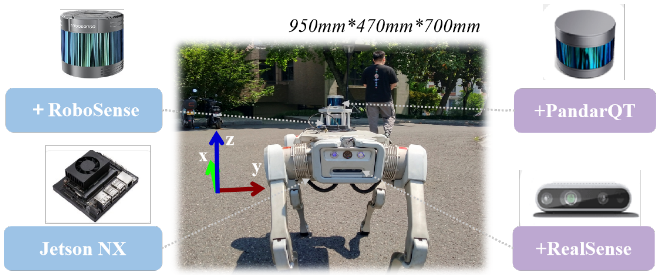

Fig. 4. The JueYing X20 robot platform and its onboard sensors.

Datasets. We use two kinds of tracking datasets to evaluate PVTrack. First is the open-source SOT datasets (**KITTI Tracking** [7] and **Waymo** [48]), which are widely adopted for general 3D tracking tasks. But only using them remains limited because they just focus on driving scenarios rather than the robotic perspective. Based on it, we provide a new on-robot dataset called **PVT-3D** that contains 30 sequences of videos and over 10000 frames of point cloud collected from the campus. Compared with existing datasets, **PVT-3D** represents the 3D human tracking more effectively: 1) This is the first 2D+3D human-target tracking dataset designed for the quadruped robot with synchronized RGB and LiDAR information. 2) It achieves a high diversity in 20+ challenging scenarios including indoor narrow corridors, outdoor rough terrains, static and dynamic obstacles, etc.

Evaluation Protocol. We follow the One Pass Evaluation (OPE) as the evaluation metric to calculate the center bias and 3D IoU between the predicted and ground-truth BBox. The success and precision scores are used as the final metric.

Implementation Details. For our network, we use the Adam optimizer with batch size 256 and an initial learning rate of 0.001 for training, which decayed by 10 times every 20 epochs. We set N^t to 1024 and N^s to 2048 for the input template and search regions by randomly duplicating and discarding points. The layer number of the encoders L and decoders M is set to 3, and The coefficients for the loss terms are $\lambda_{cls} = 0.5$ and $\lambda_{reg} = 0.5$. All experiments are conducted on the same system with an Intel Core™ i7-9700 CPU and a Nvidia GTX 1080Ti GPU.

B. 3D Tracking on the KITTI/Waymo Dataset

The KITTI and Waymo tracking dataset have 6088 frames and 510533 frames separately for the pedestrian category. All scenarios are split into training, validation, and test sets in a 9:1:1 ratio. Table.II gives the comparison results of PVTrack with state-of-the-art (SOTA) methods. It is worth noting that most 3D trackers, even the latest STNet and M^2-Track, only gain prediction scores of 40-60 for KITTI pedestrians and lower for Waymo pedestrians. Compared with them, our human-ware PVtrack achieves the highest score, in terms of both Success (73.3%) and Precision (86.9%). The experimental results demonstrate that the 3D human tracking field is still at a very low level, and urgently needs to be further improved. Our PVTrack innovates right on it while significantly boosting the precision by around 10%.

TABLE II
Comparison among our PVTrack and the State-of-the-art Methods on the KITTI and Waymo Tracking datasets

Tracker	KITTI pedestrian		Waymo pedestrian		run
	Success	Precision	Success	Precision	(ms)
SC3D [15]	18.2	37.8	14.2	16.2	542
P2B [49]	28.7	49.6	15.6	29.6	23.6
PSRCNN [50]	48.2	75.2	27.8	60.6	36.7
PTTR [28]	50.9	81.6	/	/	**19.9**
STNet [4]	49.9	77.2	38.1	73.2	28.6
M^2-Track [5]	61.5	86.2	42.1	67.3	/
PVTrack(Ours)	**73.3**	**86.9**	**61.3**	**74.1**	20.8

Runtime Analysis. We test the model inference time with hardware in IV-A. Under the same configurations, PVTrack achieves the second-fastest runtime of 20.8 ms, including 8ms for processing dual inputs, 12.8 ms for network forward propagation and 0.5 ms for post-processing.

C. 3D Tracking on the Quadruped Robot

This paper also verifies that PVTrack can be better migrated to quadruped robot tasks. In terms of scene selection, as shown in Figure 5 (a-d), we used more than ten random scenes such as open areas, up and down steep slopes, tree-filled groves, and indoor scenes. In each scene, a target person walks normally at a constant speed and JueYing robot follows closely with an adaptive gait. The results show that regardless of the laser-beam numbers and the terrain changes, it achieves a tracking success rate of more than 80% within only 256 training episodes. Besides, PVTrack is fast enough for real-time robot interaction with 35 fps. Extensive evaluations on robots are given in Table. III.

Generalization to Difficult Scenes. In addition to random scenes, this paper also conducts physical verification on many specially designed difficult scenes in PVT-3D (the last row in Figure 5), like the multi-people occlusion when passing the corner and the overexposure under street lights, etc. Especially in night scenes as Fig.5 (f) shows, the human object in videos flickers and blurs significantly which fails most other trackers. Remarkably, our PVTrack effectively suppresses the influence of illumination changes with the help of multi-modalities, and the red 3D bounding box always follows the human body consistently more than 75% of the time, which clearly confirms the robustness of the dynamic human tracking system to complex environmental changes.

Generalization to different Human Objects. Sometimes due to the laser beam's sparsity and the robot's height limitation, the point clouds of the human body cannot be

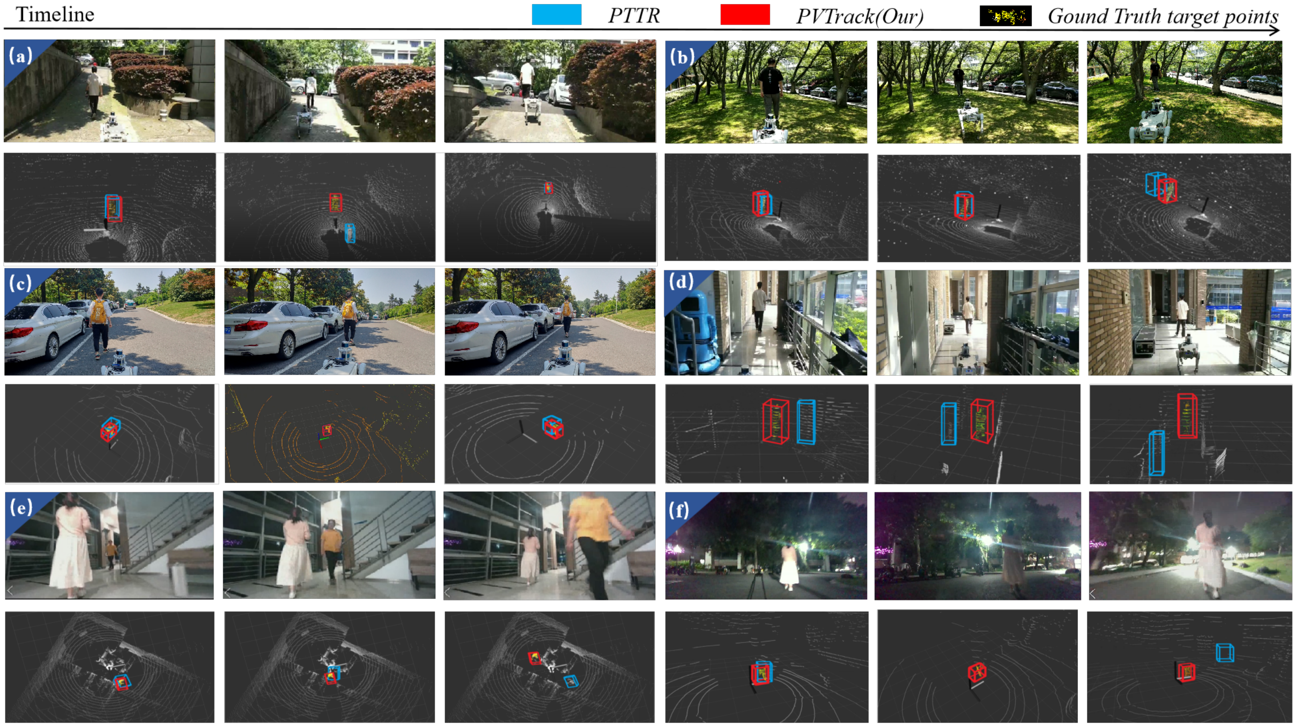

Fig. 5. The original videos and tracking performance on JueYing quadruped robot in various scenes (a-f). The blue box represents the PTTR prediction results, which usually miss and frame the wrong area. While the red box, our PVTrack, robustly tracks the ground truth human highlighted in yellow.

TABLE III
TRACKING RESULTS OF OUR PVTRACK ON QUADRUPED ROBOTS

Point-input	Video-input	SPVT	HPN	PVTrack	
				Success	Precision
-	✓	✓	✓	50.8	59.1
✓	-	✓	✓	71.3	79.6
✓	✓	-	✓	63.2	72.5
✓	✓	✓	-	71.9	80.4
✓	✓	✓	✓	**76.5**	**78.3**

entirely scanned in. To verify the effectiveness of PVTrack on this problem, we invited dozens of representative persons as testing samples that differ vastly in height, body shape, and attire as shown in Fig. 6 (a). It turns out that the proposed network can transfer well to all persons and the targets even don't need to follow the classic human-body architecture (two legs, two arms, one head). For example, if the target person is wearing a skirt, meaning that her lower body is barrel-shaped rather than legs, the transfer can also succeed. Furthermore, when the scanned areas are merely two walking legs, our approach is still valid and applicable.

Effectiveness of the Anti-shake Design. Considering the violent shaking and bumping problem of the robot platform, this article subsequently analyzed the received signals from the robot controller, as shown in Fig. 6 (b). In the left picture without the anti-shake proposal compensation module, the curve has an obvious "sawtooth" due to the frequent missing target prediction and the unstable input frequency. While the right one has little fluctuation but is overall smoother after adding the adjustment module. The relevant response is stable (within the acceptable range of 10:1 to 4:1), which further verifies the importance of our anti-shake design that significantly strengthens the reliability of deploying our PVTrack into dynamic robotic systems.

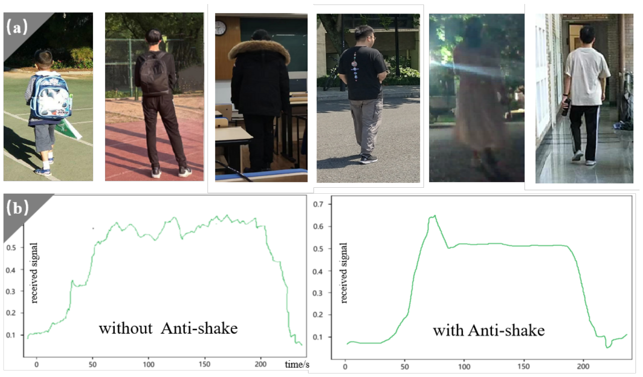

Fig. 6. Some ablation study on PVTrack (a) to different human objects (b) to overcome the specificity of the strong shaking robot platform.

V. CONCLUSIONS

This paper proposes a novel Point-Video-based Transformer tracking framework (PVTrack) for the 3D human tracking task on robots, which merges LiDAR and RGB as dual-inputs and utilizes a multi-level Siamese Point-Vieo Transformer to enrich the template and search region features jointly. Furthermore, a Human-ware Proposal Network with an Anti-shake Proposal Compensation module is designed to select the target-ware 3D bounding box. Our method achieves state-of-the-art results in both open-source datasets and JueYing quadruped robot platform while running in real-time. In the future, PVtarck is expected to bring robots to more intelligent applications and more complex scenarios.

REFERENCES

[1] A. Xiao, W. Tong, L. Yang, J. Zeng, Z. Li, and K. Sreenath. "Robotic guide dog: Leading a human with leash-guided hybrid physical interaction". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 11470–11476.

[2] Z. Zhang, J. Yan, X. Kong, G. Zhai, and Y. Liu. "Efficient motion planning based on kinodynamic model for quadruped robots following persons in confined spaces". In: *IEEE/ASME Transactions on Mechatronics* 26.4 (2021), pp. 1997–2006.

[3] C. Zheng, X. Yan, J. Gao, W. Zhao, W. Zhang, Z. Li, and S. Cui. "Box-aware feature enhancement for single object tracking on point clouds". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13199–13208.

[4] L. Hui, L. Wang, L. Tang, K. Lan, J. Xie, and J. Yang. "3D Siamese transformer network for single object tracking on point clouds". In: *European Conference on Computer Vision*. Springer. 2022, pp. 293–310.

[5] C. Zheng, X. Yan, H. Zhang, B. Wang, S. Cheng, S. Cui, and Z. Li. "Beyond 3d siamese tracking: A motion-centric paradigm for 3d single object tracking in point clouds". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 8111–8120.

[6] T. Ma, M. Wang, J. Xiao, H. Wu, and Y. Liu. "Synchronize Feature Extracting and Matching: A Single Branch Framework for 3D Object Tracking". In: *arXiv preprint arXiv:2308.12549* (2023).

[7] A. Geiger, P. Lenz, and R. Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite". In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE. 2012, pp. 3354–3361.

[8] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. "nuscenes: A multimodal dataset for autonomous driving". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11621–11631.

[9] N. Bellotto and H. Hu. "Multisensor-based human detection and tracking for mobile service robots". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.1 (2008), pp. 167–181.

[10] I. Ullah, Q. Ullah, F. Ullah, and S. Shin. "Integrated collision avoidance and tracking system for mobile robot". In: *2012 International Conference of Robotics and Artificial Intelligence*. IEEE. 2012, pp. 68–74.

[11] M. Munaro and E. Menegatti. "Fast RGB-D people tracking for service robots". In: *Autonomous Robots* 37 (2014), pp. 227–242.

[12] M. Wang, D. Su, L. Shi, Y. Liu, and J. V. Miro. "Real-time 3D human tracking for mobile robots with multisensors". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2017, pp. 5081–5087.

[13] M. Wang, Y. Liu, D. Su, Y. Liao, L. Shi, J. Xu, and J. V. Miro. "Accurate and real-time 3-D tracking for the following robots by fusing vision and ultrasonar information". In: *IEEE/ASME Transactions On Mechatronics* 23.3 (2018), pp. 997–1006.

[14] D. Benz, J. Weseloh, D. Abel, and H. Vallery. "CIOT: Constraint-Enhanced Inertial-Odometric Tracking for Articulated Dump Trucks in GNSS-Denied Mining Environments". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 10587–10593.

[15] S. Giancola, J. Zarzar, and B. Ghanem. "Leveraging shape completion for 3d siamese tracking". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 1359–1368.

[16] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. "Fully-convolutional siamese networks for object tracking". In: *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II 14*. Springer. 2016, pp. 850–865.

[17] A. He, C. Luo, X. Tian, and W. Zeng. "A twofold siamese network for real-time object tracking". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4834–4843.

[18] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. "High performance visual tracking with siamese region proposal network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8971–8980.

[19] H. Zou, J. Cui, X. Kong, C. Zhang, Y. Liu, F. Wen, and W. Li. "F-siamese tracker: A frustum-based double siamese network for 3d single object tracking". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 8133–8139.

[20] Z. Fang, S. Zhou, Y. Cui, and S. Scherer. "3d-siamrpn: An end-to-end learning method for real-time 3d single object tracking using raw point cloud". In: *IEEE Sensors Journal* 21.4 (2020), pp. 4995–5011.

[21] Z. Wang, Q. Xie, Y.-K. Lai, J. Wu, K. Long, and J. Wang. "Mlvsnet: Multi-level voting siamese network for 3d visual tracking". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 3101–3110.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[24] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu. "Transformer tracking". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 8126–8135.

[25] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun. "Point transformer". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 16259–16268.

[26] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu. "Pct: Point cloud transformer". In: *Computational Visual Media* 7 (2021), pp. 187–199.

[27] J. Shan, S. Zhou, Z. Fang, and Y. Cui. "PTT: Point-track-transformer module for 3D single object tracking in point clouds". In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 1310–1316.

[28] C. Zhou, Z. Luo, Y. Luo, T. Liu, L. Pan, Z. Cai, H. Zhao, and S. Lu. "Pttr: Relational 3d point cloud object tracking with transformer". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 8531–8540.

[29] M. Wang, T. Ma, X. Zuo, J. Lv, and Y. Liu. "Correlation Pyramid Network for 3D Single Object Tracking". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 3215–3224.

[30] T. Wilhelm, H.-J. Böhme, and H.-M. Gross. "Sensor fusion for vision and sonar based people tracking on a mobile service robot". In: *Proceedings of the International Workshop on Dynamic Perception*. 2002, pp. 315–320.

[31] M. Kobilarov, G. Sukhatme, J. Hyams, and P. Batavia. "People tracking and following with mobile robot using an omnidirectional camera and a laser". In: *Proceedings 2006*

IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. IEEE. 2006, pp. 557–562.

[32] J. Cui, H. Zha, H. Zhao, and R. Shibasaki. “Multi-modal tracking of people using laser scanners and video camera”. In: *Image and vision Computing* 26.2 (2008), pp. 240–252.

[33] T. Germa, F. Lerasle, N. Ouadah, and V. Cadenat. “Vision and RFID data fusion for tracking people in crowds by a mobile robot”. In: *Computer Vision and Image Understanding* 114.6 (2010), pp. 641–651.

[34] C. Dondrup, N. Bellotto, F. Jovan, M. Hanheide, et al. “Real-time multisensor people tracking for human-robot spatial interaction”. In: (2015).

[35] U. Kart, J.-K. Kamarainen, and J. Matas. “How to make an rgbd tracker?” In: *proceedings of the european conference on computer vision (ECCV) Workshops*. 2018, pp. 0–0.

[36] S. Knoop, S. Vacek, and R. Dillmann. “Sensor fusion for 3D human body tracking with an articulated 3D body model”. In: *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE. 2006, pp. 1686–1691.

[37] M. Luber, L. Spinello, and K. O. Arras. “People tracking in rgb-d data with on-line boosted target models”. In: *2011 ieee/rsj international conference on intelligent robots and systems*. IEEE. 2011, pp. 3844–3849.

[38] A. Pieropan, N. Bergström, M. Ishikawa, and H. Kjellström. “Robust 3D tracking of unknown objects”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2015, pp. 2410–2417.

[39] A. Bibi, T. Zhang, and B. Ghanem. “3d part-based sparse tracker with automatic synchronization and registration”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 1439–1448.

[40] J. Yang, Z. Zhang, Z. Li, H. J. Chang, A. Leonardis, and F. Zheng. “Towards generic 3d tracking in RGBD videos: Benchmark and baseline”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 112–128.

[41] A. Asvadi, P. Girao, P. Peixoto, and U. Nunes. “3D object tracking using RGB and LIDAR data”. In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2016, pp. 1255–1260.

[42] J. Koh, J. Kim, J. H. Yoo, Y. Kim, D. Kum, and J. W. Choi. “Joint 3d object detection and tracking using spatio-temporal representation of camera image and lidar point clouds”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 1. 2022, pp. 1210–1218.

[43] D. Su and J. V. Miro. “An ultrasonic/RF GP-based sensor model robotic solution for indoors/outdoors person tracking”. In: *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. IEEE. 2014, pp. 1662–1667.

[44] Z. Lin, W. Xu, and W. Wang. “A Moving Target Tracking System of Quadrotors with Visual-Inertial Localization”. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 3296–3302.

[45] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”. In: *Advances in neural information processing systems* 30 (2017).

[46] Y. Pang, W. Wang, F. E. Tay, W. Liu, Y. Tian, and L. Yuan. “Masked autoencoders for point cloud self-supervised learning”. In: *European conference on computer vision*. Springer. 2022, pp. 604–621.

[47] Y. Chen, X. Dai, D. Chen, M. Liu, X. Dong, L. Yuan, and Z. Liu. “Mobile-former: Bridging mobilenet and transformer”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 5270–5279.

[48] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, et al. “Scalability in perception for autonomous driving: Waymo open dataset”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 2446–2454.

[49] H. Qi, C. Feng, Z. Cao, F. Zhao, and Y. Xiao. “P2b: Point-to-box network for 3d object tracking in point clouds”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 6329–6338.

[50] H. Zou, C. Zhang, Y. Liu, W. Li, F. Wen, and H. Zhang. “PointSiamRCNN: Target-aware Voxel-based Siamese Tracker for Point Clouds”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 7029–7035.