

A Tool for Automated Testing of the HEVC Video Encoder

José Raimundo Barbosa
Grupo de Processamento
Digital de Sinais (GPDS)
Instituto Federal da Paraíba
j.zmais@gmail.com

Carlos Danilo Miranda
Regis
Grupo de Processamento
Digital de Sinais (GPDS)
Instituto Federal da Paraíba
regis.danilo@gmail.com

Ruan Delgado Gomes
Laboratório de Computação
Embarcada e Distribuída
(Laced)
Instituto Federal da Paraíba
ruandgomes@gmail.com

Jean Felipe Fonseca de
Oliveira
Instituto de Estudos
Avançados em Comunicações
(IECOM)
jeanfelipecfonseca@gmail.com

ABSTRACT

H.265/HEVC is the state-of-the-art video codec, focusing in Ultra-High Definition video (UHD) and it is 50% more efficient in bit-rate compression than its predecessor, the H.264/AVC. However this efficiency requires a high computational cost, which may compromise the use of the encoder in limited machines or in real time applications that use UHD videos. Several studies look for ways to enhance the HEVC encoder and reduce the computational cost. However, to develop optimized versions of the encoder it is necessary to perform tests that can last for hours and usually requires manual collection of coding information. In this work, a tool to perform automated tests of HEVC encoders was developed. The tool was implemented in a parallel architecture that allows the execution of several instances of the encoder. It collect the informations and executes objective metrics to evaluate the efficiency of the encoder, the duality of the encoder video and the encoding time. [This work intends to provide a useful solution to help analyze encoders in long sequences of tests.](#)

Keywords

Automated testing, HEVC encoder, UHD videos

1. INTRODUCTION

Currently, about three-quarters of annual data traffic on the Internet is assigned to video content, according to Cisco's Visual Networking Index report. In 2014 this consumption was 64% and in 2020 this value will be about 80% [1]. One of the contributing factors is the increase in video quality, which makes it possible to explore more and more new experiences on Internet-broadcast television channels, video con-

ferencing and on-demand video transmission services such as YouTube and Netflix, which Already distribute videos in UHD [2].

The UHD videos are highly popular in the market, because high pixel density makes it possible to increase the perception of image details, allowing new experiences for viewers [2]. UHD 4K videos, for example, have resolution of 4096x2160 pixels while previous standards such as Full HD have 1920x1080. However, a UHD 4K stream raw, with 24 bits per pixel and 24 frames per second, requires a bandwidth of 5 Gbit/s [3]. These settings represent challenges in to storage and transmission of such videos.

To enable the intense and growing traffic of videos on the Internet and to satisfy the bandwidth conditions about the UHD characteristics, it is necessary to develop codecs responsible for digital video signal compression and descompression, which reduces the storage space and adjusts data volume to bandwidth limitations [4][5]. However, when compression occurs several times or in the wrong way, it generates loss of information that can degrade image quality [6]. In this context, the challenge arises of developing codecs capable of meeting the demand for UHD videos, and maintaining compression efficiency and image quality.

The ITU-T H.265 [7], also known as High Efficiency Video Coding (HEVC), is the state-of-the-art codec which focus primarily on new UHD video technologies and presents a compression efficiency 50% higher than its predecessor. H.264/Advanced Video Coding (AVC) [8][9][10]. However, the HEVC has a computational cost up to four times higher than H.264. This is attributed to the processing of the new block structures used in the compression, which increases the computational cost presents challenges for situations that demand real-time video reproduction or in machines with processing limitations [11][12].

To promote HEVC optimization research, JVC-TV provided the HM reference encoder which features a basic implementation of HEVC that makes possible to carry out studies on the efficiency of the codec and develop new features and

optimizations [13]. However, evaluating codecs developed based on the HM reference source code is a normally time-consuming process, since each test implies in the execution of several instances of the evaluated codec and the reference codec, which can last for several hours. After execution, is necessary to compare the UHD videos using objective or subjective metrics. With the comparisons it is possible to monitor the efficiency and performance of the new encoders compared to stable versions in the UHD scenario [6].

This paper describes a tool for automated testing of HM based encoders. The tool was implemented in a parallel architecture in which multiple instances of the encoder are executed simultaneously. To calculate the encoder efficiency and the difference between the codecs the Bjontegaard metric is used, this metric determines the difference of the efficiency of the codecs from the calculation of the mean difference of two curves adjusted by four points [14]. For the formulation of the standardized test cases, the recommendations of JCT-VC were followed [15]. With the tool the comparison between the [4] codec and reference codec [13] was obtained and reduction of the test time by 86% in a simple test case was verified, in which one video and one configuration were used, totaling the execution of 8 instances.

2. STANDARDIZATION OF TESTS

The Joint Collaborative Team on Video Coding (JCT-VC) provided a open source HM reference encoder, which offers a basic implementation of HEVC in C++ [15]. The HM code provides an environment for the development of new features or optimizations, such as those developed in [4][11][12][16]. In conjunction with the encoder, JCT-VC also provides the recommended settings and videos for standardized test suites. These settings will be presented later.

After encoding, HM generates, in addition to the encoded video, a record file in which important information about the encoding process is present such as bit rate, total encoding time, output video file size, PSNR for each YUV color matrix (Y = Luma, U = Chroma Blue and V = Chroma Red).

The way the tool was developed also makes it possible to perform tests with the Joint Exploration Model (JEM) reference encoder [17]. This encoder was developed based on HM and was presented in [18] as the future of the environment for standardized development of HEVC video compression technology, including its current extensions and near-term extensions for screen content coding and high-dynamic-range coding.

To ensure common testing conditions, the JCT-VC committee has developed the JCTVC-L1100 documentation [19], which describes the settings and contents for standardized tests that explore different situations. This standardization is applied in the development of new features or encoder optimizations developed using HM. The settings discussed in the document are:

- All Intra: All frames are encoded independently with only intra-frame prediction.
- Random Access (RA): Coding with high rates of compression. The coding order and the out-

put order of the frames differ, inducing a coding delay. This setting represents broadcast and streaming applications.

- Low Delay P (LP): Coding without the structural coding delay presented when using RA parameters. This configuration is evaluated using only P-frames (unidirectional prediction).
- Low Delay B (LB): Similar to LP scenarios, but this configuration is evaluated using only B-frames, (unidirectional and bidirectional prediction).

These settings specify situations with videos with 8 bits of depth, besides these there are also the configurations directed to videos with 10 bits. In the documentation, the videos present in <ftp://hevc@ftp.tnt.uni-hannover.de/testsequences/> are also recommended, requiring a registration to access the videos. Finally, the quantization values are specified: 22, 27, 32 and 37. In the tool, these values are already configured with standards, but it is possible to add more quantization values.

3. OBJECTIVE METRICS

3.1 PSNR Metric

3.2 SSIM Metric

The Structural Similarity Index (SSIM) is a consolidated and constantly used model for video comparison, it is based on the assumption that the human visual system is highly adapted to extract structural information from images [21]. Therefore, each pixel has a strong dependence on the others and this dependence increases with the proximity. This dependence presents important information about the structure of the objects in the image and that quantifying the structural change of an image can provide a good approximation to the perceived quality [22].

To identify the structural information, the SSIM metric uses a statistical approach based on the average luminance and nxn block contrast of the image. The mean (μ), standard deviation (σ) and covariance are calculated for each block. The mean and standard deviation are approximate estimates of the luminance and contrast of the image, respectively. Covariance (σ_{fg}) is the measure of how much a signal is different from the other.

SSIM uses the measure of structural distortion rather than the error itself. If $x = \{xi|i = 1, 2, \dots, N\}$ represents the original signal and $y = \{yi|i = 1, 2, \dots, N\}$ represents the distorted signal, where i is the pixel index value. The structural similarity index can be calculated according to the equation 1 [4][22].

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (1)$$

In the equation 1, μ_x represents the average of x, μ_y represents the average of y. The variance of x is represented by σ_x^2 and the variance of y by σ_y^2 . The $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ are the variables to stabilize the division with weak, watch $L = 2^n - 1$ is the dynamic range of bits and $k_1 = 0.01$ and $k_2 = 0.03$ by default.

3.3 PW-SSIM Metric

$$SI = \left(\frac{1}{N-1} \sum_{i=1}^N (v_S - S)^2 \right)^{\frac{1}{2}} \quad (2)$$

$$PWSSIM(f, h) = \frac{\sum_{d=1}^D SSIM_d(f, h) * SI_d}{\sum_{d=1}^D 1^{SI_d}} \quad (3)$$

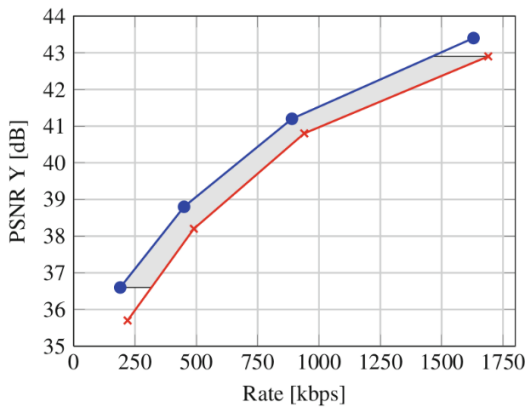
3.4 Bjontegaard Metric

The Bjontegaard metric is considered state-of-the-art in relation to the evaluation of video coding mechanisms and is one of the most widespread in the current literature [14][20]. This allows to obtain the values for overall bit rate (BD-Rate) in percent for two different encoding algorithms considering the same PSNR or overall video quality difference (BD-PSNR) for two different encoding algorithms considering the same bit rate.

The BD-PSNR is the measure corresponding to the average difference of Peak Signal-to-Night Ratio (PSNR) in decibels, between two encoders, considering the same bit rate. While the BD-Rate value corresponds to the average percentage bit difference for the two encoders. In this work, more attention was given to bit rate evaluation.

In [20], exemplifies the BD-Rate by means of Figure 1, where the curves are further drawn by linear interpolation. For the BD calculations, the rate axis is logarithmized and the rate-distortion curves are interpolated by an advanced interpolation method. The overlapping portions of the rate-distortion curves are used for calculation of the BD values. The BD-rate, the difference in horizontal direction. Depending on the range of overlap of the two curves, the resulting BD measurements get more or less meaningful. Therefore, careful interpretation of the resulting numbers is required.

Figure 1: Interpolation of the BD-Rate [20]



4. RELATED WORKS

In the literature it is possible to find several studies related to HEVC optimization techniques. Usually the methodological characteristics used in the development of the new technique as metrics and videos used are presented, besides the obtained results. However the test stage is vaguely mentioned or simply has its description ignored. As it happens

in [4], an alternative is presented that uses machine learning to predict the partitioning of the macroblocks and to avoid the extra processing, and the proposed solution is compared with the HM reference encoder. In [23], a parallel approach is used to accelerate coding by distributing multi-core processor efforts. And in [5], it is proposed an optimized partitioning algorithm, the texts refer to a scenario in which the coded videos are processed by metrics and then the results are compared, without the aid of a tool for extraction and calculation.

The alternative proposed in [24] offers a coder comparison service, in which several configurations are predefined for the tests. For this it is necessary to submit the encoder to one of the paid services offered. Considering the best service, the premium, includes the SSIM, PSNR objective metrics and subjective metrics psycho visual enhancements, also includes the configurations High Speed, Normal and High. Quality, and presets analysis of sythetic motion, distortion in tail area and spatially variable noise. However, this alternative requires payment for more complex tests and also limits testing to a fixed number of configurations and test cases, nor is more current and advanced metrics like BD-Rate and BD-PSNR mentioned. The postponed solution also uses as reference encoders implemented based on X.265, for UHD video tests, this encoder is constantly mentioned in the literature, however the test scenarios proposed by the JVET specify HM encoders, these specifications help to standardize and document studies on the enhancement of the HEVC encoder.

5. TOOL DESCRIPTION

To verify the efficiency it is necessary to perform several tests in which the modified encoder is executed in several coding scenarios, and then the results of each coding are compared with the results of the reference encoder. Each test sequence of an encoder can last for hours, depending on the videos and test settings, which hinders the development stage. The comparison is made between values such as bit rate, PSNR, coding time, and so on. These values are generated automatically by the HM encoder and are saved in log files. However, for their use it is necessary to manually collect in each log file so that it can be a repetitive work and result in errors from human failure.

The tool developed in this work executes instances of the modified encoder and reference encoder in all required configurations and then automatically collects all the information necessary for the comparison. With the data collected from the log files the calculation of the Bjontegaard [25] metric is performed, the tool architecture can be viewed in Figure 2. The modular way the tool was implemented enables other metrics to make use of the encoding values. To reduce the time of the tests, the tool was implemented in a parallel architecture in which multiple instances of the encoder are executed simultaneously, thus considerably reducing the time of a test.

Figure 2: Tool architecture

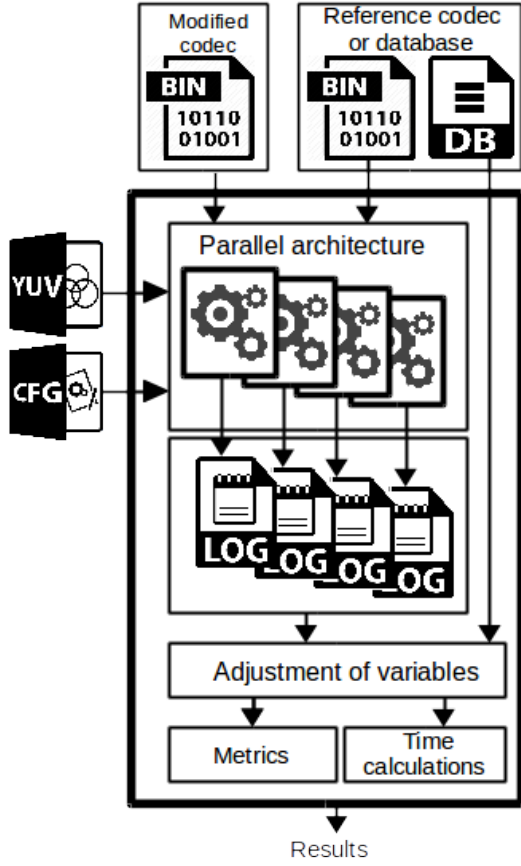


Table 1: Parameters for tool execution

Command	Example	Description
./CodecTest	–	Tool executable
-thr	2	Number of threads
-vin	1 path/video.yuv	Number of video followed by the video files
-cfg	1 path/File.cfg	Number of configuration followed by the configurations files
-outl	path/out/logs/	Path where log files will go
-outv	path/outvideos/	Path where encoded videos will go
-eva	path/BinEncoder	Rated encoder binary
-ref	path/BinEncoder	Reference encoder binary
-fr	30	Frame rate
-f	130	Number of frames
-q	4 22 27 32 37	Number of quantization followed by the quantization values
-wdt	3840	Video width
-hgt	2160	Video height
-bkp	–	Backup flag

The output file structure of the testing tool presents the key information about the execution of each instance, in addition to the total time values and the values of the metrics. A log file example is shown in Figure 3.

To prevent data loss during an unexpectedly stopped execution, a tool log file is generated that saves the current state of the test and makes it possible to return from where it left off. It is also possible to save the data of the tests performed by the encoders, thus avoiding to repeat the process and thus to promote the construction of a database with the results of the tests.

To execute the tests, the tool receives the parameters specified in Table 1, these parameters indicate the settings that will be addressed equally by the modified encoder and the reference encoder that will generate different outputs. If not inform the parameters, default values will be used that will execute the tool used a CIF video (352x288) and a random configuration. These parameters have minimum specifications only for checking the operation of the tool. It is important to check the memory and thread limitations before running the tool, it is recommended to run one thread for every two GBs of RAM. In case of an unexpected interruption, the execution of the tool with the -bkp flag activates the backup and the test will return to the moment of interruption.

Figure 3: File log example

```

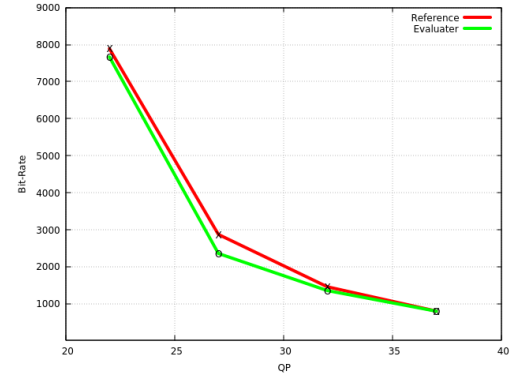
-----
CODEC LOG EXTRACT
-----
CONF:          /test/config.cfg
VIDEO0:        /test/videos/test.yuv
Chroma format: 4:2:0
Color depth:   8
Width:         3840
Height:        3160
Frames:        130
Frame Rate:    30

QT:           22
PSNR-Y:       39.7910
PSNR-U:       43.4607
PSNR-V:       42.7158
Bitrate:      32723.0400
QT:           27
PSNR-Y:       39.7910
PSNR-U:       43.4607
PSNR-V:       42.7158
Bitrate:      32723.0400
QT:           32
PSNR-Y:       39.7910
PSNR-U:       43.4607
PSNR-V:       42.7158
Bitrate:      32723.0400
QT:           37
PSNR-Y:       39.7910
PSNR-U:       43.4607
PSNR-V:       42.7158
Bitrate:      32723.0400
-----
TESTS LOG
-----
Total Time:    04:33:56
BD-Rate:       2.1240
BD-PSNR:       -0.1754

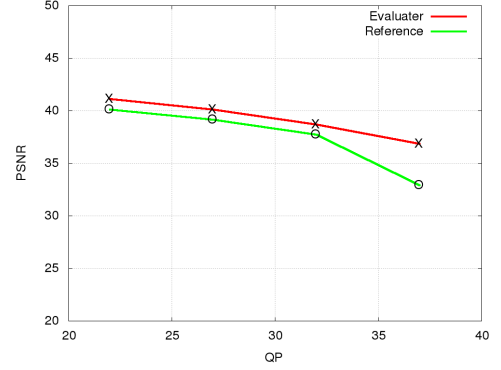
```

For more complex tests, the implementation of a distributed version of the tool is in progress, which will allow the execution of more instances of the encoder according to the configuration and quantity available machines. Online services are also developed, in which researchers and developers can submit the executable program of their encoders and the configurations so that the tests are performed on a robust server. With this proposal also intends to build a database fed with the data of the tests of the users, this database will contribute to other researchers.

It is also possible to observe the values of Bit Rate per QP used in the test in 4a, and also the PSNR values by QP in 4b, these graphs are generated automatically after the tool execution.



(a) Bit Rate x QP



(b) PSNR x QP

Figure 4: Chart examples

6. RESULT

To validate the tool, a sequence of tests was performed to compare the encoder developed in REFERENCIA with the reference encoder HM.16. The tests were performed on a computer with processor SPECIFICATIONS. We used the UHD 4K videos available in REFERENCE. In total, N frames of X sequences were used, being SEQUENCENAME.

The total time spent executing the test sequence using the tool was considerably less than the test performed manually, ie running one encoder or more at a time, calculating the values of the metrics and documenting the results. Because it is an alternative with multi-threaded processing, time reduction is something that is already expected, but adequate management is required so that the execution of different threads does not compromise the integrity of the values used to calculate the metrics, time and other information present in the sequence of tests.

7. REFERENCES

- [1] CISCO. “Cisco VNI Forecast and Methodology, 2015-2020”. Technical report, CISCO, 2016. Disponíble: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html>, Access: 8 January 2016.
- [2] M. Cheon and J. S. Lee. “Objective Quality Comparison of 4K UHD and Up-Scaled 4K UHD Videos”. In *2014 IEEE International Symposium on Multimedia*, pp. 78–81, Dec 2014.
- [3] R. D. Gomes, Y. G. G. d. Costa, L. L. Aquino Júnior, M. G. d. Silva Neto, A. N. Duarte and G. L. d. Souza Filho. “A Solution for Transmitting and Displaying UHD 3D Raw Videos Using Lossless Compression”. In *Proceedings of the 19th Brazilian Symposium on Multimedia and the Web*, WebMedia ’13, pp. 173–176, New York, NY, USA, 2013. ACM.
- [4] J. F. F. de Oliveira. “Técnica de Aprendizagem Automática Aplicada a um Codificador HEVC em Tempo Real”. In *Tese (Doutorado) - Curso de Pós Graduação em Engenharia Elétrica*, p. 112, 2016. Departamento de Engenharia Elétrica, Campina Grande.
- [5] C. C. Wang, Y. C. Liao, J. W. Wang and C. W. Tung. “An Effective TU Size Decision Method for Fast HEVC Encoders”. In *2014 International Symposium on Computer, Consumer and Control*, pp. 1195–1198, June 2014.
- [6] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy and M. Manohara. “Toward A Practical Perceptual Video Quality Metric”. Technical report, Netflix, 2016. Disponíble: <http://techblog.netflix.com/2016/06/toward-practical-perceptual-video.html>, Access: 10 January 2016.
- [7] I. T. Union. “Recommendation H.265: High Efficiency Video Coding”. Recommendation, ITU, 2016. Disponíble: <https://www.itu.int/rec/T-REC-H.265>, Access: 18 January 2016.
- [8] F. Bossen, B. Bross, K. Suhling and D. Flynn. “HEVC Complexity and Implementation Analysis”. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685–1696, Dec 2012.
- [9] P. Hanhart, M. Rerabek, F. D. Simone and T. Ebrahimi. “Subjective quality evaluation of the upcoming HEVC video compression standard”. In *Applications of Digital Image Processing XXXV*, 2012.
- [10] G. J. Sullivan, J. R. Ohm, W. J. Han and T. Wiegand. “Overview of the High Efficiency Video Coding (HEVC) Standard”. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.
- [11] J. S. Yoon, C. Lee, C. Park, G. Lee, K. Lee, S. Roh, M. Jeon, Y. Jung, J. Oh and J. A. Lee. “An H.265/HEVC codec for UHD(3840 x00D7;2160) capturing and playback”. In *2013 International SoC Design Conference (ISOC)*, pp. 218–220, Nov 2013.
- [12] G. Correa, P. Assuncao, L. Agostini and L. A. da Silva Cruz. “Performance and Computational Complexity Assessment of High-Efficiency Video Encoders”. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1899–1909, Dec 2012.
- [13] I. T. Union. “HEVC Test Model (HM)”. Recommendation, ITU, 2016. Disponíble: <https://hevc.hhi.fraunhofer.de/HM-doc/>, Access: 18 January 2016.
- [14] G. Bjøntegaard. “Document VCEG-M33: Calculation of average PSNR differences between RD-curves”. In *ITU-T VCEG Meeting*, 2001.
- [15] F. Bossen, D. Flynn, K. Sharman and K. Sühling. “HM Software Manual”. *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, p. 30, Oct 2015.
- [16] R. Weerakkody, M. Mrak, V. Baroncini, J. R. Ohm, T. K. Tan and G. J. Sullivan. “Verification testing of HEVC compression performance for UHD video”. In *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 1083–1087, Dec 2014.
- [17] F. Bossen, D. Flynn, K. Sharman, K. Sühling and X. Li. “JVET JEM software”. Technical report, JVET, 2015. Disponíble: <https://jvet.hhi.fraunhofer.de/>, Access: 8 April 2017.
- [18] JVET. “Algorithm Description of Joint Exploration Test Model 1 (JEM 1)”. Technical report, JVET, 2015. Disponíble: <http://mpeg.chiariglione.org/standards/exploration/future-video-coding/n15790-algorithm-description-joint-exploration-test-model>, Access: 8 April 2017.
- [19] F. Bossen. “Common HM test conditions and software reference configurations”. Technical report, JCT-VC, 2013. Disponíble: <http://phenix.it-sudparis.eu/>, Access: 18 January 2017.
- [20] M. Wien. “High Efficiency Video Coding: Coding Tools and Specification”. *Signals and Communication Technology*, 2015.
- [21] J. V. de Miranda Cardoso, A. C. S. Mariano, C. D. M. Regis and M. S. Alencar. “Comparação das Métricas Objetivas de Qualidade de Vídeos Baseadas na Similaridade Estrutural e na Sensibilidade ao Erro”. *Revista de Tecnologia da Informação e Comunicação*, 2012.
- [22] Z. Wang, L. Lu and A. C. Bovik. “Video quality assessment using structural distortion measurement”. In *Proceedings. International Conference on Image Processing*, volume 3, pp. III–65–III–68 vol.3, 2002.
- [23] Y. Wang, X. Guo, Y. Lu, X. Fan and D. Zhao. “GPU-based optimization for sample adaptive offset in HEVC”. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 829–833, Sept 2016.
- [24] M. G. M. Lab. “Video Codec Analysis and Testing”. Technical report, MSU Graphics Media Lab, 2012. Disponíble: http://compression.ru/video/codec-comparison/codec_analysis.html, Access: 10 August 2017.
- [25] G. Bjøntegaard. “Calculation of Average PSNR Differences Between RD-curves”. *ITU - Telecommunications Standardization Sector*, 2001.