

Scalable Video Technology for the Visual Cloud

With AWS* Cloud Instance Measurements

Executive Summary

The transcoding of video data remotely in the cloud is experiencing significant growth that is driven by a variety of cloud video applications. As a result, there is a pressing need for a video coding technology that enables encoders to address the many transcoding requirements of such video applications. Scalable Video Technology (SVT) is a software-based video coding technology that allows encoders to achieve, on Intel® Xeon® Scalable processors, the best possible tradeoffs between performance, latency, and visual quality. SVT also allows encoders to scale their performance levels, given the quality and latency requirements of the target applications. The efficiency and scalability of SVT are enabled through mainly architectural and algorithmic features, and via specific optimizations for the Intel Xeon Scalable processor. SVT-HEVC and SVT-VP9 encoders are made available to the open source community via a highly-permissive BSD+Patent license, allowing adopters to reduce the time-to-market and cost-of-ownership of each of their SVT-enabled cloud video transcoding solutions. For SVT-AV1 we are changing from BSD+Patent to AOM patent and BSD-2 clause in order to create a tighter alignment with emerging codec standards. This paper explains the benefits and features of SVT, and it also presents some results that illustrate the performance quality tradeoffs of two SVT encoders: SVT-HEVC and SVT-AV1.

Introduction

Over the past decade, a slew of applications has emerged that require the sharing and/or consumption of visual content and experiences, leading to the accelerated growth in video data traffic. Examples of such applications are over the top (OTT) linear video streaming, live broadcast of user-generated video content, media analytics, and cloud gaming. The *visual cloud* is created to enable such visual applications.

In general terms, the visual cloud refers to the amalgamation of cloud hardware, software, and networking infrastructure that allows the efficient remote processing and delivery of media, graphics, and gaming content, as well as enabling some demanding applications such as media analytics and immersive media. The visual cloud supports five core services, as shown in Figure 1. Such core services are enabled via one or more of the following building blocks: decoding, inferencing, rendering, and encoding.

With the ever-increasing amount of visual data being generated from various sources, encoding has become a critical part of most visual cloud applications. Encoding is required to compress the source visual content into the least number of bits in the least amount of time, without significantly affecting the visual quality. Many of the visual compression technologies and standards (for example MPEG-2, AVC, HEVC, VP9, and AV1) that have been developed achieve high compression efficiency; however, standard-compliant encoders can be very complex, requiring large computational and memory resources. The challenge then is to achieve the best possible cost quality tradeoffs for a given application, subject to the constraints on the available cloud resources. For some high-density and/or low- power constrained visual cloud applications, hardware (based on ASICs or SoCs) encoders may be the only encoding solutions. For most other visual cloud applications, however, high-performance and high-quality software (based on CPUs such as Intel® Xeon® Scalable processors) encoders are highly desirable.

Table of Contents

Executive Summary	1
Introduction	1
SVT Benefits for the Visual Cloud	2
Scalable Video Technology: Overview	3
SVT Architecture and Features ...	3
SVT Applications: SVT-HEVC and SVT-AV1 Encoders	8
Summary	12
Access to SVT-HEVC and SVT-AV1	13
More Information	13
Notices and Disclaimers	14

Toward such an objective, a successful software-based video encoder is expected to navigate the complex landscape of conflicting requirements and provide gradual transitions in cost performance quality tradeoffs. Scalable Video Technology (SVT) was developed to include both architectural capabilities and algorithmic features that would enable a software-based video encoder to efficiently and effectively address the various requirements of the different visual cloud applications.

Visual Cloud Services

All require high performance, high scalability, and full hardware virtualization

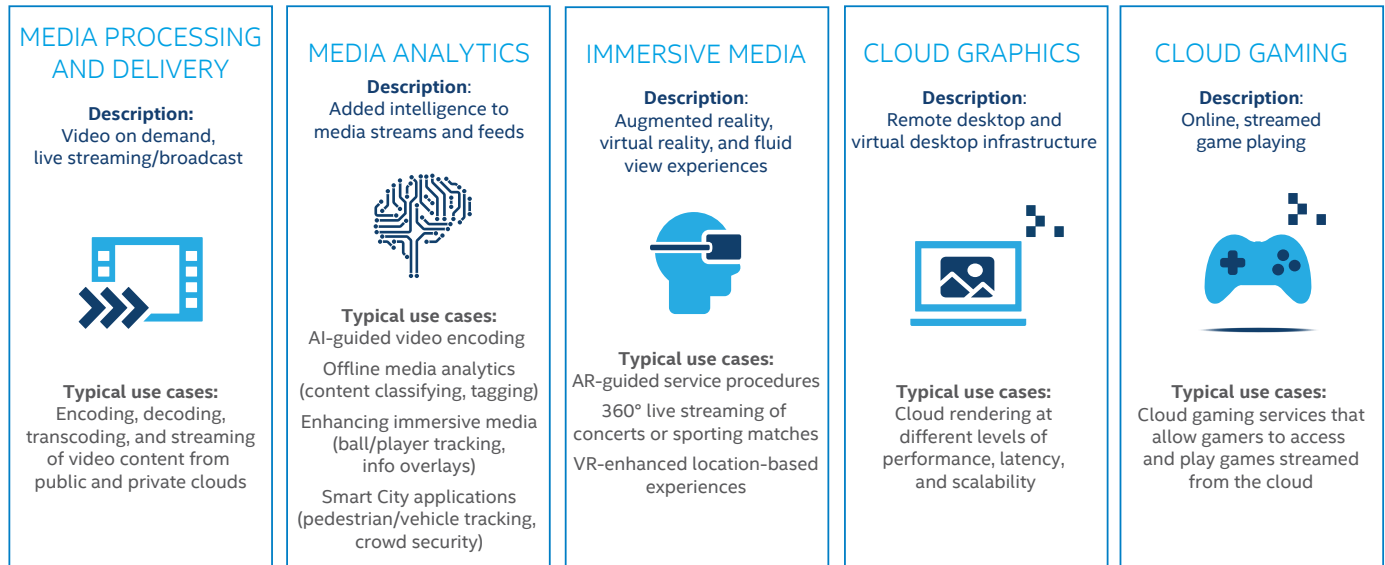


Figure 1. Five major core services supported in the visual cloud.

SVT Benefits for the Visual Cloud

SVT was developed to enable the efficient processing and encoding of multiresolution video content on Intel Xeon Scalable processors, as well as the scalable performance of visual cloud transcoding solutions. More specifically:

- SVT achieves excellent tradeoffs between performance, latency, and visual quality. In fact, SVT encoders feature multiple tradeoffs (up to 13 presets) between performance and quality and are therefore capable of addressing the requirements of various visual cloud applications including video on demand (VOD), broadcast, streaming, surveillance, cloud graphics, and video conferencing.
- SVT is highly optimized for Intel Xeon Scalable processors and Intel® Xeon® D processors with a special focus on Intel® Advanced Vector Extensions 512 (Intel® AVX-512) instructions to enhance performance. With most of the data centers using Intel Xeon Scalable processors, Intel® Xeon® processor targeted optimizations will increase the processing efficiency of workloads in the visual cloud. The large number of cores available on some of the Intel Xeon Scalable processors (for example, up to 56 cores per dual-socket processor) makes it possible to scale the performance of SVT encoders well as a function of the available computational resources. Moreover, the optimization of the SVT encoders for the Intel Xeon processor memory architecture allows for the efficient execution of memory-demanding components in the SVT algorithms. As such, cloud service providers can employ their existing infrastructure to deliver optimized workloads.
- SVT enables the development of software-based real-time encoding solutions. This provides many advantages, including the ease of integrating upgrades and/or enhancements, flexibility in creating various operating points corresponding to existing and future visual cloud applications, and the ease of interfacing with other visual processing tools and components for the development of complete, end-to-end visual workloads.
- SVT standard-compliant encoders are being made available to the open source community through the Open Visual Cloud. The Open Visual Cloud is an open source project consisting of highly optimized cloud native media, AI, and graphics components and sample reference pipelines to easily construct visual cloud services. SVT plays a critical role with the observation that encode is a required building block across all the visual cloud services. In cooperation with industry partners, Intel is helping build this ecosystem to support development of video processing solutions for the visual cloud, and is continuously optimizing such solutions for new processors. This enables a faster time to market (TTM) and reduces the total cost of ownership (TCO) of the solutions for customers.

Scalable Video Technology: Overview

A typical video encoder would involve core encoder modules (or processes) and peripheral modules (or processes), as illustrated in Figures 2 and 3. Examples of core modules include an analysis module where spatiotemporal characteristics of the input pictures are analyzed and described through various parameters, a mode decision module that is responsible for the partitioning and coding mode decisions, an encode/decode module that is responsible for the compliant or normative encoding of the pixels, and an entropy coding module that produces the compliant bit stream. The peripheral modules include, for example, pre-processing tasks such as de-noising, resizing or chrominance sub-sampling, and various rate control algorithms that would allow applications to effectively utilize the core encoder. The Scalable Video Technology (SVT) was developed to increase the scalability of the core encoder and improve its tradeoffs between performance and visual quality, particularly for high-resolution video content for example, 4K and 8K. SVT introduces novel standard-agnostic architectural features and algorithms to increase the encoder's performance and improve its visual quality (simultaneously) for any given level of resources.

The SVT architecture allows for the encoder core to be split into independently operating threads, each thread processing a different segment of the input picture that run in parallel on different processor cores, without introducing any loss in fidelity. This SVT architecture is standard agnostic; in other words, it can be applied for the development of encoders that are compliant with different standards. SVT allows any standard-compliant encoder to scale its performance properly in response to the compute and memory constraints, while maintaining a graceful degradation in video quality with increasing performance.

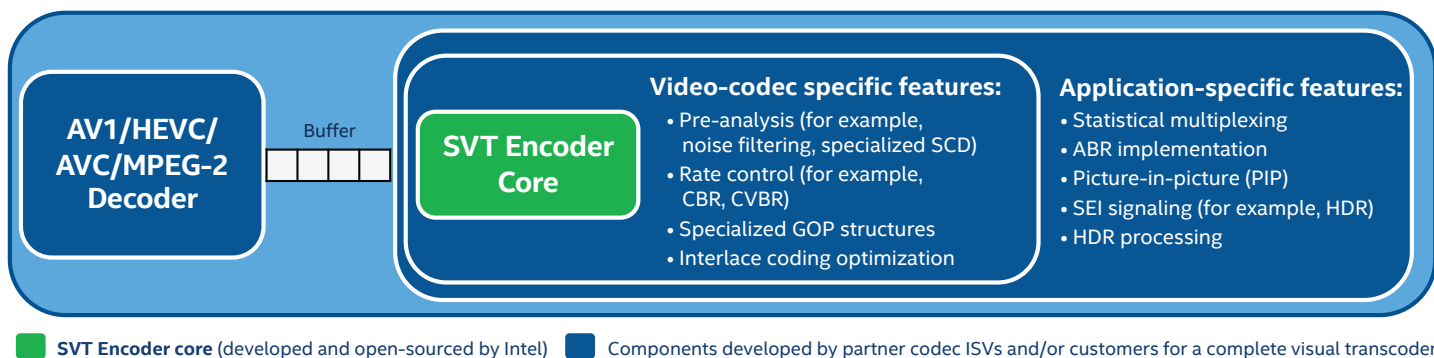


Figure 2. SVT encoder core and visual transcoder components

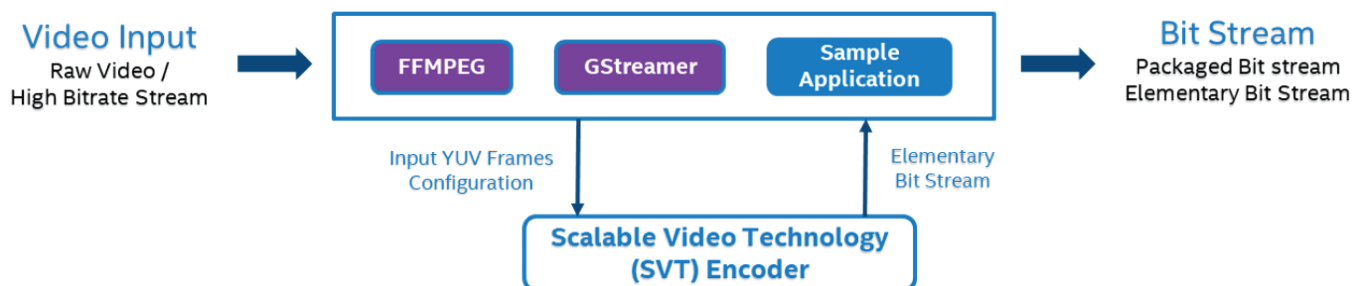


Figure 3. Interface between an SVT encoder and a sample application.

SVT Architecture and Features

In the following section, the key features of SVT are discussed in more detail. The SVT architecture and SVT's three-dimensional parallelism are followed by a description of the two main SVT features: (1) Human Visual System (HVS) optimized classification for (a) data-efficient processing, (b) computationally efficient partitioning and mode decision, and (c) bit rate reduction; and (2) Resource adaptive scalability.

SVT Architecture

The SVT architecture is designed to maximize the performance of an SVT encoder on Intel Xeon Scalable processors. It is based on three-dimensional parallelism. SVT supports process-based parallelism, which involves the splitting of the encoding operation into a set of independent encoding processes, where partitioning/mode decisions and normative encoding/decoding are de-coupled. SVT also supports picture-based parallelism using hierarchical GOP structures. Most importantly, however, is SVT's segment-based parallelism, which involves the splitting of each picture into segments and processing multiple segments of a picture in parallel to achieve better utilization of the computational resources with no loss in video quality. Each of the three parallelism aspects in SVT is discussed in more detail in the following sections.

Process-based parallelism

As shown in Figure 4, the SVT encoder's operation is split into a number of independent processes, namely analysis, partition/mode decision, encoding and decoding (reconstruction) and entropy coding. From an execution perspective, the SVT encoder is divided into execution threads, with each thread executing one encoder task. Threads are either data processing oriented (designed to process mainly data), as in the case of the analysis process, for example, or control oriented (designed mainly to synchronize the operation tasks of the encoder). The communication between processes is designed to facilitate parallel processing on multicore processors. As a result, different processes (or multiple instances of the same process) could be invoked simultaneously, with each process (or instance of the same process) running on a separate core of the processor to process a different picture or a different part of the same picture. This particular design aspect of the SVT architecture is at the core of the ability of SVT to scale appropriately with, and fully exploit, any additional resources (for example, more cores, higher frequency, bigger cache, and higher I/O bandwidth).

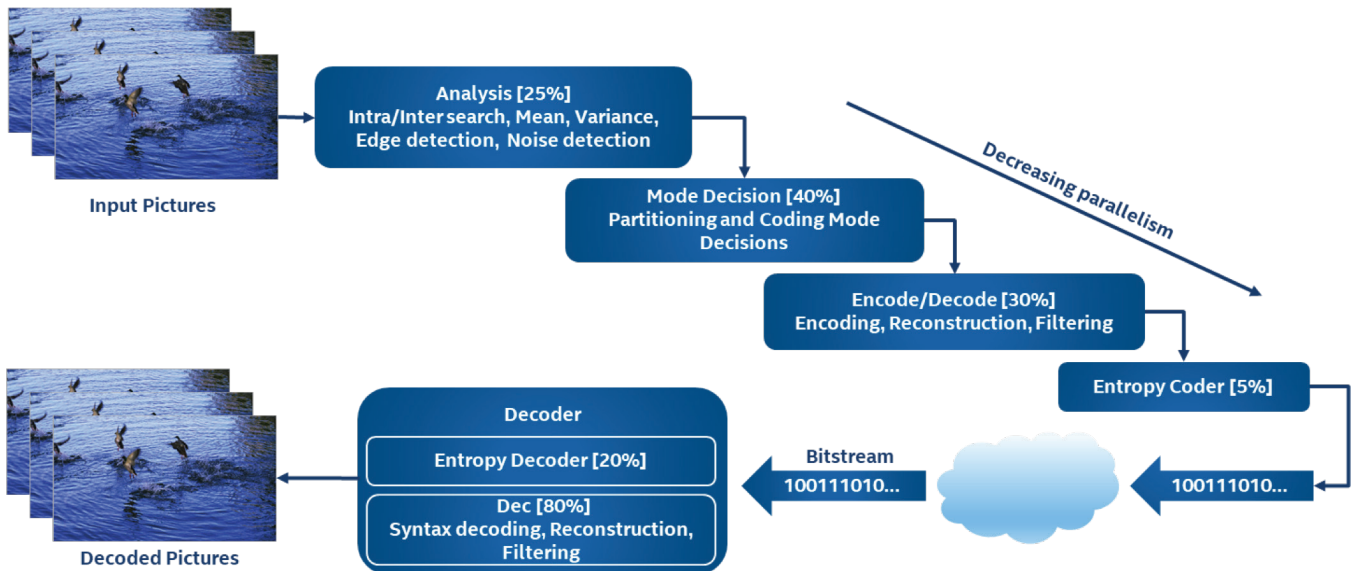


Figure 4. Encoder/decoder process dataflow, with typical CPU loads for SVT-HEVC indicated for each process¹.

Picture-based parallelism

SVT allows for the video pictures to be organized into periodic groupings, where each group is referred to as a group of pictures (GOP). The GOP itself consists of a number of periodic mini-GOPs. The pictures in a mini-GOP are organized into a hierarchical prediction structure where some pictures serve as reference pictures for another subset of pictures. The latter in turn serve as reference pictures for another subset of pictures, and so on. As a result, the encoding of pictures in a given subset could proceed simultaneously as soon as the reference pictures become available, hence the picture-level parallelism feature of SVT.

Segment-based parallelism

Picture-level parallelism may not be sufficient to fully utilize the available resources, as the encoding of a given picture has to wait for the reference pictures associated with that picture to become available. A better utilization of the computational resources could be achieved if the processing of a given picture was performed by processing different parts of the picture simultaneously. First, the picture is split into smaller parts, each referred to as segments. An example of such splitting is shown in Figure 5, where the picture is split into 40 segments. All such segments could then ideally be processed by separate processor cores simultaneously. However, there are several dependencies between the segments in the picture, in that the processing of a given segment naturally requires the top and left neighboring segments to be processed first. As a result, the processing of the segments then needs to be performed in a wavefront manner, where the processing of a given segment would not start until the top and left neighboring segments were processed. In Figure 5, the lightly shaded segments are the segments that have already been processed, and the remaining darker-shaded segments are the segments that could be processed in parallel. It follows then that, in this arrangement, segments could be processed in parallel without any loss in video quality; that is, the resulting bit streams would be the same, regardless of the number and size of the segments of the employed configuration, for as long as the encoding dependencies are respected.

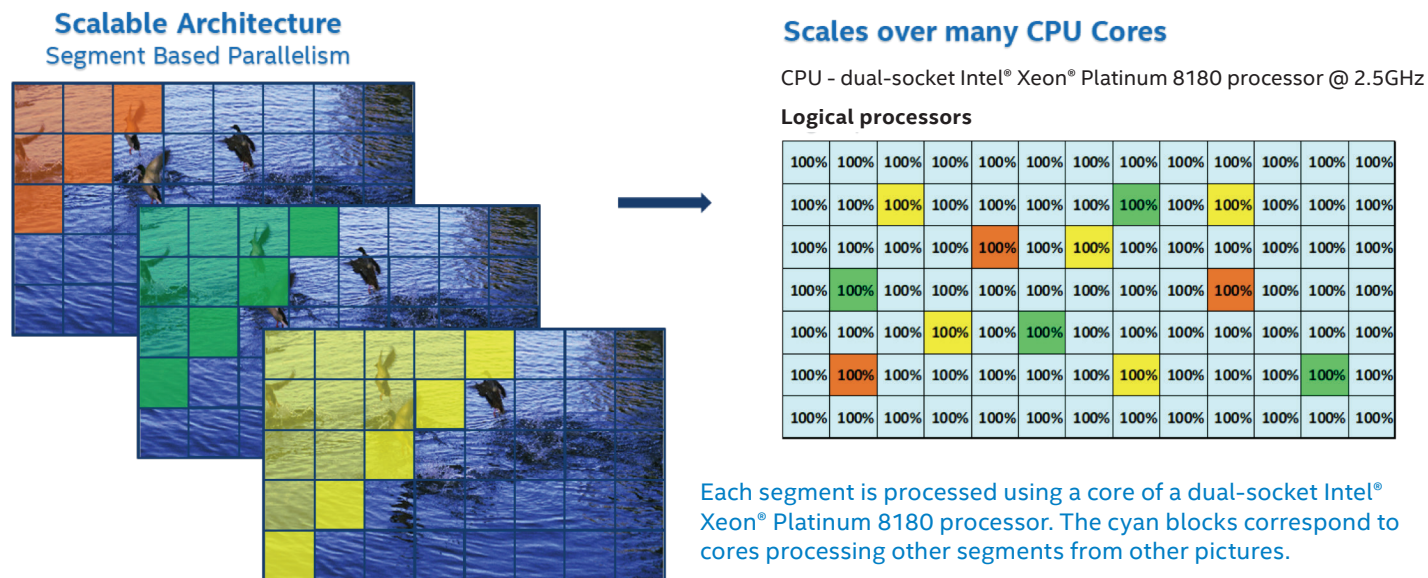


Figure 5. Example of segment-level parallelism.

SVT Features: HVS-Optimized Classification

HVS-based classification allows the SVT encoder's features to be optimized based on well-established characteristics of the human visual system (HVS), as well as feedback from extensive visual quality evaluations. The underlying principle is very similar to the noise masking principle used successfully in audio processing and coding. However, the HVS is unfortunately much more complex than the human auditory system, making HVS-based classification very dependent on the results of extensive/expensive viewing experiments. Nonetheless, SVT's classification has so far been quite successful at identifying many areas of the input video where the processing/encoding's accuracy levels could be lowered substantially, while yielding few or no perceivable visual artifacts. The SVT classification consists of mapping each block of a picture into a unique class based on its spatiotemporal characteristics, and then encoding each class of blocks with the lowest level of accuracy, while introducing the minimum number of visual artifacts. SVT's HVS-optimized classifier takes as input the video pictures and provides as output various classes, with each class used to improve the tradeoffs of one or more of the applicable mode decision (MD) and encode/decode (EncDec) features (See Figure 6). Example applications of HVS-optimized classification are discussed in the following section.

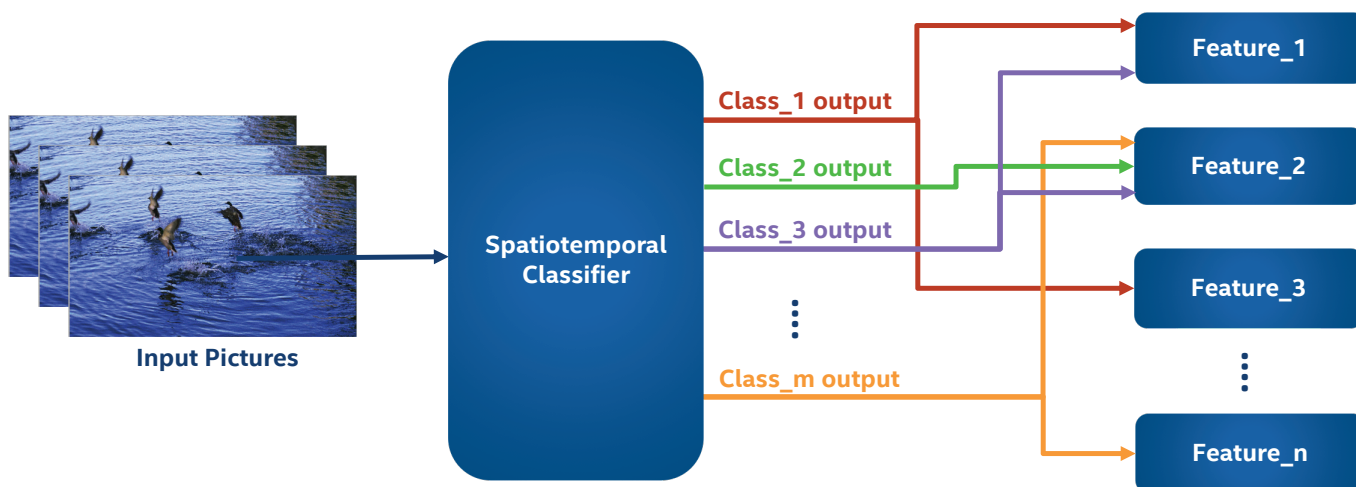


Figure 6. HVS-optimized features based on class outputs from the SVT spatiotemporal classifier.

HVS-optimized classification for computation-efficient partitioning/mode decisions

The mode decision process in the encoder is responsible for generating the partitioning and coding mode decisions. The main question in this process is how to test/consider the least number of region partitions and corresponding coding modes, while still keeping any degradation in visual quality to a minimum. The SVT encoder addresses this problem through use of the HVS-optimized classifier primarily where the number and type of tested partitions and corresponding modes would depend on some of the classifier outputs. The classifier output is also used to favor (through carefully designed biases) the partitions and coding modes that would minimize the appearance of video quality artifacts. The two approaches are outlined in more detail in the following section.

Classification-based selection of partitioning and coding mode decision algorithms

As illustrated in Figure 7, SVT allows for the ability to select, for a given set of classifier output values (or, equivalently, a region or area type), a partitioning and coding mode decision algorithm from among a set of such algorithms. More specifically, the SVT encoder assigns to each classifier output a partitioning and coding mode decision algorithm in such a way that the overall computational cost is closest to the target computational cost for the whole picture, while mitigating any potential objectionable visual artifacts that are likely to appear as a result of the reduction in the number of tested partitions and/or coding modes.

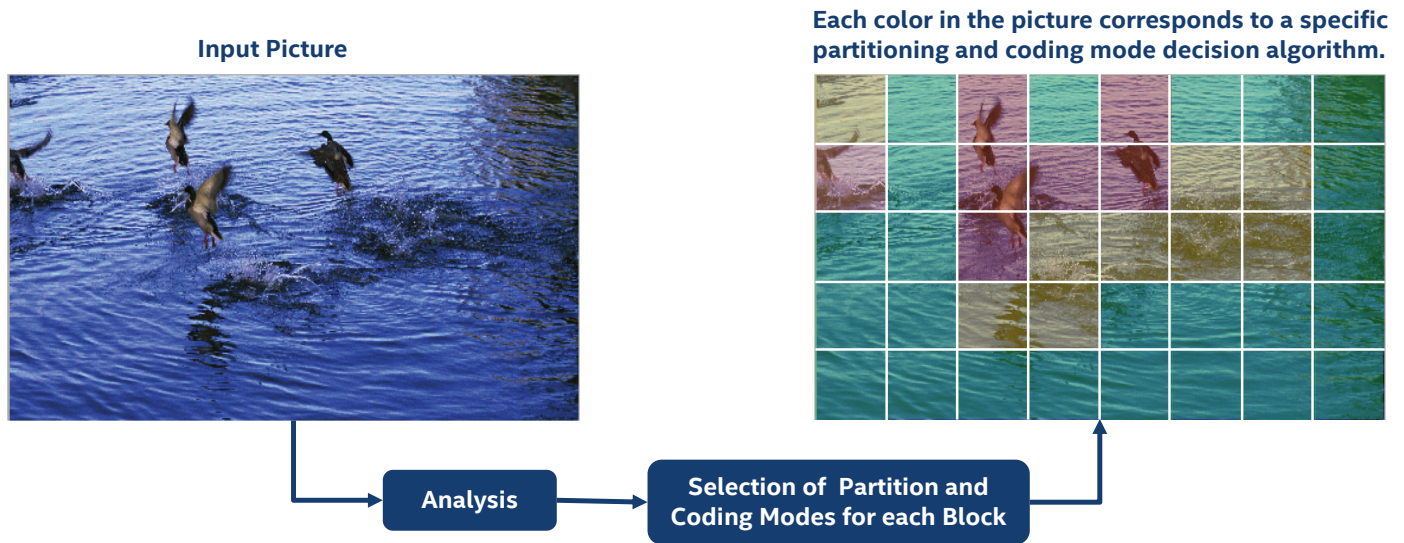


Figure 7. An example of the distribution of partitioning and coding mode decision algorithms among different areas of the input picture.

Classification-based biases

Classification-based biases are introduced in the mode decision process, as shown in Figure 7, to favor the partitions and/or coding modes that are least probable to introduce visual artifacts. The use of the biases is driven by outputs of the same HVS-optimized classifier, which indicate when the region/block under consideration is susceptible to the appearance of visual artifacts as result of the selected partition and/or coding mode. For example, the intra/inter bias favors the use of fewer intra modes in the high-temporal layers to better propagate the sharpness from the base layer. The bias is used, except in some areas (signaled by classifier outputs) where failed motion estimation, fades, edges, logo, and so on, are detected.

HVS-optimized classification for better tradeoffs between rate and visual quality

The HVS-optimized classifier could also be used at the encode/decode process, as illustrated in Figure 8. This would avoid the aggressive coding of areas where the visual quality artifacts would be easy to detect by the human eye, and to be more aggressive in the coding of areas where it would be easy to mask visual quality artifacts. Examples of algorithms that take advantage of HVS-optimized classification are the quantization parameter (QP) modulation algorithm and the perceptual masking algorithm, which we describe next.

HVS-optimized classification → Average of 17% in bit rate savings for 4Kp60/10-bit video content²



Figure 8. Example of how bit rate savings are achieved using HVS-optimized classifications.

QP Modulation Algorithm

The QP modulation (QPM) algorithm modulates the quantization parameter for different regions of the picture based on multiple classifier outputs that are dependent on the values of the prediction distortion, flatness, and grass and background area detection signals, among other HVS-sensitive parameters. The QPM targeted classifier allows the encoder to generate an initial QP offset, for each region/area, based on how far the spatiotemporal complexity of the block is from the average spatiotemporal complexity of the whole picture. The resulting QP offset is further refined locally, based on other classifier outputs that are driven by multiple detectors that are designed to reduce or eliminate specific visual artifacts.

Perceptual Masking Algorithm

Most of the bits in the bit stream generated by the encoder are associated with the coding of the transform coefficients. Reducing the number of bits of the coefficients without affecting the visual quality of the encoder would provide higher compression efficiency. One way to reduce the number of bits of the coefficients is to perform filtering in the frequency domain; that is, to selectively remove some of the high-frequency residual information that is not easily detectable by the human eye, with the selection being determined by the outputs of the HVS-optimized classifier. Such filtering is applied to the transform coefficients before quantization in the encode/decode process, and involves the use of well-designed weighting matrices, to weigh each transform coefficient by the corresponding weight in the weighting matrix. The weights range in value from 0 to 1; with smaller values, the chances increase that some of the coefficients would ultimately be quantized to zero. Which weighting matrix to use would depend on the output class of the HVS-optimized classifier. An illustration of the algorithm is shown in Figure 9.

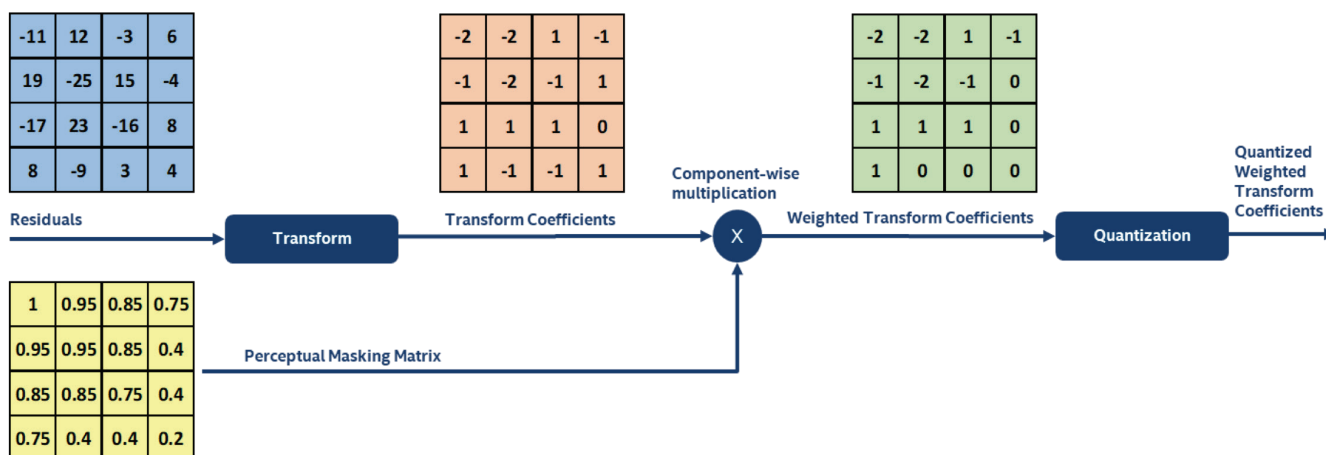


Figure 9. An example of the application of perceptual weighting for a 4x4 block.

SVT Features: Resource-Adaptive Scalability

Resource adaptive scalability refers to the ability of the SVT encoder to accommodate the constraints on the available computational and memory resources and to transition smoothly its performance as a function of the imposed constraints. SVT's ability to handle the constraints stems from SVT's two key design characteristics, namely its multidimensional parallelism in the encoding process, and its extensive parameterization of the encoder algorithms, which enables various operating points.

From a parallelization perspective, the encoder is capable of making effective use of the available computational (processor cores) and memory resources in a seamless way. More specifically, the encoder makes use of more resources if they become available simply by creating the maximum number of threads that could be executed in parallel. The factor that could limit the extent of the allowed parallelism is the amount of memory available to complete the encoding task.

From an algorithmic perspective, almost all of the key encoder algorithms are parameterized to yield different tradeoffs between performance and encoding accuracy. For example, the extent of the search area in motion estimation could be controlled size-wise in a way that would best match the target application and the intended accuracy. Another example is that the accuracy of the partitioning and mode decisions would depend largely on the maximum computational budget, which is a key parameter that controls the performance accuracy tradeoffs. Yet another example is the extent of the partial frequency approximations that depend on the values of various classifier outputs. Even the level of accuracy of the color representation for each block could vary from no color (zero chrominance) to 4:2:0, to 4:2:2, and all the way to full color (same chrominance information as input), depending again on the values of some outputs of the HVS-optimized classifier.

The multidimensional parallelism and the extensive parameterization of the SVT encoder algorithms allow for the development of multiple operating points or presets for the encoder. These presets are designed to offer the best possible tradeoffs between performance and visual quality, going from the slowest speed and highest quality preset, M0, to the fastest speed preset, M12. These presets are designed to address various application areas that range from VOD, to broadcast, premium OTT, live streaming, and gaming. Figure 10 shows the presets of an SVT encoder.

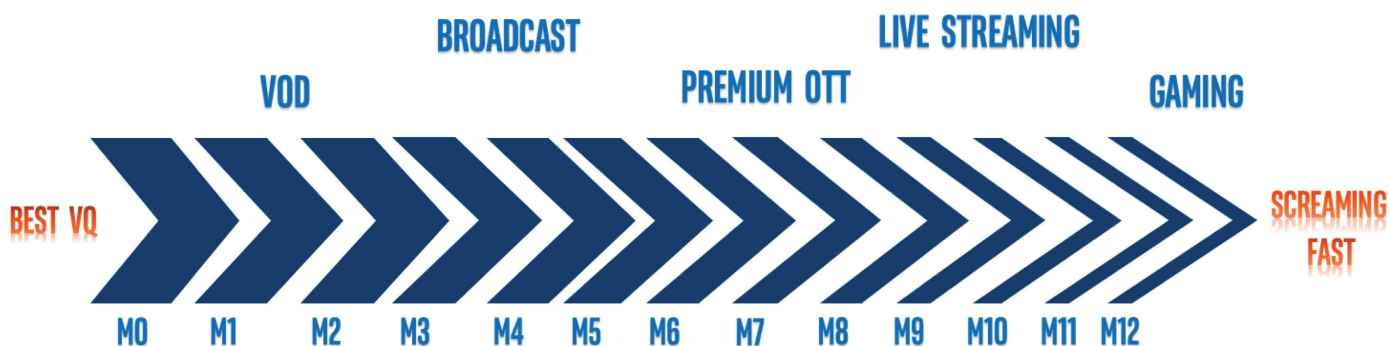


Figure 10. Example application areas for an SVT encoder.

SVT Applications: SVT-HEVC and SVT-AV1 Encoders

SVT-HEVC Encoder Core: Performance-Quality Tradeoffs

SVT was used initially in the development of the SVT-HEVC encoder that was released to the open source community on September 28, 2018. The SVT-HEVC encoder supports HEVC Main and Main10 profiles (up to Level 6.2) and video input resolutions up to 8Kp60, 4:2:0, 8-bit, and 10-bit. The SVT-HEVC encoder includes three modes: Visual Mode, PSNR/SSIM Mode and VMAF Mode. As shown in Figure 11, each of the three modes supports up to thirteen presets, from M0 to M12, providing fine granularity in the selection of the quality versus density tradeoffs. In the table, the target speed data for 4K content on a single Intel® Xeon® Platinum 8180 processor (28 cores, 2.5 GHz) is used to represent the speed level of each of the different presets. Note that fewer presets are supported for the lower resolutions, since some of the features do not yield good tradeoffs for such resolutions at very high speeds. Similarly, since many of the features are optimized for best possible tradeoffs between performance and visual quality, some of the highest speed presets are also not supported in the PSNR/SSIM and VMAF Modes.

Preset	4K Target Speed on 1x8180 (fps)	Visual Mode (-tune 0)					PSNR/SSIM Mode (-tune 1) and VMAF Mode (-tune 2)				
		480p Resolution Class	720p Resolution Class	1080p Resolution Class	4K Resolution Class	8K Resolution Class	480p Resolution Class	720p Resolution Class	1080p Resolution Class	4K Resolution Class	8K Resolution Class
M0	3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M1	5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M2	10	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M3	15	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M4	25	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M5	40	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M6	60	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M7	75	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M8	90	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M9	120	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
M10	150	x	x	✓	✓	✓	x	x	✓	✓	✓
M11	165	x	x	x	✓	✓	x	x	x	x	x
M12	180	x	x	x	✓	✓	x	x	x	x	x

Figure 11. Presets as a function of resolution and the corresponding target speed in the SVT-HEVC encoder³.

As shown in Figure 12, SVT-HEVC allows for demanding workloads to be accommodated using a compact 1U form factor with a dual-socket Intel Xeon Platinum 8180 processor that is capable of encoding up to two 8Kp50 streams. For less demanding workloads, a compact 1U form factor with dual-socket Intel® Xeon® Gold 6148 processors would be capable of encoding four 4Kp60 streams.



Compact
1U form factor



Dual-socket Intel® Xeon® Platinum 8180 Processor / RU⁴

- One 8Kp60 10-bit HDR encode stream (M11)
- Up to 2 x 8Kp50 10-bit HDR encode streams (M12)
- Up to 6 x 4Kp60 10-bit HDR encode streams (M12)

Dual-socket Intel® Xeon® Gold 6148 Processor / RU⁵

- Up to 4 x 4Kp60 10-bit HDR encode streams (M12)
- Up to 10 x 1080p60 10-bit HDR encode streams (M10)

Figure 12. Performance for SVT-HEVC (Visual mode) on the Dual-socket Intel Xeon Platinum 8180 and Gold 6148 Processors.

For workloads that require multiple resolution transcodings according to an ABR profile, where each alternate resolution version of the same video content is encoded at a different rate, the Intel® Xeon® D-2191 processor offers a power-efficient solution. In fact, as shown in Figure 13, a compact 1U form factor encoder could accommodate four such processors, and could then handle four simultaneous 1:4 ABR profiles, including a 4Kp60 decode, scaling, and the encoding of the 4Kp60, 1080p60, 720p60, and 360p60 versions of the decoded video.

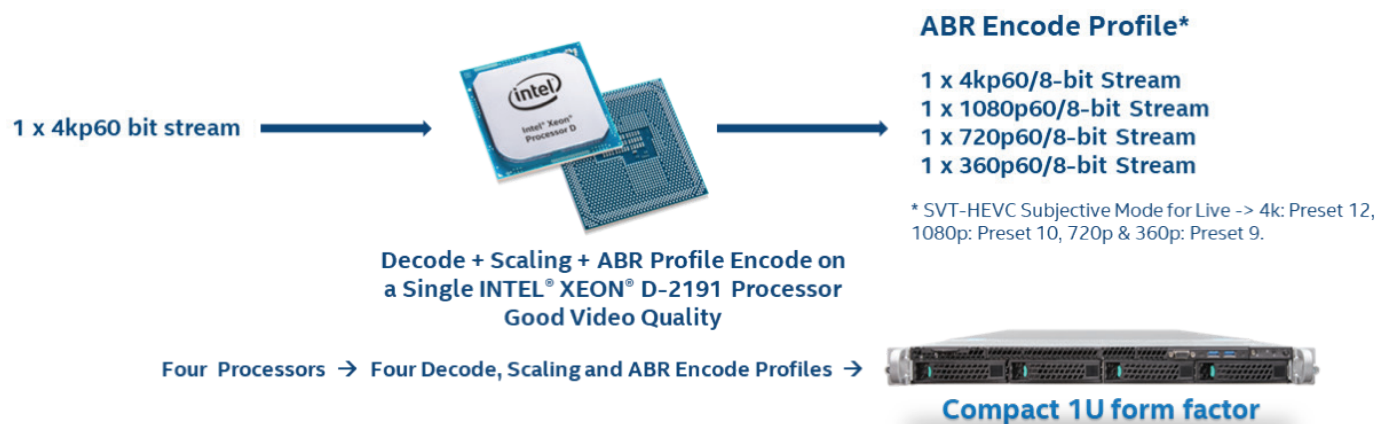


Figure 13. SVT-HEVC-based ABR transcoding on the Intel Xeon D-2191 Processor. All of the decode, scaling, and encode operations for each ABR profile run on a single processor.⁶

The following section describes both the objective quality and the visual quality levels of SVT-HEVC, relative to those of well-known open source encoders. For the best possible tradeoffs between speed and objective quality, the SVT-HEVC encoder includes an objective quality (default) mode that is optimized for PSNR and SSIM. Figure 16 compares the BD rate (relative to HM16) of the SVT-HEVC (PSNR/SSIM Mode) and x265 encoders as a function of speed. The simulation experiments used the 1080p clips of the JTC-VC test set, and in CQP mode for all encoders. As indicated in Figure 14, SVT-HEVC outperforms x265 for all presets.

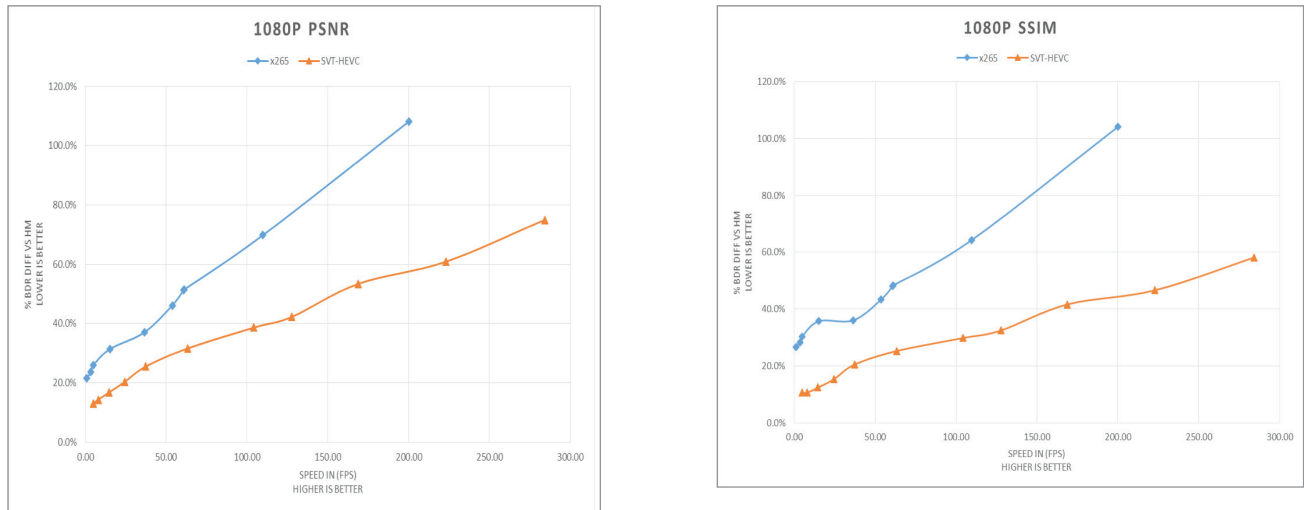


Figure 14. PSNR/SSIM versus speed performance of the SVT-HEVC and x265 encoders relative to HM16⁷.

However, SVT-HEVC has been much more optimized for the best possible tradeoffs between speed and visual quality, particularly for 4Kp60/10-bit content. The result is a huge gain in performance for the same visual quality, relative to competing open source encoders. More specifically, as evidenced by the VQ speed evaluation results of Table 1, the speedup factor can be 70:1, while maintaining similar visual quality levels to those of HM16. In this evaluation, the Visual Mode (M0 preset) was evaluated against HM16 using 15 publicly available 4Kp60/10-bit clips by 20 viewers. The viewing experiment includes a side-by-side comparison of SVT-HEVC reconstructed clips with HM16 reconstructed clips, where the viewer is asked to assign a score in the $[-5, 5]$ range, and where a negative score is in favor of HM16, and a positive score is in favor of SVT-HEVC. Table 1 below indicates that M0 of SVT-HEVC Visual Mode achieves similar visual quality (0.0 on the ADMOS scale, which is explained in Table 2) as HM16, with a 70:1 speedup factor.

Table 1. SVT-HEVC Visual Mode M0 vs. HM 16 (4Kp60 Content)⁸.

Average DMOS VQ Score	Speedup factor SVT-HEVC vs. HM 16
0.0	70

Table 2. Explanation of the average difference mean opinion score (ADMOS) scale.

$[-5,-2]$	$[-2,-1]$	$[-1,-0.25]$	$[-0.25,0.25]$	$[0.25,1]$	$[1,2]$	$[2,5]$
Much worse	Worse	Slightly worse	Same quality	Slightly better	Better	Much Better

An even stronger indication of the benefit of the 4Kp60 targeted visual quality optimizations is the result described in Table 3, where SVT-HEVC Visual Mode M11 was compared to x265 very slow mode, with similarly optimized ABR rate control algorithms. The ADMOS score is based on viewing results of a group of 21 viewers for a total of 12 4Kp60/10-bit Netflix test clips. In comparison, x265 SVT-HEVC Visual Mode achieves similar visual quality, but with a whopping speedup factor of 176:1.

Table 3. SVT-HEVC Visual Mode M11 vs. x265 very slow (out of the box, 4Kp60 content)⁹.

Average DMOS VQ Score	Speedup factor of SVT-HEVC vs. x265
-0.2	176

Despite that most of the SVT-HEVC visual optimizations targeted 4Kp60 video content, SVT-HEVC Visual Mode can still attain a sizeable speed advantage over x265, as shown in Table 4, which presents the result of a comparison between SVT-HEVC Visual Mode M9 and x265 medium, both with ABR rate control. The VQ viewing involves a pool of 20 viewers and 12 1080p test clips. With respect to the speed data, the encodings were run on an AWS* Z1D.12xlarge instance (48 virtual CPU cores, 384 GB RAM). In comparison to x265 medium, SVT-HEVC Visual Mode M9 achieves similar visual quality to x265 medium, while being 15:1 faster.

Table 4. SVT-HEVC Visual Mode M9 versus x265 medium (out of the box, 1080p content)¹⁰.

Average DMOS VQ Score	Speedup factor SVT-HEVC vs. x265
-0.1	15

To illustrate the cost advantage of SVT-HEVC over other existing solutions, the SVT-HEVC Visual Mode M9 encoder was also compared to x265 medium and another publicly available encoder, the NGCodec* HEVC/H.265 encoder D02. In this section, we will discuss several performance and cost operating points seen in public cloud instances. The performance of all encoders was evaluated on the same Intel Xeon processor-based AWS instance (C5.9xLarge) for the SW encoders, and on a field-programmable gate array (FPGA) instance (F1.2xLarge) for NGCodec's FPGA encoder. The evaluation data was developed using nine 1080p60 and three 1080p50 clips. A summary of the data that includes the number of 1080p60 streams that could be supported for each encoder, the cost per stream (based on Amazon's list prices), and the comparative VQ viewing results, is provided in Table 5.

Using the AWS instance list prices (as of December 20, 2018) for the used instances, SVT-HEVC Visual Mode M9 represents the most cost-effective solution, with a cost per 1080p60 stream per hour at \$0.38 using the AWS C5.9xLarge instance. From a visual quality perspective, SVT-HEVC Visual Mode M9 is on average at par with both the x265 medium and the NGCodec encoder. SVT-HEVC's cost advantage is, however, quite large, with cost advantages at 7:1 for the case of NGCodec, and 12:1 for the case of x265.

SVT-AV1 Encoder Core: Performance-Quality Tradeoffs

Table 5. Number of streams, cost and average DMOS of SVT-HEVC M9 relative to NGCodec's HEVC Encoder D02 and x265 medium (1080p content)¹¹.

Reference Encoder	# 1080p60 Streams C5.9xlarge	\$/1080p60/hour C5.9xlarge	# 1080p60 Streams F1.2xlarge	\$/1080p60/hour F1.2xlarge	Min Cost \$/1080p60/hour	Average DMOS
SVT-HEVC	4	\$0.38	N/A - CPU cores required	N/A - CPU cores required	\$0.38	N/A
NGCodec	N/A - FPGA required	N/A - FPGA required	1	\$2.90	\$2.90	-0.1
x265 medium	0.3	\$4.59	N/A - CPU cores required	N/A - CPU cores required	\$4.59	-0.1

The open source SVT-AV1 encoder was developed based on the same standard-agnostic SVT architecture, while also using many of the features of the open source AOMedia (AOM) Video 1 (AV1) encoder. The SVT-AV1 encoder is expected to support, by end of April 2019, video input resolutions up to 4Kp60, 4:2:0, 8-bit and 10-bit (HDR), and it will also feature multiple presets. The performance of the latest SVT-AV1 encoder is illustrated in Figure 15 below. The evaluation covers both speed and SSIM BD-rate performance. The blue diamonds on the graph correspond to the different SVT-AV1 presets, starting from preset 0 on the left up to preset 8 on the right. As an illustration of the performance of the SVT-AV1 encoder, the performance of the AOM-AV1 fastest preset (preset 8) was evaluated against AOM-AV1 preset 0. The results indicate that at the same objective quality as that of the AOM-AV1 preset 8, the SVT-AV1 encoder is more than fifty times faster than the AOM-AV1 encoder. Moreover, by evaluating the performance of the HM encoder against AOM-AV1 preset 0 (green mark), it is shown that the SVT-AV1 encoder achieves similar objective quality to that of HM while being more than sixty five times faster. The SVT-AV1 encoder at preset 8 achieves for 1080p content more than 45fps on the Intel Xeon Platinum Processor 8160¹².

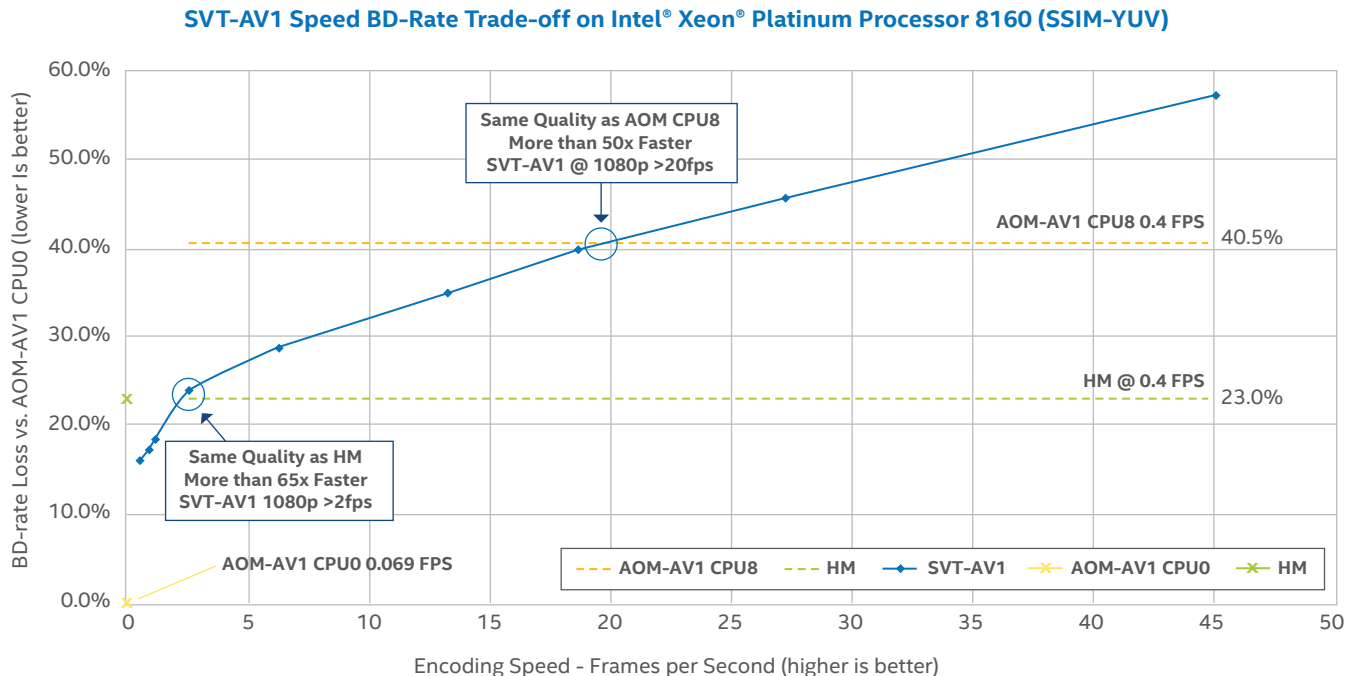


Figure 15. SVT-AV1 SSIM BD-rate vs. speed performance for the different presets on an Intel Xeon Platinum 8160 processor¹².

Summary

By adopting architectural design features that fully exploit the multicore processor capabilities, and by parameterizing features to allow for various tradeoffs between performance and visual quality, SVT becomes an ideal solution for the multitude of requirements emerging from the various visual cloud applications. The SVT encoders are optimized to run on Intel Xeon Scalable processors; they harness the multicore-enabled parallelism and the advanced low-level optimization features available on these processors. As a first SVT application, the SVT-HEVC encoder was recently released to the open source community. SVT-HEVC achieves similar quality levels to those of HM16, while being 70:1 faster, and similar quality levels to those of x265 very slow, while being 176:1 faster. Moreover, SVT-HEVC was shown to be the most cost-effective solution running on AWS instances, yielding similar or better visual quality levels than those of other encoders, while being less costly.

SVT-HEVC and SVT-VP9 encoders are made available to the open source community via a highly-permissive BSD+Patent license, allowing adopters to reduce the time-to-market and cost-of-ownership of each of their SVT-enabled cloud video transcoding solutions. For SVT-AV1 we are changing from BSD+Patent to AOM patent and BSD-2 clause in order to create a tighter alignment with emerging codec standards. They are supported by not only Intel but also its partners and customers. As a result, and as illustrated in Figure 16, SVT presents a fast and cost-effective path to productization, with continuous improvements and customizations expected by the Open Visual Cloud community over the next months and years!

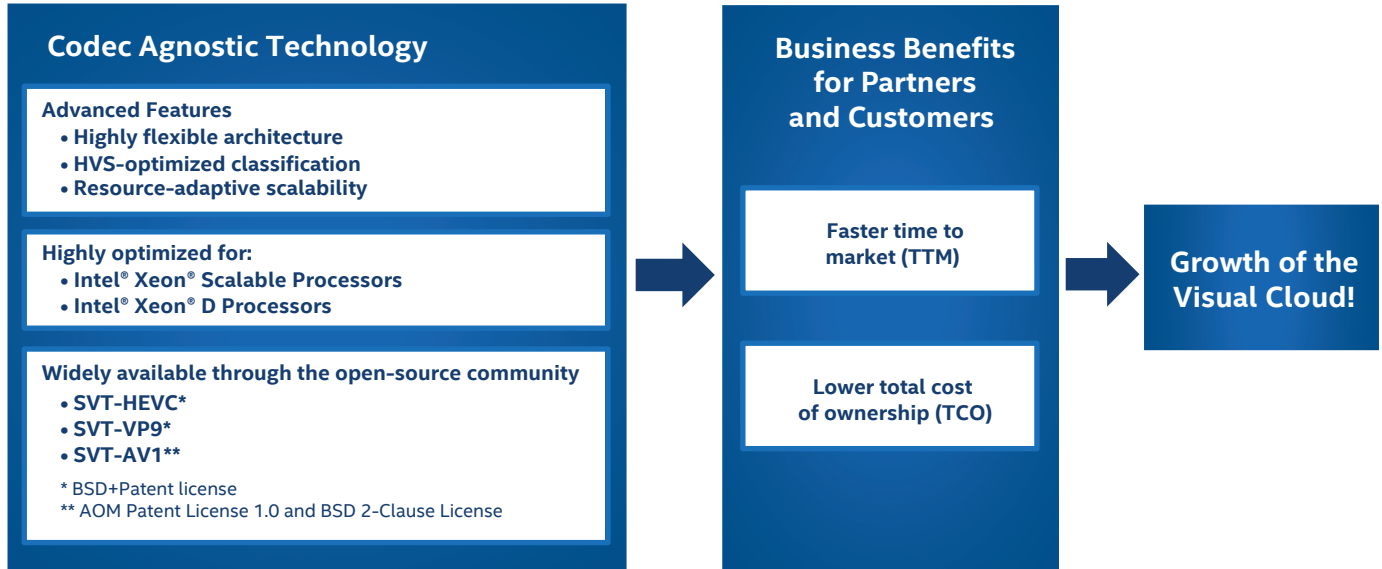


Figure 16. SVT as an enabler of the visual cloud growth.

Access to SVT-HEVC and SVT-AV1

• SVT-HEVC

- SVT-HEVC is **available** for download at <https://github.com/OpenVisualCloud/SVT-HEVC>
- To provide **feedback**, use the “issues” tab <https://github.com/OpenVisualCloud/SVT-HEVC/issues>
- Submit your contributions using the pull request functionality: <https://github.com/OpenVisualCloud/SVT-HEVC/pulls>
- To receive **release** updates, subscribe to the mailing list <https://lists.01.org/mailman/listinfo/svt-hevc>

• SVT-AV1

- Intel welcomes companies to join as major contributors to the **SVT-AV1** project. Learn more at <https://01.org/svt>
- SVT-AV1 is **available** for download at <https://github.com/OpenVisualCloud/SVT-AV1>
- To receive release updates, subscribe to the mailing list <https://lists.01.org/mailman/listinfo/svt-av1>

More Information

- **White paper** - Rethinking Visual Cloud Services for Evolving Media:
<https://www.intel.com/content/www/us/en/cloud-computing/visual-cloud.html>
- **Open Visual Cloud** - <https://01.org/openvisualcloud>
- **Video** - Innovations for Visual Cloud - Lynn Comp:
<https://techdecoded.intel.io/big-picture/open-innovation-for-visual-cloud/>
- **Video** - Software Defined Visualization: CPU Graphics Render New Realities – Jim Jeffers:
<https://techdecoded.intel.io/big-picture/software-defined-visualization-cpu-graphics-render-new-realities/>
- **Podcast** – The Visual Cloud - Lynn Comp with Daniel Newman:
<https://www.dropbox.com/s/bsv5dvschbk9wge/SMACTalk%20Intel%2020180906.mp3?dl=0>
- **Blog** - Opening a World of Innovations with Visual Cloud – Lynn Comp:
<https://itpeernetwork.intel.com/intelcdcsummit-visual-cloud/>
- **Blog** - Open Source and the Future of Visual Cloud – Imad Sousou:
<https://itpeernetwork.intel.com/open-source-visual-cloud>
- **Blog** - Intel® Open Source Libraries Playing a Major Role in Hollywood - Jim Jeffers:
<https://itpeernetwork.intel.com/intel-open-source-libraries-hollywood/>
- **Blog** - Ready. Set. Play! Gamestream, Collaborating with Intel, has announced a Cloud Gaming Private-label Solution for Telecom Operators – Lynn Comp: <https://itpeernetwork.intel.com/gamestream-intel-cloud-gaming-telecom/>

Notices and Disclaimers

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to www.intel.com/benchmarks.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804

Performance results are based on testing as of March 12th, 2019 and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer.

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Xeon, the Intel logo and others are trademarks of Intel Corporation and its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

For details on EC2 instance protections for various vulnerabilities including side-channel, please refer to Amazon security bulletins: <https://aws.amazon.com/security/security-bulletins/>.

¹ CPU load estimates based on real-time encoding (preset M11) of 4Kp60/10-bit/HDR video content.

HW Configuration: 1x Intel® Xeon® Platinum processor 8180, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, / OS: Windows Server 2016 Standard security mitigations applied, ucode 0x200004d.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit www.intel.com/benchmarks.

² Configuration: 1x Intel® Xeon® Gold processor 6148, 2.4GHz, 20 cores, turbo and HT on, 96GB total memory, 6 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 480GB, Intel SSD / OS: Windows Server 2016, ucode 0x200004d.

Sample cli [without classifier]: ebHevcEncApp.exe -encMode 10 -w 4096 -h 2160 -i in.yuv -q 27 -fps 60 -b out.bin -intra-period 55 -bit-depth 10 -tune 0 -brr 0

Sample cli [with classifier]: ebHevcEncApp.exe -encMode 10 -w 4096 -h 2160 -i in.yuv -q 27 -fps 60 -b out.bin -intra-period 55 -bit-depth 10 -tune 0 -brr 1

Performance results are based on testing as of Dec, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

³ All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Performance results are based on testing as of Sept, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

Configuration: 1x Intel® Xeon® Platinum processor 8180, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 800GB, Intel SSD / OS: Windows Server 2016, ucode 0x200004d.

⁴ Specific System Configuration and Workload Details: 2x Intel® Xeon® Platinum processor 8180, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 800GB, Intel SSD SC2BA800G4 / OS: Ubuntu Server 18.04, ucode 0x200004d.

8k60: 7680x4320 @ 60 fps encoding @ 20Mbps, 4k60: 3840x2160 @ 60 fps encoding @ 10Mbps.

Performance results are based on testing as of Sept, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

⁵ Specific System Configuration and Workload Details: 2x Intel® Xeon® Gold processor 6148, 2.4GHz, 20 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 800GB, Intel SSD SC2BA800G4 / OS: Ubuntu Server 16.04 Linux 4.4.0-87-generic, ucode 0x200004d.

4k60: 3840x2160 @ 60 fps encoding @ 10Mbps, 1080p60: 1920x1080 @ 60 fps encoding @ 7Mbps

Performance results are based on testing as of Sept, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

⁶ Specific System Configuration per System: 1x Intel® Xeon® D-2191 processor, 1.6GHz, 18 cores, turbo and HT on, 128GB total memory, / OS: Fedora 27, ucode 0x200004d. Resolution and Bitrate Details: 4k: 3840x2160 @ 13Mbps / 1920x1080 @ 4.5Mbps / 1280x720 @ 2Mbps / 480x360 @ 500Kbps

Performance results are based on testing as of Sept, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.



⁷ Specific System Configuration per System: 1x Intel® Xeon® Gold processor 6140, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, / OS: Windows Server 2016 Standard ucode 0x200004d.

Config: Encoders: SVT-HEVC: 1.3 candidate ([pr#47](#)) / x265 version 2.9. CQP mode for both. QP set for frame [27,32,37,42] across presets 0-9 for x265 and 0-10 for svt. Sample cli [svt-hevc]: ebHevcEncApp.exe -encMode 0 -tune 1 -w 1920 -h 1080 -i in.yuv -q 27 -fps 60 -b out.bin -intra-period 55 -bit-depth 8

Sample cli [x265]: x265.exe -p 9 --input-res 1920x1080 --input in.yuv --qp 27 --fps 60 -o out.bin -l 60 --input-depth 8 --tune ssim

Content: JCTVC 1080p content: [ftp://hevc@ftp.tnt.uni-hannover.de/testsequences](http://hevc@ftp.tnt.uni-hannover.de/testsequences), JVET 1080p content: [ftp://ftp.iient.rwth-aachen.de/testsequences/testset/](http://ftp.iient.rwth-aachen.de/testsequences/testset/)

Performance results are based on testing as of Dec, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

⁸ Testing conditions based on ITU-T P.913, with a pool of 20 viewers on 15 4k test clips. Encoder versions: SVT-HEVC v 1.2.0 M0 Visual Mode vs HM 16 Specific System Configuration and Workload Details:

1x Intel® Xeon® Platinum processor 8180, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 800GB, Intel SSD SC2BA800G4 / OS: Windows Server 2016, ucode 0x200004d.

Performance results are based on testing as of October, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

⁹ Video quality tests performed by Intel using visual MOS comparison of the output of both encoders using VBR at iso bitrate. Testing conditions based on ITU-T P.913, with a pool of 21 viewers on 12 4k test clips.

Encoder versions: SVT-HEVC 1.2.0 M11 / x265 HEVC encoder version 2.9 [veryslow] Specific System Configuration and Workload Details:

2x Intel® Xeon® Platinum processor 8180, 2.5GHz, 28 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, 1 x 800GB, Intel SSD SC2BA800G4 / OS: Windows Server 2016, ucode 0x200004d.

Comparison done using 12 public 4k clips, SVT-HEVC running four concurrent encodings at an average of 83fps each and x265 p8 running four concurrent encodings at an average of 0.47fps each. Bitrates tested: 7, 10 and 12 Mbps depending on the test clip.

Above comparison employed "out of box" versions of the two encoders. The results also apply to 4Kp60 content only. SVT-HEVC's advantage expected to decrease for lower resolutions and/or optimized versions of the x265 encoder.

Performance results are based on testing as of October, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure.

¹⁰ Performance results are based on testing as of November, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure. Video quality tests performed by Intel using visual MOS comparison of the output of both encoders using VBR at iso bitrate. Testing conditions based on ITU-T P.913, with a pool of 20 viewers on 12 1080p test clips.

Encoder versions: SVT-HEVC 1.2.0 M9 + sharpness ON / x265 HEVC encoder version 2.9 [medium] Specific System Configuration and Workload Details: AWS Z1D.12xlarge (48 virtual CPU cores, 38 GB RAM). Comparison done using 12 1080p clips. Bitrates tested: 1.5Mbps to 7.5Mbps depending on the test clip.

¹¹ Performance results are based on testing as of November, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure. Video quality tests performed by Intel using visual MOS comparison of the output of both encoders using VBR at iso bitrate. Testing conditions based on ITU-T P.913, with a pool of 20 viewers on 12 1080p test clips.

Encoder versions: SVT-HEVC 1.3.0 M9 + sharpness ON / x265 HEVC encoder version 2.9 [medium] / NGCodec HEVC/H.265 Encoder D02

Specific System Configuration and Workload Details: AWS C5.9xlarge instance (36 logical cores 72GB RAM), Ubuntu 16.04; f1.2xlarge (1 FPGA + 8 logical cores, 122GB RAM) AWS pricing information:

C5.9xlarge instance : <https://aws.amazon.com/ec2/pricing/on-demand/>

f1.2xlarge instance: https://aws.amazon.com/marketplace/pp/B078X2J222?cid=1541453068385&sr=0-2&ref=srh_res_product_title#pdp-pricing

¹² Performance results are based on testing as of March, 2019 and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to www.intel.com/benchmarks. Configuration: see Software and Hardware Configuration slide.

HW Configuration: 1x Intel® Xeon® Platinum processor 8160, 2.1GHz, 24 cores, turbo and HT on, 192GB total memory, 12 slots / 16GB / 2666 MT/s / DDR4 LRDIMM, / OS: Windows Server 2016 Standard security mitigations applied, ucode 0x200004d.

Encoding Configuration:

Config: Encoders: SVT-AV1: version 0.5.0 [candidate](#) / lib aom master build 106193a / hm 16.20 tag CQP mode for both. QP set av1 - [20,32,43,55], hevc [27,32,37,42] tested presets.

Sample cli [svt-av1]: SvtAv1EncApp.exe -enc-mode <preset> -w 1920 -h 1080 -i input.yuv -q <qp> -b output.bin -n 60 -intra-period 119 -bit-depth 8 -fps-num 60000 -fps-denom 1001

Sample cli [hm]: TAppEncoder.exe -wdt 1920 -hgt 1080 -i input.yuv -q 27 -fr 60 -b output.bin -f 60 -ip 120 -c encoder_randomaccess_main.cfg

Sample cli [lib aom]: aomenc.exe --codec=av1 --psnr --verbose --passes=1 --threads=48 --i420 --profile=0 --frame-parallel=0 --tile-columns=0 --test-decode=fatal --kf-min-dist=120 --kf-max-dist=120 --end-usage=q --lag-in-frames=25 --auto-alt-ref=2 --cq-level=20 --aq-mode=0 --bit-depth=8 --input-bit-depth=8 --fps=30000/1000 --width=1920 --height=1080 -o output.bin input.yuv --cpu-used=<preset>

Content Source: <https://media.xiph.org/video/derf/objective-1.1.tar.gz> Speed ran on concatenated 1080 clips.