A STUDY ON SOFTWARE TOOLS FOR DETECTING FRAME ERRORS AND BOUNDARY DISTORTIONS IN VIDEO STREAMS

Anastasia Puziy
Associate Professor, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi
Mukhriddin Arabboev
Associate Professor, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi
Shohruh Begmatov
Associate Professor, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi

https://doi.org/10.5281/zenodo.14023652

ARTICLE INFO

Qabul qilindi: 25- Oktyabr 2024 yil
Ma'qullandi: 28- Oktyabr 2024 yil
Nashr qilindi: 31- Oktyabr 2024 yil

KEYWORDS

frame error detection, video processing tools, boundary distortion, video, software tool, video streams.

ABSTRACT

Video quality is crucial in ensuring the successful delivery and consumption of digital content across various platforms, such as streaming services, broadcasting, and surveillance systems. However, visual distortions and frame errors can significantly degrade the quality of video streams, affecting both the user experience and automated processing systems. This paper reviews various software tools for detecting and analyzing frame errors and boundary distortions in video streams.

Introduction

The widespread use of digital video content in applications like streaming media, surveillance, remote sensing, and video conferencing has created a need for maintaining high-quality video standards. Despite advancements in video compression and transmission technologies, visual distortions and frame errors continue to be common issues in video streams. These issues often lead to reduced quality and customer dissatisfaction. Distortions can occur due to encoding and decoding errors, transmission loss, and improper video processing.

Frame errors, such as dropped or repeated frames, and boundary distortions, such as jagged edges or blurring around objects, are among the most common types of visual anomalies encountered in video streams. These errors not only degrade visual quality but can also hinder automated video analysis tasks like object detection, tracking, and scene understanding. Detecting and mitigating these errors is, therefore, essential for ensuring high-quality video output. To address these challenges, various software tools have been developed to identify and analyze frame errors and boundary distortions in video content. This paper aims to provide a comprehensive study of the existing software tools used for detecting frame errors and boundary distortions in video streams.

Analysis of Existing Software Tools for Detecting Frame Errors and Boundary Distortions in Video Streams

This section presents a detailed analysis of current software tools used for detecting frame errors and boundary distortions in video streams. This section presents nine existing software tools for detecting frame errors and boundary distortions in video streams. It includes information about features, applications, advantages, and disadvantages of these software tools.

MSU Video Quality Measurement Tool

The MSU Video Quality Measurement Tool (VQMT), developed by Moscow State University's Graphics & Media Lab, is a specialized software designed for evaluating and benchmarking video quality[1]. This tool is highly regarded in both academic and industrial sectors due to its extensive set of video quality metrics and its ability to assess the performance of various video codecs, transmission scenarios, and processing techniques. Its versatility and accuracy make it a go-to solution for researchers, codec developers, and video production teams aiming to ensure optimal video quality.

One of the tool's strengths lies in its wide range of quality metrics, such as PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index), and VMAF (Video Multi-Method Assessment Fusion), among others. These metrics are designed to measure both objective pixel-level differences and perceived visual quality, offering a balanced evaluation of video performance. Additionally, the MSU tool includes advanced metrics like PSNR-HVS, which factors in human visual perception, and DE100 for color distortion analysis. Frame error detection is another key feature, allowing the software to identify issues such as missing, repeated, or corrupted frames, as well as temporal inconsistencies like jitter.

The tool also excels in visual distortion analysis, evaluating common problems like blocking, ringing, and blurring, as well as boundary distortions in image objects. These analyses are particularly useful for ensuring edge integrity in video frames. The tool generates comprehensive reports that include graphical plots and visual overlays to highlight specific areas of distortion or frame errors, making it easier for users to pinpoint the exact location and nature of quality issues. With support for batch processing and command-line automation, the tool can efficiently handle multiple video files and integrate into larger analysis workflows.

The MSU tool finds application in codec and compression evaluation, transmission error analysis, and video quality benchmarking. It is frequently used to compare the effects of different video codecs and compression settings on quality, and it is especially valuable in streaming and broadcasting environments where transmission errors can degrade video performance. Researchers and industry professionals rely on the tool to benchmark video processing algorithms and to detect frame-level errors, such as dropped frames and jitter, that could impact playback quality.

Available exclusively on Windows, the tool comes in several versions: a free version with basic features, a Pro version with advanced metrics and support for high-resolution files, and an Enterprise version tailored for industrial applications. While the tool offers detailed visual and quantitative reports, its platform limitations and complex interface may pose challenges for new users. Nonetheless, it remains a leading tool in academic research, codec development, and quality assurance, helping to maintain high standards of video quality in a range of applications.

FFmpeg (with Custom Scripting)

FFmpeg is an open-source multimedia framework that excels in a wide range of video and audio processing tasks, including decoding, encoding, transcoding, and streaming [2]. Beyond its common uses, FFmpeg is highly versatile for detecting and analyzing frame errors and distortions in video streams through custom scripts. It offers powerful command-line utilities for frame-by-frame analysis, anomaly detection, and quality assessment, making it an invaluable tool for identifying issues such as missing, duplicated, or corrupted frames and visual distortions like boundary anomalies. Its flexibility and integration with other programming tools further expand its utility in both research and production environments [3].

FFmpeg's key features include robust frame analysis and error detection capabilities. By analyzing frame continuity, timestamps, and pixel-by-pixel differences, it can identify missing frames, dropped frames, or repeated frames in video streams [4]. Additionally, FFmpeg provides boundary distortion detection by applying image processing filters such as edge detection and contour analysis, highlighting issues in object boundaries within frames. Custom scripts can be developed using FFmpeg's filters and encoding parameters to automate the detection of frame errors and visual artifacts, enabling highly tailored video analysis workflows [5]-[6].

Moreover, FFmpeg supports the calculation of video quality metrics like Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). These built-in filters allow users to generate heatmaps and visual overlays, making it easier to compare original and distorted video frames and assess overall visual quality. FFmpeg's integration capabilities with external tools like OpenCV and MATLAB enhance its analytical power, allowing for more advanced analyses, while its ability to export video frames to image formats enables detailed external processing.

Available on Windows, Linux, and macOS, FFmpeg is open-source and free under LGPL or GPL licenses, making it accessible to a wide user base. Its flexibility and support for various multimedia formats and codecs make it a preferred tool for tasks such as video quality monitoring, automated testing, and post-processing optimization. However, its reliance on command-line interfaces and scripting may present a steep learning curve for those unfamiliar with video processing. Despite this, FFmpeg remains an essential tool in video analysis pipelines, offering extensive customization and integration potential for detecting and correcting video stream errors.

OpenCV

OpenCV (Open Source Computer Vision Library) is a widely-adopted open-source library designed for computer vision and image processing tasks [7]. Its comprehensive suite of tools and algorithms enables the analysis and manipulation of images and video streams, making it a versatile option for detecting frame errors and boundary distortions in video content [8]. OpenCV's broad language support, including C++, Python, and Java, makes it accessible for both academic research and industrial applications. Whether used for object detection, segmentation, or motion analysis, OpenCV provides robust capabilities for frame-by-frame analysis and visual quality assessment.

One of OpenCV's core features is its ability to perform frame-by-frame analysis, which is crucial for identifying missing, duplicated, or corrupted frames. By extracting and comparing individual frames, OpenCV can detect anomalies and temporal inconsistencies that degrade video quality. Additionally, OpenCV offers multiple edge detection algorithms, including Canny, Sobel, and Laplacian of Gaussian (LoG), which are essential for identifying strong edges and boundaries in video frames. These edge detection methods help to pinpoint boundary distortions, enabling precise analysis of object edges and contours.

Contour detection and analysis is another key feature of OpenCV. Using built-in functions like “findContours” and “approxPolyDP”, the library allows for the extraction and analysis of object contours, which can be used to assess the smoothness and continuity of boundaries. This makes OpenCV ideal for detecting boundary distortions and other frame-level errors that affect visual quality. Moreover, the library supports motion detection through frame differencing, optical flow analysis, and background subtraction, helping to identify motion-related artifacts like jitter and frame drops.

OpenCV's versatility extends to object detection and segmentation, where it offers various methods such as template matching, Haar cascades, and deep learning-based models. This functionality can be combined with boundary detection to identify and measure distortions in segmented objects. OpenCV also supports a wide range of filtering and noise reduction techniques, such as Gaussian, median, and bilateral filtering, which help preprocess video frames to reduce noise and enhance boundary detection. The library’s seamless integration with other tools, such as NumPy and TensorFlow, further enhances its utility for complex image processing and machine learning tasks.

Available on multiple platforms, including Windows, Linux, macOS, and mobile operating systems like Android and iOS, OpenCV is licensed under Apache 2.0, allowing for both commercial and non-commercial use. While the library offers high performance and efficiency, especially for large datasets and high-resolution videos, it may require fine-tuning for specific tasks, which could be a challenge for beginners. Despite this, OpenCV’s extensive support community and documentation make it a popular choice for video quality control, computer vision research, and automated surveillance systems, where frame error detection and boundary distortion analysis are critical.

PySceneDetect

PySceneDetect is an open-source Python tool designed for detecting scene boundaries and cuts in video files, making it ideal for shot boundary detection and video analysis [9]. The tool automatically detects transitions between scenes based on visual content changes and supports both static cuts (hard cuts) and gradual transitions (fade-ins, fade-outs, dissolves). While PySceneDetect’s primary focus is on identifying scene changes, it also proves useful in detecting frame-level anomalies, such as dropped frames or sudden transitions, thus assisting in quality control and video stream analysis [10].

The tool offers two main detection algorithms: the Content Detector, which identifies hard cuts based on changes in pixel content between frames, and the Threshold Detector, which captures gradual transitions by monitoring changes in average frame intensity. PySceneDetect supports frame-by-frame analysis, allowing users to generate metadata for each frame, such as timestamps and difference metrics, enabling the identification of temporal discontinuities or frame errors. It is compatible with a wide range of video formats and works seamlessly with tools like OpenCV and PyAV for advanced video processing tasks.

PySceneDetect’s flexibility is further enhanced by customizable detection parameters, where users can fine-tune thresholds for scene changes, adjust fade detection sensitivity, and even create custom detectors. This customization makes it easy to adapt the tool for different types of video content and specific needs. Additionally, the tool supports batch processing and automation, allowing users to analyze multiple video files simultaneously and export results in various formats like CSV or JSON, making it suitable for large-scale video analysis projects or integration into larger workflows.

The tool's applications range from shot boundary detection in movies and TV shows to identifying sudden visual anomalies that may indicate frame errors or distortions. It can also be used for content-based video retrieval, where scene-based metadata is generated to create video summaries or assist in automated editing processes by splitting long recordings into individual scenes. With its intuitive command-line interface, Python API, and detailed documentation, PySceneDetect is accessible to both beginners and advanced users.

Available on Windows, Linux, and macOS, and licensed under the BSD 3-Clause License, PySceneDetect is free for both academic and commercial use. While the tool is limited to scene and shot boundary detection and not designed for comprehensive video quality assessment, it excels in its niche. However, detection accuracy may vary depending on the complexity of the video content, particularly in high-motion scenes that may lead to false positives. Despite this, PySceneDetect remains a valuable tool for video editors, analysts, and developers working on scene detection and video analysis tasks.

VideoLAN (VLC Media Player)

VideoLAN's VLC Media Player is a widely-used open-source multimedia player known for its versatility in handling various video and audio formats [11]. Although primarily used for playback, VLC offers several tools for basic video analysis, frame-level inspection, and detection of video stream errors. Through its intuitive graphical user interface (GUI) and powerful command-line interface (CLI), VLC enables users to detect frame drops, corrupted frames, and other anomalies, providing basic diagnostics for video quality issues. These capabilities make VLC not only a popular media player but also a useful tool for quick video analysis and troubleshooting.

One of VLC's standout features is its ability to perform frame-by-frame navigation, allowing users to closely inspect individual video frames for anomalies like missing, repeated, or visually corrupted frames. Additionally, VLC provides real-time playback statistics, offering insights into performance metrics such as dropped frames, lost buffers, and decoding errors. This information can be crucial for diagnosing playback issues and identifying transmission-related distortions, especially when monitoring live streams or network-based video content. Users can also log this data to keep a record of playback errors for later analysis.

VLC further supports codec and stream analysis, displaying critical information such as resolution, frame rate, and codec type, helping users detect potential mismatches or encoding errors that could impact video quality. The player's built-in filters, such as deinterlacing and edge detection, can be applied during playback to highlight distortions or visual artifacts in the video frames. Frame extraction and snapshot capturing are additional tools that allow users to manually isolate and examine specific frames, making VLC a useful option for basic video quality inspection.

For more advanced users, VLC offers a comprehensive command-line interface (CLI) that supports tasks like format conversion, error logging, and precise frame extraction. The CLI commands, such as "rate for playback speed" and "start-time for segment analysis", allow for the automation of video analysis processes. VLC's support for Lua scripting and third-party plugins also enables users to extend its functionality, integrating it with other video analysis tools like FFmpeg and OpenCV for more specialized tasks. While VLC is not designed as a dedicated video quality analysis tool, its ability to combine manual inspection with automated processes makes it a flexible option for basic diagnostics.

Available on a wide range of platforms including Windows, Linux, macOS, Android, and iOS, VLC Media Player is licensed under GPL-2.0+, making it open-source and free for both commercial and non-commercial use. Its extensive format support, user-friendly interface, and real-time performance monitoring make it a go-to solution for initial video quality assessments and troubleshooting. However, compared to dedicated tools like VQMT or PySceneDetect, VLC lacks advanced video quality metrics like SSIM or PSNR, and its manual inspection processes can be time-consuming for large datasets. Nonetheless, VLC remains a highly versatile tool for basic error detection, codec diagnostics, and real-time stream monitoring.

AviSynth

AviSynth is an open-source, script-based video editing and processing tool for Windows, known for its powerful flexibility and customization capabilities [12]. Unlike conventional video editors, AviSynth functions as a frameserver, processing video through text-based scripts in the .avs format. This approach allows users to apply a wide range of video and audio filters for tasks such as frame rate conversion, color correction, deinterlacing, and advanced frame analysis. Its ability to handle frame-level editing and complex workflows has made AviSynth a go-to tool for video enthusiasts and professionals involved in quality control, editing, and diagnostics of video content.

One of AviSynth's core strengths lies in its advanced filtering capabilities and frame error detection. Users can create custom scripts to detect missing, repeated, or corrupted frames, as well as correct issues like jitter and stuttering. The tool provides a comprehensive set of built-in filters such as TemporalSoften, DeHalo, and RemoveGrain to smooth video and reduce artifacts that distort boundaries. For more precise frame-by-frame analysis, filters like Sharpen and EdgeDetect can be applied to enhance boundaries and contours, making it ideal for identifying boundary distortions or visual anomalies. AviSynth's scripting flexibility allows users to perform frame-specific processing through conditional statements, enabling a high degree of customization.

AviSynth also supports third-party plugins, extending its functionality far beyond its default capabilities. Plugins such as TIVTC for inverse telecine, MVTools for motion estimation, and QTGMC for deinterlacing provide advanced video processing options like motion compensation, chroma denoising, and scene change detection. These plugins are crucial for creating sophisticated workflows in video restoration, quality enhancement, and error detection. Additionally, AviSynth integrates seamlessly with other video tools like FFmpeg and VirtualDub, allowing processed video to be re-encoded or further edited, thus broadening its application in video editing and analysis environments.

One of the standout features of AviSynth is its script-based video editing system, where users can perform precise frame-level edits using commands such as Trim, Loop, and Splice. This enables users to cut, join, or modify individual frames with exceptional precision, making it suitable for complex tasks like frame interpolation, time stretching, or format conversion. Moreover, the tool supports visual analysis through overlays and diagnostic filters such as

Histogram, Waveform, and Vectorscope, which help users visualize frame properties and detect visual distortions. The ability to generate difference maps and error overlays further enhances its utility in video diagnostics and quality assessment.

Despite its powerful capabilities, AviSynth requires a strong familiarity with scripting, making it less accessible to beginners. It lacks a graphical user interface, so all operations must be executed through script commands. Furthermore, its functionality is limited to Windows, which can be a drawback for users looking for cross-platform solutions. Nevertheless, for those comfortable with scripting, AviSynth offers unmatched flexibility for video quality analysis, advanced editing, and automated batch processing. Its open-source nature and extensive support for plugins and filters make it an indispensable tool in video processing workflows.

SSIMPlus

SSIMPlus, developed by SSIMWave, is an advanced video quality assessment tool designed to provide objective, accurate measurements of perceived video quality [13]. Unlike the traditional Structural Similarity Index (SSIM), which compares images or videos directly, SSIMPlus predicts video quality based on human visual perception, offering a more realistic evaluation of how viewers experience video content [14]-[15]. The tool accounts for factors such as luminance, contrast, structural information, and even viewing conditions, like screen size and viewing distance. This makes SSIMPlus particularly useful in real-world applications such as video streaming [16], broadcasting, and codec evaluation, where accurate quality assessments are crucial.

One of the main features of SSIMPlus is its ability to generate a comprehensive SSIMPlus Score, which ranges from 0 to 100 and represents the perceived video quality. This score takes into account the properties of the human visual system and goes beyond the limitations of conventional metrics like SSIM, PSNR, or MSE. SSIMPlus offers frame-by-frame quality analysis, allowing users to detect temporal variations such as dropped frames or sudden quality shifts, and can evaluate visual quality across different resolutions and viewing devices. This makes it a versatile tool for both real-time monitoring and in-depth video analysis across a range of devices, from TVs and desktops to smartphones and tablets.

The device-dependent nature of SSIMPlus is another key advantage, as it models different viewing environments to predict how video content will be perceived on various screens [17]. This feature helps content providers and broadcasters optimize video quality for specific devices by factoring in screen size, viewing distance, and resolution. SSIMPlus can detect visual distortions such as compression artifacts, blurring, and color mismatches, enabling video producers to ensure that their content meets quality standards before distribution. Additionally, SSIMPlus supports multiple video codecs and formats, making it suitable for evaluating both lossy and lossless compression methods.

SSIMPlus also excels in generating comprehensive quality reports. These reports include detailed graphs, frame-by-frame analysis, and visual representations of detected distortions, making it easier to identify and address quality issues. The tool supports batch processing, and logs or CSV files can be generated for integration with automated quality control pipelines. Furthermore, SSIMPlus can be integrated into existing video processing workflows through APIs or command-line tools, allowing for seamless integration with popular frameworks like FFmpeg for frame extraction and preprocessing. Real-time quality monitoring is also supported, enabling broadcasters and streaming platforms to detect and address quality drops on-the-fly.

Despite its robust features, SSIMPlus comes with certain limitations. Its commercial licensing may be costly for smaller projects or non-commercial use, and the tool is focused solely on video quality assessment, lacking additional video editing or processing features. Moreover, real-time analysis and large-scale quality evaluations can be resource-intensive, requiring significant computational power. However, its perceptual quality scoring, close correlation with human visual perception, and ability to optimize video for different devices make SSIMPlus an indispensable tool for video streaming platforms, production workflows, and codec research, ensuring that content is delivered with the highest possible quality to end-users.

MATLAB Video Processing Toolbox

The MATLAB Video Processing Toolbox, part of MATLAB's Image Processing and Computer Vision Toolboxes, is a comprehensive suite of tools designed for advanced video processing, analysis, and computer vision applications. It provides an extensive set of functions that allow users to manipulate video streams, detect frame-level errors, and perform detailed quality assessments. The toolbox is widely used by researchers, engineers, and developers to prototype and test algorithms for tasks like frame extraction, object detection [18], motion analysis [19], and video quality evaluation [20]. Its versatility and user-friendly environment make it a valuable tool for both academic research and industrial applications.

One of the primary capabilities of the MATLAB Video Processing Toolbox is frame error detection and analysis. Users can easily identify anomalies such as missing, repeated, or corrupted frames through pixel-level comparisons and frame differencing. The toolbox also supports boundary distortion detection using edge detection and contour extraction methods like Canny, Sobel, and Prewitt. These tools allow users to evaluate the smoothness and continuity of object boundaries, making it possible to detect visual artifacts such as jagged edges or misaligned objects. Moreover, the toolbox offers several video quality metrics, including PSNR, SSIM, and MSE, enabling users to assess the quality of video frames and streams comprehensively.

In addition to frame-level analysis, the toolbox provides powerful functions for object detection, tracking, and segmentation. Algorithms such as Viola-Jones, HOG, and YOLO are available for object detection, while tracking methods like KLT and Kalman Filter are used to follow objects through video streams. Segmentation techniques, including k-means clustering and background subtraction, allow for the separation of foreground and background elements. These capabilities are essential for applications like video surveillance, automated video editing, and real-time video analysis, where identifying and tracking objects is critical.

Motion analysis is another key feature of the MATLAB Video Processing Toolbox. The toolbox supports optical flow methods, such as Horn-Schunck and Lucas-Kanade, for detecting motion anomalies and visualizing motion vectors within video frames. These tools are useful for analyzing motion dynamics, detecting jitter, or correcting motion artifacts caused by frame errors or unstable recordings. By combining motion analysis with frame error detection, users can address a wide range of video quality issues, from simple frame drops to complex motion distortions.

The MATLAB Video Processing Toolbox also offers robust integration with external tools and libraries, such as OpenCV and TensorFlow, for enhanced image and video analysis. It supports custom scripting in MATLAB, allowing users to create complex video processing pipelines tailored to specific needs. Although MATLAB's commercial licensing can be costly and its cross-platform compatibility is limited outside the MATLAB environment, the toolbox provides a powerful, user-friendly platform for video analysis, quality control, and computer vision prototyping. Its extensive documentation and community support make it an indispensable tool for researchers and professionals working on video-related projects.

DaVinci Resolve

DaVinci Resolve, developed by Blackmagic Design, is a professional-grade video editing, color correction, visual effects, and audio post-production software widely used in the film, television, and video production industries [21]. Its robust suite of tools enables high-end video editing, frame error detection, video quality analysis, and content enhancement, making it indispensable for both creative and technical professionals. DaVinci Resolve is available in two versions: a free version and DaVinci Resolve Studio, a paid version offering additional features such as support for 4K+ resolutions, advanced noise reduction, and collaborative workflows.

One of DaVinci Resolve's key strengths is its ability to detect and correct frame errors. The software provides advanced video analysis tools that help identify issues like frame drops, corrupted frames, and visual distortions. Users can view frame integrity using visual scopes and a playback timeline, while features like Power Windows and Bezier Curve Adjustments allow precise corrections to boundary distortions and object contours. The software’s color correction capabilities, which include industry-leading tools for adjusting color balance, contrast, and saturation, also play a vital role in restoring visual quality, especially in footage impacted by distortions or compression artifacts.

DaVinci Resolve integrates Fusion, a powerful node-based compositing environment that enables users to create advanced visual effects and motion graphics. It also includes filters for noise reduction, sharpening, and distortion correction, further expanding its ability to restore and enhance video quality. In terms of motion analysis, DaVinci Resolve employs optical flow algorithms that detect and correct motion artifacts like jitter or frame blending. These tools are essential for stabilizing shaky footage or adjusting motion flow within video sequences, making DaVinci Resolve a comprehensive solution for video production and quality assurance.

A standout feature of DaVinci Resolve is its real-time playback and monitoring capabilities, which allow users to identify frame errors, visual distortions, and boundary issues as they occur. The software supports high-resolution video processing, including up to 8K resolution and HDR formats, making it suitable for high-end video projects. DaVinci Resolve also handles a wide range of professional video formats and codecs, such as ProRes, DNxHR, H.264, and H.265, providing flexibility in various production environments. Additionally, its custom scripting support and compatibility with third-party plugins allow users to automate workflows, integrate with external tools, and extend its functionality for specialized video analysis tasks.

Despite its powerful features, DaVinci Resolve has a steep learning curve, especially for beginners dealing with complex color correction and visual effects. It is also resource-intensive, requiring a high-performance system to handle high-resolution and complex projects efficiently. While it excels in video editing and quality control, DaVinci Resolve lacks some of the automation capabilities found in dedicated video quality assessment tools like SSIMPlus or VQMT. Nonetheless, its versatility and extensive feature set make it a go-to choice for video production, frame error detection, and video quality enhancement in professional settings. It is widely used by broadcasters and streaming platforms to maintain high-quality video standards across various devices and distribution channels.

Comparison of Existing Software Tools for Detecting Frame Errors and Boundary Distortions in Video Streams

This section provides a detailed comparison of software tools based on their functional capabilities.

№	Name of software tool	Applications	Advantages	Disadvantages	Use cases
1.	MSU Video Quality Measurement Tool	• Codec and Compression Evaluation • Transmission Error Analysis • Video Quality Benchmarking • Error and Distortion Detection	Wide variety of metrics, highly accurate, detailed reporting, integrates with other tools.	Limited to Windows, free version lacks some advanced metrics, complex interface for non-experts.	Codec evaluation, video quality benchmarking, transmission error analysis.
2.	FFmpeg (with Custom Scripting)	• Frame Error Detection • Boundary Distortion Detection • Video Quality Assessment	Highly customizable, open-source, integrates with other tools, supports various formats.	Requires scripting knowledge, steep learning curve, manual configuration for quality analysis tasks.	Video quality monitoring, automated testing, post-processing optimization.
3.	OpenCV	• Frame Error Detection • Boundary Distortion Detection • Video Quality Assessment • Object Detection and Motion Analysis	Versatile for image and video processing, supports machine learning, integrates with external libraries.	Requires fine-tuning for specific applications, lacks specialized video quality metrics like VMAF or MS-SSIM.	Video quality control, computer vision research, automated surveillance.

4.	PySceneDetect	• Shot Boundary Detection • Video Analysis and Quality Control • Content-Based Video Retrieval	Simple command-line interface, customizable scene detection, batch processing support.	Limited to scene detection, accuracy depends on video content.	Scene detection in movies/TV shows, video analysis, automated editing.
5.	VideoLAN (VLC Media Player)	• Basic Frame Error Detection • Boundary Distortion Inspection • Codec and Stream Diagnostics • Playback Performance Monitoring	User-friendly, supports virtually all formats, free, supports real-time monitoring.	Limited video quality analysis capabilities, time-consuming for large datasets, not a dedicated analysis tool.	Initial video quality assessment, frame-by-frame inspection, network stream monitoring.
6.	AviSynth	• Frame Error Detection • Boundary Distortion Detection • Video Quality Enhancement • Video Editing and Manipulation	Highly customizable, supports complex workflows, numerous filters and plugins.	Scripting required, Windows-only, lacks a graphical interface.	Video quality analysis, video editing and manipulation, automated batch processing.
7.	SSIMPlus	• Video Streaming and Broadcasting • Codec Evaluation and Optimization • Quality Control for Video Production • Video Quality Benchmarking • Device-Dependent	Accurate human-perception-based scores, real-time analysis, device-dependent evaluation.	Commercial license required, computationally expensive, limited to video quality assessment.	Streaming quality control, codec evaluation, video quality benchmarking.

		Quality Optimization			
8.	MATLAB Video Processing Toolbox	• Frame Error Detection and Correction • Boundary Distortion Detection • Object Detection and Tracking • Motion Analysis	Highly flexible, extensive built-in functions, strong integration with other MATLAB toolboxes.	Expensive licensing, high computational requirements, limited cross-platform support outside MATLAB.	Academic research, quality control, computer vision prototyping.
9.	DaVinci Resolve	• Frame Error Detection and Correction • Boundary Distortion and Contour Detection • Color Grading and Visual Quality Enhancement • Object Detection and Visual Effects	Industry-standard tools, real-time monitoring, powerful compositing and color correction.	High learning curve, resource-intensive, limited automation for quality analysis.	Video production and post-production, frame error analysis, quality enhancement, broadcasting quality control.

Conclusion

In this review, we examined a range of video quality measurement and processing tools, each offering unique features and functionalities tailored to specific use cases. These tools, including both commercial and open-source options, provide diverse solutions for video quality analysis, from objective metrics like PSNR, SSIM, and VMAF to advanced custom scripting capabilities and professional-grade video editing environments. While specialized software such as VQMT and SSIMPlus excels in providing accurate quality assessments and benchmarking across different codecs and devices, open-source tools like FFmpeg and OpenCV offer flexibility and extensive customization for researchers and developers.

Moreover, tools like AviSynth and MATLAB Video Processing Toolbox cater to users requiring detailed frame-by-frame analysis and boundary detection, whereas solutions like PySceneDetect focus on scene segmentation, aiding in content analysis and editing tasks. Professional editing software like DaVinci Resolve combines video quality analysis with robust editing and color correction capabilities, making it ideal for production environments where both quality control and creative editing are necessary.

Choosing a tool depends on factors like the required level of detail, user expertise, platform compatibility, and budget. While commercial options like SSIMPlus and MATLAB offer high accuracy and comprehensive analysis, open-source solutions like VLC and FFmpeg provide cost-effective alternatives with significant flexibility.

References:

1. "MSU Video Quality Measurement Tool," Moscow State University Graphics & Media Lab. [Online].
2. Available: http://www.compression.ru/video/quality_measure/video_measurement_tool_en.html.
3. "FFmpeg Documentation," FFmpeg Official Website. [Online]. Available: https://ffmpeg.org/documentation.html.
4. X. Wu, P. Qu, S. Wang, L. Xie, and J. Dong, "Extend the FFmpeg Framework to Analyze Media Content," arXiv, 2021.
5. H. S. Singha and J. Bhuvana, "A Study on FFmpeg Multimedia Framework," Int. J. Trend Sci. Res. Dev., vol. 5, no. 4, pp. 580–584, 2021.
6. . Li, "Special Treatment of Video Image Based on FFmpeg," in 2018 Joint International Advanced Engineering and Technology Research Conference (JIAET 2018), 2018, vol. 137, no. Jiaet, pp. 266–271.
7. H. Zeng, Z. Zhang, and L. Shi, "Research and Implementation of Video Codec Based on FFmpeg," 2016 Int. Conf. Netw. Inf. Syst. Comput., pp. 184–188, 2017.
8. OpenCV documentation," OpenCV Official Website. [Online]. Available: https://docs.opencv.org.
9. Shubham Mishra, Mrs. Versha Verma, Dr. Nikhat Akhtar, Shivam Chaturvedi, and Dr. Yusuf Perwej, "An Intelligent Motion Detection Using OpenCV," Int. J. Sci. Res. Sci. Eng. Technol., no. March, pp. 51–63, 2022.
10. "PySceneDetect Documentation," PySceneDetect. [Online]. Available: https://pyscenedetect.readthedocs.io.
11. Z. Shou et al., "Online Detection of Action Start in Untrimmed, Streaming Videos," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11207 LNCS, pp. 551–568, 2018.
12. "VLC Media Player Documentation," VideoLAN Organization. [Online]. Available: https://www.videolan.org/doc/.
13. "AviSynth - Script-Based Video Editing," AviSynth Official Site. [Online]. Available: http://avisynth.nl/index.php/Main_Page.
14. "SSIMPlus - Video Quality Metrics for Streaming," SSIMWave. [Online]. Available: https://www.ssimwave.com/.
15. L. Zhang, "Image Quality Assessment and Saliency Detection: human visual perception modeling and applications. Signal and Image Processing. Université de Rennes 1 (UR1)," 2020.
16. Z. Duanmu, K. Zeng, K. Ma, A. Rehman, and Z. Wang, "A Quality-of-Experience Index for Streaming Video," IEEE J. Sel. Top. Signal Process., vol. 11, no. 1, pp. 154–166, 2017.
17. Z. Duanmu, K. Ma, and Z. Wang, "Quality-of-experience of adaptive video streaming: Exploring the space of adaptations," MM 2017 - Proc. 2017 ACM Multimed. Conf., pp. 1752–1760, 2017.

18. A. Rehman, K. Zeng, and Z. Wang, "Display Device-Adapted Video Quality-of-Experience Assessment," vol. 9394, 2015.

19. S. Saranya and A. A. Priya, "Object Detection and Tracking from Video Sequence using MATLAB," Int. J. Innov. Res. Sci. Technol., vol. 4, no. 3, pp. 43–47, 2017.

20. U. P. Nagane and A. O. Mulani, "Moving Object Detection and Tracking Using Matlab," J. Sci. Technol., vol. 6, no. 01, pp. 63–66, 2021.

21. L. S. Alandkar and S. R. Gengaje, "Study of Object Detection Implementation Using Matlab," Int. J. Res. Eng. Technol., vol. 05, no. 08, pp. 109–114, 2016.

22. "DaVinci Resolve," Blackmagic Design. [Online]. Available: https://www.blackmagicdesign.com/products/davinciresolve/.