SoftwareX 24 (2023) 101592

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

IV-PSNR: Software for immersive video objective quality evaluation

Jakub Stankowski, Adrian Dziembowski *

Institute of Multimedia Telecommunications, Poznan University of Technology, 60-965 Poznań, Poland

ARTICLE INFO

Keywords:
Image quality assessment
Video quality assessment
Immersive video

ABSTRACT

In this article, we present the IV-PSNR software, which allows for objective quality assessment of immersive video. The software outputs the IV-PSNR metric calculated for a pair of raw video files. The IV-PSNR metric is designed to mimic the subjective perception of the quality of rendered virtual views, which are characterized by artifacts induced by reprojection of pixels between different views. Since 2019, the IV-PSNR software is a reference software for immersive video quality assessment of the ISO/IEC MPEG VC standardization group and is commonly used both in standardization activities towards the development of the ISO/IEC 23090–12: MPEG immersive video (MIV) standard and scientific research on immersive video.

Metadata

Nr	Code metadata description	*Please fill in this column*
C1	Current code version	v5.0
C2	Permanent link to code/ repository used for this code version	https://github.com/jstankowski/ivpsnr
C3	Permanent link to reproducible capsule	https://github.com/jstankowski/ivpsnr/tree/ main/example
C4	Legal code license	BSD-3-Clause
C5	Code versioning system used	git
C6	Software code languages, tools and services used	C++
C7	Compilation requirements, operating environments and dependencies	CMake 3.15 or newer, C++17 conformant compiler (e.g., GCC 8.0 or newer, clang 5.0 or newer, MSVC 19.15 or newer), fmtlib (https://github.com/fmtlib/fmt) – automatically downloaded during CMake build process
C8	If available, link to developer documentation/ manual	https://github.com/jstankowski/ivpsnr#readme
C9	Support email for questions	jakub.stankowski@put.poznan.pl

1. Motivation and significance

The key idea of the immersive video system is to allow a user to freely navigate within the scene [1] captured by a multicamera system equipped with perspective or omnidirectional cameras (Fig. 1). The possibility of free navigation is provided by using the information from input views, corresponding depth maps, and camera parameters [2] for rendering (synthesis) of virtual views (viewports) [3].

Although immersive video is an extension of traditional, two-dimensional video, its crucial requirement is the same – to provide possibly highest quality of experience, being focused on maximizing the quality subjectively perceived by the users. Unfortunately, conducting a proper subjective test is a tedious and time-consuming process, which makes it highly impractical. Therefore, an obvious solution is to assess the quality objectively.

The objective quality evaluation research field is one of the widest among image and video processing. Therefore, in many cases (such as two-dimensional video), state-of-the-art metrics successfully mimic the subjective perception of video. However, the principle of immersive video – reprojection of information captured by multiple cameras – makes it an exception, where typical objective quality metrics fail. Considering characteristics of artifacts typical for immersive video, we have developed the IV-PSNR metric [4], implemented in the software described in this article.

IV-PSNR is a PSNR-based metric, which includes two major modifications, that adapt it to typical artifacts of immersive video [4]: slight shift of pixels (caused by imperfect reprojection) and minor change of global illumination of video (caused by different color characteristics of different cameras). These artifacts significantly degrade objective quality while being unnoticeable to a viewer (Fig. 2).

As presented in the first row of Fig. 2, a slight shift of reprojected pixels is unnoticeable for a viewer but decreases the PSNR value

* Corresponding author.
E-mail address: adrian.dziembowski@put.poznan.pl (A. Dziembowski).

https://doi.org/10.1016/j.softx.2023.101592
Received 18 September 2023; Received in revised form 26 October 2023; Accepted 15 November 2023

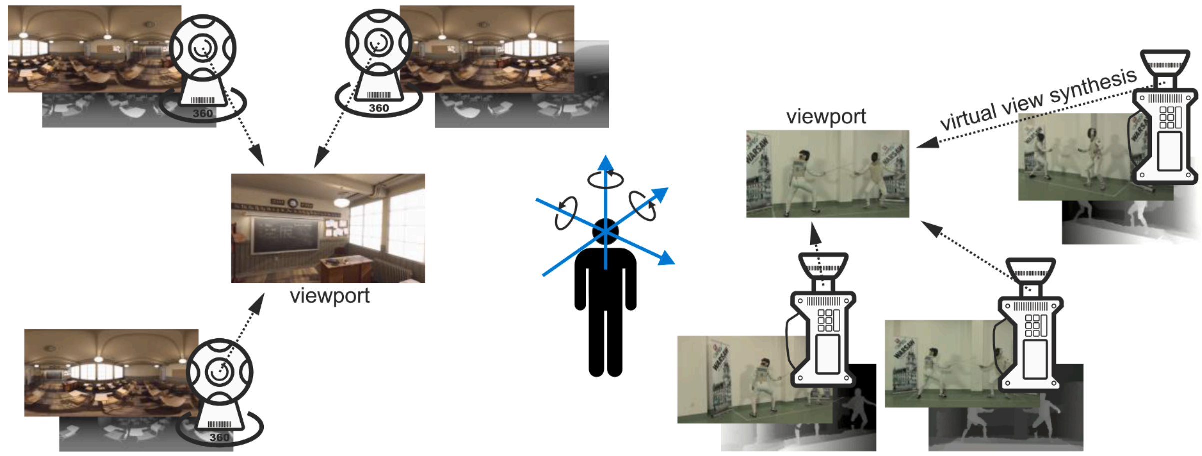

Fig. 1. The idea of immersive video.

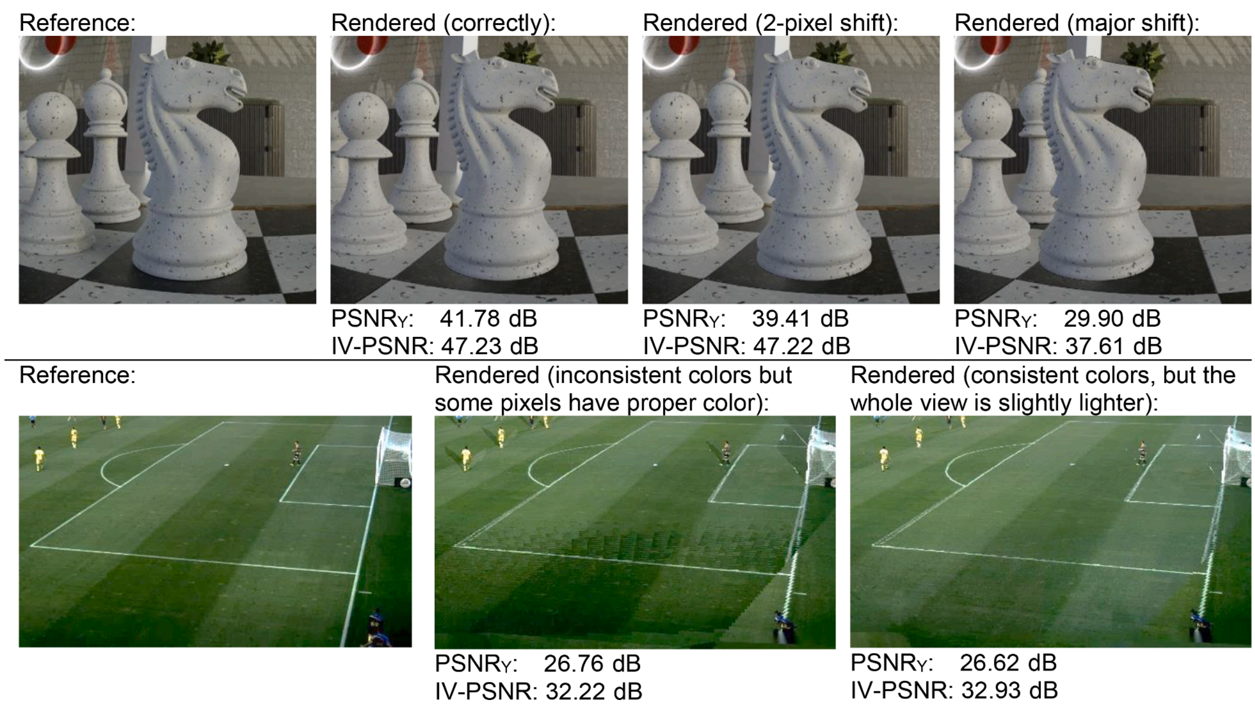

Fig. 2. Typical artifacts in immersive video and objective quality measured by PSNR and IV-PSNR: slight pixel shift (top) and change of illumination (bottom). Examples from [4].

significantly (the PSNR of the luma component smaller by more than 2 dB). On the other hand, the value of IV-PSNR does not change, being in line with the subjective perception of the quality. Moreover, the second row of Fig. 2 shows that the IV-PSNR metric is insensitive to slight change of the global illumination of the scene which is barely perceptible to a viewer.

The high correlation between the IV-PSNR metric and subjective quality was appreciated by the Experts of the ISO/IEC JTC1/SC29/WG11 MPEG standardization group, and the IV-PSNR software has been used for immersive video quality evaluation since 2019 when it became the ISO/IEC MPEG reference software for quality assessment of immersive video [5]. As the reference software, it allowed for reducing the necessity of performing tedious subjective tests for evaluation of new coding techniques proposed to be included in the ISO/IEC 23090–12: MPEG immersive video (MIV) standard [6]. Therefore, it can be stated that the IV-PSNR software contributed to the development of the MIV standard.

Outside of the MPEG and development of the MIV [7,8], the IV-PSNR software is commonly used in research regarding various aspects of immersive video processing (e.g., [9–12]) and other research fields (i.e., lightfield video [13] or even medicine [14]).IV-PSNR is a full-reference objective quality metric, therefore the input of the IV-PSNR software comprises two raw video files (in YUV format): reference and tested video. In contrast to several state-of-the-art objective quality metrics designed for multiview video [15–17], IV-PSNR is designed to assess the quality of the rendered viewports, thus it does not require access to input views and depth maps.

Besides two video files, the software requires several input parameters, such as video resolution, bit depth, chroma subsampling format, etc. The list of input parameters, as well as examples of how to use the software, are included in the manual, and available at the repository.

The IV-PSNR software outputs three objective quality metrics: IV-PSNR, PSNR, and WS-PSNR [18]. All the output values are averaged over all frames being assessed. Both the PSNR and WS-PSNR values are

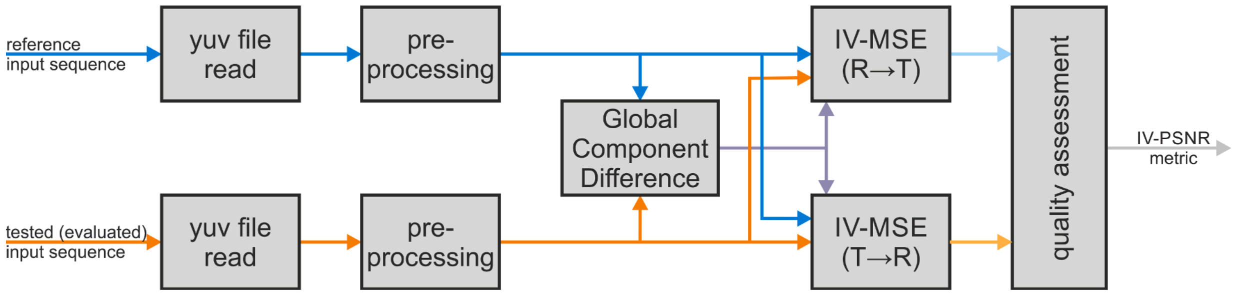

Fig. 3. General scheme of the IV-PSNR software.

provided separately for each color component (Y, C_B, and C_R). The value of the IV-PSNR combines the quality of all color components [4].

2. Software description

The IV-PSNR software was developed as free and open-source software available under the 3-clause BSD license. The software was written in portable C++ language (to be precise, modern C++17 revision).

2.1. Software architecture

The IV-PSNR implementation is divided into a set of underlying libraries and a console application. This organization makes it suitable for use as both a standalone tool and inclusion in other software projects.

As the IV-PSNR software can be considered a "measurement tool", the authors pay significant attention to developing a tool that is easy to use, fast, reliable, and generates reproducible results. Therefore, the IV-PSNR tool is a console application designed following the "Unix philosophy" – "do one thing and do it well". The application takes its input arguments from the commandline and/or configuration file (see README.MD in the code repository).

The high performance and short computation time were achieved by solicitous optimization at every possible level. At the most general level, the calculation of the IV-PSNR metric is algorithmically optimized to avoid unnecessary operations and improve data reuse. All computation-heavy operations are parallelized to exploit the performance of modern multi-core CPUs. Moreover, the authors decided to develop and use their own implementation of a thread pool. The reason for using dedicated thread pool implementation is the significant overhead of a simpler, commonly used OpenMP-based approach. At the lowest level, all critical functions were implemented using instructions operating on vectors of data. The x86–64 SIMD instructions from SSE4.1, AVX2, and AVX512 instruction sets were used [19].

The numerical stability, accuracy, and repeatability of results were achieved by avoiding (or at least reducing) any errors related to floating-point computations while preserving high performance and low computational complexity. In a significant part of computations, the rounding and accumulation errors were effectively eliminated by performing calculations in the integer numbers domain. For example, the IV-PSNR metric calculation is purely integer up to row-level accumulated distortion. At higher levels (picture and sequence), all floating-point values are stored in dedicated buffers and the Kahan-Babuška-Neumaier Summation (KBNS, [20]) algorithm is used to mitigate floating point related errors.

The IV-PSNR application uses a CMake-based build system. The build system is designed to create the fastest possible binary by using inter-procedural optimization and exploiting processor instruction set extensions. On x86–64 compatible microarchitectures, the build system can create four versions of compiled code, each optimized for one predefined x86–64 Microarchitecture Feature Levels (x86–64, x86–64-v2, x86–64-v3, x86–64-v4) [21]. The final binary consists of these four optimized variants and a runtime dynamic dispatcher. The dispatcher uses CPUID [22] instruction to detect available instruction set extensions and select the fastest possible code path.

The metric calculation is processed in several stages illustrated in Fig. 3. The first stage is the reading of the picture (or sequence) data from the source YUV file. The read picture is then preprocessed. The preprocessing includes:

- Conversion of input data chroma format (4:2:0 or 4:2:2) to common format 4:4:4 (if necessary) to achieve size parity of each input component.
- Pixel values validation – the software scans the content of the YUV file in order to evaluate if any pixel values exceed the range $[0,\ 2^{BitDepth}-1]$ since may lead to unreliable metric values.
- Padding loaded image with margins to avoid irregularities during searching of the most similar pixel.

The main stage is the calculation of metrics for each picture. The main goal of the described application is to calculate the IV-PSNR metric. However, the computational complexity of PSNR and WS-PSNR was measured as negligible thus these metrics are also calculated. The computation of IV-PSNR begins with the calculation of the Global Component Difference [4]. Later, the $IV-MSE_c^{R\to T}$ and $IV-MSE_c^{T\to R}$ (mean square error adapted for immersive video characteristics, cf. [4]) values for component c are calculated – from reference (R) to tested (T) picture and from tested picture to the reference one. Finally, $IV-MSE_c^{R\to T}$ and $IV-MSE_c^{T\to R}$ are converted to $\mathrm{IV}-\mathrm{PSNR}_c^{R\to T}$ and $\mathrm{IV}-\mathrm{PSNR}_c^{T\to R}$, respectively. These per-component values are combined with weights 4.0, 1.0, and 1.0 for Y, C_B, and C_R, respectively. The IV-PSNR value of the picture is then calculated as a minimum of $\mathrm{IV}-\mathrm{PSNR}_{YCbCr}^{R\to T}$ and $\mathrm{IV}-\mathrm{PSNR}_{YCbCr}^{T\to R}$.

The last stage is the calculation of sequence statistics. The per-frame IV-PSNR values (stored in a dedicated buffer) are averaged, once again using KBNS to mitigate floating inaccuracies at this stage. Optionally, statistics related to the average computation time of each operation are generated.

2.2. Software functionalities

The IV-PSNR software is intended to operate on YUV files as input data format. The "YUV" file is a simple lossless raw format, commonly used in research on video compression. This format is used to store video sequences (or pictures) in YCbCr color space with various chroma formats (4:4:4, 4:2:2, or 4:2:0) and a wide range of bit depths.

The software was written in a way to accept all commonly used chroma formats (4:4:4, 4:2:2, or 4:2:0), bit depths in the range of 8–14 for input data, and 8–16 for masks. The user is able to choose the starting frame for each input sequence separately and set the number of consecutive frames to be processed. Moreover, the software pre-calculates the total number of frames in both compared sequences in order to avoid end-of-file access when a user selects frames which do not exist. As mentioned in Section 2.1, the input data are validated to avoid calculating unreliable metric values.

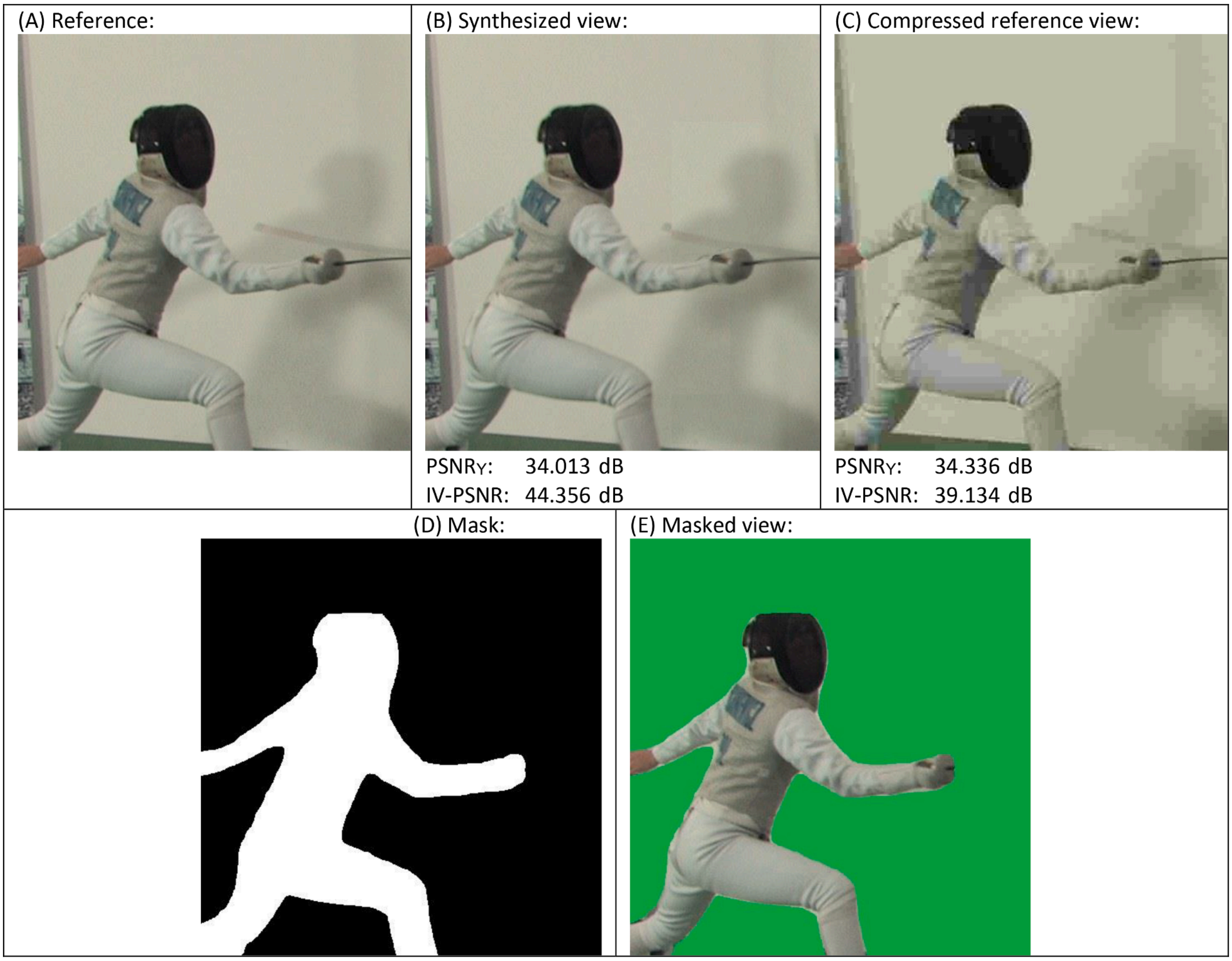

Fig. 4. Fragments of a single view (v8), *Fencing* sequence. A: reference view, B: view v8 synthesized using views v7 and v9, C: reference view heavily compressed with JPEG, D: binary mask (used in masked mode), E: masked view (green area excluded from quality assessment in masked mode).

The software includes support for omnidirectional video stored in equirectangular representation. In the case of video being only part of a sphere (e.g., a hemisphere), the user can provide the lateral and longitudinal ranges.

Since version 4.0, IV-PSNR software is capable to assess the quality not only for the entire frame but also only for selected parts of the image, e.g., a chosen region of interest or a single object. Such a capability allows for the use of this software also in augmented reality (AR) as well as in virtual reality (VR) systems and scenarios.

The default setup uses an internal thread pool set to use all available threads. However, the user can manually set the number of worker threads or even completely disable the use of a thread pool.

All necessary input parameters can be passed by commandline arguments or/and configuration files. It is possible to use multiple configuration files. In such a case, values from the later (in commandline order) configuration files overwrite values from earlier ones.

3. Illustrative examples

In this section, the authors have presented two examples showing typical usage of the IV-PSNR software. Both examples, including input video files and the experiment pipeline, are available at the repository, in the 'example' directory. In order to avoid excessive growth of the repository size, only the first frame from each sequence was provided.

3.1. Full frame quality assessment

The first example presents a typical application of the IV-PSNR software – quality assessment of the synthesized (rendered) virtual view. As mentioned in Section 1, the synthesized video contains a specific type of distortion – slight changes in the position of the objects. Such a change is almost unnoticeable for the viewer (cf. Fig. 4A and 4B), nonetheless significantly influences the PSNR values. For comparison, two distorted images were shown in Fig. 4: Fig. 4B with synthesis artifacts and Fig. 4C with errors caused by strong compression. The values of the PSNR metric are very similar for both examples, however, one can notice that Fig. 4C contains severe block artifacts and can be considered significantly more distorted. When the IV-PSNR metric is considered, the quality of the view presented in Fig. 4B was measured as ~44 dB which is significantly higher than ~39 dB for the compressed picture. This difference matches the subjective quality perceived by viewers.

3.2. Masked mode – single object quality assessment

In the case of augmented reality (AR) and virtual reality (VR) scenarios, it is often necessary to measure the quality of a specific object or selected part of a picture. To meet such AR and VR requirements, the IV-PSNR tool contains a masked mode. In this mode, a user can provide a mask video (Fig. 4D) which restricts metric calculation to selected regions (Fig. 4E). All the remaining areas of the views (green area in Fig. 4E) are not involved in the quality assessment.

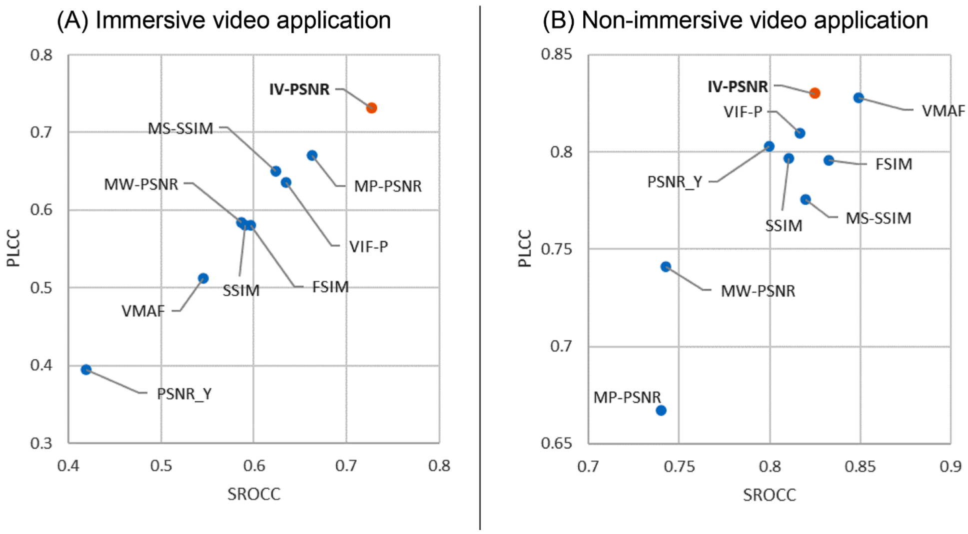

Fig. 5. SROCC and PLCC values for considered metrics in two applications: A: immersive video, B: non-immersive video.

4. Impact

The IV-PSNR software is designed for efficient and reliable objective video quality assessment in immersive media applications, including virtual, mixed, and augmented reality systems. It was developed during works on the ISO/IEC 23090–12 MPEG immersive video (MIV) standard [6], which exposed weaknesses of existing objective quality metrics in immersive video applications. As it was proven in [4], the IV-PSNR metric mimics the subjective quality of immersive video perceived by a human visual system much better than other state-of-the-art objective quality metrics.

A good correlation between the IV-PSNR metric and MOS (mean opinion score) resulted in appreciation by the experts of the ISO/IEC JTC1/SC 29/WG 04 MPEG Video Coding (VC) standardization group. The first version of the IV-PSNR software became the ISO/IEC MPEG VC reference software [5], and the IV-PSNR metric is included in the Common test conditions for MPEG immersive video (MIV CTC, [23]).

As a part of the MIV CTC, the IV-PSNR software simplified the process of quality evaluation of new techniques proposed for the MIV [7,8], which reached the status of the international standard in August 2023 [6].

The IV-PSNR software is being used in various immersive video applications, both within and outside the ISO/IEC MPEG VC standardization activities. It was used for objective quality evaluation in over 250 MPEG documents (since October 2019). Moreover, the IV-PSNR metric (thus also the IV-PSNR software) was used in ∼40 research publications regarding immersive video processing (the total number of references to [4] and [5], according to Google Scholar).

Fig. 5 presents a comparison of the IV-PSNR metric with eight state-of-the-art objective quality metrics: $PSNR_Y$, VMAF [24], FSIM [25], VIF-P [26], SSIM [27], MS-SSIM [28], and two metrics designed for immersive video – MP-PSNR [29] and MW-PSNR [30]. The comparison is performed in terms of two commonly used correlation coefficients: PLCC (Pearson linear correlation coefficient) and SROCC (Spearman rank-order correlation coefficients, [31]). A more detailed comparison (including 30 quality metrics and 4 databases) was presented in [4].

Detailed descriptions of both experiments can be found in Sections VII and IX of [4], respectively.

Fig. 5A presents the results obtained for the typical immersive video system scenario, in which various algorithms of video processing and

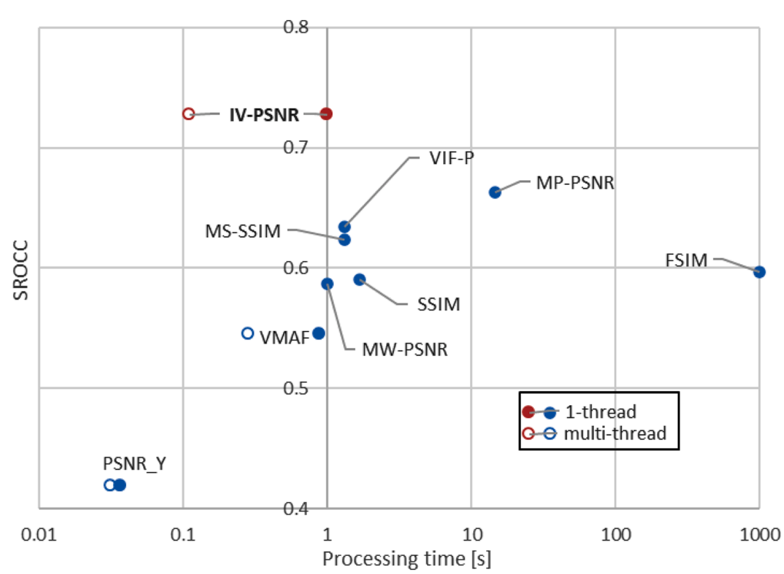

Fig. 6. SROCC and computational time (time required for quality assessment of a single frame of 4 $K \times 4$ K sequence) for considered metrics; immersive video application.

encoding were evaluated. As presented, when used for immersive video applications, the IV-PSNR metric clearly outperforms state-of-the-art objective quality metrics (including other metrics designed for assessment of immersive video, such as MP-PSNR [29] and MW-PSNR [30]).

Fig. 5B contains the results of the evaluation on the TID2013 database [32], which includes static images distorted in numerous ways. This experiment has shown that the IV-PSNR metric could be effectively used also in non-immersive applications (in contrast to other metrics adapted specifically to immersive video, i.e., MP-PSNR [29] and MW-PSNR [30]).

The IV-PSNR software is optimized in terms of computational time required for quality assessment, and its multi-thread implementation is significantly faster than other tested state-of-the-art metrics (except for trivial $PSNR_Y$ metric, Fig. 6). Such an advantage enables the possibility of using this software in real-time immersive video systems, e.g., providing feedback for rate control algorithms.

Regarding the facts of high correlation with MOS (subjective quality) and very short computational time, the IV-PSNR software allows the researchers working on immersive video to assess the performance of their video processing techniques properly and instantly. Therefore, conducting research and developing new algorithms dedicated to immersive video became much easier and more accessible.

5. Conclusions

In this work, we proposed the IV-PSNR software, which allows to reliably assess the immersive video quality by measuring the quality of rendered virtual views. The software measures the similarity between two raw video files and outputs the value of the IV-PSNR metric, which properly handles artifacts typical to immersive video, such as pixel reprojection inaccuracy and inconsistent color characteristics of different cameras [4].

The IV-PSNR software is an ISO/IEC MPEG VC standardization group's reference software for immersive video quality assessment [5]. As reference software, IV-PSNR was used for the evaluation of new coding techniques proposed to be included in the new ISO/IEC 23090–12: MPEG immersive video standard [6]. Moreover, IV-PSNR is commonly used in research on various aspects of immersive video, significantly simplifying the quality evaluation step thus allowing for faster development of new algorithms for immersive video processing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

The work was supported by the Ministry of Education and Science of Republic of Poland.

References

[1] Wien M, Boyce J, Stockhammer T, Peng WH. Standardization status of immersive video coding. IEEE J Emerg Sel Top Circuit Syst Mar. 2019;9(9):5–17.

[2] Müller K, Merkle P, Wiegand T. 3-D video representation using depth maps. Proc IEEE Apr. 2011;99(4):643–56.

[3] A. Dziembowski, D. Mieloch, O. Stankiewicz, and M. Domański, "Virtual view synthesis for 3DoF+ video," 2019 Picture Coding Symposium (PCS), Ningbo, China, Nov. 2019.

[4] Dziembowski A, Mieloch D, Stankowski J, Grzelka A. IV-PSNR – the objective quality metric for immersive video applications. IEEE Trans Circuits Syst Video Technol Nov. 2022;32(11):7575–91.

[5] ISO/IEC JTC1/SC29/WG11 MPEG N18709, "Software manual of IV-PSNR for immersive video", *editor* A. Dziembowski, Jul. 2019.

[6] International standard "ISO/IEC 23090-12:2023 Information technology — coded representation of immersive media — Part 12: MPEG immersive video," Aug. 2023.

[7] Boyce J, et al. MPEG immersive video coding standard. Proc IEEE Sep. 2021;119 (9):1521–36.

[8] Vadakital V, et al. The MPEG immersive video standard – current status and future outlook. IEEE Multimedia Sep. 2022;29(3):101–11.

[9] Ferrara S, et al. The next frontier for MPEG-5 LCEVC: from HDR and immersive video to the metaverse. IEEE Multimedia Dec. 2022;29(4):111–22.

[10] Milovanović M, Henry F, Cagnazzo M. Depth patch selection for decoder-side depth estimation in MPEG immersive video. In: 2022 Picture Coding Symposium (PCS). San Jose, CA, USA; Dec. 2022. p. 343–7.

[11] Fachada S, Bonatto D, Teratani M, Lafruit G. Polynomial image-based rendering for non-Lambertian objects. In: 2021 International Conference on Visual Communications and Image Processing (VCIP). Munich, Germany; Dec. 2021. p. 1–5.

[12] Jeong JB, et al. Towards viewport-dependent 6DoF 360 video tiled streaming for virtual reality systems. In: 28th ACM International Conference on Multimedia; Oct. 2020. p. 3687–95.

[13] Khosroshahi HR, et al. Assessment of multi-plenoptic 2.0 camera depth maps for DIBR. In: International Workshop on Advanced Imaging Technology (IWAIT). Jeju, Korea; Mar. 2023.

[14] Sancho J, et al. SLIMBRAIN: augmented reality real-time acquisition and processing system for hyperspectral classification mapping with depth information for in-vivo surgical procedures. J Syst Archit Jul. 2023;140.

[15] Liu X, Zhang Y, Hu S, Kwong S, Jay Kuo CC, Peng Q. Subjective and Objective video quality assessment of 3D synthesized views with texture/depth compression distortion. IEEE Trans Image Process Dec. 2015;24(12):4847–61.

[16] Shao F, Yuan Q, Lin W, Jiang G. No-reference view synthesis quality prediction for 3-d videos based on color–depth interactions. IEEE Trans Multimedia Mar. 2018;20 (3):659–74.

[17] Li L, Huang Y, Wu J, Gu K, Fang Y. Predicting the quality of view synthesis with color-depth image fusion. IEEE Trans Circuits Syst Video Technol Jul. 2021;31(7): 2509–21.

[18] Sun Y, Lu A, Yu L. Weighted-to-spherically-uniform quality evaluation for omnidirectional video. IEEE Signal Process Lett Sep. 2017;24(9):1408–12.

[19] E. Demikhovsky, "Intel® AVX-512 Architecture. Comprehensive vector extension for HPC and enterprise," LLVM Developers' Meeting, San Francisco, USA, 2013.

[20] Neumaier A. Rundungsfehleranalyse einiger Verfahren zur Summation endlicher Summen. J Appl Math Mech 1974;54(1):39–51.

[21] H.J. Lu et al., "System V application binary interface," AMD64 architecture processor supplement, Version 1.0, May 2021.

[22] Intel. "Intel® 64 and IA-32 architectures software developer manuals," vol. 3B. System Programming Guide; 2011.

[23] ISO/IEC JTC1/SC29/WG04 MPEG VC N0342, "Common test conditions for MPEG immersive video," *editors* A. Dziembowski, B. Kroon, and J. Jung, July 2023.

[24] The Netflix Tech Blog, 2016. Accessed: Sep. 2023. [Online]. Available: https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-653f208b9652.

[25] Zhang L, Zhang L, Mou X, Zhang D. FSIM: a feature similarity index for image quality assessment. IEEE Trans Image Process Aug. 2011;20(8):2378–86.

[26] Sheikh HR, Bovik AC. Image information and visual quality. IEEE Trans Image Process Feb. 2006;15(2):430–44.

[27] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error measurement to structural similarity. IEEE Trans Image Process Apr. 2004;13: 600–13.

[28] Wang Z, Simoncelli EP, Bovik AC. Multiscale structural similarity for image quality assessment. In: 37th Asilomar Conference on Signals, Systems & Computers; Nov. 2003. p. 1398–402.

[29] D. Sandić-Stanković, D. Kukolj, and P.Le Callet, "DIBR synthesized image quality assessment based on morphological pyramids," 3DTV-CON, Lisbon, Portugal, pp. 1–4, Jul. 2015.

[30] Sandić-Stanković D, Kukolj D, Callet PLe. DIBR-synthesized image quality assessment based on morphological multi-scale approach. EURASIP J Image Video Process Dec. 2016;2017(1):1–23.

[31] Myers JL, Well A. Research design and statistical analysis. London, UK: Lawrence Erlbaum Associates; 2003.

[32] Ponomarenko N, et al. Image database TID2013: peculiarities results and perspectives. Signal Process Image Commun Jan. 2015;(30):57–77.