# Detection of Video Artifacts using AI

**Vinutha Raghavendra[1], Shashank H P[2]**

[1]Post Graduate Student, [2]Professor
Dept. of Computer Science and Engineering Maharaja Institute of Technology Mysore
Mysore, India.

**Abstract: The paper presents the product suite for automation of testing, workflows, and monitoring across video delivery platforms such as mobile, STB, OTT players, consoles etc. The proposed Video QoE (Quality of Experience) includes an Artificial Intelligence engine, to monitor Video anomaly for better Quality of Experience. This is coupled with an integrated development environment, which would enable to monitor the Quality of Experience and detect the anomaly in the videos which would further drive a reduction of manual effort in the detection**.

**Keywords: Deep learning, machine learning algorithms, supervised learning, Faster-RCNN, Inception v2, object detection,Video artifacts.**

## I. INTRODUCTION

~~Introduction to a comprehensive suite designed for automating testing, workflows, and monitoring across various video~~ delivery platforms, including mobile, STB, OTT players, and consoles. Central to this suite is a novel Video Quality of Experience (QoE) framework enhanced by an Artificial Intelligence engine. This AI engine plays a crucial role in detecting video anomalies, thereby enhancing overall viewer experience. The incorporation of a robust development environment further empowers real-time monitoring and anomaly detection, significantly reducing manual effort.This primarily emphasizes the elucidation of design principles and optimal methodologies for developing an AI-driven monitoring system specialized in detecting anomalies in video content. Specifically, it addresses the effective identification of:

- **Macroblocks:** In video frames, macroblocks are crucial components encoded within I- Frames. Variations or inconsistencies in these patterns often indicate anomalies, which the proposed Video QoE system effectively identifies.
- **Pixelation:** Detecting pixelation involves identifying significant blocky artifacts within high-definition images, where blocks equal to or exceeding 0.9 cm can be flagged as anomalies.
- **Blur:** The system is adept at discerning blurred frames, irrespective of whether the blur is intentional or due to motion. Special consideration is given to minimize false positives caused by motion blur.

This further outlines the implementation of an advanced AI engine designed explicitly to detect major video anomalies such as macroblocks, pixelation, and blur. By leveraging these principles, the system significantly reduces manual intervention in anomaly detection processes. It also provides a detailed roadmap for setting up and customizing AI-driven vision models for video anomaly detection, along with covering aspects like inference speed and network efficiency.

## II. RELATED WORK

Various research studies have been conducted to address the issue of detecting various video artifacts using various techniques and compression techniques.

Sashikala Mishra, Shruti Patil [1] conducted research on Video compression techniques aim to minimize the quantity of bits required to encode data, thereby minimizing storage requirements and transmission bandwidth. Mishra and Patil explore deep learning methodologies applied to video compression, emphasizing their effectiveness in achieving a goal of efficient compression and decompression operations. The study highlights advancements in neural network architectures designed for video data, showcasing how advanced learning models enhance compression efficiency while maintaining or improving video quality. The idea of graceful degradation is discussed within the idea of maintaining acceptable user experience under bandwidth constraints or transmission errors, underscoring the importance of robust compression techniques in modern multimedia applications.

Borgo and Chen [2] conducted a comprehensive survey on advancements in video-based graphics and visualization, focusing on two primary aspects: photo-realistic and artistic computer-generated imagery (CGI) from videos, and summary and abstract visualization of videos. They reviewed techniques for generating CGI that closely mimics real-world videos and methods for transforming video content into visually appealing imagery. Additionally, they explored

approaches for creating summary and abstract visual representations of videos to highlight key features and events. Borgo and Chen introduced a new taxonomy to categorize concepts and techniques in this field and discussed the integration of automated video analysis techniques, such as feature extraction, detection, and tracking, into video-based modeling and rendering pipelines. Their work provides valuable insights into current trends and future directions in video-based graphics and visualization.

Bovik [3] addressed the critical task of detecting and mapping impairments in video content, with a focus on improving video quality assessment methodologies. His research explores innovative approaches for objectively quantifying and evaluating video impairments such as pixelation, macroblocks, and blur. Bovik's study emphasizes the development of robust metrics and algorithms to accurately measure and analyze video quality degradation. Key contributions include the introduction of objective metrics for quantitative assessment, incorporation of subjective evaluation methods to capture human perception of video quality, and the application of AI and machine learning techniques to automate the detection and classification of video impairments. Bovik's work advances the field by proposing new methodologies and frameworks that enhance efficiency and accuracy in video quality assessment, ultimately improving user experience in multimedia applications.

Babić et al. [4] conducted a comprehensive study on real-time detection tools for audio and video artifacts, focusing on multimedia quality assessment. Their research develops efficient algorithms and methodologies to identify and mitigate artifacts that compromise the quality of audio and video streams. Key contributions of the study include:

- **Algorithm Development:** The authors introduce novel algorithms capable of detecting various artifacts in real-time, such as pixelation, macroblocks, audio distortions, and visual anomalies.
- **Implementation Framework:** They describe the implementation framework for the detection tool, emphasizing its integration with existing multimedia systems for seamless operation.
- **Performance Evaluation:** Babić et al. provide an extensive evaluation of the tool's performance metrics, including accuracy, speed, and reliability across different multimedia formats and environments.

Their work advances the field by offering practical solutions for real-time artifact detection, significantly improving the quality of audiovisual content delivery and addressing the need for reliable multimedia quality assessment tools.

In the realm of video quality assessment, there exists a critical need for efficient and accurate methods to detect and classify visual artifacts such as pixelation, macroblocks, blur, and other impairments. Current approaches often rely on manual inspection or simplistic algorithms that are insufficient for real-time applications and large-scale video datasets. Moreover, distinguishing between intentional visual effects and undesirable anomalies remains a significant challenge.To tackle these challenges, this project aims to develop an AI-driven system capable of automating the detection and classification of video impairments. The system will leverage machine learning techniques, encompassing deep learning models, to analyze video frames and identify instances of pixelation, macroblocks, blur, and other visual artifacts. By improving the accuracy and performance efficiency of video quality assessment, the proposed system seeks to increase the overall user experience across various multimedia platforms, including streaming services, video surveillance, and digital**. In the existing system**, we examine how video quality assessment is handled. This system can be divided into two main approaches: manual inspection and basic artifact detection.

A) **Manual Inspection:** Manual inspection involves human evaluators reviewing video content to assess quality. This approach relies on individual judgment to identify visual artifacts, which can include pixelation, macroblocks, and blur. The effectiveness of this method depends heavily on the evaluators' expertise and consistency.

B) **Basic Artifact Detection:** Basic artifact detection uses simplistic algorithms to identify common video quality issues. These algorithms are designed for fundamental detection tasks but lack the sophistication needed for robust analysis. They often struggle to differentiate between intentional visual effects and undesirable anomalies.

The existing system for video quality assessment has several limitations, which can be addressed and improved with advanced techniques:

- **Manual Inspection Inefficiency:** Manual inspection is effective but becomes time-consuming and impractical for large volumes of video content. This limitation affects the ability to promptly identify and address quality issues.
- **Algorithmic Limitations:** Basic algorithms are not robust enough to handle complex video artifacts or distinguish between visual effects and anomalies, leading to potential inaccuracies in quality assessment.
- **Scalability Issues:** The existing system lacks scalability, making it challenging to manage and evaluate large-scale video datasets efficiently.
- **Subjectivity in Evaluation:** Manual inspections can be subjective and vary among evaluators, leading to inconsistent assessments and potential oversight of quality issues.

**In the proposed system**, we introduce a comprehensive approach to object detection and video quality assessment using advanced deep learning techniques. The system leverages the Faster R-CNN model to identify

and analyze video impairments with high precision. The implementation is broken down into iterative steps to ensure thorough development and evaluation. The key components and processes of the proposed system are outlined as follows:

[1] Enhanced Object Detection through Advanced Algorithms: The proposed system utilizes the Faster R- CNN model for detecting video impairments such as pixelation, macroblocks, and blur. By training on diverse video datasets, the system enhances its ability to recognize and delineate these impairments with high accuracy, reducing reliance on manual inspections.

[2] Efficient Data Handling and Preprocessing: The system incorporates a structured approach to data collection and curation, ensuring that the video datasets are well-organized and preprocessed for optimal model training. This meticulous preparation improves the efficiency and effectiveness of the training process.

[3] Scalable Model Training and Evaluation: The use of GPU-accelerated platforms for training the Faster R-CNN model allows for scalable and efficient processing of large-scale video datasets. Performance evaluation metrics are employed to assess accuracy and computational efficiency, ensuring robust detection capabilities.

[4] Real-Time Impairment Detection: By implementing advanced algorithms and iterative training steps, the proposed system can detect video impairments in real-time. This capability facilitates prompt identification and mitigation of quality issues, enhancing overall video quality and user experience.

[5] Comprehensive Performance Metrics: The proposed system includes detailed performance evaluations to measure detection rates and computational performance against established benchmarks. This ensures that the system maintains high accuracy and reliability in various multimedia environments.

**Advantages of the Proposed System:**

- **Reduced Workload:** The machine learning algorithms in the proposed system are capable of analyzing large volumes of video data autonomously. This reduces the need for manual inspection, streamlining the process of detecting video artifacts such as pixelation, macroblocks, and blur.
- **Reduced Subjectivity:** Unlike manual evaluation, which can be influenced by personal biases, the machine learning algorithms operate based on objective data patterns. This ensures a consistent and unbiased detection of video artifacts, improving the accuracy of the quality assessment.
- **Improved Scalability:** The system is designed to handle extensive video datasets, making it scalable to different video formats and resolutions. This scalability allows for efficient artifact detection across a variety of multimedia content and platforms.
- **Enhanced User Experience:** By providing accurate and timely detection of video artifacts, the proposed system enhances the overall viewing experience. Users benefit from improved video quality, as artifacts are identified and addressed promptly, leading to a more enjoyable viewing experience.
- **Enhanced Responsiveness:** The machine learning algorithms can be integrated into video processing workflows to detect artifacts in real time. This capability allows for immediate identification and correction of quality issues, ensuring that video content remains at a high standard throughout its distribution.

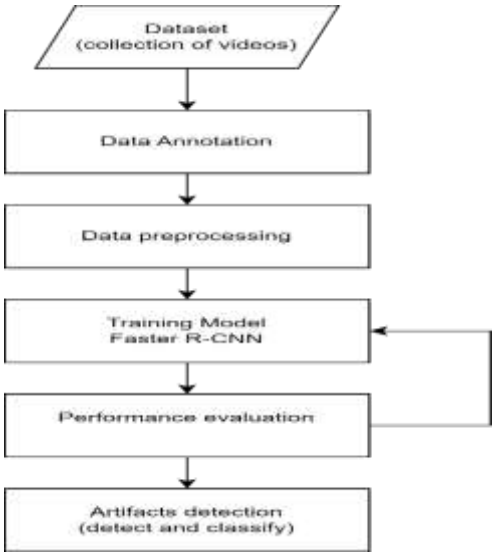
## III. METHODOLOGY

The proposed system for detecting video artifacts utilizes a sophisticated architecture involving Faster R-CNN and Inception-V2 networks. The system's workflow is outlined below and depicted in Figure 1, illustrating the step-by- step process of video artifact detection using machine learning techniques.

Data preprocessing played a crucial role, as the collected videos were carefully formatted and annotated to ensure consistency. This involved cleaning the data, normalizing resolutions, and applying accurate annotations for different types of artifacts such as macroblocks, pixelation, and blur. The preprocessing steps were essential for preparing high-quality input for model training. The core of the methodology was model training, where the Faster R-CNN framework, with Inception-V2 as its backbone, was employed. The Region Proposal Network (RPN) within Faster R-CNN was utilized to generate region proposals, which were then processed to detect and classify artifacts. The use of ROI Pooling ensured that varying sizes of proposed regions were handled effectively, resulting in consistent feature maps and improved detection accuracy.

Performance evaluation of the trained model was conducted through rigorous testing on a separate validation set. Metrics such as accuracy, precision, and recall were used to assess the model's effectiveness in detecting video artifacts. The evaluation process also included testing the model's real-time inference capabilities to ensure practical applicability.

**Figure 1:** Proposed System Workflow




1. **Data Collection:**The initial phase involves gathering a diverse set of video datasets containing examples of various video artifacts such as pixelation, macroblocks, and blur. This collection process is critical as it forms the foundation for training the machine learning model. Videos are sourced from multiple origins to ensure a broad representation of different artifact types, which helps in building a robust model capable of detecting a wide range of impairments.

2. **Data Preprocessing:**Once the data is collected, it undergoes a preprocessing phase where it is formatted and organized to prepare it for effective model training. This step includes cleaning the data to remove any inconsistencies and annotating it in a uniform format. Proper preprocessing ensures that the data is consistent and appropriately labeled, which is essential for accurate and efficient training of the machine learning model.

3. **Model Training:**In the training phase, the Faster R-CNN model, leveraging the Inception-V2 network as its backbone, is utilized to detect video impairments. This process involves configuring various hyperparameters and employing GPU-accelerated platforms to enhance training efficiency. The Faster R-CNN model integrates a Region Proposal Network (RPN) to generate region proposals and a subsequent network that processes these proposals to identify artifacts. To handle varying sizes of proposed regions and ensure consistent feature maps, ROI Pooling is applied, optimizing the model's ability to accurately detect and classify video artifacts.

4. **Performance Evaluation:**After training, the model is evaluated to assess its accuracy and efficiency. Performance metrics are carefully measured to ensure the model achieves the desired detection rates and computational performance. This evaluation involves assessing the model's accuracy in detecting artifacts, its processing speed, and its capability to handle various video formats, ensuring it meets the performance standards required for real-world applications.

5. **Artifact Detection:**Upon successful training, the model is deployed to detect artifacts in new video content. The inference process involves using the trained model to analyze video frames and identify those with artifacts. The model provides probabilities and classifications for detected anomalies, enabling real-time assessment of video quality. The ability to detect and classify artifacts accurately is facilitated by the inference graph created from the training checkpoints, which allows for efficient and precise artifact detection.

## IV. MODELING AND ANALYSIS

The performance of Convolutional Neural Networks (CNNs) in detecting video artifacts is significantly influenced by the network's architecture and design choices. A crucial aspect of this design is the selection of kernel sizes for convolution operations. Different kernel sizes can impact the network's ability to capture various types of features, from fine details to broader patterns.

1. **Kernel Size Selection:**Choosing the appropriate kernel size for convolution operations is challenging due to the significant variation in the location of information within images. Larger kernels are preferred for capturing features distributed more globally, while smaller kernels are better suited for capturing locally distributed features. To address this, filters of multiple sizes are used at the same level in the network, making it wider rather than deeper. This design principle

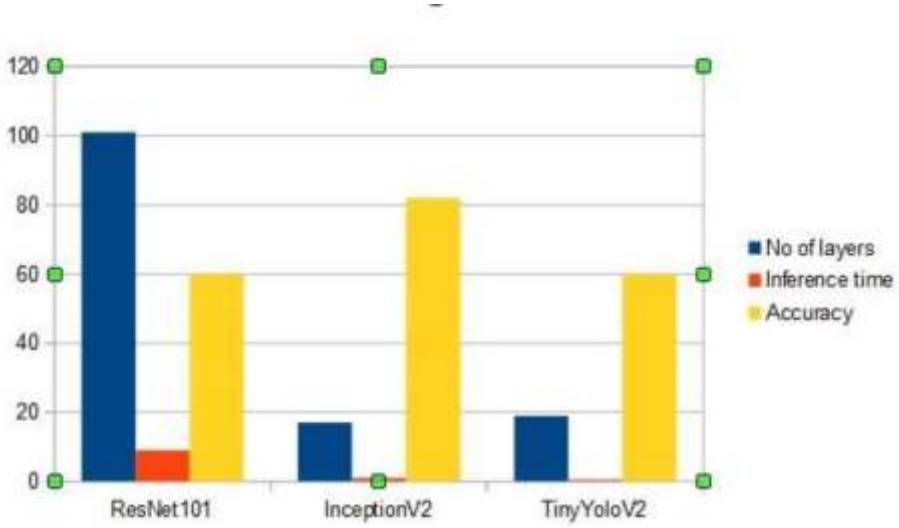
is embodied in the inception module.

2. **Findings from InceptionV2 Layers:Inference Time (per frame)**: Approximately 1 second on a system with 16 GB RAM. **Accuracy**: Achieves 89% accuracy in detecting video artifacts.**Number of Layers**: The network consists of 17 layers. **Learning Rates**: Utilizes various learning rates, including 0.002, 0.0002, 1.9999994948e-05, and 1.9999999495e-06. **Image Resolution (Single Channel)**: Operates on images with a resolution of 600 x 1200 pixels. **CPU Utilization**: During inference, CPU utilization is approximately 62%. **Inference Time (Windows vs. Linux)**: The inference time per frame is notably longer on Windows, averaging 4 seconds per frame. In contrast, the same code runs approximately 8 times faster on Linux, with an inference time of 0.5 seconds per frame. **CPU Utilization on Different OS**: On Linux, CPU utilization during inference is around 30%, while on Windows, it reaches 99%. This difference is due to variations in TensorFlow package implementations between the operating systems.

3. **Algorithm Performance Comparison:**To evaluate the performance of different algorithms for video artifact detection, a comparison was conducted among various models, including ResNet and YOLO. The evaluation focused on inference time and accuracy.**Graph Reference**: Figure X illustrates the comparison of algorithms based on their inference time and accuracy.**Details from Graph**: **ResNet**: Notable for its deep residual learning framework, providing high accuracy but with varying inference times depending on the network depth. **YOLO**: Known for its speed and real-time object detection capabilities, it showed competitive performance in terms of inference time.

**Insights**: The graph highlights the trade-offs between accuracy and inference time across different algorithms. For example, while ResNet offers high accuracy, YOLO provides faster inference times suitable for real-time applications.

**Figure 2:** Comparison of Inference Time and Accuracy Across Different Algorithms




The Inception module's design allows for capturing a wide range of features by using multiple filter sizes in parallel. This helps the network be more flexible and adaptable to varying feature scales, improving overall
detection performance. Effectiveness of Inception modules can be evaluated based on how well they generalize to unseen data and different types of artifacts. The ability to handle diverse feature sizes enhances the robustness of the model in practical applications.

The discrepancy in inference time between Windows and Linux highlights the importance of optimizing software and hardware configurations. Discussing how TensorFlow implementations and system optimizations affect performance can provide useful insights. Exploring ways to optimize TensorFlow and other frameworks for better performance on various operating systems could be a valuable area for future research.

## V. RESULTS and DISCUSSIONS

In evaluating the performance of the proposed system, it is crucial to understand how well the model performs in practical scenarios. The following analysis focuses on the predictive capability of the trained model, particularly how it identifies and classifies artifacts in video frames. **Predicted Frames and Artifact Detection**: The model's efficacy is demonstrated through its ability to accurately detect and classify video artifacts. By applying the trained model to new video content, we can observe the predictions made for each frame. The predicted frames illustrate how effectively the model identifies various artifacts such as pixelation, macroblocks, and blur. These predictions are crucial for assessing the real-world applicability and accuracy of the artifact detection system**.**

**Figure 3:** Actual image v/s Model predicted macroblock frame

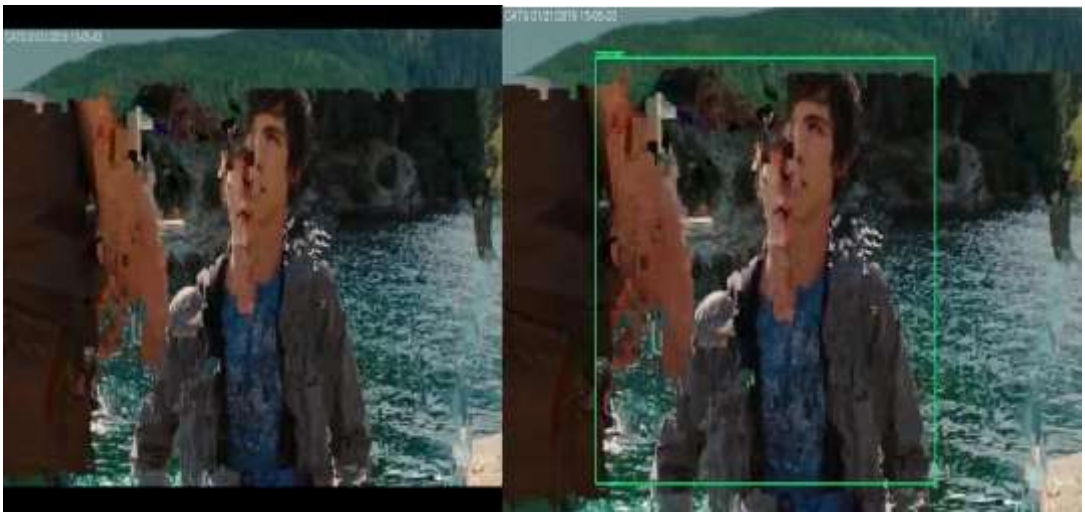
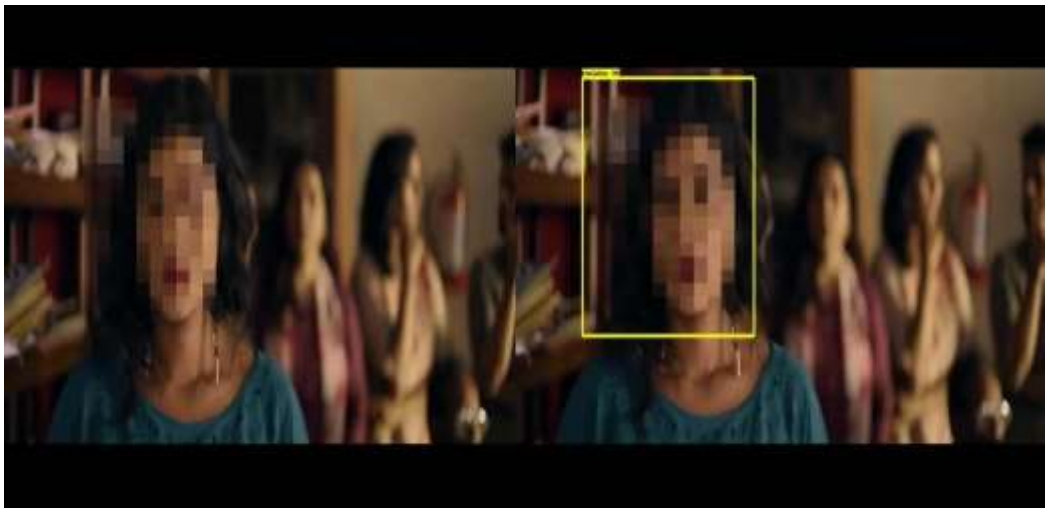


Figure 3 shows the actual image which was fed into the trained model and the model has detected the frame as Macroblock having marked the bounding boxes around the features with the detection confidence score.

**Figure 4:** Actual image v/s Model predicted Pixelated frame



In Figure 4, the original image input into the trained model depicts a frame identified as pixelated. The model has annotated the image, marked with bounding boxes around the pixelation features, accompanied by their respective detection confidence scores.

## VI. CONCLUSION

In conclusion, the development and evaluation of a machine learning model for detecting and classifying video artifacts represent significant advancements in video quality assessment and user experience enhancement. The primary objective was to address challenges posed by video artifacts, such as macroblocks, pixelation, and compression issues, which can negatively impact various applications, including video streaming, surveillance, and digital media production. The implementation of the InceptionV2-based model has proven effective in identifying these artifacts, achieving an impressive average accuracy of 89% across a diverse dataset of video clips. This performance is complemented by a notable real-time inference capability, with an average processing time of just 1 second per frame on standard hardware configurations.

The project has yielded several valuable insights, particularly regarding kernel size selection for convolution operations, **which** significantly affects the model's accuracy in capturing and classifying artifacts. Larger kernels are more effective for globally distributed features, while smaller kernels excel at identifying local features. The incorporation of multi-

scale filters within the model architecture has enhanced its ability to handle artifacts distributed across various spatial frequencies. Additionally, platform-specific performance differences were observed, with the model performing significantly faster on Linux (0.5 seconds per frame) compared to Windows (4 seconds per frame), highlighting the impact of different TensorFlow implementations.

Despite these achievements, the project faced some limitations, including challenges in acquiring diverse and representative datasets, which could affect the model's generalizability across different types of video content. The dependency on specific hardware configurations also influences real-time performance and scalability. To address these issues and further enhance the model, future research should focus on refining the model with larger and more diverse datasets, exploring advanced techniques such as ensemble learning or transfer learning, and collaborating with industry stakeholders for real-world validation.

Overall, this project underscores the transformative potential of machine learning in video artifact detection. By leveraging advanced model architectures and optimization techniques, significant progress has been made towards improving video quality assessment. The continued evolution of video technology and methodologies promises further enhancements in accuracy and efficiency, leading to better user experiences and more reliable video content.

**REFERENCES:**

[1] 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, Apr. 2004.

[2] BT.500 : Methodology for the Subjective Assessment of the Quality of Television Pictures, **https://www.itu.int/rec/R-REC-BT.500**

[3] M.-J. Chen and A. C. Bovik, "Fast Structural Similarity Index Algorithm," Journal of Real- Time Image Processing, vol. 6, no. 4, pp. 281–287, Dec. 2011.

[4] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, and V. Lukin, "On Between-coefficient Contrast Masking of DCT Basis Functions," in Proceedings of the 3 rd International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM '07), Scottsdale, Arizona, Jan. 2007.

[5] Daala codec. **https://git.xiph.org/daala.git/**

[6] T.-J. Liu, J. Y. 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.

[7] J. Y. Lin, T.-J. Liu, E. C.-H. Wu, and C.-C. J. Kuo, "A Fusion-based Video Quality Assessment (FVQA) Index," APSIPA Transactions on Signal and Information Processing, 2014.

[8] C.Cortes and V.Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[9] H. Sheikh and A. Bovik, "Image Information and Visual Quality," IEEE Transactions on Image Processing, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[10] S. Li, F. Zhang, L. Ma, and K. Ngan, "Image Quality Assessment by Separately Evaluating Detail Losses and Additive Impairments," IEEE Transactions on Multimedia, vol. 13, no. 5, pp. 935–949, Oct. 2011. S. Wolf and M. H. Pinson, "Video Quality Model for Variable Frame Delay (VQM_VFD)," U.S. Dept. Commer., Nat. Telecommun. Inf. Admin., Boulder, CO, USA, Tech. Memo TM- 11–482, Sep. 2011.

[11] Video Quality Experts Group (VQEG), "Report on the Validation of Video Quality Models for High Definition Video Content," June 2010, **http://www.vqeg.org/**