2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3503-7770-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/IROS58592.2024.10802166

A Robotic-centric Paradigm for 3D Human Tracking Under Complex Environments Using Multi-modal Adaptation

Shuo Xin, Zhen Zhang, Liang Liu, Xiaojun Hou, Deye Zhu, Mengmeng Wang, Yong Liu*

Abstract— **The goal of this paper is to strike a feasible tracking paradigm that can make 3D human trackers applicable on robot platforms and enable more high-level tasks. Till now, two fundamental problems haven't been adequately addressed. One is the computational cost lightweight enough for robotic deployment, and the other is the easily-influenced accuracy varied greatly in complex real environments. In this paper, a robotic-centric tracking paradigm called MATNet is proposed that directly matches the LiDAR point clouds and RGB videos through end-to-end learning. To improve the low accuracy of human tracking against disturbance, a coarse-to-fine Transformer along with target-ware augmentation is proposed by fusing RGB videos and point clouds through a pyramid encoding and decoding strategy. To better meet the real-time requirement of actual robot deployment, we introduce the parameter-efficient adaptation tuning that greatly shortens the model's training time. Furthermore, we also propose a five-step Anti-shake Refinement strategy and have added human prior values to overcome the strong shaking on the robot platform. Extensive experiments confirm that MATNet significantly outperforms the previous state-of-the-art on both open-source datasets and large-scale robotic datasets.**

I. INTRODUCTION

With the explosive growth of perception solutions, single object tracking (abbreviated as SOT), has gradually entered the application scope of intelligent robots and systems. As the basic building block of many advanced tasks, its addition greatly expands the application scenarios and task types, enabling them to have clearer "eyes". For example, it can be widely used in various high-level tasks such as following in unmanned driving, robot collision prediction, and human-machine collaboration, which greatly helps to promote the sustainable development of robot technology. At present, researches on 2D [1]–[4] and 3D SOT [5]–[13] have entered a period of steady development and have achieved fruitful accuracy on well-known open-source datasets [14]–[16]. Whilst, there are still some crucial bottlenecks when migrating them to the physical quadruped robot platform.

Since not all tracking scenes of robots are as perfect as the open-source datasets mentioned above, complex environmental disturbance remains a long-standing problem. In the actual robot application tasks, the existing 3D trackers [17]–[19] suffered greatly from illumination, occlusion, rainy days, and other disturbances, leading to a very big fluctuation in robust tracking. To solve this, we utilize multi-modalities features from RGB videos and point cloud to effectively alleviate the disturbance caused by complex scenes, instead of only using 3D LiDAR sensor. To mitigate the risk of overfitting on one modality, the correspondence between different modalities is deeply exploited for accurate modeling andit achieves a tracking success rate of more than 82%.

Secondly, how to enable trackers to meet the real-time requirements of actual robot deployment becomes an unsolved but key issue. Nowadays, the trend of replacing CNN with Transformer [13], [20]–[22] induces the model parameters to become larger and larger, yielding massive computation costs. Inspired by [23], we hypothesize that the weight change during model training also has a low "intrinsic rank", leading to our adaptation tuning approach. By optimizing rank decomposition matrices of dense layers of a neural network, we train some dense layers indirectly while keeping the pre-trained weights frozen, which not only uses relatively little data for training but also supports online learning when applied to complex environments.

The third challenge is primarily caused by the bumpy robot platform itself. Once the current human trackers are applied to a fast-moving robot, their precision rate will significantly decrease by an average of 10-20%, and most of the frames will lose objects. To relieve the difficult scenarios like violent swaying and jumping of the quadruped robot platform itself, a well-designed robotic-centric anti-shake refinement strategy is proposed and the lost target bounding boxes are supplemented under the guidance of the given robot's motion trend. Therefore, the feedback in the learning process of the algorithm is more anastomotic for the existing algorithm, thus better overcoming platform specificities.

To summarize, the main contributions of this paper are as follows:

- A novel multi-modal tracking model is proposed for 3D human tracking that complements positional and appearance information from 3D point clouds and 2D videos through end-to-end learning, effectively alleviating the disturbance caused by complex environments.
- We introduce a parameter-efficient Adaptation tuning to the task, which reduces considerable computational cost caused by the Siamese Transformer and significantly improves real-time performance.
- We well designed the Human-ware Correlation module and robotic-centric Anti-shake Refinement to promote tracking accuracy for robotic applications.
- Extensive experiments on both KITTI and robotic platforms show our MATNet achieves state-of-the-art performance and impressive tracking in various scenes.

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

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

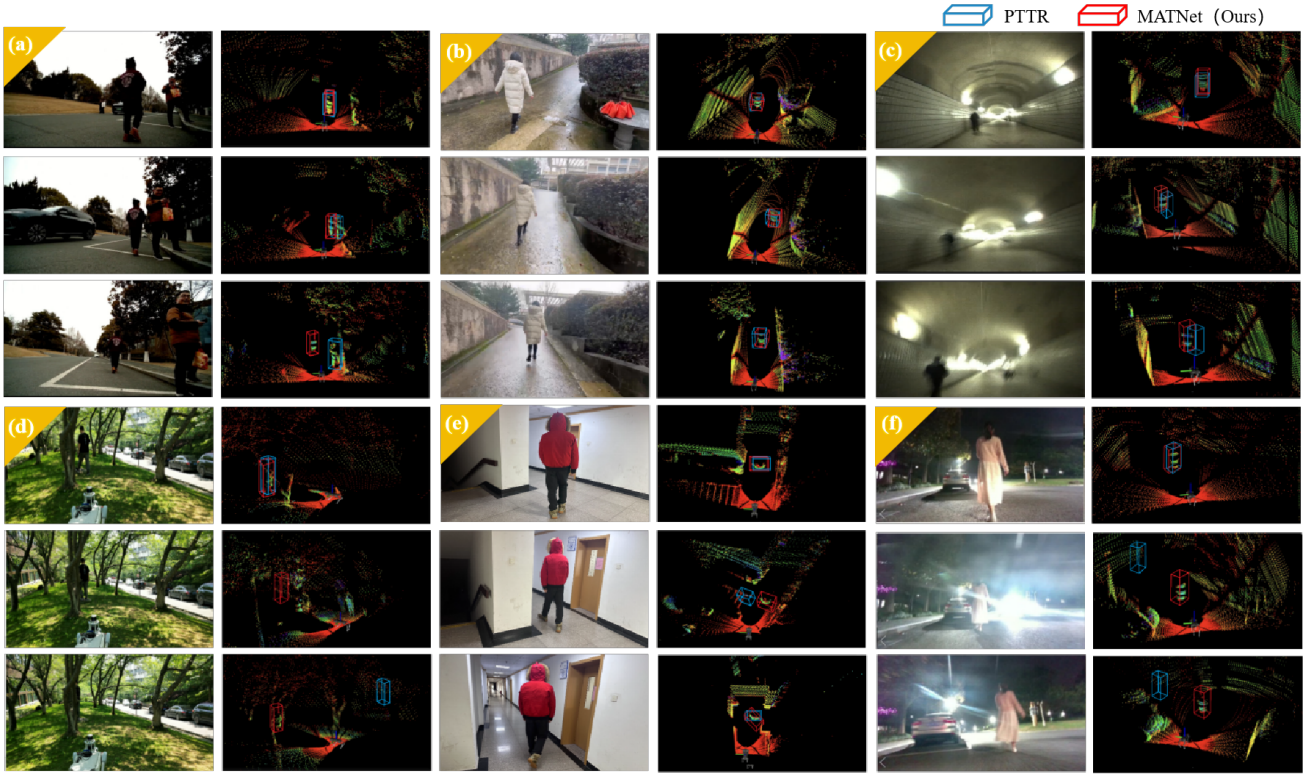

Fig. 1. Visualization results of proposed MATNet(red box) and state-of-the-art PTTR (blue box)in various scenes. The goal of 3D human tracking is to track the right person highlighted in yellow so that the quadruped robot can keep up with the target person.

II. RELATED WORK

A. 3D SOT Tracking

The LiDAR-based 3D SOT is a new task that emerged in recent years. Since the pioneering tracker SC3D [11], the Siamese-like Matching paradigm [24]–[28] become the prevalent backbone, deriving subsequent [10], [29]–[31] that use the CNN-based network with two branches for deeper feature extraction and aggregation. In recent years, the success of Transformer in 3D vision [32]–[34] stimulates numerous attempts to embed them to reshape tracker design. More and more researches [12], [20], [35], [36] weighted the cosine similarity based on the attention after calculating the region features to improve the tracking performance.

Despite the successful development of LiDAR-based SOT, challenges still exist and pose threats to successful tracking. The LiDAR inputs have a deficiency of density variance, points sparsity, insufficient appearance information, and implicit features in data locality. Therefore, the LiDAR-based methods require relatively complex data pre-processing and are sensitive to the sampling quality of raw data. To improve deficiency, some pioneers tried to combine the LiDAR with the RGB format due to the promising complementation of multimodal data, but the relevant work is limited and urgently needs to be further improved. Our previous work [37], [38] systematically explored the approach through the Siamese Point-Video Transformer. Building on the foundation, our new work restructured the feature extraction and fusion architecture greatly and focused more on lightweight the model to be better deployed on the robot platform.

B. Parameter-efficient Adaptation Techniques

There are still some unresolved challenges in training the multi-modal network to fewer model parameters and successfully converge. Using multi-modality as input will inevitably increase the computational cost, as each modality requires its own feature extraction network, leading to longer training periods. To address this issue, parameter-efficient adaptation derived from [23], [39] is a particularly useful approach. [40] inserts adapters into pre-trained multi-modal models for less training time. [41], [42] freezes the pre-trained model and only trains a few additional parameters. However, none of the existing research has been applied practically. To the best of our knowledge, this paper is the first to implement adaptation to robotic tracking tasks. It is particularly beneficial since there are few robotic-centric datasets available for sufficient training and adapter-based fine-tuning from large-scale vehicle-mounted dasasets can remedy the lack while significantly reducing training time.

III. METHODOLOGY

Considering the appearance video provides rich texture and the LIDAR sensor is robust to light variations, making them a suitable complement to each other, we propose a lightweight human-oriented tracking framework MATNet for generating relatively dense correspondences between the RGB videos $V \in R^{H\times W\times 3}$ and LIDAR points $P \in R^{N\times 3}$((point cloud sequence with N points)) as dual inputs.

The overall architecture of the proposed MATNet is shown in Fig.2, which mainly consists of three parts: (1) Multi-modal Feature Extraction Module to model the tracked instance with multi-modal adaptation, (2) Coarse-Level Transformer for 2D-3D matching, and (3) Fine-level Matching Module for goal-conditioned feature aggregation and final 3D bounding box generation.

A. Multi-modal Feature Extraction

The conventional Siamese tracking pipeline suffers from great computational complexity with the separate template branch Z and search branch X along with two sets of local descriptors (such as PointNet [43] and PointNet++ [44]) to extract the feature twice. However, once the application scenario migrates to a multi-modal input scenario, the number of branches and the number of corresponding feature extraction networks doubles with each new modality added, which requires a very large amount of calculation. Therefore, in this part, we pioneered the process of joining the template branch and the search branch together to form a single-stream network, thus greatly reducing the network complexity.

In this formulation, the Feature Pyramid Network (FPN) is used to extract the video feature while the KPConv-FPN is used for the point clouds to extract multi-level features. For each frame of video V, template frames z^V and search frames x^V are projected into patch embeddings, namely a coarse feature $F_c^V = [\hat{z}_c^V, \hat{x}_c^V] \in R^{(N_z+N_x)\times D}$ with 1/16 spatial resolution of the input image and a fine feature $F_f^V = [\hat{z}_f^V, \hat{x}_f^V] \in R^{(N_z+N_x)\times D}$ with 1/4 ones. For point cloud feature Z^N and X^N input, we perform a multi-layer down-sampling operator to obtain five point groups $P^k \in R^{N_k\times 3}$ with different resolutions $k = 1,2,3,4,5$. The point cloud features can be extracted by five encoder layers and one decoder layer. In Fig 2, $F_c^P = [\hat{z}_c^P, \hat{x}_c^P]$ denotes the coarse matching feature and $F_f^P = [\hat{z}_f^P, \hat{x}_f^P]$ represents the fine-matching feature. Both of them are encoded into an intermediate representation to help the model better distinguish the target from its surroundings.

To further lightweight our backbone and better migrate to the robotic platform, we utilize Adapter-based tuning to fine-tune the pre-trained Transformer tracker in a symmetric manner. As shown in Fig.2(c), the adapter is a bottleneck architecture, which consists of two fully connected (FC) layers, a GELU activation layer, and a residual connection. The first FC layer (FC Down) projects the input to a lower dimension, and the second FC layer (FC Up) projects it back to the original dimension. We then insert adapters into the FPN block after each layer with MLP in parallel. The computation in the i-th fusion stage can be formulated as:

$$\tilde{\mathrm{H}}^{(l)} = \mathrm{H}^{(l-1)} + \mathrm{Adapter}(\mathrm{CNN}(\mathrm{H}^{(l-1)})) \quad (1)$$

$$\mathrm{H}^{(l)} = \tilde{\mathrm{H}}^{(l)} + \mathrm{MLP}(\mathrm{LN}(\tilde{\mathrm{H}}^{(l)})) + r\cdot\mathrm{Adapter}(\mathrm{LN}(\tilde{\mathrm{H}}^{(l)})) \quad (2)$$

where $\mathrm{H}^{(l-1)}$ and $\mathrm{H}^{(l)}$ are the output feature $[F_i^V, F_i^P]_{i=c,f}$ of the (l-1)-th and l-th block of FPN, and r is a scaling factor that regulates the influence of the adapter's output weight. In this way, we can efficiently reuse part of the pre-trained trackers that have achieved excellent success on open datasets like KITTI and only train lightweight adapters while keeping the pre-trained weights frozen. When the actual platform is turned to the bumpy robots, this computation leads to a noteworthy reduction in the required size of the input training datasets, thereby substantially enhancing the matching efficiency while simultaneously preserving accuracy.

B. Coarse-level Transformer

This module is designed to accurately perform one-to-one assignments through a seq2seq Transformer-based framework and determine the pixel coordinates of the candidate bounding boxes. As Fig.3. shown, after obtaining the position embedding $[P^4, P^5]$ with the coarse feature$[F_c^V, F_c^P]$, we processed through multiple layers of transformer models, capturing contextual dependency relationships among tokens. Similar to PVTrack, it is a hierarchical feature learning network, consisting of each-modality *self-attention* and multi-modality *cross attention*. The attention value is obtained by calculating the similarity between the query Q and the key K and then weighting and summing the values V according to the product of this set of weights $[W_q, W_k, W_v]$ and the corresponding value, which are indicated as follows:

$$A_c = \mathrm{Attention}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \phi(\mathrm{Q} - \mathrm{softmax}(\mathrm{Z})\cdot(W_v)\mathrm{V}) \quad (3)$$

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

Matching between 2D videos and 3D point clouds. The substantial differences in data types between 2D and 3D make it a heterogeneous data-matching problem. Inspired by [45], [46], we explicitly encode the geometry and estimate the relative geometric relationship between 2D and 3D using global and explicit cues. This work is like information retrieval. Specifically, after obtaining the attention-augmented features $[A_c^V, A_c^P]$, we first normalize the feature output and calculate the transition matrix by $S \in R^{(1/k)^2 HW\times N} = < normalize(A_c^V), normalize(A_c^P) >$. Since not all 3D points can be found on the 2D image, we extend the matrix S to $\hat{S}$ by adding a new row as a bin filled with a variable. Then, as depicted in Fig.2(b), we calculate the coarse-level confidence matrix M_c by Softmax with temperature t and select the coarse candidate regions (i_c^x, i_c^y, i_c^z) that beyond the preset threshold th_c:

$$M_c = \{(i_c^x, i_c^y, i_c^z) | \hat{S} = max(Softmax(\hat{S}(\cdot, i/t))), \hat{S} > th_c\} \quad (5)$$

C. Fine-Level Matching Module

To gain more precise positioning, we implement part-aware and size-aware processing to improve the precision of the matching results, which can be depicted by point-to-box relation. For the 2D image, we crop the window of size $w\times w$ around the highest M_c and concat candidates region with F_f^V to update track queries. For the 3D point cloud, we execute the k-nearest neighbors (k-NN) for geometric aggregation and sample a local point cloud P^{bb_c} from P^3 based on the Euclidean distance. Furthermore, a *Human-ware Correlation Head* proposed in our previous work PVTrack is used for filtering the coarse candidates by immediately discarding

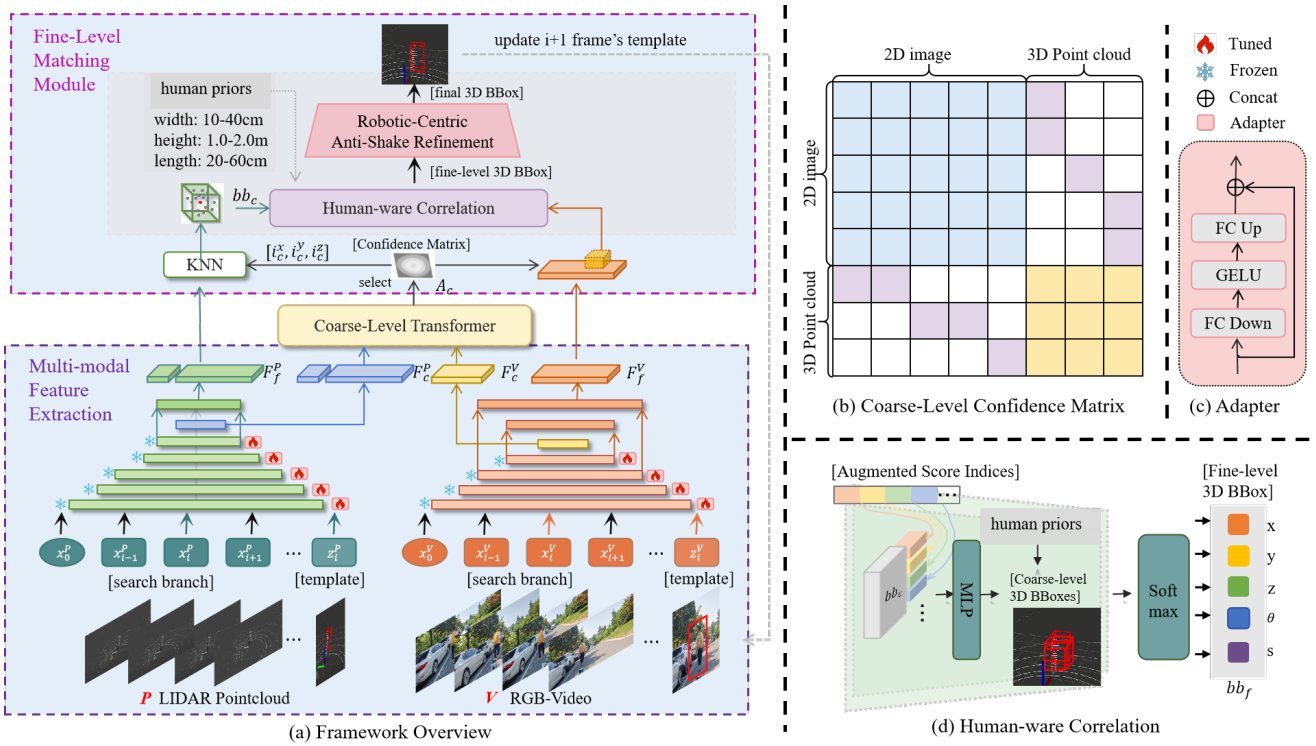

Fig. 2. (a) The overview of MATNet framework. Given N^v video frames and N^p point sequences as input, MATNet models the 3D SOT as a bottom-up learning problem with multimodal summaries. (b) Coarse-level Confidence Matrix is applied to align and fuse each video and point pair by similarity-based matching. (c) The structure of the Adapter. (d) Human-ware correlation decodes the augmented score indices to output the 3D tracking results.

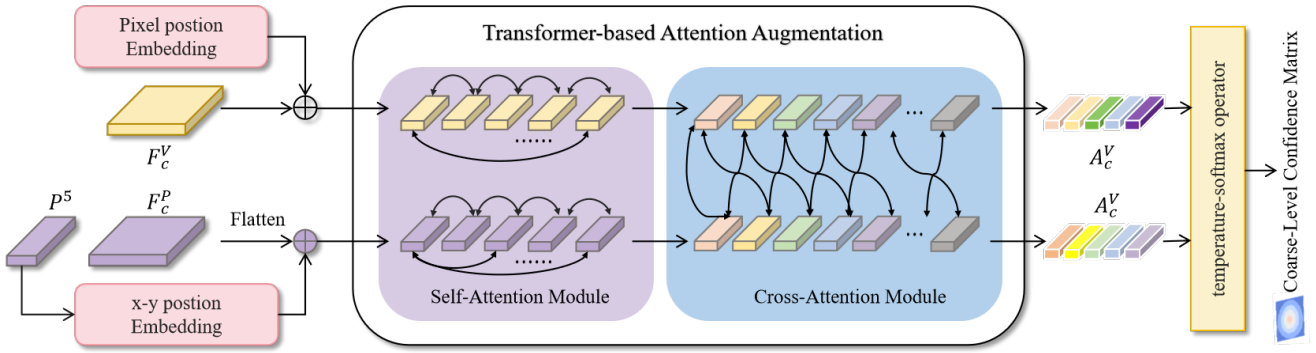

Fig. 3. The Coarse-level Transformer module and its internal principles. Best viewed in color.

the oversized ones, as illustrated in Fig.2(d). In this way, the learned features are geometrically discriminative and can effectively resolve the problem of matching ambiguity, thereby reducing the number of outlier matches. We name the filtered box as fine-level 3D BBox.

Robotic-centric Anti-Shake Refinement. Another breakthrough of MATNet is the proposed five-step refinement strategy that is robust to violent platform shaking or rugged terrain. First, our dynamic model computes the goal velocity and its next move trends over the last 10 frames using the robot's inbuilt speed-measuring module. The process can be written as $S = max\{5, \sum_{i=t-9}^{t-1} \|X_i - X_{i-1}\|_2\}$, where X_i is the i-th 3D location of the target. Secondly, once the tracking human disappears due to the movement of robots, the response score r^i of foreground BBox will fall below threshold Ω, and the anti-shake algorithm will be triggered. On the one hand, the candidate area is adaptively expanded according to its recorded move trends, namely:$W^n = n*S+2*bb(1), H^n = n*S+2*bb(2), L^n = n*S+2*bb(3)$, where bb is the three-dimensional (width, height, and length) size of the individual. Thirdly, we extract points in $I_t^{W^n \times L^n \times H^n} \subset I_t$ around the center$\mathbf{X}_{i-1}$ and use new r^n to represent the maximum response r^i. Subsequently, note that the tracker response of a recovered object must be similar, our refinement will automatically turn to the next frame if max response r^n still falls below threshold Ω. Otherwise, it will interrupt the loop so that our system can produce 3D BBoxes of the absent frames and start tracking again. Finally, with M proposals generated above, we utilize a 3-layer MLP to select the highest-score one as the final prediction. The final tracking output will replace the template input in the next iteration.

IV. EXPERIMENT

In this section, we first compare our proposed MATNet with other state-of-the-art methods on open-source tracking datasets KITTI. We then validate the efficiency of MATNet on the robot platform Unitree Aliengo, as illustrated in Fig.4. The robot is equipped with an onboard computing device, NVIDIA Jetson Xavier NX (21 TOPS, 16GB), one compact and lightweight LiDAR (Livox Mid-360, 40-line), and a forward-facing camera (RMONCAM G200, 1080P), publishing point cloud measurements at 10Hz and video frames at 30Hz, respectively.

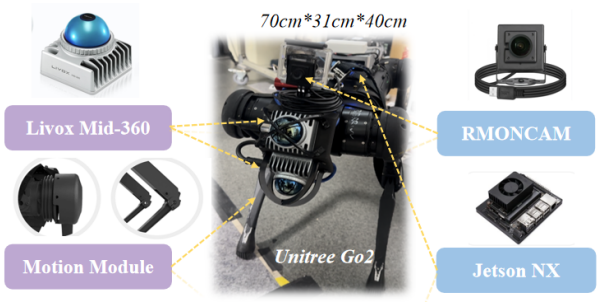

Fig. 4. The Unitree Aliengo robot platform and its onboard sensors.

Implementation Details. All data processing and model training validation were completed based on an NVIDIA 3090Ti GPU with a global batch size of 64. During the training phase, we utilize the Adam optimizer with the initial learning rate set to 1×10^{-3} and reduce it by 5 every 30 epochs.

Evaluation protocol. The loss function for T frames in total can be defined as:

$$L = \frac{1}{T}(\lambda^{cls} \sum_{i=1}^{T} L_i^{cls} + \lambda^{reg} \sum L_i^{reg}(bb)) \tag{6}$$

where L^{cls} is a standard cross-entropy loss to distinguish the foreground human-ware BBoxes from the background. L^{reg} is defined as the Huber loss between the prediction and ground-truths. $\lambda^{cls} = 0.4$ and $\lambda^{reg} = 0.6$ are the corresponding weight coefficients. Besides, we evaluate the models using the One Pass Evaluation (OPE) and report *Success* and *Precision* as the evaluation metrics of each model. Specifically, *Success* defines the overlap as the intersection over union the Area (IoU) of a bounding box with its ground truth while *Precision* measures the Area Under Curve (AUC) with the error threshold varying from 0 to 2 meters.

A. Comparison with State-of-the-arts on KITTI

KITTI MOT, as one of the most prevalent tracking datasets, provides 29GB high-quality laser point clouds, 12GB video sequences, and 5MB labels for target tracking. Due to the lack of an official partition of train/val/test, we followed the approach of [20] by using sequences 01, 06, 08, 10, 12-19 from KITTI MOT as the validation set and the remaining sequences as the training set. To equivalently test human tracking performance, we specifically focus on two categories: cyclists and pedestrians. Table. I compares the proposed MATNet with four representative 3D human tracking methods including [10], [12], [22], [31]. It can be seen that our MATNet performance exceeds with an average improvement of 1.93%/ 9.41% (precision/success) in the pedestrian category and 3.33%/15.0% in the cyclist category. Our human-ware MATNet achieves the highest score, in terms of both *Success* (89.13%) and *Precision* (82.11%). In addition, MATNet takes nearly three times less than the latest Transformer-based tracker PTTR, enabling real-time tracking (26.1 FPS) while achieving SOTA accuracy.

Robustness performance. We also conducted multiple comparative and ablation experiments to verify the feasibility of the algorithm against various complex disturbances with an average success score of almost 73%. As shown in Fig.5, MATNet performs exceptionally well under attributes such as full occlusion, rotation, and multiplayer alternation.

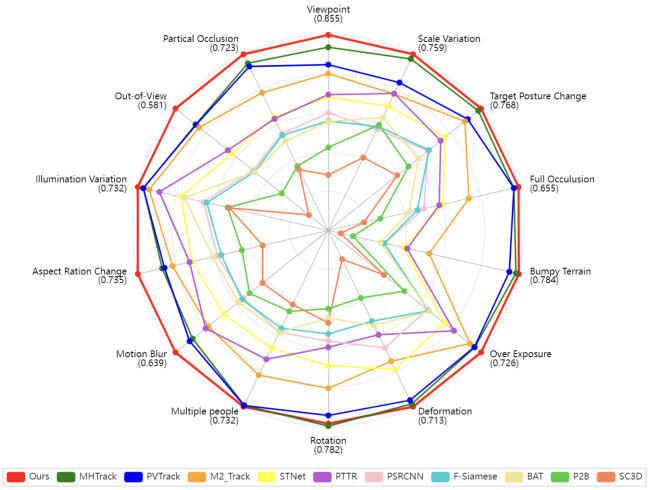

Fig. 5. Success scores of different attributes on the KITTI test set.

B. Application experiment on the quadruped robot

In this part, we validate our method's efficiency in 30+ challenging on-robot scenarios, including narrow corridors, rough terrain, forests, dark tunnels, static and dynamic obstacles, etc. The physically collected datasets achieve a high diversity of over 15000 frames of point clouds and 38 sequences of RGB videos on the foundation of the PVT-3D dataset [37]. To better test real scenarios, the target person engages in different modes of movement (such as walking, staying, and running) and the robot used for tracking adopts various gaits initially. As Fig.6. shown, successful tracking needs to follow the same person for at least 45 seconds.

As shown in Table. I, we compare MATNet with top-performance approaches on the robot platform. As for easy scenarios illustrated in Fig.1(e), the prediction form MATNet can more accurately predict the person's orientation and position, mostly by large margins of 5%. Furthermore, we conduct various specially designed difficult scenes in Fig.1, such as a person brushing against another person (a), rainy day (b), entering a dark tunnel (c), going uphill and downhill

TABLE I

COMPARISON OF MATNET AGAINST STATE-OF-THE-ARTS ON THE KITTI OPEN DATASETS AND ROBOTIC COLLECTION DATASETS.

Datasets		**KITTI open-source datasets**			**Collected human-tracking datasets from quadruped robot**				
Category		Pedestrian	Cyclist	**Mean***	Flat Outdoor	Flat Indoor	Strong Shaking	Complex Environment	**Mean***
Frame Number		*6088*	*308*	*/*	*1528*	*1382*	*3567*	*1211*	*/*
Success	P2B	28.72	35.11	31.92	27.54	25.83	9.36	11.79	18.63
	F-Siamese	49.85	72.22	61.03	48.72	45.99	29.63	31.45	38.94
	PTTR	50.93	67.94	59.43	48.28	47.61	32.71	36.68	41.32
	M^2-Track	67.53	70.30	68.61	67.49	66.47	40.26	42.89	54.27
	MATNet(Ours)	**76.94**	**87.29**	**82.11**	**75.55**	**74.98**	**73.28**	**74.77**	**74.65**
	Improvement	↑9.41	↑15.0	↑13.5	↑8.06	↑8.51	↑33.0	↑31.8	↑20.38
Precision	P2B	49.61	52.17	50.89	48.53	47.44	28.89	33.84	39.68
	F-Siamese	70.36	75.18	72.77	70.84	68.02	47.28	51.63	59.44
	PTTR	81.66	82.19	81.92	79.26	78.07	50.02	55.20	65.64
	M^2-Track	86.27	86.73	86.50	84.29	83.26	67.41	70.36	76.33
	MATNet(Ours)	**88.20**	**90.06**	**89.13**	**87.30**	**87.06**	**76.52**	**78.90**	**82.44**
	Improvement	↑1.93	↑3.33	↑2.63	↑3.01	↑3.80	↑9.11	↑7.54	↑6.11

*__Mean__ denotes the average results of counter-categories. **Bold** and underline denote the best and the second-best performance.

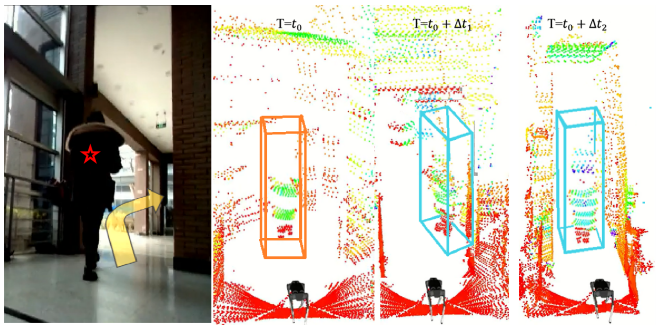

Fig. 6. Given a target bounding box in the first frame, MATNet tracker on quadruped robot recognizes and locates the person in all subsequent frames.

(d), and lighting flickers and blurs s(f). Remarkably, in this sort of scenario, the majority of existing algorithms would greatly degenerate below 50% in precision, yielding intermittent or false tracking. Nevertheless, MATNet can retain approximately 82.44% success score while guaranteeing real-time mobility. No matter is rainy or sunny, indoors or outdoors, summer or winter, day or night, the experimental results show that our improvement for pedestrians is significant (6.1%/20.3% in terms of success/precision). This greatly verifies the great tolerance to complex environments.

Comparison of computational cost. Under the same configurations, the traditional training time (TT) for MATNet is 15 hours with 100+ epochs, and the TT after acceleration is 4 hours with only 45 epochs, which greatly facilitates successful convergence during training. Besides, the parameter required is smaller. The model parameter before the lightweight operation is M-Param, and after adding the Adapter, the required-to-tuned parameter (A-Param) is reduced by 2 to 3 times. Instead of training from scratch, the trackers can be fine-tuned upon the models that are pre-trained on large-scale LiDAR-Video datasets, which makes our tracking algorithm more useful in practical applications.

V. CONCLUSIONS

This paper proposes a novel human-oriented 3D SOT paradigm that can be applied to mobile robots. On the one hand, it can achieve high-precision tracking and alleviate the disturbance in complex environments through multi-modal fusion and well-designed coarse-level Transformer. On the other hand, it can effectively overcome the particularity of the robot platform with the goal-conditioned anti-shake refinement strategy and correctly frame the target in 3D BBox through human-ware correlation. Furthermore, the overall model is simplified a lot by our parameter-efficient adaptation, greatly lightweighting the model's training process and tuned parameters. Extensive analysis verifies each component's effectiveness and the promising robustness to strong-shaking platforms and challenging scenes. In the future, we believe that the robotic-centric paradigm can serve as a primary principle to guide more architectural designs and bring intelligence systems to more high-level applications and more complex scenarios.

TABLE II

ABLATION STUDIES ON THE EFFECT OF THE COMPONENTS ON ROBOT.

MMF	**CLT**	**ASR**	MATNet			
			Suc/Pre(%)	M-Param	A-Param	TT
-	✓	✓	77.3/ 69.5	6.4M	3.1M	∼2h
✓	-	✓	73.2/ 65.9	6.6M	3.3M	∼4h
✓	✓	-	62.8/ 55.1	6.5M	3.9M	∼4h
✓	✓	✓	82.4/ 74.7	6.6M	3.9M	∼4h

REFERENCES

[1] 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*, pp. 8971–8980, 2018.

[2] F. Xie, C. Wang, G. Wang, Y. Cao, W. Yang, and W. Zeng, "Correlation-aware deep tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8751–8760, 2022.

[3] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, "Learning spatio-temporal transformer for visual tracking," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10448–10457, 2021.

[4] 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)*, pp. 10587–10593, IEEE, 2023.

[5] M. Munaro and E. Menegatti, "Fast rgb-d people tracking for service robots," *Autonomous Robots*, vol. 37, pp. 227–242, 2014.

[6] 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*, pp. 112–128, Springer, 2022.

[7] H.-N. Hu, Y.-H. Yang, T. Fischer, T. Darrell, F. Yu, and M. Sun, "Monocular quasi-dense 3d object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 1992–2008, 2022.

[8] 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, pp. 1210–1218, 2022.

[9] 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," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 4995–5011, 2020.

[10] 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*, pp. 6329–6338, 2020.

[11] 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*, pp. 1359–1368, 2019.

[12] 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)*, pp. 8133–8139, IEEE, 2020.

[13] Z. Luo, C. Zhou, L. Pan, G. Zhang, T. Liu, Y. Luo, H. Zhao, Z. Liu, and S. Lu, "Exploring point-bev fusion for 3d point cloud object tracking with transformer," *arXiv preprint arXiv:2208.05216*, 2022.

[14] 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*, pp. 3354–3361, IEEE, 2012.

[15] 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*, pp. 11621–11631, 2020.

[16] 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*, pp. 2446–2454, 2020.

[17] 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," *IEEE/ASME Transactions On Mechatronics*, vol. 23, no. 3, pp. 997–1006, 2018.

[18] 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," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 4, pp. 1997–2006, 2021.

[19] 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)*, pp. 3296–3302, IEEE, 2023.

[20] 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*, pp. 3215–3224, 2023.

[21] 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)*, pp. 1310–1316, IEEE, 2021.

[22] 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*, pp. 8531–8540, 2022.

[23] H. J., Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models.," *arXiv: Computation and Language,arXiv: Computation and Language*, Jun 2021.

[24] 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*, pp. 850–865, Springer, 2016.

[25] 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*, pp. 4834–4843, 2018.

[26] 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*, pp. 8971–8980, 2018.

[27] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4282–4291, 2019.

[28] Y. Xu, Z. Wang, Z. Li, Y. Yuan, and G. Yu, "Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, pp. 12549–12556, 2020.

[29] L. Hui, L. Wang, M. Cheng, J. Xie, and J. Yang, "3d siamese voxel-to-bev tracker for sparse point clouds," *Advances in Neural Information Processing Systems*, vol. 34, pp. 28714–28727, 2021.

[30] 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*, pp. 3101–3110, 2021.

[31] 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*, pp. 8111–8120, 2022.

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

[33] 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*, pp. 16259–16268, 2021.

[34] 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*, pp. 16259–16268, 2021.

[35] Y. Cui, Z. Fang, J. Shan, Z. Gu, and S. Zhou, "3d object tracking with transformer," *arXiv preprint arXiv:2110.14921*, 2021.

[36] 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*, pp. 293–310, Springer, 2022.

[37] S. Xin, Z. Zhang, M. Wang, X. Hou, Y. Guo, X. Kang, L. Liu, and Y. Liu, "Multi-modal 3d human tracking for robots in complex environment with siamese point-video transformer," in *ICAISISAS*, 2024.

[38] S. Xin, L. Liu, X. Kang, Z. Zhang, M. Wang, and Y. Liu, "Beyond traditional driving scenes: A robotic-centric paradigm for 2d+3d human tracking using siamese transformer network," in *ICRA*, 2024.

[39] M. Li, Y. Wang, H. Zhang, and J. Wang, "Prft: A fuzz testing method for tire pressure monitoring system based on protocol reverse," in *2023 2nd International Conference on Big Data, Information and Computer Network (BDICN)*, Jan 2023.

[40] B. Chen, R. Wang, D. Ming, and X. Feng, "Vit-p: Rethinking data-efficient vision transformers from locality,"

[41] X. Hou, L. Liu, Y. Qian, Y. Guo, S. Xin, J. Cheng, K. Tang, and Y. Liu, "Sdstrack: Self-distillation symmetric adapter learning for multi-modal visual object tracking," in *CVPR*, 2024.

[42] J. Xing, M. Wang, X. Hou, G. Dai, J. Wang, and Y. Liu, "Multimodal adaptation of clip for few-shot action recognition," 2023.

[43] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

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

[45] L. Li, Y. Ma, K. Tang, X. Zhao, C. Chen, J. Huang, J. Mei, and Y. Liu, "Geo-localization with transformer-based 2d-3d match network,"

[46] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2021.