Towards Visual Query Localization in the 3D World

Supplementary Material

Figure 1. Qualitative results of several baselines and our proposed LaF. We can see that, the proposed LaF locates target object in different scenarios, showing its robustness for 3DVQL.

The following materials provide implementation details, dataset statistics, and additional qualitative visualizations.

S1 Mobile Robotic Platform
In this section, we demonstrate more details of our mobile robotic platform used for multimodal data collection.

S2 Annotation Tool
We display more details of the annotation tool in labeling sequences with 9DoF 3D bounding boxes and its reliability analysis for high-quality annotation.

S3 More Statistics
We show the distribution of 3DVQL across spatial locations, as well as more detailed statistical information on the response tracklets.

S4 Evaluation Metrics and 3D IoU
We describe the formulation of different 3DVQL tasks.

S5 Details of Baselines
We present the details of baselines.

S6 Qualitative Results
We offer more qualitative analysis of our LaF and its comparison to other trackers on 3DVQL.

S7 Maintenance and Responsible Usage of 3DVQL for Research
We discuss the maintenance and responsible usage of our proposed 3DVQL for research.

S1 Mobile Robotic Platform

To build the multimodal data resources of 3DVQL, we integrated and constructed a mobile robotic platform on the Clearpath Husky A200 chassis. The platform is equipped with a 64-beam LiDAR, an RGB camera, and a depth camera, and with the tool of [1] we completed multi-sensor time synchronization and extrinsic calibration to ensure cross-modal geometric consistency. The physical platform is shown in Fig. 2, which displays the mobile robotic platform used for multimodal data acquisition during the development of 3DVQL, and the specific configurations of the sensors and the robot chassis are given in Tab. 1.

Table 1. Specific configuration of our mobile robotic platform.

Device Name	Specification
LiDAR Sensor	Ouster OS-64 (64-beam)
Depth Camera	OAK D-Pro
RGB Camera	FLIR BFS-U3-32S4C-C
Robot Chassis	Clearpath Husky A200

S1.1 3DVQL Task Definition and Problem Setting

In 3DVQL, the input consists of a visual query and a multimodal search sequence, and the goal is to predict both the temporal interval and the frame-wise 9DoF 3D bounding boxes of the queried target. Depending on the sensor configuration, the task can be instantiated as point-cloud-only, RGB-D, RGB-PC, or other multimodal variants. All 9DoF 3D bounding boxes are defined in the camera coordinate system. The RGB camera, depth camera, and LiDAR are geometrically aligned through time synchronization and extrinsic calibration, so boxes and multimodal features can be interpreted in a shared calibrated space.

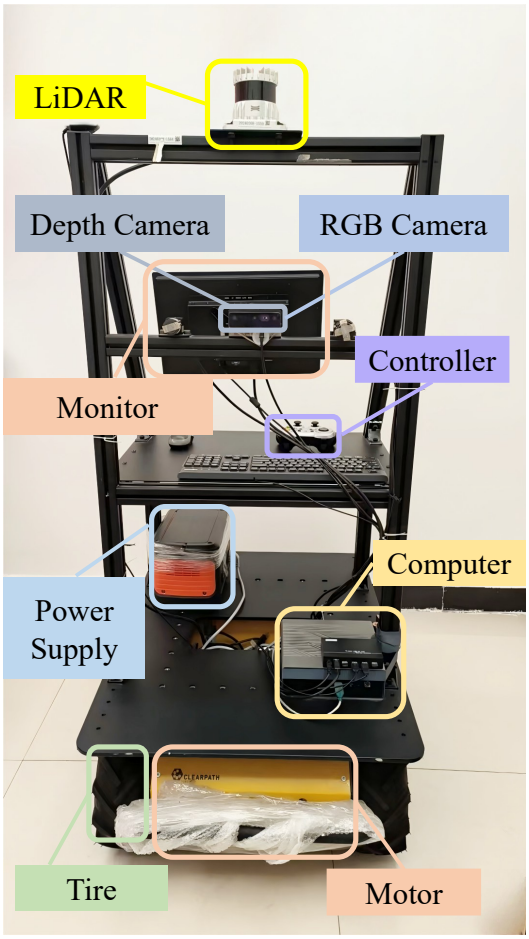

Figure 2. Our mobile robotic platform for data collection.

3DVQL is not equivalent to standard 3D single-object tracking. In 3D SOT[3], the tracker is typically initialized with the target state in the first frame and then follows a continuously visible target. In contrast, 3DVQL starts from a query template rather than first-frame ground truth, and the target may disappear and reappear across multiple response segments. The model must therefore retrieve and re-localize the queried target under viewpoint changes, distractors, and intermittent visibility, rather than merely propagate a known track.

Compared with Ego4D VQ3D [2], which mainly focuses on egocentric human videos, 3DVQL targets embodied robot scenarios and provides calibrated RGB, depth, and

point cloud data for full 3D localization. Moreover, 3DVQL evaluates stricter 9DoF spatial alignment together with temporal localization, making it better suited for multimodal embodied perception in physical 3D environments.

S1.2 Query and Search Sequence Acquisition

Our query template frame is not sampled from the same video as the search sequence. For each category, we separately collect a template video and a search sequence, where the template sequence is about 80 frames long, and the two are kept independent in data source to reduce the matching bias caused by adjacent temporal information. The template frame used to construct the query is selected from the template video, and the target is required to have complete and clear appearance information. During collection, in order to further ensure effective differences between the template frame and the search sequence, the template video is usually not collected in the same sub-scene as the search sequence, and differences in appearance, motion state, and sub-scene are maintained as much as possible. For example, the target in the search sequence may appear in a forward-walking state, while the template frame may capture its back view. Compared with the target instance in the search sequence, the template frame usually has more obvious differences in viewpoint, position, and surrounding context. This design further increases the difficulty of the task and makes it essentially different from a normal 3DSOT setting, so it can more effectively evaluate the model's query-driven target localization ability and cross-view matching ability in complex scenes.

S2 Annotation Tool

The 3D annotation in this study was completed using a professional annotation platform provided by a certain company. The interface for its 3D bounding box annotation is shown in Fig. 3. In practice, we annotated each frame of the point cloud on a continuous trajectory segment basis: the annotator first sketched the approximate 3D bounding box of the target in a scalable annotation view, completing the initial bounding; then, switching to the XY, XZ, and YZ projection views, the corresponding 2D bounding boxes were fine-tuned to ensure the 3D bounding boxes closely matched the target shape in the three orthogonal directions. Simultaneously, the tool projected the obtained 3D bounding boxes onto an RGB image, providing an intuitive visual preview so that the annotator could re-verify and fine-tune the results from an image perspective.

S2.1 Annotation Cost and Workflow

The annotation process involved 20 members in total, including 16 for initial labeling and subsequent revision, and 4 for quality checking and result review. Before formal annotation started, we first screened the raw collected videos and removed video sequences that did not meet the requirements, such as cases where the point cloud was present but the target was not within the frustum range, the target was unclear and no visible target could be observed, the sensor data was abnormal, or the content did not satisfy the annotation requirements, so as to ensure the usability of the data for subsequent annotation. On this basis, all annotators received unified training to ensure a consistent understanding of annotation standards, bounding box definition, and operation procedures. For each clip, we first annotated its first frame as an important reference for subsequent continuous-frame annotation, so as to improve annotation efficiency and maintain temporal consistency. The whole annotation process lasted about 8 months, and multiple rounds of checking and revision were used to further ensure the accuracy and reliability of the annotation results.

S3 More Statistics

In this section, we further analyze the spatial statistics of 3DVQL. Specifically, Fig. 4 shows the distribution of the 3D center coordinates of all targets on the X, Y, and Z axes. It can be seen that the targets are mainly concentrated in the region $X! \in [0, 10]\,\text{m}$, $Y! \in [-2, 2]\,\text{m}$, and $Z! \in [-1, 1]\,\text{m}$. This is also the spatial range setting we adopted in our implementation: $\texttt{space\_range} = [[0.0, -2.0, -1.0], [10.0, 2.0, 1.0]]$. Simultaneously, we evenly divided this range into $16 \times 16 \times 16$ grid cells in the three directions ($\texttt{X\_center\_num} = \texttt{Y\_center\_num} = \texttt{Z\_center\_num} = 16$) for subsequent 3D retrieval and localization. Fig. 5 shows the size distribution of the target's 3D bounding box along the three dimensions of length (L), width (W), and height (H). It can be seen that most targets are concentrated in a relatively small scale, but still cover a variety of sizes from everyday small objects to medium-sized facilities. Fig. 6 statistically analyzes the distribution of target poses in the three angles of roll, pitch, and yaw. Roll and pitch are approximately symmetrical around 0°, while yaw exhibits several dominant orientations.

As shown in Fig. 7, we count the distributions of the indices of the starting frames and ending frames of the response tracklets. At the same time, we can clearly observe that the indices of the starting frames are mainly concentrated from frame 0 to frame 200, which indicates that most queries can obtain a response in the early stage of the video, although there are still some cases after frame 200, which means that higher temporal robustness is required for the query algorithms. The indices of the ending frames mostly fall between frame 40 and frame 400, and there is also a certain long tail, indicating that the lengths of the response tracklets in the dataset exhibit a clear long tail distribution.

These distributions also explain why we used a 9 DoF threshold of 0.05. We also hope that this statistical information will help readers gain a more comprehensive un-

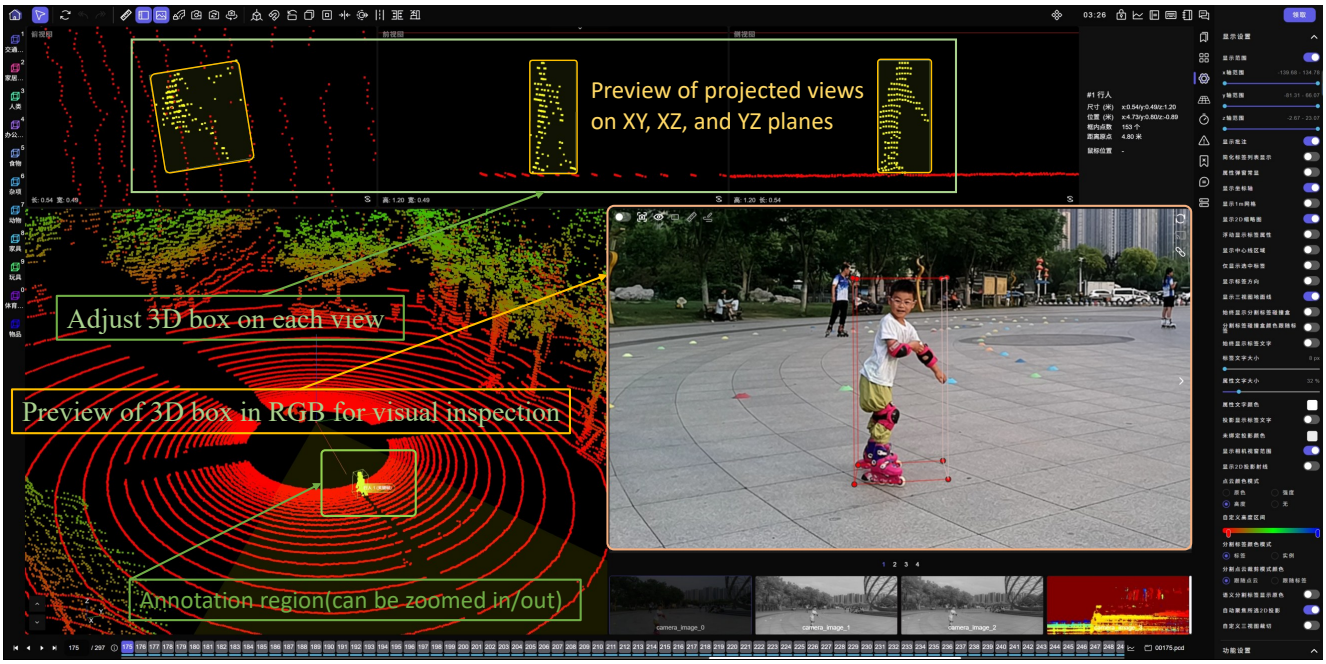

Figure 3. Annotation interface of our used annotation tool.

derstanding of the characteristics of the 3DVQL dataset in terms of spatial range, object scale, and pose diversity.

S3.1 Video and Response Statistics

We define the video length as the number of frames in which the target appears. Under 20 FPS, this length ranges from 40 to 390 frames, with an average of 86 frames. The statistics here reflect the duration of target visibility, rather than the full clip length including negative sample segments. In addition, each sequence contains 1 to 5 response segments, with an average of about 3 response segments per sequence. This shows that the target is usually visible intermittently rather than continuously within one interval, and this also makes cross-segment localization a key property of 3DVQL. The nature of 3DVQL is not continuous tracking in the traditional sense, but query-driven target localization in complex video scenes. Given a query, the model needs to complete target identification and localization across multiple discrete response segments, and handle the matching challenges caused by target disappearance, reappearance, and viewpoint changes.

S3.2 Short Sequences and Episodic Memory

When the robot switches areas or the target leaves the field of view, it usually means the clip ends. This keeps the sequence length short and also provides enough viewpoint changes for learning negative samples. This segmentation also naturally covers cases where the target is temporarily absent, re-enters the field of view, or recovers from occlusion. Sequences with negative samples have an average length of 311 frames, which is about 15.5 seconds at 20 fps, and are suitable for 3D visual query localization with frequent occlusions and viewpoint changes. In future work, we will add 200 longer sequences for long-term 3D VQL research.

S4 Evaluation Metrics and 3D IoU

Evaluation Protocol. Inspired by the 2DVQL evaluation protocol in [2], we define the following metrics for 3D visual query localization under the Top-1 retrieval setting. For each query, the model outputs a single 3D response track with an associated confidence score. Average Precision (AP) is computed by ranking predictions according to confidence and integrating the precision–recall curve in the standard way.

Let $\mathcal{Q}$ be the set of all queries and $|\mathcal{Q}|$ its cardinality. We use $\mathcal{T} = \{0.25, 0.50, 0.75, 0.95\}$ for temporal IoU (tIoU) thresholds and $\mathcal{S} = \{0.05, 0.25, 0.50, 0.75, 0.95\}$ for 3D spatio-temporal IoU (stIoU_{3D}) thresholds.

Temporal Average Precision (tAP). tAP measures how well the predicted temporal interval matches the ground-truth response interval. For a given tIoU threshold $\delta \in \mathcal{T}$, we compute the temporal AP, denoted as $\mathrm{AP}^t(\delta)$. The final tAP is obtained by averaging over all tIoU thresholds:

$$\mathrm{tAP} = \frac{1}{|\mathcal{T}|} \sum_{\delta \in \mathcal{T}} \mathrm{AP}^t(\delta). \tag{1}$$

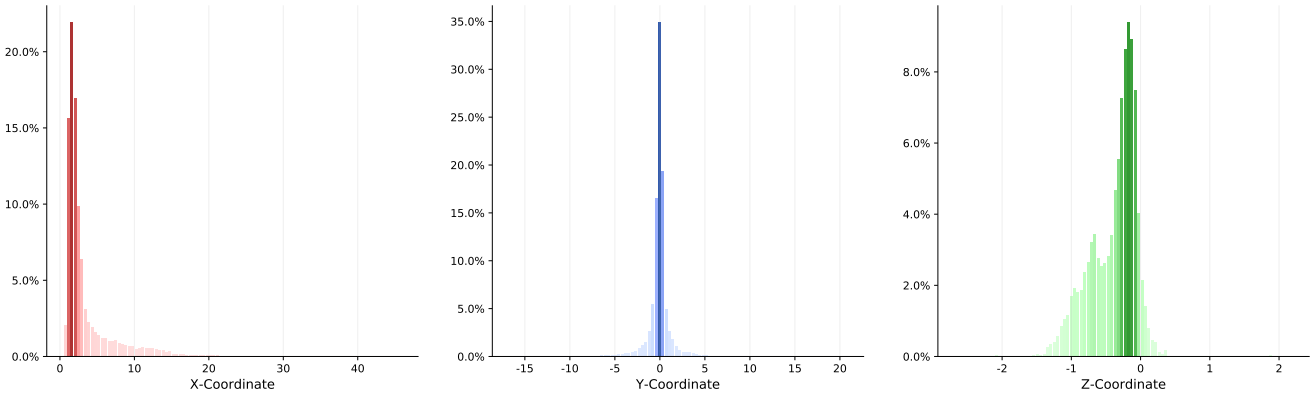

Figure 4. Statistics on 3DVQL. Distribution of target object center coordinates (X, Y, Z) in 3D space.

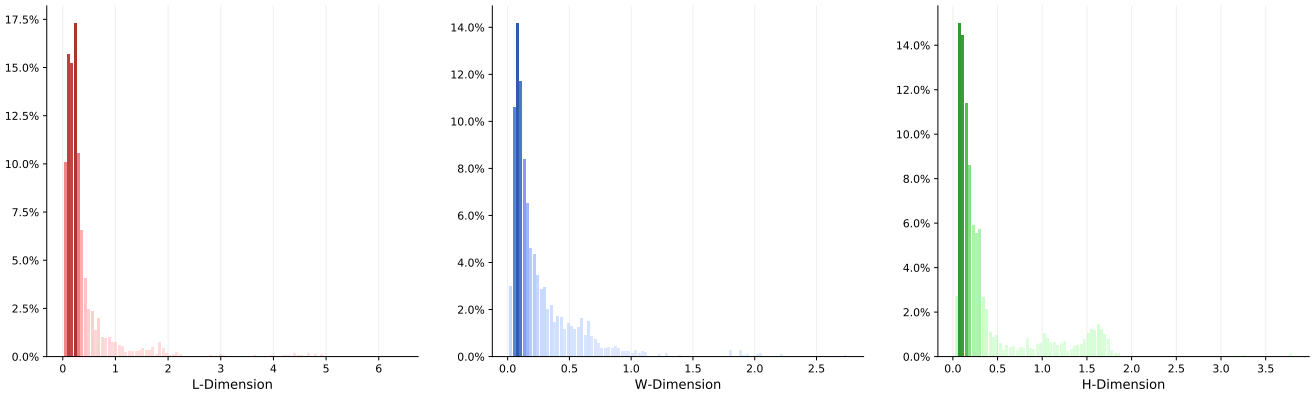

Figure 5. Statistics on 3DVQL. Distribution of target object sizes in length, width, and height (L, W, H).

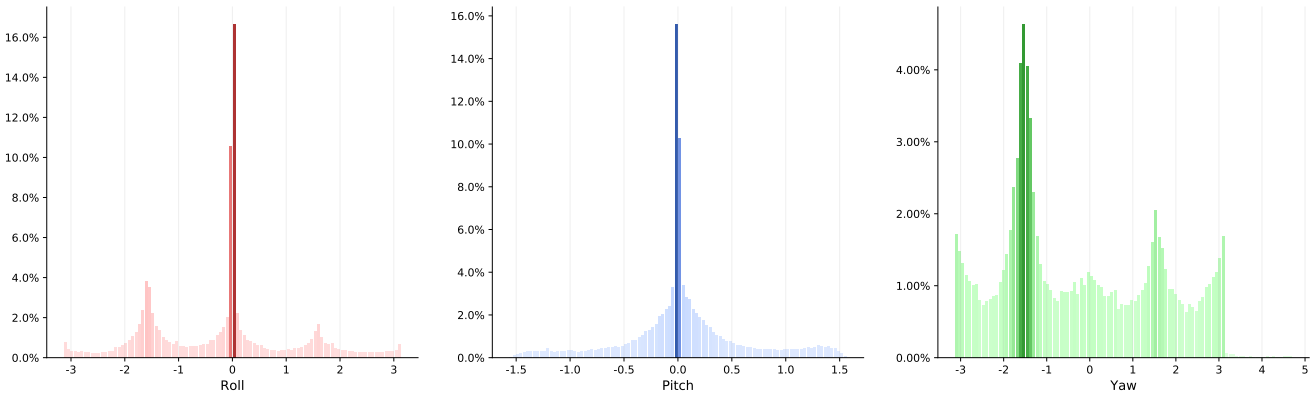

Figure 6. Spatial range statistics on 3DVQL: distribution of target object orientations in Roll, Pitch, and Yaw.

3D Spatio-Temporal Average Precision (3D-stAP). 3D-stAP evaluates how well the 3D spatio-temporal tube of the prediction aligns with the ground-truth response track. For query $q \in \mathcal{Q}$, let $\mathcal{F}_q$ be the set of frames within its ground-truth response interval, and $\mathrm{IoU}_{3D}^{q,t}$ the 3D IoU between the predicted and ground-truth oriented 3D bounding boxes at frame $t \in \mathcal{F}_q$. Each 3D box is parameterized by 9 degrees of freedom,

$$b = (x, y, z, l, w, h, \text{yaw}, \text{pitch}, \text{roll}), \tag{2}$$

where (x, y, z) denotes the 3D center, (l, w, h) the box size, and (yaw, pitch, roll) the three rotation angles around the

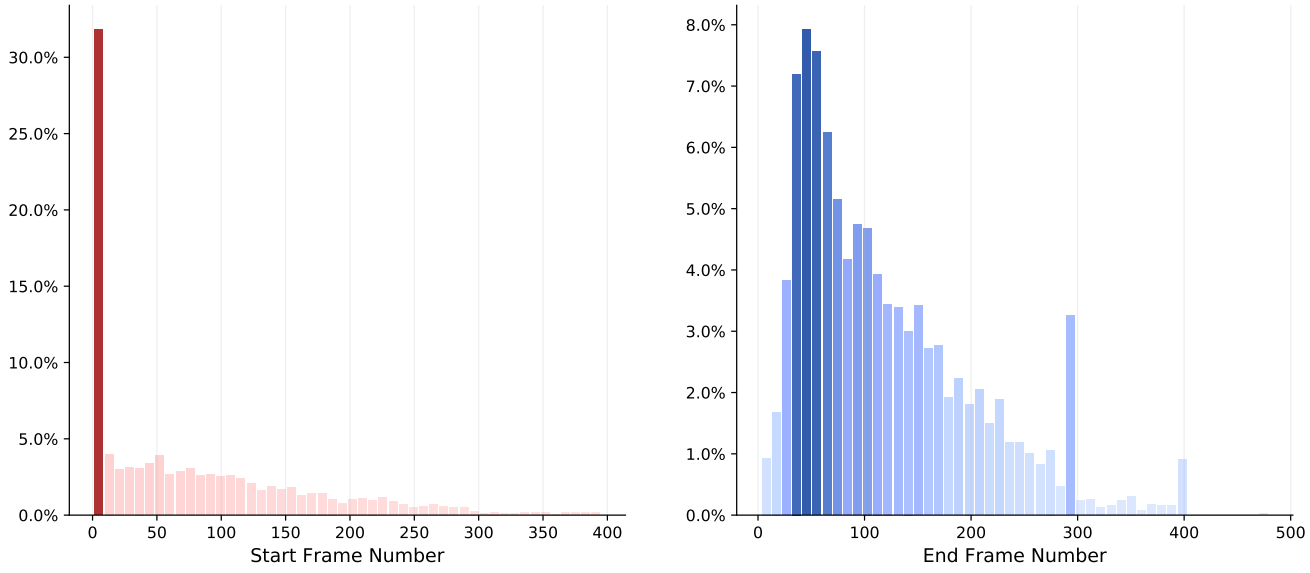

Figure 7. Statistics on 3DVQL: Distribution of the start and end frame indices of response tracklets.

vertical, lateral, and longitudinal axes, respectively. We first aggregate frame-wise 3D IoU over time to obtain the 3D spatio-temporal IoU,

$$\mathrm{stIoU}_{3D}(q) = \frac{1}{|\mathcal{F}_q|} \sum_{t \in \mathcal{F}_q} \mathrm{IoU}_{3D}^{q,t}. \tag{3}$$

For a given $\tau \in \mathcal{S}$, we then compute the AP at 3D spatio-temporal IoU threshold τ, denoted as $\mathrm{AP}^{\text{3D-st}}(\tau)$. The 3D-stAP is defined as

$$\text{3D-stAP} = \frac{1}{|\mathcal{S}|} \sum_{\tau \in \mathcal{S}} \mathrm{AP}^{\text{3D-st}}(\tau). \tag{4}$$

Given the increased difficulty of full 9-DoF localization (position, scale, yaw, pitch, and roll) in 3D space, we include a more permissive low threshold $\tau = 0.05$ in $\mathcal{S}$.

Success (Succ). Succ measures whether the prediction achieves any effective overlap with the ground truth in 3D spatio-temporal space. A query q is regarded as successful if its 3D spatio-temporal IoU satisfies $\mathrm{stIoU}_{3D}(q) \geq 0.05$. The overall success is the fraction of successful queries:

$$\mathrm{Succ} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbb{I}\big(\mathrm{stIoU}_{3D}(q) \geq 0.05\big), \tag{5}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

Recovery% (Rec%). Rec% focuses on how much of the ground-truth response track is recovered at the frame level. For each query q, let $\mathcal{F}_q$ be its ground-truth response frames as above. We count a frame $t \in \mathcal{F}_q$ as recovered if $\mathrm{IoU}_{3D}^{q,t} \geq 0.5$. Rec% is defined as the percentage of such recovered frames among all frames on all response tracks:

$$\mathrm{Rec\%} = \frac{\sum_{q \in \mathcal{Q}} \sum_{t \in \mathcal{F}_q} \mathbb{I}\big(\mathrm{IoU}_{3D}^{q,t} \geq 0.5\big)}{\sum_{q \in \mathcal{Q}} |\mathcal{F}_q|} \times 100\%. \tag{6}$$

This metric follows the robustness spirit of the VOT challenge [4] by emphasizing stable recovery of the target over time.

S5 Details of Baselines

In this section, we present a detailed elaboration on the baseline variants introduced in the main text. Given that the overall architecture has been previously outlined, the following discussion focuses specifically on the implementation details of the modifications in each variant.

AnchorFusion-3DVQL (AF) As illustrated in Fig. 8(b), the absence of a differentiable 9-DoF IoU operator impedes stable supervision across multiple candidate boxes. To address this, we adopt a 7-DoF Generalized IoU (GIoU) loss as the training objective, replacing the standard center-point regression loss. Specifically, the roll and pitch parameters are zeroed out during the GIoU calculation between ground truth and predicted boxes. The total loss function is formulated as:

$$\mathcal{L} = \lambda_{\mathrm{c}}\mathcal{L}_c + \lambda_{\mathrm{s}}\mathcal{L}_s + \lambda_{\mathrm{r}}\mathcal{L}_r + \lambda_{\mathrm{cls}}\mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{GIoU}}\mathcal{L}_{\mathrm{GIoU}}. \tag{7}$$

Guided-Attention Fusion-3DVQL (GAF) As depicted in Fig. 8(c), this variant integrates a Guided-Attention Fu-

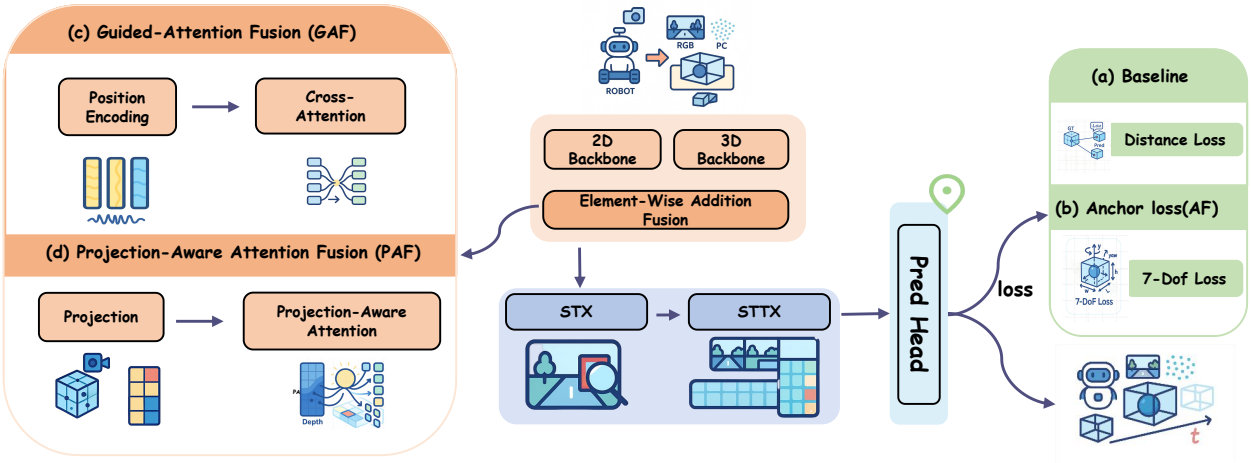

Figure 8. Architectural comparison of the baseline variants: (a) Baseline, (b) AnchorFusion-3DVQL (AF), (c) Guided-Attention Fusion (GAF), (d) Projection-Aware Attention Fusion (PAF).

sion module designed to explicitly inject depth cues via depth positional encodings and depth-aware blocks. In this setup, learnable parameters are augmented to the 3D features along the depth axis. Subsequently, the 2D and 3D features are fused through a depth-wise cross-attention mechanism.

Projection-Aware Attention Fusion-3DVQL (PAF) As shown in Fig. 8(d), this variant incorporates a Projection-Aware Attention Fusion module. Here, the centers of 3D voxels are projected onto the 2D image plane via the camera matrix, after which corresponding image features are sampled using bilinear interpolation. These sampled 2D features are then fused with the 3D voxel features via a multi-head attention mechanism along the depth axis, where the 3D features serve as queries and the sampled 2D features serve as keys and values.

S5.1 Additional Modality Settings and Broader Baselines

Although this paper mainly designs and evaluates baseline models under the RGB–PC setting, in order to further verify the role of the dataset, we supplement more modality settings, including RGB-only, PC-only, RGB-D, and PC-D. This makes it possible to more directly compare the effects of appearance information, geometric information, and their combinations in 3D visual query localization. In addition, we also include the point-tracking-style baseline TAPIP3D [7], as well as the two-stage baseline TWO-Stage built upon CenterPoint [6] and MBPTracker [5], for supplementary comparison with single-stage fusion methods. The results are shown in Tab. 2. The first four rows are modality variants based on LaF, and the last two rows are broader comparison baselines. It can be seen that, in embodied environments, using only RGB or Depth often leads to insufficient model learning, while settings with point cloud information usually perform better. This further shows the reasonableness of the RGB–PC fusion setting. At the same time, the results of TAPIP3D and TWO-Stage also show that 3DVQL is challenging for methods of different paradigms.

Table 2. Additional modality settings and broader baselines on 3DVQL. The first four rows correspond to modality variants, while the last two rows report broader baselines beyond the single-stage fusion framework.

Method	tAP	$\text{tAP}_{0.25}$	stAP	$\text{stAP}_{0.05}$	Rec.%	Succ.
LaF (RGB-only)	0.007	0.028	0.003	0.015	0.020	12.302
LaF (PC-only)	0.030	0.505	0.012	0.062	0.095	26.481
LaF (RGB-D)	0.135	0.335	0.006	0.035	0.055	16.516
LaF (PC-D)	0.225	0.545	0.015	0.075	0.115	30.632
TAPIP3D	0.143	0.485	0.007	0.041	0.063	18.824
TWO-Stage	0.162	0.411	0.009	0.045	0.075	21.431

S6 Qualitative Results

In this section, we present the visualization results of several proposed baseline methods (PAF, GAF, AF) and our method LaF in more scenarios using 3D VQL in Figure 1. As can be seen from Fig. 1, these baseline methods often struggle to consistently and accurately locate targets in 3D space in complex scenes with frequent occlusion and similar interference. In contrast, LaF can more stably complete spatial query and localization in such situations, demonstrating better robustness and adaptability.

S7 Maintenance and Responsible Usage of 3DVQL for Research

Maintenance. We will host 3DVQL on the widely used GitHub platform (where all dataset download links and our models will be publicly released). This allows us to promptly view feedback from the community and answer all user questions. It also enables us to maintain and update our benchmarks as needed by researchers, thus continuously improving them. Simultaneously, the authors will make every effort to collect and organize evaluation results of various algorithms on 3DVQL, forming a relatively complete and dynamically updated comparative analysis. Our ultimate goal is for 3DVQL to gradually develop into a long-term, sustainable public platform for multimodal 3D visual query localization research.

Responsible Usage of 3DVQL. 3DVQL aims to promote research and application related to three-dimensional visual query localization. It is development and use are limited to ***research purpose only***.

References

[1] Ankit Dhall, Kunal Chelani, Vishnu Radhakrishnan, and K Madhava Krishna. Lidar-camera calibration using 3d-3d point correspondences. *arXiv*, 2017. 2

[2] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012, 2022. 2, 4

[3] Yifan Jiao, Yunhao Li, Junhua Ding, Qing Yang, Song Fu, Heng Fan, and Libo Zhang. Gsot3d: Towards generic 3d single object tracking in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5469–5478, 2025. 2

[4] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, Ondrej Drbohlav, et al. The eighth visual object tracking vot2020 challenge results. In *European conference on computer vision*, pages 547–601. Springer, 2020. 6

[5] Tian-Xing Xu, Yuan-Chen Guo, Yu-Kun Lai, and Song-Hai Zhang. Mbptrack: Improving 3d point cloud tracking with memory networks and box priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9911–9920, 2023. 7

[6] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 7

[7] Bowei Zhang, Lei Ke, Adam W Harley, and Katerina Fragkiadaki. Tapip3d: Tracking any point in persistent 3d geometry. *arXiv preprint arXiv:2504.14717*, 2025. 7