International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Improving the Efficiency of Video Transmission in Computer Networks

Murooj Khalid Ibraheem Ibraheem[1], Prof. Alexander V. Dvorkovich[2], Israa M. Abdalameer Al-khafaji[3]
[1] *Phystech School of Radio Engineering and Computer Technologies (FRKT), Department of multimedia technologies and telecommunications*
Moscow Institute of Physics and Technolohy (MIPT)
Dolgoprudny, Russia
ibragim.m@phystech.edu
[2]*Phystech School of Radio Engineering and Computer Technologies (FRKT), Department of multimedia technologies and telecommunications*
Moscow Institute of Physics and Technolohy (MIPT)
Dolgoprudny, Russia
dvork.alex@gmail.com
[3]*Institute of Information Technologies, Department of Corporate Information Systems*
RTU MIREA
Moscow, Russia
misnew6@gmail.com

Abstract— In-depth examination of current techniques for enhancing the efficiency of video transmission over digital networks is provided in this study. Due to the growing need for high-quality video content, optimizing video transmission is an important area of research. This review categorizes and in-depth examines a range of methods proposed in the literature to enhance video transmission effectiveness. ABR, DNN architecture, adaptive streaming, Quality of Service (QoS), error resilience, congestion control, video compression, and hardware acceleration for video provisioning are just a few of the cutting-edge techniques that are covered in the discussion, which ranges from the more traditional to the cutting-edge. This essay provides a methodical evaluation of the numerous tactics that are available, along with an analysis of their guiding principles, advantages, and disadvantages. The paper also offers a comparative analysis of various approaches, highlighting trends, gaps, and potential future research directions in this crucial domain, all of which help to create more efficient video compression and transmission paradigms in computer networks.

Keywords-Video Transmission ; QoS ; ABR, DNN; video compression.

I. Introduction

The exponential growth of multimedia content, especially video, has had a significant impact on how information is exchanged and consumed. However, effective video transmission over computer networks has become challenging due to the rise in demand for high-quality video content [1]. Video transmission must be done efficiently to ensure continuous streaming, less buffering, and the greatest user experience. This in-depth investigation examines a variety of strategies meant to improve the efficiency of video transmission via computer networks [2]. These methods cover a wide range of topics, including video compression, adaptive streaming, error-correcting systems, and QoS (Quality of Service) improvements. This survey provides useful insights into the progress made in improving video transmission efficiency by examining this diversified terrain, taking into account both technological and practical issues. In this study, we explore many different strategies that academics and professionals have put forth and used. We seek to provide insight on the development of video compression technology through a thorough analysis. Understanding the fundamental ideas behind these methods will help us better appreciate how they all work together to address the issues brought on by variable network circumstances and restricted resources [3]. Many methods and plans that have been created to increase video transmission effectiveness will be examined in more detail in the parts that follow in this study. We will go through the creative solutions that influence the present and future of video streaming, starting with an awareness of the fundamental problems and moving on to novel paradigms like Content Delivery Networks (CDNs) and latency reduction strategies. Due to the growing demand for effective and high-quality content delivery, the landscape of video transmission in computer networks has undergone substantial changes. Many different methods have been suggested and put into practice by researchers and professionals in order to increase the effectiveness of video broadcasting. An overview of the various tactics used in this quest is given in this literature review, highlighting the creativity and invention that have formed this dynamic field.

MOTIVATION

The desire to improve user experience, the difficulties posed by the limited network resources, and the rising demand for high-quality video content all contribute to the need to increase the efficiency of video transmission in computer networks. This section offers a glimpse into the wide range of factors that have encouraged practitioners and researchers to investigate and use different methods in this field.

Demand for high-quality video material: The demand is increasing due to the growth of digital platforms, streaming services, and social media. This demand is unprecedented [4]. Today's users demand flawless, buffer-free streaming on a variety of devices. In order to satisfy this demand, it is imperative to optimize video transmission in order to guarantee consistent and outstanding visual quality independent of network circumstances.

Limitations on available bandwidth and network congestion: As a result of the increase in data traffic, networks frequently experience congestion, which drastically lowers the quality of video transmission. There is an inherent motive to develop methods that effectively use the available resources to transmit video content without interruptions due to the limited available bandwidth, especially in cases involving mobile networks.

Network Conditions and Device Diversity: Network circumstances vary naturally, ranging from high-speed broadband to low-bandwidth connections. This phenomenon is known as device diversity. Additionally, a range of devices, from powerful desktops to resource-constrained mobile devices, can view video content. There are incentives to improve video transmission so that it can continue to be flexible to a wide range of network configurations and devices, guaranteeing a consistent user experience.

User Experience and Viewer Retention: In the digital world, user pleasure is crucial. Unsatisfactory video streaming might make users give up on the service, stop watching the video, or even switch to another platform. The goal to deliver high-quality, uninterrupted video streaming in order to keep viewers engaged and loyal is the driving force behind efficiency improvements.

Real-Time Applications and Interactive Content: Video material is necessary for real-time applications like video conferences, online gaming, and live streaming in addition to being entertainment. To keep the content's immediacy and interaction in these situations, decreasing latency is essential. Here, the goal is to make sure that video data transfer complies with the applications' need for real-time performance.

Emerging Technologies and Innovations: There are potential to increase the effectiveness of video transmission due to the quick development of networking technologies, such as the switch to 5G networks and the appearance of new transport protocols like QUIC. Researchers are driven to take advantage of these advancements in order to push the limits of video delivery performance and quality.

Economic considerations and resource optimization: Improving video transmission efficiency results in cost savings for content providers. Through reduced server load, reduced user data consumption, or energy-efficient distribution to mobile devices, efficient transmission lowers the resources needed to deliver video information.

Competitive Environment and Industry Standards: As streaming services compete more fiercely, providing great video quality becomes a differentiator. Additionally, industry standards like video compression codecs have a significant impact on how videos are sent. Innovation is driven by incentives to accept or surpass these norms.

In a nutshell maximizing resource usage, exploiting technological improvements, and fulfilling user expectations are what drive efforts to increase the effectiveness of video transmission in computer networks. In order to improve the quality and accessibility of video experiences across the digital sphere, a variety of strategies are being investigated as a result of this multifaceted goal.

CONTRIBUTIONS

The potential advantages of study "Comprehensive Survey for Improving the Efficiency of Video Transmission in Computer Networks" would provide:

Detailed Overview of Techniques: The study presumably offers a thorough analysis of the numerous methods and approaches designed to increase the effectiveness of video transmission in computer networks. It would include many different techniques, such as video compression, adaptive streaming, error correction, Quality of Service (QoS) mechanisms, content delivery networks (CDNs), peer-to-peer (P2P) architectures, network protocols, latency reduction, DNN architecture and video compression based on DNN technique and energy-efficient transmission methods.

In-Depth Explanation: A thorough discussion of the underlying concepts, methods, and algorithms for each technique may be provided in the paper. This would make it easier for readers to comprehend how these methods serve to increase the effectiveness of video transmission.

Comparative Analysis: The article might include a comparison of the advantages and disadvantages of various methodologies. The performance, applicability, and trade-offs of each approach would be assessed so that readers could choose the best method to use in a given situation.

Historical Context: It is important to comprehend how video transmission methods have changed through time. The research might shed light on the causes for these strategies' development as well as how they have changed over time.

Applications in the actual World: Useful examples and case studies could demonstrate how these strategies are utilized in the actual world. This might make it easier for readers to understand how these strategies can increase the effectiveness of video broadcasting.

Emerging Trends and Challenges: Because technology is always evolving, the paper may include new developments and difficulties related to video transmission efficiency. This could involve talking about the effects of future network technologies (like 5G), the effects of higher video resolutions (like 4K and 8K), and potential security and privacy issues.

Advice on Making Decisions: The publication could work as a reference for researchers, practitioners, and decision-makers by providing a thorough overview of the methodologies that are now accessible. The reader would learn which methods fit their objectives and needs the best.

Future Directions: The report can point up possible areas for additional study and investigation in its conclusion. This would inspire additional research in the area and inspire readers to venture into unexplored territory.

Citation Resource: Because it gathers and cites a variety of related works in one place, a thorough survey article frequently proves to be a useful tool for researchers. For readers looking for references for their own research, this would save time.

II. VIDEO COMPRESSION

When looking to improve the efficiency of video transmission via digital networks, video compression is a crucial building block. This key procedure employs sophisticated algorithms to shrink video data while maintaining its perceived quality. Compression helps smooth out streaming and maximize

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

playing by reducing the amount of data needed to depict a video sequence in light of constrained network bandwidth and storage capacity [5].

The core premise of video compression is to take advantage of redundancy both inside individual video frames and between frames. Compressing individual frames is the goal of intra-frame compression, also known as spatial compression, which does so by detecting and encoding spatial redundancy within the frame. Pixel data is transformed into frequency-domain coefficients using methods like the Discrete Cosine Transform (DCT) and the Discrete Wavelet Transform (DWT). The modified coefficients are then quantized and encoded using entropy coding methods like Huffman coding or arithmetic coding to further reduce data size [6].

Temporal compression, or inter-frame compression, takes advantage of the commonalities shared by successive frames in a video sequence. By using previously encoded frames as a basis for making predictions about the current frame, only the differences (residuals) between the prediction and the actual frame need to be stored in the code. This method streamlines the video-representation process by eliminating unnecessary repetition between frames. Effective inter-frame compression requires the use of methods like Motion Estimation and Motion Compensation.

H.264 (AVC), H.265 (HEVC), and AV1 are only some of the most recent video compression technologies that have raised the bar for compression efficiency. Intra-frame and inter-frame compression, as well as cutting-edge methods like quantization, entropy coding, and context modeling, are all incorporated into these guidelines [7]. For example, H.265 expands on H.264 by including new, more sophisticated coding methods, resulting in increased compression efficiency without a corresponding drop in picture quality.

Video compression is crucial because it affects the entire video transmission environment, not just file sizes. Compressed videos have many benefits, including easier creation and dissemination, smaller file sizes, and faster transfer rates. This has implications beyond only media platforms, including video conferencing, monitoring, and even medical imaging. Keeping up with the increasing data requirements of higher-resolution videos is essential for providing uninterrupted, high-quality video streaming that enhances our digital experiences and lives.

Video compression has become a cornerstone of the modern multimedia landscape. Video compression was developed to help with the urgent need of improving data transfer and storage efficiency. The importance of compression in optimizing bandwidth in an era defined by an insatiable desire for video material cannot be overstated. By drastically lowering the amount of data needed to send videos, it enables smooth streaming experiences despite constrained network bandwidth. By reducing the frequency and severity of buffering and latency issues, this not only improves user satisfaction but also creates a more immersive watching experience.

Video compression has repercussions across various systems and gadgets. Videos that have been compressed offer a flexible alternative, as they can be played on every screen size, from smartphones to televisions [8]. Users may easily access material on any device they want thanks to this responsive design. The intersection of video compression and storage efficiency is a win-win for producers and viewers alike. Videos can be compressed to lower their storage footprint, which in turn saves money for makers and frees up space on users' devices.

A. Adaptive Bitrate Streaming (ABR)

Due to the rising need for high-quality video content distribution over the internet, the efficiency of video transmission in computer networks is of paramount importance. Adaptive Bitrate Streaming (ABR) [9], rate adaptation algorithms like MPC [9], BOLA [10] and Pensieve [10], and related methods are all included in this overview. Due to the prevalence of online video, it is crucial that video transmission within data networks be optimized. This research paper examines the various methods and approaches that have been created to speed up the process of video distribution.

Table 1: Advantage and disadvantages of Adaptive Bitrate Streaming (ABR).

Advantages of ABR	Disadvantages of ABR
Improved-quality viewing	Implementation complexity
Reduced buffering	Latency from frequent bitrate changes
Efficient bandwidth use	Quality oscillations
Smooth playback	Noticeable quality shifts
Adaptation to changing network conditions	Increased network communication overhead

To maximize video transmission quality, ABR dynamically adjusts video quality in response to changes in the underlying network. The ABR algorithms allow for seamless playback with minimal buffering and high quality video output [9]. Popular ABR algorithms consist of:

- **BOLA Algorithms**

BOLA is an adaptive bitrate streaming technique that reduces re-buffering while maintaining high video streaming quality. It functions by automatically adjusting the video bitrate to the current buffer size and network bandwidth. The goal of BOLA is to provide the finest possible video quality while yet providing a fluid viewing experience.

Salient Features:

The primary goal of BOLA is to make an educated guess as to the amount of video data currently being buffered for playback. The system may intelligently modify the video bitrate by keeping an eye on the buffer's fill level.

BOLA calculates the available network throughput by observing how quickly video segments download. The ideal bitrate that can be maintained without triggering excessive buffering is calculated using this data.

BOLA's primary goal is to avoid buffer underflow (when the buffer is exhausted before the next segment arrives) and buffer overflow (when the buffer is excessively full) by choosing the bitrate that provides the highest possible video quality. The decision is based on the current buffer occupancy and the network throughput.

- **Model Predictive Control (MPC)**

By employing predictive models of the network and buffer dynamics, Model Predictive Control (MPC) is an advanced control approach used in adaptive bitrate streaming to achieve optimal video quality and buffer occupancy. In contrast to other ABR algorithms [10], MPC considers the big picture by basing its judgments on predictions across a number of segments in the future rather than just one.

Salient Features:

Model Prediction: MPC uses models that can anticipate the future state of the network (such as throughput changes) and the buffer (how quickly it fills and empties). These simulations shed light on the system's expected future development.

With quality and buffering habits in mind, MPC strives to improve the entire user experience. It does this by choosing the sequence of bitrates that maximizes a selected objective, typically balancing quality, stability, and rebuffering, based on predictions of the effects of those decisions over a limited time horizon.

Dynamic Bitrate Selection: MPC computes bitrate sequences for numerous future segments as opposed to adjusting the bitrate on a per-segment basis. This preventative measure lessens the likelihood of rebuffering due to sudden quality drops and rapid bitrate adjustments.

- **Pensieve algorithm for use with ABR**

Through adaptively adjusting the bitrate based on network conditions, Pensieve, an ABR algorithm powered by deep reinforcement learning (DRL), improves video streaming quality and decreases buffering events. Pensieve is data-driven rather than relying on predefined rules like other compression algorithms.

Salient Features:

Pensieve uses DRL, a deep learning technique, to discover the best bitrate adaption policy via trial-and-error interactions with the surrounding environment (the state of the network). Because of this, it can use the patterns it detects in the data to guide its judgments.

Pensieve uses a reward algorithm that takes into account the compromise between video quality and buffering to direct its learning process. The algorithm's goal is to optimize the long-term benefits, which include both the quality of the current playback and the reliability of playback in the future.

Pensieve takes into account things like network performance, buffer occupancy, and the current video bitrate to accurately depict the status of the environment. The DRL model uses this data as an input.

Table 2: Algorithms of Adaptive-Bitrate Streaming (ABR).

ABR Algorithm	**Description**	**Key Features**
BOLA (Buffer-Based Optimal Rate Adaptation)	Estimates the optimal bitrate to prevent buffer underflow and overflow, improving quality and reducing rebuffering	- Buffer occupancy estimation - Throughput estimation - Rate selection for optimal quality
MPC (Model Predictive Control)	Utilizes predictive models of network and buffer dynamics to make adaptive bitrate decisions, optimizing both quality and buffering behavior	- Predictive modeling of network and buffer states - Minimizes rebuffering and maximizes quality - Long-term and short-term reward optimization
Pensieve	Applies deep reinforcement learning to adapt video quality based on network conditions, aiming for optimal quality and minimal buffering	- Reinforcement learning framework - Consideration of long-term and short-term rewards - Adaptive quality adjustment based on history

III. VIDEO CODING STANDARDS

Essential to the science of video compression, video coding standards define how digital video should be encoded, transmitted, and decoded. These guidelines enable compatibility across a wide range of hardware and software, allowing for effective data compression without compromising on video quality [11]. The ability to compress video while maintaining its perceived quality is crucial for a wide variety of uses, including video streaming, video conferencing, broadcasting, and more.

There have been numerous improvements in video compression technologies that have led to new video coding standards. Let us take a closer look at a few of the most used video encoding formats today:

a) H.261 (1988) was the world's first standardized format for compressing video across long distances. It was developed by the International Telecommunication Union's Telecommunication Standardization Sector (ITU-T). Motion compensation, discrete cosine transform (DCT), and variable-length coding were among the strategies offered. These would reduce the amount of data that needn't be sent during video broadcasts.

b) In 1992, the Moving Picture Experts Group (MPEG) created a video compression standard called MPEG-1 with the goal of producing high-quality video CDs. Using DCT, we were able to compress the data, and the bit rate could be adjusted to work with various media formats.

c) MPEG-2 (1995), a refined variant of MPEG-1, is suitable for use in both television transmissions and DVDs. It supported interlaced video, scalable video coding, and featured profiles tailored to certain uses, such as the "main" profile for digital television and the "4:2:2" profile for studio use.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

d) MPEG-4 (1998) marked a shift away from video-centric to multimedia-centric standards. In order to achieve better scene-based compression, it was an early adopter of object-based coding. Video, audio, and 3D graphics editing facilities are all incorporated within MPEG-4.
e) The introduction of H.264 (MPEG-4 Part 10 AVC, Advanced Video Coding) [12] in 2003 was a major step forward in the efficiency of video compression. Entropy coding, adaptive motion compensation, intra-frame prediction, and variable-size blocks are only some of the cutting-edge methods that were used. H.264 is widely used for video streaming and broadcasting because of its high compression ratios and negligible quality loss.
f) H.265 HEVC (High Efficiency Video Coding, 2013) [13] is intended to improve compression efficiency even farther than its forerunner, H.264. There are now larger blocks, enhanced motion prediction, and sophisticated entropy coding. With HEVC, the bandwidth needed to stream a video is cut by nearly half compared to H.264 while still retaining a high degree of quality.
g) AV1 (2018) [11], an open and royalty-free video coding standard, was developed by the Alliance for Open Media (AOMedia). To attain its great compression efficiency, it makes use of cutting-edge methods developed independently by many contributors. Unlike HEVC and H.264, AV1 doesn't require expensive licenses to use, making it a viable alternative to those standards.
h) H.266 VVC (Versatile Video Coding, 2020) [12] is designed to be an improvement on HEVC's already impressive compression ratios. It uses a number of methods, including enhanced intra prediction, larger block sizes, and more sophisticated motion compensation. VVC is optimized for high-definition and fully immersive video.

Collectively, these video coding standards have made significant strides in the compression, distribution, and reception of digital video. Because they allow high-quality video content to be delivered over different networks and devices, they have had a significant impact on industries like broadcasting, entertainment, communications, and surveillance [12]. Video coding standards are anticipated to evolve and improve in the future to accommodate new uses and higher-quality content.

Table 3: Each video coding standard with their notable features.

Standard	**Release Year**	**Notable Feature**
H.261	1988	Motion compensation, DCT, variable-length coding
MPEG-1	1992	CD-quality video, DCT, variable bit rate
MPEG-2	1995	Broadcast and DVD-quality, interlacing support
MPEG-4	1998	Object-based coding, multimedia framework
H.264 (AVC)	2003	Advanced motion compensation, intra prediction, variable block sizes
H.265 (HEVC)	2013	Larger block sizes, improved intra and motion prediction
AV1	2018	Open, royalty-free, high compression efficiency
H.266 (VVC)	2020	Advanced motion compensation, improved intra prediction

These standards take advantage of repetitions in both time and space within a video.

Using the fact that successive frames in a video sequence generally share commonalities in terms of content and motion, *inter-frame prediction* is able to make accurate predictions about the frames to come. Inter-frame prediction uses a reference frame and the differences between it and the current frame (predicted frame) as its encoding basis. The prediction residual is a common term for this discrepancy. It is possible to improve the efficiency of the encoding process by sending simply the prediction residual. High-frequency information (details) that may not be reliably anticipated is often found in the prediction residual and, as a result, must be encoded more precisely.

Inter-frame prediction and motion compensation are related techniques. Predicting a frame from one or more reference frames entails predicting the mobility between frames. The calculated motion vector is then used to transform the reference frame(s) into a coherent coordinate system with the present frame. This method takes the mobility of objects into account, which helps eliminate or greatly minimize the temporal redundancy. The prediction residuals are typically communicated together with the motion vectors, which describe the translation of blocks or macroblocks between frames.

Entropy coding is a data-representation method that makes use of codes of varying lengths to cut down on wasted space. It takes advantage of the fact that more common symbols need fewer digits in their codes, whereas less common symbols can have more digits. Huffman coding and arithmetic coding are two methods that can accomplish this goal. Entropy coding decreases the total bit rate by shortening the average length of the coded data by utilizing shorter codes for common patterns.

When it comes to maximizing the compression and quality of video data, the trifecta of intra-frame prediction, transform coding, and quantization is crucial. *Intra-frame prediction* is a method for independently compressing each frame by using the spatial redundancy present in each frame to foretell pixel values based on nearby pixels or blocks. It includes a variety of prediction modes, including mode-based prediction, directional prediction, planar prediction, and DC prediction, which judiciously selects the most effective prediction technique for specific regions of the frame.

Transform coding converts spatial information into the frequency domain, which makes it more suitable for compression. Applying well-known mathematical transformations to video blocks, such as the Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT), highlights crucial frequency components while suppressing less crucial ones. Quantization follows the transformation and converts continuous data to a discrete set. Quantization allow to control the precision of data representation in the frequency domain.

An important factor that balances video quality and compression effectiveness is the *quantization* step size. Smaller steps offer better quality at the cost of higher bit rates, while larger steps produce higher compression but lower quality.

B. Caching and Content Delivery Networks (CDNs) in video compression

When it comes to delivering digital content, speed and efficiency are of the utmost importance. Users expect instant playback of videos and other media, without buffering or pauses. Specifically in the case of video compression, this is where caching and Content Delivery Networks (CDNs) come into play [13, 14]. Together, these innovations improve the user experience and lessen the load on networks as video content is distributed more efficiently.

To reduce the amount of time spent retrieving data from its original source, caching includes storing data that is often accessed in a location closer to the user or application. Caching plays an important part in video compression, helping to speed up load times and reduce bandwidth consumption.

Caching works on numerous tiers for video content:

Web page elements, including embedded movies, can be cached in a user's browser. When a user returns to a previously visited website, the browser just needs to access the local cache rather than the server, thus reducing load times. Caching in a Content Management System (CMS) is a technology used by the CMS to save time by reusing previously created web pages [15]. Video thumbnails and video player components can be pre-rendered in this way, reducing load time for the entire page.

In order to store copies of frequently viewed video content, businesses may use proxy servers or reverse proxies. By delivering content from their own caches, these proxy servers assist lighten the burden on the primary server. Networks of servers strategically situated in different parts of the world work together to deliver content to users quickly and reliably. Their major function is to shorten the time it takes for digital content, such as videos, to reach the end user by reducing the distance the content must travel [16].

Video compression and distribution using CDNs has many benefits:

- Since CDNs typically have numerous server locations, they can direct user requests to the one that is geographically closest to them, thereby reducing the amount of time it takes for data to travel. As a result, users will experience much less delay when watching videos online.
- Distributing incoming traffic among multiple servers to avoid overloading any one of them is a key function of content delivery networks (CDNs). This makes for more reliable video streaming, especially at peak usage times.
- By storing copies of frequently accessed videos on their own servers, content delivery networks (CDNs) can quickly send them to users. This reduces the stress on the primary server and speeds up the distribution to end users.
- CDNs typically allow adaptive bitrate streaming, which adapts the video stream's quality to the user's connection speed and latency. This enhances the viewing experience by reducing the need for buffering and facilitating uninterrupted playback.
- CDNs not only distribute content, but also provide failsafes. Having a backup server ready to take over content delivery in the event of a failed server improves availability.
- Effective video compression methods like H.264 (AVC), H.265 (HEVC), and newer codecs like AV1 are frequently used in tandem with CDNs. These codecs compress videos without sacrificing too much quality. Then, content delivery networks (CDNs) distribute these compressed video files to end users in a way that minimizes data transmission and load times.

Table 4: Advantages and disadvantages of caching and content delivery network.

Aspect	Advantages	Disadvantages
Caching	Faster load times, reduced server load	Stale content, limited scalability
	Improved user experience, content control	Inconsistent UI, cache invalidation
	Cost efficiency, adoption ease	Maintenance overhead, update control
CDN	Reduced latency, global distribution	Setup complexity, potential cost
	Scalability, reliable content delivery	Performance dependency on CDN

In an effort to improve the dissemination of digital content throughout the internet, numerous Content Delivery Network (CDN) architectures have been suggested and analyzed [17]. When it comes to content distribution, scalability, efficiency, and user experience, each of these architectures takes a somewhat different approach. The hierarchical, peer-assisted, and hybrid models are three of the most well-known CDN architectures.

a) Hierarchical CDN

The content delivery network (CDN) uses a hierarchical design to distribute data across numerous levels of servers. Origin servers, intermediate (regional) caches, and edge servers (sometimes called PoPs - Points of Presence) are the usual components of this model's three tiers.

Advantages:

- The hierarchy facilitates content replication from a centralized source to regional caches, which improves the efficiency of data dissemination.
- Edge servers located closer to end users reduce latency by shortening the distance data must travel.
- The flexibility to add more levels as demand increases allows for scalability.

Disadvantages:

- Control hubs: too much reliance on one location can cause problems down the line.
- The complexity of upkeep increases when there are several layers to manage.
- Congestion in the network occurs when a large number of users attempt to access a single resource.

b) Peer-assisted CDN

In a peer-assisted CDN, the standard server-based method of delivering data is supplemented with P2P technology. In this arrangement, users' devices (peers) help distribute content by exchanging copies of it between themselves.
Advantages:

- Peer-to-peer computing allows for scalability, as it allows for increased traffic capacity without requiring a corresponding increase in server capacity.
- The redundancy provided by peer contributions increases content accessibility and robustness.
- Saving money is possible because of the decreased server load.

Disadvantages:

- Inconsistent peer connections have the potential to degrade content integrity.
- Open peers may put data at risk of being viewed by the wrong people or tampered with.
- Copyright and licensing issues may arise with P2P file sharing.

c) Hybrid CDN

In a hybrid CDN, the hierarchical and peer-assisted designs are both used. They use existing server infrastructure and combine it with peer-assisted components to distribute data more quickly and effectively.
Advantages:

- Improved Efficiency: Draws from the best features of both models to increase content accessibility and distribution velocity.
- Adaptability: Can swiftly switch between server-based and P2P delivery to meet the needs of the network and the users.

Disadvantages:

- Implementation difficulty bringing together distinct architectures demands meticulous planning and coordination.
- It can be difficult to strike a balance between the demands of the server and those of the users.

These CDN architectures were designed to solve unique problems associated with content distribution. Considerations including network topology, user demographics, content popularity, and desired trade-offs between speed, scalability, and cost all play a role in determining the best architecture to deploy. Content delivery over the internet is always being refined by testing out new models and improvements as technology advances.

C. QoS for Video Compression on a Network

By efficiently assigning and managing network resources, QoS networks provide a consistent and dependable experience for certain types of traffic, such as video streaming, over the network [18]. To keep video quality, reduce transmission delays, and avoid interruptions, QoS methods are essential in the context of video compression. In video compression, several essential QoS techniques include:

a) Traffic Shaping: Smoothing out network traffic and avoiding congestion are two primary goals of traffic shaping. In order to guarantee constant video streaming quality, video traffic might be allotted a specific amount of the available bandwidth.

b) Packet Prioritization: Prioritizing packets allows for video data to be sent at a faster rate than other forms of data. This prevents video from being interrupted by buffering or lag and guarantees its timely delivery.

When additional video streams are added to a network, admission control checks to see if the system has enough resources to keep the quality of service (QoS) where it needs to be. This avoids saturating the network and lowering the quality of currently streaming videos.
Video Compression Traffic Management Methods:
The most efficient use of network resources for video distribution is achieved through the application of traffic control techniques [19]. These methods facilitate the network-friendly delivery of video streams. Here are some applicable methods:

a) Multicast: When opposed to sending many separate unicast streams, the network burden is significantly reduced when using multicast to send a single video stream to multiple recipients at once. This is especially helpful when numerous people are watching the same video at once.

b) Content-Aware Routing: When determining the most efficient network path for transporting a video stream, "content-aware routing" takes into account the type of video being transmitted. One method for doing so is through the use of dynamic route selection depending on factors like bandwidth and latency.

Table 5: Network management and QOS for the video quality and experience.

QoS/Traffic Management Technique	**Description**

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

Traffic Shaping	Regulates data flow to prevent congestion and ensure steady video streaming
Packet Prioritization	Assigns higher priority to video packets to reduce latency and buffering
Multicast	Efficiently delivers a single video stream to multiple recipients, reducing network load
Content-Aware Routing	Selects optimal network paths based on video content characteristics for better performance

D. Error Resilience and Error Correction

In video compression and transmission, error resilience and error correction are essential ideas, especially when working with unreliable networks susceptible to packet loss or mistakes. By minimizing the consequences of mistakes, maintaining visual quality, and minimizing the need for retransmission or re-buffering, these strategies aim to increase the efficiency of video transmission [20].

Error Resilience:

Error resilience describes a video compression system's capacity to survive errors without significantly lowering the quality of the decoded video. To ensure that errors have a minimal impact, this includes integrating specific tactics and processes into the compression process. Here are some popular methods for preventing errors:

a) Slice Structuring: To enable partial decoding, video frames are split into smaller slices. Only a section of the frame is impacted when a slice is lost due to a mistake, lessening the influence on overall quality.
b) Intra Refresh: Error recovery is aided by intermittently inserting completely encoded intra-coded frames (I-frames) within the video stream. The decoder can still reconstruct the movie even if some frames are lost since these frames serve as reference points for succeeding frames.
c) Resilient Coding: A few video coding standards have resilient coding options. These modes use less error-sensitive coding methods to give resilience precedence over compression effectiveness.

Error Correction:

To help the receiver identify and fix mistakes, error correction techniques entail adding redundant information to the video feed. Forward Error Correction (FEC), a frequently used error correction method, pre-encodes the video stream with redundant data before transmission. Even if some packets are lost or distorted while being transmitted, FEC enables the recipient to reassemble the original contents. FEC can take a number of forms:

a) Low-Density Parity Check (LDPC) Codes:
A class of linear error-correcting codes known as LDPC codes has grown in prominence in a number of communication systems, including video compression. They were initially proposed by Robert G. Gallager in the 1960s, but have recently gained popularity because of their exceptional error-correcting capabilities.
b) Polar Codes:
Polar codes, introduced by Erdal Arikan in 2008, represent another class of FEC codes that have gained recognition for their capacity-achieving properties. They are known for their simplicity and effectiveness in error correction.
c) Turbo Codes:
A subset of FEC codes known as turbo codes have been employed to compress video, especially in earlier technologies like 3G and 4G. Even though they might not be as common in more recent standards, they nonetheless perform well in terms of error correction and are helpful in older systems.
d) Raptor codes:
These are a sort of fountain code created specifically for effective erasure correction. They are utilized in applications like streaming video over shaky networks, when packet loss is frequent. Raptor codes are reliable and effectively recover lost packets.

Retransmission and Error Concealment.

Retransmissions, however, could result in extra network overhead and delays. Error concealing algorithms are used to reduce these issues. These algorithms make an effort to reconstruct damaged or missing portions of the video using nearby, error-free data.

a) Temporal Interpolation: To mask a lost frame, the decoder replicates the prior or next frame. Although there may be temporal artefacts, this is frequently better than freezing or skipping frames.
b) Spatial Interpolation: The decoder takes information from nearby blocks to fill in any gaps in a frame via spatial interpolation. Small portions of a frame can be successfully recovered using this technique.

Table 6: Relevant technique of error processing in transmission of video compression modules.

Concept	Description
Error Resilience	Techniques to withstand errors without major quality loss
Error Correction	Adding redundancy for error detection and correction
Retransmission	Requesting lost data again
Error Concealment	Algorithms to fill in missing parts using nearby information

E. Peer-to-Peer (P2P) Video Streaming

Peer-to-peer (P2P) video streaming uses the resources of individual network users, also referred to as peers, to deliver video content over a network. P2P streaming spreads the load and resources across several peers in a decentralized way, as opposed to conventional client-server models, where a central server is in charge of providing content to clients [21].

Important Features of P2P Video Streaming:

a) Decentralization: Peer-to-peer video streaming networks reduce the dependency on a single central server by sharing the workload of content delivery among peers. Each peer contributes to the network's upload and download bandwidth.

b) Scalability: P2P architecture scales more effectively than traditional client-server systems because it can handle larger networks when more peers are added. This makes it possible to manage enormous audiences without taxing the main servers.

c) Redundancy and Reliability: Since content is copied among several peers, P2P systems can be more reliable and resilient to failures. The content can still be retrieved from other available peers even if one peer goes offline.

d) Effective Bandwidth Use: In P2P networks, peers cooperate to distribute content to other peers by sharing their available bandwidth. By doing so, you may optimize bandwidth use and lessen the overall load on the network infrastructure.

e) Accessibility of Content: By obtaining data from the closest and fastest peers, P2P networks can enable faster content delivery, reduce latency, and enhance user experience.

f) Challenges: The early seeding of content (when few peers have the entire video), maintaining a stable network as peers join and leave, and addressing free rider issues — where some peers watch content without contributing resources — can all be difficulties for P2P streaming.

Components of P2P Video Streaming:

a) Overlay Construction: P2P networks use overlay topologies to connect peers efficiently. Overlays can be structured (like trees or meshes) or unstructured (random connections). Efficient overlay construction is vital for optimizing content distribution.

b) Peer Selection Algorithms: Algorithms determine which peers to connect to for content delivery. These algorithms aim to select peers that have the desired content, low latency, and good bandwidth availability.

c) Incentive Mechanisms: Since P2P relies on peers' voluntary contribution of resources, incentive mechanisms may be used to motivate peers to contribute upload bandwidth and storage. Examples include token rewards or content prioritization.

d) Buffer Management: P2P streaming clients often use buffer management techniques to ensure a continuous playback experience. Buffers can help mitigate network delays and interruptions.

Table 7: Advantages and disadvantages of P2P video streaming.

Advantages	Disadvantages
Scalability and reduced server load	Initial seeding challenges
Efficient bandwidth utilization	Network instability due to peer dynamics
Redundancy and reliability	Free rider problem (uneven resource contribution)
Faster content delivery	Quality of service varies with peers' resources
Resilience against single points of failure	Complexity in managing the network

P2P video streaming effectively delivers video material by utilizing the distributed resources of peers. Although it has advantages like scalability and effective bandwidth usage, difficulties like early seeding and managing peer dynamics must be overcome for best performance.

F. <u>Network Protocol Enhancements</u>

For current applications, especially those requiring video transmission, to satisfy the ever-increasing demands, network protocol improvements have been a critical field of research and development [22]. Improvements have been made to protocols like Real-Time Transport Protocol (RTP) and the upcoming Internet Protocol version 6 (IPv6) to facilitate more effective and dependable video distribution. Among the notable improvements are:

- *RTP Enhancements:*

RTP is a protocol used to send audio and video data in real time across IP networks. Several enhancements have been made to increase the performance of video distribution, including:

a) RTP now allows adaptive bitrate streaming, in which the video stream's quality is constantly modified in accordance with the amount of network bandwidth that is available. This guarantees more fluid playback and minimizes buffering.

b) Forward Error Correction (FEC): Since RTP now supports FEC, lost or corrupted packets can be recovered by the receiver without having to be retransmitted. By doing this, video quality across erratic networks is improved.

c) Reduced Latency: RTP improvements have been made with the goal of reducing end-to-end latency, which makes it better suited for real-time applications like video conferencing and live streaming.

- *IPv6 Improvements:*

IPv6 is the Internet Protocol's sixth generation, which was created to replace IPv4 and fix its shortcomings. Features of IPv6 are advantageous for video transmission:

a) Greater Address Space: IPv6 offers a much greater address space, which is crucial for supporting the growing number of devices, particularly those used for communication and video streaming.

b) Support for Quality of Service (QoS): IPv6 includes QoS features that enable network administrators to give video traffic top priority, resulting in reliable and high-quality video delivery.

c) Improved Security: IPv6 comes with security measures like IPsec that can shield video streams from tampering and unauthorized access.

- *Network Coding:*

By enabling intermediary nodes to mix numerous packets before forwarding them, network coding improves the efficiency of data transmission. This can increase the video streams' resilience to packet loss and network congestion during transmission.

- *Adaptive Congestion Control:*

Congestion control algorithms have been improved to better respond to changing network conditions. These adaptive algorithms assist preserve excellent video quality even in busy networks. Video traffic frequently requires different handling than conventional data traffic.

- *Cross-Layer Optimizations:*

To enhance video delivery, cross-layer optimizations entail collaboration between several network protocol stack layers. For instance, the network layer's routing decisions can be improved by using information from the application layer (such as video codec needs).

Table 8: Network protocol enhancements allowing for more reliable, effective, and smooth video transmission over IP networks.

Protocol	Enhancements
Real-Time Transport Protocol (RTP)	Adaptive bitrate control, Forward Error Correction (FEC), Reduced latency
Internet Protocol version 6 (IPv6)	Larger address space, Quality of Service (QoS) support, Improved security
Network Coding	Enhanced data transmission efficiency
Adaptive Congestion Control	Adaptation to varying network conditions
Cross-Layer Optimizations	Improved coordination between protocol layers

G. Packet prioritization in video transmission

A method for ensuring the best transmission of video data over networks with variable degrees of congestion and quality is packet prioritization in video compression. Typically, video data is sent across networks in the form of packets, which are compact data units. In situations like real-time streaming, video conferencing, or online gaming where network resources are constrained or conditions change, prioritizing these packets becomes essential [23].

By ensuring that crucial video packets get at their destinations without delay or loss, packet prioritization primarily aims to maintain the quality of video playback even in less-than-ideal network conditions. This is accomplished by giving packets varying degrees of priority dependent on how crucial they are to the overall video viewing experience.

The operation of packet prioritization in video compression is as follows:

- *Packetization:*

Before transmission, video data is divided up into smaller units called packets. A piece of the video frame is included in each packet, together with headers that describe its place in the frame sequence, a time stamp, and other pertinent metadata.

- *Classification of Packets:*

As packets are set up for transmission, they are categorized according to their importance. There are typically three primary types of packets used in video compression:

a) I-Frames (Intra-frames): these are crucial frames that serve as references for decoding succeeding frames because they are entire information frames. Despite being less compressible, they are essential for preserving video quality. Because the loss or delay of an I-frame could dramatically lower the total video quality, I-frames are given the greatest priority.

b) P-Frames (Predictive frames): For reconstruction, these frames rely on earlier I- and/or P-frames. They can be compressed more effectively and contain changes from the preceding frame. Moderate importance is given to P-frames.

c) B-Frames (Bidirectionally predictive frames): These frames require both the previous and subsequent frames for reconstruction. Although they provide the greatest compression, they are less important for preserving quality. B-frames are prioritized lower.

Table 9: Packets with the level of priority to maintain the quality of video playback even in less-than-ideal network conditions.

Packet Type	Description	Priority Level
I-Frames	Key frames with complete information	High
P-Frames	Frames dependent on previous I/P-Frames	Medium
B-Frames	Frames dependent on surrounding frames	Low

The *Priority Level* in this table shows the degree of importance given to each packet type during transmission. Due to their crucial role in preserving video quality, I-Frames are given priority over P-Frames and B-Frames. This prioritization guarantees the quick delivery of vital video data, resulting in a more fluid and high-quality video playback experience.

H. Hardware Acceleration

To make video storage, transmission, and playback easier, video compression decreases the size of the video data. It's a time-consuming, computationally demanding procedure, especially for high-resolution videos [24]. The term "hardware acceleration" refers to the offloading of some of the computing processes associated with video compression using specialized hardware elements, such as GPUs and FPGAs. This greatly accelerates the encoding and decoding procedures, enabling the real-time handling of greater resolutions and frame rates.

GPUs (Graphics Processing Units): Due to their parallel processing capabilities, GPUs are a powerful tool for video compression. They excel at repetitive calculation-based

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 25 July 2023 **Revised:** 12 September 2023 **Accepted:** 30 September 2023*

activities, which are typical of compression algorithms. GPUs can simultaneously handle several data streams by dividing jobs over multiple cores, speeding up the compression process as a whole. When compared to employing simply conventional central processing units (CPUs), this results in quicker encoding and decoding times [25].
FPGAs (Field-Programmable Gate Arrays) are hardware elements that can be programmed to carry out particular functions. Contrary to CPUs or GPUs, which are more general-purpose, FPGAs can be configured to efficiently carry out certain algorithms. They are therefore ideal for jobs involving video compression. FPGAs may achieve extremely low latency and high throughput by integrating specialized logic circuits for different compression methods, which makes them perfect for real-time video processing applications.
Advantages of Hardware acceleration:

- Speed: Real-time processing of high-definition videos is made possible by hardware acceleration, which dramatically decreases the time needed for video compression.
- Efficiency: When compared to conventional CPUs, dedicated hardware components use less power and produce less heat when performing specialized tasks.
- Scalability: By adding more specialized hardware components, hardware acceleration enables systems to accommodate growing workloads.
- Applications that require real-time visual processing, such as video conferencing, live streaming, and gaming, require hardware acceleration.
- Higher Resolutions: With hardware acceleration, high-resolution videos may be compressed and decompressed quickly and without sacrificing quality.

Disadvantages of Hardware acceleration:

- Cost: Integrating specialized hardware into systems can be pricey.
- Complexity: Developing algorithms for hardware acceleration can be more difficult than creating software in the traditional sense.
- Limited Flexibility: Compared to software-based solutions, hardware is less responsive to changes in compression algorithms once it has been built and put into place.

Table 10: Comparison of Hardware Acceleration Components (GPU vs. FPGA).

Aspects	GPUs	FPGAs
Architecture	Parallel processing with multiple cores	Customizable logic circuits
Programming Flexibility	Software-based programming	Hardware-based programming
Task Specialization	Broad range of tasks	Tailored to specific algorithms
Processing Efficiency	High throughput, suitable for parallel tasks	Extremely low latency
Power Consumption	Relatively higher	Relatively lower
Adaptability	Adaptable to new algorithms via software updates	Requires hardware modifications
Cost	Moderate to high	Moderate to high
Use Cases	Real-time graphics rendering, general-purpose computing	Real-time video processing

I. Deep Learning based compression

To optimize and refine video compression methods, researchers have turned to deep learning-based approaches like neural networks. In order to minimize the quantity of video data while keeping the acceptable level of visual quality, conventional video compression technologies, such as those based on standards like H.264 or H.265, have been widely utilized. These approaches, however, frequently necessitate sophisticated handmade algorithms that may not make optimal use of neural networks [26].
By teaching neural networks to learn and generate compressed representations of video frames, deep learning-based video compression hopes to improve compression efficiency according to [27, 28]. There are numerous essential stages to this procedure:

- Data Preparation: In order to train the neural network, a huge dataset of high-quality video sequences must be acquired. In most cases, the dataset will include both uncompressed and compressed copies of a video's frames.
- Architecture Design: Designing the architecture around a neural network that can be trained to compress data is done. To capture spatial and temporal dependencies between video frames, this architecture may use convolutional and recurrent layers.
- Loss Function: The quality of the compressed video frames is measured against the original frames using a loss function. The neural network learns with the help of this loss function.
- Training: The dataset is then used to train the neural network. The network is trained to transform the original video frames into more efficient representations. Network parameters are tuned during training to achieve optimal performance relative to the specified loss function.
- Inference: The trained network can be applied to the task of compressing fresh video clips. Video frames are fed into the network, and compressed representations are output.
- Decoding: On the receiving end, a similar neural network decodes the compressed representations. The goal of this decoding procedure is to recreate video frames as faithfully as possible to the source material.

Since deep learning-based video compression allows neural networks to learn complex patterns and representations contained in video data, it has the potential to outperform conventional compression techniques [29]. Better compression efficiency while preserving or improving video quality is possible thanks to the network's capacity to adapt to various video formats and content characteristics.

Table 11: Compression using the traditional technique and DNN technique.

Aspects	Traditional Technique	DNN Technique
How it Works	Uses fixed rules for compression	Learns from data using DNN
Learning Approach	Uses preset methods	Adapts and improves with experience
Compression Efficiency	Good but limited by fixed methods	Potential for better compression efficiency
Quality at Low Bitrates	Quality can drop	Better quality at low bitrates
Complexity Handling	Struggles with complex framing scenes	Handles complexity and diverse content
Real-time Compression	Challenging in real-time	Can achieve real-time with hardware
Flexibility	Less adaptable	Adapts to different data and changes
Research Needed	Stable, less research	Ongoing research needed
Deployment	Widely used	Evolving technology

Video compression with deep learning relies heavily on the architecture of deep neural networks. Various architectures have been investigated, such as:

a) Autoencoders are networks that automatically learn how to compress and decompress data. Both Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) have been modified for video compression.
b) To detect temporal dependencies in videos, convolutional layers are combined with LSTM (Long Short-Term Memory) or GRU (Gated Recurrent Unit) layers in ConvLSTM and ConvGRU networks.
c) Networks based on transforms: VQ-VAE-2 uses vector quantization to describe video frames, which increases the efficiency of the compression process.

Metrics for Evaluating Video Compression Quality: It is crucial to evaluate the quality of compressed videos. Some typical measures are:

a) PSNR (Peak Signal-to-Noise Ratio) compares the original and compressed versions of a video to see how much quality was lost during compression.
b) SSIM (Structural Similarity Index) compares compressed and uncompressed frames' structural similarity.
c) Multi-Scale SSIM (MS-SSIM) is an improvement on SSIM that takes into account more than one image scale at once.
d) Video Multimethod Assessment Fusion (VMAF) is a perceptual quality metric that considers a wide range of elements, such as brightness, contrast, and motion.

Effect on High-Quality Video Transmission:
Multiple aspects of effective video transmission are influenced by deep learning-based video compression:

a) Reduced Bandwidth: Better compression ratios make it possible to send high-quality videos even when available bandwidth is limited.
b) Real-time Streaming: Hardware-accelerated deep learning compression paves the way for low-latency, real-time streaming.
c) Interactive Applications: Video games and virtual reality are two examples of interactive applications that benefit from efficient compression.
d) Remote Communication: Telemedicine and remote meetings both benefit from the ease with which high-quality video may be broadcast over long distances.
e) Storage Efficiency: Video file sizes can be decreased, which improves both storage and content transmission.

CONCLUSION

This extensive review has covered a wide range of topics related to optimizing video transmission over digital networks. Examining recent studies has shed light on the ever-changing video compression market, revealing the dominance of deep learning-based algorithms over more conventional coding standards. Understanding the evolution of video compression and its consequences for effective transmission is made easier with new knowledge about deep neural network topologies, compression frameworks, and assessment measures.

In conclusion, this review has successfully traversed the complex landscape of optimizing computer network video transmission efficiency. This survey provides a concise yet comprehensive roadmap for researchers and practitioners who want to shape the future of efficient video transmission by synthesizing recent advancements and illuminating future directions in video compression, adaptive streaming, QoS mechanisms, DNN architecture for video compression and P2P architectures.

The study identified a variety of solutions, such as error correction mechanisms to prevent data loss, content delivery networks (CDNs) to distribute content closer to end-users, and adaptive video streaming approaches that dynamically modify video quality to match network conditions. The development of new codecs and compression techniques that provide a happy medium between quality and data size has the potential to completely transform video transmission.

Improvements to the underlying network architecture were emphasized as a key factor in attaining high quality video transmission. Network congestion and latency can be greatly reduced with the introduction of 5G technology, fiber-optic connections, and edge computing facilities, all of which contribute to an improved user experience. The survey also looked into the growing role that machine learning and AI play in optimizing video transmission through the prediction of network behavior, adaptation of transmission settings, and improvement of video quality in real-time. Solutions to the difficult problems of visual transmission can be found in these technologies.

ACKNOWLEDGMENT

This research is supported by Moscow Institute of Physics and Technology (MIPT), Phystech School of Radio Engineering and Computer Technologies (FRKT), Department of multimedia technologies and telecommunications, Dolgoprudny, Russia.

References

[1]. Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. In Advances in Neural Information Processing Systems, 2017.
[2]. Alexander Alemi, Ben Poole, Ian Fischer, Joshua Dillon, Rif A Saurous, and Kevin Murphy. Fixing a broken ELBO. In International Conference on Machine Learning, 2018.
[3]. Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. International Conference on Learning Representations, 2018.
[4]. Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. International Conference on Learning Representations, 2016.
[5]. Charu Pandey, Satish Kumar, Rajinder Tiwari, "AnInnovative Approach towards the Video Compression Methodology of the H.264 Codec: Using SPIHT Algorithms ", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-5, November 2012.
[6]. B. Girod, E. G. Steinbach, and N. Faerber, "Comparison of the H. 263 and H. 261 video compression standards," in Standards and Common Interfaces for Video Information Systems: A Critical Review, 1995.
[7]. N. Johnston, D. Vincent, D. Minnen, M. Covell, S. Singh, T. Chinen, S. J. Hwang, J. Shor, and G. Toderici, "Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4385–4393, 2018.
[8]. B. Bross, J. Chen, J.-R. Ohm, G. J. Sullivan, and Y.-K. Wang, "Developments in international video coding standardization after AVC, with an overview of Versatile Video Coding (VVC)," Proceedings of the IEEE, 2021.
[9]. Nishat, Muhammad Kamran & Gnawali, Omprakash & Abdelhadi, Ahmed. (2020). Adaptive Bitrate Video Streaming for Wireless nodes: A Survey.
[10]. Spiteri, Kevin & Urgaonkar, Rahul & Sitaraman, Ramesh. (2016). BOLA: Near-Optimal Bitrate Adaptation for Online Videos.
[11]. G. Lu, C. Cai, X. Zhang, L. Chen, W. Ouyang, D. Xu, and Z. Gao, "Content adaptive and error propagation aware deep video compression," in European Conference on Computer Vision, pp. 456–472, Springer, 2020.
[12]. S.VETRIVEL, & K.SUBA, & Dr, G.ATHISHA. (2010). AN OVERVIEW OF H.26x SERIES AND ITS APPLICATIONS. International Journal of Engineering Science and Technology. 2.
[13]. Sullivan, Gary J., Jens-Rainer Ohm, Woojin Han and Thomas Wiegand. "Overview of the High Efficiency Video Coding (HEVC) Standard." *IEEE Transactions on Circuits and Systems for Video Technology* 22 (2012): 1649-1668..
[14]. A. Djelouah, J. Campos, S. Schaub-Meyer, and C. Schroers, "Neural inter-frame compression for video coding," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.
[15]. E. Agustsson, D. Minnen, N. Johnston, J. Balle, S. J. Hwang, and G. Toderici, "Scale-space flow for end-to-end optimized video compression," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8503–8512, 2020.
[16]. C.-Y. Wu, N. Singhal, and P. Krähenbühl, "Video compression through image interpolation," in ECCV, 2018.
[17]. R. Yang, Y. Yang, J. Marino, and S. Mandt, "Hierarchical autoregressive modeling for neural video compression," 9th International Conference on Learning Representations, ICLR, 2021.
[18]. T. Ladune, P. Philippe, W. Hamidouche, L. Zhang, and O. Déforges, "Optical flow and mode selection for learning-based video coding," in 22nd IEEE International Workshop on Multimedia Signal Processing, 2020.
[19]. J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, "Variational image compression with a scale hyperprior," 6th International Conference on Learning Representations, ICLR, 2018.
[20]. D. Minnen, J. Ballé, and G. Toderici, "Joint autoregressive and hierarchical priors for learned image compression," arXiv preprint arXiv:1809.02736, 2018.
[21]. J. Pessoa, H. Aidos, P. Tomás, and M. A. Figueiredo, "End-to-end learning of video compression using spatio-temporal autoencoders," in 2020 IEEE Workshop on Signal Processing Systems (SiPS), pp. 1–6, IEEE, 2020.
[22]. A. Habibian, T. v. Rozendaal, J. M. Tomczak, and T. S. Cohen, "Video compression with rate-distortion autoencoders," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7033–7042, 2019.
[23]. Z. Hu, Z. Chen, D. Xu, G. Lu, W. Ouyang, and S. Gu, "Improving deep video compression by resolutionadaptive flow coding," in European Conference on Computer Vision, pp. 193–209, Springer, 2020.
[24]. W. Bao, W.-S. Lai, C. Ma, X. Zhang, Z. Gao, and M.-H. Yang, "Depth-aware video frame interpolation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3703–3712, 2019.
[25]. S. Niklaus and F. Liu, "Softmax splatting for video frame interpolation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5437–5446, 2020.
[26]. H. Wang, W. Gan, S. Hu, J. Y. Lin, L. Jin, L. Song, P. Wang, I. Katsavounidis, A. Aaron, and C.-C. J. Kuo, "MCL-JCV: a JND-based H. 264/AVC video quality assessment dataset," in 2016 IEEE International

Conference on Image Processing (ICIP), pp. 1509–1513, IEEE, 2016.

[27]. T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," International Journal of Computer Vision (IJCV), vol. 127, no. 8, pp. 1106–1125, 2019.

[28]. D. Xu, G. Lu, R. Yang, and R. Timofte, "Learned image and video compression with deep neural networks," in 2020 IEEE International Conference on Visual Communications and Image Processing, VCIP 2020, Macau, China, December 1-4, 2020, pp. 1–3, IEEE, 2020.

[29]. Deep Neural Network based Video Compression for Next Generation MPEG Video Codec Standardization," Tae Meon Bae, http://ofinno.com/article, Oct. 2020