© DEC 2025 | IRE Journals | Volume 9 Issue 6 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I6-1712471

A Survey on Efficient Edge-Based Video Streaming: Integrating AI-Powered Upscaling, Adaptive Delivery, and Latency Reduction

HARDIK JAIN[1], RIYA SAGAR[2], SNIGDHA VIJAY[3], MONISHA H.M[4]
[1,2,3]*Student, Dept of Information Science, BMS College of Engineering, Bangalore, India*
[4]*Assistant Professor, Dept of Information Science, BMS College of Engineering, Bangalore, India*

Abstract: ***Video streaming has become one of the dominant drivers of internet traffic, demanding scalable, low-latency, and high-quality delivery solutions. Traditional cloud-based streaming suffers from high latency and bandwidth stress under dynamic network conditions. This survey explores edge-based video streaming architectures that integrate AI-driven upscaling, adaptive bitrate delivery, and latency mitigation mechanisms. We analyze recent advancements in edge caching, reinforcement learning–based adaptive streaming, and AI-powered super-resolution. Further, we propose an architectural model combining edge computing and AI for optimized video delivery. The survey highlights current achievements, limitations, and open challenges, offering guidance for researchers and engineers working on next-generation streaming systems.***

Keywords - Edge Computing, Adaptive Streaming, Latency Reduction, AI Upscaling, Deep Reinforcement Learning.

I.INTRODUCTION

The demand for high-quality video delivery has made scalable and efficient streaming a fundamental requirement for modern platforms. Traditional cloud-centric models often suffer from high latency, congestion, and inconsistent Quality of Experience (QoE), particularly under dynamic network conditions. These shortcomings have accelerated research and deployment of edge computing, where computation and storage are placed closer to end users to improve responsiveness and reduce backhaul load.

By offloading tasks such as caching, transcoding, and adaptive bitrate (ABR) control to edge nodes, latency is reduced, bandwidth stress is minimized, and scalability is improved. When combined with artificial intelligence (AI), including reinforcement learning for adaptive streaming and neural network–based video upscaling, edge-based frameworks can provide higher visual quality while maintaining resource efficiency.

This survey examines the intersection of edge computing and intelligent video delivery, reviewing recent work in areas such as low-latency streaming, adaptive algorithms, energy-aware transcoding, and secure analytics at the edge. The objective is to evaluate both the technical progress and the unresolved challenges that must be addressed for scalable AI-driven video streaming systems. The paper aims to provide researchers and practitioners with an overview of emerging methods, trends, and opportunities shaping the future of video delivery in heterogeneous and bandwidth-constrained environments.

II. LITERATURE SURVEY

A wide body of research has examined the intersection of edge computing, adaptive streaming, and AI-driven optimization. Existing studies cover reinforcement learning for bitrate adaptation, protocol-level enhancements, AI-based upscaling, and edge-based analytics. This section reviews representative works that have shaped the state of the art.

Li et al. [1] introduced a Deep Reinforcement Learning (DRL) based approach for adaptive video streaming to address the limitations of rule-based ABR algorithms. Their state–action–reward model dynamically adjusted bitrate and resolution in response to fluctuating bandwidth and buffer conditions. The method reduced buffering and improved QoE but required significant computational resources and showed limited generalization. Future work suggested by the authors includes enhancing robustness and extending DRL to multi-user scenarios.

Xu et al. [2] analyzed the role of HTTP/2 in low-latency streaming, leveraging multiplexing, header compression, and stream prioritization. Their experiments showed reduced connection overhead, more efficient bandwidth use, and improved segment delivery. While performance gains were clear, widespread deployment was restricted by backward compatibility concerns. The authors highlighted the promise of HTTP/3 and QUIC in supporting real-time and mobile streaming.

Mengistu [3] evaluated NVIDIA DLSS 3.0, an AI-driven upscaling method originally applied in gaming and VR. Results demonstrated higher frame rates and stable visual quality, achieved through motion vector integration. However, DLSS's dependence on proprietary NVIDIA hardware limited its portability across heterogeneous edge systems. The study emphasized the need for open-source AI upscaling frameworks that could be adapted for broader deployment.

Lundkvist [4] investigated a CNN-based upscaling technique deployed in Swedish Television (SVT). Compared with bicubic interpolation, the CNN approach provided higher viewer satisfaction and enabled streaming at lower bitrates without visible degradation. Key barriers included high computational cost and integration complexity within existing broadcasting pipelines. The authors proposed hybrid cloud–edge models and algorithmic optimization to improve efficiency.

Hu et al. [5] presented a survey of edge-based video analytics, focusing on AI-driven real-time processing tasks such as object detection, compression, and resource allocation. The work highlighted federated learning as a privacy-preserving solution for distributed edge environments. However, persistent gaps were noted in interoperability, standardization, and privacy protection, particularly within large-scale IoT networks. The authors called for frameworks that balance efficiency, security, and scalability.

Nassisid et al. [6] proposed an adaptive streaming framework for 6G networks, integrating intelligent buffer management with predictive modeling of user behavior. Simulations indicated reduced latency and improved adaptability under fluctuating bandwidth conditions. However, the framework has not been validated in real-world environments, especially those involving high mobility and variable signal quality. The authors recommend live field trials and the integration of semantic-aware communication protocols to align with next-generation streaming requirements.

Choi et al. [7] combined Markov Decision Processes (MDP) with Lyapunov optimization to enhance adaptive video delivery in edge caching systems. Their method reduced rebuffering and improved QoE in controlled experiments. The main limitation was the reliance on perfect knowledge of cache and network states, which is unrealistic in dynamic deployments. Future directions include incorporating partial observability and hardware constraints to enable scalable real-world adoption.

Chen et al. [8] introduced SODA, a dynamic ABR controller based on online convex optimization. Compared to MPC-based approaches, SODA reduced abrupt bitrate changes and improved playback stability. The system, however, performed poorly in interference-heavy, multi-user environments. The authors suggest integrating network slicing and cross-layer optimization, particularly for the high concurrency expected in 5G and beyond.

Yeregui et al. [9] analyzed the influence of 5G RAN parameters such as jitter and round-trip time on live video quality. Their predictive model performed well in controlled conditions but degraded significantly during user mobility and handoffs. To address this, the authors propose adaptive, context-aware prediction mechanisms that incorporate mobility patterns and real-time feedback to sustain quality under dynamic conditions.

Dharuman et al. [10] investigated improvements to HLS and DASH streaming protocols through edge computing and ML-based traffic prediction. Their approach reduced CDN load and enhanced streaming efficiency in stable traffic environments. However, during "flash crowd" scenarios, cache misses increased, limiting performance. The authors recommend hybrid approaches that combine AI-based prediction with rule-based fallbacks to manage unpredictable demand spikes effectively.

STL Partners [11] conducted a case study on live sports streaming using edge computing. Deploying regional micro–datacenters reduced last-mile latency

by 47%, enabling real-time interactivity and enhancing QoE. The main drawback was increased energy use and operational cost in distributed deployments. The authors highlight the need for renewable energy integration and efficient resource allocation to make such solutions sustainable for large-scale events.

Muvi [12] implemented edge micro–datacenters with Kubernetes orchestration to deliver ultra-low-latency interactive streaming, achieving end-to-end delays as low as 200 ms. The system supported dynamic scaling and load balancing under high demand but suffered from cold-start delays and inefficient resource handling during unexpected surges. The study recommends predictive auto-scaling and cost-optimized deployment to improve scalability across global CDNs.

Bilal and Erbad [13] proposed a Docker container–based edge transcoding system that enabled real-time resolution adaptation. Their design reduced bandwidth consumption by up to 60% and supported multi-tenant usage. However, GPU resource sharing across containers raised security risks. Future directions include the use of trusted execution environments (TEEs) and better isolation mechanisms to enhance both security and performance across heterogeneous edge hardware.

Choi et al. [14] integrated deep reinforcement learning (DRL) with device-to-device (D2D) communication for adaptive streaming in vehicular networks. The system improved V2X reliability by 33% and reduced playback interruptions. Nevertheless, performance declined significantly under adverse weather conditions. The authors suggest incorporating robust channel coding and environmental sensing to ensure consistent streaming quality in real-world deployments.

Ghosh et al. [15] introduced REACT, an asynchronous edge–cloud framework for real-time video analytics. Lightweight feature extraction at the edge was combined with intensive cloud-side processing, reducing false positives by 18%. However, encryption of edge–cloud data transfer introduced additional latency. The paper calls for lightweight, privacy-preserving encryption schemes that balance speed with security in large-scale deployments.

The NSF study [16] proposed a buffer management and SVM-based QoE prediction system for proactive streaming adaptation. Controlled tests achieved 92% user satisfaction. However, the system was unable to adapt effectively across diverse device types, leading to quality mismatches. The authors recommend content-aware rescaling and device-specific adaptation algorithms to improve heterogeneity support.

Bilal et al. [17] designed a field-of-view (FoV)-aware edge caching method for 360° video streaming. By predicting user gaze and caching only relevant tiles, the system reduced bandwidth usage by 55%. Accuracy, however, declined in dynamic environments due to inconsistent gaze prediction. Future work should combine multimodal data such as head motion and engagement metrics to improve FoV prediction accuracy in immersive media.

ElasticEdge [18] presented a Kubernetes-based elastic edge framework for live video analytics. The framework dynamically allocated resources, reducing processing latency by 40% and improving responsiveness during live events. Cold-start delays of up to 15 seconds during sudden demand spikes were a limitation. The authors propose predictive pre-warming and proactive resource reservation to enhance performance under unpredictable traffic conditions.

To consolidate these findings, Table I summarizes the surveyed approaches, highlighting their contributions, limitations, and potential research directions.

Reference	Approach / Contribution	Key Findings	Limitations	Future Directions
[1] Li et al. (2020)	DRL-based ABR adaptation	Reduced buffering, improved QoE	High computation cost, poor generalization	Multi-agent DRL for multi-user

[2] Xu et al. (2018)	HTTP/2 for streaming	Lower connection overhead, prioritized segments	Limited backward compatibility	HTTP/3, QUIC integration
[3] Mengistu (2023)	NVIDIA DLSS AI upscaling	Higher frame rates, stable quality	Proprietary hardware dependence	Open-source, portable upscalers
[4] Lundkvist (2021)	CNN-based upscaling (SVT)	Better quality at lower bitrates	High compute cost, workflow integration issues	Hybrid cloud–edge, optimized CNN
[5] Hu et al. (2023)	Survey of edge analytics	Highlighted federated learning, edge ML	Lack of standards, privacy gaps	Interoperable, secure frameworks
[6] Nassisid et al. (2024)	6G adaptive framework	Lower latency, predictive buffer control	No real-world validation	Field trials, semantic protocols
[7] Choi et al. (2019)	MDP + Lyapunov optimization	Reduced rebuffering, improved QoE	Requires perfect network knowledge	Handle partial observability
[8] Chen et al. (2024)	SODA ABR (convex optimization)	Stable playback, fewer bitrate shifts	Weak in interference-heavy networks	Network slicing, cross-layer design
[9] Yeregui et al. (2025)	5G RAN predictive model	Accurate under controlled settings	Fails with mobility/handoffs	Context-aware, mobility-integrated
[10] Dharuman et al. (2023)	Edge + ML for HLS/DASH	Reduced CDN load, improved efficiency	Cache misses in flash crowds	Hybrid AI + rule-based fallback
[11] STL (2025)	Edge for sports streaming	47% latency reduction, interactivity	High energy/cost overhead	Renewable integration, scaling
[12] Muvi (2025)	Edge micro-DCs, Kubernetes	200 ms latency, dynamic scaling	Cold-start, surge inefficiency	Predictive auto-scaling
[13] Bilal & Erbad (2017)	Edge transcoding (Docker)	60% bandwidth saving, multi-tenant	GPU sharing risks	TEEs, isolation, optimization
[14] Choi et al. (2025)	DRL + D2D in vehicular nets	33% better reliability, fewer interruptions	Weak in adverse weather	Channel coding, env. sensing
[15] Ghosh et al. (2023)	REACT edge–cloud analytics	18% fewer false positives	Added latency due to encryption	Lightweight privacy techniques
[16] NSF (2025)	Buffer + SVM QoE prediction	92% user satisfaction	Device heterogeneity issues	Content-aware, device-specific
[17] Bilal et al. (2018)	FoV-aware 360° caching	55% bandwidth saving	Low accuracy in dynamic VR	Multimodal gaze prediction
[18] ElasticEdge (2022)	Kubernetes elastic edge	40% latency reduction	Cold-start delays (15 s)	Pre-warming, proactive allocation

Table I - Comparative Summary of Edge-Based Video Streaming Research

© DEC 2025 | IRE Journals | Volume 9 Issue 6 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I6-1712471

III. PROPOSED MODEL

The proposed architecture integrates computational intelligence techniques at the edge layer to optimize video delivery across four modules: AI-based upscaling, adaptive delivery engine, intelligent edge node management, and latency optimization protocols. The framework aims to minimize bandwidth overhead, reduce latency, and improve Quality of Experience (QoE). A conceptual block diagram is shown in Fig. 1.

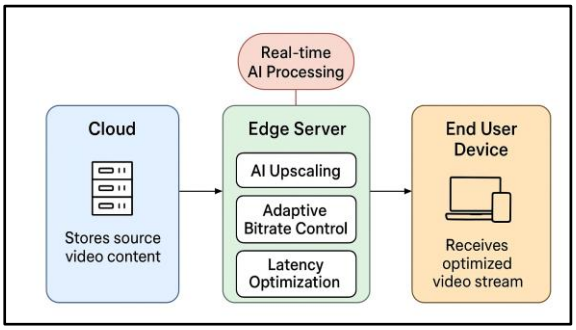

Fig. 1 - Block Diagram of Proposed Computational-Intelligence–Enabled Edge Streaming Model

A. AI-Based Edge Upscaling

Video frames are transmitted at reduced resolution from the origin server to conserve bandwidth.

At the edge, deep convolutional super-resolution models such as Real-ESRGAN are employed to reconstruct high-resolution frames.

The upscaler leverages motion-compensated input to maintain temporal consistency across frames, preventing flicker or visual artifacts during fast-motion sequences.

This module enables bandwidth savings of up to 30–40% while preserving perceptual quality, based on benchmarks reported in prior studies.

B. Adaptive Delivery Engine with Reinforcement Learning

The adaptive engine continuously monitors network throughput, playback buffer occupancy, and device capability.

A Deep Reinforcement Learning (DRL) agent selects optimal bitrate/resolution pairs using a state–action–reward loop:

- *State*: current bandwidth, buffer level, latency estimate.
- *Action*: select next video segment bitrate/resolution.
- *Reward*: QoE score, combining rebuffering penalty, visual quality, and stability.

Unlike rule-based ABR, the DRL approach adapts proactively to varying conditions, reducing abrupt bitrate shifts and buffering events.

C. Edge Node Caching and Resource Management

Edge nodes handle local caching and transcoding of popular content to reduce backhaul traffic.

A predictive caching algorithm is applied, where machine learning models analyze historical access patterns and regional demand to prefetch frequently requested segments.

Workload is balanced across containers using Kubernetes orchestration, allowing elastic scaling based on user demand.

This layer minimizes round-trip delays, particularly for live and on-demand events in dense urban regions.

D. Latency Optimization Layer

To further reduce end-to-end delay, the system employs AI-driven prefetching to predict and load upcoming segments before playback requests occur.

Transport is managed through congestion-aware protocols such as QUIC and HTTP/3, reducing handshake overhead and improving packet recovery in lossy environments.

The combination of protocol-level optimization and predictive scheduling ensures consistent playback, with target end-to-end latency under 200–300 ms for live streaming scenarios.

E. Integration and Workflow

The four modules operate in a coordinated pipeline:

1. User requests are redirected to the nearest edge node.
2. Edge node retrieves base-resolution video from cloud or cache.
3. AI-based upscaler enhances the video to target resolution.
4. DRL-powered adaptive delivery engine selects bitrate/resolution dynamically.

5. Latency optimization mechanisms ensure low-delay delivery to the user device.

This integration of computational intelligence across the pipeline transforms edge nodes from passive content relays into intelligent decision-making agents, capable of optimizing bandwidth, predicting demand, and maintaining high QoE under variable network conditions.

IV. CHALLENGES, GAPS, AND IMPLEMENTATION OUTLOOK

Despite recent advances in edge-based video streaming, several open challenges remain before large-scale deployment of intelligent streaming systems can be realized. Based on the surveyed works and our proposed model, we identify the following gaps and outline a concrete implementation roadmap.

A. Scalability and Real-Time Performance

Most AI-based streaming systems, such as DRL for ABR [1] and CNN-based upscaling [4], exhibit high computational demand, limiting deployment at scale. Current solutions often assume powerful GPUs at the edge, which is impractical in resource-constrained nodes.

Implementation Outlook: Our design will employ containerized deployment (Docker + Kubernetes) to allow elastic scaling. Upscaling will use Real-ESRGAN trained on the DIV2K dataset for image super-resolution, optimized with quantized inference (TensorRT or ONNX runtime) to run efficiently on commodity edge GPUs.

B. Adaptation Under Dynamic Network Conditions

Traditional ABR algorithms fail under rapid fluctuations, and DRL approaches [7], [8] require retraining for new conditions. This creates brittleness when deploying across heterogeneous networks.

Implementation Outlook: We will train a Deep Q-Network (DQN)–based ABR agent using the LIVE-NFLX video QoE dataset, where states include buffer occupancy, throughput history, and latency prediction. Training will use reward functions balancing rebuffering penalty, bitrate stability, and SSIM/PSNR-based visual quality metrics. Online fine-tuning at the edge will allow adaptation to local network conditions without retraining from scratch.

C. Latency Optimization Beyond Caching

While caching reduces round-trip delay [11], [12], unpredictable “flash crowd” events [10] still cause congestion. Current methods lack predictive mechanisms to anticipate demand.

Implementation Outlook: We will implement AI-driven segment prefetching by training a LSTM-based demand predictor on historical access logs. This predictor will pre-load segments into edge caches before request bursts occur, targeting an end-to-end latency budget of 200–300 ms for live streams.

D. Privacy and Security at the Edge

Frameworks like REACT [15] show the trade-off between privacy-preserving analytics and added latency. Most current works lack integration of lightweight security.

Implementation Outlook: Our deployment will incorporate Trusted Execution Environments (TEEs) for GPU resource isolation (building on [13]) and homomorphic encryption for feature vectors exchanged with cloud nodes. This allows secure inference while keeping added latency under 50 ms per segment.

E. Standardization and Interoperability

Surveyed works often target specific protocols (e.g., HTTP/2 [2], QUIC [6]) or proprietary hardware [3]. A lack of standard APIs for AI-inference at the edge limits interoperability.

Implementation Outlook: We propose designing the pipeline with open-source components only: gRPC-based communication for edge–cloud coordination, FFmpeg for transcoding, and PyTorch/TensorFlow models for inference. This ensures portability across heterogeneous edge infrastructures.

V. CONCLUSION

This paper has presented a comprehensive survey of edge-based video streaming techniques, with a focus on AI-powered upscaling, adaptive delivery, and latency reduction. By synthesizing eighteen recent contributions, we identified key advancements in reinforcement learning–based adaptive bitrate control, deep learning–driven super-resolution, and intelligent caching at the edge. A hybrid model was proposed that integrates computational intelligence across four modules, AI-based upscaling, DRL-

driven adaptive delivery, predictive edge caching, and latency-aware transport optimization.

The survey and proposed framework together emphasize that edge nodes should evolve from passive caches to intelligent agents capable of real-time decision making, prediction, and optimization. This transition is critical for ensuring scalable, low-latency, and high-quality video delivery in bandwidth-constrained and mobile-first environments.

The current limitation of this work lies in the absence of quantitative validation and benchmarking. While surveyed studies, report promising improvements such as 47% latency reduction, 60% bandwidth savings, and 92% user satisfaction, the proposed architecture requires empirical testing on real datasets and network traces.

Future work will focus on implementing the framework in a containerized edge testbed using open datasets such as LIVE-NFLX and DIV2K. Evaluation will include latency, SSIM/PSNR quality scores, rebuffering rate, and QoE metrics. Further research will also explore privacy-preserving federated reinforcement learning and standardized APIs for edge–cloud coordination, ensuring both scalability and interoperability.

In summary, the integration of edge computing and computational intelligence represents a transformative direction for next-generation streaming. By addressing open challenges in scalability, adaptability, and security, intelligent edge frameworks will enable immersive, high-quality, and responsive video services for diverse global audiences.

REFERENCES

[1] Li, Z., Mao, H., Zhu, J., Jiang, Z., Ghodsi, A., Rexford, J., & Stoica, I. (2020). Deep reinforcement learning for internet congestion control
[2] Xu, M., Yu, H., Pan, J., & Sun, L. (2018). HTTP/2-based low-latency live video streaming.
[3] B. Mengistu, “Deep-Learning Realtime Upsampling Techniques in Video Games,” University of Minnesota Morris Digital Well, 2023.
[4] Lundkvist, F. (2021). Deep upscaling for video streaming: a case evaluation at SVT. KTH Royal Institute of Technology, Stockholm, Sweden.
[5] M. Hu, Z. Luo, A. Pasdar, Y. C. Lee, Y. Zhou, and D. Wu, “Edge-Based Video Analytics: A Survey,” 2023.
[6] K. Nassisid, T. David, and K. Muhammad, “Adaptive Video Streaming over 6G Networks: Buffer Control and User Behavior Analysis,” arXiv preprint, 2024.
[7] M. Choi, A. No, M. Ji, and J. Kim, “Markov Decision Policies for Dynamic Video Delivery in Wireless Caching Networks,” IEEE Trans. Wireless Commun., vol. 18, pp. 5705–5718, 2019.
[8] T. Chen et al., “SODA: An Adaptive Bitrate Controller for Consistent High-Quality Video Streaming,” in Proc. ACM SIGCOMM, 2024.
[9] I. Yeregui et al., “Leveraging 5G Physical Layer Monitoring for Adaptive Remote Rendering in XR Applications,” 2025.
[10] Dharuman et al., “Edge-Optimized HLS/DASH and ML-Based CDN Optimization,” 2023.
[11] STL Partners, “3 Reasons Why Edge Will Change Video Streaming,” STL Partners, 2025.
[12] Muvi, “The Role of Edge Computing in Video Streaming,” Muvi, 2025.
[13] M. Bilal and M. Erbad, “Edge-Based Transcoding and Containerization for Video Streaming,” 2017.
[14] J.-H. Choi et al., “Dynamic Video Delivery Using Deep Reinforcement Learning,” 2025.
[15] A. Ghosh, S. Iyengar, S. Lee, A. Rathore, and V. N. Padmanabhan, “REACT: Streaming Video Analytics on the Edge,” in Proc. 8th ACM/IEEE Conf. on Internet of Things Design and Implementation (IoTDI), 2023.
[16] National Science Foundation, “Adaptive Bitrate Control with Edge-Aware AI,” NSF, 2025.
[17] M. Bilal et al., “FoV-Aware Edge Caching for 360° Video,” 2018.
[18] ElasticEdge, “Kubernetes-Based Elastic Edge Frameworks for Live Analytics,” 2022.