Video Performance Metric Assessment of Coding Standards H.264/AVC and MPEG-4

Andreja Samčović

Faculty of Transport and Traffic Engineering, University of Belgrade, Belgrade, Serbia
Email: andrej@sf.bg.ac.rs (A.S.)
Manuscript received July 29, 2024; revised August 23, 2024; accepted September 29, 2024; published November 13, 2024

***Abstract*—This paper presents a comprehensive video quality assessment that focuses on the comparison of two predominant video coders, H.264/AVC and MPEG-4, particularly at very low resolutions pertinent to web-based applications and security cameras. Objective quality metrics such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) were employed to evaluate the performance of these coders. Through experimental analysis, it was observed that H.264/AVC offered superior performance over MPEG-4 in terms of both PSNR and SSIM values. This result underscores the efficiency of H.264/AVC in scenarios where high-quality video is essential, despite bandwidth or storage constraints.**

***Keywords*—video coding, quality metrics, standards, Peak Signal-to-Noise Ratio (PSNR), Similarity Index Measure (SSIM)**

I. Introduction

We define video quality assessment as the study of a video's visual characteristics in a specific encoding (or encoder implementation) in comparison with reference material deemed original [1]. Ideally, this original is the interaction with the real world, just as a human would see. In practice, we refer to static high-quality or camera-raw footage as the original. The goal is to derive quality metrics that measure the degradation of video quality. For higher deviations in quality metrics, it should be concluded the degradation level for the coding is also higher. The major reason for such quality degradation is the presence of noise [2]. However, this noise can be defined subjectively as the difference between the viewer's perception of the depicted and the actual information.

The objectives for video coding include various goals related to video compression and quality enhancement [3, 4]. Some common objectives highlighted in the sources are: reducing file size; reducing buffering for video streaming; changing video resolution or aspect ratio; changing audio formats or quality; converting obsolete files to modern formats, and making different types of videos (natural videos, videos with a human face, cartoons, video games, recorded videos of computer desktop) compatible with different devices [5]. These objectives aim to enhance the efficiency of video transmission, reduce storage requirements, and ensure high-quality video content across different platforms and devices.

We were motivated to compare H.264/AVC, which is today the most commonly video coding used standard with previous standard MPEG-4 (less used), because H.264/AVC achieves higher compression ratios, allowing significantly smaller file sizes [6]. For example, a 30 second full HD (High Definition) video would require 5.2 GB without compression, but only 65.4 MB using H.264 compression.

Furthermore, H.264/AVC supports HD video, enabling applications like Blu-ray, HD streaming on the internet, HD video on smart phones, and public TV broadcast in Europe. H.264/AVC trades off more computing power for requiring less bandwidth and storage space, which is a worthwhile trade-off for many applications [7]. One interesting advanced application is image processing in learning-based networks [8, 9].

Last but not least, H.264/AVC exploits both spatial redundancy (correlation between neighboring pixels in a frame) and temporal redundancy (correlation between frames) to achieve high compression ratios [10].

In summary, the combination of much higher compression, support for HD video, hardware acceleration, and the ability to trade-off computing power for reduced bandwidth makes H.264/AVC an attractive standard for a wide range of video applications [11].

Video quality assessment is very important characteristic in a specific coding scenario to some reference material deemed the original. There are mainly two types of quality evaluation methods: subjective and objective [12]. Subjective methods are very time-consuming and complex to evaluate. Furthermore, it is difficult to perform an accurate scientific analysis. Therefore, this paper focuses on more objective metrics for quality assessment. When we classify the objective methods for evaluation that can be categorized into two types: *Full-reference* and *No-reference* [13].

Full-reference—In this approach, the sample video or image is compared with the reference image to assess the quality of the output video or image. In the case of higher similarity with the reference sample, it can be stated that the sample is likewise or similar. Among the full-reference approach the MSE (Mean Squared Error) is the most common method used for quality assessment. It evaluates the squared intensity difference between the ground truth and the test sample (distorted).

No-reference—While in this approach there is no need for a reference image or ground truth, the measurement techniques are directly applied to the sample video or frames [14].

The degradation of quality shows artifacts like blurring, noise, signal distortion, etc. The quality assessment metrics mostly include MSE and PSNR (Peak Signal-to-Noise Ratio) as they are comparatively easier to compute and simple to implement. Hence, these two measures are the most commonly used in the community for video quality evaluation.

This paper is organized as follows. In Section II, the main

video quality assessment methods are mentioned. The video coding approach is described briefly in Section III. Section IV presents some experimental results obtained using encoder options. Future work and conclusions are presented n Section V.

II. Objective Quality Assesment Metrics

In this section, we briefly explain the most common types of video quality assessment methods that are available and used today. Afterwards, we compare the assessment methods with our requirements. This section will help us to understand the advantages and disadvantages. This also aids in the decision on selecting a proper assessment tool.

Peak Signal-to-Noise Ratio (PSNR) is a full-reference metric that requires two frames of video or images to compute [15]. Here, the values are interpreted as higher being the better result. The total PSNR is calculated on the basis of the geometric mean of the MSE of all frames [11]. However, it fails to recognize the difference in distortion of a frame to another.

PSNR is the simplest method for video quality assessment, and its calculation is also faster than other methods. Hence, it is the most common method compared with other quality assessment methods. The signal is considered as the original data while the noise is considered to be the error mainly induced due to the degradation of video due to compression or distortion. If the data is of the 8-bit type, then the PSNR values are in the range of 30-50dB, whereas in the case of 16-bit data, the typical range of PSNR is between 60 and 80 dB [16]. In the case of wireless transmission of data 20-25 dB loss of quality is considered normal [17]. The PSNR mathematical formula can be expressed as [15]

$$PSNR = 10\log\left(\frac{peakvalue^2}{MSE}\right)[dB] \quad (1)$$

where the *peakvalue* is the highest range of the image data. *MSE* is the mean squared error of the image frame. If the data are of the 8-bit type then the *peakvalue* is 255.

Structural Similarity Index Measure (SSIM) is also a full-reference metric. Its value ranges are between -1 and 1 [18]. The value -1 means that the frames are completely different from each other, while 1 means that the frames are similar [19]. The values are interpreted as better for higher metric values, resulting in similar frames. Its visualization is pixel-wise unlike Naturalness Image Quality Evaluator (NIQE) and Video Multimethod Assessment Fusion (VMAF), which are block-wise visualization methods.

SSIM mainly compares the three components of the images: luminance, contrast, and structure. The result of each component is compared pixel-wise and the arithmetic mean of the SSIM values is taken into the metric. The algorithm performs a window function that performs the convolution of the window. The SSIM mathematical formula can be expressed as [20]

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)}{(\mu_x^2 + \mu_y^2 + C_1)} \cdot \frac{(2\sigma_{xy} + C_2)}{(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2)$$

The SSIM is based on the calculation of different window sizes. The values of x and y for a fixed window size are represented in the above formula, where μ_x and μ_y are the pixel sample means of (x,y).

Furthermore, σ_x and σ_y denote variances of the input values, while σ_{xy} is the covariance. C_1 and C_2 stabilize the division with a weak denominator.

Basically, it is a model based on perception. If there is a change or degradation in the quality of frames then the structural information changes. These changes can be in the form of luminance and contrast. In the case of luminance, the parts of degradation are not clearly identifiable. On the other hand, for contrast the texture- based information is changed in the case of degradation [21].

III. Methodology for Video Signal Metric Calculation

The frame-by-frame version of the movie *Bugs Bunny* (Fig. 1) will be the basis of our measurements, but more on that later. In order to actually get meaningful metrics, we must first re-encode the movie into the different formats that should be compared. In this study, these are H.264/AVC and MPEG-4 coders [7, 22]. For both variants, we used ffmpeg version 4 for all encoding-related tasks, which allows us to use a *Matroska* container and select a different encoder for each output.

Fig. 1. One frame of the short film *Bugs Bunny*.

In total, dataset amount was about 500 GB of raw data. Since the experiments only target Common Intermediate Format (CIF) and Quarter CIF (QCIF) resolutions it is not necessary to keep all raw data. Our algorithm fetches the images, crops into form by discarding one of the two images and then only saves the resized versions which are ready for coding. This produces an output of about 2.4 GB – the other data from the render farm is discarded.

Noteworthy here is that the input data doesn't quite match our target resolutions in terms of aspect ratio: the movie is rendered at an industry-standard ratio of 16:9, whereas both CIF and QCIF use 22:15. Therefore, the resulting video files are bit smaller on the vertical axis than defined. This should not impose a problem for the analysis, but it should be noted nonetheless. Furthermore, all video is exported without audio, subtitles or any additional metadata.

To reasonably compare the two coders with each other, we settled on a few options for each, resulting in more than two output videos. H.264/AVC provides some templates here, like a predefined set of options for often used scenarios. These templates are named *Tune* and *Preset*. The former specifies the general content for which the video should be optimized, for example, for *Animation* or *Film*. With the other option, users may nearly specify the amount of time invested into efficient encoding – values here range from *Ultrafast* to *Veryslow*. Table 1 shows all the possible values for both parameters. In our analysis, we rendered one output file for each combination (a total of 72 files for each resolution).

However, the options we used when encoding with MPEG-4 are a bit more limited. For this coder, we selected a sample of values for the *Quality* option. This argument is a numeric scale that ranges from 0 to 31. Higher values yield better results, but at the cost of encoding time.

Table 1. Values used for the coder options

H.264/AVC		MPEG-4
Tune	Preset	Quality
Ultrafast	Film	3
Superfast	Animation	12
Veryfast	Grain	16
Faster	Stilimage	20
Fast	Psnr	24
Medium	Ssim	27
Slow	Fastdecode	31
Slower	Zerolatency	
Veryslow		

For the actual metric calculation, we refer to ffmpeg's fast toolset. Using the -lavfi option, ffmpeg lets users specify an output file. This requires providing two input files for comparison. In this study, the first input is the dataset of raw movie frames used to encode our video files. The second input is the encoded video in which we want to measure performance. This works well for both PSNR and SSIM. In both cases, the provided output file is filled with one line per frame of the video. Here, we obtain a metric for each dimension in the color space (YUV) as well as a weighted average. Because more information is contained in the luminance component Y than in the others (U,V), this value is weighted higher accordingly.

After running this process on all input files, we receive a metric file for each combination of encoder and options. To extract information on the overall performance of each encoder, we aggregate the value for each frame over varying sets of parameters. That means we calculate median values for each H.264/AVC tune value, combining results from the different preset options. Similarly, for MPEG-4 files, the different quality metrics are combined. Furthermore, we aggregate these values once more to produce one value for each second of the film rather than each frame.

IV. Experimental Results

In this section we evaluate and discuss the results obtained from our experiments. Again, see Table 1 for the set of encoder options that we used.

Approximately 7½ minutes of the short film *Bugs Bunny*, both PSNR and SSIM values reach a global minimum, independent of the encoder. These few frames are a jump cut from one almost completely blue scene with the camera pointed upwards to the sky in the same location, but with the camera pointing directly down to the earth. This change is shown in Fig. 2. Here, a character is shown falling downwards, so a sudden camera change is necessary once it is out of view.

Fig. 2. Still frames of the jump cut with minimal measured quality by both metrics.

For H.264/AVC, the PSNR average is above 37 dB, while the corresponding MPEG-4 measurement starts higher, but falls to around 30 dB. We can definitely state that for our test video, the H.264/AVC encoding outperforms MPEG-4.

A similar conclusion can be made for the second metric. SSIM values are not quite as stable as the corresponding PSNR, but that is to be expected because this metric is more subjective and varies more depending on the current scene of the movie. For H.264/AVC, the average values of SSIM are just over 0.95, with the lowest dip is still above 0.85. MPEG-4 yields different results – SSIM values here go as far down as 0.7. Another interesting observation is that the fluctuation is much higher for MPEG-4-based encoding. It appears that the other encoder is more resilient to sudden scene changes.

However, this leads to encoders needing to effectively re-encode the entire picture because of the sudden change, which offers a few very low-quality frames. Because these changes only last for a small fraction of a second, they are not noticeable when watching the movie. This means that both encoders are perfectly suited for dealing with sudden changes and real-life video recordings (for example in the context of security cameras).

Another observation can be made when looking at the relative difference between these two metrics. Here, a basic difference metric is calculated by first normalizing the PSNR value with the maximum observed value (excluding infinity). Then, the linear difference is calculated. What we can deduce is a confirmation that both metrics produce similar results.

Instead of aggregating all possible option choices, we can produce a more fine-grained evaluation. An example can be seen in Fig. 3. Here, both metrics are evaluated with regard to the *tune* parameter given to the H.264/AVC encoder. As expected, the *psnr tune* produces the best PSNR results,

which are marginally better by 1-5dB throughout the video. An interesting observation is that this profile also performs best when measuring the SSIM metric. The self-proclaimed *SSIM tune* still comes second in our measurements.

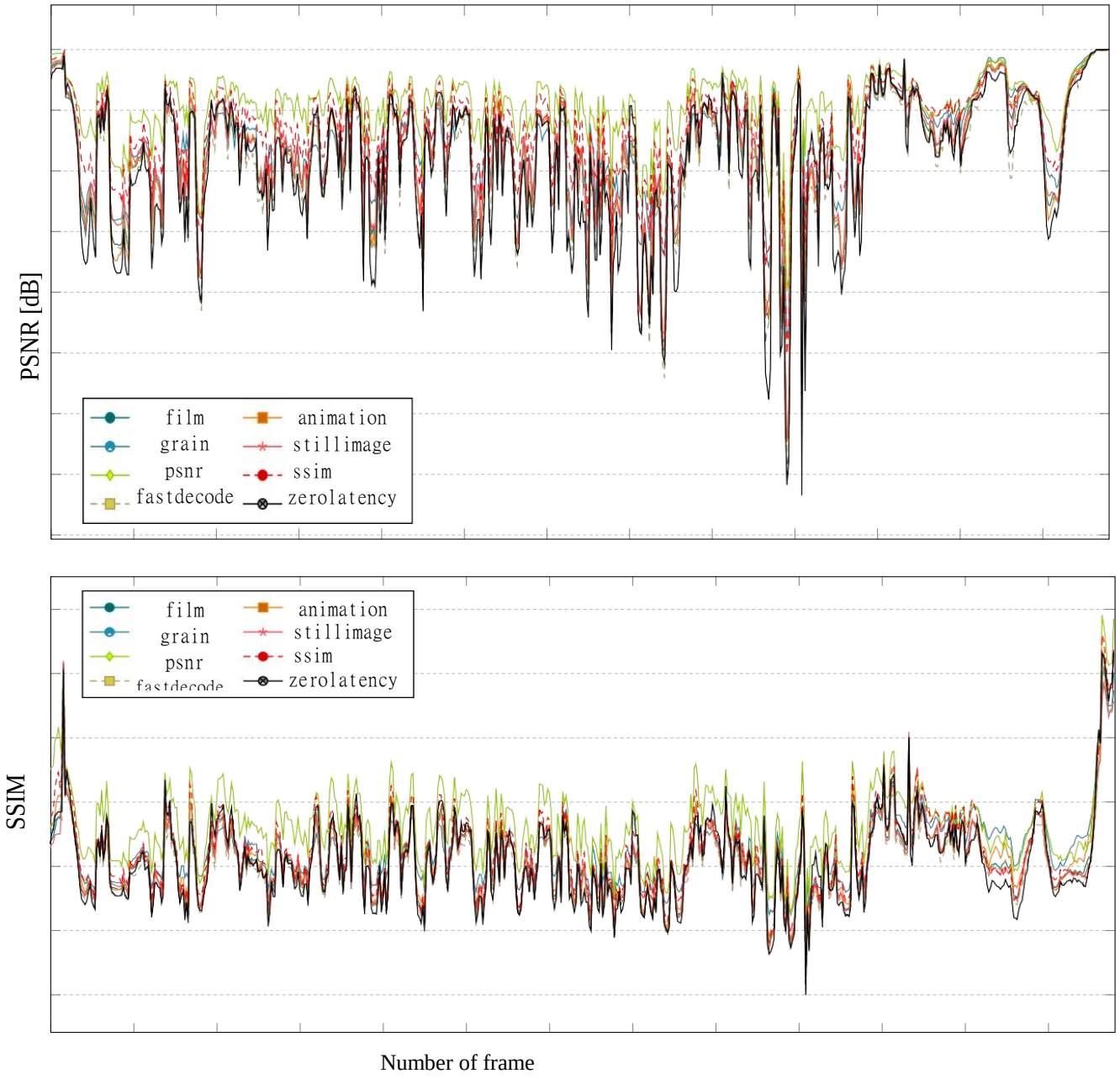

Fig. 3. Values used for encoding regard to the *tune* parameter for H.264 measuring both metrics (PSNR for upper and SSIM for lower image).

A similar evaluation was performed for various quality parameters of MPEG-4. The different quality values produce graphs that are approximately translated copies of each other.

Another observation can be made when examining the relative difference between these two metrics. Here, we calculated a basic difference metric by first normalizing the PSNR value with the maximum observed value (excluding infinity). Then, the linear difference is calculated. We can deduce that this is a confirmation that both metrics produce similar results.

V. Conclusions

Future work in this area should consider broadening the scope to compare H.264/AVC and MPEG-4 across various applications beyond low-resolution environments. This could include testing performance in high-resolution scenarios, which are progressively relevant to the growth of 4K and 8K media content. Diverse genres and types of reference videos, including fast action clips and low-light environments, should be evaluated to provide a comprehensive understanding of the performance across different contexts. Furthermore, incorporating subjective quality assessment methods could complement the objective metrics used, offering insight into user perception of video quality.

Understanding the applicability of each algorithm to different network conditions, storage capabilities, and streaming demands is also an avenue worth exploring. Assessing the computational efficiency and energy consumption of both standards would add valuable data for applications where these factors are crucial.

The limitations of the current study are primarily related to

low resolutions and the singular reference video, which may not represent the varied use cases in modern video applications. Thus, it is necessary to confirm whether the observed superiority of H.264/AVC holds across different content types and resolutions. The conclusions are also based on just two quality metrics, PSNR and SSIM, which might not capture all aspects of perceived video quality, particularly for varied content and distortions introduced by different compression levels. Future research should involve a more diverse set of content, resolutions, and quality metrics to validate and expand upon our findings.

H.264/AVC offers several benefits for video encoding, making it a popular choice for various applications. Some key benefits include: low bandwidth usage, high-resolution monitoring and low storage demands. H.264 provides good video quality at low bit rates, making it cost-effective for delivering video content over the internet, especially in areas with limited bandwidth.

H.264 supports a wide range of resolutions, making it suitable for high-resolution video monitoring applications like security cameras and drones. Its efficient compression allows video to be stored in smaller file sizes, reducing storage requirements for video-intensive applications. These benefits highlight H.264's versatility, efficiency, and compatibility, making it a valuable choice for video coding across different industries and applications.

To conclude, this study clearly demonstrates that regarding to low-resolution video encoding H.264/AVC surpasses MPEG-4 in terms of objective quality metrics, PSNR and SSIM. The significant improvements in video quality, as illustrated by the increase in PSNR by approximately 10 dB and SSIM by 10-15%, emphasize the efficiency of H.264/AVC for applications requiring high-quality video streams under constrained bandwidth conditions.

The results are particularly relevant for security-critical applications, where the superior performance of H.264/AVC can be leveraged in web-based platforms, IP security cameras, and camera network infrastructure. While MPEG-4 may still hold utility in situations with less stringent requirements, the study's findings steer stakeholders toward considering H.264/AVC as the more suitable option in scenarios demanding high-fidelity video quality. Finally, this research has not only contributed valuable empirical evidence to the body of knowledge on video coding standards but has also opened avenues for further exploration in optimized video quality assessments, particularly for deep learning applications in security and surveillance fields.

Conflict of Interest

The author declares no conflict of interest.

Funding

The research presented in this paper is partially supported by the Ministry of Science, Technological Development and Innovation of the Republic of Serbia.

References

[1] T. J. Liu, Y. C. Lin, W. Lin, and C. C. J. Kuo, "Visual quality assessment: recent developments, coding applications and future trends," *APSIPA Transactions on Signal and Information Processing*, 2013.

[2] R. R. Choudhary, A. Jangid, and G. Meena, "A novel approach for edge detection for blurry images by using digital image processing," in *Proc. International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*, pp. 1029–1034, 2017.

[3] Y. Di, X. Wang, L. Wang, and M. Zhao, "Research and application of video codec technology," in *Proc. International Conference on Optoelectronics Materials and Devices (ICOMD 2022*), vol. 12600, 1260025, 2022.

[4] M. Uhrina, L. Sevcik, J, Bienik, and L. Smatanova, "Performance comparison of VVC, AV1, HEVC, and AVC for high resolutions," *Electronics*, 13, 953, 2024.

[5] R. Strukov and V. Athitsos, "Evaluation of video compression methods for network transmission on diverse data: A case study," in *Proc. the 16th International Conference on Pervasive Technologies Related to Assistive Environments (PETRA '23*), p. 6, Corfu, Greece, July 5–7, 2023.

[6] D. Petreski and T. Kartalov, "Next generation video compression standards—performance overview," in *Proc. 30th International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, Ohrid, North Macedonia, 2023.

[7] D. Marpe, T. Wiegand, and G. J. Sullivan, "The H.264/MPEG4 advanced video coding standard and its applications," *IEEE Communications Magazine*, vol. 44, no. 8, pp. 134–143, August 2006.

[8] G. Meena, K. K. Mohbey, and S. Kumar, "Monkeypox recognition and prediction from visuals using deep transfer learning-based neural networks," *Multimedia Tools and Applications*, February 2024.

[9] G. Meena and K. K. Mohbey, "Sentiment analysis on images using different transfer learning models," *Procedia Computer Science*, vol. 218, pp. 1640–1649, 2023.

[10] S. Habib, W. Albattah, M. F. Alsharekh, M. Islam, M. M. Shees, and H. I. Sherazi, "Computer network redundancy reduction using video compression," *Symmetry*, vol. 15, p. 1280, 2023.

[11] A. K. Singam, "Coding estimation based on rate distortion control of H.264 encoded videos for low latency applications", arXiv, March 16, 2023.

[12] R. R. Choudhary, V. Goel, and G. Meena, "Survey paper: Image quality assessment," in *Proc. International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM*), Jaipur, India, February 26–28, 2019.

[13] A. Gavrovska, A. Samčović, and D. Dujković, "No-reference image quality assessment based on machine learning and outlier entropy samples," *Pattern Recognition and Image Analysis*, vol. 34, no. 2, pp. 275–287, 2024.

[14] A. Звездакова, A. Antsiferova, Д. Куликов, Д. Кондранин, Д. Ватолин, "Barriers toward no-reference metrics application to compressed video quality analysis: On the example of no-reference metric," pp. 22–27, 2019. doi: 10.30987/graphicon-2019-2-22-27.

[15] D. Poobathy and R. M. Chezian, "Edge detection operators: Peak signal to noise ratio based comparison," *International Journal of Image, Graphics and Signal Processing*, vol. 6, no. 10, pp. 55–61, September 2014.

[16] D. R. Bull and F. Zhang, "Digital picture formats and representations," *Intelligent Image and Video Compression*, 2nd ed., ch. 4, pp. 107–142, 2021.

[17] N. Thomos, N. V. Boulgouris, and M. G. Strintzis, "Optimized transmission of JPEG2000 streams over wireless channels," *IEEE Transactions on Image Processing*, vol. 15, no. 1, January 2006.

[18] A. Hor é and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *Proc. 20th International Conference on Pattern Recognition*, pp. 2366–2369, 2010.

[19] D. R. Bull, "Measuring and managing picture quality," *Communicating Pictures*, Oxford: Academic Press, ch. 10, pp. 317–360, 2014.

[20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment, from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.

[21] A. C. Bovik, "Content-weighted video quality assessment using a three-component image model," *Journal of Electronic Imaging*, vol. 19, no. 1, 011003, January 2010.

[22] J. W. Chen, C. Y. Kao, and Y. L. Lin, "Introduction to h.264 advanced video coding," vol. 2006, p. 6, 2006.