Towards Online Interactors: A Comprehensive Survey on Streaming Video Understanding

Zhenyu Yang, Kairui Zhang, Qi Liu, Tiancheng Liu, Long Ying, Dizhan Xue, Qibin Hou, *Member, IEEE,* Shengsheng Qian, *Member, IEEE,* and Changsheng Xu, *Fellow, IEEE*

***Abstract*—Conventional Video Large Language Models (Video-LLMs) operate under an offline paradigm: they ingest a complete video before producing any response. This post-hoc design fundamentally limits their ability to serve as real-time interactors that perceive, reason, and communicate while the video is still unfolding. Streaming video understanding breaks this constraint by requiring models to process frames as they arrive and generate temporally grounded responses on the fly, thereby transforming passive narrators into active, always-on interactors. In this survey, we present a comprehensive and structured overview of recent advances in streaming video understanding, with a particular focus on Video-LLM-based approaches. We categorize existing methods into two complementary paradigms: *proactive streaming models*, which autonomously determine *when* to initiate responses in dynamic visual environments, and *reactive streaming models*, which focus on *how* to maintain long-term context and deliver efficient inference upon user queries over unbounded video streams. Together, these two paradigms characterize the core capabilities required for online video interactors: temporal awareness for timely engagement and scalable memory for sustained reasoning. We further provide a systematic review of emerging *benchmarks and datasets* designed to evaluate streaming-specific capabilities, including multi-turn dialogue, real-time captioning, and proactive response timing. Finally, we identify the key open challenges and outline promising future directions toward building truly interactive, efficient, and scalable streaming video systems. An accompanying resource repository is maintained at https://github.com/sotayang/Awesome-Streaming-Video-Understanding.**

***Index Terms*—Streaming Video Understanding, Video Large Language Models, Real-Time Inference, Long-Context Modeling**

I. Introduction

VIDEO has become one of the most dominant carriers of information, driven by online platforms [1], ubiquitous sensing devices [2], and autonomous systems [3]. This abundance has propelled rapid progress in video understanding, recently accelerated by Multimodal Large Language Models (MLLMs) [4]–[9] that span closed-source systems such as GPT-4o [7] and open-weight counterparts such as Qwen2.5-VL [8], with notable gains in fine-grained spatio-temporal reasoning [10]–[12] and long-context comprehension [13]–[17].

Despite these advances, mainstream Video Large Language Models (Video-LLMs) [18]–[22] remain *offline*: they assume a complete, pre-recorded clip and must buffer it in full before responding. This pattern is incompatible with continuously arriving frames and incurs latency that grows with video length, a bottleneck most acute in the applications now driving demand, including intelligent surveillance, autonomous driving [23], and embodied intelligence [24]. These settings instead call for *streaming* video understanding, in which a model perceives frames as they arrive and responds under real-time constraints [25]. This paradigm shift from offline to online video understanding gives rise to three core challenges [26]. **(1) Temporal sensitivity.** Unlike the offline setting where a single static answer suffices, streaming requires responses that evolve as new visual evidence accrues, so that both the content and the timing of an output become part of the prediction. **(2) Long-context modeling.** Tasks such as summarization and planning demand reasoning over extended histories, yet caching all visual tokens in the key-value (KV) cache of the underlying LLM incurs memory that grows linearly and attention cost that grows quadratically with stream length, which is untenable for unbounded streams. **(3) Real-time interaction.** The system must sustain an inference throughput at or above the input frame rate while supporting interactive behaviors such as interruption handling and non-wake-up dialogue [27]. Critically, these challenges are not independent but mutually constraining: timely and frequent responses, faithful retention of long histories, and bounded per-frame latency cannot be maximized simultaneously under a fixed computational budget. This tension, rather than any single difficulty in isolation, distinguishes streaming from offline video understanding and motivates the two complementary paradigms examined in this survey. Recent Video-LLMs [28]–[32] have begun to address these challenges, as their strong cross-modal reasoning underpins progress in spatiotemporal perception [33]–[35], memory-augmented compression [36]–[38], and long-term video understanding [39]–[41]. Yet most of this progress remains anchored in the offline paradigm, feeding complete videos into the model for multi-turn textual interaction. Only recently has research moved beyond interleaved multimodal dialogue [42], [43] toward genuinely streaming settings [44]–[46] that target continuous video processing and online response generation.

Fig. 1 provides a unified view of the evolution of streaming video understanding, integrating proactive and reactive paradigms with representative methods and benchmarks. Building on the above challenges, existing approaches fall into two complementary interaction paradigms: **Proactive Streaming Models** and **Reactive Streaming Models**. Proactive streaming models [26], [47], [48] endow systems with autonomous, anticipatory capabilities: rather than passively waiting for input, they actively decide when and whether to initiate interactions, such as issuing early hazard warnings, narrating

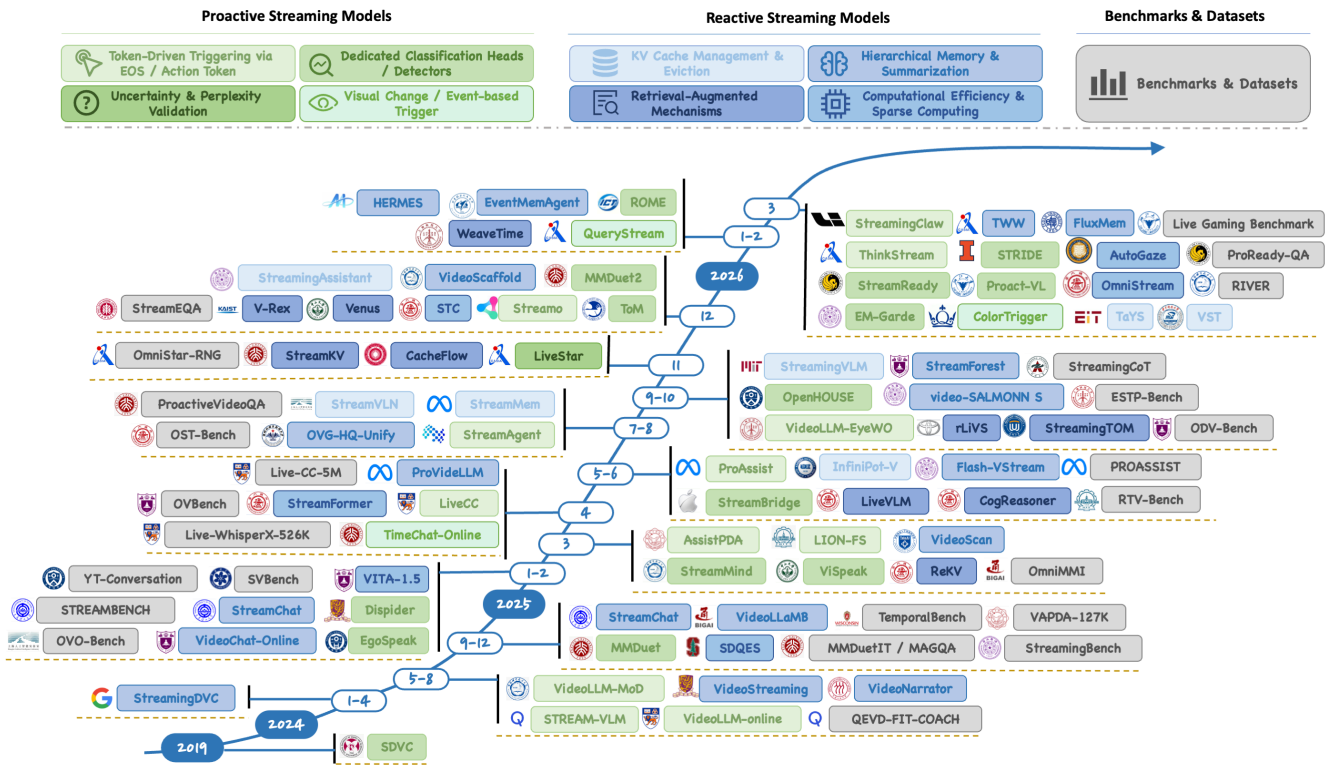

Fig. 1. Timeline of existing research on streaming video understanding, categorized into three key dimensions: Proactive Streaming Models, Reactive Streaming Models, and Benchmarks & Datasets.

salient events in real time, or requesting clarification when input is ambiguous. Their key challenge is decision-making under uncertainty, namely designing reliable and generalizable triggering mechanisms that identify the "just-in-time" moment for interaction, avoiding both premature responses based on incomplete observations and delayed reactions that miss critical events. In contrast, reactive streaming models [49]–[51] follow a query-driven paradigm: they continuously encode the incoming stream into a persistent memory and remain idle until explicit user queries arrive (e.g., "What just happened?" or "Is there any anomaly?"). Upon receiving a query, the model retrieves and aggregates relevant information from a potentially long temporal context to produce coherent responses. The central challenge here is scalability and efficiency: maintaining informative representations of long histories while avoiding prohibitive memory growth and latency, particularly in resource-constrained environments.

Furthermore, recent efforts have introduced a growing number of benchmarks [52]–[54] to evaluate streaming video understanding [55], [56], moving beyond offline evaluation to capture the dynamic, interactive nature of streaming scenarios. These benchmarks span multi-turn dialogue and question answering [57]–[62], real-time captioning and narration, and proactive response and timing prediction. Reactive settings assess a model's ability to maintain coherent multi-turn interactions, perform long-context reasoning, and generate temporally grounded responses over continuously observed streams, whereas proactive evaluation emphasizes a more challenging dimension: whether a model can autonomously determine when to respond, requiring precise temporal awareness and decision-making under uncertainty. Despite these advances, current benchmarks remain fragmented, often addressing dialogue coherence, temporal reasoning, or response timing in isolation, and a unified framework jointly considering these dimensions remains an open challenge.

To position our contribution against existing survey literature, Table I compares the scope of closely related surveys. Existing works approach video streams from fundamentally different angles: a body of surveys treats *video streaming* from a computation- and network-driven, systems-level perspective, where "streaming" denotes content delivery, transcoding, compression, and caching rather than semantic understanding [63], [64]; others primarily target either *general video understanding with LLMs* [3], *video-language understanding* [65], or *video temporal grounding with MLLMs* [54]. In contrast, this survey focuses on *streaming* video understanding under causal and real-time constraints, covering online interaction and organizing the field through a unified proactive–reactive triggering taxonomy together with a benchmark-oriented analysis.

To fill this gap and provide a structured overview of recent advances, we present, to the best of our knowledge, the **first comprehensive survey dedicated to streaming video understanding** based on multimodal large language models, offering a unified perspective on this rapidly developing area and a reference for future research. The main contributions are summarized as follows:

- **A Systematic Review of Streaming Tasks:** According to the task scenarios addressed, we summarize existing tasks

TABLE I
POSITIONING THIS SURVEY WITH RESPECT TO CLOSELY RELATED SURVEYS. ✓, △, AND ✗ DENOTE SYSTEMATIC COVERAGE, PARTIAL COVERAGE, AND NO EXPLICIT COVERAGE, RESPECTIVELY.

Survey	Year	Primary Focus	Streaming Setting	Online Interaction	Triggering Taxonomy	Streaming Benchmarks
Laghari *et al.* [64]	2023	Video streaming technologies (compression/protocols)	△	✗	✗	✗
Dao *et al.* [63]	2022	Computation-driven live video streaming (systems/delivery)	△	✗	✗	✗
Tang *et al.* [3]	2025	LLM-based video understanding (general survey)	△	△	✗	✗
Nguyen *et al.* [65]	2024	Video-language understanding (architecture/training/data)	✗	✗	✗	△
Wu *et al.* [54]	2026	MLLM-based video temporal grounding	✗	✗	✗	✗
This survey	**2026**	**Streaming video understanding with Video-LLMs**	✓	✓	✓	✓

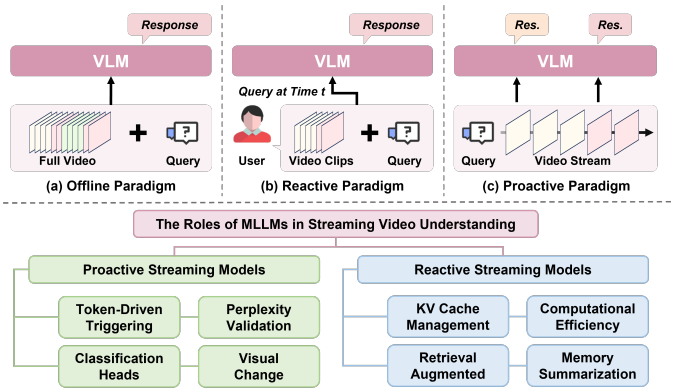

Fig. 2. Paradigms and technical taxonomy of streaming video understanding. (Top) Comparison of (a) Offline Paradigm with full video access, (b) Reactive Paradigm triggered by external queries, and (c) Proactive Paradigm featuring autonomous interaction. (Bottom) Taxonomy of MLLM roles categorized into proactive response triggering and reactive memory / efficiency management.

into real-time perception and event/action understanding, real-time description and narrative generation, agent-based interaction and task planning, and video dialogue.

- **A Holistic Taxonomy of Streaming Video Understanding:** We establish a formal classification that explicitly distinguishes reactive and proactive streaming paradigms, highlighting their unique technical challenges and application scenarios, which frames the subsequent technical discussion.
- **A Panoramic Summary of Benchmarks:** We categorize existing benchmarks and datasets into a clear taxonomy covering multi-turn dialogue & QA, real-time captioning & narration, and proactive response & timing evaluation, providing guidance for researchers.
- **Challenges and Future Directions:** We identify key challenges, including temporal modeling, efficiency, and response triggering, and outline promising directions toward more adaptive, efficient, and predictive streaming systems.

II. PRELIMINARIES

In this section, we formalize streaming video understanding and introduce its fundamental interaction paradigms.

A. Streaming Video Understanding

Streaming video understanding refers to a setting in which a model processes video frames in a strictly causal manner: at any time step, it can access only past and present observations, with no visibility into future frames. This requirement fundamentally distinguishes it from conventional offline video understanding, where the entire video is available prior to inference. Under this causal constraint, the model must incrementally update its representation of the evolving scene and produce temporally consistent interpretations or responses while operating within strict latency and memory budgets.

Formally, a streaming video is represented as a (potentially unbounded) sequence of frames:

$$V = \{f_1, f_2, \cdots, f_t, \cdots\}, \tag{1}$$

where f_t denotes the frame observed at time step t. At each time step t, the model only has access to the prefix of the stream:

$$V_{\leq t} = \{f_1, f_2, \cdots, f_t\}. \tag{2}$$

The model is thus required to generate outputs based solely on the available observations:

$$y_t \sim p(\cdot \mid V_{\leq t}), \tag{3}$$

where y_t can correspond to textual descriptions, question answering results, event notifications, or other forms of interactive responses. This formulation provides a unified probabilistic view of streaming prediction and serves as the foundation for modeling various streaming tasks discussed in the following sections.

As illustrated in Fig. 2 (top), we contrast the conventional offline paradigm with two fundamental streaming paradigms: reactive and proactive. These differ primarily in their triggering mechanisms: whether responses are query-driven or autonomous. This distinction leads to diverse technical requirements for memory management and decision-making, which we categorize into a holistic taxonomy in Fig. 2 (bottom).

1) Proactive Response Paradigm: The proactive paradigm represents a general formulation of streaming video interaction, where the model not only processes continuous visual streams $V = \{f_1, \cdots, f_t\}$ but also learns to decide when to generate responses. At each time step t, a response triggering variable is introduced:

$$a^t \in \{0, 1\}, \tag{4}$$

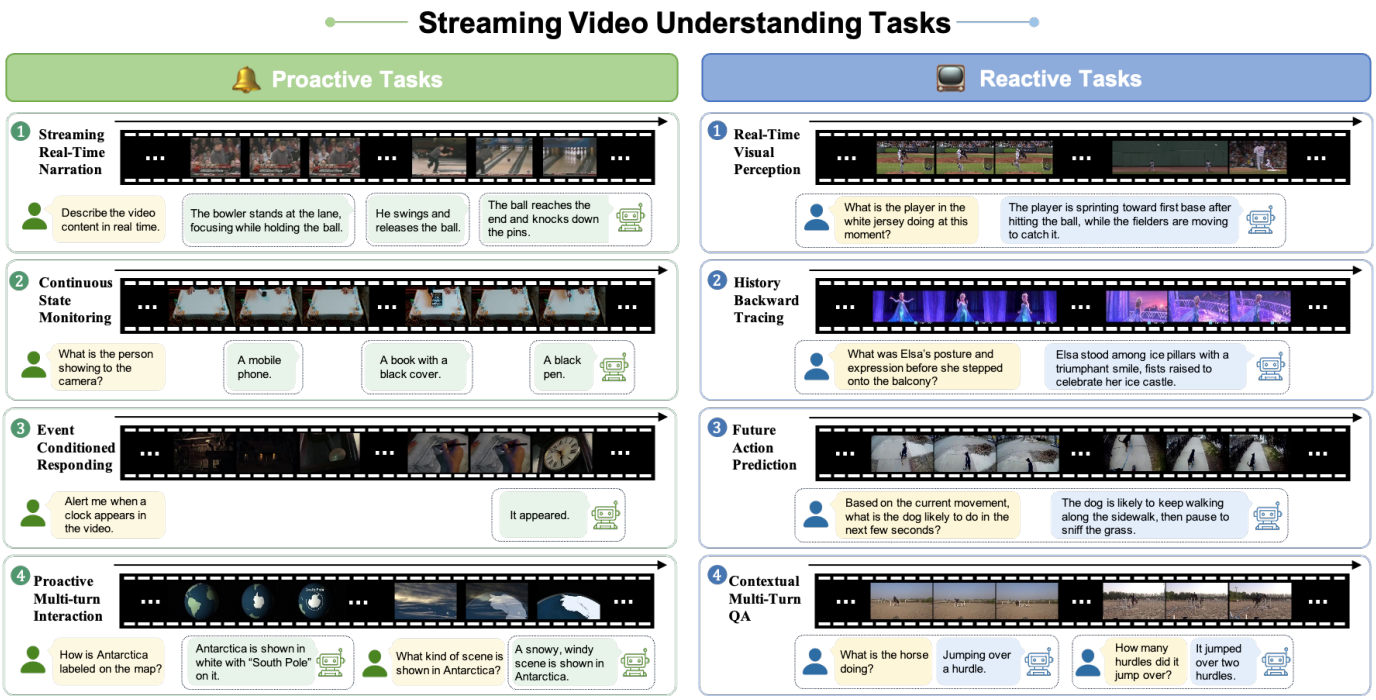

Fig. 3. Taxonomy and examples of streaming video understanding tasks, divided by interaction logic into Proactive Streaming Tasks (left) and Reactive Streaming Tasks (right). Proactive tasks follow a "one-query, multiple-updates" pattern, where the model decides when to respond after observing sufficient evidence, whereas reactive tasks follow a "one-query, one-answer" pattern requiring an immediate response when a query appears at a given timestamp. Yellow triangles mark query or trigger points in the stream.

where $a^t = 1$ indicates that the model autonomously generates a response based on the current visual context. The triggering process is modeled as:

$$P(a^t \mid V_{\leq t}), \tag{5}$$

and when $a^t = 1$, the model produces an output:

$$P(y^t \mid V_{\leq t}). \tag{6}$$

This formulation enables fully autonomous interaction in streaming environments without requiring explicit user intervention.

2) Reactive Response Paradigm: The reactive paradigm can be viewed as a constrained special case of the proactive setting, where response generation is explicitly triggered by external user queries. At time t, given a query Q^t, the model produces a response conditioned on both the query and the observed video history:

$$P(y^t \mid V_{\leq t}, Q^t). \tag{7}$$

In this case, the response triggering mechanism is externalized, i.e., a^t is implicitly determined by user input rather than learned by the model. As a result, the system focuses on query-driven reasoning rather than autonomous decision-making.

B. Streaming Video Understanding Tasks

As illustrated in Fig. 3, streaming video understanding tasks are organized according to their interaction logic. In accordance with our formulation, we first introduce proactive tasks, followed by reactive tasks, highlighting their differences in response triggering and interaction dependency.

1) Proactive Streaming Video Understanding Tasks: Governed by an autonomous triggering mechanism, proactive tasks require the model to continuously monitor the stream $V_{\leq t}$ and decide *when* and *what* to output. This category emphasizes the coupling between perception, temporal decision-making, and information value.

- *Streaming Real-Time Narration:* Requires the model to act as a live commentator, autonomously generating continuous and non-redundant descriptions as new semantic events unfold, emphasizing narrative timing and fluency.
- *Continuous State Monitoring:* Implements a "one-query, multiple-updates" paradigm. Given a standing task (e.g., "count the people"), the model must proactively update its response whenever the state of the target variable changes in the stream.
- *Event Conditioned Responding:* Focuses on evidence-driven output. The model must monitor the stream for specific conditions or hidden evidence and only initiate a response (such as an alert or a deferred answer) at the exact moment the trigger appears.
- *Proactive Multi-turn Interaction:* Simulates a human-like assistant in a duplex conversation. The model must autonomously manage interaction flow, including judging user input validity, interrupting redundant content, and proactively initiating new turns.

2) Reactive Streaming Video Understanding Tasks: In contrast, Reactive tasks are explicitly triggered by user-initiated queries Q^t. The model is required to perform a single-point analysis over the observed video history $V_{\leq t}$, focusing on semantic accuracy and depth of understanding without the

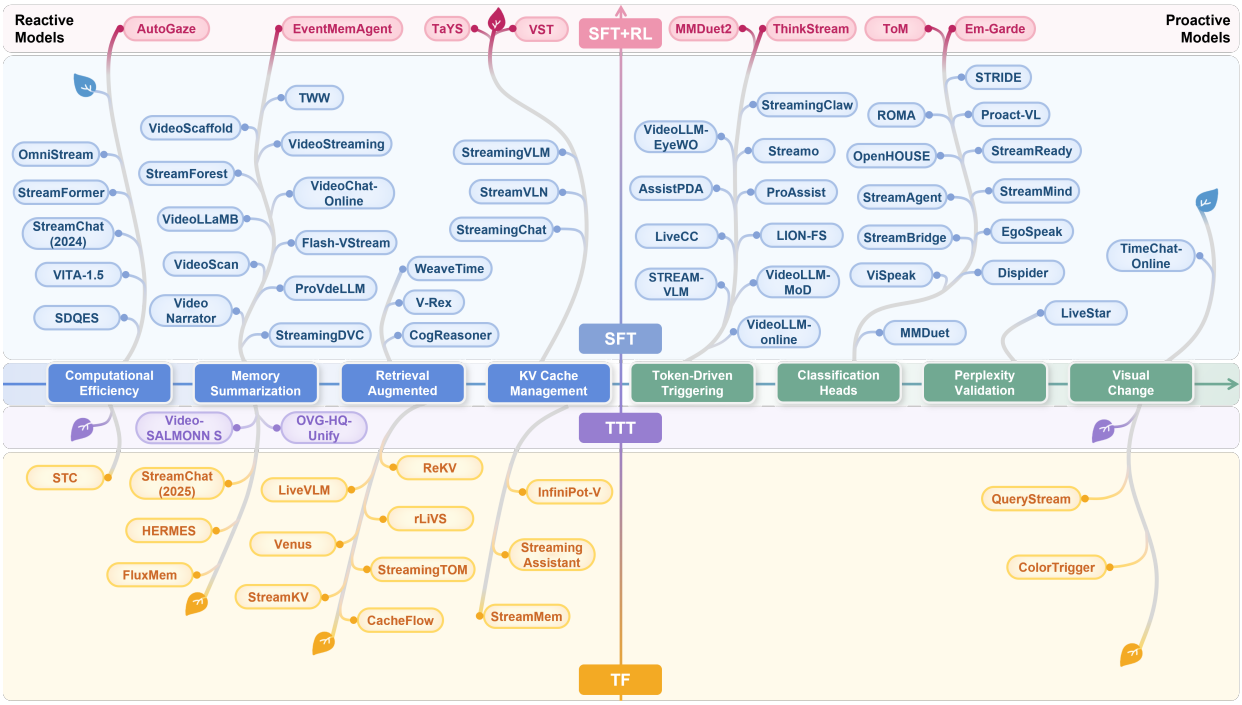

Fig. 4. A comprehensive landscape of streaming video understanding models, organized along interaction paradigms (x-axis), separating reactive (left, blue) and proactive (right, green) models with further divisions by core technical approach, and training strategies (y-axis), covering training-free (TF), test-time training (TTT), supervised fine-tuning (SFT), and reinforcement learning (RL).

need for autonomous timing decisions.

- *Real-Time Visual Perception:* Evaluates the model's ability to capture the "present" state. It asks about ongoing actions or immediate attributes at the current timestamp, testing the efficiency of real-time visual feature extraction.
- *History Backward Tracing:* Requires the model to retrieve specific details or event sequences from the "past". It assesses the robustness of long-term episodic memory and the ability to locate historical clues within the accumulated video stream.
- *Future Action Prediction:* Focuses on the "near-future" by requiring the model to anticipate the next steps or potential outcomes based on current trends and causal reasoning, moving beyond simple recognition to anticipatory inference.
- *Contextual Multi-Turn QA:* Evaluates the consistency of the "dialogue history". The model must resolve coreferences (e.g., pronouns like "it" or "he") and maintain logical coherence across sequential questions that depend on previous interactions.

III. Proactive Streaming Models

Existing proactive streaming models aim to equip systems with **autonomous response triggering capability**, where the model decides *when* to produce outputs based on the evolving video stream, rather than passively waiting for external queries. In this setting, the model continuously monitors incoming observations and selectively activates responses only when sufficient or salient evidence is detected. This selective activation mechanism resembles human attention, where cognitive resources are allocated to critical events rather than uniformly processing all incoming information.

As shown in Fig. 5, we provide a systematic overview of representative approaches to proactive streaming video understanding. These methods mainly differ in how they determine when to trigger a response. Based on this perspective, existing works can be broadly categorized into four technical classes: (1) **Token-Driven Triggering via EOS / Action Token**, where models implicitly make decisions through the generation of special action tokens; (2) **Dedicated Classification Heads / Detectors**, which employ lightweight auxiliary modules to explicitly determine whether to trigger the main model; (3) **Uncertainty / Perplexity Validation**, which monitors the model's confidence in existing outputs to decide whether updates are necessary; (4) **Visual Change / Event-based Trigger**, which directly leverages low-level or high-level changes in the video stream as triggering signals. These four classes form a spectrum along which the triggering decision is progressively decoupled from the generative backbone: from being embedded in autoregressive decoding (1), to being delegated to dedicated modules (2), and finally to being derived from cheap output- or input-level signals (3)–(4). Moving along this spectrum tends to lower per-frame computational cost and reliance on the heavyweight model, but shifts the burden onto the design of reliable external triggers. The following sections analyze the technical principle, representative works, advantages, and limitations of each class in turn.

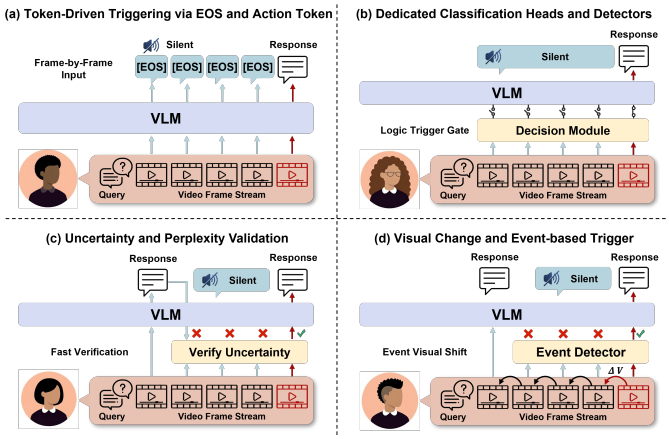

Fig. 5. Taxonomy of proactive response triggering mechanisms for streaming video understanding: (a) Token-Driven Triggering via EOS and Action Token, (b) Dedicated Classification Heads and Detectors, (c) Uncertainty and Perplexity Validation, and (d) Visual Change and Event-based Trigger.

A. Token-Driven Triggering via EOS and Action Token

Token-driven triggering formulates proactive response timing as a token prediction problem within an autoregressive decoding framework. Instead of relying on external classifiers or separate decision modules, the model determines whether to respond directly through its generated tokens at each time step.

1) Binary Triggering and Action Extensions: In its simplest form, token-driven triggering reduces to a binary decision: an End-of-Sequence (EOS) token signals no response, whereas any other token initiates generation. VideoLLM-online [26] introduced this via its "Streaming EOS" objective, suppressing redundant per-frame responses by emitting EOS on non-informative frames, though visual tokens must still be processed every frame. Two orthogonal refinements reduce this cost within the same binary formulation: VideoLLM-MoD [66] adapts the Mixture-of-Depths idea from Mixture-of-Experts [67] to skip computation for many redundant vision tokens per layer (roughly 42% time and 30% memory savings), while LION-FS [68] uses a dual-path design where a lightweight "fast path" triggers at high frame rates and a heavier "slow path" activates only when a response is due.

Beyond binary EOS, a second line enlarges the decision space with dedicated action tokens, differing mainly in the *semantic role* of the actions and the *granularity* of the state machine. On semantic role, STREAM-VLM [69] adds <next> and <feedback> tokens that request additional observations or volunteer feedback for asynchronous interaction, whereas VideoLLM-EyeWO [70] adds an *Ask-High-Res* action that requests higher-resolution input when evidence is insufficient, turning triggering into active perception. On granularity, Streamo [71] models decoding as transitions among <Silence>, <Standby>, and <Response> states with focal weighting [72] to counter state-token imbalance, while ThinkStream [73] interposes a *Think* token within a Watch–Think–Speak loop, reasoning over a Reasoning Compressed Streaming Memory before answering. Extending this idea to multi-agent settings, StreamingClaw [74] defines scenario-specific action tokens that communicate the agent's internal state not only to the user but also to downstream action modules, bridging proactive triggering and embodied control. The specific token sets and their intended semantics are summarized in Table II.

2) Training Challenges and Solutions: Embedding the triggering decision in decoding makes token-driven models particularly sensitive to supervision. The central difficulty is the extreme sparsity of response events: since most frames warrant silence, the silence-to-response ratio can reach 29:1 in a one-minute clip at 2 FPS, biasing the model toward never responding. This sparsity, plus the cost of frame-accurate "speak" annotations, has motivated three complementary mitigations acting on the *data source*, the *training distribution*, and the *supervisory signal*. First, LiveCC [75] reduces annotation cost by interleaving automatic speech recognition (ASR) tokens with frames by temporal alignment, learning streaming behavior from large-scale commentary without manual labels, and adds a pause-aware ellipsis token ("...") to separate genuine end-of-sequence signals from transient pauses. Second, ProAssist [76], built on VideoLLM-online [26], rebalances training via Negative Frame Sub-sampling to curb silence-frame dominance and Iterative Progress Summarization to compress context near the context-length limit. Third, AssistPDA [77], while unifying anomaly prediction, detection, and analysis in one streaming pipeline, strengthens long-range supervision by distilling spatio-temporal relations from offline video-language models via its Spatio-Temporal Relation Distillation module.

3) Discussion: Token-driven triggering provides a unified end-to-end framework by integrating response timing into autoregressive generation, jointly modeling *when to respond* and *what to generate*. It naturally supports extensions from binary EOS prediction to richer action spaces and avoids separate decision modules, simplifying system design and unified optimization. However, this tight coupling has inherent limits. Embedding decision logic within language modeling may degrade general language capabilities under distribution shifts, while the dominance of silence tokens causes severe class imbalance and unstable training. Despite recent mitigations, this remains a fundamental challenge in streaming settings.

B. Dedicated Classification Heads and Detectors

To alleviate the heavy burden of LLM-based token generation, researchers proposed decoupled designs that offload the decision of *when to respond* to a lightweight external module. Taking visual features or their temporal summaries as input, this module outputs a trigger signal (e.g., a probability or binary decision) that activates the downstream generative model. Such designs range from lightweight classification heads to structured detectors or agents, but share the principle of separating decision-making from language generation, keeping the heavyweight LLM dormant until explicitly invoked.

1) Direct Trigger Prediction and Event Modeling: A straightforward instantiation directly predicts whether to trigger a response at each time step from current observations. Works here differ mainly along three axes: the *form* of the trigger signal (a discrete decision versus a graded readiness

score), the *placement* of the decision module (a head on the generative Video-LLM versus a separate external model), and the *training signal* used to supervise it.

The earliest designs emit a *discrete* trigger from heads attached to the backbone. MMDuet [47] attaches an *informative* and a *relevance* head to a pre-trained Video-LLM to decide, frame by frame, whether to interrupt playback and respond, supporting a Video-Text Duet format in which user and model insert messages asynchronously. ViSpeak [27] similarly trains an informative head to predict *when to speak* for subtasks such as wake-up and interruption. A recurring difficulty here is that merging generated responses with incoming inputs in a shared embedding space induces semantic drift, addressed by two-stream templates [27] and dedicated positional-encoding schemes (overlapped, group-decoupled, and gap-isolated) [78]. ToM [79] keeps the discrete formulation but replaces supervised heads with an agent-like loop predicting a binary respond/silence action, trained via a progressive curriculum and reinforcement learning to balance premature responses (hallucination) against late ones (missed utility).

A second group replaces the hard binary decision with a *graded* signal, triggering only when an estimated sufficiency or confidence measure crosses a threshold. EgoSpeak [80] casts speech initiation as continuous speech-activity estimation over a future interval in multi-speaker egocentric settings; StreamReady [81] monitors a learnable readiness token with a lightweight MLP to emit an Answer Readiness Score, firing only once evidence is sufficient; Proact-VL [82] uses a gated response head over a special $<|\text{FLAG}|>$ token for per-second speak/silence decisions; and ROMA [83] adds a lightweight "Speak Head" over synchronized audio-video units to decouple response initiation from content generation in a unified omni-modal model. On module placement, while the above embed the decision head within the backbone, StreamBridge [84] externalizes it entirely, using a small (e.g., 0.5B) activation model to gate a heavyweight offline-trained Video-LLM, converting existing offline models into proactive streamers in a plug-and-play manner. Detailed signal types, placements, and training regimes are summarized in Table II.

Beyond point-wise predictions, another line derives triggering from structured temporal signals, i.e., detected events, boundaries, or state transitions rather than instantaneous observations. Dispider [85] adopts a fully disentangled design with three asynchronous modules for perception, decision, and reaction: a lightweight decision module monitors streaming features and detects relevant events, triggering the heavy reaction module only when necessary. Similarly, StreamMind [86] introduces an event-gated cognition mechanism, where a lightweight gate initialized from shallow language-model layers filters streaming inputs and activates higher-level reasoning only upon detecting event-relevant signals; combined with an Event-Preserving Feature Extractor (EPFE) based on a state-space formulation, it achieves constant-cost feature extraction at high frame rates (up to 100 FPS) while maintaining full temporal awareness.

A further refinement replaces instantaneous decisions with span-level prediction. STRIDE [87] reformulates proactive triggering as span-level activation sequence modeling, employing a lightweight masked diffusion module over sliding temporal windows whose iterative denoising jointly predicts and refines activation signals, yielding temporally coherent "when-to-speak" triggers that respect event boundaries. Other works explicitly model action boundaries: OpenHOUSE [88] uses a lightweight streaming RNN to detect hierarchical action boundaries (goal, step, and sub-step), triggering a frozen Video-LLM to generate descriptions only at these key transitions, while earlier SDVC [89] follows a similar paradigm with event proposal mechanisms (e.g., pointer networks) that pass temporally localized events to downstream captioning. These structured approaches provide more stable and interpretable decision signals, at the cost of increased modeling complexity and potential latency from event detection.

2) Predictive Decision Making.: Unlike reactive triggering, which decides from current or past observations, predictive decision-making anticipates future events and proactively schedules responses, reasoning not only whether to respond but also when relevant information is likely to occur. StreamAgent [90] exemplifies this with an agent-as-detector framework, where a lightweight anticipatory agent integrates task semantics and historical observations to predict the temporal intervals and spatial regions in which task-relevant events may appear, focusing computation on informative segments rather than passively processing the entire stream; this enables timely responses when critical events are sparse or delayed, but adds challenges in modeling uncertainty and error propagation. Taking anticipation toward query intent, Em-Garde [91] introduces a "Propose-Match" paradigm that decouples semantic understanding from streaming perception: it parses user queries into structured, perceptually grounded visual proposals at query time, then a lightweight embedding-based Proposal Matching Module monitors the stream and triggers a response only upon a significant similarity surge between incoming frames and the predefined proposals.

3) Discussion: The primary strength of dedicated heads or detector-based approaches is modularity and flexibility. Decoupling the decision module from the generative Video-LLM enables efficient triggering aligned with the real-time requirements of streaming video understanding, and different designs reflect trade-offs from direct prediction for low-latency decisions to structured or predictive mechanisms for improved reliability and proactivity. However, this decoupling introduces challenges. Separating perception, decision, and generation may cause suboptimal coordination when modules are trained independently. Moreover, effective triggers often require task-specific supervision (e.g., annotated "speak" timestamps or event boundaries), which is expensive to obtain and may limit generalization across domains.

C. Uncertainty and Perplexity Validation

Acting as a bridge between generation-heavy and lightweight approaches, validation mechanisms frame response triggering as a measure of state decay. The core insight is that a valid response maintains its "explanatory power" over subsequent frames. Concretely, the model computes perplexity (PPL) or uncertainty measures (e.g., prediction entropy)

of the previously generated response conditioned on new frames. A low PPL (or low uncertainty) indicates that the current response remains consistent with the incoming content, whereas a significant increase suggests the emergence of new information that warrants an updated response.

1) PPL-based Verification Triggering: LiveStar [48] is representative of this direction. It observes that EOS-based methods (e.g., VideoLLM-online [26], VideoLLM-MoD [66], LION-FS [68]), despite their strong performance, suffer from two characteristic failure modes: visually similar adjacent frames may call for entirely different behaviors, and the discrete EOS token introduces semantic confusion at the decision boundary. To address these, LiveStar proposes a Streaming Verification Decoding (SVeD) framework, in which a decoding gate continuously computes the PPL of the previously generated response on each new frame and triggers a fresh round of full decoding only when the PPL exceeds a threshold, signalling that the prior response is no longer valid. This verification-based formulation is well suited to tasks that require dialogue coherence or stability monitoring over slowly varying content.

2) Discussion: This verification-based paradigm suits scenarios where responses remain valid over extended temporal intervals, such as monitoring stable processes or maintaining coherent descriptions across multiple frames. Its effectiveness, however, relies on the alignment between uncertainty metrics and actual semantic changes, requiring carefully designed thresholding or adaptation strategies to handle dynamic variations across streaming contexts.

D. Visual Change and Event-based Trigger

This is an intuitive and computationally inexpensive triggering mechanism, directly inspired by the physical characteristics of the video itself. This method holds that significant visual changes (such as scene transitions, object appearance/disappearance, violent motion) often signal the occurrence of new events and can therefore serve as natural triggers for responses. Implementation can range from pixel-level frame differencing and motion-level optical flow changes to semantic-level event detection to quantify this change.

1) Visual-change-based Triggering: Enlightened by the phenomenon of *Change Blindness* in human visual perception [92], TimeChat-Online [93] employs a strategy called Differential Token Drop (DTD) to prune redundant tokens by calculating pixel-level and feature-level redundancy. It monitors the ratio of visual tokens dropped between adjacent frames to perceive scene changes: an abrupt change in the drop rate is treated as a scene-transition signal and used as a trigger point for a proactive response. This design couples computational efficiency (token pruning) with proactive decision-making (change detection) within a single mechanism, reducing redundant stream content while preserving accuracy.

However, QueryStream [94] points out that this "change is important" philosophy used in TimeChat-Online [93] conflates raw visual dynamics with true scene transitions. Simply employing this method may cause the model to erroneously trigger responses when encountering a visual change that is unrelated to the user's query. To address this, QueryStream proposes a novel training-free framework that builds query-aware temporal representations to improve performance. It contains two core mechanisms: Query-Aware Differential Pruning (QDP) and Relevance Triggered Active Response (RTAR) policy. QDP filters visual tokens by jointly evaluating semantic relevance to the user's query and temporal novelty and RTAR dynamically determines optimal response moments by monitoring semantic relevance and information density.

ColorTrigger [95] further expands the paradigm to the foundational level of sensory acquisition, proposing a grayscale-always, color-on-demand paradigm for streaming video sensing. Recognizing that continuous high-fidelity RGB capture is prohibitively expensive and often redundant, ColorTrigger employs a causal, windowed grayscale affinity analysis to monitor underlying visual changes. It uses a lightweight training-free quadratic programming trigger and credit-budgeted controller to selectively activate RGB capture, combined with dynamic token routing to reduce sensing and inference costs.

2) Discussion: Visual change-based triggering is attractive because it sidesteps the $O(T)$ cost of invoking the backbone on every frame: delegating the "wake-up" signal to lightweight, often training-free heuristics keeps idle cost near zero, which is especially valuable as multimodal models move from desktop applications to embodied edge devices such as AR glasses and autonomous robots. This efficiency carries a reliability caveat: behavior hinges entirely on how "significant change" is defined, since over-sensitive criteria produce frequent false triggers while conservative thresholds miss salient events. More fundamentally, low-level visual change does not reliably correspond to high-level semantic transition, the gap that motivates query-aware variants such as QueryStream. Closing this gap between visual dynamics and semantic relevance, without reintroducing the heavyweight computation this family is designed to avoid, remains the central open challenge for change-based triggering.

IV. Reactive Streaming Models

Reactive streaming video understanding focuses on real-time perception, reasoning, and interaction over continuously arriving video streams. Unlike offline models that process an entire video at once, reactive models must incrementally digest continuously arriving frames and deliver accurate responses the moment a user query arrives. This paradigm is fundamentally constrained by a structural paradox: the *unbounded temporal nature* of video streams versus the *strictly bounded computational and memory resources* of physical hardware.

To systematically analyze how current methodologies resolve this paradox, we organize existing research along the **lifecycle of information flow** within a streaming system, as illustrated in Fig. 6. This lifecycle serves as an intuitive view of our technical taxonomy and consists of four sequential stages: (1) **Information Input Interception** (Computational Efficiency & Sparse Computing), which filters redundancy before it enters the model; (2) **Working Memory Maintenance** (KV Cache Management & Eviction), which bounds the internal KV cache during continuous processing; (3)

TABLE II
OVERVIEW OF REPRESENTATIVE STREAMING VIDEO UNDERSTANDING MODELS, INCLUDING THEIR CATEGORIES, BACKBONE ARCHITECTURES, MODEL SCALES, TRAINING STRATEGIES, AND AVAILABLE RESOURCES.

#	Model	Date	Venue	Categories	Backbone	Scale	Training	Github
1	VST [96]	2026.03	ECCV	KV Cache Management	Qwen2.5-VL	7B	SFT+RL	Link
2	TaYS [97]	2026.03	CVPR	KV Cache Management	Qwen2.5-VL	7B	SFT+RL	Link
3	FluxMem [98]	2026.03	CVPR	Memory Summarization	Qwen2.5-VL	7B	Training-free	Link
4	TWW [99]	2026.03	ECCV	Memory Summarization	Qwen3-VL	8B	SFT	Link
5	OmniStream [100]	2026.03	arXiv	Computational Efficiency	DINOv3	7B	SFT	Link
6	AutoGaze [101]	2026.03	CVPR	Computational Efficiency	NVILA-8B-Video	8B	SFT+RL	Link
7	STRIDE [87]	2026.03	arXiv	Classification Heads	Qwen3-VL	2B	SFT	Link
8	ColorTrigger [95]	2026.03	CVPR	Visual Change	InternVL3.5	8B	Training-free	Link
9	StreamingClaw [74]	2026.03	arXiv	Token-Driven Triggering	/	/	SFT	Link
10	Em-Garde [91]	2026.03	arXiv	Classification Heads	Qwen2.5-VL	7B	SFT+RL	Link
11	ThinkStream [73]	2026.03	ECCV	Token-Driven Triggering	Qwen2.5-VL	3B	SFT+RL	Link
12	StreamReady [81]	2026.03	CVPR	Classification Heads	Qwen2-VL	7B	SFT	N/A
13	Proact-VL [82]	2026.03	ICML	Classification Heads	Qwen2-VL	7B	SFT	Link
14	EventMemAgent [102]	2026.02	arXiv	Memory Summarization	Qwen3-VL	8B	SFT+RL	Link
15	WeaveTime [103]	2026.02	CVPR	Retrieval Augmented	LLaVA-OV	7B	SFT	Link
16	ROMA [83]	2026.01	arXiv	Classification Heads	Qwen2.5-Omni	/	SFT	Link
17	QueryStream [94]	2026.01	ICLR	Visual Change	Qwen2.5-VL	7B	Training-free	Link
18	HERMES [104]	2026.01	ACL	Memory Summarization	Qwen2.5-VL	7B	Training-free	Link
19	STC [105]	2025.12	CVPR	Computational Efficiency	Qwen2-VL	7B	Training-free	Link
20	VideoScaffold [106]	2025.12	arXiv	Memory Summarization	Vicuna	7B	SFT	Link
21	Streamo [71]	2025.12	arXiv	Token-Driven Triggering	Qwen2.5-VL	7B	SFT	Link
22	StreamingAssistant [107]	2025.12	arXiv	KV Cache Management	Qwen2.5-VL	7B	Training-free	N/A
23	V-Rex [108]	2025.12	HPCA	Retrieval Augmented	Llama-3	8B	SFT	N/A
24	Venus [109]	2025.12	INFOCOM	Retrieval Augmented	Qwen2-VL	7B	Training-free	N/A
25	ToM [79]	2025.12	arXiv	Classification Heads	Qwen2.5-VL	3B	SFT+RL	N/A
26	MMDuet2 [110]	2025.12	ICLR	Token-Driven Triggering	Qwen2.5-VL	3B	SFT+RL	Link
27	LiveStar [48]	2025.11	NeurIPS	Perplexity Validation	InternLM2.5	8B	SFT	Link
28	CacheFlow [111]	2025.11	arXiv	Retrieval Augmented	LLaVA-OV	7B	Training-free	N/A
29	StreamKV [112]	2025.11	AAAI	Retrieval Augmented	LLaVA-OV	7B	Training-free	Link
30	VideoLLM-EyeWO [70]	2025.10	NeurIPS	Token-Driven Triggering	LLaMA-3	8B	SFT	Link
31	StreamingVLM [113]	2025.10	arXiv	KV Cache Management	Qwen2.5-VL	7B	SFT	Link
32	StreamingTOM [114]	2025.10	arXiv	Retrieval Augmented	LLaVA-OV	7B	Training-free	Link
33	rLiVS [115]	2025.10	NeurIPS	Retrieval Augmented	LLaVA-OV	7B	Training-free	N/A
34	video-SALMONN S [116]	2025.10	arXiv	Memory Summarization	Qwen3-VL	8B	SFT+TTT	N/A
35	StreamForest [117]	2025.09	NeurIPS	Memory Summarization	Qwen2	7B	SFT	Link
36	OpenHOUSE [88]	2025.09	ICCV	Classification Heads	InternVL2	8B	SFT	N/A
37	StreamMem [118]	2025.08	arXiv	KV Cache Management	Qwen2.5-VL	3B	Training-free	Link
38	StreamAgent [90]	2025.08	arXiv	Classification Heads	Qwen2.5-VL	7B	SFT	N/A
39	OVG-HQ-Unify [119]	2025.08	ICCV	Memory Summarization	/	/	TTT	Link
40	StreamVLN [120]	2025.07	arXiv	KV Cache Management	Qwen2	7B	SFT	Link
41	InfiniPot-V [121]	2025.06	NeurIPS	KV Cache Management	Qwen2.5-VL	7B	Training-free	Link
42	CogReasoner [122]	2025.06	arXiv	Retrieval Augmented	Qwen2.5	7B	SFT	Link
43	ProAssist [76]	2025.06	EMNLP	Token-Driven Triggering	LLaMA-3.1	8B	SFT	Link
44	Flash-VStream [51]	2025.06	ICCV	Memory Summarization	Qwen2-VL	7B	SFT	Link
45	StreamBridge [84]	2025.05	NeurIPS	Classification Heads	Qwen2-VL	7B	SFT	Link
46	LiveVLM [123]	2025.05	arXiv	Retrieval Augmented	LLaVA-OV	7B	Training-free	N/A
47	TimeChat-Online [93]	2025.04	ACM MM	Visual Change	Qwen2.5-VL	7B	SFT	Link
48	Streamformer [124]	2025.04	ICCV	Computational Efficiency	LLaVA-Next	7B	SFT	Link
49	LiveCC [75]	2025.04	CVPR	Token-Driven Triggering	Qwen2-VL	7B	SFT	Link
50	ProVideLLM [125]	2025.04	ICCV	Memory Summarization	Llama-3.1	8B	SFT	Link
51	ViSpeak [27]	2025.03	ICCV	Classification Heads	Qwen2.5	7B	SFT	Link
52	AssistPDA [77]	2025.03	arXiv	Token-Driven Triggering	Qwen2-VL	2B	SFT	N/A
53	VideoScan [126]	2025.03	arXiv	Memory Summarization	LLaVA-Video	7B	SFT	Link
54	LION-FS [68]	2025.03	CVPR	Token-Driven Triggering	Llama-3	8B	SFT	Link
55	StreamMind [86]	2025.03	ICCV	Classification Heads	VideoLLaMA2	8B	SFT	Link
56	ReKV [49]	2025.03	ICLR	Retrieval Augmented	LLaVA-OV	7B	Training-free	Link
57	EgoSpeak [80]	2025.02	NAACL	Classification Heads	LSTR	3B	SFT	Link
58	StreamingChat [127]	2025.02	ICLR	KV Cache Management	InternVL2	8B	SFT	Link
59	StreamChat [128]	2025.01	ICLR	Memory Summarization	LongVA	7B	Training-free	Link
60	Dispider [85]	2025.01	CVPR	Classification Heads	Qwen2	7B	SFT	Link
61	VITA-1.5 [129]	2025.01	NeurIPS	Computational Efficiency	Qwen2	7B	SFT	Link
62	VideoChat-Online [50]	2025.01	CVPR	Memory Summarization	InternVL2	4B	SFT	Link
63	StreamChat [130]	2024.12	CVPR	Computational Efficiency	Qwen2.5	7B	SFT	Link
64	SDQES [131]	2024.12	NeurIPS	Computational Efficiency	EgoVideo	7B	SFT	N/A
65	MMDuet [47]	2024.11	EMNLP	Classification Heads	LLaVA-OV	7B	SFT	Link
66	VideoLLaMB [132]	2024.09	ICCV	Memory Summarization	Vicuna	7B	SFT	Link
67	VideoLLM-MoD [66]	2024.08	NeurIPS	Token-Driven Triggering	Llama-3	8B	SFT	N/A
68	STREAM-VLM [69]	2024.07	NeurIPS	Token-Driven Triggering	LLaMA-2	7B	SFT	Link
69	VideoLLM-online [26]	2024.06	CVPR	Token-Driven Triggering	LLaMA-3	8B	SFT	Link
70	VideoStreaming [44]	2024.05	NeurIPS	Memory Summarization	Vicuna	7B	SFT	N/A
71	VideoNarrator [133]	2024.05	ACL	Memory Summarization	Baichuan	7B	SFT	Link
72	StreamingDVC [45]	2024.04	CVPR	Memory Summarization	T5-Base decoder	/	SFT	Link

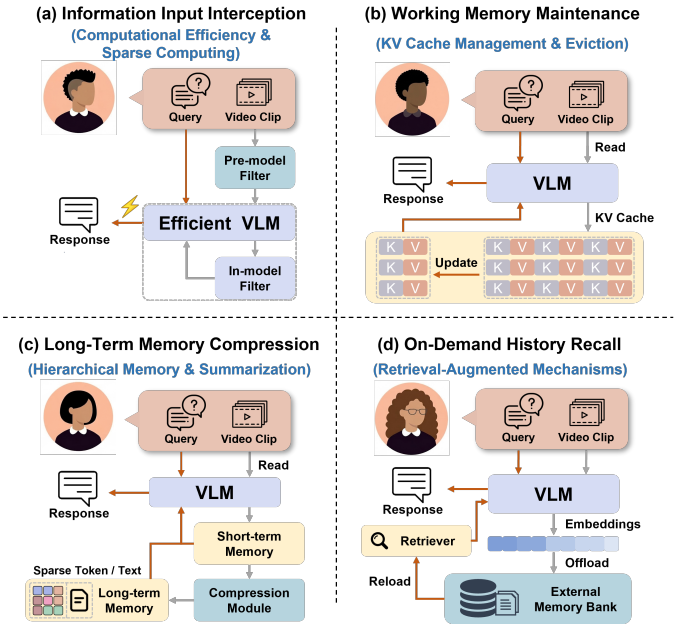

Fig. 6. Lifecycle view of reactive streaming models with corresponding technical categories: (a) Information Input Interception, (b) Working Memory Maintenance, (c) Long-Term Memory Compression, and (d) On-Demand History Recall.

Long-Term Memory Compression (Hierarchical Memory & Summarization), which abstracts historical context into compact representations; and (4) **On-Demand History Recall** (*Retrieval-Augmented Mechanisms*), which retrieves relevant past information when a query arrives.

The following sections review these categories from the lifecycle perspective while maintaining the corresponding technical taxonomy used throughout this survey.

A. Information Input Interception

The first line of defense in streaming video understanding occurs at the input stage. To prevent redundant visual tokens from overwhelming the generative model, researchers focus on intercepting and sparsifying the data stream. By modifying attention mechanisms or aggressively pruning visual tokens, models can maintain low-latency throughput even at high frame rates.

1) Token-Level Sparsification and Re-use: Rather than modifying the architecture, a direct strategy is to eliminate spatial and temporal redundancy at the token level.

STC [105] introduces a hierarchical token compression framework that combines token reuse and selective pruning. During the visual encoding stage, STC employs a caching mechanism to reuse intermediate representations from reference frames, identifying dynamic tokens based on inter-token similarity and recomputing only these regions while directly reusing static content, which cuts redundant computation in the backbone network. Before feeding tokens into the language model, STC further applies a pruning strategy guided by both temporal and spatial saliency signals, where tokens are selected based on their relevance to historical context and current-frame importance. By jointly leveraging token reuse and selective sparsification, this approach reduces sequence length while preserving critical information for downstream reasoning.

Taking this further, AutoGaze [101] proposes a lightweight module that removes redundant image patches before visual Transformer (ViT) processing. Unlike existing methods that prune tokens solely within large language models, AutoGaze adopts an autoregressive approach to select a minimal set of multi-scale patches, enabling video reconstruction within a user-specified reconstruction error threshold. This eliminates spatiotemporal redundancy at the earliest stage of the pipeline. Trained through a combination of next-token prediction and reinforcement learning, the method reduces visual token counts by 4× to 100× and achieves up to 19× acceleration for both ViT and multimodal large language models, enabling efficient scaling to video understanding tasks involving 1024 frames at 4K resolution.

2) Attention Mechanism Optimization: Standard self-attention scales quadratically with sequence length, making it intractable for infinite streams. Rather than discarding tokens, the methods below restructure the attention computation itself so that the per-frame cost becomes independent of stream length, differing in whether they exploit temporal recurrence, cross-modal asymmetry, or causal structure.

SDQES [131] proposes the QR-Adapter and RN-Adapter, which adopt recurrent or linear attention mechanisms to achieve incremental state updates. This design transforms the computational graph in the temporal dimension from globally dense connections to locally sparse connections, making the computational cost for processing each new frame constant and independent of historical length. Although designed for offline video understanding tasks, this design may also serve as a reference for streaming video understanding tasks.

Other approaches exploit the asymmetry between modalities to further reduce complexity. StreamChat (CVPR'25) [130] designs a cross-attention mechanism where only text tokens serve as queries to retrieve visual tokens, reducing the number of pairwise interactions. This is complemented by a visual feed-forward refinement module (V-FFN), which iteratively enhances visual features to better match sparse attention queries, and a parallel 3D-RoPE encoding that provides explicit spatiotemporal alignment, allowing the model to focus attention on the correct temporal segments without additional search overhead.

In addition to architectural redesign, causal attention is often introduced to explicitly enforce streaming constraints. StreamFormer [124] incorporates causal temporal attention into vision Transformers, ensuring that each frame only attends to its past observations. Combined with KV caching, this design avoids redundant recomputation of historical frames and enables linear scaling with respect to sequence length. Moreover, its multi-level multi-task training encourages the model to learn unified spatiotemporal representations that can be efficiently reused across different downstream tasks. Building upon these principles, OmniStream [100] proposes a unified streaming visual backbone that integrates causal spatiotemporal attention with a persistent KV cache to support frame-by-frame online processing without recomputing

past frames. By extending two-dimensional rotary position embedding to the spatiotemporal domain (3D-RoPE), the model enables relative reasoning about both spatial "where" and temporal "when" simultaneously, improving streaming processing efficiency.

3) Discussion: Input interception dramatically reduces latency but operates "blind" to future user queries: however sophisticated, aggressive pruning heuristics risk discarding latent visual cues that later prove critical for unexpected questions. Balancing early-stage efficiency against the preservation of subtle semantic details remains a fundamental friction point in reactive systems.

B. Working Memory Maintenance

Once visual information successfully enters the Transformer, it is stored in the Key-Value (KV) cache. However, accumulating KV states across an endless video stream will inevitably exhaust hardware memory ($O(T)$). Working memory maintenance strategies aim to enforce a strictly bounded memory footprint ($O(1)$) by selectively retaining, merging, or evicting cache tokens.

1) Query-Agnostic KV Eviction: Because the streaming model must manage its memory before the user poses a query, eviction policies cannot rely on query relevance and must instead infer token importance from intrinsic signals: raw feature statistics, attention as a proxy for relevance, and spatial redundancy. InfiniPot-V [121] is training-free and statistic-driven, operating at two levels: a Temporal-axis Redundancy (TaR) measure discards frames whose similarity $Sim(V_t, V_{t-1})$ to their predecessor exceeds a threshold, and a Value-Norm (VaN) criterion exploits the heavy-tailed Value distribution to retain only the top-K tokens with the largest L2 norms ($\|V\|_2$). StreamMem [118] uses attention as a proxy via a *Proxy Query* (a generic learnable token approximating user attention), performing intra-frame pruning of low-scoring tokens and inter-frame merging of similar tokens to keep a constant $O(1)$ footprint. StreamingAssistant [107] models spatial redundancy through the Maximum Similarity to Spatially Adjacent Video Tokens (MSSAVT) metric, pruning locally redundant tokens while a masked pruning strategy preserves representative tokens to retain spatial layout.

2) Training-Inference Alignment and Multi-Timescale Buffers: Beyond query-agnostic eviction, robust working memory must address the distribution shift between offline training and online inference, and the varying temporal utility of historical information. StreamingVLM [113] identifies an "attention sink" phenomenon and retains these initial tokens alongside a sliding window of recent observations, while an overlapped sliding window during supervised fine-tuning simulates the limited-context inference setting to stabilize long-horizon processing. TaYS [97] aligns training and inference through three designs: a streaming attention mask enforcing causal visibility, decoupled positional encoding that separates visual and reasoning token spaces, and a parallel dual KV cache enabling concurrent visual absorption and text generation, together yielding near-zero time-to-first-token (TTFT) latency without inference-time adaptation. VST [96] adopts a dual-timescale system, pairing a second-level visual buffer for immediate perception with a minute-level textual memory of reasoning summaries under event-boundary FIFO eviction; supported by a knowledge-graph-based synthesis pipeline for multi-hop "streaming thought" data, it is post-trained via SFT plus RL, where the RL stage propagates final-answer rewards over the reasoning trajectory using group-relative advantage.

3) Discussion: KV cache eviction prevents Out-Of-Memory (OOM) crashes but is fundamentally limited by its query-agnostic nature: permanently deleting "unimportant" tokens without knowing what the user will care about inherently restricts the depth of retrospective reasoning. Context-aware eviction that safely compresses rather than irreversibly destroys information is crucial for future KV management.

C. Long-Term Memory Compression

For temporal sequences that far exceed the capacity of a bounded KV cache sliding window, reactive models must abstract historical data into higher-level semantic structures. This stage of the lifecycle transforms raw, dense visual tokens into compact, persistent long-term memory.

1) Explicit Hierarchical Structures: A recurring template underlies this line of work: a fine-grained short-term buffer for immediate perception is paired with an abstracted long-term store for global coherence. What distinguishes methods is how the long-term tier is *organized*, ranging from fixed-depth summary pyramids to tree-structured event hierarchies and discrete event archives. Organized by temporal granularity, the Pyramid Memory Bank (PMB) of VideoChat-Online [50] stacks units from fine to coarse, with a bottom layer of recent-frame detail, a middle layer of segment-level semantics, and a top layer of global summary, supporting past-current-future perspectives under dynamically changing answers. Organized instead by functional role, StreamChat (ICLR'25) [128] (whose benchmark StreamBench is discussed in Sec. V) splits storage into short-term visual features, online-compressed long-term semantic summaries, and dialog memory for multi-turn consistency. Imposing an explicit growing structure, StreamForest [117] couples "Fine-grained Spatio-Temporal Windows" for short-term high-resolution perception with a "Persistent Event Memory Forest" that organizes long-term content via adaptive event-node clustering and tree-like merging. Pushing this further, EventMemAgent [102] segments frames into coherent event clips via boundary detection and reservoir sampling for short-term memory, and archives completed events into first-frame images, captions, embeddings, and change logs for long-term memory, enabling multi-granularity retrieval coupled with a perception toolkit and reinforcement learning.

2) Semantic Summarization and Implicit Memory: Explicit hierarchical structures incur growing overhead as the stream progresses. For extreme compression, an alternative paradigm distills history into textual summaries (*explicit textual abstraction*) or encodes context into model parameters (*implicit parametric storage*), trading visual detail for scalable semantic retention; we organize methods by where the compressed history resides. Keeping history inspectable

as text, ProVideLLM [125] compresses minutes-long procedural observations into concise text summaries interleaved with short-term high-resolution visual tokens, and its DETR-QFormer connector focuses on hand-object interaction regions for fine-grained low-memory understanding. Pushing this further, TWW [99] eliminates raw visual tokens entirely, encoding each temporal segment as a textual "memory note" of entities, actions, and state changes that is retrieved implicitly via attention. The second route folds history into network weights: OVG-HQ-Unify [119] introduces a Parameter Memory Block (PMB) that implicitly stores history by dynamically updating parameters during inference, supporting text, image, and video-clip queries. Likewise weight-based, Video-SALMONN S [116] pairs a Test-Time Training (TTT) module using lightweight Hessian-free optimization to encode long-term dependencies into parameters with a fixed-size memory bank pruned by cosine-similarity eviction. Straddling both routes, HERMES [104] operates at the level of memory bookkeeping via cross-layer importance-score smoothing from deep to shallow layers and a position re-indexing mechanism for dynamic remapping and rotation correction.

3) Discussion: Semantic summarization drastically reduces storage overhead at a severe cost: the irreversible loss of visual grounding. Once a scene is abstracted into a text summary or a parametric weight update, the model can no longer answer fine-grained spatial queries (e.g., "What was written on the license plate in the background 10 minutes ago?"), as summarization inherently prioritizes narrative continuity over pixel-level fidelity.

D. On-Demand History Recall

The final stage of the reactive lifecycle occurs when a user actually submits a query Q^t. Since the vast majority of historical data has been compressed, archived, or evicted, the system must utilize Retrieval-Augmented Generation (RAG) concepts to retrieve relevant context and guide the LLM's response.

1) KV-Based Memory Construction and Indexing: Before any historical context can be retrieved, the continuous and unstructured video stream must first be transformed into an organized, indexable format. This line of work focuses on converting raw streaming inputs into discrete key-value (KV) memory units paired with compact semantic descriptors, laying the foundational architecture for efficient, on-demand recall.

The foundational instantiation treats the stream itself as a retrievable knowledge base. Drawing on RAG [134]–[136], ReKV [49] divides incoming frames into fixed segments, computes their key-value (KV) pairs and summary indexes, and stores them in memory or on disk; at query time, it selects the Top-K similar historical KV chunks by question-index similarity and adds them to the context to guide generation. Since ReKV stores the complete KV, wasting storage and introducing noise, LiveVLM [123] applies dual compression: it first discards low-semantic tokens using internal attention scores, then merges all original KVs within the same frame to prevent dropping entire frames.

Rather than compressing after the fact, StreamKV [112] decides retention as tokens arrive, evaluating visual-token similarity to predefined criteria and dynamically setting retention thresholds under a fixed memory budget to organize selected tokens into a structured KV library. In a similar spirit, StreamingTOM [114] leverages temporal redundancy by distinguishing dynamic and static tokens, allocating budgets by their proportions, selecting dynamic tokens by spatial saliency while clustering and merging static tokens.

2) Adaptive Retrieval Strategies: While memory construction dictates *how* history is stored, the retrieval strategy determines *what* is extracted. Standard top-K cosine similarity often yields redundant or fragmented results, prompting a shift toward dynamic, context-aware selection. Concerning what signal retrieval operates over, CacheFlow [111] introduces a consensus-based mechanism that jointly considers similarity scores across Transformer layers, prioritizing tokens consistently selected at different abstraction levels for robustness. In contrast, rLiVS [115] avoids direct KV retrieval and operates on textual summaries, using Maximum Marginal Relevance (MMR) to balance similarity and diversity. Going beyond fixed policies, WeaveTime [103] gates memory access through predictive entropy: its PCDF-Cache assesses whether current observations suffice and triggers a hierarchical coarse-to-fine pipeline, from frame-level filtering to token-level ranking, only when uncertainty is high. Shifting the axis toward where retrieval executes, Venus [109] adopts an edge–cloud collaborative architecture, offloading memory construction and keyframe selection to edge devices during ingestion while the cloud performs query-time retrieval and reasoning. In a different direction, V-Rex [108] introduces a hardware–software co-designed solution with a dynamic KV retrieval algorithm that loads only a minimal subset of relevant clusters by cumulative attention scores, executed on a dedicated retrieval engine. Collectively, these methods move retrieval from fixed top-K lookup toward query-adaptive, layer-aware, and resource-aware selection.

3) Discussion: Retrieval bridges infinite video length and finite context windows but fundamentally disrupts continuous spatiotemporal causality: concatenating semantically similar "chunks" destroys the chronological sequence of events. Such models thus excel at finding static historical facts yet struggle with complex multi-hop causal reasoning (e.g., "Did Event A directly cause Event B, or did they just look similar?").

V. Benchmarks and Datasets

In contrast to conventional offline video benchmarks, streaming video understanding requires evaluation protocols that account for temporal dynamics, continuous interaction, and real-time constraints. We categorize existing benchmarks based on task types, including multi-turn dialogue and question answering, real-time captioning and narration, and proactive response with timing evaluation. Importantly, these task categories are closely related to the two interaction paradigms introduced earlier. Multi-turn dialogue and QA tasks primarily correspond to the reactive paradigm, where responses are triggered by user queries. Real-time captioning and narration

TABLE III
REPRESENTATIVE BENCHMARKS AND DATASETS FOR STREAMING VIDEO UNDERSTANDING, COVERING MULTI-TURN DIALOGUE & QA, REAL-TIME CAPTIONING & NARRATION, AND PROACTIVE RESPONSE WITH TIMING EVALUATION.

#	Dataset	Date	Venue	Focus Area	Scale	Link
1	ProReady-QA [81]	2026.03	CVPR 2026	Proactive Response & Timing Evaluation	5.0K QAs	Link
2	Live Gaming Benchmark [82]	2026.03	ICML 2026	Proactive Response & Timing Evaluation	3.0K videos	Link
3	RIVER Bench [137]	2026.03	arXiv	Multi-Turn Dialogue & QA	4.3K QAs	Link
4	StreamEQA [138]	2025.12	arXiv	Multi-Turn Dialogue & QA	21.0K QAs	N/A
5	StreamGaze [139]	2025.12	arXiv	Proactive Response & Timing Evaluation	8.5K QAs	Link
6	OmniStar-RNG [48]	2025.11	NeurIPS 2025	Real-time Captioning & Narration	20.1K videos	Link
7	StreamingCoT [140]	2025.10	ACM MM 2025	Multi-Turn Dialogue & QA	5.7K videos	Link
8	ESTP-Bench [70]	2025.10	NeurIPS 2025	Proactive Response & Timing Evaluation	2.3K QAs	Link
9	ODV-Bench [117]	2025.09	NeurIPS 2025	Multi-Turn Dialogue & QA	32.0K QAs	Link
10	OST-Bench [141]	2025.07	NeurIPS 2025	Multi-Turn Dialogue & QA	10.0K QAs	Link
11	ProactiveVideoQA [142]	2025.07	arXiv	Proactive Response & Timing Evaluation	3.5K QAs	Link
12	PROASSIST [76]	2025.06	EMNLP 2025	Proactive Response & Timing Evaluation	30.1K QAs	Link
13	RTV-Bench [143]	2025.05	NeurIPS 2025	Multi-Turn Dialogue & QA	4.6K QAs	Link
14	Live-WhisperX-526K [75]	2025.04	CVPR 2025	Real-time Captioning & Narration	526.0K videos	Link
15	Live-CC-5M [75]	2025.04	CVPR 2025	Real-time Captioning & Narration	5.0M videos	Link
16	OmniMMI [144]	2025.03	CVPR 2025	Proactive Response & Timing Evaluation	2.3K QAs	Link
17	VAPDA-127K [77]	2025.03	arXiv	Proactive Response & Timing Evaluation	2.4K videos	N/A
18	YT-Conversation [80]	2025.02	NAACL 2025	Multi-Turn Dialogue & QA	414 videos	Link
19	SVBench [127]	2025.02	ICLR 2025	Multi-Turn Dialogue & QA	50.0K QAs	Link
20	OVBench [50]	2025.01	CVPR 2025	Multi-Turn Dialogue & QA	4.9K QAs	Link
21	OVO-Bench [56]	2025.01	CVPR 2025	Proactive Response & Timing Evaluation	3.1K QAs	Link
22	StreamBench [128]	2025.01	ICLR 2025	Multi-Turn Dialogue & QA	1.8K QAs	Link
23	StreamingBench [55]	2024.11	arXiv	Multi-Turn Dialogue & QA	4.5K QAs	Link
24	MMDuetIT [47]	2024.11	EMNLP 2025	Proactive Response & Timing Evaluation	109.0K videos	Link
25	TemporalBench [145]	2024.10	arXiv	Multi-Turn Dialogue & QA	10.0K QAs	Link
26	QEVD-FIT-COACH [69]	2024.07	NeurIPS 2024	Proactive Response & Timing Evaluation	74 videos	Link

can be viewed as a hybrid setting, depending on whether generation is continuous or user-driven. In contrast, proactive response benchmarks directly evaluate the **proactive paradigm**, focusing on when a system should respond under streaming constraints.

A. Multi-Turn Dialogue & QA

This category evaluates query-driven streaming understanding, where models answer user questions over continuously evolving video streams. Aligning with the reactive paradigm, these benchmarks emphasize long-context reasoning, temporal grounding, and multi-turn interaction, capabilities essential for follow-up-rich dialogues that isolated video QA cannot assess, with growing attention to cross-turn coreference resolution, long-term memory retention, and temporally consistent reasoning over unbounded inputs.

An early wave of benchmarks enforces the causal constraint that distinguishes streaming QA from its offline counterpart, requiring answers from past and current frames alone. StreamingBench [55], SVBench [127], and StreamBench [128] forbid future access during answering, while adding multi-turn dialogue scenarios (SVBench) and latency-aware metrics over diverse media types (StreamBench). A more demanding dimension is temporal dynamism, where the correct answer to a fixed question drifts as the video unfolds. TemporalBench [145] stresses fine-grained motion understanding over event progression, action frequency, and motion magnitude, adopting a Multiple Binary Accuracy (MBA) metric to suppress language priors, whereas RTV-Bench [143] instantiates dynamism through multi-timestamp QA, where the same question yields different answers as the stream evolves.

Other benchmarks pursue comprehensive evaluation across temporal scales. OVBench [50] decomposes streaming perception into fine-grained subtasks across past, present, and future contexts, covering spatial perception, temporal reasoning, memory retrieval, and future prediction; RIVER Bench [137] quantifies memory decay curves across four temporal intervals with fine-grained timestamp annotations for query, cue, and response times. A further group grounds evaluation in embodied scenarios. OST-Bench [141] measures whether agents retain spatial awareness and long-term memory during incremental exploration, while StreamEQA [138] jointly evaluates perception, interaction, and planning under backward, real-time, and forward perspectives from partial observations.

Domain-specialized benchmarks target particular skills. YT-Conversation [80] builds interactive scenarios from untrimmed YouTube videos, emphasizing when an agent should speak rather than merely what to say in egocentric conversations. ODV-Bench [117] targets autonomous driving with traffic sign recognition, future risk forecasting, and accident analysis under highly dynamic conditions. StreamingCoT [140] introduces dynamic multimodal Chain-of-Thought reasoning through hierarchical temporal annotations and evolving QA pairs, yielding interpretable reasoning chains grounded in spatiotemporal object transitions.

B. Real-time Captioning & Narration

This category evaluates a model's ability to generate temporally aligned descriptions of streaming video content. Depending on the interaction setting, these tasks can be either reactive (on-demand captioning) or proactive (continuous or event-triggered narration), making them a bridge between the two paradigms. Recent captioning datasets [146] emphasize not only what to say but when to say it, introducing temporal alignment and response timing as key factors.

LiveCC [75] proposes LiveSports-3K, leveraging streaming ASR signals to evaluate low-latency and human-like commentary generation. OmniStar-RNG (LiveStar) [48] introduces a real-time narration task with timing-sensitive metrics such as Timing Difference (TimDiff), explicitly measuring whether models generate responses at appropriate semantic transitions. QEVD-FIT-COACH [69] further extends captioning to interactive scenarios, requiring models to proactively identify user errors and provide timely corrective feedback without explicit queries. These benchmarks highlight the importance of balancing semantic accuracy with temporal precision in streaming narration.

C. Proactive Response & Timing Evaluation

This category directly evaluates the **proactive paradigm**, focusing on a model's ability to determine *when* to respond in a dynamic and evolving video stream. Unlike traditional QA tasks that assume explicit queries, proactive benchmarks require models to continuously monitor visual input, identify critical moments, and generate timely responses without user intervention, jointly assessing semantic correctness and response timing while avoiding premature or redundant outputs.

Egocentric video poses unique challenges from its unstructured visual dynamics and the tight coupling between the wearer's intent and visual context. PROASSIST [76] studies proactive assistant dialogue in egocentric procedural videos, framing proactive response as a turn-taking and timing decision problem in which the model anticipates user needs before errors occur. ESTP-Bench [70] emphasizes just-in-time reasoning in ego-streaming videos via an ESTP-F1 protocol that evaluates whether coherent answers fall within valid temporal intervals. StreamGaze [139] uses human gaze as a proxy for intent and focus, requiring response timing to align with user attention dynamics.

Several benchmarks formalize response timing more explicitly. ProactiveVideoQA [142] introduces a proactive window, turning video QA into a temporal localization problem in which the model answers only after relevant evidence becomes available. ProReady-QA [81] refines this with annotated answer-evidence windows and the Answer Readiness Score, which applies asymmetric penalties to early and late responses. MMDuetIT [47] interleaves video and text as a "duet" to align responses with the evolving stream. OVO-Bench [56] generalizes online video understanding into backward tracing, real-time perception, and forward active responding, the last requiring the model to delay its answer until sufficient future context arrives.

Other benchmarks broaden proactive timing beyond generic QA. VAPDA-127K [77] targets online anomaly prediction, detection, and analysis, where models should identify abnormal events early while avoiding costly false alarms. Omni-iMMI [144] combines state grounding, proactive reasoning, and proactive turn-taking. The Live Gaming Dataset [82] extends the setting to real-time AI companions, where chunk-wise speak/silence labels require per-second scheduling decisions, making response density and latency part of the benchmark itself.

D. Discussion

Existing benchmarks for streaming video understanding still show clear gaps in how model capabilities are evaluated. Many benchmarks mix different abilities together, making it hard to separate short-term perception from long-term reasoning and memory. At the same time, long-horizon temporal dependencies are not sufficiently emphasized, which limits the ability to clearly compare different streaming approaches. Another limitation lies in the evaluation of proactive capabilities. Although recent benchmarks attempt to include proactive settings, most of them still rely on pre-defined query timestamps or annotated response windows. This design remains essentially reactive, and does not fully reflect a model's ability to decide *when* and *what* to respond in open-ended streaming scenarios. These observations suggest the need for benchmark designs that more clearly separate different abilities and better capture the core challenges of streaming video understanding.

VI. Challenges and Future Directions

A. Challenges

Streaming video understanding operates under a causal and resource-constrained setting, where models must process continuously incoming data without access to future frames or the ability to revisit past observations. This introduces inherent challenges in maintaining long-term temporal context, performing real-time perception, and making timely decisions under limited computation. Across both reactive and proactive paradigms, the core challenges lie in temporal modeling, response triggering, and the trade-off between efficiency and reasoning accuracy.

1) Proactive Paradigm for Streaming Video Understanding: The proactive response paradigm requires models to continuously perceive streaming inputs, accumulate evidence, and autonomously decide when and what to respond without explicit query triggers. Compared to reactive settings, this paradigm introduces additional challenges in temporal decision-making, long-horizon reasoning, and real-time efficiency under streaming constraints.

A central challenge is the *(1) Temporal Triggering Decision*: models must align evolving visual evidence with implicit query intent to decide when sufficient information has been observed. This is inherently ambiguous, as different tasks require different levels of evidence accumulation with no unified criterion, forcing a balance between premature responses based on incomplete evidence and delayed responses that reduce efficiency. *(2) Long-Horizon Memory Management* poses a

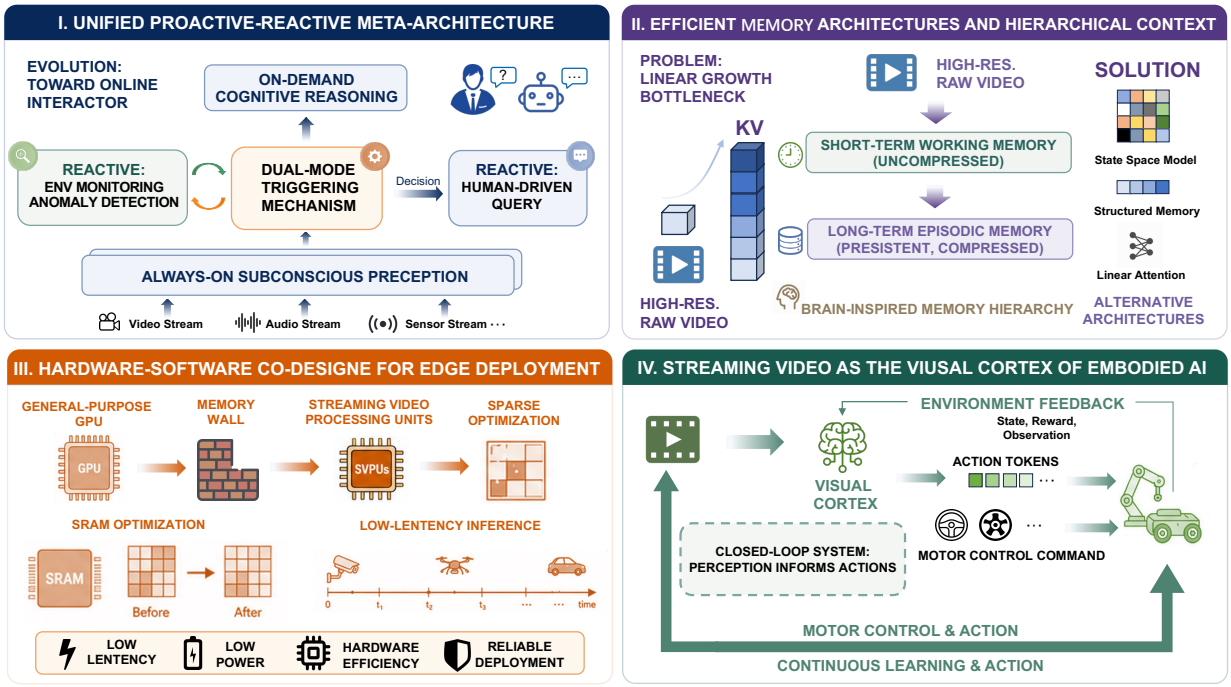

Fig. 7. The future roadmap of streaming video understanding: a full-stack, closed-loop ecosystem bridging physical hardware and embodied applications across four tiers, namely (1) a unified meta-architecture routing proactive-reactive triggers, (2) efficient $\mathcal{O}(1)$ episodic memory, (3) hardware-software co-design with dedicated SVPUs, and (4) embodied AI integration, where the active perception loop couples an agent's actions with its subsequent egocentric input.

second tension, since proactive responses may depend on both past observations and anticipated future events: naïve context accumulation quickly exceeds memory limits, while aggressive truncation discards critical temporal dependencies, leaving selective long-term memory and temporally consistent reasoning a fundamental challenge. *(3) Real-Time Inference Efficiency* is equally demanding, as continuous per-step monitoring and decision-making impose strict latency constraints; although visual encoding can be relatively efficient, autoregressive decoding and repeated decision evaluation introduce significant overhead, trading reasoning accuracy against latency. Finally, *(4) Adaptive Triggering and Response Granularity* remains open, because different tasks demand different response frequencies and detail levels, so models must jointly model temporal dynamics, task complexity, and uncertainty rather than rely on fixed triggering strategies.

2) Reactive Paradigm for Streaming Video Understanding: The reactive response paradigm is characterized by continuous streaming input, query-driven interaction, and immediate, non-revisable responses. Compared to offline processing, it operates under strict causal and real-time constraints, leading to several fundamental challenges arising from the tension between unbounded temporal context, limited computational resources, and the need for accurate query-conditioned reasoning.

The first challenge concerns *(1) Memory and Information Management*. Models must maintain long-term visual information under fixed budgets, where aggressive compression or truncation may discard critical temporal cues while unbounded accumulation hinders scalability; lacking prior knowledge of future queries, they must further preserve salient information in a query-agnostic manner without excessive redundancy. The second concerns *(2) Retrieval and Reasoning*, since unpredictable query arrival demands efficient retrieval and accurate reasoning over accumulated context under strict latency. As context grows, real-time localization of relevant spatiotemporal segments becomes harder, especially when queries lack explicit temporal anchors, and compressed representations used for efficiency may lose the fine-grained details needed for precise grounding and high-level reasoning. The third concerns *(3) Computation and Encoding*, as real-time processing constrains per-frame cost and requires balancing encoding efficiency against representation quality: lightweight designs improve speed but may degrade semantic understanding, while expressive architectures raise cost, and varying scene complexity and multimodal input rates further demand adaptive encoding and robust cross-modal alignment.

B. Future Directions

Building on the challenges discussed above, several promising directions emerge for advancing streaming video understanding. These directions aim to improve the efficiency, adaptability, and predictive capabilities of future systems under real-world streaming constraints.

1) Towards a Unified Proactive-Reactive Meta-Architecture: Existing literature fundamentally bifurcates into proactive and reactive paradigms, with most models optimized for either autonomous event triggering (proactive) or query-driven history retrieval (reactive). This dichotomy contradicts the holistic nature of human visual cognition: in deployments such as embodied AI or autonomous driving, an ideal agent must act as a unified "Online Interactor",

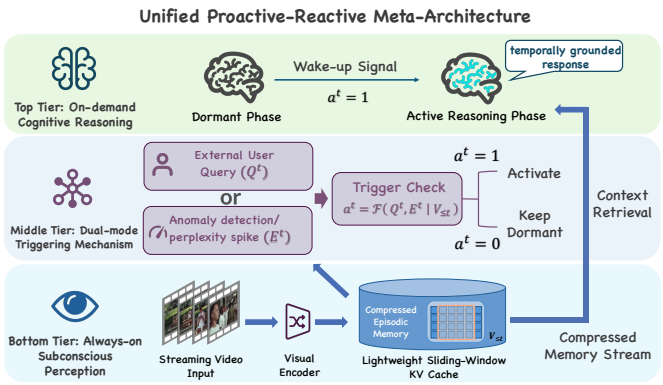

Fig. 8. Conceptual diagram of the Unified Proactive-Reactive Meta-Architecture. The system maintains a lightweight "subconscious" memory stream, which seamlessly switches between query-driven reactive reasoning and event-driven proactive triggering to activate the heavyweight cognitive LLM only when necessary.

smoothly answering queries on demand while continuously monitoring the environment to proactively warn of anomalies. To bridge this gap, we advocate a unified proactive-reactive meta-architecture that dynamically shifts between passive listening and active reasoning, conceptualized as a three-tier system:

- **Always-on Subconscious Perception (Bottom Tier):** This layer operates continuously at a high frame rate with minimal computational cost. It maintains a lightweight, sliding-window KV cache to encode streaming visual tokens without invoking the heavyweight LLM. Its primary goal is to maintain a compressed episodic memory of the unbounded stream.
- **Dual-mode Triggering Mechanism (Middle Tier):** This layer acts as the routing controller. It continuously evaluates two condition streams: an external user query stream Q^t and an internal visual change/uncertainty stream E^t. The triggering function can be formalized as $a^t = \mathcal{F}(Q^t, E^t \mid V_{\leq t})$, where $a^t = 1$ indicates that either a reactive query has arrived or a proactive threshold (e.g., anomaly detection or perplexity spike) has been crossed.
- **On-demand Cognitive Reasoning (Top Tier):** The computationally intensive multimodal LLM remains dormant until explicitly awakened by the $a^t = 1$ signal. Once triggered, the model retrieves contextually relevant tokens from the subconscious perception layer and generates a temporally grounded response $y^t \sim P(y^t \mid V_{\leq t}, Q^t, a^t = 1)$.

By decoupling continuous perception from discrete cognitive reasoning, this meta-architecture addresses the latency-accuracy trade-off: it bounds computational overhead ($O(1)$ during the subconscious phase) while preserving the capacity for complex, long-horizon multi-turn interactions. Future benchmarks and designs should prioritize this unified objective, pushing streaming models from fragmented task-specific tools toward general-purpose, always-on visual companions.

2) Efficient Memory Architectures and Hierarchical Context: The fundamental bottleneck in processing unbounded streaming video is the linear growth of the KV cache ($\mathcal{O}(T)$) against constrained physical memory. Current heuristic token pruning or sliding windows often cause catastrophic forgetting of distant events or irreversible loss of fine-grained spatiotemporal details. Future research must transcend binary "retain-or-discard" policies toward brain-inspired *hierarchical memory systems* that decouple short-term working memory, retaining uncompressed high-resolution KV tokens for immediate perception, from long-term episodic memory that abstracts historical context into compact, persistent semantic vectors.

Resolving the memory crisis also requires foundational algorithmic shifts. First, ultra-low-bit quantization for dynamic KV caching and learnable eviction policies driven by predictive uncertainty rather than static similarity remain largely underexplored. Second, backbones beyond standard Transformers, such as linear attention or State Space Models (SSMs, e.g., Mamba), are promising: they maintain a bounded, fixed-size latent state ($\mathcal{O}(1)$ memory) while digesting infinite streams, offering a structural solution to streaming scalability.

3) Hardware-Software Co-design for Edge Deployment: Algorithmic innovations such as token pruning, KV cache eviction, and sparse attention reduce theoretical FLOPs but rarely yield proportional wall-clock speedups, because the bottleneck during autoregressive decoding is rarely computation but the "Memory Wall." Dynamic token pruning and unstructured sparse attention produce highly irregular memory access patterns; on conventional GPUs or edge Neural Processing Units (NPUs), these destroy cache locality, causing frequent SRAM cache misses and heavy reliance on slow off-chip High Bandwidth Memory (HBM) or DRAM, so the theoretical speedup is offset by memory bandwidth constraints.

True real-time, low-latency inference thus requires *Hardware-Software Co-design*. On the algorithmic side, models must evolve from unstructured token dropping to hardware-aware, block-wise sparsification that maximizes SRAM utilization and guarantees contiguous memory reads. On the hardware side, general-purpose GPUs alone cannot meet the strict power and thermal envelopes of edge devices (e.g., AR/VR headsets, mobile phones, and drones), motivating application-specific accelerators for streaming multimodal LLMs; V-Rex [108] already demonstrates this potential with a hardware-level dynamic KV cache retrieval engine. Ultimately, mass deployment will require dedicated Streaming Video Processing Units (SVPUs) that natively support asymmetric multi-modal fusion, streaming causal attention, and highly parallelized KV cache updates, achieving microsecond-level Time-to-First-Token (TTFT) within stringent edge-power budgets.

4) Streaming Video as the Visual Cortex of Embodied AI: Current methodologies and benchmarks predominantly emphasize multi-turn QA or proactive narration, treating the streaming model as a passive bystander. Yet its ultimate frontier lies in deep integration with Embodied Artificial Intelligence (EAI): from autonomous vehicles to humanoid robots, the streaming model must transition from a passive narrator to an active participant serving as the real-time "visual cortex" of the agent. This necessitates a shift from Vision-Language Models (VLMs) to Vision-Language-Action (VLA) models under strict streaming constraints, directly outputting

continuous motor commands or discrete action tokens from the unfolding stream. Here temporal challenges are magnified: a latency spike that merely causes an awkward conversational pause may cause catastrophic physical collisions [147], [148], so future models must couple real-time efficiency with *anticipatory reasoning*, predicting near-future physical states to compensate for mechanical actuation delays.

Embodied streaming is further a closed-loop system characterized by *active perception*: an agent's actions directly alter its subsequent visual input (e.g., turning its head changes the field of view). Future research must investigate how streaming memory mechanisms such as KV cache retention and eviction can robustly handle rapid egocentric viewpoint shifts and self-induced occlusions. Consequently, the community must pivot toward closed-loop evaluation, evolving from static semantic metrics (e.g., QA accuracy, F1, or BLEU) to physical task success rates, collision avoidance, and long-horizon survival metrics within continuous, unsegmented simulation environments.

VII. Conclusion

In this survey, we provide a comprehensive overview of streaming video understanding under causal and real-time constraints. We organize existing methods into two complementary paradigms, proactive modeling and reactive modeling, offering a unified perspective on when to respond and how to maintain long-term understanding in streaming settings. We also summarize recent benchmarks for evaluating streaming capabilities. Despite recent progress, several challenges remain, including long-horizon temporal reasoning, efficient memory management, and the lack of fully proactive evaluation protocols. Future research is expected to develop more adaptive and scalable systems that support continuous perception, prediction, and interaction in dynamic environments.

References

[1] R. Zellers, J. Lu, X. Lu, Y. Yu, Y. Zhao, M. Salehi, A. Kusupati, J. Hessel, A. Farhadi, and Y. Choi, "Merlot reserve: Neural script knowledge through vision and language and sound," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 375–16 387. I

[2] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 995–19 012. I

[3] Y. Tang, J. Bi, S. Xu, L. Song, S. Liang, T. Wang, D. Zhang, J. An, J. Lin, R. Zhu *et al.*, "Video understanding with large language models: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, 2025. I, I, I

[4] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742. I

[5] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *Advances in neural information processing systems*, vol. 36, pp. 34 892–34 916, 2023. I

[6] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023. I

[7] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024. I

[8] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge *et al.*, "Qwen3-vl technical report," *arXiv preprint arXiv:2511.21631*, 2025. I

[9] W. Wang, Z. Gao, L. Gu, H. Pu, L. Cui, X. Wei, Z. Liu, L. Jing, S. Ye, J. Shao *et al.*, "Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency," *arXiv preprint arXiv:2508.18265*, 2025. I

[10] J. Lee, J. Chang, D. Lee, and J. Choi, "ca^{2}st: Cross-attention in audio, space, and time for holistic video recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 3, pp. 2803–2819, 2026. I

[11] Q. Ye, Z. Yu, R. Shao, Y. Cui, X. Kang, X. Liu, P. Torr, and X. Cao, "Cat+: Investigating and enhancing audio-visual understanding in large language models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 10, pp. 8674–8690, 2025. I

[12] L.-H. Chen, S. Lu, A. Zeng, H. Zhang, B. Wang, R. Zhang, and L. Zhang, "Motionllm: Understanding human behaviors from human motions and videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–15, 2025. I

[13] E. Song, W. Chai, T. Ye, J.-N. Hwang, X. Li, and G. Wang, "Moviechat+: Question-aware sparse memory for long video question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 1, pp. 374–389, 2026. I

[14] J. Li, M. Gao, X. He, S. Tang, W.-S. Zheng, J. Xiao, M. Wang, T.-S. Chua, and Y. Zhuang, "Momentor++: Advancing video large language models with fine-grained long video reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 6, pp. 6208–6224, 2026. I

[15] K. Zhang, Z. Yang, M. Han, Y. Zhuge, H. Hao, C. Li, Z. Li, and X. Chang, "Selongvlm: Empowering long video language models with self-corrective clip selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–16, 2026. I

[16] S. Tian, R. Wang, H. Guo, P. Wu, Y. Dong, X. Wang, J. Yang, H. Zhang, H. Zhu, and Z. Liu, "Ego-r1: Agentic chain-of-tool-thought for ultra-long egocentric video reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–16, 2026. I

[17] S. A. Peirone, F. Pistilli, A. Alliegro, T. Tommasi, and G. Averta, "Hier-egopack: Hierarchical egocentric video understanding with diverse task perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 2, pp. 1917–1931, 2026. I

[18] Y. Zhang, J. Wu, W. Li, B. Li, Z. Ma, Z. Liu, and C. Li, "Llava-video: Video instruction tuning with synthetic data," *arXiv preprint arXiv:2410.02713*, 2024. I

[19] B. Lin, Y. Ye, B. Zhu, J. Cui, M. Ning, P. Jin, and L. Yuan, "Video-llava: Learning united visual representation by alignment before projection," in *Proceedings of the 2024 conference on empirical methods in natural language processing*, 2024, pp. 5971–5984. I

[20] M. Maaz, H. Rasheed, S. Khan, and F. 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)*, 2024, pp. 12 585–12 602. I

[21] K. Li, Y. He, Y. Wang, Y. Li, W. Wang, P. Luo, Y. Wang, L. Wang, and Y. Qiao, "Videochat: Chat-centric video understanding," *Science China Information Sciences*, vol. 68, no. 10, p. 200102, 2025. I

[22] Y. Wang, X. Li, Z. Yan, Y. He, J. Yu, X. Zeng, C. Wang, C. Ma, H. Huang, J. Gao *et al.*, "Internvideo2. 5: Empowering video mllms with long and rich context modeling," *arXiv preprint arXiv:2501.12386*, 2025. I

[23] H. Shao, Y. Hu, L. Wang, G. Song, S. L. Waslander, Y. Liu, and H. Li, "Lmdrive: Closed-loop end-to-end driving with large language models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 15 120–15 130. I

[24] J. Yang, S. Liu, H. Guo, Y. Dong, X. Zhang, S. Zhang, P. Wang, Z. Zhou, B. Xie, Z. Wang *et al.*, "Egolife: Towards egocentric life assistant," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 28 885–28 900. I

[25] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model, 2024," *URL https://arxiv. org/abs/2406.09246*, vol. 1, no. 2, p. 4, 2024. I

[26] J. Chen, Z. Lv, S. Wu, K. Q. Lin, C. Song, D. Gao, J.-W. Liu, Z. Gao, D. Mao, and M. Z. Shou, "Videollm-online: Online video large language model for streaming video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 407–18 418. I, I, III-A1, III-A2, III-C1, II

[27] S. Fu, Q. Yang, Y.-M. Li, Y.-X. Peng, K.-Y. Lin, X. Wei, J.-F. Hu, X. Xie, and W.-S. Zheng, "Vispeak: Visual instruction feedback in streaming videos," *arXiv preprint arXiv:2503.12769*, 2025. I, III-B1, III-B1, II

[28] A. Yang, A. Nagrani, P. H. Seo, A. Miech, J. Pont-Tuset, I. Laptev, J. Sivic, and C. Schmid, "Vid2seq: Large-scale pretraining of a visual language model for dense video captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 10 714–10 726. I

[29] K. Ataallah, X. Shen, E. Abdelrahman, E. Sleiman, D. Zhu, J. Ding, and M. Elhoseiny, "Minigpt4-video: Advancing multimodal llms for video understanding with interleaved visual-textual tokens," *arXiv preprint arXiv:2404.03413*, 2024. I

[30] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan, "Video-chatgpt: Towards detailed video understanding via large vision and language models," *arXiv preprint arXiv:2306.05424*, 2023. I

[31] K. Li, Y. He, Y. Wang, Y. Li, W. Wang, P. Luo, Y. Wang, L. Wang, and Y. Qiao, "Videochat: Chat-centric video understanding," *arXiv preprint arXiv:2305.06355*, 2023. I

[32] Y. Wang, K. Li, Y. Li, Y. He, B. Huang, Z. Zhao, H. Zhang, J. Xu, Y. Liu, Z. Wang *et al.*, "Internvideo: General video foundation models via generative and discriminative learning," *arXiv preprint arXiv:2212.03191*, 2022. I

[33] Z. Cheng, S. Leng, H. Zhang, Y. Xin, X. Li, G. Chen, Y. Zhu, W. Zhang, Z. Luo, D. Zhao *et al.*, "Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms," *arXiv preprint arXiv:2406.07476*, 2024. I

[34] Z. Liu, Y. Dong, Z. Liu, W. Hu, J. Lu, and Y. Rao, "Oryx mllm: On-demand spatial-temporal understanding at arbitrary resolution," *arXiv preprint arXiv:2409.12961*, 2024. I

[35] S. Ren, L. Yao, S. Li, X. Sun, and L. Hou, "Timechat: A time-sensitive multimodal large language model for long video understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 313–14 323. I

[36] H. Zhang, Y. Wang, Y. Tang, Y. Liu, J. Feng, J. Dai, and X. Jin, "Flash-vstream: Memory-based real-time understanding for long video streams," *arXiv preprint arXiv:2406.08085*, 2024. I

[37] E. Song, W. Chai, T. Ye, J.-N. Hwang, X. Li, and G. Wang, "Moviechat+: Question-aware sparse memory for long video question answering," *arXiv preprint arXiv:2404.17176*, 2024. I

[38] B. He, H. Li, Y. K. Jang, M. Jia, X. Cao, A. Shah, A. Shrivastava, and S.-N. Lim, "Ma-lmm: Memory-augmented large multimodal model for long-term video understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 13 504–13 514. I

[39] X. Wang, D. Song, S. Chen, C. Zhang, and B. Wang, "Longllava: Scaling multi-modal llms to 1000 images efficiently via hybrid architecture," 2024. [Online]. Available: https://arxiv.org/abs/2409.02889 I

[40] F. Xue, Y. Chen, D. Li, Q. Hu, L. Zhu, X. Li, Y. Fang, H. Tang, S. Yang, Z. Liu, Y. He, H. Yin, P. Molchanov, J. Kautz, L. Fan, Y. Zhu, Y. Lu, and S. Han, "Longvila: Scaling long-context visual language models for long videos," *null*, 2024. I

[41] P. Zhang, K. Zhang, B. Li, G. Zeng, J. Yang, Y. Zhang, Z. Wang, H. Tan, C. Li, and Z. Liu, "Long context transfer from language to vision," *arXiv preprint arXiv:2406.16852*, 2024. I

[42] F. Li, R. Zhang, H. Zhang, Y. Zhang, B. Li, W. Li, Z. Ma, and C. Li, "Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models," *arXiv preprint arXiv:2407.07895*, 2024. I

[43] Q. Li, Z. Chen, W. Wang, W. Wang, S. Ye, Z. Jin, G. Chen, Y. He, Z. Gao, E. Cui *et al.*, "Omnicorpus: A unified multimodal corpus of 10 billion-level images interleaved with text," *arXiv preprint arXiv:2406.08418*, 2024. I

[44] R. Qian, X. Dong, P. Zhang, Y. Zang, S. Ding, D. Lin, and J. Wang, "Streaming long video understanding with large language models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 119 336–119 360, 2024. I, II

[45] X. Zhou, A. Arnab, S. Buch, S. Yan, A. Myers, X. Xiong, A. Nagrani, and C. Schmid, "Streaming dense video captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 243–18 252. I, II

[46] J. Gao, Y. Lian, Z. Zhou, Y. Fu, and B. Wang, "Livechat: A large-scale personalized dialogue dataset automatically constructed from live streaming," *arXiv preprint arXiv:2306.08401*, 2023. I

[47] Y. Wang, X. Meng, Y. Wang, J. Liang, J. Wei, H. Zhang, and D. Zhao, "Videollm knows when to speak: Enhancing time-sensitive video comprehension with video-text duet interaction format," *arXiv preprint arXiv:2411.17991*, 2024. I, III-B1, II, III, V-C

[48] Z. Yang, K. Zhang, Y. Hu, B. Wang, S. Qian, B. Wen, F. Yang, T. Gao, W. Dong, and C. Xu, "Livestar: Live streaming assistant for real-world online video understanding," *Advances in Neural Information Processing Systems*, vol. 38, pp. 31 266–31 304, 2026. I, III-C1, II, III, V-B

[49] S. Di, Z. Yu, G. Zhang, H. Li, T. Zhong, H. Cheng, B. Li, W. He, F. Shu, and H. Jiang, "Streaming video question-answering with in-context video kv-cache retrieval," *arXiv preprint arXiv:2503.00540*, 2025. I, II, IV-D1

[50] Z. Huang, X. Li, J. Li, J. Wang, X. Zeng, C. Liang, T. Wu, X. Chen, L. Li, and L. Wang, "Online video understanding: Ovbench and videochat-online," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 3328–3338. I, II, IV-C1, III, V-A

[51] H. Zhang, Y. Wang, Y. Tang, Y. Liu, J. Feng, and X. Jin, "Flash-vstream: Efficient real-time understanding for long video streams," *arXiv preprint arXiv:2506.23825*, 2025. I, II

[52] L. Hong, Z. Liu, W. Chen, C. Tan, Y. Feng, X. Zhou, P. Guo, J. Li, Z. Chen, S. Gao, W. Zhang, and W. Zhang, "Lvos: A benchmark for large-scale long-term video object segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 1, pp. 946–961, 2026. I

[53] S. Yang, W. Yu, W. Yang, X. Liu, H. Tan, L. Lan, and N. Xiao, "Wildvideo: Benchmarking lmms for understanding video-language interaction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 10, pp. 9330–9344, 2025. I

[54] J. Wu, W. Liu, Y. Liu, M. Liu, L. Nie, Z. Lin, and C. W. Chen, "A survey on video temporal grounding with multimodal large language model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 2, pp. 1521–1541, 2026. I, I

[55] J. Lin, Z. Fang, C. Chen, Z. Wan, F. Luo, P. Li, Y. Liu, and M. Sun, "Streamingbench: Assessing the gap for mllms to achieve streaming video understanding," *arXiv preprint arXiv:2411.03628*, 2024. I, III, V-A

[56] Y. Li, J. Niu, Z. Miao, C. Ge, Y. Zhou, Q. He, X. Dong, H. Duan, S. Ding, R. Qian *et al.*, "Ovo-bench: How far is your video-llms from real-world online video understanding?" *URL https://arxiv. org/abs/2501.05510*, 2025. I, III, V-C

[57] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Learning to answer visual questions from web videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 5, pp. 3202–3218, 2025. I

[58] Y. Li, X. Wang, J. Xiao, W. Ji, and T.-S. Chua, "Transformer-empowered invariant grounding for video question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 11, pp. 9510–9522, 2025. I

[59] J. Li, P. Wei, W. Han, S.-C. Zhu, and L. Fan, "Intentqa: Intent question answering in videos by cognitive context reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–18, 2026. I

[60] J. Li, Z. Liao, F. Xiao, T. Li, Q. Zhang, H. Zhao, L. Niu, G. Chen, L. Zhang, and C. Jiang, "Parse, align and aggregate: Graph-driven compositional reasoning for video question answering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 5, pp. 5586–5603, 2026. I

[61] T. Chen, H. Liu, Y. Wang, Y. Chen, T. He, C. Gan, H. He, and W. Lin, "Mecd+: Unlocking event-level causal graph discovery for video reasoning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 3, pp. 2628–2645, 2026. I

[62] L.-L. Li, J. Fang, J. Xiao, H. Yu, C. Lv, J. Xue, Z. Li, and T.-S. Chua, "Adversa: Abductive driving accident video understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 48, no. 6, pp. 6980–6998, 2026. I

[63] N.-N. Dao, A.-T. Tran, N. H. Tu, T. T. Thanh, V. N. Q. Bao, and S. Cho, "A contemporary survey on live video streaming from a computation-driven perspective," *ACM Computing Surveys*, vol. 54, no. 10s, pp. 1–38, 2022. I, I

[64] A. A. Laghari, S. Shahid, R. Yadav, S. Karim, A. Khan, H. Li, and Y. Shoulin, "The state of art and review on video streaming," *Journal of High Speed Networks*, vol. 29, no. 3, pp. 211–236, 2023. I, I

[65] T. Nguyen, Y. Bin, J. Xiao, L. Qu, Y. Li, J. Z. Wu, C.-D. Nguyen, S. K. Ng, and L. A. Tuan, "Video-language understanding: A survey from model architecture, model training, and data perspectives," in *Findings of the Association for Computational Linguistics: ACL 2024*, 2024, pp. 3636–3657. I, I

[66] S. Wu, J. Chen, K. Q. Lin, Q. Wang, Y. Gao, Q. Xu, T. Xu, Y. Hu, E. Chen, and M. Z. Shou, "Videollm-mod: Efficient video-language streaming with mixture-of-depths vision computation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 109 922–109 947, 2024. III-A1, III-C1, II

[67] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity," *Journal of Machine Learning Research*, vol. 23, no. 120, pp. 1–39, 2022. III-A1

[68] W. Li, B. Hu, R. Shao, L. Shen, and L. Nie, "Lion-fs: Fast & slow video-language thinker as online video assistant," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 3240–3251. III-A1, III-C1, II

[69] S. Panchal, A. Bhattacharyya, G. Berger, A. Mercier, C. Böhm, F. Dietrichkeit, R. Pourreza, X. Li, P. Madan, M. Lee *et al.*, "What to say and when to say it: Live fitness coaching as a testbed for situated interaction," *Advances in Neural Information Processing Systems*, vol. 37, pp. 75 853–75 882, 2024. III-A1, II, III, V-B

[70] Y. Zhang, C. Shi, Y. Wang, and S. Yang, "Eyes wide open: Ego proactive video-llm for streaming video," *arXiv preprint arXiv:2510.14560*, 2025. III-A1, II, III, V-C

[71] J. Xia, P. Chen, M. Zhang, X. Sun, and K. Zhou, "Streaming video instruction tuning," *arXiv preprint arXiv:2512.21334*, 2025. III-A1, II

[72] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988. III-A1

[73] Z. Liu, L. Guo, H. Li, R. Zhen, X. He, R. Ji, X. Ren, Y. Zhang, H. Lu, and J. Liu, "Thinking in streaming video," *arXiv preprint arXiv:2603.12938*, 2026. III-A1, II

[74] J. Chen, Z. Chen, C. Du, M. He, W. He, H. Li, Q. Li, Z. Liu, H. Ma, X. Pan *et al.*, "Streamingclaw technical report," *arXiv preprint arXiv:2603.22120*, 2026. III-A1, II

[75] J. Chen, Z. Zeng, Y. Lin, W. Li, Z. Ma, and M. Z. Shou, "Livecc: Learning video llm with streaming speech transcription at scale," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 29 083–29 095. III-A2, II, III, V-B

[76] Y. Zhang, X. L. Dong, Z. Lin, A. Madotto, A. Kumar, B. Damavandi, J. Chai, and S. Moon, "Proactive assistant dialogue generation from streaming egocentric videos," *arXiv preprint arXiv:2506.05904*, 2025. III-A2, II, III, V-C

[77] Z. Yang, C. Gao, J. Liu, P. Wu, G. Pang, and M. Z. Shou, "Assistpda: An online video surveillance assistant for video anomaly prediction, detection, and analysis," *arXiv preprint arXiv:2503.21904*, 2025. III-A2, II, III, V-C

[78] J. Lin, J. Tong, H. Wu, J. Zhang, J. Liu, X. Jin, and X. Shen, "Speak while watching: Unleashing true real-time video understanding capability of multimodal large language models," *arXiv preprint arXiv:2601.06843*, 2026. III-B1

[79] J. Qian, H. Du, G. Nan, S. Huang, J. Yu, H. Wang, J. Chen, M. Cai, M. Yang, J. Li *et al.*, "Learning to respond: A large-scale benchmark and progressive learning framework for trigger-centric online video understanding." III-B1, II

[80] J. Kim, M.-S. Kim, J. Chung, J. Cho, J. Kim, S. Kim, G. Sim, and Y. Yu, "Egospeak: learning when to speak for egocentric conversational agents in the wild," in *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025, pp. 2990–3005. III-B1, II, III, V-A

[81] S. Azad, V. Vineet, and Y. S. Rawat, "Streamready: Learning what to answer and when in long streaming videos," *arXiv preprint arXiv:2603.08620*, 2026. III-B1, II, III, V-C

[82] W. Yan, Y. Dai, Q. Ran, H. Li, W. Lin, H. Liao, X. Xie, T. Jin, and J. Lian, "Proact-vl: A proactive videollm for real-time ai companions," *arXiv preprint arXiv:2603.03447*, 2026. III-B1, II, III, V-C

[83] X. Tian, W. Li, B. Xu, H. Dong, Y. Wang, and H. Shen, "Roma: Real-time omni-multimodal assistant with interactive streaming understanding," *arXiv preprint arXiv:2601.10323*, 2026. III-B1, II

[84] H. Wang, B. Feng, Z. Lai, M. Xu, S. Li, W. Ge, A. Dehghan, M. Cao, and P. Huang, "Streambridge: Turning your offline video large language model into

a proactive streaming assistant," *arXiv preprint arXiv:2505.05467*, 2025. III-B1, II

[85] R. Qian, S. Ding, X. Dong, P. Zhang, Y. Zang, Y. Cao, D. Lin, and J. 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*, 2025, pp. 24 045–24 055. III-B1, II

[86] X. Ding, H. Wu, Y. Yang, S. Jiang, Q. Zhang, D. Bai, Z. Chen, and T. Cao, "Streammind: Unlocking full frame rate streaming video dialogue through event-gated cognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 13 448–13 459. III-B1, II

[87] J. Kim, H. Lee, J. M. Rehg, M. Kim, and Y. M. Ro, "Stride: When to speak meets sequence denoising for streaming video understanding," *arXiv preprint arXiv:2603.27593*, 2026. III-B1, II

[88] H. Kang, Y. Park, Y. Yoo, Y. Choi, and S. J. Kim, "Open-ended hierarchical streaming video understanding with vision language models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 20 715–20 725. III-B1, II

[89] J. Mun, L. Yang, Z. Ren, N. Xu, and B. Han, "Streamlined dense video captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6588–6597. III-B1

[90] H. Yang, F. Tang, L. Zhao, X. An, M. Hu, H. Li, X. Zhuang, Y. Lu, X. Zhang, A. Swikir *et al.*, "Streamagent: Towards anticipatory agents for streaming video understanding," *arXiv preprint arXiv:2508.01875*, 2025. III-B2, II

[91] Y. Zheng, X. Ding, Y. Yang, S. Jiang, H. Wu, Q. Zhang, W. Wang, T. Cao, and Y. Liu, "Em-garde: A propose-match framework for proactive streaming video understanding," *arXiv preprint arXiv:2603.19054*, 2026. III-B2, II

[92] D. J. Simons and R. A. Rensink, "Change blindness: Past, present, and future," *Trends in cognitive sciences*, vol. 9, no. 1, pp. 16–20, 2005. III-D1

[93] L. Yao, Y. Li, Y. Wei, L. Li, S. Ren, Y. Liu, K. Ouyang, L. Wang, S. Li, S. Li *et al.*, "Timechat-online: 80% visual tokens are naturally redundant in streaming videos," in *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025, pp. 10 807–10 816. III-D1, III-D1, II

[94] K. Zhang, Z. Yang, B. Wang, S. Qian, and C. Xu, "Querystream: Advancing streaming video understanding with query-aware pruning and proactive response," in *The Fourteenth International Conference on Learning Representations*, 2026. III-D1, II

[95] W. Cai, H. Zhang, Y. Huang, S. Sun, J. Deng, S. Xu, J. Song, and Z. Zhang, "Color when it counts: Grayscale-guided online triggering for always-on streaming video sensing," *arXiv preprint arXiv:2603.22466*, 2026. III-D1, II

[96] Y. Guan, L. Yin, D. Liang, J. Ju, Z. Luo, J. Luan, Y. Liu, and X. Bai, "Video streaming thinking: Videollms can watch and think simultaneously," *arXiv preprint arXiv:2603.12262*, 2026. II, IV-B2

[97] J. Zhang, J. Tong, J. Lin, H. Wu, Y. Sun, Y. Ma, and X. Shen, "Think-as-you-see: Streaming chain-of-thought reasoning for large vision-language models," *arXiv preprint arXiv:2603.02872*, 2026. II, IV-B2

[98] Y. Xie, B. He, J. Wang, X. Zheng, Z. Ye, and Z. Wu, "Fluxmem: Adaptive hierarchical memory for streaming video understanding," *arXiv preprint arXiv:2603.02096*, 2026. II

[99] L. Wang, Z. Jin, Y. Hao, Y. Chen, K. Liu, Y. Ao, and J. Zhao, "Think while watching: Online streaming segment-level memory for multi-turn video reasoning in multimodal large language models," *arXiv preprint arXiv:2603.11896*, 2026. II, IV-C2

[100] Y. Yan, J. Xu, S. Di, H. Wu, and W. Xie, "Omnistream: Mastering perception, reconstruction and action in continuous streams," *arXiv preprint arXiv:2603.12265*, 2026. II, IV-A2

[101] B. Shi, S. Fu, L. Lian, H. Ye, D. Eigen, A. Reite, B. Li, J. Kautz, S. Han, D. M. Chan *et al.*, "Attend before attention: Efficient and scalable video understanding via autoregressive gazing," *arXiv preprint arXiv:2603.12254*, 2026. II, IV-A1

[102] S. Wen, Z. Wang, X. Zhang, L. Huang, and W. Wu, "Eventmemagent: Hierarchical event-centric memory for online video understanding with adaptive tool use," *arXiv preprint arXiv:2602.15329*, 2026. II, IV-C1

[103] Y. Zhang, C. Shi, and S. Yang, "Weavetime: Stream from earlier frames into emergent memory in videollms," *arXiv preprint arXiv:2602.22142*, 2026. II, IV-D2

[104] H. Zhang, S. Yang, J. Fu, S.-K. Ng, and X. Qiu, "Hermes: Kv cache as hierarchical memory for efficient streaming video understanding," *arXiv preprint arXiv:2601.14724*, 2026. II, IV-C2

[105] Y. Wang, X. Liu, X. Gui, X. Lin, B. Yang, C. Liao, T. Chen, and L. Zhang, "Accelerating streaming video large language models via hierarchical token compression," *arXiv preprint arXiv:2512.00891*, 2025. II, IV-A1

[106] N. Zheng, J. Huang, Q. Guo, and F. Zhao, "Videoscaffold: Elastic-scale visual hierarchies for streaming video understanding in mllms," *arXiv preprint arXiv:2512.22226*, 2025. II

[107] X. Jin *et al.*, "Streamingassistant: Efficient visual token pruning," *arXiv preprint arXiv:2512.12560*, 2025. II, IV-B1

[108] D. Kim, S. Yang, W. Shin, and J.-Y. Kim, "V-rex: Real-time streaming video llm acceleration via dynamic kv cache retrieval," *arXiv preprint arXiv:2512.12284*, 2025. II, IV-D2, VI-B3

[109] S. Ye, B. Ouyang, T. Qian, L. Zeng, M. Yuan, X. Chu, W. Hong, and X. Chen, "Venus: An efficient edge memory-and-retrieval system for vlm-based online video understanding," *arXiv preprint arXiv:2512.07344*, 2025. II, IV-D2

[110] Y. Wang, S. Liu, D. Wang, N. Xu, G. Wan, H. Zhang, and D. Zhao, "Mmduet2: Enhancing proactive interaction of video mllms with multi-turn reinforcement learning," *arXiv preprint arXiv:2512.06810*, 2025. II

[111] S. Patel and D. Patel, "Cacheflow: Compressive streaming memory for efficient long-form video understanding," *arXiv preprint arXiv:2511.13644*, 2025. II, IV-D2

[112] Y. Chen, X. Bai, Z. Wang, C. Bai, Y. Dai, M. Lu, and S. Zhang, "Streamkv: Streaming video question-answering with segment-based kv cache retrieval and compression," *arXiv preprint arXiv:2511.07278*, 2025. II, IV-D1

[113] R. Xu *et al.*, "Streamingvlm: Real-time understanding for infinite video streams," *arXiv preprint arXiv:2510.09608*, 2025. II, IV-B2

[114] X. Chen, K. Tao, K. Shao, and H. Wang, "Streamingtom: Streaming token compression for efficient video understanding," *arXiv preprint arXiv:2510.18269*, 2025. II, IV-D1

[115] V. Dorovatas, S. Seifi, G. Gupta, and R. Aljundi, "Recurrent attention-based token selection for efficient streaming video-llms," *arXiv preprint arXiv:2510.17364*, 2025. II, IV-D2

[116] G. Sun, Y. Li, X. Wu, Y. Yang, W. Li, Z. Ma, and C. Zhang, "video-salmonn s: Streaming audio-visual llms beyond length limits via memory," *arXiv preprint arXiv:2510.11129*, 2025. II, IV-C2

[117] X. Zeng, K. Qiu, Q. Zhang, X. Li, J. Wang, J. Li, Z. Yan, K. Tian, M. Tian, X. Zhao *et al.*, "Streamforest: Efficient online video understanding with persistent event memory," *arXiv preprint arXiv:2509.24871*, 2025. II, IV-C1, III, V-A

[118] Y. Yang *et al.*, "Streammem: Query-agnostic kv cache memory," *arXiv preprint arXiv:2502.XXXXX*, 2025. II, IV-B1

[119] R. Zeng, J. Mao, M. Lai, M. H. Phan, Y. Dong, W. Wang, Q. Chen, and X. Hu, "Ovg-hq: Online video grounding with hybrid-modal queries," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 21 085–21 096. II, IV-C2

[120] M. Wei *et al.*, "Streamvln: Streaming vision-and-language navigation," *arXiv preprint arXiv:2507.05240*, 2025. II

[121] M. Kim *et al.*, "Infinipot-v: Memory-constrained kv cache compression," *arXiv preprint arXiv:2506.15745*, 2025. II, IV-B1

[122] Z. Zhao, K. Wang, S. Li, R. Qian, W. Lin, and H. Liu, "Cogstream: Context-guided streaming video question answering," *arXiv preprint arXiv:2506.10516*, 2025. II

[123] Z. Ning, G. Liu, Q. Jin, W. Ding, M. Guo, and J. Zhao, "Livevlm: Efficient online video understanding via streaming-oriented kv cache and retrieval," *arXiv preprint arXiv:2505.15269*, 2025. II, IV-D1

[124] Y. Yan, J. Xu, S. Di, Y. Liu, Y. Shi, Q. Chen, Z. Li, Y. Huang, and W. Xie, "Learning streaming video representation via multitask training," *arXiv preprint arXiv:2504.20041*, 2025. II, IV-A2

[125] D. Chatterjee, E. Remelli, Y. Song, B. Tekin, A. Mittal, B. Bhatnagar, N. C. CamgÃkz, S. Hampali, E. Sauser, S. Ma *et al.*, "Memory-efficient streaming videollms for real-time procedural video understanding," *arXiv preprint arXiv:2504.13915*, 2025. II, IV-C2

[126] R. Li, Y. Tan, Y. Shi, and J. Shao, "Videoscan: Enabling efficient streaming video understanding via frame-level semantic carriers," *arXiv preprint arXiv:2503.09387*, 2025. II

[127] Z. Yang, Y. Hu, Z. Du, D. Xue, S. Qian, J. Wu, F. Yang, W. Dong, and C. Xu, "Svbench: A benchmark with temporal multi-turn dialogues for streaming video understanding," in *The Thirteenth International Conference on Learning Representations*. II, III, V-A

[128] H. Xiong, Z. Yang, J. Yu, Y. Zhuge, L. Zhang, J. Zhu, and H. Lu, "Streaming video understanding and multi-round interaction with memory-enhanced knowledge," *arXiv preprint arXiv:2501.13468*, 2025. II, IV-C1, III, V-A

[129] C. Fu, H. Lin, X. Wang, Y.-F. Zhang, Y. Shen, X. Liu, H. Cao, Z. Long, H. Gao, K. Li *et al.*, "Vita-1.5: Towards gpt-4o level real-time vision and speech interaction," *arXiv preprint arXiv:2501.01957*, 2025. II

[130] J. Liu, Z. Yu, S. Lan, S. Wang, R. Fang, J. Kautz, H. Li, and J. M. Alvare, "Streamchat: Chatting with streaming video," *arXiv preprint arXiv:2412.08646*, 2024. II, IV-A2

[131] C. Eyzaguirre, E. Tang, S. Buch, A. Gaidon, J. Wu, and J. C. Niebles, "Streaming detection of queried event start," *Advances in Neural Information Processing Systems*, vol. 37, pp. 100 698–100 733, 2024. II, IV-A2

[132] Y. Wang, Y. Song, C. Xie, Y. Liu, and Z. Zheng, "Videollamb: Long streaming video understanding with recurrent memory bridges," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 24 170–24 181. II

[133] D. Yang, C. Zhan, Z. Wang, B. Wang, T. Ge, B. Zheng, and Q. Jin, "Synchronized video storytelling: Generating video narrations with structured storyline," *arXiv preprint arXiv:2405.14040*, 2024. II

[134] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in neural information processing systems*, vol. 33, pp. 9459–9474, 2020. IV-D1

[135] Z. Yang, D. Xue, S. Qian, W. Dong, and C. Xu, "Ldre: Llm-based divergent reasoning and ensemble for zero-shot composed image retrieval," in *Proceedings of the 47th International ACM SIGIR conference on research and development in information retrieval*, 2024, pp. 80–90. IV-D1

[136] Z. Yang, S. Qian, D. Xue, J. Wu, F. Yang, W. Dong, and C. Xu, "Semantic editing increment benefits zero-shot composed image retrieval," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 1245–1254. IV-D1

[137] Y. Shi, Q. Zhao, T. Jiang, X. Zeng, Y. Wang, and L. Wang, "River: A real-time interaction benchmark for video llms," *arXiv preprint arXiv:2603.03985*, 2026. III, V-A

[138] Y. Wang, Z. Li, T. Qian, H. Zheng, Z. Wang, Y. Fu, and X. Wang, "Streameqa: Towards streaming video understanding for embodied scenarios," *arXiv preprint arXiv:2512.04451*, 2025. III, V-A

[139] D. Lee, S. Mukherjee, B. Kveton, R. A. Rossi, V. D. Lai, S. Yoon, T. Bui, F. Dernoncourt, and M. Bansal, "Streamgaze: Gaze-guided temporal reasoning and proactive understanding in streaming videos," *arXiv preprint arXiv:2512.01707*, 2025. III, V-C

[140] Y. Hu, Z. Yang, S. Wang, S. Qian, B. Wen, F. Yang, T. Gao, and C. Xu, "Streamingcot: A dataset for temporal dynamics and multimodal chain-of-thought reasoning in streaming videoqa," in *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025, pp. 13 464–13 470. III, V-A

[141] J. Lin, C. Zhu, R. Xu, X. Mao, X. Liu, T. Wang, and J. Pang, "Ost-bench: Evaluating the capabilities of mllms in online spatio-temporal scene understanding," *arXiv preprint arXiv:2507.07984*, 2025. III, V-A

[142] Y. Wang, X. Meng, Y. Wang, H. Zhang, and D. Zhao, "Proactivevideoqa: A comprehensive benchmark evaluating proactive interactions in video large language models," *arXiv preprint arXiv:2507.09313*, 2025. III, V-C

[143] S. Xun, S. Tao, J. Li, Y. Shi, Z. Lin, Z. Zhu, Y. Yan, H. Li, L. Zhang, S. Wang *et al.*, "Rtv-bench: Benchmarking mllm continuous perception, understanding and reasoning through real-time video," *arXiv preprint arXiv:2505.02064*, 2025. III, V-A

[144] Y. Wang, Y. Wang, B. Chen, T. Wu, D. Zhao, and Z. Zheng, "Omnimmi: A comprehensive multi-modal interaction benchmark in streaming video contexts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025, pp. 18 925–18 935. III, V-C

[145] M. Cai, R. Tan, J. Zhang, B. Zou, K. Zhang, F. Yao, F. Zhu, J. Gu, Y. Zhong, Y. Shang *et al.*, "Temporalbench: Benchmarking fine-grained temporal understanding for multimodal video models," *arXiv preprint arXiv:2410.10818*, 2024. III, V-A

[146] Z. Yang, Z. Du, S. Qian, and C. Xu, "Never seen before: Benchmarking genuine zero-shot composed image retrieval with consistent video-sourced datasets," *arXiv preprint arXiv:2606.07032*, 2026. V-B

[147] Z. Yang, K. Zhang, S. Qian, W. Dong, and C. Xu, "Don't pause: Streaming video-language synchrony for online video understanding," *arXiv preprint arXiv:2606.06991*, 2026. VI-B4

[148] Z. Yang, K. Zhang, B. Wang, S. Qian, and C. Xu, "Livestarpro: Proactive streaming video understanding with hierarchical memory for long-horizon streams," *arXiv preprint arXiv:2606.17798*, 2026. VI-B4