Ego: Embedding-Guided Personalization of Vision-Language Models (Appendix)

Soroush Seifi* Simon Gardier Vaggelis Dorovatas* Daniel Olmeda Reino Rahaf Aljundi
Toyota Motor Europe
{soroush.seifi, vaggelis.dorovatas}@external.toyota-europe.com
{simon.gardier, daniel.olmeda.reino, rahaf.al.jundi}@toyota-europe.com

1. Implementation Details

All experiments are conducted at a fixed input resolution of 448×448 across all datasets and methods. Images are normalized using the LVLM's default preprocessing pipeline, and tiling is disabled (set to 1) to minimize computational overhead. R2P [1] and RAP [2] are adapted to InternVL3-14B based on their official Github implementations, while PeKit [7] is re-implemented from scratch following the original paper. Inference is performed with $2\times$ A100 GPUs. Fine-tuning InternVL3-14B with RAP [2] is carried out using $8\times$ A100 GPUs, a batch size of 6, and a LoRA rank of 32. No gradient updates are applied for the remaining methods. For RAP [2], cosine similarity is used as the similarity metric, and a `top-k = 3` concept candidate retrieval is adopted for evaluation where applicable. All experiments are conducted using Pytorch 2.9. *Ego*'s implementation will be available on Github.

2. Ablation

In this section, we present ablation studies to assess the impact of key design components in *Ego*. Most experiments in this section are conducted on the **This-Is-My** dataset (single concept) the **InternVL3-14B** model, focusing primarily on the **recognition task**.

2.1. Dynamic K_c vs Fixed K

In this section, we analyze the impact of using a dynamic concept memory size K_c—as introduced in Section 3.3 of the main paper—compared to enforcing a fixed memory size K on the recognition task. We consider a baseline with a constant $K = 50$ for all concepts, represented using a single reference view. Note that *Ego* caps the number of visual tokens to the same $K = 50$ to maintain a minimal number of visual tokens representing an concept.

As shown in Table 1, the dynamic memory selection strategy outperforms the fixed-budget approach (while using less token budget). A representative example is the concept *Zak's Dog Coffee* from the This-is-My dataset. As illustrated in Fig. 1, a fixed K forces the model to store extra background tokens that may confuse the model for recognizing the object in novel environments. In contrast, the dynamic strategy enables *Ego* to effectively filter out such background noise, yielding higher recall and a substantially improved F1-score. This confirms that aligning the memory size with the visual footprint of the object is crucial for robust personalization, particularly for smaller objects.

Table 1. K vs K_c: *Ego* adapts to the size of the target concept in the reference views and boosts the performance compared to a fixed memory size selection strategy.

Experiment	Prec.	Rec.	F1
	All concepts		
$K_c \leq 50$	81.3	77.0	**79.1**
$K = 50$	82.4	74.3	78.1
	Zak's Dog Coffee 1		
$K_c \leq 50$	81.8	34.6	**48.6**
$K = 50$	100.0	3.85	7.41

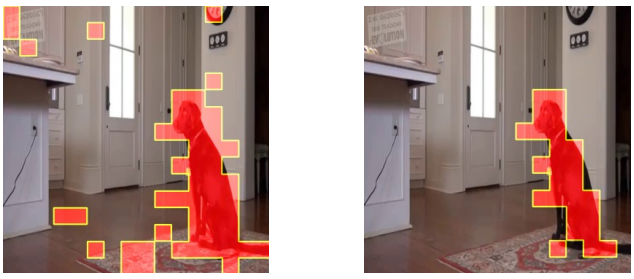

Figure 1. Extracted visual tokens for *Zak's Dog Coffee*. **Left:** Fixed $K = 50$ **Right**: Dynamic $K_c = 25$. *Ego* removes 25 background patches by adapting to the concept's size.

*Providing contracted services at Toyota Motor Europe.

Table 2. LVLM Layer selection: Parentheses indicate the selected layers for InternVL3-14B. *Ego* outperforms the baselines by emphasizing more discriminative features while suppressing background noise.

Experiment	Prec.	Rec.	F1
Ego (29, 30, 35, 36, 39)	81.3	**77.0**	**79.1**
Manual selection (20-24)	**81.7**	73.5	77.4
Uniform layer selection	78.7	73.1	75.8

2.2. Layers Selection Strategy

In Section 3.4 of the main paper, we proposed an LVLM-agnostic strategy to select the layers that exhibit the strongest text–visual token interaction. We evaluate its effectiveness by comparing *Ego* with two baselines: manually choosing five consecutive intermediate layers inspired by prior work [3], and uniformly sampling five layers across the network.

As shown in Table 2, *Ego* surpasses the baselines, with uniform layer selection performing the worst. Manual selection exhibits a substantial decline in recall, highlighting the effectiveness of our layer selection strategy in achieving optimal performance.

It is important to note that personalized concepts typically occupy a large portion of the reference views in current personalization datasets. As a result, even a uniform selection of 20% of visual tokens is likely to include tokens overlapping with the target subject, resulting in a reasonable F1-score even when the selected LVLM layers are not tuned.

Nevertheless, the **3.3%** gain achieved by *Ego* over the uniform layer selection represents the utility of our automatic layer selection in identifying the key layers where visual tokens are interacting the most with the generated textual keywords.

2.3. Full-caption vs Keywords Attention-based Selection

As detailed in Sections 3.1 and 3.2 of the main paper, *Ego* leverages cross-modal attention maps to identify the most informative visual tokens that contribute to generating keywords describing the personalized concepts in the reference views. Table 3 compares this approach with using full concept captions, which include generic terms (e.g., pronouns, articles) and sometimes background details. Although the performance gap is small in the 1-view setting, it becomes more pronounced with 5 views, indicating that full descriptions introduce cumulative noise from background tokens or attention sinks.

Table 3. Full description vs Keywords cross-attention on This-is-my single concept recognition task

Experiment	Prec.	Rec.	F1
	1 Ref. View		
Keywords	**81.3**	77.0	**79.1**
Caption	80.4	77.6	79.0
	5 Ref. views		
Keywords	86.1	**68.7**	**76.5**
Caption	**87.2**	65.9	75.1

Table 4. Visual token selection over 5 samples vs no sampling on This-is-my single concept recognition task.

Experiment	Prec.	Rec.	F1
	1 Ref. View		
No sampling	**81.3**	**77.0**	**79.1**
$N = 5$	81.2	75.9	78.5
	5 Ref. views		
No sampling	**87.2**	**65.9**	**75.1**
$N = 5$	86.8	63.5	73.4

2.4. Sampling for Keywords Attention-based Selection

In this section, we assess whether increasing diversity in the descriptive keywords via sampling can improve the robustness of the selected visual memory, compared to a deterministic decoding strategy (temperature=0, do_sample=False). We compare *Ego* against an aggregation-based baseline, in which keywords are sampled from five non-deterministic runs (do_sample=True), and the resulting attention scores are averaged prior to selecting the top-K_c visual tokens.

As shown in Table 4, the deterministic decoding strategy yields the strongest performance, suggesting that the most confident generated keywords describe the personalized concept better and are more reliably grounded in the visual content. Conversely, sampling introduces randomness that can produce weakly relevant or even hallucinated attributes. These noisy terms may divert attention toward irrelevant image regions, degrading the quality of the extracted concept memory and leading to reduced performance.

2.5. *Ego* vs. Full reference Views and Base VLM

In this section, we compare *Ego*, which selects a compact subset of highly attended visual tokens from a single reference view per concept, against a baseline that provides the entire reference image as in-context input. We also include

Table 5. **VQA Accuracy and Captioning Recall.** Full Ref. View vs *Ego* vs Base VLM on all This-is-my tasks.

Method	Single VQA	Multi VQA	Video VQA	Single Recognition			Multi Recognition		
				Prec.	Rec.	F1	Prec.	Rec.	F1
Ego (Ours)	**88.0**	**72.2**	**70.0**	81.3	**77.0**	**79.1**	93.9	**78.2**	**88.6**
Full Ref. View	86.0	66.7	68.2	**84.3**	67.7	75.1	**100.0**	65.4	79.1
Base VLM	86.0	55.7	55.1	50.3	32.3	39.3	81.4	16.4	27.2

a **Base VLM** configuration, where the model is queried without reference images or concept memories, in order to quantify the benefit of personalization over blind predictions by a non-personalized LVLM.

As shown in Table 5, *Ego* consistently surpasses the Full Reference View baseline while using less than one-fifth of the context budget and eliminating the need to repeatedly reprocess the reference view through the LVLM's vision encoder. *Ego* only processes a reference view once when the concept is introduced.

On the **Recognition** task, the Full Reference View achieves perfect precision (**100%**) in the multi-concept setting, but suffers from substantially lower recall compared to *Ego* (**65.4%** vs. **78.2%**). This indicates that *Ego* effectively removes background content that biases the model toward rejecting images where the concept appears in novel environments, resulting in higher recall and F1-score (**79.1%** vs. **88.6%**). The benefits of reducing noise in concept memory and extracting representative visual tokens become even more evident in more complex tasks such as VQA, where *Ego* consistently surpasses the baseline by a significant margin.

Overall, these results demonstrate that providing full reference views as in-context input not only incurs higher computational overhead but also injects harmful background noise. In contrast, the attention-guided selection used by *Ego* yields a compact yet highly effective concept memory, eliminating irrelevant background information and removing the need to re-encode reference images at inference time.

Finally, the Base VLM results highlight the necessity of personalization. Without access to concept-specific information, the model's performance drops across all tasks. Although the Base VLM retains reasonable accuracy on the Single VQA task—likely by leveraging general visual cues to make an informed guess between the two available options—it performs poorly elsewhere. In particular, on the Multi-concept recognition task, the F1-score drops drastically from **88.6%** to **27.2%**. These findings confirm that the strong performance of *Ego* does not stem from dataset biases or generic LVLM capabilities, but instead arises directly from the effectiveness of our attention-guided personalization strategy.

Table 6. *Ego*'s performance with various model sizes and reference objects segmented with G-SAM on Yo'LLaVA dataset.

InternVL Size	Prec. ↑	Rec. ↑	F1 ↑
2B (tiny)	48.2	**91.5**	63.1
8B (small)	88.4	81.1	84.6
14B (medium)	**85.0**	86.4	**85.7**
14B & **G-SAM**	83.2	94.3	88.4

2.6. Performance on smaller models

Kang et al. [4] present a systematic evaluation of in-context learning across VLM families and scales, demonstrating that while recent high-capacity VLMs exhibit strong in-context abilities, tiny models (<4B parameters) still lack visual in-context learning. In contrast, models in the small–medium range (4B–40B) consistently retain this capability. Our findings in Table 6 reflect this trend: for Yo'LLaVA dataset, InternVL3-14B achieves the strongest performance, the 8B variant remains competitive, and the 2B model shows reduced precision due to increased false positives. Moreover, as discussed in the main paper, *Ego* assumes a VLM with sufficient in-context reasoning ability; we argue that, given the availability of capable open-source models in this size range, older models naturally become less relevant. *Ego* thus enables training-free personalization for modern, capable small- and medium-scale VLMs.

2.7. *Ego* with segmentation masks

Ego's attention-guided token selection is inherently a soft region localization method. Therefore, in this section we compare it to the scenario where we incorporate Grounded-SAM (G-SAM) [6] to obtain segmentation masks of reference objects. Tab. 6 (last row) shows that full segmentation masks improve recall and increase F1 on Yo'LLaVA dataset but at the cost of lower precision, reliance on an external model, added compute/memory, and semantic category supervision. As it encodes full-mask embeddings, it includes many redundant (possibly un-informative) patches that can exceed in-context limits when many concepts appear. In contrast, *Ego* achieves competitive performance without external modules or semantic labels via attention-guided, dy-

namic patch selection, keeping concept memory compact and informative.

3. Prompt Templates

In this section, we detail the prompts used in different parts of our pipeline for personalization and its evaluation. For simplicity, we provide prompts concerning a single personalized object here. However, multi-concept scenarios could be solved by incrementally appending similar prompts for each personalized subject to the query.

3.1. *Ego*'s Prompts

As discussed in Section 3.2 of the main paper, for each concept c, we identify a minimal subset of visual tokens $\mathbf{X}_R^c$ from the reference views R_c that best represent the concept. The selection process is guided by cross-modal attention between the visual tokens and the model-generated descriptive keywords. The number of selected tokens is defined based on an estimated concept size within each reference view. To obtain the size estimates and keywords, we use the prompt format shown below when querying the LVLM.

Concept-Size Estimation

```
Please analyze the image and estimate the
↪ percentage of the total image area that
↪ the main subject occupies. If you can
↪ not answer, say 0%. Answer **only** the
↪ percentage. Answer example: 50%.
```

Keywords Generation

```
Give me a list of important words to
↪ describe the **main** subject of the
↪ image (e.g. blue wheels, green eyes,
↪ zigzag pattern, tinted windows...).
↪ Provide the list in this exact format:
↪ <characteristic 0>, <characteristic 1>,
↪ <characteristic 2>,... . Do not answer
↪ anything else than the list of
↪ important words. Do not mention
↪ anything about the background or other
↪ objects in the image.
```

In-context Prompting As mentioned in section 3.5 of the main paper, at inference, we retrieve the memories $\mathbf{X}_R^c$ of the personalized concepts and inject them into the context of the LLM as soft-prompts.

```
Image <i> shows the entity <c>. Image <i>:
↪ X_R^c.
```

Depending on the task the in-context prompt might include one or more concepts from the concept set C. $< i >$ denotes each concept's corresponding index in the personalized object list containing I concepts in total.

Recognition Given that LVLMs exhibit varying sensitivities to prompt formulations, usually instruction prompts are specific for each model and task. Since our method is training-free, a specific prompt is crafted based on the base model's behavior. We noticed that for more capable models, as in the case of InternVl3, a simple instruction prompt is sufficient. However, we observed that QWEN2.5-VL is trained to abstain from saying yes on people, for this, we adapted the prompt to allow for first probability estimation and then instruct the model to provide a final answer based on the estimated probability of object presence.

InternVL3

```
Focusing on each subject's distinctive
↪ features, check the presence of the
↪ subjects **one by one** in the **new**
↪ {media}. Answer with the following
↪ template: subject_name: yes/no.
```

Where the {media} placeholder could be any of 'Video' or 'Image'.

Qwen2.5-VL

```
Do you see any entity in IMAGE <I+1> that
↪ resemble <c>? It can appear in a
↪ different context, pose or size in
↪ IMAGE <N+1>. Answer with a similarity
↪ score between 0 - 100. If the
↪ similarity score is lower than 50, give
↪ the Final Answer as 'No', otherwise as
↪ 'Yes'. Your answer should follow this
↪ format: Similarity Score: [0-100] Final
↪ Answer: [Yes/No].
```

VQA For the VQA task we simply append the question to the prompt:

```
Answer the following question about Image
↪ <I+1>: {question}
```

Captioning For the captioning task we use the following prompt:

```
Generate a detailed caption describing what
↪ you see in Image <I+1>. If an entity
↪ was detected, include its given name in
↪ the caption.
```

3.2. Autograding with GPT

As explained in Section 4.1 of the main paper, we adapt an autograding mechanism via GPT 3.5, introduced by [5] and used by [7], to measure the open-ended VQA performance of different methods. Specifically, we use the following prompt template:

```
You are an intelligent chatbot designed for
↪ evaluating the correctness of
↪ generative outputs for question-answer
↪ pairs. Your task is to compare the
↪ predicted answer with the correct
↪ answer and determine if they match
↪ meaningfully. Here's how you can
↪ accomplish the task:
```

```
INSTRUCTIONS:
- Focus on the meaningful match between the
 ↪  predicted answer and the correct
 ↪  answer.
- Consider synonyms or paraphrases as valid
 ↪  matches.
- Evaluate the correctness of the
 ↪  prediction compared to the answer.
Please evaluate the following
 ↪  question-answer pair:
Question: {question}
Correct Answer: {answer}
Predicted Answer: {pred}
Provide your evaluation only as a Yes/No.
DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR
 ↪  EXPLANATION.
```

4. Qualitative Results

In this section, we provide additional qualitative ablations of our method. Figure 2 visualizes the visual tokens extracted by *Ego* across multiple personalization datasets, overlaid on the corresponding reference images. As shown, *Ego* reliably localizes the target concepts and provides accurate estimates of their spatial extent. However, as noted in Section 2.1 of the appendix, objects in the reference views often occupy a large portion of the image. To maintain efficiency and avoid excessive redundancy, we therefore cap the number of extracted visual tokens per reference view to 50.

Figure 2 also compares *Ego* to existing personalization approaches on the multi-concept split from the This-is-My dataset. RAP's training-based personalization strategy [2] can degrade the base VLM's generation behavior, occasionally producing incorrect or redundant text. PeKit [7] relies on a fixed similarity threshold for instance identification, which is sub-optimal for distinguishing intra-category instances (e.g., Alex's bag), and its bounding-box overlays may introduce visual biases that lead to hallucinated answers. In contrast, *Ego* consistently produces correct responses, even in cases where prior methods fail.

References

[1] Deepayan Das, Davide Talon, Yiming Wang, Massimiliano Mancini, and Elisa Ricci. Training-free personalization via retrieval and reasoning on fingerprints. *arXiv preprint arXiv:2503.18623*, 2025. 1

[2] Haoran Hao, Jiaming Han, Changsheng Li, Yu-Feng Li, and Xiangyu Yue. Rap: Retrieval-augmented personalization for multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14538–14548, 2025. 1, 5, 7

[3] Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in middle layers of large vision-language models: Interpreting, detecting and mitigating object hallucinations via attention lens. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 25004–25014, 2025. 2

[4] Zhiqi Kang, Rahaf Aljundi, Vaggelis Dorovatas, and Karteek Alahari. Online in-context distillation for low-resource vision language models. *arXiv preprint arXiv:2510.18117*, 2025. 3

[5] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12585–12602, 2024. 4

[6] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 3

[7] Soroush Seifi, Vaggelis Dorovatas, Daniel Olmeda Reino, and Rahaf Aljundi. Personalization toolkit: Training free personalization of large vision language models. *arXiv preprint arXiv:2502.02452*, 2025. 1, 4, 5, 7

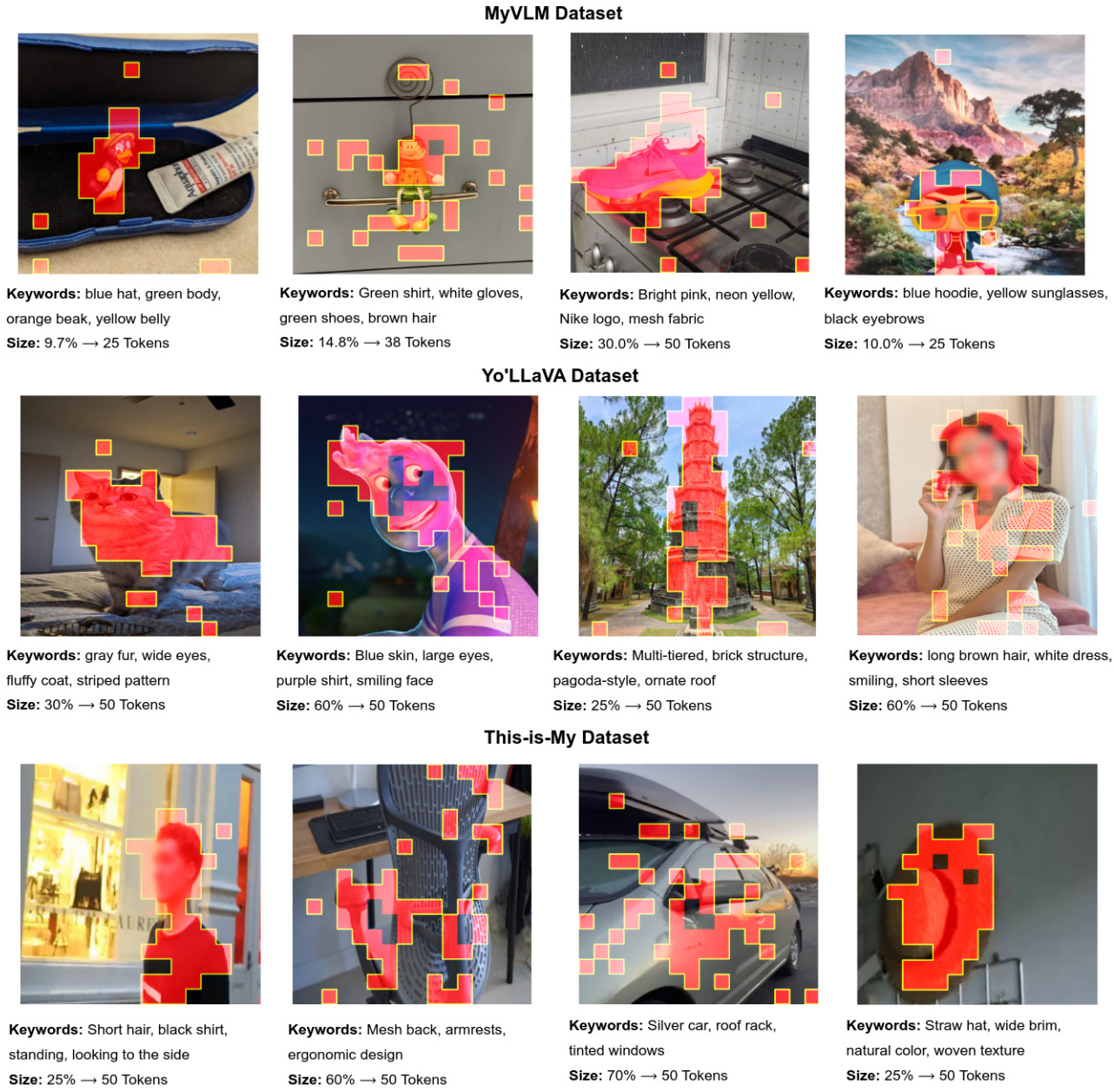

Figure 2. *Ego*'s generated keywords, estimated concept sizes, and selected patches are shown using examples from various datasets. *Ego* demonstrates the ability to accurately estimate concept sizes and extract informative patches for each object while minimizing background interference.

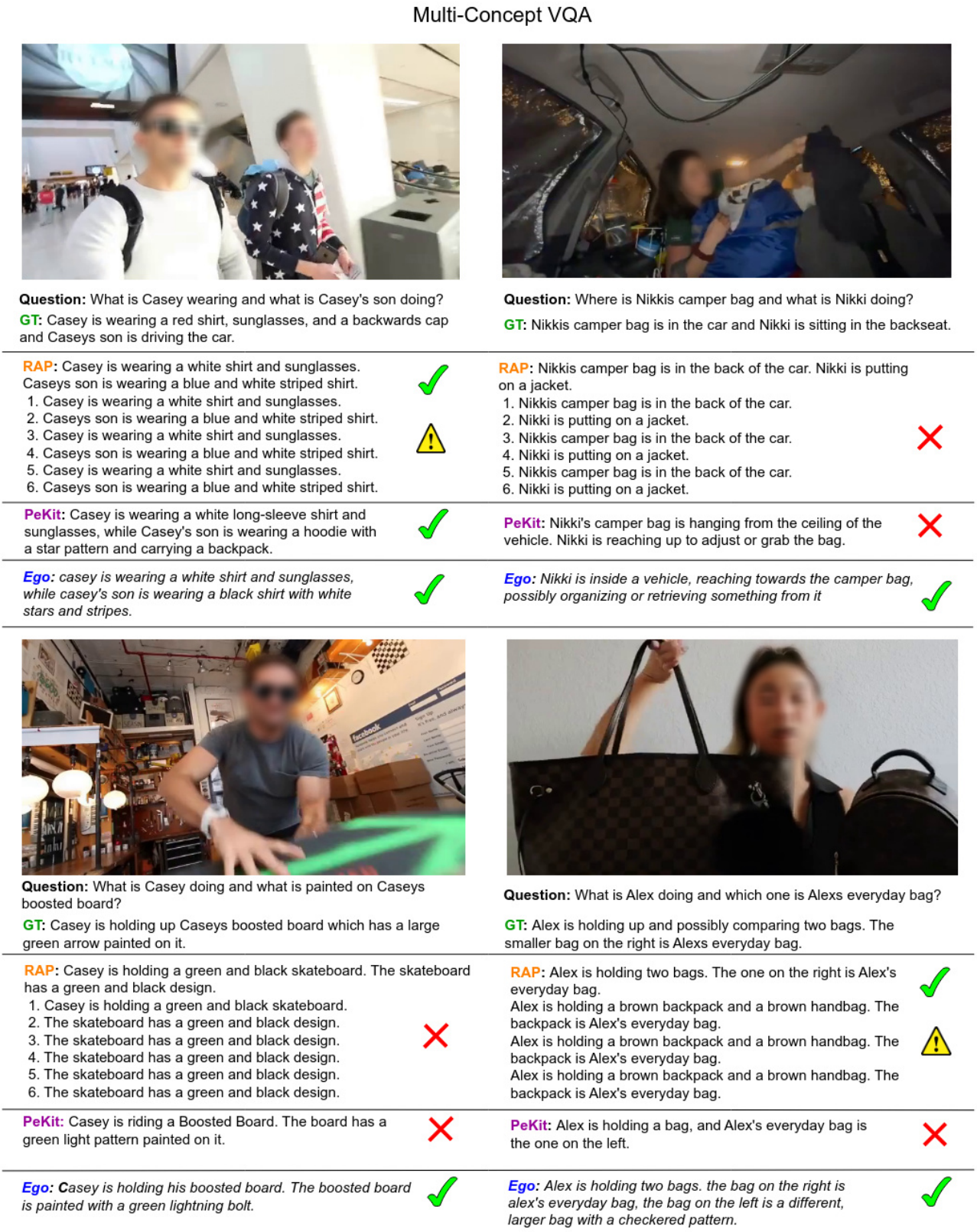

Figure 3. Qualitative results on multi-concept VQA in the This-is-My dataset. *Ego* maintains accurate responses, while RAP [2] modifies the model's behavior and PeKit [7] fails on fine-grained distinctions.