PEARL: PERSONALIZED STREAMING VIDEO UNDERSTANDING WITH RUNTIME-DEFINED CONCEPTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Human cognition of new concepts is inherently a streaming process: we continuously recognize new objects or identities and update our memories over time. However, current methods are largely limited to static images or offline videos. This disconnects continuous visual input from instant real-world feedback, limiting their ability to provide the real-time, interactive personalized responses essential for future AI assistants. To bridge this gap, we first propose and formally define the novel task of Personalized Streaming Video Understanding (PSVU). To facilitate research in this new direction, we introduce PEARL-Bench, the first comprehensive benchmark designed specifically to evaluate this challenging setting. It evaluates a model's ability to respond to runtime-defined personalized concepts across Concept-Definition QA, Real-Time QA, and Past-Time QA, each further organized into fine-grained reasoning categories. Concept-Definition QA introduces user-defined concepts into the stream, Real-Time QA queries their current states and includes a new Action category for clip-grounded personalized motion concepts in addition to frame-grounded entity concepts, and Past-Time QA tests long-range retrieval of historical personalized evidence. PEARL-Bench comprises 132 unique videos and 2,173 fine-grained annotations with precise timestamps. To tackle this challenging new setting, we further propose PEARL, a plug-and-play strategy that serves as a strong baseline. Evaluations across 8 offline and online models demonstrate that it brings consistent PSVU improvements when applied to 3 distinct architectures, proving to be highly effective and robust. We hope this work advances vision-language model (VLM) personalization and inspires further research into streaming personalized AI assistants.

1 INTRODUCTION

Recent advancements in Vision-Language Models (VLMs) (Wang et al., 2025; Bai et al., 2025; Li et al., 2024; Yu et al., 2025; Google DeepMind, 2025; Zhang et al., 2024a) have remarkably expanded the boundaries of multimodal understanding, empowering models to recognize and interact with personalized, user-specific concepts. Despite these strides, current personalization methods (An et al., 2024; Nguyen et al.; Hao et al., 2025; Shi et al., 2025; Xu et al., 2025; An et al., 2025; Nguyen et al., 2025; Yang et al., 2025a; Kim et al., 2025) remain fundamentally constrained. As shown in Fig. 1, approaches such as Yo'LLaVA (Nguyen et al.) and MC-LLaVA (An et al., 2024) are mainly designed for static image-text tasks. Furthermore, while PVChat (Shi et al., 2025) pioneers personalized video understanding, it operates strictly in offline settings and only supports single turn interaction, failing to accommodate the open-ended, streaming nature of real-world environments.

In contrast, humans continuously recognize new individuals and objects, forming memories over time as they process the world as a seamless visual stream. This fundamental cognitive mechanism highlights a critical limitation of existing methods, which remain confined to static images or pre-recorded videos. Bridging this gap is not merely a technical step, but an essential prerequisite for the next generation of personalized AI assistants (Ahmed & Aziz, 2025): such systems must be capable of handling streaming visual inputs and delivering real-time, interactive, and personalized responses in dynamic real-world environments (Gasteiger et al., 2023). For instance, in customized fitness coaching (Fig. 1), an AI assistant must continuously monitor a user's specific weightlifting

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



Figure 1: Comparison between our proposed **P**ersonalized **S**treaming **V**ideo **U**nderstanding (PSVU) task and traditional Personalized Image/Video Understanding tasks. Unlike traditional settings, PSVU better aligns with real-world scenarios by featuring continuous streaming video inputs, dynamically defined concepts, and multi-turn conversations regarding these customized concepts.

actions across a video stream to provide instant, tailored form correction. This real-time, streaming personalization capability is indispensable for deploying truly practical AI assistants.

To bridge this gap, we first propose and formally define the novel task of Personalized Streaming Video Understanding (PSVU). To facilitate research in this new direction, we introduce PEARL-Bench, the first comprehensive benchmark specifically designed to evaluate personalized streaming video understanding. Unlike traditional offline tasks, PEARL-Bench distinguishes itself through two core properties: (1) *Continuous Temporal Precision*, requiring models to localize and reason about personalized concepts at exact timestamps within an ongoing stream, and (2) *Interactive Concept Definition*, challenging models to grasp user-specific concepts dynamically defined on-the-fly, rather than relying on predefined pools. To thoroughly assess model capabilities, the benchmark covers three QA types: Concept-Definition QA for dynamically registering user-defined concepts, Real-Time QA for querying their current states, and Past-Time QA for retrieving historical personalized evidence. Real-Time QA further includes an Action category that evaluates clip-grounded personalized motion concepts, extending beyond frame-grounded entity concepts. PEARL-Bench comprises 132 unique videos, and 2,173 fine-grained annotations with precise timestamps. The video data is carefully curated through a combination of expert manual collection and rigorous programmatic synthesis pipelines. By sourcing data from diverse domains including anime, movies, reality shows, and digital humans, we ensure both extensive concept diversity and high annotation quality.

The challenging PSVU task presents significant efficiency and architectural hurdles for existing models, as they struggle to maintain streaming visual context and instantly acquire new concepts without computationally expensive retraining. To tackle this, we propose PEARL, a plug-and-play framework designed to serve as a strong baseline. Specifically, **PEARL** features a Dual-grained Memory System that explicitly decouples concept-centric knowledge from stream-centric observations, incrementally archiving continuous video clips while dynamically registering user-defined concepts. To ensure fast and accurate response, we further introduce a Concept-aware Retrieval Algorithm that leverages stored concept descriptions to precisely retrieve relevant historical visual evidence. Consequently, without any parameter updates, PEARL seamlessly empowers off-the-shelf VLMs to deliver real-time, personalized responses in continuous video streams. Extensive evaluations on 8

offline and online models demonstrate PEARL's effectiveness on the PSVU task. Notably, equipping models with PEARL drives consistent improvements across 3 distinct architectures, yielding average performance gains of 15.00% on Real-Time QA and 12.19% on Past-Time QA, thereby proving its generality and robustness.

We summarize our contributions as follows:

- **New Task and Benchmark:** We are the first to propose and formally define the novel task of Personalized Streaming Video Understanding. To facilitate evaluation in this direction, we introduce PEARL-Bench, the first comprehensive benchmark specifically designed for this challenging setting.
- **Novel Framework:** We propose PEARL, a novel, plug-and-play method. By seamlessly integrating into existing models, it demonstrates remarkable effectiveness and robustness across multiple architectures.
- **Substantial and Generalizable Gains:** Extensive experiments over 8 offline and online video understanding methods show that PEARL delivers consistent and substantial gains, averaging 15.00% on Real-Time QA and 12.19% on Past-Time QA across 3 distinct architectures. We hope this work inspires the field of VLM personalization and paves the way for next-generation interactive AI assistants.

2 PEARL-BENCH

2.1 TASK DEFINITION

In the task of Personalized Streaming Video Understanding, a streaming video is processed as a continuous sequence of frames. As illustrated in the left part of Fig. 2, throughout the stream, a user can dynamically introduce new concepts at any timestamp via instructions. For a subsequent query, the model must retrieve the relevant concepts and visual context to generate an accurate response. Unlike existing personalized VLM methods, which internalize a pre-defined concept set during training and are evaluated on static images or short offline videos (Nguyen et al.; An et al., 2024; Shi et al., 2025), PSVU requires concepts to be introduced by the user at runtime and maintained throughout the stream.

Based on their temporal and functional requirements, we categorize the queries into three types:

1. **Concept-Definition QA:** Introduces new concepts at specific timestamps, covering Direct and Contextual categories. The model registers the concept into memory based on the current scene for subsequent personalized queries.
2. **Real-Time QA:** Queries established concepts at the immediate moment, covering Presence, Behavior, Appearance, Location, Relation, and Action categories. The model grounds its response on the present scene, evaluating its proficiency in answering real-time questions without historical distraction.
3. **Past-Time QA:** Inquires about the historical states or activities of established concepts, covering Event-based and Time-based categories. The model must retrieve relevant historical sequences, requiring long-term temporal reasoning and precise evidence retrieval.

Detailed descriptions of these categories are provided in the appendix. The task is inherently multi-turn, enabling flexible concept definitions and queries at arbitrary future timesteps. This interactive format lays the foundation for the next generation of personalized AI assistants.

2.2 BENCHMARK OVERVIEW

Existing personalized benchmarks suffer from notable limitations and are largely disconnected from real-world scenarios, as shown in Table 1. MyVLM (Alaluf et al., 2024), Yo'LLaVA (Nguyen et al.), MC-LLaVA (An et al., 2024), UnifyBench (An et al., 2025) and MMPB (Kim et al., 2025) are all image-based, supporting neither video input nor streaming scenarios, and lacking multi-turn interaction. PVChat (Shi et al., 2025) and This-isMy (Yeh et al., 2023) introduce video modality but are limited to short offline videos (each video is shorter than 5 seconds), with no support for streaming

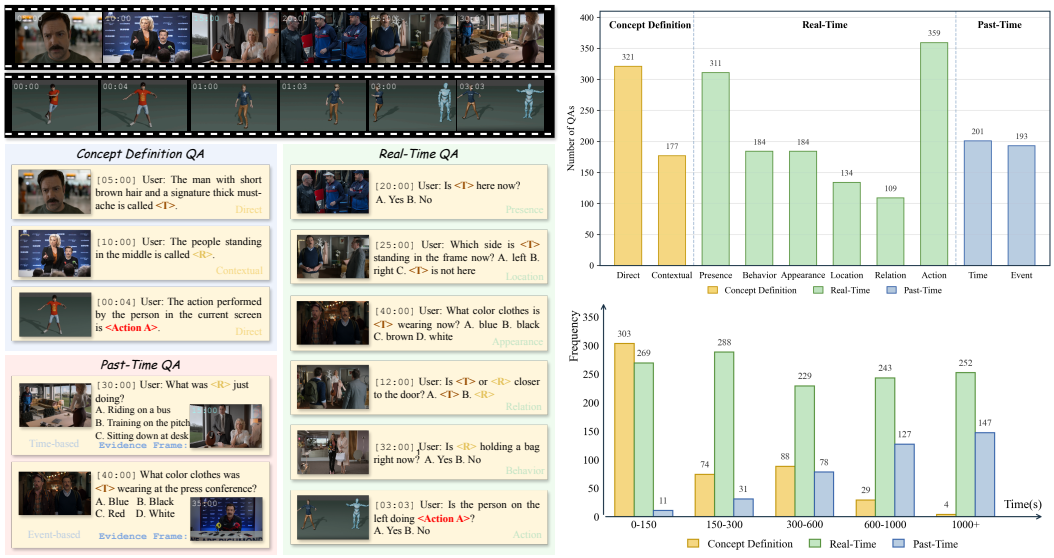


Figure 2: **Overview of PEARL-Bench.** *(Left)* A streaming video timeline with representative examples of Concept-Definition QA, Real-Time QA, and Past-Time QA, where Past-Time QA includes explicit historical evidence frames. *(Right)* Benchmark statistics, including the number of QAs across fine-grained categories and the temporal distribution of queries over the video stream.

or multi-turn concept interaction. Moreover, none of the above benchmarks supports personalized action concepts, i.e., recognizing user-defined concepts grounded in continuous motions across frames. PEARL-Bench is the first benchmark to simultaneously support long-form streaming video input, multi-turn concept interaction, and personalized concepts covering both entities and actions. As shown in the right part of Fig. 2, each QA type in PEARL-Bench contains multiple fine-grained categories. The query timestamps also exhibit a reasonable temporal distribution: Concept-Definition QA is mainly placed in the early stages to define concepts, Real-Time QA is distributed throughout the stream, and Past-Time QA is primarily asked in later stages after sufficient evidence has accumulated.

Table 1: **Comparison of PEARL-Bench with existing personalized benchmarks.**

Benchmark	Modality	Streaming	Multi-turn	Concept Type		Multi-Concept
				Entity	Action	
MyVLM (Alaluf et al., 2024)	Image	–	✗	✓	✗	✗
Yo'LLaVA (Nguyen et al.)	Image	–	✗	✓	✗	✗
MC-LLaVA (An et al., 2024)	Image	–	✗	✓	✗	✓
UnifyBench (An et al., 2025)	Image	–	✗	✓	✗	✗
MMPB (Kim et al., 2025)	Image	–	✗	✓	✗	✓
PVChat (Shi et al., 2025)	Video (short)	✗	✗	✓	✗	✓
This-is-My (Yeh et al., 2023)	Video (short)	✗	✗	✓	✗	✓
PEARL-Bench	Video (long)	✓	✓	✓	✓	✓

2.3 Curation Pipeline

Our curation pipeline consists of three stages: video collection and filtering, annotation of three QA types (Concept-Definition, Real-Time, and Past-Time), and a quality control phase. We employ a diverse set of question templates to annotate the three QA types. Representative examples are illustrated in Fig. 2, and complete templates are provided in the appendix.

2.3.1 Video Collection and Filtering

We collect videos from publicly available internet sources and manually filter them to ensure high dynamics, repeatedly appearing personalized concepts, and resolution no lower than 480p. These videos cover diverse domains including anime, movies, and reality shows. For the Action category, since clean personalized action annotations are difficult to obtain from internet videos, we additionally synthesize action videos using Mixamo (Adobe Systems Inc., 2026) by combining diverse characters, actions, and scenes, where each distinct action serves as a personalized action concept.

2.3.2 QA ANNOTATION

For Concept-Definition QA, annotators locate target concepts in the stream and create registration instructions at timestamps where the concept is visible. These items are not included in final scoring, since their role is to introduce user-defined concepts. To reduce reliance on prior knowledge, original concept names are randomly replaced with common names sampled from the U.S. SSA database (Social Security Administration, 2026). For Real-Time QA, annotators identify suitable query timestamps after concept definition and write concept-related questions whose answers require grounding the defined concept in the current scene. For Past-Time QA, annotators additionally specify historical evidence timestamps, ensuring that each question cannot be answered from the current clip alone and instead requires retrieving and reasoning over earlier stream content. Representative examples are shown in the left part of Fig. 2, with more examples provided in the appendix.

2.3.3 QUALITY CONTROL

To ensure high annotation quality, our curation team consists of 10 researchers with multimodal research experience, including 6 annotators and 4 reviewers. We adopt a combined pipeline of automated filtering and human verification. In the automated stage, we apply an ablation-based filtering method similar to Section 4.3.1: Real-Time QA is tested with and without provided concepts, while Past-Time QA is tested with and without historical evidence clips. Questions that remain answerable when the required information is withheld are deemed trivial and filtered. In the human verification stage, reviewers conduct multiple rounds of manual inspection to ensure that each QA item and timestamp is aligned with the video content. We additionally collect human evaluation scores as an upper-bound reference, reported in Table 2.

3 PEARL FRAMEWORK

To address the challenges of the task of PSVU, we propose a plug-and-play framework, PEARL. As illustrated in Fig. 3, it dynamically defines concepts at specific timestamps of streaming video via user instructions and provides real-time responses to user queries in subsequent timestamps.

In Section 3.1, we present a formal formulation of the task. In Section 3.2, we propose a Dual-grained Memory System to store historical video stream clips and defined concepts. In Section 3.3, we present an efficient Concept-aware Retrieval Algorithm for fast retrieval and response.

3.1 FORMULATION

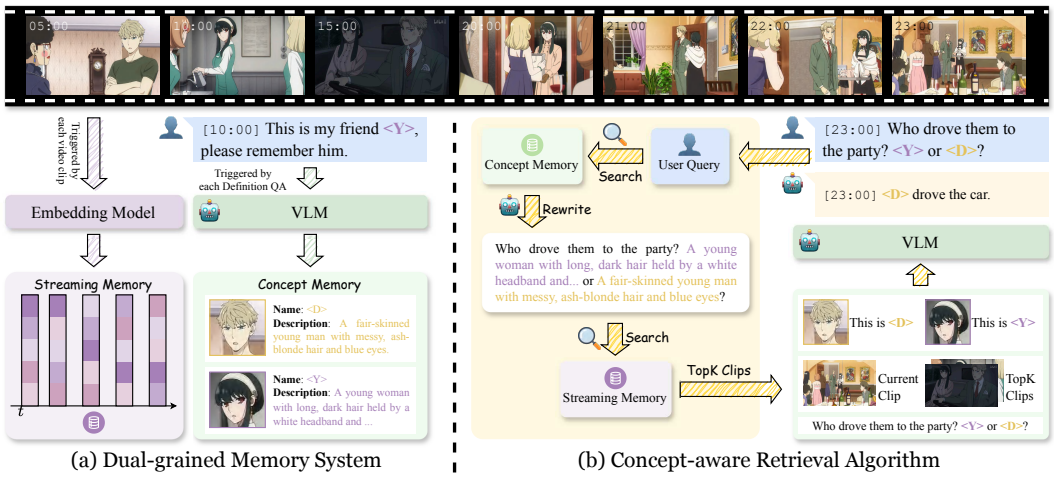
Formally, we define a streaming video as an infinite sequence $V = [\mathcal{X}_1, \mathcal{X}_2, \dots]$, where $\mathcal{X}_i$ denotes a video clip representing a semantic scene. Throughout the stream, a user can dynamically introduce new concepts at any timestamp t_c via instructions, forming an evolving set of defined concepts $\mathcal{C} = \{C_1, C_2, \dots\}$. For a query Q issued at time $t_q \geq t_c$, the model $\mathcal{M}$ must dynamically construct a context to generate a response A:

$$A = \mathcal{M}(\mathcal{C}_{sub}, \mathcal{V}_{context}, Q) \tag{1}$$

where $\mathcal{C}_{sub} \subseteq \mathcal{C}$ is the query-relevant concept subset, and $\mathcal{V}_{context}$ is the necessary visual context. Solving this requires overcoming two key challenges: the prohibitive cost of maintaining unbounded stream history alongside evolving concepts, and the difficulty of accurately retrieving personalized $\mathcal{C}_{sub}$ and $\mathcal{V}_{context}$ in real-time. This motivates our design of a scalable dual-grained memory and a concept-aware retrieval strategy.

3.2 DUAL-GRAINED MEMORY SYSTEM

To support PSVU, the model must (i) retain user-defined concepts introduced at arbitrary timestamps and (ii) maintain access to long-range visual evidence from the evolving video stream for real-time retrieval and response. We therefore design a Dual-grained Memory System that explicitly decouples concept-centric knowledge from stream-centric observations. Concretely, it consists of a Streaming Memory that incrementally archives segmented clips with compact multimodal embeddings for efficient retrieval, and a Concept Memory that stores structured representations of user-defined concepts. We next describe these two memory components in detail.



(a) Dual-grained Memory System (b) Concept-aware Retrieval Algorithm

Figure 3: **PEARL framework. (a) Dual-grained Memory System:** *Concept Memory* stores user-defined concepts with visual evidence and textual descriptions, while *Streaming Memory* archives segmented clips with multimodal embeddings. **(b) Concept-aware Retrieval Algorithm:** Upon a user query, PEARL retrieves relevant concepts and top-K historical clips, then feeds them together with the current clip into a VLM for real-time personalized response.

3.2.1 Streaming Memory

Streaming Memory maintains a set of entries, each consisting of a video clip $\mathcal{X}_i$ and its corresponding embedding $\mathbf{e}_i$. Given a continuously arriving video stream, we first detect scene boundaries and segment the stream into an ordered sequence of clips $V = [\mathcal{X}_1, \mathcal{X}_2, \dots]$. For each newly detected clip $\mathcal{X}_i$, we employ a multimodal embedding model $f_{\text{emb}}(\cdot)$ to compute an embedding $\mathbf{e}_i = f_{\text{emb}}(\mathcal{X}_i)$, and store the pair $(\mathcal{X}_i, \mathbf{e}_i)$ in Streaming Memory. Each clip embedding $\mathbf{e}_i$ captures semantic information about the scene and is used for retrieval. Detailed settings are provided in the appendix.

3.2.2 Concept Memory

When a Concept-Definition query Q_{def} is issued at timestamp t_c, the model creates a new entry with three components: (i) a concept name, (ii) associated visual evidence, and (iii) a textual description. The model first invokes an external tool to extract the visual evidence from the current clip $\mathcal{X}_{t_c}$, using either the most relevant frame or the corresponding short clip depending on the concept-definition instruction. Conditioned on this extracted visual evidence, the model then generates a compact description that summarizes the concept's salient characteristics, using a standardized prompting template provided in the appendix. The resulting entry is finally inserted into the Concept Memory for subsequent retrieval and querying.

3.3 Concept-aware Retrieval Algorithm

When a user issues a Real-Time query or a Past-Time query at timestamp t_q, the model needs three types of information to respond accurately: (i) the query Q, (ii) the query-relevant concept subset $\mathcal{C}_{sub} \subseteq \mathcal{C}$ retrieved from Concept Memory, and (iii) the visual evidence $\mathcal{V}_{context}$ retrieved from Streaming Memory, producing the final answer as $A = \mathcal{M}(\mathcal{C}_{sub}, \mathcal{V}_{context}, Q)$. To obtain $\mathcal{C}_{sub}$, we identify the concept names mentioned in Q and use them as keys to retrieve the corresponding Concept Memory entries. To obtain $\mathcal{V}_{context}$, we use the model $\mathcal{M}$ to rewrite the query into $\tilde{Q}$ by replacing each concept name with its associated description, then encode it using the same multimodal embedding model as Streaming Memory to compute $\mathbf{e}_Q = f_{\text{emb}}(\tilde{Q})$. We compute cosine similarities between $\mathbf{e}_Q$ and all stored clip embeddings $\{\mathbf{e}_i\}_{i \le t_q}$ (where $\mathbf{e}_i = f_{\text{emb}}(\mathcal{X}_i)$), select the top-$K$ most relevant clips, and further expand each selected clip with its adjacent N clips to capture temporally local context, yielding $\mathcal{V}_{context} \subseteq \{\mathcal{X}_1, \dots, \mathcal{X}_{t_q}\}$. Finally, we feed the retrieved concept entries, the retrieved historical clips, the current clip $\mathcal{X}_{t_q}$, and the original query Q into the VLM to generate the response, preserving real-time responsiveness while maximizing retrieval of task-relevant evidence.

Table 2: **Results on PEARL-Bench.** We report fine-grained Real-Time QA and Past-Time QA metrics with their respective averages, where Overall Avg denotes the average of Real-Time QA Avg and Past-Time QA Avg. **Bold** and underline denote the best and second-best results among open-source models and PEARL, respectively.

Method	#Frames	Real-Time QA							Past-Time QA			Overall Avg
		Presence	Behavior	Appearance	Location	Relation	Action	Avg	Event	Time	Avg	
Human Score												
Human	-	94.21	93.48	94.02	93.28	94.50	93.59	93.85	93.26	91.54	92.40	93.13
Text-only												
Qwen3-VL-8B (Bai et al., 2025)	-	12.54	10.87	11.41	9.70	11.01	6.96	10.42	19.17	15.42	17.30	13.86
Proprietary Offline Model												
Gemini3-pro-preview (Google DeepMind, 2025)	64	50.16	46.74	48.91	44.78	45.87	24.51	43.50	51.81	46.27	49.04	46.27
Open-source Offline Model												
LLava-OV-7B (Li et al., 2024)	64	27.01	23.37	25.00	23.13	25.69	10.86	22.51	36.27	31.84	34.06	28.29
Qwen2-VL-7B (Wang et al., 2024)	64	25.08	21.74	22.83	22.39	23.85	17.83	22.29	38.34	33.33	35.84	29.07
InternVL3.5-8B (Wang et al., 2025)	64	32.15	29.35	31.52	28.36	28.44	5.57	25.90	37.31	32.84	35.08	30.49
Qwen3-VL-8B (Bai et al., 2025)	64	28.94	26.09	27.72	26.12	28.44	25.63	27.16	32.64	27.86	30.25	28.71
Open-source Online Model												
ReKV(LLava-OV-7B) (Di et al., 2025)	0.5fps	27.97	25.00	26.63	23.88	27.52	24.23	25.87	39.38	35.32	37.35	31.61
StreamForest-7B (Zeng et al., 2025)	1fps	31.19	28.80	29.89	27.61	28.44	10.86	26.13	43.01	38.31	40.66	33.40
TimeChat-Online-7B (Yao et al., 2025)	1fps	34.08	30.43	32.07	30.60	31.19	22.28	30.11	37.31	33.33	35.32	32.72
PEARL Framework												
LLava-OV-7B+PEARL	1fps	35.05	32.61	34.24	31.34	34.86	20.06	31.36	45.08	40.30	42.69	37.03
Qwen2-VL-7B+PEARL	1fps	35.37	32.07	33.70	31.34	33.94	24.23	31.78	47.15	41.79	44.47	38.13
Qwen3-VL-8B+PEARL	1fps	**56.91**	**53.80**	**56.52**	**52.24**	**55.05**	**48.47**	**53.83**	**52.33**	**46.77**	**49.55**	**51.69**

4 EXPERIMENTS

4.1 IMPLEMENTATION DETAILS

As a plug-and-play framework, we employ Qwen3-VL-Embedding-2B (Li et al., 2026) as the multimodal embedding model, and use LLaVA-OV-7B (Li et al., 2024), Qwen2-VL-7B (Wang et al., 2024), and Qwen3-VL-8B (Bai et al., 2025) as base models, respectively. For the Dual-grained Memory System, we use the corresponding base model to generate concept descriptions and rewrite user queries. We use PySceneDetect (Castellano, 2026) to detect scene boundaries and segment the streaming video into clips. For the Concept-aware Retrieval Algorithm, we set $K = 4$, $N = 1$, and sample the input video stream at 1 FPS.

As a pioneering effort in this area, where no directly comparable baselines exist, we instead adopt representative offline and online video understanding methods as our baselines. For offline models, we adopt different sampling protocols according to the queried concept type: for entity-related questions, we uniformly sample 64 frames, and for action-related questions, we use a 64-second window and sample at 1 FPS to ensure the model observes continuous motion patterns. For online models, given the unique paradigm of the PSVU task, which necessitates concept registration prior to question answering, we exclusively select models capable of multi-turn dialogue as our baselines. Furthermore, we strictly follow their original sampling configurations. All experiments are conducted on NVIDIA H200 GPUs. Note that to minimize the model's bias toward answer options and prevent random guessing (Pezeshkpour & Hruschka, 2024; Atabuzzaman et al., 2025), we evaluate each question using a cyclic option rotation strategy, with detailed settings provided in the appendix.

4.2 MAIN RESULTS

We report Human Score and a Text-only baseline as upper and lower bounds in Table 2. The large gap between them shows that PEARL-Bench is highly solvable with sufficient visual information, but cannot be reliably answered from text priors alone and requires grounding in the streaming video.

4.2.1 COMPARISON WITH OFFLINE BASELINES

We evaluate several representative open-source offline models, along with a strong proprietary offline baseline. As shown in Table 2, these offline approaches struggle on the PSVU task. In contrast, applying PEARL to the same base models yields consistent and substantial improvements. On Real-Time QA, LLaVA-OV-7B+PEARL, Qwen2-VL-7B+PEARL, and Qwen3-VL-8B+PEARL improve their offline counterparts by 8.85%, 9.49%, and 26.67%, respectively. On Past-Time QA, the corresponding gains are 8.63%, 8.63%, and 19.30%. These results demonstrate the generality and robustness of PEARL across distinct architectures. The performance gap reveals a key limitation of

offline models in the streaming setting: to satisfy low-latency inference, they operate with restricted visual context and typically lack explicit memory mechanisms to preserve and retrieve long-range historical evidence. Consequently, the visual information needed to answer many queries is often missing, leading to degraded accuracy.

4.2.2 Comparison with Online Baselines

We compare PEARL against three representative open-source online models: ReKV (LLaVA-OV-7B Di et al., 2025), StreamForest-7B (Zeng et al., 2025), and TimeChat-Online-7B (Yao et al., 2025). As shown in Table 2, PEARL consistently outperforms these online baselines on both Real-Time QA and Past-Time QA. Compared with the strongest online baseline, Qwen3-VL-8B+PEARL achieves gains of 23.72% on Real-Time QA and 8.89% on Past-Time QA. Notably, LLaVA-OV-7B+PEARL comprehensively outperforms ReKV on both metrics, achieving gains of 5.49% and 5.34%, respectively. Since ReKV shares the same backbone and is likewise a plug-and-play framework, this controlled comparison strongly indicates that the performance gains stem from the PEARL framework design itself rather than differences in backbone capability. We attribute these advantages to PEARL's Dual-grained Memory System and Concept-aware Retrieval Algorithm: whereas traditional online models compress historical information into a fixed-size state and lack concept-grounded retrieval, PEARL explicitly stores user-defined concept representations and precisely retrieves query-relevant historical clips, enabling more accurate personalized responses in the streaming setting.

4.2.3 Analysis by QA Type

The fine-grained results in Table 2 further show that PSVU requires both immediate concept grounding and historical evidence retrieval. Real-Time QA evaluates whether models can recognize user-defined concepts in the current scene across diverse categories such as presence, behavior, appearance, location, relation, and action. Among these categories, Action is especially challenging because it requires recognizing a personalized motion pattern over continuous frames rather than grounding a static entity in a single frame. Past-Time QA further requires retrieving historical evidence before answering, covering event-based questions that ask about previous occurrences or states of personalized concepts and time-based questions that require locating when a relevant historical moment happened. The consistent improvements on all these categories demonstrate that PEARL's dual-grained memory system and concept-aware retrieval strategy provide the relevant context needed for diverse scenarios in PSVU task.

4.3 Ablation Study

4.3.1 Effectiveness of PEARL Design

We ablate PEARL by progressively enabling its components in Table 3. Using Qwen3-VL-8B on PEARL-Bench, text-only performance starts near random chance, and adding the current clip provides only a marginal improvement. In contrast, adding Concept Memory with concept retrieval leads to a dramatic gain of over 35% in Real-Time accuracy, showing that explicit concept grounding is crucial. Subsequently, incorporating Streaming Memory with retrieval yields a substantial jump of more than 20% in Past-Time accuracy, indicating the need for historical evidence. Finally, adding Query Rewriting (full PEARL) further improves both Real-Time and Past-Time performance, elevating the average accuracy by an additional 3.73% over the non-rewritten version.

These results suggest three takeaways. (i) Concept Memory is indispensable: without concept-specific information, the model cannot reliably link user-defined names to personalized entities, so the benchmark is hard to solve. (ii) Streaming Memory is essential for Past-Time QA, which depends on retrieving and reasoning over historical clips rather than the current scene. (iii) Query Rewriting turns personalized names into descriptive semantics that embedding models can match more effectively, improving evidence retrieval and final answers.

4.3.2 Efficiency of PEARL Design

We further evaluate end-to-end inference efficiency. As shown in Table 4, although PEARL introduces a minor latency overhead compared to the respective base models, both LLaVA-OV-7B+PEARL and Qwen3-VL-8B+PEARL achieve substantial improvements in Overall Avg, with gains of 8.74%

Table 3: **Ablation of PEARL components** on PEARL-Bench with Qwen3-VL-8B, showing progressive gains from Text-only to the full PEARL pipeline.

Text	Current	Concept	Streaming	Rewrite	Real-Time	Past-Time	Avg
✓	✗	✗	✗	✗	10.42	17.30	13.86
✓	✓	✗	✗	✗	15.84	20.30	18.07
✓	✓	✓	✗	✗	51.41	25.43	38.42
✓	✓	✓	✓	✗	50.22	45.69	47.96
✓	✓	✓	✓	✓	**53.83**	**49.55**	**51.69**

Table 4: Comparison of end-to-end inference latency across models. Overall Avg denotes the average of Real-Time QA Avg and Past-Time QA Avg.

Method	#Frames	Overall Avg	Latency (ms)
LLaVA-OV-7B	64	28.29	670
Qwen3-VL-8B	64	28.71	1,594
ReKV (LLaVA-OV-7B)	0.5fps	31.61	1,818
StreamForest-7B	1fps	33.40	1,164
TimeChat-Online-7B	1fps	32.72	4,769
LLaVA-OV-7B+PEARL	1fps	37.03	775
Qwen3-VL-8B+PEARL	1fps	51.69	2,111

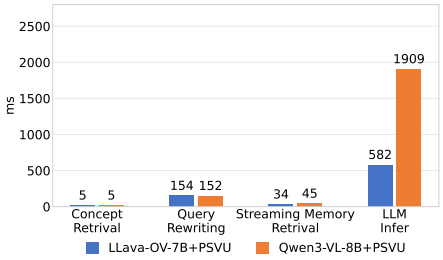


Figure 4: End-to-end latency breakdown of PEARL across different models.

and 22.98%, respectively. Moreover, LLaVA-OV-7B+PEARL not only maintains higher accuracy than all online baselines but also remains faster than all of them. Although Qwen3-VL-8B+PEARL has a slightly higher latency, its accuracy substantially surpasses the strongest online baseline, StreamForest-7B, leading by 18.29%.

Furthermore, as shown in Fig. 4, the end-to-end latency of our PEARL framework breaks down into Concept Retrieval, Query Rewriting, Streaming Memory Retrieval, and LLM Inference. Notably, the latency introduced by PEARL's core modules (retrieval and rewriting) is exceptionally low and invariant across different models, and the underlying LLM inference constitutes the primary latency bottleneck. This demonstrates that PEARL can seamlessly adapt to diverse model architectures while maintaining real-time retrieval capabilities.

4.3.3 Generalization on Non-personalized Tasks

To verify whether our framework maintains its effectiveness on general tasks, we evaluate the base models with and without PEARL on the non-personalized streaming video understanding benchmark, OVO-Bench (Niu et al., 2025). As shown in Table 5, integrating PEARL not only preserves the general streaming video understanding capabilities of the base models but even brings slight performance improvements. This demonstrates that PEARL remains effective on non-personalized streaming video understanding tasks, enhancing personalized capabilities without compromising its proficiency.

Table 5: Comparison of base models with and without PEARL on OVO-Bench.

Model	Real-Time	Backward	Avg
LLava-OV-7B	64.02	43.71	53.87
LLava-OV-7B+PEARL	65.34	47.60	56.47
Qwen2-VL-7B	55.98	46.46	51.22
Qwen2-VL-7B+PEARL	58.02	49.51	53.77

5 Limitations

While PEARL provides a plug-and-play baseline for PSVU, several limitations remain. First, PEARL depends on the quality of concept registration and historical clip retrieval, where inaccurate descriptions or missed evidence may propagate errors to the final answers. Second, PEARL-Bench contains a relatively smaller number of Past-Time QA examples, as constructing them requires locating repeated appearances of the same concept and establishing traceable temporal connections, making annotation more challenging. Finally, personalized streaming video understanding may raise privacy and misuse concerns in real-world AI assistant deployments, especially when user-specific identities or behaviors are stored over time. Future work may improve concept registration and retrieval, expand Past-Time QA annotations, and explore privacy-preserving memory mechanisms.

REPRODUCIBILITY STATEMENT

We provide anonymized code and example data as supplementary material to facilitate reproduction of our results. The complete implementation of PEARL, including the Dual-grained Memory System (Section 3.2), the Concept-aware Retrieval Algorithm (Section 3.3), the multi-GPU serving and inference pipeline, and the evaluation script that realizes the cyclic option rotation strategy, is included in `code.zip`, together with the pinned environment specification, the server launch scripts, and the prompt templates used for concept description generation and query rewriting (Appendix D). A sample input for the inference service is provided in `example_data.zip`, which contains one annotated streaming video. All hyperparameters required to reproduce the reported numbers are stated in Section 4.1 and Appendix F. The collection, annotation, and quality-control procedures behind PEARL-Bench are described in Section 2, with the complete question templates in Appendix C and the fine-grained category definitions in Appendix B. Full benchmark resources, evaluation code, and data construction scripts will be publicly released under the licenses described in Appendix G.

AI USE DISCLOSURE

The research ideas, task formulation, method design, and experimental analysis presented in this paper were conceived and carried out by the authors. Large language models (LLMs) were used as assistive tools in the three aspects described below. We have not used them for proposing or refining hypotheses, designing or evaluating research methodology or experiments, developing theoretical models or conceptual frameworks, formulating mathematical claims, or interpreting results, and the remaining tasks requiring disclosure (translation, assistance in writing proofs, and support for qualitative or thematic data analysis) are not applicable to this work; we did not use LLMs to create or modify scientific figures, nor to retrieve or discover literature. All LLM outputs were reviewed, verified, and edited by the authors, and the authors take full responsibility for the content of this paper. LLMs are not listed as authors.

Writing assistance. All sections of this paper were drafted by the authors. LLMs were then used to polish the already-written text, including grammar correction, word choice, and fluency, without generating the substantive content of any section. LLMs were also used to format references.

Benchmark construction. LLMs participated in the construction of PEARL-Bench, following the curation pipeline described in Section 2. For Real-Time and Past-Time QA, human annotators first determined the query timestamps, the questions, and the correct answers; the corresponding video clips and answers were then fed to a strong VLM to generate the multiple-choice distractors. The generated options were not adopted as-is: annotators and reviewers inspected every item, aligned it with the video content, and revised or discarded it as part of the quality-control procedure in Section 2, so no annotation entered the final benchmark without human verification.

Code implementation. LLMs assisted in implementing the codebase, including the functions and modules of the Dual-grained Memory System and the Concept-aware Retrieval Algorithm. In these cases, the authors first specified the algorithm design and the intended behavior of each component, and then directed the LLM to write, complete, and debug the corresponding implementation. All algorithms, system designs, and experimental protocols are the authors' own; no algorithmic design originated from an LLM.

REFERENCES

Adobe Systems Inc. Mixamo: Adobe Animated 3D Characters and Motion Capture Library. `https://www.mixamo.com/`, 2026. Accessed: 2026-03-03.

Saif Ahmed and Norzalita Abd Aziz. Impact of ai on customer experience in video streaming services: a focus on personalization and trust. *International Journal of Human–Computer Interaction*, 41(12):7726–7745, 2025.

Yuval Alaluf, Elad Richardson, Sergey Tulyakov, Kfir Aberman, and Daniel Cohen-Or. Myvlm: Personalizing vlms for user-specific queries. In *European Conference on Computer Vision*, pp. 73–91. Springer, 2024.

Ruichuan An, Sihan Yang, Ming Lu, Renrui Zhang, Kai Zeng, Yulin Luo, Jiajun Cao, Hao Liang, Ying Chen, Qi She, et al. Mc-llava: Multi-concept personalized vision-language model. *arXiv preprint arXiv:2411.11706*, 2024.

Ruichuan An, Sihan Yang, Renrui Zhang, Zijun Shen, Ming Lu, Gaole Dai, Hao Liang, Ziyu Guo, Shilin Yan, Yulin Luo, et al. Unictokens: Boosting personalized understanding and generation via unified concept tokens. *arXiv preprint arXiv:2505.14671*, 2025.

Md Atabuzzaman, Ali Asgarov, and Chris Thomas. Benchmarking and mitigating mcqa selection bias of large vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33536–33550, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Brandon Castellano. Pyscenedetect. `https://www.scenedetect.com`, 2026. URL `https://github.com/Breakthrough/PySceneDetect`. Video Cut Detection and Analysis Tool.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large language model for streaming video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18407–18418, 2024.

Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *European conference on computer vision*, pp. 558–577. Springer, 2022.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Tao Zhong, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. Streaming video question-answering with in-context video kv-cache retrieval. *arXiv preprint arXiv:2503.00540*, 2025.

Junyu Feng, Binxiao Xu, Jiayi Chen, Mengyu Dai, Cenyang Wu, Haodong Li, Bohan Zeng, Yunliu Xie, Hao Liang, Ming Lu, et al. M2a: Multimodal memory agent with dual-layer hybrid memory for long-term personalized interactions. *arXiv preprint arXiv:2602.07624*, 2026.

Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, et al. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv preprint arXiv:2501.01957*, 2025.

Norina Gasteiger, Mehdi Hellou, and Ho Seok Ahn. Factors for personalization and localization to optimize human–robot interaction: A literature review. *International Journal of Social Robotics*, 15(4):689–701, 2023.

Google DeepMind. A new era of intelligence with Gemini 3. `https://blog.google/products-and-platforms/products/gemini/gemini-3/`, 2025.

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*, pp. 14538–14548, 2025.

Mengze Hong, Chen Jason Zhang, Chaotao Chen, Rongzhong Lian, and Di Jiang. Dialogue language model with large-scale persona data engineering. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pp. 961–970, 2025.

Jaeik Kim, Woojin Kim, Woohyeon Park, and Jaeyoung Do. Mmpb: It's time for multi-modal personalization. *arXiv preprint arXiv:2509.22820*, 2025.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, et al. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026.

Junming Lin, Zheng Fang, Chi Chen, Zihao Wan, Fuwen Luo, Peng Li, Yang Liu, and Maosong Sun. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. *arXiv preprint arXiv:2411.03628*, 2024.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, et al. Nvila: Efficient frontier visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4122–4134, 2025.

Thao Nguyen, Haotian Liu, Yuheng Li, Mu Cai, Utkarsh Ojha, and Yong Jae Lee. Yo'llava: Your personalized language and vision assistant, 2024. *URL https://arxiv. org/abs/2406*, 9400.

Thao Nguyen, Krishna Kumar Singh, Jing Shi, Trung Bui, Yong Jae Lee, and Yuheng Li. Yo'chameleon: Personalized vision and language generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 14438–14448, 2025.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, et al. Ovo-bench: How far is your video-llms from real-world online video understanding? In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18902–18913, 2025.

Yeongtak Oh, Dohyun Chung, Juhyeon Shin, Sangha Park, Johan Barthelemy, Jisoo Mok, and Sungroh Yoon. Repic: Reinforced post-training for personalizing multi-modal language models. *arXiv preprint arXiv:2506.18369*, 2025.

Yeongtak Oh, Sangwon Yu, Junsung Park, Han Cheol Moon, Jisoo Mok, and Sungroh Yoon. Contextualized visual personalization in vision-language models. *arXiv preprint arXiv:2602.03454*, 2026.

OpenAI. Gpt-4v(ision) system card. `https://openai.com/contributions/gpt-4v/`, 2023.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2006–2017, 2024.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 24045–24055, 2025.

Yufei Shi, Weilong Yan, Gang Xu, Yumeng Li, Yucheng Chen, Zhenxi Li, Fei Yu, Ming Li, and Si Yong Yeo. Pvchat: Personalized video chat with one-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23321–23331, 2025.

Social Security Administration. Popular Baby Names. `https://www.ssa.gov/oact/babynames/`, 2026. Accessed: 2026-03-03.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Junda Wu, Hanjia Lyu, Yu Xia, Zhehao Zhang, Joe Barrow, Ishita Kumar, Mehrnoosh Mirtaheri, Hongjie Chen, Ryan A Rossi, Franck Dernoncourt, et al. Personalized multimodal large language models: A survey. *arXiv preprint arXiv:2412.02142*, 2024.

Binxiao Xu, Junyu Feng, Shaolin Lu, Yulin Luo, Shilin Yan, Hao Liang, Ming Lu, and Wentao Zhang. Jarvis: Towards personalized ai assistant via personal kv-cache retrieval. *arXiv preprint arXiv:2510.22765*, 2025.

Shuhang Xun, Sicheng Tao, Jungang Li, Yibo Shi, Zhixin Lin, Zhanhui Zhu, Yibo Yan, Hanqian Li, Linghao Zhang, Shikang Wang, et al. Rtv-bench: Benchmarking mllm continuous perception, understanding and reasoning through real-time video. *arXiv preprint arXiv:2505.02064*, 2025.

Sihan Yang, Huitong Ji, Shaolin Lu, Jiayi Chen, Binxiao Xu, Ming Lu, Yuanxing Zhang, Wenhui Dong, and Wentao Zhang. Small-large collaboration: Training-efficient concept personalization for large vlm using a meta personalized small vlm. *arXiv preprint arXiv:2508.07260*, 2025a.

Zhenyu Yang, Yuhang Hu, Zemin Du, Dizhan Xue, Shengsheng Qian, Jiahong Wu, Fan Yang, Weiming Dong, and Changsheng Xu. Svbench: A benchmark with temporal multi-turn dialogues for streaming video understanding. *arXiv preprint arXiv:2502.10810*, 2025b.

Linli Yao, Yicheng Li, Yuancheng Wei, Lei Li, Shuhuai Ren, Yuanxin Liu, Kun Ouyang, Lean Wang, Shicheng Li, Sida Li, et al. Timechat-online: 80% visual tokens are naturally redundant in streaming videos. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10807–10816, 2025.

Chun-Hsiao Yeh, Bryan Russell, Josef Sivic, Fabian Caba Heilbron, and Simon Jenni. Meta-personalizing vision-language models to find named instances in video. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 19123–19132, 2023.

Tianyu Yu, Zefan Wang, Chongyi Wang, Fuwei Huang, Wenshuo Ma, Zhihui He, Tianchi Cai, Weize Chen, Yuxiang Huang, Yuanqian Zhao, et al. Minicpm-v 4.5: Cooking efficient mllms via architecture, data, and training recipe. *arXiv preprint arXiv:2509.18154*, 2025.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, et al. Streamforest: Efficient online video understanding with persistent event memory. *arXiv preprint arXiv:2509.24871*, 2025.

Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024a.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024b.

Yu Zhong, Tianwei Lin, Ruike Zhu, Yuqian Yuan, Haoyu Zheng, Liang Liang, Wenqiao Zhang, Feifei Shao, Haoyuan Li, Wanggui He, et al. Unified personalized understanding, generating and editing. *arXiv preprint arXiv:2601.06965*, 2026.

A Related Works

Personalized VLMs. As the capabilities of Vision-Language Models (VLMs) continue to advance (Wang et al., 2025; Bai et al., 2025; Li et al., 2024; Yu et al., 2025; Google DeepMind, 2025; Zhang et al., 2024a;b; OpenAI, 2023; Liu et al., 2025), growing attention has been directed toward unleashing their potential to serve as personalized AI assistants (Hong et al., 2025; Cohen et al., 2022; Wu et al., 2024; Oh et al., 2026). Existing VLM personalization efforts can be broadly categorized into three areas: personalized image understanding, combined with personalized generation and personalized video understanding. Previous research has predominantly focused on personalized image understanding. The paradigm can be summarized as finetune-based (Alaluf et al., 2024; Nguyen et al.; An et al., 2024; Yang et al., 2025a), Retrieval-Augmented Generation (RAG-based, Hao et al., 2025; Xu et al., 2025) and reinforcement learning (Oh et al., 2025; Feng et al., 2026). However, they are inherently limited to static images and fail to generalize to dynamic video domains. Parallel studies have also explored methods that unify personalized understanding and generation (Nguyen et al., 2025; An et al., 2025; Zhong et al., 2026). Yet, these approaches heavily rely on pre-defined concepts, which contradicts the flexible nature of real-world user interactions. In the domain of personalized video understanding, early explorations (Yeh et al., 2023) were mostly restricted to personalized retrieval. While a recent work, PVChat (Shi et al., 2025) pioneers to focus on personalized VQA but it is strictly designed for offline scenarios. Meanwhile, the emerging field of streaming video understanding (Di et al., 2025; Yao et al., 2025; Zeng et al., 2025; Niu et al., 2025; Yang et al., 2025b; Xun et al., 2025; Lin et al., 2024; Chen et al., 2024; Qian et al., 2025; Fu et al., 2025) has made significant strides in processing continuous visual inputs for real-time interactions, yet these methods remain agnostic to user-defined concepts. Consequently, existing approaches fall short of meeting the combined demands for real-time responding, streaming inputs, and flexible concept definition. To address these limitations, this paper introduces the novel task of Personalized Streaming Video Understanding (PSVU) for the first time. Furthermore, we propose PEARL, a plug-and-play framework designed to achieve highly efficient concept registration and real-time inference within video streams.

B QA Sub-categories

To thoroughly evaluate model capabilities under diverse scenarios, we further categorize the queries in PEARL-Bench. We systematically classify the questions within Concept-Definition QA, Real-Time QA, and Past-Time QA into fine-grained sub-categories based on their reasoning requirements.

Concept-Definition QA. We classify the concept definitions into two distinct types:

- **Direct:** The concept is defined straightforwardly without relying on complex descriptions. For entity concepts, it points out the main character or object in the scene (e.g., *"This character is called {ConceptName}. Please remember this name."*). For action concepts, it identifies a specific personalized motion pattern over a continuous clip (e.g., *"The sequence of movements shown in this clip is defined as {ConceptName}. Please remember this name."*)

- **Contextual:** The concept is defined by describing its explicit visual attributes (like clothing color) or its interactions and relationships with other objects in the scene. (e.g., *"The character wearing white clothes is named {ConceptName}. Please remember this name."*)

Real-Time QA. To evaluate multi-dimensional perception, we categorize the real-time queries into six distinct sub-tasks:

- **Presence:** Asking whether the concept is present in the current scene. (e.g., *"Is {ConceptName} here now?"*)
- **Behavior:** Querying the current action or state of the defined concept. (e.g., *"What is {ConceptName} doing now?"*)
- **Appearance:** Focusing on the transient visual details of the concept, such as current clothing. (e.g., *"What color is {ConceptName} wearing now?"*)
- **Location:** Identifying the spatial positioning of the concept within the scene. (e.g., *"Where is {ConceptName} located in this scene now?"*)
- **Relation:** Inquiring about the interaction or relationship between the target concept and other entities or objects. (e.g., *"Who is standing next to {ConceptName} now?"*)
- **Action:** Querying whether a dynamically defined action concept is being performed across the current continuous clip. (e.g., *"Is the person doing {ConceptName} now?"*)

Past-Time QA. We divide the past-time queries into two types based on the retrieval mechanism required to answer them:

- **Event-based:** The historical evidence can be localized purely based on a specific event or action, without requiring strict temporal reasoning. (e.g., *"What was {ConceptName} holding when he was cooking?"*)
- **Time-based:** The historical evidence requires the model to understand the temporal sequence or order of events to accurately retrieve the correct clip. (e.g., *"What did {ConceptName} do right before he left the room?"*)

C Complete Question Templates

As mentioned in the main text, we provide the complete question templates used in PEARL-Bench for annotating Concept-Definition QA, Real-Time QA, and Past-Time QA. In Table 6, we summarize the representative question templates corresponding to each fine-grained sub-category. It is worth noting that these question templates serve primarily as a reference for the annotators. In practice, annotators often diversify the content, phrasing, and sentence structures of the questions based on specific video scenarios, thereby ensuring the richness and naturalness of the data. To represent the varied entities and properties systematically, we utilize the following placeholders in the templates:

- `{ConceptName}`: The dynamically assigned user-defined name for the target concept.
- `{OtherConcept}`: The name of another previously defined concept.
- `{EntityPronoun}`: General references to the subject, such as *person, character, man, girl, object*, etc.
- `{ConceptFeature}`: Specific visual characteristics or attributes used to describe the concept, such as *short hair, red clothes, sitting by window*, etc.
- `{Object}`: Variables pertaining to the question, representing objects interacted with or queried.
- `{Location}`: A specific location or scene setting referenced in the queries.
- `{Event}`: A specific action or event referenced in the queries.
- `{DefinitionSuffix}`: A standardized suffix appended to concept definitions to enforce model compliance. For entity concepts, it is: *"From now on, regardless of who this character might be in real life or in any media, you must refer to them as {ConceptName}."* For action concepts, it is: *"From now on, regardless of how this action might be typically described or what similar movements exist, whenever you see this specific pattern of motion, you must refer to it as {ConceptName}."*

Table 6: Representative question templates across different QA types and sub-categories in PEARL-Bench.

QA Type	Sub-category	Question Templates
Concept-Definition	Direct	• This {EntityPronoun} is called {ConceptName}. Please remember this name. {DefinitionSuffix} • This {EntityPronoun} is my friend, named {ConceptName}. Please remember this name. {DefinitionSuffix} • The sequence of movements shown in this clip is defined as {ConceptName}. Please remember this name. {DefinitionSuffix}
	Contextual	• The {EntityPronoun} in the {Location} is named {ConceptName}. Please remember this name. {DefinitionSuffix} • The {EntityPronoun} with {ConceptFeature} in the scene is my friend, named {ConceptName}. Please remember this name. {DefinitionSuffix} • The {EntityPronoun} who is {Event} is called {ConceptName}. Please remember this name. {DefinitionSuffix}
Real-Time	Presence	• Is this {EntityPronoun} {ConceptName} at this moment? • Are {ConceptName} and {OtherConcept} here now? • Is {ConceptName} currently among this group of people?
	Behavior	• What is {ConceptName} doing now? • Is {ConceptName} {Event} right now? • What is {ConceptName}'s expression now?
	Appearance	• What color is {ConceptName} wearing now? • What is the shape of the {Object} {ConceptName} is wearing? • What color is {ConceptName}'s {Object} now?
	Location	• Where is {ConceptName} located in the frame? • Is {ConceptName} on the {Location} in the picture? • Where is {ConceptName} now?
	Relation	• Who is standing next to {ConceptName} now? • What is {ConceptName} giving to {OtherConcept} right now? • What is {ConceptName} holding right now?
	Action	• Is the person doing {ConceptName} now? • What is the person doing? Is it {ConceptName}?
Past-Time	Event-based	• Did {ConceptName} just walk into the {Location}? • What color clothes was {ConceptName} wearing when {Event}? • What did {ConceptName} just put into the {Object}?
	Time-based	• Is {ConceptName}'s hair color the same as it was just now? • What was {ConceptName} holding before {EntityPronoun} tried to {Event}? • What was {ConceptName} doing before {OtherConcept} {Event}?

D Prompt Templates

D.1 Concept Description Generation Prompts

We provide standardized prompting templates used to generate a compact description that summarizes the concept's salient characteristics. Since PEARL-Bench evaluates both entity and action concepts, we design two distinct prompts for their respective characteristics:

Entity Concept Prompt. This prompt is designed to guide the model to focus on permanent and stable visual features (such as gender, facial features, hair, and body type) while instructing it to ignore temporary elements like clothing, accessories, or poses, which are likely to change across a long video stream.

Action Concept Prompt. Conversely, this prompt directs the model to focus on the core kinematics and stable movement patterns of a customized action, while explicitly ignoring the specific identity or appearance of the person performing it, as well as the background and surrounding environment, ensuring the extracted action features are generalizable across different characters and scenes.

Specifically, both templates include two key placeholders:

- `{concept_name}`: The user-defined name assigned to the concept.

- `{original_description}`: The user instruction from the Concept-Definition QA. For entity concepts, it helps the model locate the target subject (e.g., "The character wearing white clothes is named {Adaliz}."). For action concepts, it helps the model identify the specific action (e.g., "The sequence of movements shown in this clip is {Action A}.").

The complete prompts for both concept types are provided below:

Entity Concept Description Generation Prompt

```
Based on the image and the original description provided, generate a
concise visual description of this character/object that focuses on
PERMANENT/STABLE features for video clip retrieval.

Original description: "{original_description}"
Concept name: {concept_name}

Your task:
1. Use the original description to understand WHICH
character/object to focus on in the image
2. Generate a description focusing on STABLE features that DON'T
change throughout the video:
   - Gender (male/female/other)
   - Face features (eye shape, facial structure, distinctive marks)
   - Hair (color, length, style if distinctive)
   - Body type (build)
   - Age appearance (young/middle-aged/elderly)

AVOID or minimize:
- Clothing details (they change in long videos)
- Accessories (they may be removed)
- Temporary expressions or poses
- Background, location, surroundings, or nearby objects in the scene
- Relative position or size compared to objects/environment in the
scene

Requirements:
- Keep it concise and simple (1 sentence, around 10-15 words)
- Focus on features that remain consistent across different scenes
- Write in English using simple descriptive terms
- Use third person (e.g., "a young male with...", "the girl
with...")
- Make it natural enough to replace the concept name in a question

Please provide the distinctive visual description focusing on
PERMANENT features:
```

Action Concept Description Generation Prompt

```
Based on the provided video clip and the original description,
generate a concise textual description of the specific ACTION or
MOVEMENT PATTERN that focuses on the CORE KINEMATICS for video clip
retrieval.

Original description: "{original_description}"
Concept name: {concept_name}

Your task:
1. Use the original description to understand WHICH specific action
or sequence of movements to focus on in the video clip
2. Generate a description focusing on the STABLE MOVEMENT PATTERNS
that define this action, regardless of who is performing it:
   - Core body movements (e.g., raising arms, squatting, twisting)
   - Sequence of motions (the order of the gestures)
   - Body parts involved (hands, legs, torso)
AVOID or minimize:
- The specific identity, gender, age, or appearance of the person
```

```
performing the action
- Background, location, surroundings, or irrelevant objects in the
scene
- Any static features that do not contribute to the dynamic action
itself

Requirements:
- Keep it concise and simple (1 sentence, around 10-20 words)
- Focus strictly on the dynamic movement pattern that can be
performed by different characters
- Write in English using simple descriptive action terms
- Use general action phrases (e.g., "the action of swinging arms in
a circle", "the action of squatting down and then leaping forward")
- Make it natural enough to replace the concept name in a question

Please provide the distinctive action description focusing on CORE
MOVEMENT PATTERNS:
```

D.2 Query Rewrite Prompt

As discussed in our Concept-aware Retrieval Algorithm, we use a prompt to rewrite user queries by replacing concept names with their descriptions to improve retrieval accuracy. This process helps translate user-defined concept names (which the multimodal embedding model has not seen) into explicit visual or kinematic semantics that facilitate accurate historical clip retrieval.

Specifically, the template includes two key placeholders:

- `{query}`: The original user question containing the customized concept names.
- `{replacement_instructions}`: A set of automatically constructed rules mapping each concept name found in the query to its generated description (e.g., *"{Adaliz}" should be replaced with "a young female with long black hair"* for an entity concept, or *"{Action A}" should be replaced with "the action of squatting down and then leaping forward"* for an action concept).

The complete prompt is provided below:

Query Rewrite Prompt

```
Rewrite the following question by replacing the concept names (in
curly braces) with their visual descriptions.  Keep the sentence
grammatically correct and natural.

Original question:
{query}

Replacement rules:
{replacement_instructions}

Requirements:
- Replace each {ConceptName} with the provided visual description
- Adjust grammar as needed (e.g., articles, verb forms) to keep the
sentence natural
- Do NOT change the meaning of the question
- Do NOT add or remove any information
- Output ONLY the rewritten question, nothing else
```

E More Visualization Examples

To better illustrate the complexity and diversity of our dataset, as well as the effectiveness of our proposed framework, we present additional visualization examples from PEARL-Bench. These examples demonstrate the model's capability to handle continuous video streams, dynamically register personalized concepts, and accurately answer both real-time and past-time queries.

As shown in Fig. 5, we showcase a comprehensive multi-turn interaction process using the Qwen3-VL-8B+PEARL model on a long animated video. As the video stream progresses, the user dynamically defines multiple concepts at different timestamps (e.g., <Nuriya> at [00:44], <Kavery> at [04:57], and <Truz> at [05:05]). Upon receiving these Concept-Definition instructions, the PEARL framework successfully invokes the Register_Concept tool. This tool specifically functions to extract the current visual evidence (i.e., the current frame), generate a concise visual description for the target concept, and subsequently update the Concept Memory by storing the visual evidence, the generated description, and the concept name as a unified entity.

Subsequently, the model demonstrates robust real-time perception by accurately answering Real-Time QA based on the current scene. For example, at [01:53], the model not only successfully recognizes <Nuriya>'s presence but also accurately describes her combat posture. Later, at [18:39], it correctly identifies <Kavery> on the left side of the screen. Furthermore, the model exhibits strong long-term temporal reasoning in Past-Time QA. For instance, at [08:01], when asked who saved <Truz>, the model successfully retrieves the historical evidence from [06:48] to provide the correct answer. Similarly, at [13:42], it accurately recalls the terrifying snake encounter that occurred at [11:04]. These results highlight PEARL's ability to maintain and retrieve personalized memories effectively.

We also provide an additional example for the Action category in Fig. 6. The model registers several user-defined action sequences (e.g., <Action A>, <Action B>, and <Action C>) and recognizes them when they reappear later in the stream, showing that PEARL can also support personalized motion concepts.



Figure 5: Visualization example of multi-turn interactions in PEARL-Bench. The model successfully registers multiple user-defined concepts (e.g., `<Nuriya>`, `<Kavery>`, `<Truz>`) and accurately answers subsequent Real-Time and Past-Time queries by retrieving corresponding historical evidence.

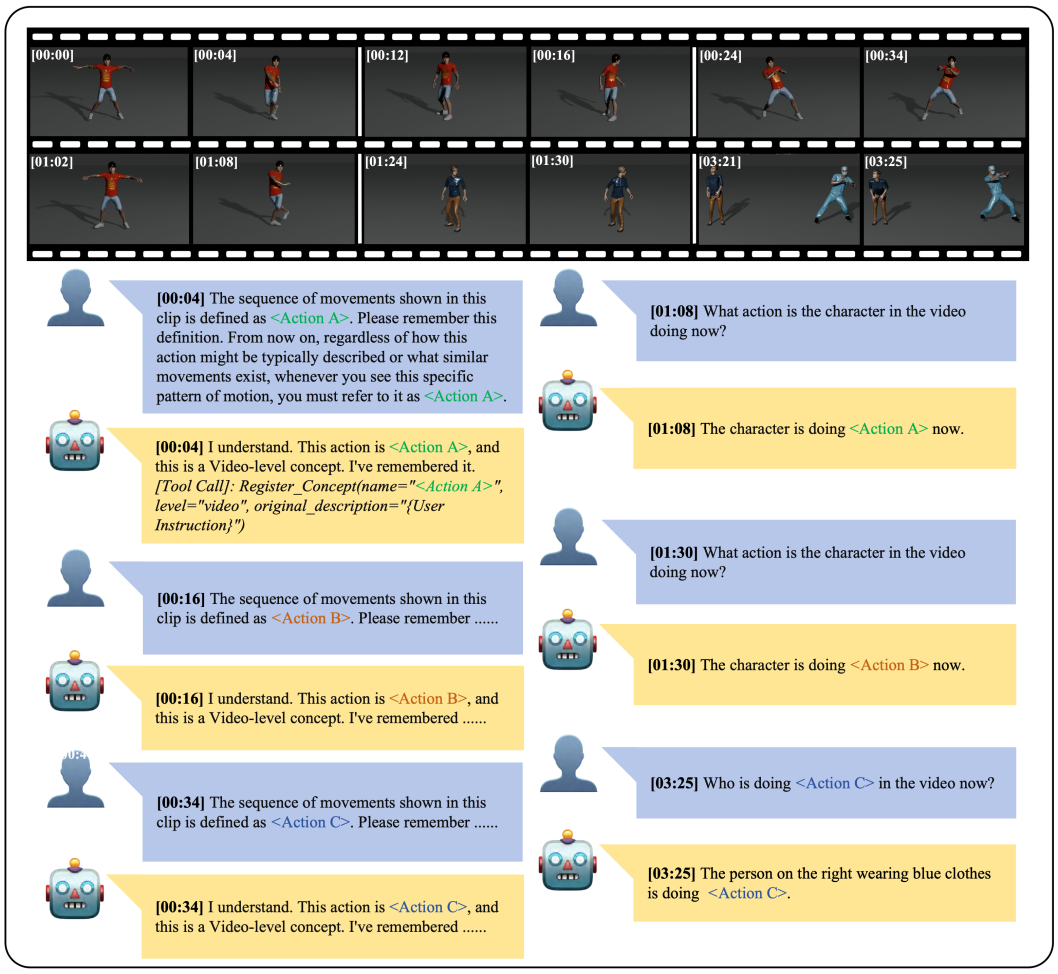


Figure 6: Additional visualization example for the Action category. The model registers multiple user-defined action sequences (e.g., `<Action A>`, `<Action B>`, `<Action C>`) and recognizes these customized actions when they are performed by different characters later in the video stream.

F Detailed Implementation Settings

F.1 Multimodal Embedding Model and Scene Detection

We provide additional details regarding the multimodal embedding model and scene detection configurations used in our Dual-grained Memory System.

Multimodal Embedding Model. We adopt Qwen3-VL-Embedding-2B (Li et al., 2026) as our multimodal embedding model. This model encodes both the generated visual descriptions and the streaming video clips into a unified feature space, enabling efficient and accurate historical evidence retrieval via cosine similarity computation. Video clips are sampled at 1 FPS for embedding extraction.

Scene Detection. We employ PySceneDetect (Castellano, 2026) to segment the continuous streaming video into semantically coherent clips by detecting fast cuts based on pixel changes in the HSV colorspace between adjacent frames. We set the scene detection threshold to 27.0. To ensure the segmented clips contain sufficient temporal context while avoiding excessively long segments that might dilute the semantic focus, we enforce a minimum clip duration of 1.0 second and a maximum clip duration of 8.0 seconds. Any detected scenes that exceed the maximum duration are proportionally split into multiple smaller segments.

F.2 Cyclic Option Rotation Evaluation Strategy

We employ a cyclic option rotation strategy during evaluation. Specifically, for each multiple-choice question, we iteratively evaluate the model four times, each time rotating the correct answer to a different option position (A, B, C, and D). During each rotation, we swap the contents of the originally correct option with the target option, leaving the other distractors unchanged, which minimizes perturbation to the overall option distribution. The model is considered to have answered the question correctly only if it consistently selects the correct option across all four rotated variations (i.e., 4/4 correct). This strict evaluation criterion ensures that the model's performance truly reflects its multimodal reasoning capabilities rather than lucky guesses or positional biases.

F.3 Compute Resources

All main experiments are conducted on 8 NVIDIA H200 GPUs and can be completed in approximately 20 minutes. Each GPU uses approximately 80GB of memory. Since PEARL leaves all model parameters frozen, our experiments do not require additional model training compute beyond benchmark evaluation and inference, making the overall computational cost primarily determined by VLM forward passes.

G Data Release and License

Upon acceptance, we will publicly release PEARL-Bench resources, evaluation code, and data construction scripts for research use. The code and scripts will be released under the MIT License, while the annotations, metadata, and rendered synthetic videos created by us will be released under CC BY-NC 4.0. For public internet videos, we will not redistribute raw video files, and will instead provide source identifiers or URLs so users can obtain the videos from their original sources under the corresponding licenses and platform terms. For the synthetic Action split, we will release rendered videos and annotations but not raw Mixamo (Adobe Systems Inc., 2026) characters, animation files, or standalone assets. We will also provide a dataset card covering sources, intended use, license notes, privacy considerations, and a removal request procedure.

H Additional Experimental Results

H.1 Hyperparameter Analysis for Past-Time QA

We analyze two key hyperparameters on Past-Time QA: the number of top-K retrieved clips (K) and the number of adjacent clips to expand per retrieved clip (N). As shown in Fig. 7, when $K = 0$, no historical clips are retrieved, so the model cannot leverage historical video evidence and the metric remains low. As K increases, accuracy improves rapidly and plateaus after $K \geq 3$, indicating that moderate retrieval suffices to cover the visual evidence needed for answering. Regarding N, larger adjacent expansion provides richer local temporal context, but the gap between $N = 1$ and $N = 2$ is small, suggesting that one adjacent clip already captures most of the useful information. Balancing performance and efficiency, we adopt $K = 4$ and $N = 1$ as our default configuration.

H.2 Hyperparameter Analysis for Real-Time QA

Real-Time QA can be accurately addressed without relying heavily on historical clips, making it inherently less sensitive to the parameters K and N. We present the corresponding experimental results in Fig. 8. We observe two key findings: (1) Overall, the impact of K and N on the results is not significant, with accuracy fluctuations remaining within a tight 5% range. (2) Compared to the baseline of $K = 0$ (where no historical clips are retrieved), retrieving a small amount of historical clips generally yields slightly better performance. This suggests that historical clips related to the query can provide supplementary context that modestly improves Real-Time QA accuracy. However, as K and N increase further (introducing more historical clips), the accuracy exhibits a downward trend. This decline indicates that an excessive amount of historical information introduces irrelevant noise, which ultimately interferes with the model's judgment on real-time queries.

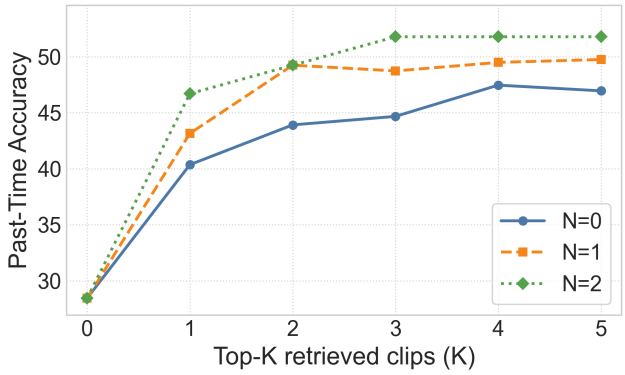


Figure 7: Past-Time accuracy under different top-K (K) and expansion sizes (N).

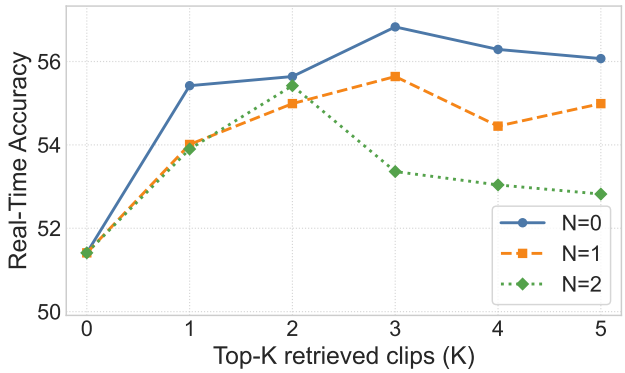


Figure 8: Real-Time accuracy under different top-K (K) and expansion sizes (N).

H.3 Effect of Model Scales

We investigate the performance of PEARL-Bench across different model scales to reveal how parameter size influences personalized streaming video understanding capabilities. Specifically, we evaluate the Qwen2-VL series (2B and 7B Wang et al., 2024) and the Qwen3-VL series (4B and 8B Bai et al., 2025) with and without our PEARL framework on PEARL-Bench. We do not conduct experiments on models with larger scales, as the PSVU task requires real-time responsiveness in practical applications. The experimental results are summarized in Table 7.

Table 7: Performance comparison across different model scales on PEARL-Bench. **Bold** and underline denote the best and second-best results within each model family (Qwen2-VL and Qwen3-VL), respectively.

Model	Real-Time	Past-Time	Avg
Qwen2-VL-2B (Wang et al., 2024)	<u>31.24</u>	22.84	27.04
Qwen2-VL-7B (Wang et al., 2024)	22.29	<u>35.84</u>	29.07
Qwen2-VL-2B+PEARL	29.93 ↓1.31	32.49 ↑9.65	<u>31.21</u> ↑4.17
Qwen2-VL-7B+PEARL	**31.78** ↑9.49	**44.47** ↑8.63	**38.13** ↑9.06
Qwen3-VL-4B (Bai et al., 2025)	24.08	<u>30.96</u>	27.52
Qwen3-VL-8B (Bai et al., 2025)	27.16	30.25	28.71
Qwen3-VL-4B+PEARL	<u>40.78</u> ↑16.70	**50.25** ↑19.29	<u>45.52</u> ↑18.00
Qwen3-VL-8B+PEARL	**53.83** ↑26.67	49.55 ↑19.30	**51.69** ↑22.98

Based on the results, we draw two key conclusions:

1. Robustness across model scales. Our method consistently yields substantial performance improvements across all sizes and architectures. For instance, the average accuracy of the Qwen3-VL

4B and 8B models is boosted by 18.00% and 22.98%, respectively, with similar trends observed in the Qwen2-VL series (e.g., boosting the 2B and 7B models by 4.17% and 9.06%, respectively). This demonstrates the robustness of the PEARL design, proving its effectiveness regardless of the underlying model capacity or architecture.

2. Paradigm mismatch for offline models. When evaluating the standard offline baselines, increasing the model scale (which generally correlates with stronger comprehension capabilities) does not lead to significant performance gains. This highlights that the traditional offline paradigm is fundamentally ill-suited for the PSVU task, as a model's inherent reasoning capability cannot compensate for the lack of visual context. It is only when integrated with a framework specifically designed for PSVU, like PEARL, that the benefits of scaling up the model size are successfully unleashed, with the larger models ultimately outperforming the smaller models by a significant margin.