PRVQL: Progressive Knowledge-guided Refinement for Robust Egocentric Visual Query Localization

Supplementary Material

For better understanding of this work, we provide additional details, analysis, and results as follow:

A. Detailed Architectures of Modules
In this section, we display the detailed architectures for the cross-attention block `CAB` and masked self-attention block `MaskedSA` in the main text.

B. Inference Details
We provide more details for the inference of PRVQL.

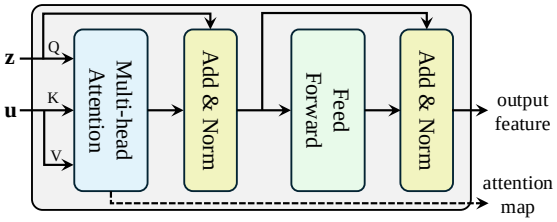
C. Additional Experimental Results
We offer more experimental results in this work, including more ablations and comparison of different method across different scales on the Ego4D dataset.

D. Visualization Analysis of Target Appearance and Spatial Knowledge
We provide visual analysis to show the learned target appearance and spatial knowledge.

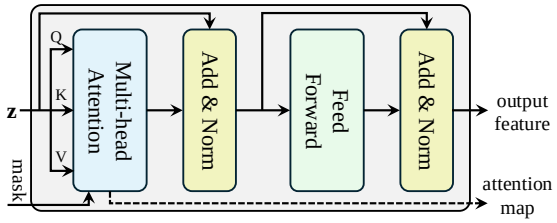
E. Computational Cost Analysis
We analyze computational cost of the single-stage baseline and our three-stage PRVQL.

F. Analysis on the Refinement Stages
We provide discussion on refinement stages of PRVQL.

G. More Qualitative Results
We demonstrate more qualitative results of our method for localizing the target object.



(a) Cross-Attention Block



(b) Masked Self-Attention Block

Figure 1. Detailed architectures of `CAB` and `MaskedSA`.

Table 1. Ablation studies on the parameter α in SKG.

		tAP_{25}	$stAP_{25}$	rec%	Succ
❶	α=0.4	0.33	0.26	47.27	57.24
❷	α=0.5	**0.35**	**0.27**	**47.87**	**57.93**
❸	α=0.6	0.31	0.26	46.34	55.97

Table 2. Ablation studies on combination methods in QFR.

	Method	tAP_{25}	$stAP_{25}$	rec%	Succ
❶	Addition	0.31	0.23	46.57	56.37
❷	Concatenation	0.34	0.25	47.31	56.64
❸	Cross-Attention	**0.35**	**0.27**	**47.87**	**57.93**

Table 3. Comparison on object of different scales in videos.

Method	Scale	tAP_{25}	$stAP_{25}$	rec%	Succ
CocoFormer	*small*	**0.067**	**0.030**	**19.565**	**21.113**
VQLoC	*small*	0.047	0.001	2.447	13.043
PRVQL (ours)	*small*	0.036	0.004	2.351	16.087
CocoFormer	*medium*	0.206	0.127	32.583	40.804
VQLoC	*medium*	0.213	0.138	33.738	44.719
PRVQL (ours)	*medium*	**0.261**	**0.179**	**34.359**	**49.923**
CocoFormer	*large*	0.338	0.271	40.737	56.164
VQLoC	*large*	0.454	0.387	**53.635**	67.680
PRVQL (ours)	*large*	**0.469**	**0.396**	52.127	**68.664**

A. Detailed Architectures of Modules

In each stage of PRVQL, we adopt the cross-attention block `CAB` to fuse the query feature into the video feature and then utilize the masked self-attention block `MaskedSA` for further enhancing the video feature. The architectures of `CAB` and `MaskedSA` are shown in Fig. 1.

B. Inference Details

Similar to [1], for inference, we first predict the confidence scores for target occurrence in all frames. Given the scores, we then smooth them through a median filter with the kernel size of 1. After this, we perform peak detection on the smoothed scores. We detect the peak based on the highest score h and use $0.79 \cdot h$ as the threshold to filter non-confident peaks. Finally, we can determine a spatio-temporal tube that corresponds to the most recent peak as the prediction result. In order to detect start and end time of the tube, we threshold the confidences scores using the threshold of $0.585 \cdot \tilde{h}$, where $\tilde{h}$ is the confidence score at the most recent peak.

Figure 2. Visualization of the target appearance knowledge.

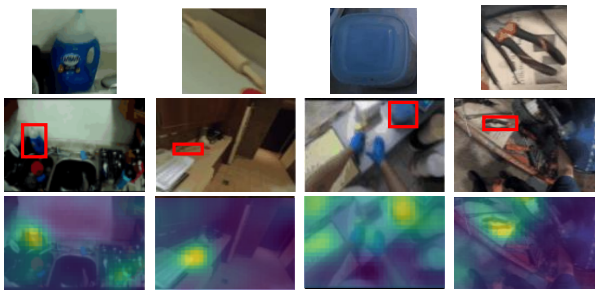


Figure 3. Visualization of the target spatial knowledge. First row: given visual query; second row: foreground target represented by red boxes; third row: learned target spatial knowledge.

Table 4. Computational cost comparison on the single A6000 GPU. The MACs and inference/training GPU memories are assessed on a 30-frame video clip.

Method	MACs	Infer. GPU Memory	Train. GPU Memory	stAP_{25}
Baseline (single stage)	1.7T	8G	11G	0.23
PRVQL (three stages)	1.8T	6G	8G	0.27

C. Additional Experimental Results

In this section, we show more ablation studies and comparison to other methods on the Ego4D validation set.

Impact of Balance Parameter α in SKG. The interpolated attention map $\varphi_{\text{int}}(\mathcal{T}_k^{\text{d}})$, obtained via bilinear interpolation from $\mathcal{T}_k^d$, is merged with $\mathcal{S}_k$ through a balance parameter α. We conduct an ablation on α in Tab. 1. We can observe that, when setting α to 0.5, we show the best result (see ❷).

Different Combination Methods in QFR. In QFR, the appearance knowledge $\mathcal{K}_k^a$, obtained by AKG, is used to guide the refinement of query feature. In PRVQL, we use a cross-attention block to combine $\mathcal{K}_k^a$ and $\mathcal{Q}_k$ for achieving refinement. Besides cross-attention, we conduct experiments using other manners for refinement, including element-wise addition and concatenation, in Tab. 2. As shown in Tab. 2, when using the cross-attention block for query feature refinement, we achieve the best performance (see ❸).

Comparison in Different Scales. Following [1], we provide comparison for objects of different scales in videos. As in [1], the objects are categorized to three scales, including

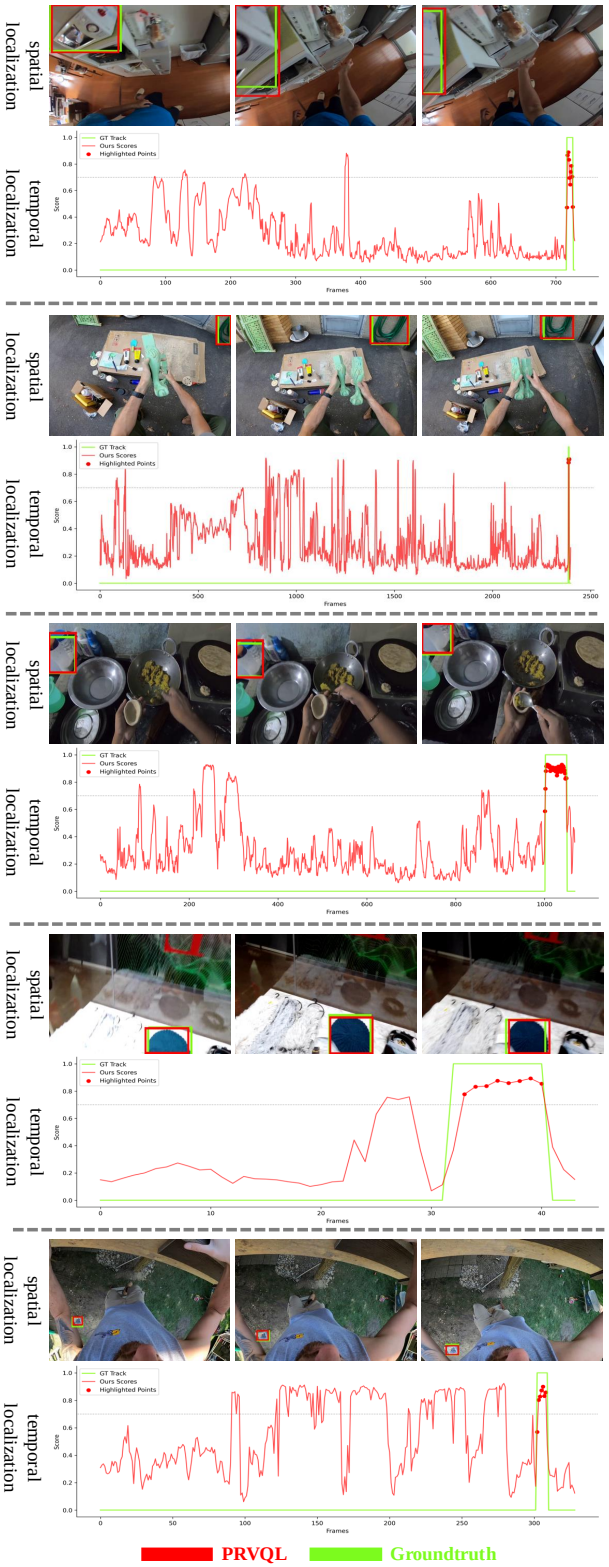


Figure 4. Qualitative results of our method.

small scale with target area in the range of [0, 64^2], *medium* scale with the target area in the range (64^2, 192^2], and *large* scale with target area greater than 192^2. Tab. 3 reports the comparison result. As in Tab. 3, we can observe that, CocoFormer performs better for small-scale objects. We argue that the reason is CocoFormer adopts higher-resolution images for localization and employs detector that is good at small object detection, while VQLoC and our method use downsampled frames for localization and do not specially deal with small objects. In comparison to CocoFormer and VQLoC, our PRVQL achieves better overall performance for medium- and large-scale objects, which shows the efficacy of target knowledge for robust target localization.

D. Visualization Analysis of Spatial Knowledge

The target appearance and spatial knowledge by AKG and SKG aim to explore target cues from videos for improving EgoVQL by refining video features. We show the learned target appearance knowledge in Fig. 2. From Fig. 2, we can see that, the mined target cues can greatly improve the discriminative power of query when used for refining the query feature, thereby boosting performance. In Fig. 3, we show the learned target spatial knowledge that leverages the readily available attention maps. From Fig. 3, we can see that, our spatial knowledge focuses more on the target object while less on the background, and thus can be applied to refine video features for better localization.

E. Computational Cost Analysis

We analyze the computation cost of the single-stage baseline and three-stage PRVQL, including MACs (T), and inference and training GPU memory (G) in Tab. 4. As shown, despite using three stages, PRVQL only results in slight increment in computation, yet largely boosts the performance.

F. Analysis on the Refinement Stages

In our PRVQL, more refinement generally benefits the localization for cases with large pose variations, occlusion, motion blur, and background clutter (see Fig. 7 in the main text). Nonetheless, when using more stages, *e.g.*, increasing the number of stages from 3 to 4, the performance slightly drops. We argue that this may be caused by accumulated noises or overfitting in more refinement stages. Despite this, it improves baseline with no refinement from 0.23 to 0.26 in stAP_{25} as shown in the ablation on the number stages.

G. More Qualitative Results

In order to further validate the effectiveness of our PRVQL, we provide additional examples of target localization results in Fig. 4. From the shown visualizations, we can observe that, with the help of target knowledge, our method can accurately locate the target in both space and time.

References

[1] Hanwen Jiang, Santhosh Kumar Ramakrishnan, and Kristen Grauman. Single-stage visual query localization in egocentric videos. *NeurIPS*, 2023. 1, 2