Enabling Resource-Efficient AI Video Analytics with Hierarchical Codecs

Guendalina Cobianchi†
V-Nova Ltd, London, UK
guendalina@v-nova.com

Manzoor Razaak
V-Nova Ltd, London, UK
manzoor.razaak@v-nova.com

Ben Cope
Intel Corp.
ben.cope@intel.com

Palanivel Guruvareddiar
Intel Corp.
palanivel.guruvareddiar@intel.com

ABSTRACT

The demand for intelligent video analytics in applications such as smart cities, smart retail and media delivery is driving the need for pipelines that are both bandwidth- and compute-efficient. While prior work has focused on optimizing inference models and memory transfers, little attention has been paid to optimizing the data format itself; this paper addresses that gap. Conventional codecs (e.g. AVC) are widely used as-is. Yet, they waste resources on full-resolution decoding and preprocessing, limiting the overall supported number of channels at the system level, when AI inference typically requires only lower resolution or regions of interest (ROI). Hierarchical codecs address this by exposing multiple fidelity layers, enabling low-resolution inference while preserving access to higher detail when needed and reallocating power and resources (e.g., memory bandwidth) to demanding AI tasks without compromising accuracy.

This paper evaluates two standardized approaches. MPEG-5 Low Complexity Enhancement Video Coding (LCEVC) [1] overlays lightweight enhancement layers on top of a base codec, reducing complexity while preserving perceptual and inference quality. SMPTE VC-6 [2] is a hierarchical intra-frame codec designed for scalable decoding and fetching, random access, and selective data access. Using Intel's Deep Learning Streamer framework [3] and MobileNetV2-SSD [SSD] object detection AI model [4] on UHD traffic sequences, we show that **LCEVC reduces processing pipeline time by 33-52% on CPU, 66% on GPU**, and cuts data transfer by 20-25%, while **VC-6 achieves order-of-magnitude efficiency gains** in processing time and in data transfer through selective frame and resolution fetching.

Methodology: Intel DL Streamer Video-Analytics

Experiments were conducted using Intel DL Streamer on high-resolution traffic clips from the VisDrone dataset [5], processed with SSD (448x256 input). The codec choice reflects their relevance to AI inference applications, where latency, compute, and memory constraints dominate; all tested codecs (AVC, VVC, LCEVC, VC-6) have real-time encoding/decoding solutions available, making them practical candidates for deployment. Whilst base codecs, such as AVC and VVC, offer some scalability features (e.g. SVC or subpictures); the focus of this work is on the standardized LCEVC and VC-6 hierarchical approaches.

Two criteria guided the evaluation: (1) preservation of inference accuracy and (2) reduction in decode, scaling, and detection time. For hierarchical codecs, only base layers or selected quality levels were decoded when possible. More details in Appendix [6].

Use case I: LCEVC enhancing AVC and VVC

CPU-Based Evaluation

LCEVC was tested with AVC (x264 [7], 30 Mbps) and VVC (VVenC [8], 15 Mbps) base codecs. In LCEVC pipelines, only the HD base layer was decoded.

Table I shows that LCEVC reduced CPU compute time by 33% for AVC and 52% for VVC with performance gains in decoding, scaling, and detection (all stages). Accuracy remained consistent across codecs, with average detected objects being 650.7 for AVC and VVC, 651.3 for LCEVC-AVC, and 650.3 for LCEVC-VVC. The LCEVC-enhanced codec achieved a precision of 98.6%, comparable to the native ones, confirming that LCEVC preserves AI-relevant visual features. Recall and F1-score were omitted, as the SSD model was used only for inference on an unseen dataset.

Component	AVC	LCEVC AVC	*Savings (%)*	VVC	LCEVC VVC	*Savings (%)*
Decoder	45.9	30.5	*-34%*	59.3	74.1	*-53%*
Detection	11.2	8.3	*-26%*	21.4	13.5	*-37%*
Misc.	9.5	5.9	*-38%*	26.2	11.7	*-55%*
Total	**66.6**	**44.7**	***-33%***	**207.0**	**99.3**	***-52%***

Table I: AVC/VVC vs. LCEVC AVC/VVC CPU time comparison (sec.)

GPU-Based Evaluation (with hardened codec support)

To assess scalability, tests were run on an Intel Core Ultra 7 258V [9] using the hardened AVC support. Each clip was looped 25x to form long videos, processed through eight asynchronous inference streams with a 5-frame interval and object tracking enabled.

Results (Table II) show that across diverse resolutions, LCEVC-AVC achieves consistently faster pipeline execution, with a 66% overall reduction compared to native AVC. With LCEVC, the GPU decode load decreases nearly 4x, and CPU-side inference is accelerated thanks to lighter scaling to model resolution. The smaller base layer also reduces memory bandwidth and data transfer demands, further contributing to the overall reduction in pipeline time and enabling higher AI stream density.

Clip Resolution	AVC	LCEVC AVC	*Fps gain*	AVC	LCEVC AVC	*Savings (%)*
	Total fps			Execution time (s)		
2688x1512	1345.0	479.2	*2.80x*	96.4	33.9	-65%
2720x1530	1396.7	486.3	*2.87x*	142.8	48.9	-66%
3840x2160	663.2	223.3	*2.96x*	355.0	119.7	-66%
Total	**3404.9**	**1188.9**	***2.86x***	**594.2**	**202.5**	**-66%**

Table II: AVC vs. LCEVC AVC GPU fps and time comparison

Since the base layers account for approximately 75–80% of the total bitrate, transmitting only the base layer reduces data transfer and storage by 20–25%, complementing LCEVC's intrinsic compression efficiency. The results demonstrate that LCEVC not only enhances both legacy and next-generation codecs but also improves end-to-end efficiency under constrained compute, storage and bandwidth conditions. The corresponding reduction in per-stream processing time directly translates into higher channel density, enabling significantly more concurrent AI inference streams on the same hardware infrastructure.

Use Case II – Selective Decoding with VC-6

SMPTE VC-6 is a fully hierarchical intra-codec that supports selective decoding, partial data recall, and massive parallelism. It introduces Levels of Quality (LoQs) and Regions of Interest (RoIs), enabling decoding and fetching only the required frames,

resolutions or portions of a frame. Intra-frame codecs are used not only in media mezzanine and contribution, but also in interactive applications such as KVM-over-IP and VDI, where low latency and frame independence are critical.

We compared AVC inter (30 Mbps), AVC intra (120 Mbps), and VC-6 (120 Mbps) at comparable qualities, simulating the common practice of sampling one frame in ten. Unlike AVC inter, which decoded all frames before discarding most, both AVC intra and VC-6 decoded only the requested frames. Furthermore, VC-6 could decode directly at a reduced LoQ, in this case LoQ3 (480x270), which is sufficient for feeding the AI model.

Component	AVC (s)	AVCi (s)	VC-6 LoQ3 (s)	*Savings vs AVC (%)*	*Savings vs AVCi (%)*
Decoder	27.9	7.8	1.0	***-97%***	***-72%***
Obj Detection	2.1	0.6	0.3	***-87%***	***-71%***
Misc.	1.6	0.5	0.1	***-96%***	***-66%***
Total	31.6	9.0	1.3	***-96%***	***-72%***

Table III: AVC, AVCi vs. VC-6 time comparison

As shown in Table III, VC-6 cut overall processing time by up to 96% vs. AVC inter and 72% vs. AVC intra, with consistent reductions across decoding, detection, and scaling. For data transfer, AVC inter decodes and transmits all frames, while AVC intra samples 1/10 frames but still transfers them at full resolution. VC-6 instead transmits only the required LoQ3 layer (8.6% of full resolution) of the sampled frames, reducing traffic to <10% of AVC intra and <4% of AVC inter, equivalent to >10x and >30x I/O savings. This confirms VC-6's suitability for large-scale and edge AI inference, where minimizing compute and I/O is critical.

Conclusion

MPEG-5 LCEVC and SMPTE VC-6 demonstrate complementary strengths for AI video pipelines. LCEVC pragmatically enhances legacy and next-generation codecs, reducing processing and data transfer while maintaining AI accuracy and bandwidth efficiency, while SMPTE VC-6 enables order-of-magnitude efficiency gains through selective frame, resolution, and ROI decoding/ fetching. Together, these results show that flexible, hierarchical data formats form a complementary toolkit for building accurate and resource-efficient AI pipelines, and highlight codec optimization itself as a promising direction to scale AI to the edge.

REFERENCES

[1] G. Meardi, S. Ferrara, L. Ciccarelli, G. Cobianchi, S. Poularakis, F. Maurer, S. Battista, and A. Byagowi, "MPEG-5 part 2: Low Complexity enhancement Video Coding Overview and performance evaluation," Applications of Digital Image Processing XLIII, vol. 11510, pp. 238–257, 2020.

[2] ST 2067-71:2024 – SMPTE Standard: Interoperable Master Format - Application 7: Studio Profile for LCEVC, https://pub.smpte.org/doc/st2067-71/20240701-pub/st2067-71-2024.pdf, Society of Motion Picture and Television Engineers (SMPTE) Std., 2024, accessed: 2025-07-21.

[3] Intel Corporation, "Intel Deep Learning Streamer (DL Streamer)," https://dlstreamer.github.io/, 2025, accessed: 2025-07-21.

[4] S. Saunack, "MobileNetV2-SSD: Implementation of Single Shot MultiBox Detector with MobileNetV2 as feature extractor," GitHub repository, https://github.com/saunack/MobileNetv2-SSD, accessed: Sep. 30, 2025.

[5] D. Du, P. Zhu, L. Wen, X. Bian, H. Lin, Q. Hu, T. Peng, J. Zheng, X. Wang, Y. Zhang, and others, "The VisDrone-DET2018 dataset: A large-scale benchmark for object detection in video captured by drones," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops, 2018, pp. 214–215. Available: http://aiskyeye.com/

[6] Appendix available here: [TBD].

[7] x264 - v0.163, LCEVC available at: https://download.v-nova.com/

[8] VVenC – v1.9.0, LCEVC available at: https://download.v-nova.com/

[9] Xe2 and Lunar Lake GPU Deep Dive, https://cdrdv2-public.intel.com/824434/2024_Intel_Tech%20Tour%20TW_Xe2%20and%20Lunar%20Lakes%20GPU.pdf