Pre-print of a paper to be published in SPIE Proceedings, vol. 9217, Applications of Digital Image Processing XXXVII

Comparative Assessment of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC Encoders for Low-Delay Video Applications

Dan Grois*[a], Detlev Marpe[a], Tung Nguyen[a], and Ofer Hadar[b]
[a]Image Processing Department, Fraunhofer Heinrich-Hertz-Institute (HHI), Berlin, Germany
[b]Communication Systems Engineering Department, Ben-Gurion University of the Negev, Beer-Sheva, Israel

ABSTRACT

The popularity of low-delay video applications dramatically increased over the last years due to a rising demand for real-time video content (such as video conferencing or video surveillance), and also due to the increasing availability of relatively inexpensive heterogeneous devices (such as smartphones and tablets). To this end, this work presents a comparative assessment of the two latest video coding standards: H.265/MPEG-HEVC (High-Efficiency Video Coding), H.264/MPEG-AVC (Advanced Video Coding), and also of the VP9 proprietary video coding scheme. For evaluating H.264/MPEG-AVC, an open-source x264 encoder was selected, which has a multi-pass encoding mode, similarly to VP9. According to experimental results, which were obtained by using similar low-delay configurations for all three examined representative encoders, it was observed that H.265/MPEG-HEVC provides significant average bit-rate savings of *32.5%,* and *40.8%,* relative to VP9 and x264 for the *1-pass* encoding, and average bit-rate savings of *32.6%,* and *42.2%* for the *2-pass* encoding, respectively. On the other hand, compared to the x264 encoder, typical low-delay encoding times of the VP9 encoder, are about *2,000* times higher for the *1-pass* encoding, and are about *400* times higher for the *2-pass* encoding.

Keywords: H.265, High Efficiency Video Coding (HEVC), VP9, H.264, AVC, x264, low-delay video coding, coding efficiency, computational complexity.

1. INTRODUCTION

According to a recent forecast of Cisco® Incorporation [1], it would take an individual over five million years to watch the amount of video that will cross a global communication network each month in 2018, while about a million minutes of video content is expected to cross the network every second. Also, the Internet Protocol (IP) video traffic is expected to be 79% of all consumer Internet traffic in 2018, while the sum of all forms of video, i.e. TV, Internet, Video-on-Demand (VoD), and Peer-to-Peer (P2P), is expected to be in the range of 80%-90% of the global consumer traffic [1].

In order to allow such enormous video content transfer over the network, especially due to the significantly increasing demand for real-time video content, such as video conferencing, video chats, or video surveillance, efficient video coding methods should be employed. The High-Efficiency Video Coding (HEVC) standard is the latest standard developed by a Joint Collaborative Team on Video Coding (JCT-VC), which was established by both ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Pictures Expert Group (MPEG) in 2010. In turn, the final HEVC specification was approved by ITU-T as Recommendation H.265 and by ISO/IEC as MPEG-H, Part 2 [2] in 2013. When compared to its predecessor, i.e. the H.264/MPEG-AVC standard [3], H.265/MPEG-HEVC allowed achieving dramatic bit-rate savings [4]-[6] due to employing state-of-the-art technological achievements. It should be noted that H.265/MPEG-HEVC was also especially designed for the High Definition (HD) and Ultra-High Definition (UHD) video content, the demand for which is expected to dramatically increase in the near future (it is noted that that the term UHD often refers to both 3840x2160 (4K) or 7680x4320 (8K) resolutions in terms of luma samples). As a result, it is highly expected that H.265/MPEG-HEVC is going to replace H.264/MPEG-AVC in the very near future.

At the time JCT-VC group was working on H.265/MPEG-HEVC, several private companies developed their own proprietary video codecs. One of such video codecs is VP8 codec [7]-[9], which was developed privately by On2 Technologies® Inc. that was later acquired by Google® Inc. In turn, the development of the VP9 coding scheme, was mainly based on VP8 [10], [11], and it was announced as final in June 2013 [12]. However, very little is known about the coding efficiency of VP9, especially for low-delay (LD) applications in comparison to the two latest representatives of ITU-T and ISO/IEC video coding standards, i.e., H.264/MPEG-AVC and H.265/MPEG-HEVC. Therefore, the authors are confident

*This work was carried out during Dr. Dan Grois tenure of an ERCIM "Alain Bensoussan" Fellowship Programme. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant n° 246016. Correspondence Email: grois@ieee.org.

that such information is very essential, especially due to a dramatic increase in demand for real-time video communication applications. It should be noted that related comparative study for the random access configuration has been performed by the authors in their previous research [13].

This paper is organized as follows. In the next section, the selected representative encoders are introduced. *Section 3* contains a description of the test methodology and evaluation setup. Then, the detailed experimental results are presented in *Section 4*, while *Subsections 4.1.* and *4.2.* contain detailed comparative results of the H.265/MPEG-HEVC encoder vs. *1-pass* and *2-pass* VP9 and x264 encoders, respectively. A conclusion is given in *Section 5.*

2. SELECTED ENCODER IMPLEMENTATIONS

In this section, a brief overview of the selected representative encoders is presented.

2.1. H.265/MPEG-HEVC Encoder

The most popular available encoder implementation for H.265/MPEG-HEVC-based encoding [14], [15] is currently considered the HM reference software encoder [16]. As a result, the recent reference model 10 (HM 10.0) was selected for conducting performance evaluation presented in this work.

2.2. VP9 Encoder

The final VP9 bitstream format and its corresponding encoder were released by Google® Inc. on June 12, 2013 [12]. The VP9 encoder has a multi-pass encoding mode, which results in improved rate-distortion (R-D) performance. Particularly, at the first pass, a file with statistic data about every input frame is generated. In turn, at the second pass, this information is used to improve the R-D optimization process in the encoder, thereby achieving a significant decrease in the bit rate for the same video quality. When performing this so-called *2-pass* encoding, the encoding process of the whole video sequence has to be executed out twice.

According to senior developers of Google® Inc., as discussed for example in [17]-[19], the *1-pass* encoding in VP9 is currently "broken". In addition, since the *2-pass* encoding was always used during the development and testing of VP9 [17]-[19], many of the VP9 features are only useable for the *2-pass* encoding.

Therefore, in order to allow a fair comparison of VP9 with the HEVC reference model and x264 encoder, the detailed experimental results are provided for both *1-pass* and *2-pass* encoding, according to the common test conditions [20].

2.3. H.264/MPEG-AVC Encoder

For evaluating H.264/MPEG-AVC *High Profile*, authors selected the open-source x264 encoder [21]-[23], which is considered to be one of the most popular encoders for H.264/MPEG-AVC-based video coding due to its flexible trade-off between coding efficiency and computational complexity. In addition, the x264 encoder is considered to be very fast, and it can achieve real-time performance, which is of course critical for real-time applications. Further, the x264 encoder has a multi-pass encoding mode, similarly to VP9 [22]-[23].

As a result, the x264 encoder is considered to be one of the best representatives of publicly available H.264/MPEG-AVC-based encoding implementations. Particularly, the authors used one of the latest versions of the x264 encoder, i.e., the “r2334” version, which was released on May 22, 2013 [22]-[23].

3. TEST METHODOLOGY AND EVALUATION SETUP

For performing the detailed performance analysis and in order to be as fair as possible due to the significant difference in the capabilities of the individual encoders, the authors of this paper used very similar settings for all tested encoders, i.e., for the HM reference model, VP9, and x264 video encoders. Below, the test methodology and the evaluation setup are explained in detail. As R-D performance assessment, the authors used the Bjøntegaard-Delta bit-rate (BD-BR) measurement method for calculating average bit-rate differences between R-D curves for the same objective quality (e.g., for the same $PSNR_{YUV}$ values) [13], [24]. Particularly, in *Subsection 3.1.,* the HM reference software configuration is discussed, followed by the discussion of VP9 and x264 configurations, in *Subsection 3.2.*

3.1. HM Reference Software in Low-Delay Configuration

For the HM reference software encoder, a “Low-Delay P” configuration was selected [16]. The Group of Picture (GOP) size was set to 4 pictures, and the *I-picture* was set to be only the first picture of each sequence. Also, hierarchical *P pictures* were used with a Quantization Parameter (QP) increase of 1 (i.e., the quantization step size increase of 12%) between each temporal level [4]. The coding order was set to 1, 2, 3, 4. It is noted that the above test conditions were selected similarly to the test conditions presented in [4]. *Table 1* below summarizes the above-mentioned HM reference software encoder [16] configuration.

Table 1. Settings for the HM Reference Software Encoder

CODING OPTIONS	CHOSEN PARAMETER
Encoder Version	HM 10.0
Profile	Main
Reference Frames	4
R/D Optimization	Enabled
Motion Estimation	TZ search
Search Range	64
GOP	4
Hierarchical Encoding	Enabled
Temporal Levels	4
Intra Period	I-picture is the first frame only
Deblocking Filter	Enabled
Coding Unit Size/Depth	64/4
Transform Unit Size (Min/Max)	4/32
TransformSkip	Enabled
TransformSkipFast	Enabled
Hadamard ME	Enabled
Asymmetric Motion Partitioning (AMP)	Enabled
Fast Encoding	Enabled
Fast Merge Decision	Enabled
Sample adaptive offset (SAO)	Enabled
Rate Control	Disabled
Internal Bit Depth	8

3.2. VP9 and x264 in Low-Delay Configuration

The VP9 and x264 *High Profile* configuration settings are presented in *Table 2* below. It should be noted that since there is currently no official VP9 specification and no VP9 encoder manual, the authors used VP9 settings recommended by leading VP9 senior developers [25], [26], as well as the most recommended VP8 low-delay settings [7]. The reader is referred to [7], [23] for obtaining more detailed information with regard to all VP9 and x264 parameters, respectively, as presented in *Table 2*.

The QP values in the above VP9 and x264 configuration were set to be similar to those used for running the HM 10.0 encoder in order to be consistent (they are presented as $QP). It was also ensured that the *I-picture* is set to be only the first picture (i.e., the $MAX values in *Table 2* are set to relatively large numbers in order to prevent the *I-picture* to appear more than once within each video sequence).

Table 2. Selected Settings for the VP9 and x264 Encoders

CODEC	VP9	x264
Versions	**Defined as Final [12]: v1.2.0-3088-ga81bd12 of June 12, 2013**	**One of the Most Recent [22]: r2334 of May 22, 2013**
Recommended Settings	--rt --cpu-used=0 --profile=0 --threads=0 --lag-in-frames=0 --min-q=$QP --max-q=$QP --end-	--qp $QP --profile high --tune psnr --ref 4 --direct auto --weightp 2 --level 5.1 --subme 8

	usage=vbr --auto-alt-ref=0 --kf-max-dist=$MAX --kf-min-dist=$MAX --static-thresh=0 --bias-pct=50 --drop-frame=0 --minsection-pct=0 --maxsection-pct=2000 --arnr-maxframes=0 --undershoot-pct=100 --sharpness=0 --codec=vp9 --passes=$P	--b-pyramid none --bframes 0 --b-adapt 0 --merange 24 --me tesa --no-fast-pskip --trellis 2 --min-keyint=$MAX --keyint=$MAX --pass $P

It should be noted that both VP9 and x264 configurations were tuned for the best PSNR values. Also, when running the x264 encoder in the one-pass mode, the "--slow-firstpass" command was added to the command line for improving coding efficiency. In addition, it should be noted that the VP9 command line executed either with "--target-bitrate " parameter (i.e., specifying the target bit-rate) or without it, led to similar results in all cases, as well as using the term "good" instead of "rt", or setting the "cq_level" parameter to a predefined level (e.g., to be equal to $QP) [7].

4. EXPERIMENTAL RESULTS

In this work the authors focused on video conferencing test sequences, which are represented by Class E according to the common test conditions [20], as presented in *Table 3*.

Table 3. Test Video Sequences

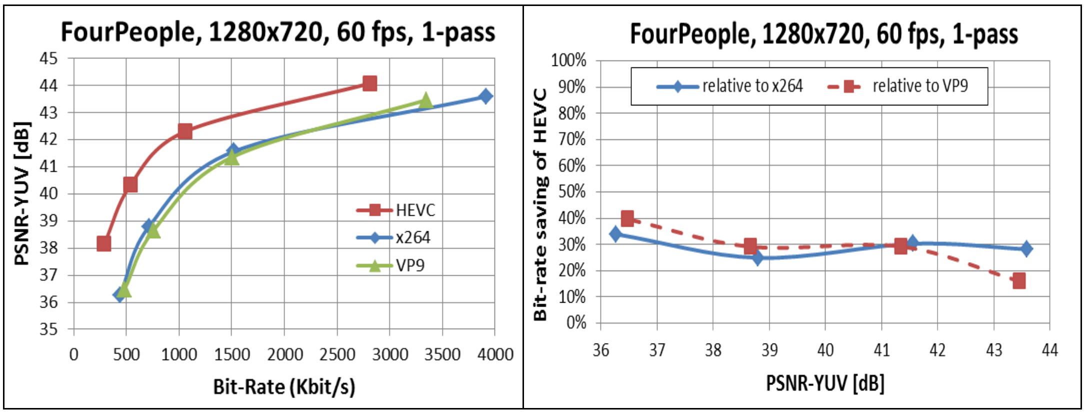
Class	Sequence Name	Resolution	Frame rate
E	FourPeople	1280x720	60fps
E	Johnny	1280x720	60fps
E	KristenAndSara	1280x720	60fps

Further, for each of these video sequences, four quantization parameter (QP) values were selected: 22, 27, 32, and 37, which are the QP values used for the *I-picture* coding of the HM reference software encoder [20]. For simplicity, 150 frames of each sequence were tested. Also, the tests were mostly carried out on computers with Intel® Core i5 CPU, 2.4 GHz, 4GB RAM.

In the following *Subsections 4.1.* and *4.2.*, the detailed comparative results of the HM reference encoder [16] vs. *1-pass* and *2-pass* VP9 [18] and x264 [22] encoders are presented, respectively.

4.1. Performance Comparison of HEVC vs. VP9 and x264 for the 1-Pass Encoding

Figure 1 below presents R-D curves of HEVC, x264, and VP9 encoders, along with corresponding HEVC bit-rate savings for typical *1-pass* encoding examples of the tested video sequences [20].



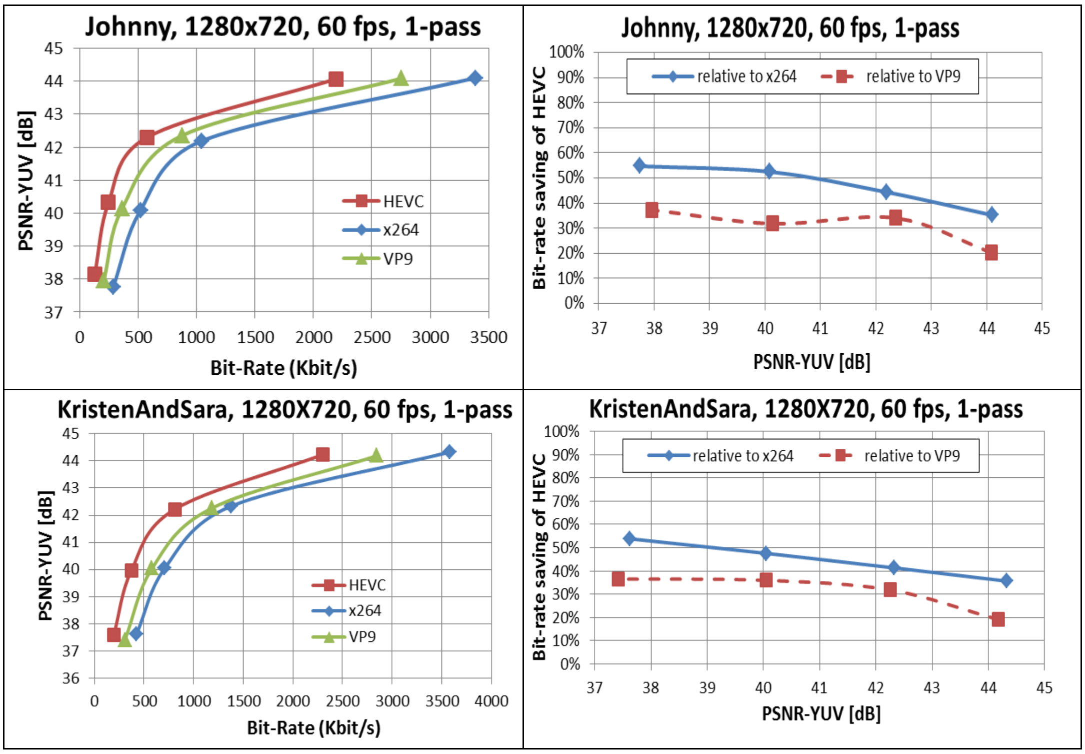


Figure 1. *1-pass* encoding mode: R-D curves and corresponding bit-rate saving plots for the tested *Class E* sequences, which refer to different video conferencing scenarios.

As it is clearly seen from *Figure 1*, the HEVC encoder provides significant gains in terms of coding efficiency compared to both VP9 and x264 encoders. Particularly, for *FourPeople* and *KristenAndSara* video sequences, the R-D curves of VP9 and x264 encoders are very close, while the calculated BD-BR savings [24] of the x264 encoder vs. VP9 encoder for the *FourPeople* video sequence are 5.8% in favour of x264.

Table 4 below presents the detailed calculated BD-BR [24]savings (*negative BD-BR values* indicate actual bit-rate savings). The average BD-BR savings of the HEVC encoder relative to VP9 and x264 encoders are *32.5%,* and *40.8%,* respectively.

Table 4. HEVC calculated BD-BR savings (Compared to VP9 and x264 *High Profile* Encoders, 1-Pass Mode)

	HEVC vs. VP9 (in %)	***HEVC vs. x264 (in %)***
Sequences/QPs	**BD-BR**	**BD-BR**
FourPeople	-31.6%	-27.6%
Johnny	-33.6%	-51.3%
KristenAndSara	-32.4%	-43.4%
Averages	**-32.5%**	**-40.8%**

Furthermore, *Table 5* below presents detailed encoding run times as an indication of the computational complexity involved for each of the tested encoders. However, it is noted that all three encoders represent different degrees of software optimizations.

Table 5. Encoding Low-Delay Run Times for Equal $PSNR_{YUV}$ (Compared to VP9 and x264 *High Profile* Encoders, 1-Pass Mode)

	HEVC vs. VP9 (in %)				***VP9 vs. x264 (in %)***			
Sequences/QPs	**22**	**27**	**32**	**37**	**22**	**27**	**32**	**37**
FourPeople	325	274	253	239	160981	208502	266549	292080
Johnny	311	254	236	230	95691	120474	165065	235111
KristenAndSara	338	291	262	250	139382	172523	231339	291056
Averages	325	273	250	240	132018	167166	220984	272749

Total Average	**272**	**198229**

According to *Table 5*, the x264 *High Profile* encoding times for the *1-pass* encoding are extremely low, and the typical encoding times of the VP9 encoder are about *2,000* times higher than those measured for the x264 encoder; on the other hand, the BD-BR savings of VP9 compared to x264 are *12.5%.* In turn, when compared to the H.265/MPEG-HEVC reference encoder implementation, the VP9 encoding times are lower by a factor of *2.72*, on average, and the BD-BR savings of HEVC compared to VP9 are *32.5%.*

Table 6 below provides a full summary of the BD-BR difference, where *negative BD-BR values* indicate bit-rate savings in contrast to positive values, which indicate the required overhead in bit-rate to achieve the same $PSNR_{YUV}$ values.

Table 6. Summarized BD-BR Low-Delay Experimental Results (Compared to VP9 and x264 *High Profile* Encoders, 1-Pass Mode)

CODEC	**HEVC**	**x264**	**VP9**
HEVC		**-40.8%**	**-32.5%**
x264	**73.5%**		**17.0%**
VP9	**48.2%**	**-12.5%**	

As shown in *Table 6*, in order to achieve the same $PSNR_{YUV}$ values of HEVC, when employing VP9 and x264 encoders, the BD-BR overhead of *48.2%* and *73.5%,* respectively, is required. It is also noted that since the fitting of R-D curves slightly differs when fitting the R-D curve of one encoder to another and *vice versa*, the product $(100 + b_1)(100 + b_2)$ for each pair (b_1, b_2) of corresponding BD-BR values (e.g., x264 vs. VP9 and VP9 vs. x264) is approximately equal to 10.000.

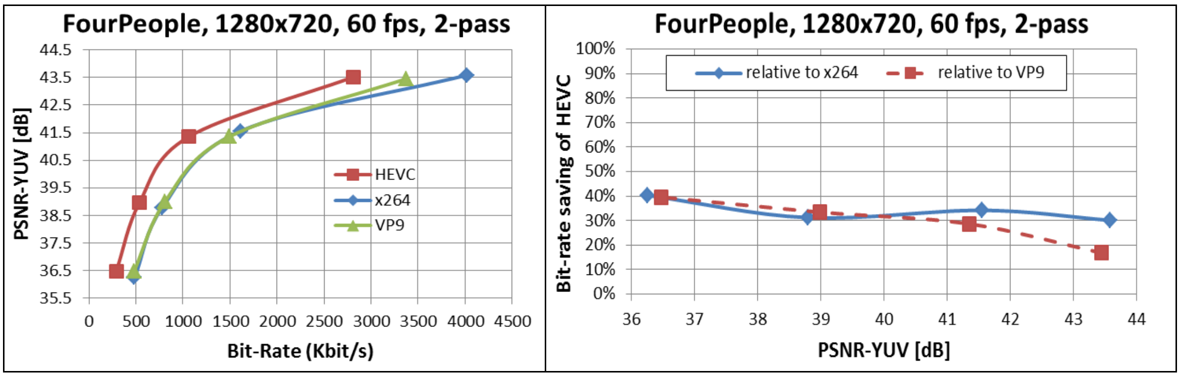
The detailed experimental results for the *2-pass* VP9 and x264 encoding vs. HEVC encoding are presented in the following *Section 4.2.*

4.2. Performance Comparison of HEVC vs. VP9 and x264 for the 2-Pass Encoding

Figure 2 below presents R-D curves of HEVC, x264, and VP9 encoders, along with corresponding HEVC bit-rate savings for typical *2-pass* encoding examples of the tested video sequences.

As it is observed from *Figure 2*, similarly to the *1-pass* encoding mode (*Subsection 4.1.*), the HEVC encoder provides significant gains in terms of coding efficiency compared to both VP9 and x264 encoders. In this case again, the Rate-Distortion (RD) curves of the *FourPeople* and *KristenAndSara* video sequences for the VP9 and x264 encoding are also very close, while the difference in BD-BR savings [24] of the VP9 encoder vs. x264 encoder for the *FourPeople* video sequence are *2.5%* in favour of VP9.

In addition, *Table 7* presents detailed experimental results with regard to the HEVC bit-rate savings for the same objective quality, such as the $PSNR_{YUV}$.



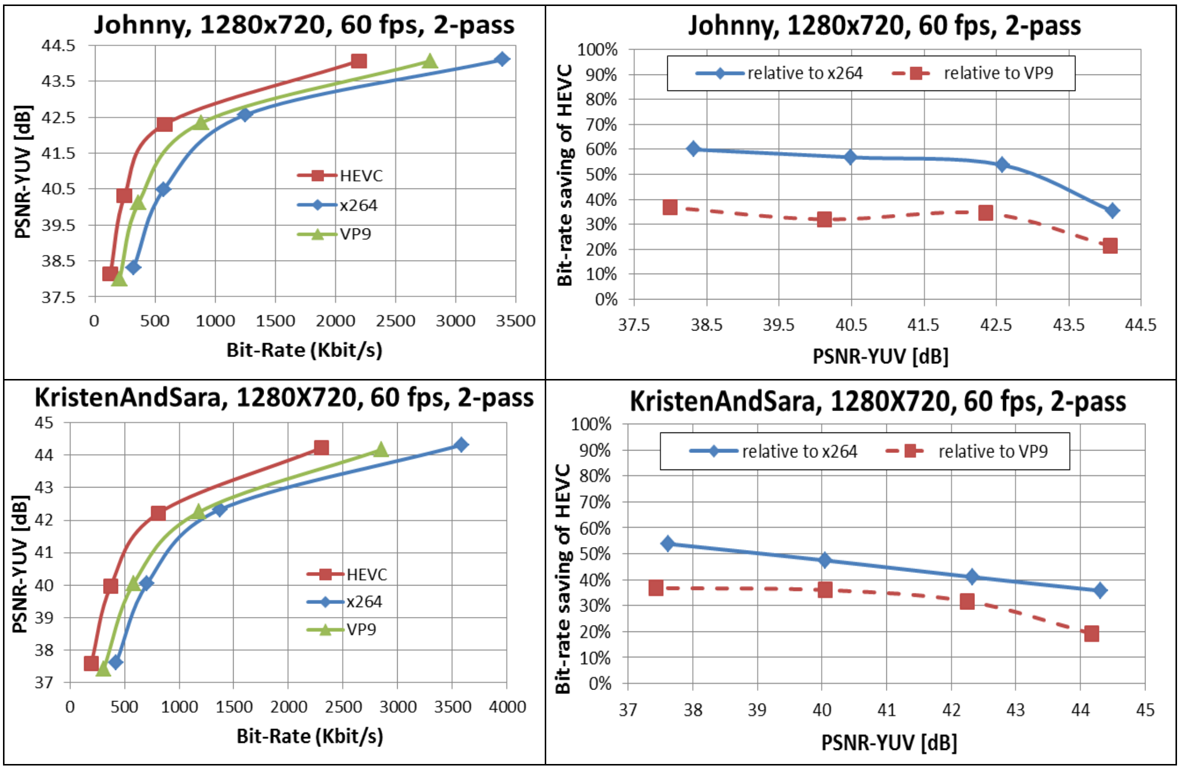


Figure 2. *2-pass* Encoding Mode: R-D curves and corresponding bit-rate saving plots for the tested *Class E* sequences, which are representing different video conferencing scenarios.

Table 7. HEVC Bit-Rate Savings for Equal $PSNR_{YUV}$ (Compared to VP9 and x264 *High Profile* Encoders, 2-Pass Mode)

	HEVC vs. VP9 (in %)					***HEVC vs. x264 (in %)***				
Sequences/QPs	**22**	**27**	**32**	**37**	**BD-BR**	**22**	**27**	**32**	**37**	**BD-BR**
FourPeople	16.8	28.7	33.5	39.6	**-31.1**	30.1	34.2	31.2	40.3	**-32.7**
Johnny	21.4	34.5	32.0	36.8	**-34.0**	35.3	53.8	56.9	60.0	**-50.6**
KristenAndSara	19.3	31.7	36.0	36.8	**-32.6**	35.8	41.1	47.4	53.8	**-43.4**
Averages	**19.2**	**31.6**	**33.8**	**37.7**	**-32.6**	**33.7**	**43.0**	**45.2**	**51.4**	**-42.2**
Total Average	**30.6**				**-32.6**	**43.3**				**-42.2**

According to *Table 7*, the average BD-BR savings of the HEVC encoder relative to the *2-pass* VP9 and x264 encoding are *32.6%,* and *42.2%,* respectively. Also, it should be noted that the bit-rate savings, on average, are increasing along with an increase of quantization parameters for both VP9 and x264 encoders. In addition, as it is clearly seen from both *Tables 4* and *7*, the average BD-BR savings for VP9-based *1-pass* and *2-pass* encoding, respectively, are almost identical.

Table 8 below provides a full summary of the BD-BR difference for the *2-pass* encoding mode.

Table 8. Summarized BD-BR Low-Delay Experimental Results (Compared to VP9 and x264 *High Profile* Encoders, 2-Pass Mode)

CODEC	**HEVC**	**x264**	**VP9**
HEVC		**-42.2%**	**-32.6%**
x264	**75.8%**		**18.5%**
VP9	**48.3%**	**-14.6%**	

As shown in *Table 8*, the VP9 encoder achieves an average gain of *14.6%* in terms of BD-BR savings compared to X264. Also, in order to achieve the same $PSNR_{YUV}$ values of HEVC, when employing VP9 and x264 encoders, the BD-BR overhead of *48.3%* and *75.8%,* respectively, is required.

Further, *Table 9* presents detailed encoding run times as an indication of computational complexity involved for each of the tested encoders.

Table 9. Encoding Low-Delay Run Times for Equal $PSNR_{YUV}$ (Compared to VP9 and x264 *High Profile* Encoders, 2-Pass Mode)

	HEVC vs. VP9 (in %)				***VP9 vs. x264 (in %)***			
Sequences/QPs	**22**	**27**	**32**	**37**	**22**	**27**	**32**	**37**
FourPeople	637	569	554	544	32453	46370	57743	62669
Johnny	638	571	604	615	20660	25786	31495	41016
KristenAndSara	694	639	637	646	26128	34074	43020	51369
Averages	**656**	**593**	**598**	**602**	**26414**	**35410**	**44086**	**51685**
Total Average	**612**				**39399**			

To summarize the results presented in these section, the difference in bit-rate savings of the VP9 encoder vs. the x264 *High Profile* encoder, for *1-pass* and *2-pass* encoding modes, is about *2%* (i.e. *12.5%* vs. *14.6%,* respectively). On the other hand, there is a very obvious variation in x264 *High Profile 1-pass* and *2-pass* encoding times, which are much lower in the *1-pass* encoding mode without any degradation (on average) in the x264 objective quality for the tested video sequences. Particularly, when comparing the *1-pass* encoding times of both VP9 and x264 *High Profile* encoders, the x264 *High Profile* encoder is able to achieve extremely fast performance with a factor of about *2,000* in its favor; further, when comparing the *2-pass* encoding times of both VP9 and x264 *High Profile* encoders, the x264 *High Profile* encoder performance is faster with a factor of about *400*.

Regarding the bit-rate savings of the HEVC encoder vs. VP9 *1-pass* and *2-pass* encoding modes, they remain substantially the same (i.e. *32.5%* vs. *32.6%,* respectively). Also, the VP9 *1-pass* encoding times are higher by a factor of *2.25* when compared to the VP9 *2-pass* encoding. Regarding the relatively high computational complexity (in term of encoding times) of the H.265/MPEG-HEVC reference encoder, it should be noted that currently leading research institutes as well as industrial companies around the globe are intensively working on developing real-time H.265/MPEG-HEVC encoders, which will allow real-time HEVC-based encoding for HD or even UHD with increasingly smaller R-D performance degradations over time, when compared to the reference encoder. First such real-time encoders are already available for 1080p resolutions [27], and for higher resolutions (such as 4K), they are expected to be available during 2014. Therefore, the computational complexity issue (in terms of the encoding times) of HEVC-based encoders is supposed to be resolved in the very near future.

5. CONCLUSION

A comparative assessment for the LD configuration of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders was presented. For evaluating H.264/MPEG-AVC, the open-source x264 *High Profile* encoder was selected. According to the detailed experimental results, the coding efficiency of VP9 was shown to be inferior to H.265/MPEG-HEVC with an average bit-rate overhead of *32.5%* at the same objective quality for the *1-pass* encoding, and *32.6%* for the *2-pass* encoding. Also, it was shown that the VP9 encoding times are larger by a factor of about 2,000 compared to those of the x264 *High Profile* encoder for the *1-pass* encoding, and about 400 for the *2-pass* encoding, as a trade-off of bit-rate savings of *12.5%* and *14.6%* for the *1-pass* and *2-pass* encoding, respectively. When compared to the full-fledged H.265/MPEG-HEVC reference software encoder implementation, the VP9 encoding times are lower, at average, by a factor of about *2.7* and *6.1* for the *1-pass* and *2-pass* encoding, respectively. In addition, it was observed that the H.265/MPEG-HEVC encoder provides significant average bit-rate savings of more than *40%* relative to the x264 *High Profile* encoder for both *1-pass* and *2-pass* encoding.

6. ACKNOWLEDGEMENTS

The authors gratefully thank Amit Mulayoff and Benaya Itzhaky from the Communication Systems Engineering Department, Ben-Gurion University of the Negev, Israel, for performing tests.

REFERENCES

[1] "Cisco visual networking index: forecast and methodology, 2013–2018", 10 June 2014, Online: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.pdf, (2014).

[2] ITU-T, Recommendation H.265 (04/13), Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of Moving Video, High Efficiency Video Coding, Online: http://www.itu.int/rec/T-REC-H.265-201304-I, (2013).

[3] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," Circuits and Systems for Video Technology, IEEE Transactions on , vol.13, no.7, 560-576 (2003).

[4] J. Ohm, G.J. Sullivan, H. Schwarz, T.K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—including High Efficiency Video Coding (HEVC)," Circuits and Systems for Video Technology, IEEE Transactions on , vol. 22, no.12, 1669-1684 (2012).

[5] B. Li, G. J. Sullivan, and J. Xu, "Comparison of compression performance of HEVC Draft 9 with AVC high profile and performance of HM9.0 with temporal scalability characteristics," JCTVC-L0322, 12th JCT-VC meeting, Geneva, Switzerland, (2013).

[6] M. Horowitz, F. Kossentini, N. Mahdi, S. Xu, H. Guermazi, H. Tmar, B. Li, G. J. Sullivan, J. Xu, "Informal subjective quality comparison of video compression performance of the HEVC and H.264/MPEG-4 AVC standards for low-delay applications," Proc. SPIE 8499, Applications of Digital Image Processing XXXV, 84990W, (2012).

[7] WebM™: an open web media project, VP8 Encode Parameter Guide, Online: http://www.webmproject.org/docs/encoder-parameters

[8] E. Ohwovoriole, and Y. Andreopoulos, "Rate-Distortion performance of contemporary video codecs: comparison of Google/WebM VP8, AVC/H.264 and HEVC TMuC," Proc. London Communications Symposium (LCS), 1-4 (2010).

[9] J. Bankoski, P. Wilkins, X. Yaowu, "Technical overview of VP8, an open source video codec for the web," Multimedia and Expo (ICME), 2011 IEEE International Conference on , 1-6 (2011).

[10] Chromium® open-source browser project, VP9 source code, Online: http://git.chromium.org/gitweb/?p=webm/libvpx.git;a=tree;f=vp9;hb=aaf61dfbcab414bfacc3171501be17d191ff8506

[11] J. Bankoski, R. S. Bultje, A. Grange, Q. Gu, J. Han, J. Koleszar, D. Mukherjee, P. Wilkins, and Y. Xu, "Towards a next generation open-source video codec," Proc. SPIE 8666, Visual Information Processing and Communication IV, 2013, 1-13 (2013).

[12] Paul Wilkins, Google® Groups "WebM Discussion", Online: https://groups.google.com/a/webmproject.org/forum/?fromgroups#!topic/webm-discuss/UzoX7owhwB0

[13] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders", Picture Coding Symposium (PCS), San Jose, USA, 394-397 (2013).

[14] G.J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," Circuits and Systems for Video Technology, IEEE Transactions on, vol.22, no.12, 1649-1668 (2012).

[15] B. Bross, W.-J. Han, J.-R. Ohm, G. Sullivan, Y.-K. Wang, and T. Wiegand "High Efficiency Video Coding (HEVC) text specification draft 10 (for FDIS & Consent)," JCT-VC, Doc. JCTVC-L1003. Geneva, Switzerland, (2013).

[16] HEVC Reference Software, Online: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/

[17] Yaowu Xu, Google® Groups "WebM Discussion", Online: https://groups.google.com/a/webmproject.org/forum/#!msg/codec-devel/agX03TbT65M/-6YyXlpa2qsJ

[18] D. Mukherjee, Chromium® open-source browser project, VP9 source code, Online: http://git.chromium.org/gitweb/?p=webm/libvpx.git;a=commit;h=0d8723f8d5372eaaadfb5373a3b9d35e2c78c42d

[19] Ronald Bultje, Google® Groups "WebM Discussion", Online: https://groups.google.com/a/webmproject.org/forum/#!msg/webm-discuss/RE5J2wBtH-I/zV4LD7Wuin4J

[20] F. Bossen, "Common HM test conditions and software reference configurations," document JCTVC-L1100 of JCT-VC, Geneva, CH, (Jan. 2013).

[21] Projects from VideoLAN™, x264 software library and application, Online: http://www.videolan.org/developers/x264.html

[22] x264 free library/codec, 32-bit, 8-bit depth version r2334 (2013), Online: http://www.divx-digest.com/software/x264.html, (2013).

[23] x264 Settings, Online: http://mewiki.project357.com/wiki/X264_Settings

[24] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves", ITU-T Q.6/SG16 VCEG 13th Meeting, Document VCEG-M33, Austin, USA, (2001).

[25] Ronald Bultje, Google® Groups "WebM Discussion", Online: https://groups.google.com/a/webmproject.org/forum/?fromgroups#!topic/webm-discuss/xopTll6KqGI

[26] John Koleszar, Google® Groups "Codec Developers", Online: https://groups.google.com/a/webmproject.org/forum/#!msg/codec-devel/yMLXzaohONU/m69TbYnEamQJ

[27] "HEVC Real-time Software Encoder: Real-time software encoding up to 1080p60", Online: http://www.hhi.fraunhofer.de/de/kompetenzfelder/image-processing/solutions/hevc-software-and-hardware-solutions/hevc-real-time-software-encoder.html.