Resource-Constrained Video-LLMs: From Single-Node Optimization to Collaborative Orchestration

Abhishek Dharmaratnakar, Srivaths Ranganathan, Anushree Sinha, Debanshu Das
Google LLC
Mountain View, CA, USA

***Abstract*—Vision-Language Models (VLMs) and Video Large Language Models (Video-LLMs) extend foundation-model capabilities into multimodal understanding, with applications in augmented reality, robotics, and industrial monitoring. Deploying these models on commercial-off-the-shelf (COTS) edge devices remains difficult: edge platforms operate under strict Space, Weight, and Power (SWaP) and thermal budgets that preclude cloud-style inference. Video input compounds the problem self-attention scales as $O(n^2)$ in compute, the Key-Value (KV) cache grows linearly during decoding, and continuous frame streams quickly exceed the context windows and memory bandwidth available at the edge.**

This paper analyzes recent advancements in Video-LLM optimization and argues for a paradigm shift from single-node device-centric optimization toward a distributed "Cluster-as-a-Camera" (CaaC) orchestration. We categorize existing work across four directions: (i) architectural alternatives to dense attention, including streaming encoders and State-Space Models such as Mamba; (ii) spatiotemporal token reduction outside the LLM, including marginal-gain budgeting, tree-based merging, and semantic-region clustering; (iii) inner-LLM techniques covering attention debiasing, KV cache compression, and verifier-guided speculative decoding; and (iv) hardware-aware execution on platforms such as the NVIDIA Jetson AGX Thor, including low-precision formats (INT8, NVFP4) and DVFS-aware scheduling. We also examine a design tension underexplored in the literature: query-aware pruning yields higher single-turn accuracy, but only query-agnostic pipelines support cached multi-turn interaction. Finally, we discuss how aggressive compression interacts with multimodal hallucination, providing a roadmap for verifiable, safety-aware pruning in future collaborative edge meshes.

***Index Terms*—Video Large Language Models, Edge Computing, Model Compression, State-Space Models, Spatiotemporal Token Pruning, KV Cache Optimization, Speculative Decoding.**

I. Introduction

The transformer architecture, originally developed for natural language processing, has become the backbone of modern artificial intelligence research. Following the widespread adoption of text-based Large Language Models (LLMs), the field expanded into multimodal understanding, producing Vision-Language Models (VLMs) and Video-Language Models (Video-LLMs). These models combine textual and visual inputs to enable downstream tasks such as zero-shot visual question answering, fine-grained temporal action recognition, and embodied robotic control [30], [32].

Despite these capabilities, Video-LLMs were developed for cloud environments with large GPU clusters and generous power budgets, which creates a mismatch with real-world deployment. There is growing demand to move inference from the cloud to edge devices such as personal computers, smartphones, autonomous vehicles, and Internet-of-Things (IoT) sensors. Edge deployment is motivated by the need for low-latency responses, operation in connectivity-limited environments, and on-device data handling for user privacy and corporate security [2], [28].

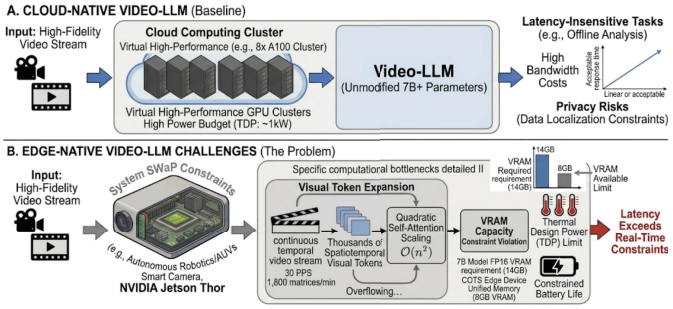


Fig. 1. The dichotomy between cloud-based and edge-native Video-LLM inference. The cloud offers limitless compute but suffers from high latency and bandwidth costs, whereas edge devices face strict SWaP (Space, Weight, and Power) constraints and thermal design power (TDP) limits.

Streaming high-fidelity video to centralized cloud servers introduces transmission latency that is often unacceptable for real-time applications, incurs bandwidth costs, and raises data privacy concerns in sensitive domains such as healthcare monitoring or enterprise surveillance. Running Video-LLMs on commercial-off-the-shelf (COTS) edge devices, however, requires working within strict Space, Weight, and Power (SWaP) constraints [41]. Edge devices must balance inference throughput against thermal design power (TDP) limits and battery life, drawing parallels to classical scheduling constraints in mobile edge networks [16], [21]. Video processing adds a further burden: long-form videos spread meaningful content sparsely across thousands of redundant frames, imposing a computational load that cannot be addressed by hardware scaling alone.

II. The Computational Architecture and Bottlenecks of Video-Language Models

To optimize Video-LLMs for edge hardware, we first characterize the computational bottlenecks of multimodal trans-

formers that process continuous temporal sequences.

A. Dimensionality and the Quadratic Scaling of Attention

Image-based VLMs use pre-trained visual encoders such as CLIP or SigLIP to generate spatial feature maps, which are aligned with textual embeddings in a shared representation space. Processing a single static image involves a fixed and predictable number of tokens. Video, by contrast, requires ingesting a continuous temporal stream. An edge camera capturing video at 30 frames per second (FPS) produces 1,800 spatial matrices per minute. Feeding this volume of uncompressed visual tokens directly into an LLM's context window is infeasible on edge hardware.

The core limitation is the transformer's self-attention mechanism, which scales as $\mathcal{O}(n^2)$ in compute with respect to sequence length n. When a single frame is encoded into hundreds of tokens, even a moderate video clip produces inputs of tens of thousands of tokens. This is commonly referred to as the *token explosion problem*, and it drives up inference latency.

B. The Prefill-Decode Dichotomy and SWaP Constraints

During the prefill phase, the initial stage of inference where the model processes the prompt and visual tokens together operations run in parallel, so the workload is compute-bound and grows with the square of the context size. The model then transitions to the autoregressive decode phase, which is memory-bandwidth bound: token generation is limited by the latency of fetching model weights from Video RAM (VRAM) for each output token.

Model parameter counts exceed what legacy optimization techniques can handle, necessitating system-algorithm co-designs for executing these heavy pipelines on target edge platforms [18]. Foundation LLMs marketed as "lightweight" contain 1–3 billion parameters, and larger models such as Qwen2.5-VL-7B contain 7 billion. An unmodified 7B model in FP16 requires roughly 14 GB of VRAM just to load weights, which rules out common edge devices such as the NVIDIA Jetson Orin Nano (8 GB of unified memory). The Key-Value (KV) cache also grows linearly during generation, further limiting the maximum context length that fits in memory.

III. Architectural Innovations: Streaming Paradigms and State-Space Models

Standard transformer architectures are poorly suited to long or unbounded-context processing due to the fixed range of their positional encodings. Several recent approaches address this by moving away from full-context ingestion toward streaming processing and recurrent alternatives.

A. Memory-Propagated Streaming Encoding

One approach to handling long videos within finite edge memory is streaming encoding with memory propagation. The VideoStreaming framework segments a long video into shorter clips [27]. A smaller secondary language model (for example, Phi-2) acts as a memory summarizer, evaluating the current clip's visual features together with a condensed "historical memory" vector.

A prefix task and a modified attention mask compress the video's history into a fixed-length memory embedding. This rolling representation avoids the $\mathcal{O}(n^2)$ cost of full-context attention. To limit loss of fine-grained details over time, the framework uses an Adaptive Memory Selection step: a discrete selection operation based on the Gumbel-Topk algorithm computes cosine similarity between the user's query and historical clip indicators, retrieving only the subset of prior memories relevant to the query.

B. State-Space Models (SSMs) as Temporal Projectors

A second approach uses State-Space Models (SSMs), in particular the Mamba architecture, to replace the linear vision-language projector. The STORM (Spatiotemporal Token Reduction for Multimodal LLMs) framework inserts a Mamba-based temporal encoder between the Vision Transformer (ViT) and the LLM backbone [12]. Mamba defines a linear recurrence $h_t = \overline{A}_t h_{t-1} + \overline{B}_t x_t$ and $y_t = C_t h_t$, where h_t is a hidden state summarizing prior history. The transition matrices are conditioned on the input x_t, which lets the model selectively emphasize salient visual information.

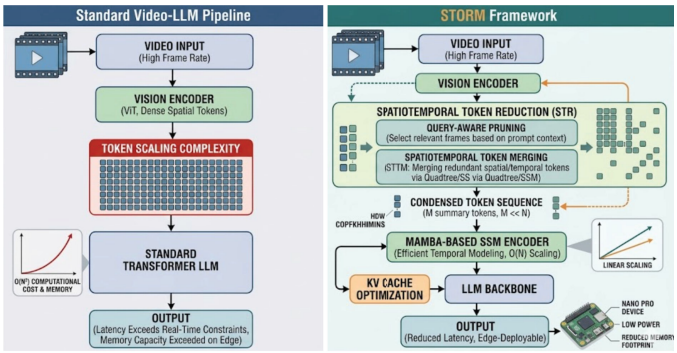


Fig. 2. Comparison between the standard Video-LLM pipeline and the optimized STORM framework [12]. (Left) The quadratic scaling penalty $\mathcal{O}(n^2)$ of traditional transformers. (Right) The architecture using Mamba-based temporal encoders and Semantic Connected Components (SCC) to achieve linear scaling and condensed token sequences.

Within STORM, the temporal projector performs a bidirectional spatiotemporal scan sweeping left-to-right and top-to-bottom within each frame, and sequentially across frames. The resulting "Summary Tokens" capture temporal dynamics, allowing downstream token compression to reduce the final token count to 12.5% of the original volume without degrading performance on LongVideoBench.

C. Online Streaming and Real-Time Edge Caching

For real-time edge applications such as autonomous driving, frameworks like STC (Streaming Token Compression) are used [39]. STC uses an STC-Cacher that intercepts raw frames and computes inter-frame cosine similarity. If similarity exceeds a threshold (> 0.85), the system skips the ViT encoding step and reuses cached features. An STC-Pruner then compresses the token sequence prior to prefill. Similarly,

StreamingAssistant targets online video understanding by measuring redundancy through Maximum Similarity to Spatially Adjacent Video Tokens (MSSAVT), applying dynamic temporal dropping (DTD) in under 1 millisecond [14].

IV. Algorithmic Approaches to Spatiotemporal Token Reduction (Outer-LLM)

Beyond fundamental architectural changes, token reduction algorithms applied outside the main LLM systematically distill the redundant spatiotemporal manifold of raw video.

A. Maximizing Marginal Gains via Dynamic Budgeting

Early token reduction uniformly applied fixed pruning rates. The MMG-Vid framework revolutionizes this by reformulating token pruning as an optimal subset selection problem driven by maximizing marginal gains [20]. It parses video into semantically coherent segments based on inter-frame similarity, executing a Marginal Gain-based Segment Budgeting protocol. A segment's value is co-determined by its "representativeness" and its "diversity."

At the token level, MMG-Vid utilizes Temporal-Guided Density Peak Clustering (TG-DPC). For subsequent frames within a segment, TG-DPC refines density metrics based on selected tokens, computing a "Temporal Relevance Density." This ensures the token budget is expended only on visual data yielding a strict marginal information gain, discarding 75% of tokens while preserving 99.5% of baseline performance [20].

B. Hierarchical and Tree-Based Redundancy Elimination

HieraVid executes pruning progressively across three levels: segment-level, frame-level, and layer-level [5]. At the frame level, it invokes a Determinantal Point Process (DPP) kernel matrix, perfectly balancing visual diversity against prompt relevance. At the layer level, it exploits the unidirectional propagation of multimodal information to aggressively shrink visual redundancy as network depth increases.

Alternatively, FlashVID utilizes Tree-based Spatiotemporal Token Merging (TSTM) [4]. FlashVID maps highly correlated visual features into complex spatiotemporal redundancy trees, capturing dynamic object motion without duplication. Preceded by an Attention and Diversity-based Token Selection (ADTS) module solving a Max-Min Diversity Problem (MMDP), FlashVID retains only 10% of visual tokens while preserving 99.1% of performance, effectively allowing a 10x expansion in the contextual input window [4].

C. Semantic Regions vs. Rigid Grid Parsing

Traditional techniques blindly discard tokens based on strict geometric grid proximity. LLaVA-Scissor argues this destroys the organic boundaries of actions, replacing grid parsing with Semantic Connected Components (SCC) [34]. SCC clusters tokens into distinct, globally related semantic regions across the entire token sequence, mapping tokens to the actual objects they represent. This guarantees comprehensive semantic coverage, outputting highly compact tokens without structural fragmentation. Relatedly, PruneVid merges temporally static tokens and applies an LLM-guided attention mechanism at an intermediate layer to preserve only prompt-critical visual tokens [8], [31].

D. Holistic Compression Frameworks

The HoliTom framework executes global redundancy-aware temporal segmentation followed by spatiotemporal merging outside the LLM, reducing tokens by over 90% [29]. It simultaneously introduces a robust inner-LLM token similarity-based merging approach designed to operate in synergy with the heavily pruned inputs, reducing computational costs to 6.9% of total FLOPs while preserving 99.1% accuracy.

V. Uncovering and Mitigating Native Attention Bias

Many token pruning paradigms rely on LLM self-attention scores to dictate token importance. However, empirical analyses reveal this introduces profound algorithmic artifacts.

A. Global and Local Biases

The AdaTP (Attention-Debiased Token Pruning) framework formally categorizes these artifacts [35]. *Global Bias* occurs because LLMs disproportionately focus on the extreme ends of sequences due to primacy and recency effects, artificially inflating attention scores of tokens at the beginning and end of a video regardless of semantic value. *Local Bias* dictates that within individual frames, attention mechanisms consistently over-concentrate on specific spatial coordinates (most frequently the center), erroneously attributing high importance to static pixels.

B. Mathematical Debiasing Transformations

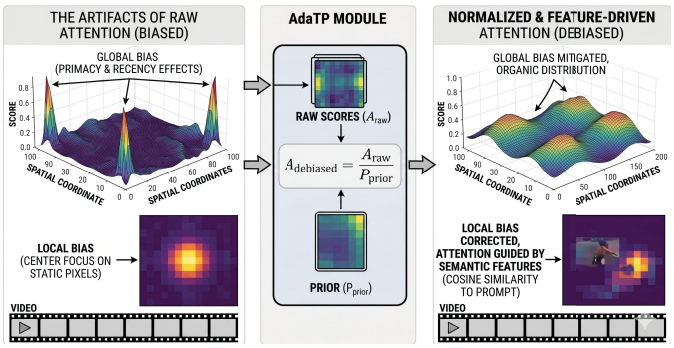


Fig. 3. Heatmap visualization of the AdaTP framework [35]. The biased attention (left) shows spikes at sequence boundaries and frame centers. The debiased attention (right) shows a normalized distribution, allowing the model to focus on semantic features guided by the prompt.

To neutralize these phenomena, AdaTP estimates a position-dependent prior by averaging attention weights across a vast corpus of random images. During inference, raw attention scores are mathematically divided by this pre-calculated prior. This yields normalized, debiased attention scores. The global module allocates higher token retention based on cosine similarity between frame and prompt features, while the local module deduplicates tokens situated at the exact same patch

position across consecutive frames, maintaining performance on par with baseline accuracy. [35].

VI. Query-Aware Selection vs. Query-Agnostic KV Cache Reuse

When engineering token reduction pipelines for edge deployment, a critical decision dictates system viability: should the pruning be conditioned dynamically on the user's text prompt (Query-Aware), or independent of text input (Query-Agnostic)?

A. The Precision of Query-Aware Compression

Query-aware methodologies excel in single-turn task execution. The KVTP (Keyframe-oriented Vision Token Pruning) framework utilizes a fine-tuned query-frame relevance predictor to execute a "soft selection" [17]. KVTP evaluates similarity between embedded text and fused frame embeddings augmented by a Context Fusion Head. Relevant frames experience minimal dropout, while irrelevant frames are heavily pruned but not entirely deleted, ensuring a coherent timeline. Similarly, AdaReTaKe adaptively allocates compression ratios across layers based directly on video-prompt attention, minimizing the upper bound of L_1 compression loss [38].

B. The Necessity of Query-Agnostic Processing for Agents

Query-aware models fail in multi-turn AI assistants because the video must be re-encoded and re-pruned for every new question, obliterating efficiency. Query-agnostic methods, like STTM (Spatio-Temporal Token Merging), operate strictly on intrinsic geometric properties [9]. STTM initiates a coarse-to-fine search over a quadtree data structure for spatial merging, and exploits spatio-temporal locality for directed pairwise merging. Because visual data is compressed prior to interacting with text queries, the resulting KV cache can be permanently stored in edge memory, allowing instant responses to infinite subsequent queries without re-triggering the vision encoder.

VII. Inner-LLM KV Cache Optimization and Speculative Decoding

While encoder-side pruning limits input size, the autoregressive generation phase introduces its own memory-bandwidth wall via the KV cache.

A. Dynamic KV Cache Compression

The DyCoke framework targets the decoding phase, continuously pruning spatially redundant tokens at each step of the decoding process while merging redundant tokens temporally [36]. Applying rigid one-shot pruning deletes tokens the model may need multiple steps later; DyCoke prevents this dynamically. This approach is complemented by plug-and-play acceleration modules like Video Compression Commander, which optimize inference without requiring retraining. The ReTake framework differentiates between low-level temporal redundancy and high-level knowledge redundancy [37]. Utilizing DPSelect to isolate keyframes by tracking inter-frame distance peaks, it deploys the PivotKV algorithm. Keyframes are locked as "pivots" in the KV cache and remain uncompressed, while non-pivot tokens are aggressively compressed based on attention scores, allowing Video-LLMs to ingest sequences up to 8x longer.

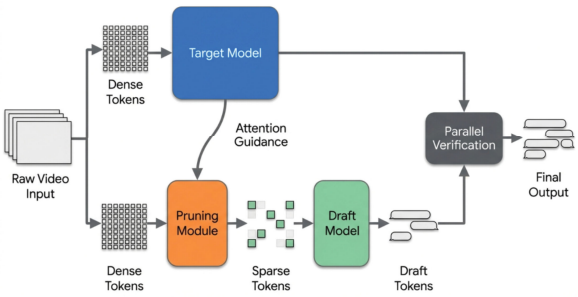


Fig. 4. Multimodal Speculative Decoding Architecture. The target model provides attention-guided pruning signals to the draft model, which generates speculative tokens for parallel verification.

B. Speculative Decoding for Multimodal Acceleration

Speculative decoding leverages a smaller, highly efficient "draft model" to rapidly predict tokens, verified by a massive "target model." The SpecVLM framework introduces verifier-guided token pruning specifically tailored for Video-LLMs [11]. Recognizing that a draft model's speculation accuracy is insensitive to extreme visual token loss, SpecVLM aggressively prunes up to 90% of video tokens fed to the draft. Stage I uses target model attention signals to perform Top-P retention of informative tokens, while Stage II executes spatially uniform reduction of background tokens. This drastically shrinks the draft's KV cache footprint, accelerating overall decoding by over 2.68x on architectures like LLaVA-OneVision-72B. The systemic flow of this verifier-guided pruning and parallel verification process is illustrated in Fig. 4.

TABLE I. Efficiency vs. Performance of Edge Token Reduction Frameworks

Framework	Pruning Methodology Focus	Tokens Retained	Accuracy Preserved
MMG-Vid	Maximal Marginal Gains (TG-DPC)	25.0%	99.5%
FlashVID	Tree-based Spatiotemporal Merging	10.0%	99.1%
HoliTom	Holistic Inner/Outer-LLM Merging	<10.0%	99.1%
HieraVid	Hierarchical Layer-wise Pruning	30.0%	99.0%
STTM	Quadtree-based Directed Pairwise	30.0%	98.0%

VIII. Hardware Benchmarking and Industrial Co-Design

Deploying optimized models requires precisely configured edge hardware. TokenPowerBench provides a phase-aligned

metrics pipeline attributing energy consumption distinctly to prefill and decode stages. On the NVIDIA Jetson AGX Thor (128GB) running TensorRT 10.13, the Qwen2.5-VL 3B model scales to 356.86 tokens/second at concurrency 8.

Hardware-aware execution utilizes advanced quantizations like NVFP4 and INT8 [7]. Frameworks like CLONE address edge variability by pairing a Mixture-of-Experts (MoE) router that integrates Low-Rank Adaptation (LoRA) modules during online inference with a Dynamic Voltage and Frequency Scaling (DVFS) controller. This mitigates energy consumption natively, yielding up to 11.92x inference acceleration on specialized 28nm AI silicon.

IX. The Case for Edge-First Model Co-Design

While current optimizations like STORM and SpecVLM demonstrate significant gains in post-training efficiency, we posit that the ”optimization-for-deployment” paradigm has reached a point of diminishing returns. To achieve true 30+ FPS real-time Video-LLM performance on mW-scale hardware, the community must shift toward **Edge-First Co-Design**.

Current models are typically pre-trained on massive cloud clusters with billions of parameters, only to be ”crushed” via quantization and pruning for the edge. This approach leads to ”representation collapse” where the model’s nuanced understanding of temporal dynamics is lost. Our position is that future Video-LLMs must incorporate *Hardware-in-the-Loop (HIL)* loss functions during pre-training. By penalizing high Energy-Delay Product (EDP) at the gradient level, we can evolve architectures that are natively sparse and quantization-robust, rather than forcing sparsity post-facto.

X. Beyond Single-Node: The Collaborative Edge Architecture

Most existing literature focuses on the ”Jetson-centric” bottleneck, assuming a single edge device must handle the entire inference pipeline. We argue that this single-node focus is a systemic limitation.

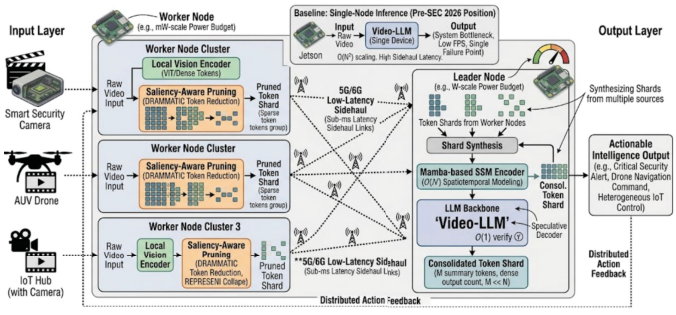


Fig. 5. The proposed Cluster-as-a-Camera (CaaC) architecture. Unlike the single-node baseline, this paradigm shards visual tokens across a low-latency 5G/6G sidehaul mesh. Worker nodes perform local saliency-aware pruning, while a Leader node synthesizes the consolidated shards for global Video-LLM inference.

A. Cluster-as-a-Camera (CaaC)

We advocate for a paradigm shift toward *Cluster-as-a-Camera* orchestration. In high-density environments (e.g., smart warehouses or urban intersections) and autonomous driving ecosystems, spatiotemporal visual tokens should not be processed in isolation [3], [47]. Connected vehicles and robotics demand low-latency, safety-critical processing, drawing heavily on next-generation edge-computing frameworks [10] and vehicular edge-cloud offloading schemes [16], [46]. This collaborative paradigm builds upon established frameworks in mobile edge networks [21], [22], which explore cost-efficient workload scheduling and dynamic task allocation in decentralized clusters.

Instead of isolated processing, we propose a distributed KV-cache mechanism where tokens are sharded across a mesh of nearby IoT and vehicular nodes. Utilizing sub-millisecond 5G/6G sidehaul networks [24], [33], a “Leader” node can perform the LLM backbone inference, while “Worker” nodes handle parallelized visual encoding, dynamic routing, and dynamic radio resource management [15].

B. Collaborative VLM Systems and Offloading Orchestration

Transitioning from single-device optimization to cloud-edge collaborative inference is key to deploying massive models. Recent frameworks like VaVLM [45], edgeVLM [25], and SpotVLM [26] enable real-time video analytics and context-transfer between edge devices and the cloud. To optimize latency in IoT settings, SEC introduces semantic-aware edge-cloud collaboration for low-latency IoT video processing [42]. Under severe hardware heterogeneity, frameworks like AdaptLink [6] and MoA-Off [43] perform adaptive modality-aware task offloading across devices. Furthermore, systems such as SwiftReTaKe [13] execute high-speed visual redundancy reduction for collaborative video-language understanding, while decoupled pipelines optimize offloading of the VLM inference workload [44]. Finally, agent-centric edge deployments like ELLMPEG [1] utilize orchestrations of multi-LLM networks to establish edge general intelligence [19].

C. Privacy-Preserving Token Sharding

Furthermore, this distributed approach offers a unique position on privacy. By ensuring that raw pixels never leave the initial capture node and only ”pruned embeddings” are shared across the local mesh, we can create a verifiable boundary against data localization violations. While reconstruction risks exist, assuming embedding-inversion attacks are mitigated by standard defenses (such as cryptographic masking or differential privacy), this decentralized strategy effectively inoculates the system against raw-pixel exfiltration. The future of Edge-AI lies not in bigger chips, but in more intelligent collaboration between heterogeneous low-power nodes.

XI. Verifiable Robustness in Pruned Architectures

A critical gap in current Video-LLM research is the lack of ”Safety-Aware Pruning.” As we move toward the 90%+

token reduction rates seen in modern frameworks, the risk of ”catastrophic forgetting” of rare but high-stakes events (e.g., a weapon in a security feed or a small obstacle for a high-speed drone) increases exponentially.

We propose that future research must prioritize *Verifiable Pruning Metrics*, particularly in latency-critical contexts like federated learning for autonomous driving in vehicle-edge-cloud networks [40] and model-driven token sparsification for onboard automated driving models [23]. Instead of evaluating models solely on mean Average Precision (mAP) or FLOPs, we must adopt “Worst-Case Saliency Preservation” benchmarks. A model’s efficiency should be disqualified if its pruning module cannot mathematically guarantee the retention of spatiotemporal regions that exceed a predefined ”criticality threshold.” Without this formal guarantee, Video-LLMs will remain restricted to ”Latency-Insensitive” tasks and fail the safety requirements of autonomous robotics and AUVs.

XII. Application Ecosystem and Hallucination Risks

A. Industrial and Affective Computing

The convergence of streaming architectures and KV cache pruning represents a critical threshold for edge AI. In Multi-Agent Video Recommenders, AI agents execute lifelong personalization directly on user devices by utilizing heavily compressed, query-agnostic spatiotemporal tokens. In e-commerce and affective computing, real-time facial expression tracking requires instantaneous continuous processing. Optimized token pruning that eliminates static backgrounds while preserving dynamic micro-expressions enables the deployment of affective systems securely on smartphones. Furthermore, Domain-Specific Language (DSL) approaches for gaming leverage lightweight vision models to integrate environmental understanding without thermal throttling.

B. The Hallucination Trade-off and Factual Reliability

However, the pursuit of extreme compression introduces severe risks regarding model factuality. As token budgets are violently squeezed, the mathematical probability of discarding critical visual evidence increases. If an algorithm relies on overly aggressive temporal dropping, it may entirely miss brief, high-velocity events (e.g., a rapid flash of text). When LLMs are deprived of exact visual grounding, they default to parameterized linguistic priors, resulting in severe *multimodal hallucinations*. The model generates fluent, syntactically correct text describing actions completely fabricated by the language engine because the true visual premise was pruned away. Thus, reliance on debiasing frameworks (AdaTP) and margin-maximizing retention (MMG-Vid) is a non-negotiable requirement for preserving the trustworthiness of edge AI.

XIII. Conclusion

Deploying Video-LLMs in resource-constrained systems is limited by the computational cost of dense spatiotemporal tokens. Achieving practical edge inference requires optimization across several layers. Architectural work on memory-propagated streaming and Mamba-based temporal encoders reduces the quadratic cost of sequence-length attention. Token reduction has moved from grid-based elimination toward structurally aware compression marginal gain selection, tree-based merging, and semantic connected components. Addressing system-level issues such as attention bias (AdaTP) and decoding cost (SpecVLM) yields further gains in latency and energy efficiency. Taken together, these methods make video-reasoning on edge devices a realistic target, though careful attention to compression-induced hallucination remains important in any deployment. Ultimately, transitioning to truly robust edge-native systems requires a shift from survey-level post-training compression toward native edge-first co-design, verifiable pruning metrics that protect safety-critical domains, and distributed Cluster-as-a-Camera (CaaC) orchestrations.

XIV. Acknowledgements

The authors acknowledge the use of AI for image generation and refining text in this paper.

References

[1] Zoha Azimi Ourimi, Reza Fahrani, Radu Prodan, and Christian Timmerer. Ellmpeg: An edge-based agentic llm video processing tool. In *Proceedings of the ACM Multimedia Systems Conference 2026*, pages 214–225, 2026.

[2] Guangji Bai, Zheng Chai, Chen Ling, Shiyu Wang, Jiaying Lu, Nan Zhang, Tingwei Shi, Ziyang Yu, Mengdan Zhu, Yifei Zhang, et al. Beyond efficiency: A systematic survey of resource-efficient large language models. *arXiv preprint arXiv:2401.00625*, 2024.

[3] Maria-Dolores Cano, Antonio Guillen-Perez, Igor Tasic, and Antonio Villafranca. A conceptual framework for the development of autonomous driving in 6g: the role of ai and edge computing. In *2023 8th International Conference on Control, Robotics and Cybernetics (CRC)*, pages 97–105. IEEE, 2023.

[4] Ziyang Fan, Keyu Chen, Ruilong Xing, Yulin Li, Li Jiang, and Zhuotao Tian. Flashvid: Efficient video large language models via training-free tree-based spatiotemporal token merging. *arXiv preprint arXiv:2602.08024*, 2026.

[5] Yansong Guo, Chaoyang Zhu, Jiayi Ji, Jianghang Lin, and Liujuan Cao. Hieravid: Hierarchical token pruning for fast video large language models. *arXiv preprint arXiv:2604.01881*, 2026.

[6] Xinyi Hu, Zihan Chen, Kun Guo, Meng Zhang, and Howard Hao Yang. Adaptlink: A heterogeneity-aware adaptive framework for distributed mllm inference. In *AAAI 2025 Workshop on Artificial Intelligence for Wireless Communications and Networking (AI4WCN)*, 2025.

[7] Jin Huang, Yuchao Jin, Le An, and Josh Park. Litevlm: A low-latency vision-language model inference pipeline for resource-constrained environments. *arXiv preprint arXiv:2506.07416*, 2025.

[8] Xiaohu Huang, Hao Zhou, and Kai Han. Prunevid: Visual token pruning for efficient video large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19959–19973, 2025.

[9] Jeongseok Hyun, Sukjun Hwang, Su Ho Han, Taeoh Kim, Inwoong Lee, Dongyoon Wee, Joon-Young Lee, Seon Joo Kim, and Minho Shim. Multi-granular spatio-temporal token merging for training-free acceleration of video llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23990–24000, 2025.

[10] Hatem Ibn-Khedher, Mohammed Laroui, Hassine Moungla, Hossam Afifi, and Emad Abd-Elrahman. Next-generation edge computing assisted autonomous driving based artificial intelligence algorithms. *IEEE Access*, 10:53987–54001, 2022.

[11] Yicheng Ji, Jun Zhang, Heming Xia, Jinpeng Chen, Lidan Shou, Gang Chen, and Huan Li. Specvlm: Enhancing speculative decoding of video llms via verifier-guided token pruning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7216–7230, 2025.

[12] Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Li, Guo Chen, Zhiqi Li, De-An Huang, Guilin Liu, Zhiding Yu, Kurt Keutzer, et al. Storm: Token-efficient long video understanding for multimodal llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5830–5841, 2025.

[13] Xinqi Jin, Fan Dang, Kebin Liu, Jiangchuan Liu, and Jingao Xu. Swiftretake: Quick and accurate redundancy reduction for cloud-edge collaborative video-language understanding. In *2025 IEEE 31th International Conference on Parallel and Distributed Systems (ICPADS)*, pages 1–8. IEEE, 2025.

[14] Xinqi Jin, Hanxun Yu, Bohan Yu, Kebin Liu, Jian Liu, Keda Tao, Yixuan Pei, Huan Wang, Fan Dang, Jiangchuan Liu, et al. Streamingassistant: Efficient visual token pruning for accelerating online video understanding. *arXiv preprint arXiv:2512.12560*, 2025.

[15] Prajwal Keshavamurthy, Emmanouil Pateromichelakis, Dirk Dahlhaus, and Chan Zhou. Edge cloud-enabled radio resource management for co-operative automated driving. *IEEE Journal on Selected Areas in Communications*, 38(7):1515–1530, 2020.

[16] Xin Li, Yifan Dang, Mohammad Aazam, Xia Peng, Tefang Chen, and Chunyang Chen. Energy-efficient computation offloading in vehicular edge cloud computing. *IEEE Access*, 8:37632–37644, 2020.

[17] Yudong Liu, Jingwei Sun, Yueqian Lin, Jianyi Zhang, Jingyang Zhang, Ming Yin, Qinsi Wang, Hai Li, and Yiran Chen. Keyframe-oriented vision token pruning: Enhancing efficiency of large vision language models on long-form video processing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20802–20811, 2025.

[18] Xudong Lu, Yinghao Chen, Cheng Chen, Hui Tan, Boheng Chen, Yina Xie, Rui Hu, Guanxin Tan, Renshou Wu, Yan Hu, et al. Bluelm-v-3b: Algorithm and system co-design for multimodal large language models on mobile devices. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4145–4155, 2025.

[19] Haoxiang Luo, Yinqiu Liu, Ruichen Zhang, Jiacheng Wang, Gang Sun, Dusit Niyato, Hongfang Yu, Zehui Xiong, Xianbin Wang, and Xuemin Shen. Toward edge general intelligence with multiple-large language model (multi-llm): architecture, trust, and orchestration. *IEEE Transactions on Cognitive Communications and Networking*, 2025.

[20] Junpeng Ma, Qizhe Zhang, Ming Lu, Zhibin Wang, Qiang Zhou, Jun Song, and Shanghang Zhang. Mmg-vid: Maximizing marginal gains at segment-level and token-level for efficient video llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 24253–24261, 2026.

[21] Xiao Ma, Shan Zhang, Wenzhuo Li, Puheng Zhang, Chuang Lin, and Xuemin Shen. Cost-efficient workload scheduling in cloud assisted mobile edge computing. In *2017 IEEE/ACM 25th International Symposium on Quality of Service (IWQoS)*, pages 1–10. IEEE, 2017.

[22] Xiao Ma, Ao Zhou, Shan Zhang, Qing Li, Alex X Liu, and Shangguang Wang. Dynamic task scheduling in cloud-assisted mobile edge computing. *IEEE Transactions on Mobile Computing*, 22(4):2116–2130, 2021.

[23] Yunsheng Ma, Amr Abdelraouf, Rohit Gupta, Ziran Wang, and Kyungtae Han. Video token sparsification for efficient multimodal llms in autonomous driving. *arXiv preprint arXiv:2409.11182*, 2024.

[24] Wendlasida Ouedraogo, Andrea Araldo, Badii Jouaber, Hind Castel, and Remy Grunblatt. Can edge computing fulfill the requirements of automated vehicular services using 5g network? In *2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring)*, pages 1–5. IEEE, 2024.

[25] Chen Qian, Xinran Yu, Zewen Huang, Danyang Li, Qiang Ma, Fan Dang, Xuan Ding, Guangyong Shang, and Zheng Yang. edgevlm: Cloud-edge collaborative real-time vlm based on context transfer. *arXiv preprint arXiv:2508.12638*, 2025.

[26] Chen Qian, Xinran Yu, Zewen Huang, Danyang Li, Qiang Ma, Fan Dang, Xuan Ding, Guangyong Shang, and Zheng Yang. Spotvlm: Cloud-edge collaborative real-time vlm based on context transfer. *arXiv e-prints*, pages arXiv–2508, 2025.

[27] Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360, 2024.

[28] Álvaro Santos, Jorge Bernardino, and Noélia Correia. Automated application deployment on multi-access edge computing: A survey. *IEEE Access*, 11:89393–89408, 2023.

[29] Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Holitom: Holistic token merging for fast video large language models. *arXiv preprint arXiv:2505.21334*, 2025.

[30] Ahmed Sharshar, Latif U Khan, Waseem Ullah, and Mohsen Guizani. Vision-language models for edge networks: A comprehensive survey. *IEEE Internet of Things Journal*, 2025.

[31] Leqi Shen, Guoqiang Gong, Tao He, Yifeng Zhang, Pengzhang Liu, Sicheng Zhao, et al. Fastvid: Dynamic density pruning for fast video large language models. *Advances in Neural Information Processing Systems*, 38:123553–123581, 2026.

[32] Gaurav Shinde, Anuradha Ravi, Emon Dey, Shadman Sakib, Milind Rampure, and Nirmalya Roy. A survey on efficient vision-language models. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 15(3):e70036, 2025.

[33] Ridha Soua, Ion Turcanu, Florian Adamsky, Detlef Führer, and Thomas Engel. Multi-access edge computing for vehicular networks: A position paper. In *2018 IEEE Globecom Workshops (GC Wkshps)*, pages 1–6. IEEE, 2018.

[34] Boyuan Sun, Jiaxing Zhao, Xihan Wei, and Qibin Hou. Llava-scissor: Token compression with semantic connected components for video llms. *arXiv preprint arXiv:2506.21862*, 2025.

[35] Fengyuan Sun, Leqi Shen, Hui Chen, Sicheng Zhao, Jungong Han, and Guiguang Ding. Adatp: Attention-debiased token pruning for video large language models. *arXiv preprint arXiv:2505.20100*, 2, 2025.

[36] Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Dycoke: Dynamic compression of tokens for fast video large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18992–19001, 2025.

[37] Xiao Wang, Qingyi Si, Jianlong Wu, Shiyu Zhu, Li Cao, and Liqiang Nie. Retake: Reducing temporal and knowledge redundancy for long video understanding. *arXiv preprint arXiv:2412.20504*, 2024.

[38] Xiao Wang, Qingyi Si, Shiyu Zhu, Jianlong Wu, Li Cao, and Liqiang Nie. Adaretake: Adaptive redundancy reduction to perceive longer for video-language understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5417–5432, 2025.

[39] Yiyu Wang, Xuyang Liu, Xiyan Gui, Xinying Lin, Boxue Yang, Chenfei Liao, Tailai Chen, and Linfeng Zhang. Accelerating streaming video large language models via hierarchical token compression. *arXiv preprint arXiv:2512.00891*, 2025.

[40] Tianao Xiang, Mingjian Zhi, Yuanguo Bi, Lin Cai, and Yuhao Chen. Flad: Federated learning for llm-based autonomous driving in vehicle-edge-cloud networks. *arXiv preprint arXiv:2511.09025*, 2025.

[41] Mengwei Xu, Wangsong Yin, Dongqi Cai, Rongjie Yi, Daliang Xu, Qipeng Wang, Bingyang Wu, Yihao Zhao, Chen Yang, Shihe Wang, et al. A survey of resource-efficient llm and multimodal foundation models. *arXiv preprint arXiv:2401.08092*, 2024.

[42] Mengyu Yang, Ye Tian, Peizhuang Cong, Lanshan Zhang, Gongli Xi, Song Wang, and Wendong Wang. Sec: Enabling mllms for low-latency iot video analysis via semantic-aware edge-cloud collaboration. *IEEE Internet of Things Journal*, 2026.

[43] Zheming Yang, Qi Guo, Yunqing Hu, Chang Zhao, Chang Zhang, Jian Zhao, and Wen Ji. Moa-off: Adaptive heterogeneous modality-aware offloading with edge-cloud collaboration for efficient multimodal llm inference. *arXiv preprint arXiv:2509.16995*, 2025.

[44] Xingyu Yuan and He Li. Video language model inference offloading in cloud-edge collaboration: A decoupled approach. In *2026 20th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, pages 1–7. IEEE, 2026.

[45] Yang Zhang, Hanling Wang, Qing Bai, Haifeng Liang, Peican Zhu, Gabriel-Miro Muntean, and Qing Li. Vavlm: Toward efficient edge-cloud video analytics with vision-language models. *IEEE Transactions on Broadcasting*, 2025.

[46] Junhui Zhao, Qiuping Li, Yi Gong, and Ke Zhang. Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks. *IEEE Transactions on Vehicular Technology*, 68(8):7944–7956, 2019.

[47] Qi Zhu, Bo Yu, Ziran Wang, Jie Tang, Qi Alfred Chen, Zihao Li, Xiangguo Liu, Yunpeng Luo, and Lingzi Tu. Cloud and edge computing for connected and automated vehicles. *Foundatiosn and Trends in Electronic Design Automation*, 14(1-2):1–170, 2023.