

Simultaneous Stream Transmission Methods for Free Viewpoint TV: A Comparative Study

Mudassar Hussain¹, Muhammad Almas Anjum⁴
Department of Computer Science
University of Wah, Wah Cantt, Pakistan^{1,4}

Abdurahman Hassan A Alhazmi²
College of Science and Engineering
Flinders University, Adelaide, South Australia

Rashid Amin³
Department of Computer Science
University of Engineering and Technology, Taxila, Pakistan

Ali Tahir⁵
College of Computer Science and Information Technology
Jazan University, Jazan, KSA

Abstract—Free Viewpoint TV is a system to view natural videos and allow users to control the viewpoint interactively. The main idea is that the users can switch between multiple video streams to find viewpoints of their own choice. The purpose of this research is to provide fast switching between video streams so that users experience less delay while viewpoint switching. In this paper, we discussed different stream switching methods in detail, including their transmission issues. In addition, we discussed various scenarios for fast stream switching in order to make services more interactive by minimizing delays. The quality of service is another factor which can be improved by assigning priorities to the packets. In addition, we discussed simultaneous stream transmission methods which are based on predictions and reduced quality streams for fast switching. Finally, we propose a prediction algorithm (Linear Regression) and system model for fast viewpoint switching and evaluate simultaneous stream transmission methods for free Viewpoint TV. The results indicate that the proposed system model improves the viewpoint switching and perform fast switching.

Keywords—Free viewpoint TV; comparison of stream switching methods; video stream switching; fast switching; simultaneous transmissions; linear regression

I. INTRODUCTION

Video streaming is getting popularity in communication and networks in this information age. It is used in many areas, e.g. live sports, education, entertainment (live/On-demand), etc. Voice over IP (VoIP) [1] has made major changes in telecommunications sectors by replacing circuit switching technology. The video streaming can be provided according to requirements and demands of applications by unicasting, multicasting, broadcasting or peer-to-peer. Different coding techniques are used for the transmission of multimedia streaming. People want fast switching between streams in living, on-demand, or Internet Protocol Television (IPTV) [31]. Free Viewpoint TV (FTV) is the latest technology which facilitates users to switch between multiple streams to find viewpoints of their own choice. This stream switching should be fast enough so that users should not be annoyed by delaying. This makes the technology more interactive, and users will really enjoy using this service. It is quite effective to use this technology in live sports events.

Users can watch any side view of the scene by switching between different viewpoints. Channel zapping time is the duration of time between channel change request by the user and the requested channel being available to the user. This zapping time should be short enough to provide fast channel change [33]. One way to provide fast channel change is to pre-join channels which are most likely to be selected next by the user [1]. Three-dimensional TV (3DTV) and FTV are critical applications of multi-view imaging. Both are based on rendering techniques, thus making the application much more interactive. 3DTV is also called stereo TV which generates 3-D depth impression of the scenes. FTV allows selection of a specific viewpoint of the scene in a particular direction, and that viewpoint is provided to the user [2]. Thus, to make applications more interactive to the users fast switching of viewpoint is necessary. This research describes different stream switching methods in detail along with transmission and switching issues related to those streams switching methods. The switching time varies from live to recorded events. The study shows that stream switching time more than 0.5 seconds is annoying to the users [3]. So, there should be fast switching between stream switching. Of course, the stream switching time varies from application to application. For example, if there is switching between some recorded scenes, then there will be fast switching, while in the case of live streaming, it will not be fast enough.

The main focus of this paper is to provide a deep knowledge of FTV based systems which are built on stream switching techniques in order to provide fast and interactive stream switching. The rest of paper is organized as follows. Section II provides related work which consists of transmission methods and video coding schemes. In Section III, stream switching methods are discussed and simultaneous stream transmission methods are described in Section IV. Section V presents the proposed system model and linear regression algorithm for simultaneous stream transmission. Finally, Section VI consists of conclusion and future work.

II. RELATED WORK

This section describes related work which consists of transmission methods and video coding techniques for video streaming.

A. Transmission Methods

Transmission methods are categorized into four parts according to the application requirements and sender/ receiver relationship.

1) *Unicasting*: This transmission method is also called point to point method. The idea behind unicasting is to transfer data from one source to one receiver. This type of communication is very costly in terms of bandwidth consumption, as each host needs to connect server in order to receive the required data. Video on demand and communication over the phone are examples of unicasting in which each host is directly connected to the server for communication.

2) *Multicasting*: In this transmission method [4] information from the server can be sent to multiple hosts at the same time. This transmission method is used in streaming and in IPTV environment.

3) *Broadcasting*: In this transmission method information is sent to all hosts on the network. This transmission mode is quite useful when large numbers of hosts are connected to the server and information needs to be sent to all hosts. If some hosts on the network do not want information, then it will create extra load on the network and bandwidth consumption also increases which can lead to congestion problems.

4) *Peer to Peer*: In this transmission method end nodes are logically connected to each other; these end nodes are called peers. This method enables to share information between peers in a distributed manner. In earlier days peer to peer (P2P) networks were designed only for file sharing but now-a-day this method is getting much popularity in multimedia streaming. In P2P streaming architecture, a peer is involved with three roles, i.e. Source, Intermediate, and Destination. The source contains media content, and it shares with other peers. Intermediate peer receives media content and shares it with other Intermediate peers. Finally, destination peer receives intended content from one or multiple intermediate peers depending on network architecture [34]. Now we shortly describe two network architectures, multiple sources, and single source.

Multiple Sources P2P Network Architecture: In this network architecture, there exist more than one source peers for the requested stream. Each sender peer can send multimedia contents to one or several requested peers and each requested peer can receive contents from one or multiple sending peers.

Single Source P2P Network Architecture: In this network architecture [5], multimedia contents are stored in one source peer and can be transmitted to one or multiple requested peers. The intermediate node plays an important role by buffering contents and if any new peer requests then the intermediate node transmits to it.

B. Video Coding

In real time multimedia communications, bandwidth requirements are quite high. To handle this constraint, multimedia contents are first compressed and then sent over the

network. Video coding [6] includes some data structures which are quite helpful during the encoding/decoding process. This data structure is shown in Fig. 1. A picture consists of several blocks. Each block has a size of 8 x 8 pixels. A group of blocks form a macroblock which is mostly used for motion estimation/compensation. A group of macroblocks form a slice which is used for resynchronization to the main data stream. This is done by inserting a unique bit sequence which is called the start code. Pictures can be encoded with the help of Intra-coded (I) and Inter-coded (P/B) pictures. In case of I pictures, it utilizes information of the same picture. However, in case of P/B pictures, it utilizes the information of previous or/and next pictures. These pictures are normally arranged into a group of pictures (GOP) where the 1st picture is I picture and the other are P/B pictures.

1) *Intra coding*: Intra-coded pictures are called I-frames in which encoding is performed by using the current frame and no reference to other frames is required. Outside of this current frame, no temporal processing is performed. Fig. 2 shows the process of encoding and decoding via intra coding. In the encoding process, image blocks are transformed via a Discrete Cosine Transform (DCT). Then DCT coefficients quantized, zigzag scanned (Q) and finally Variable Length enCoded (VLC). The decoding process is the reverse of the encoding process in which at first Variable Length Decoding (VLD) is performed then inverse Quantize function (IQ) and finally Inverse DCT (IDCT). This process is similar to JPEG compression [1].

2) *Inter coding*: Inter coding pictures are referred to as P-pictures. I- and P-pictures which act as a reference for other pictures are called reference pictures. The inter-coded pictures which take reference from previous and next pictures are called B-pictures. In video coding there exists a sequence of pictures which can be captured at a predetermined rate in which much information is static. This static information is called temporal redundancy which can be removed. The coding efficiency is much improved via redundancy minimization [7]. The basic idea behind inter-coding [6] is to find and recycle matching information from the reference pictures in decoding pictures. This idea is quite old, but it is still implemented in many video compression techniques.

3) *Hybrid coding*: Hybrid coding is a combination of both predictive and transforms coding techniques. Hybrid coding is more efficient and presently most coders are a variant of this technique. Fig. 3 represents the hybrid encoding containing ME, ME, PM, IT, etc.

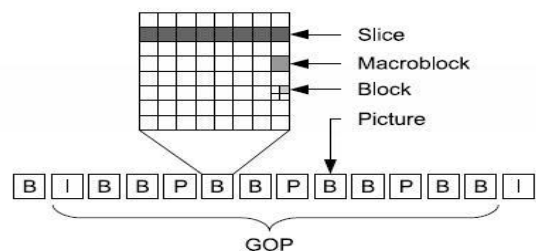


Fig. 1. Video Compression Data Structure (From [6] with Permission).

4) *SP/SI frames*: H.264 encoder [8, 9] has introduced two new frame types, SP- and SI-frames. These frame types can be used for the purpose of bitstream switching, error resilience and random access. Like I-frame, SI-frame utilizes spatial prediction and reconstructs picture which is identical to the SP-frame. In this case, SP-frame utilizes MC prediction.

5) *H.264/AVC encoder*: H.264/AVC is the latest video coding standard. The main idea behind the development of this standard was to improve the compression rate and rate-distortion efficiency [10]. It has introduced two new frame types (SP/SI-frames), which are explained in Section (II-B-4). There are many other new features introduced in H.264/AVC, like, Motion vectors over picture boundaries, redundant pictures, multiple reference picture motion compensation, etc. as shown in Fig. 4. In [11] all these features are explained.

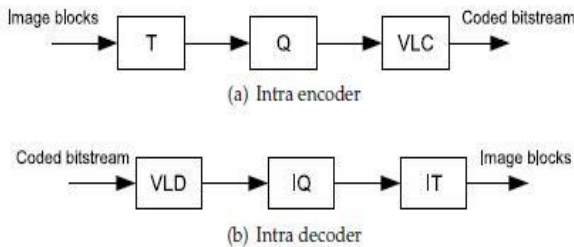


Fig. 2. Intra Encoding and Decoding (From [6] with Permission).

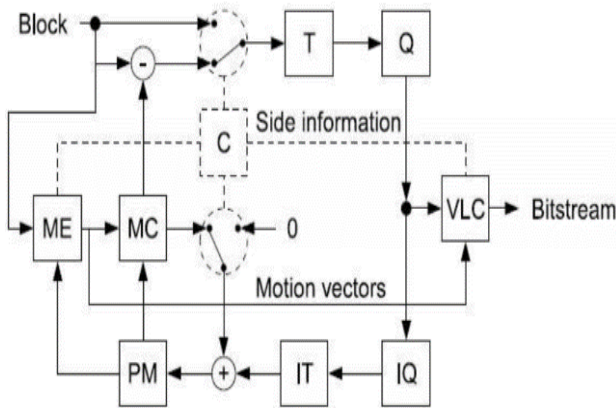


Fig. 3. Hybrid Encoding.

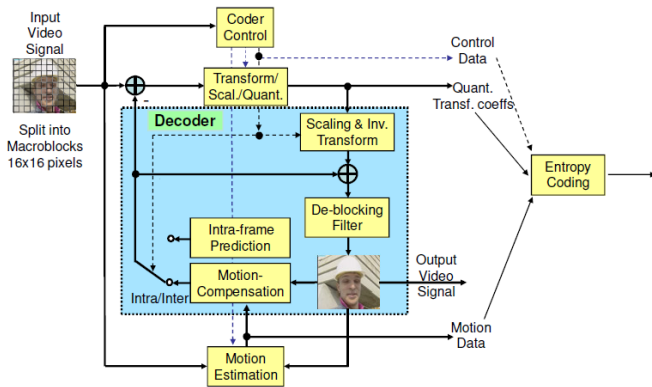


Fig. 4. H 264/AVC Encoder.

III. STREAM SWITCHING METHODS

This section describes stream switching methods which are implemented in many applications. Live, on-demand, as well as recorded video contents, can be delivered to the receivers over the Internet by using multimedia streaming. Some methods which are used for video streaming are based on uncompressed video streaming, but most are based on compressed video streaming. The issue of compressed vs. non-compressed video depends on the application and environment. The idea behind non-compressed video stream switching is that it provides a very high-quality video which can be used for entertainment, live sports events, as well as education purposes. It requires high bandwidth. In compressed video streaming bandwidth requirements are not as high as for non-compressed video. Compressed streaming methods are most likely suitable for IPTV environment. In [32], analysis of IPTV evolution in Korea is presented in which key components of IPTV identified. The key findings of this study states that content plays an important role in the IPTV industry growth. It also highlights the understanding of the evolution for IPTV players and role of open innovation approaches in the development of the IPTV firm strategies.

A. Uncompressed Video Stream Switching

The method presents in [12,13] introduces a non-compressed 4k video transmission system. Uncompressed Super high-definition (SHD) images transmit and display 4096 x 2160 pixel resolution and 36-bit color with 24 or 30 frames per second, which is equal to 6.3 to 9.5 Gbps. The JPEG 2000 compressed SHD images range from 200-400 Mbps, which can be transmitted via common Gigabit IP network [14]. In [12] time code-based video switching, and stream crossfading are introduced for this purpose. Fig. 5 shows the network configuration of this system which consists of three video sources and one receiver via 10 Gbps network.

1) *Time-code based video stream switching*: In time code-based video stream switching, the time code is produced at each streaming server and transmitted with the stream via the same path to the receiver to calculate the difference in delay time. This delay is very significant to provide gapless video stream switching. The whole process is shown in Fig. 6. At the receiver end, timecode packets are received, and time codes of streams are observed, i.e., OT_A and OT_B . Control terminal calculates the difference between both observed codes for switching purposes, i.e., $DT_B = OT_B - OT_A$. The control terminal is responsible for sending ending and starting time to the streaming servers. These times are calculated via below equations.

$$ET_A = OT_{Areq} + MT \quad (1)$$

$$ST_B = ET_A + DT_B \quad (2)$$

Equation (1) shows the end time of stream A (ET_A) which is equal to the observed time of stream A when user requests for switching and MT is the margin time which is the time of arrival of ending packet to server A when the user requests. Equation (2) shows the starting time of stream B, which is based on the sum of ending time of stream A and the difference between observed codes.

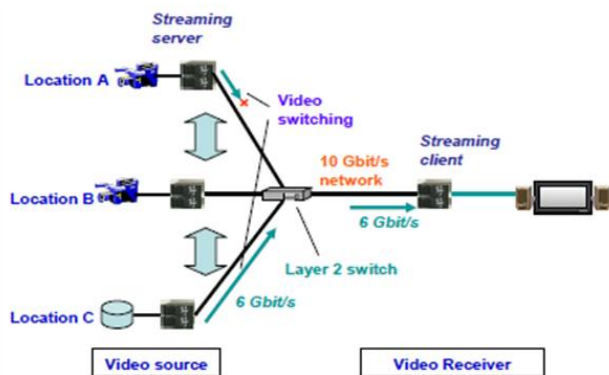


Fig. 5. Network Configuration [12].

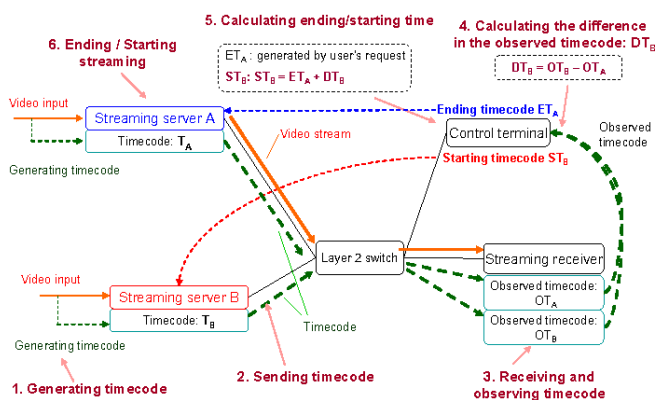


Fig. 6. Time Code-based Video Stream Switching Control [12].

The starting and ending of streams are based on the ET_A and ST_B . Server A stops streaming when its timecode is equal to ET_A and server B starts when its timecode is equal to ST_B .

2) *Prevention of bandwidth overflow with stream crossfading*: The mechanism of video stream crossfading [13] prevents stream overlapping which leads towards bandwidth overflow problem during the stream switching process. When a receiver wants to switch from stream A to stream B, then server A decreases bit rate slowly and server B increases the bit rate slowly. For stream crossfading both sending and receiving servers require buffers. During stream switching from stream A to stream B at the transmitter end, server A (B) buffers the last (first) frame of stream A (B) and sends it throughout two frames. At the receiver end, last (first) frame of stream A (B) is received. When the last frame of stream A is received and played then switching is performed from A to B as a first full frame.

B. View Switchable Multi-View Video Transport

View switchable multi-view video transport [15,16] is a system which is based on IP multicasting and an example of 3D IPTV. It is suited for both videos on-demand as well as live multimedia applications. All the implementations of this technique are software based which enable to reduce hardware cost and ease of use for configuration and maintenance. Each acquisition server compresses each view individually for scalability purposes, which is transmitted over different multicast channels. The whole process is shown in Fig. 7.

C. View Switchable Multi-View Video Transport

View switchable multi-view video transport [15,16] is a system which is based on IP multicasting and an example of 3D IPTV. It is suited for both videos on-demand as well as live multimedia applications. All the implementations of this technique are software based which enable to reduce hardware cost and ease of use for configuration and maintenance. Each acquisition server compresses each view individually for scalability purposes, which is transmitted over different multicast channels. The whole process is shown in Fig. 7. This sort of multi-view video technique can be implanted on a live sports event. Multiple views of an event (scene) can be displayed on the screen as well as one large display in which multiple views can be combined for better coverage.

1) *Issues with multi-view video transport*: In real-time multi-view video transport based on 3D IPTV, two things are important which are, encoding efficiency and complexity. By decreasing the redundancy between multi-view videos, the size of multi-view videos can be reduced but this, in turn, made random access to multi-view video more difficult. In multi-view video transport, each view is encoded independently which is an attractive idea for real time environment, but it cannot exploit redundancy.

D. Video Streaming over Peer-To-Peer Networks

Nowadays sharing of multimedia contents between users is getting much popularity. Peer-to-Peer (P2P) [17] has made it pretty much easy and fast to share multimedia contents with each other. P2P networks consist of a large number of heterogeneous computers called peers. These peers act as client and/or server at the same time. These peers can directly communicate with each other and can share contents directly. P2P networks work in two modes, Open after downloading mode and Play while downloading mode. In the open after downloading mode, media contents are played after downloading all contents from a file or other peers. In Play, while downloading mode, contents are played while downloading is in progress. This mode is used in streaming and it takes lesser memory. Moreover, the client does not need to wait for the download to complete. Due to the dynamic nature of P2P networks, these are not reliable; any peer can enter or leave the network without prior notification.

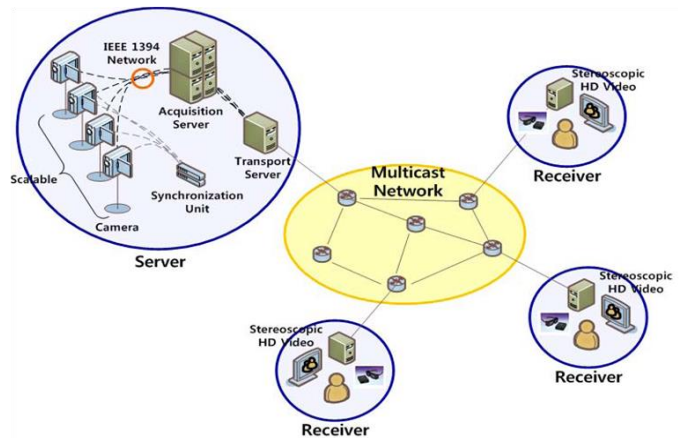


Fig. 7. Multi-View Video Transport [15].

1) *Issues with P2P streaming*: There is a problem in peer selection mechanism that if there are a large number of active peers sending the same video content then selection of a large number of peers for video transmission leads to extra overhead for establishing and monitoring those peers. Due to the dynamic nature of P2P, reliability is also another factor to consider.

E. Stream Switching based on GOP

The basic idea behind channel switching in Group of Picture (GOP) is synchronization points which are used for channel switching [18]. GOP is the organization of different frames in a specific order. It usually starts with a synchronization frame (I-frame) followed by P and B frames, which simplify the stream synchronization. Fig. 8 shows the process of channel switching by the GOP. It also shows the distance of I-frames as the quotient of the number of frames per second (fps) and the number of I-frames per second (RG). In today's IPTV systems, it is prevalent to use GOP to enable synchronization to the transmitted streams (channels). This technique is quite helpful and easy for stream switching and information loss recovery.

Let's consider a client is synchronized with channel A at the start. At the time "a" it requests for switching from channel A to B. The client then sends an IGMP to leave a message to stream A and then starts receiving stream B. The client waits for the I-frame to receive as it acts as a synchronization point for the client decoder. The frames received during this process will be discarded [19]. When the synchronization frame is received entirely and decoded, then the client is synchronized to the new stream (B).

1) *GOP packet loss*: I-frames solve information loss during transmission. Let's consider a client is receiving the channel A, and at time "c" it gets an erroneous frame, this means that now the reference to the previous frame is removed. Now the decoder will wait until an I-frame is received to decode the received frames. During this process, the frames are received, but the decoder did not decode. At the time "d" the client received and decoded the I-frame, and now the decoder is synchronized to the stream and frames will be decoded onwards [19]. This process is shown in Fig. 9.

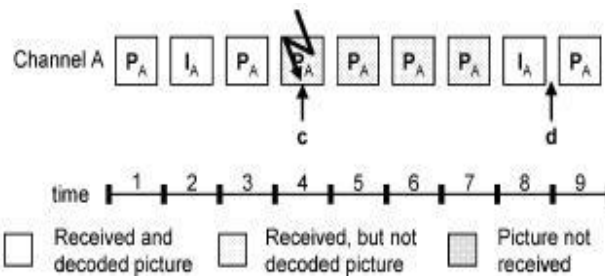


Fig. 9. GOP Packet Loss [19].

F. Synchronization Frames for Channel Switching

In this switching strategy [19], less channel switching and less of information are assumed, so it is redundant to send I-frames at a static rate. In SFCS synchronization, synchronization stream (I-frames) is separated from the main stream. In this technique, one channel is composed of two data streams. One stream consists of P-frames, which is the mainstream, and the other consists of I-frames, which is called the synchronization stream. Fig. 10 shows switching process from channel A to channel B. The client requests for switching from channel A to channel B at the time "a" and joins to the sync channel B. The client then waits for traffic to come and sends to the decoder. At time "b" client is synchronized to stream B. After that it leaves the sync channel and joins the main stream. In this switching strategy, unwanted traffic is prevented by late joining the mainstream. The main advantage compared to GOP is the bandwidth reduction for the specified channel because there is less number of synchronization frames.

1) *SFCS information loss*: The information loss recovery is pretty much the same as SFCS channel switching. If one or more data packets are corrupted, then synchronization to the stream is lost. This issue is resolved by resynchronization to the stream-like channel switching. As shown in Fig. 11, at time "c" the client receives information loss frame at channel A and then leaves the mainstream and joins sync stream of channel A. The client then waits for traffic on the sync channel and transfers to the decoder and finally synchronized to the mainstream at the time "d." The client leaves the sync channel and joins the main channel.

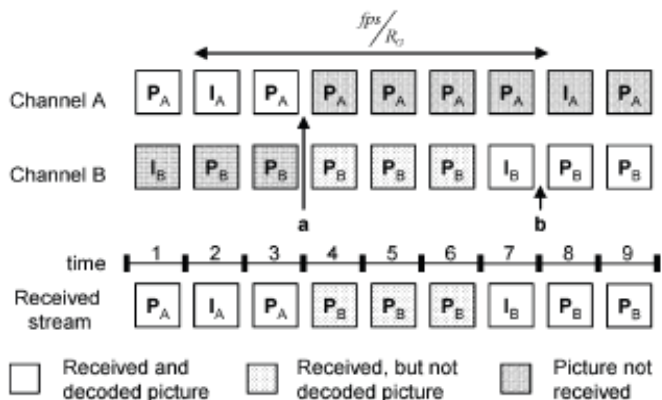


Fig. 8. GOP Channel Switching [19].

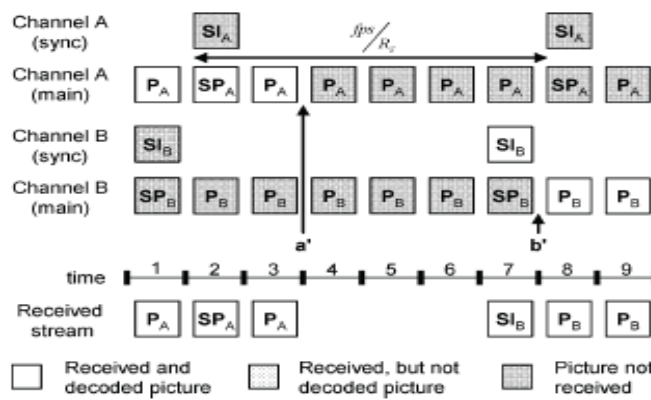


Fig. 10. SFCS Channel Switching [19].

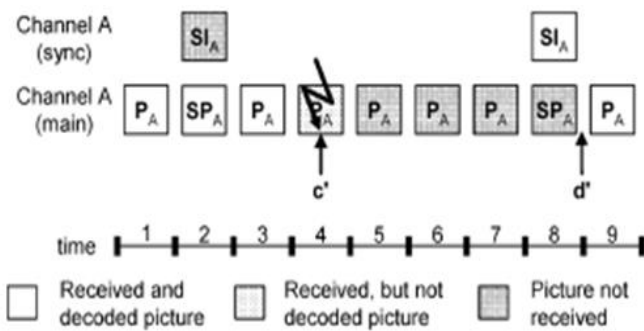


Fig. 11. SFCS Packet Loss [19].

2) *Transmission issues and video quality*: The factors which are mostly used to handle the transmission and quality of the video include priority encoding transmission, unequal packet loss protection, and priority dropping techniques. In wireless networks, mobile networks, and on the Internet, most packet losses are due to congestions, fading, and interference on the channel during communication [20]. As video contents are transmitting via multicasting, so those contents must be delivered to all recipients. Today many applications use protocols which use retransmission for missing contents in case of packet loss [21]. So, in the case of lossy networks implementing these types of protocols is difficult.

G. Fast Stream Switching

In IP networks, live streaming is mostly done through multicasting. In this scenario when any user wants to switch from one stream (channel) it simply issues channel switch command and switches from one channel to another. The time between leaving from one channel and join the other is called channel switch time [22]. This channel switching time is very important. The long channel switching time is pretty much annoying to users. The channel switching time less than 0.5 seconds is acceptable but more than that is annoying.

The channel switching time is varied in the case of recorded and live videos. Let's consider the case of channel switching in the IPTV environment. When a user wants to change the channel, it issues IGMP leave a message to the nearest router and after that by issuing IGMP join message, joins the new channel [6]. This joining of the new channel depends upon the availability of the stream. If the stream is already available, then the channel switch time will be less. Otherwise, it results in delay. In recorded videos switching, time is less as compared to live events. As after sending the join IGMP message, IP-STB at client side needs to wait until the packets arrive from the new stream. The identification of a new stream at the client side is based on I-frames, and after the identification, the decoding process starts. I-frames keep track of all the streams and have all the information about the streams. The distance between I-frames in the stream also depends upon the stream switching time and varies from zero to a reasonable amount of time.

There are many techniques used for fast channel switching. These techniques are implemented and are based on the requirements and network configurations. Boyce and Tourapis [23] proposed a new technique for fast channel switching in

which low-quality stream is multiplexed with the normal stream. These multiplexed streams are transmitted and at the receiver end, de-multiplexing and decoding process is done. In this technique, the multiplexing equipment should be placed near the client, for example, DSLAM or advanced IP switching equipment. This technique works in a DSL environment and for video on demand (VoD). This idea somehow relates to the SFCS as discussed earlier but is not pretty much as in SFCS both sync and main streams are transmitted on different multicast addresses.

H. Ray-Space based FTV Systems

With time, the demand for the number of views is increasing as compared to the number of pixels. A ray is a virtual space which can be created by multi-view images. In [30,31] a ray based FTV system is presented in which capturing, encoding, decoding and view generation is discussed in detail. This system is based on realistic viewing and free navigation on 3D scenes. The trend is now shifting from wired to wireless transmission and from computers to mobile devices which have limited computation power, memory, battery, etc. An energy-aware adaptive free viewpoint video wireless transmission system [28] is aimed to deal with issues mentioned above by minimizing encoding rates and energy consumption. The experiments on the acquisition and processing of video for FTV [29] shows that such kind of simple and cost-effective systems are implementable very easily. With the rapid advancement in cellular technology, a system based on the broadcasting of FTV over long-term evolution networks [30] is proposed with some constraints of network and transmission issues.

IV. SIMULTANEOUS STREAM TRANSMISSION METHODS

This section presents simultaneous stream transmission methods. Stream transmission methods play an important role in any video streaming system. Many constraints in the system can be handled by appropriately using these methods. These constraints include bandwidth constraints, network congestion, etc. Moreover, these methods are quite helpful for fast viewpoint switching in FTV.

A. Simultaneous Transmission of Multiple Streams

The most popular way of simultaneous transmitting of multiple streams is multicasting. However, multiple streams can be transmitted via unicasting. The selection of these techniques depends upon the application environment as well as bandwidth constraints. In 3DTV and FTV, both these strategies can be implemented. We are going to talk more about the two flavors of multicasting. These are network layer multicast and application layer multicast. Among these two, application layer multicast is proposed in this simultaneous stream transmission method.

1) *Network layer multicast*: In network layer multicasting, the transmitter sends each packet only one time. At multicast enabled routers all these packets get together and are forwarded to other routers and hosts as needed. This strategy is most efficient, but it is not implemented broadly as it requires multicast-enabled routers. So, most network providers don't bother to replace existing equipment because it requires

much cost and may be some network downtime. Albeit, many new routers are multicast enabled but many network providers disable this functionality due to security reasons. So, this strategy is mostly not implemented widely.

2) *Application layer multicast*: In the application layer multicasting, the functionality of packet forwarding, duplication, and management is shifted to end hosts, and it is done through software application instead of hardware devices. This strategy is not as efficient as network layer multicast because some packets get duplicated and also may require more hops to reach the final destination. It is very easy to implement, and also it does not require much investment for buying hardware, so due to these factors, it is widely implemented in many network environments. In [24] application layer multicast is used for transmission of the selected number of multiple streams. These streams are quite helpful to render video from its current viewpoint.

3) *NICE Protocol*: NICE is an application layer protocol which is generally used for transmitting multi-view videos. In this type of protocol, all the members of the multicast group form small clusters which are based on geographical areas. The distance of geographical areas is calculated via ping RTT. These clusters further makeup lower layer 'L₀' in a hierarchical manner. The most central member in every cluster is designated as the leader and upheld to next higher layer 'L₁'. This process is repeated until we reach a point where leaders in cluster 'L_n' become a member of 'L_{n+1}' and a single member becomes the root of the hierarchy at higher layer 'L_{nmax}'. The hierarchy in the multicast groups describes data delivery paths implicitly. This, in turn, eliminates the packet delivery tree states and control meshes. In the hierarchical tree, each host maintains detailed information about their closest neighbors which is also a big advantage as compared to other multicast protocols.

In [24] 3D delivery system is proposed for multi-view video transmission. In this system, overlay distribution trees are built for every camera view and for every depth map stream. Every receiver finds out specific parts of image-based rendering (IBR) representation for the purpose of their current view rendering and subscription of corresponding distribution trees. By using Kalman filter future viewpoints can be predicted and necessary streams are fetched in advance which provides fast switching.

4) *Multicasting framework*: Multicasting framework consists of one or more multicasting streaming servers. Every streaming server contains streams which correspond to different views of multi-view video data. Moreover, there exist some professional peers who implement the NICE protocol. The receivers execute client software at their side to request streams from a multicast peer which is already known and forward that stream to rendering software for rendering the view. This rendering software is based on IBR technique. The rendering module does two very important tasks. It first renders the current view from streams and then instructs the multicast client to request relevant streams [24].

5) *Image-based rendering techniques*: Image-based rendering (IBR) techniques are an essential part of 3DTV and FTV. These techniques are used to render views based on several streams. These streams are predicted based on Kalman filter or other prediction methods. Moreover, in 3DTV and FTV head/eye tracking system is used to track viewpoints and streams are captured based on that technique. The rendered viewpoints are called virtual viewpoints. The area of viewpoint rendering has attracted a lot of research interest. The basic idea behind IBR techniques is seven-dimensional plenoptic function [25]. It illustrates all available optical information in a given region.

6) *Network and transmission issues*: There are almost always constraints regarding bandwidth in real time applications. In the case of simultaneous transmission of multiple streams bandwidth requirements almost doubles which create a lot of problems. The implementation of multicasting strategies improves efficiency by reducing duplicate packets at the server side in the network. These strategies also pretty much useful to reduce bandwidth requirements at both transmitter and receiver ends. Independent implementation of compression techniques on every stream at the server side also helps a lot to reduce bandwidth constraints.

B. Simultaneous Transmission of Reduced and High-Quality Streams

One way to transmit reduced and high-quality streams simultaneously are via Multi-view video delivery system. In this delivery system, only those streams are transmitted to the receivers which are necessary for rendering their viewpoint. In this method, lower bit rate versions of a set of adjacent streams are also transmitted with the actual stream. So, during switching if an unpredicted viewpoint change happens then a reduced quality version of the stream is already available which can be decodable till the arrival of the actual high-quality stream. This, in turn, provides much better results in the view switching process and provides a sort of guarantee for fast view switching by minimizing delays of requested streams and providing reduced quality streams.

1) *Video delivery system*: Fig. 12 shows the architecture of the video delivery system. In this system, the multi-view video is transmitted to clients via an IP-network. At the time, 't' the receiver viewpoint is sampled and at the time 't+d' future viewpoint is predicted via kalman filter. Here 'd' is the prediction distance which is calculated based on network and decoding delay. This distance is actually the sum of both delays. After this distance, the requested stream gets started playing. The network delay is a common issue for both unicast and multicast architectures. In the case of unicasting, the RTT delay of connection establishment between client and server is called 'network delay'. However, in the case of multicasting network delay is referred to as join latency to transmit multiple streams. The decoding delay depends upon the coding structure. Compression efficiency is much better in the case of longer GOP size but it results in longer decoding

delay. This longer decoding delay can be accommodated by increasing prediction distance and it is made sure that I-frame has been received as well as the stream is decodable before displaying to the users. Moreover, with longer delays, the prediction becomes less reliable which in turn results in prediction errors and it is possible that wrong streams are fetched from the server. So there must be a tradeoff between compression efficiency and decoding delay to avoid recurrent prediction errors [26].

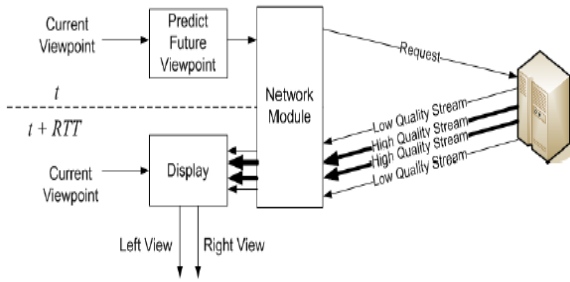


Fig. 12. Video Delivery System [26].

By using prediction on future viewpoint, streams are fetched from the server along with reduced quality neighboring streams. In the case of the free-viewpoint stereo system, the whole viewing space is divided into ‘N-1’ regions. Here N is the number of views in the multi-view sequence. In this system, the region ‘R_n’ is associated with ‘V_n’ and ‘V_{n+1}’ views. When the viewer moves from region ‘R_n’ to ‘R_{n+1}’ views ‘V_{n+1}’ and ‘V_{n+2}’ are displayed to the user. As future viewpoint is predicted, so by providing requested streams in advance, the delay of showing required stream is minimized which results in fast switching. In the case of prediction error, the reduced quality nearby streams are already available, which prevents the stoppage of stereo viewing and improves systems performance [26].

2) In a transmission system based on the simultaneous transmission of reduced and high-quality streams network and decoding, the delay is sort of problems for fast switching. These delays must be minimized to achieve fast switching. The requirement of bandwidth is also an issue to consider but it is not as much crucial as compared to the simultaneous transmission of multiple streams. As here we are going to transmit reduced quality stream with high-quality stream simultaneously. So, we can somehow sacrifice on this issue.

V. PROPOSED SYSTEM MODEL

In FTV, the stream switching time is much reduced by transmitting the next probable requesting stream along with the actual stream. In this scenario, multiple calibrated cameras are mounted on the site (live sports event) and a unique scene is captured from all distributed cameras. Simultaneous transmission of multiple streams creates a lot of problems, e.g., service delay, bandwidth consumption, etc. So, there is a need to implement some strategy at server side to transmit that stream (probable stream) along with current stream which somehow helpful to minimize above mentioned issues and

provide fast viewpoint switching. The transmission of the probable stream can be predicted by view prediction algorithm, namely, linear regression (LR).

Block diagram of the system model for simultaneous transmission of most probable next stream and the actual stream is shown in Fig. 13. This system model shows that there are four calibrated cameras distributed on the scene to cover the whole live sports event. There is implemented prediction technique namely ‘linear regression’ on the server to predict the client’s desired view sequences and video-based rendering techniques to reconstruct views. We refer these reconstructed views as virtual views which are created when the client moves from one camera location to the other. Based on the prediction technique (LR algorithm) the client’s viewpoint sequence is calculated and the regression line is marked. Based on the regression line, a new position of viewpoint is estimated.

A. Linear Regression Algorithm

Linear regression is a statistical method which is used to describe the relationship between two or more variables. These variables are called predictor and response variables. This method is suitable for a data model which is linear in model coefficients. This sort of linear regression is called least square fit. In real time applications like multimedia streaming, the implementation of such a linear model provides many benefits in terms of computational complexity and usability perspectives. Viewpoint prediction is achieved by using groupings of predictive data, received data and based on prior knowledge of working with the system. This method makes sure that output is linearly dependent on the input. This system model is described in equations (3) and (4) below.

$$\hat{Y} = \beta_0 + \beta_1 \times X \quad (3)$$

$$Y = \beta_0 + \beta_1 \times X + \epsilon \quad (4)$$

Here ‘ \hat{Y} ’ is the predicted output value and ‘X’ is the given input value which is the independent variable. ‘ β_0 ’ and ‘ β_1 ’ are parameters of the model. ‘ ϵ ’ corresponds to residual error from the regression line. These parameters are estimated from the training set which consists of n observations. These observations are structured as (X₁, Y₁), (X₂, Y₂), ..., (X_n, Y_n).

$$\beta_1 = \frac{\sum_{i=1}^n (X_i - \bar{X}) \times (Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} \quad (5)$$

$$\beta_0 = \bar{Y} - \beta_1 \times \bar{X} \quad (6)$$

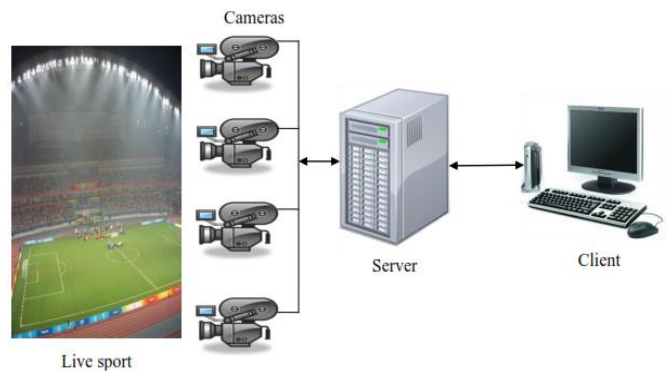


Fig. 13. Block Diagram of the Proposed System Model.

Here X and Y show the mean of all values for variables X and Y.

The calculations in equations (5) and (6) are carried out at the training level in order that the convergence of model functions is done. When these calculations are done and mentioned parameters are set up, the computation complexity at server side is much reduced and there are left only minor additions and multiplications which are performed by the LR algorithm [27], as in equation (3).

B. Regression Algorithm Explanation

This algorithm is implemented at the server side to predict the desired view sequences requested by the client. This algorithm makes it possible to predict the velocity of changing of viewpoints by the client and mark the regression line which shows the rate of change of the client's viewpoint. The algorithm's computational complexity is reduced to some extent due to the inherent nature of the system's working. Moreover, the accuracy of the regression line is also guaranteed.

In this system model, there are very fewer chances of errors during information acquisition because observations are done at the client side and the client enters the values. In this way, only two samples of training sets are generated which are used by the algorithm to provide robust regression line. As the user is changing the view by a certain velocity, so it's a free viewpoint operation. We propose that the predicted view is also transmitted simultaneously along with the actual stream. This provides fast view switching if the prediction is accurate. After this stage, it is checked at the client whether the viewpoint is accurate according to the client's request. If a viewpoint is accurate then it means that prediction is correct and viewpoint will be immediately provided to the client. In this case, fast switching of viewpoint will be guaranteed.

On the other hand, if the prediction is incorrect, i.e. viewpoint is not matching to the client's request then feedback is sent to the server which waits for the retransmission of 2nd data packet before algorithm's re-execution. When feedback is received at the server, the requested viewpoint is altered. This also makes sure that the request has been received and fulfilled. Finally, the regression line is computed, which shows a rate of change of viewpoint [28]. The previous delay is adjusted and the prediction process is resynchronized according to user's requirement. This process takes time, so the client will experience a delay while receiving a view of his own choice.

C. Viewpoint Prediction Calculation

The viewpoint prediction calculation is performed via the Linear Regression algorithm. In this algorithm prediction of viewpoint is calculated based on the previous history. The history contains user input data and observed values. In the following example, we have taken some data which contain both user input values and observed values. We calculate the parameters of the algorithm ' β_0 ' and ' β_1 ' from the data set. This parameter calculation is performed earlier and after that based on history, we apply the linear regression to get predicted values. These predicted values are actually our predicted view sequences. All these values are shown below in Table I.

TABLE. I. LINEAR REGRESSION DATA

Linear Regression		
Input Value (X)	Observed Value (Y)	Predicted Value (\hat{Y})
1	1.00	1.45
2	2.50	1.825
3	1.50	2.2
4	3.75	2.575
5	2.25	2.95

$$\beta_0 = 1.075, \beta_1 = 0.375$$

Based on predicted values, we draw the regression line, which is called the best-fitted line to all the observed values. This regression line is shown in Fig. 14. This line clearly shows the predicted value in a particular sequence and based on this sequence, the next probable viewpoint is estimated.

1) *Network and transmission issues:* There are also bandwidth constraints with this strategy. As we are transmitting multi-streams, so bandwidth requirements rise to double on the link. However, these constraints can be handled by applying compression techniques on the server side. The congestion on the bidirectional link between client and server is much reduced by applying the prediction strategy. As proposed prediction technique provides much better results [28, 29], so there are very fewer chances of errors. Thus, feedback communication between client and server is much reduced. This in turn also helps to minimize round trip delays.

D. Discussion

1) *The proposed system model for FTV is shown in Fig. 13, which is based on client/server architecture.* In this system model, we proposed that all the cameras are connected to the server, and the server is connected to the client via a bidirectional link. In linear regression algorithm is implemented at the server for most probable next view prediction. The author in [27] proposed that the requested viewpoint is transmitted to the client. We propose that the next predicted viewpoint is also transmitted along with the actual viewpoint. So, this will really help a lot towards fast switching. As this algorithm provides much prediction accuracy [27], so it results in fast view switching, and it is quite suitable for FTV.

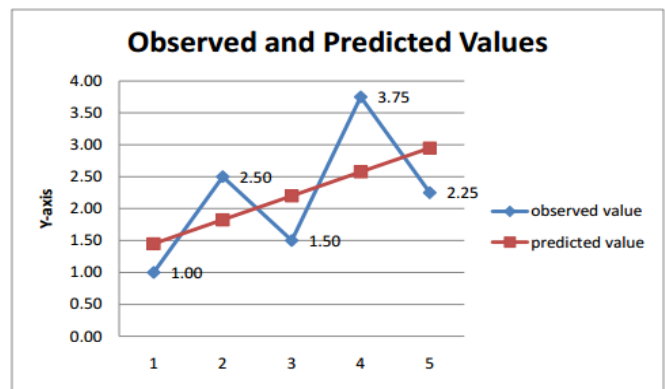


Fig. 14. Regression Line.

We have also calculated the regression line based on some data set. The regression line shows the sequence of viewpoints. As we can estimate the next probable viewpoint, so by transmitting that viewpoint along with actual viewpoint based on our proposed system model, fast viewpoint switching is guaranteed. Furthermore, our proposed system model provides a much better quality of service (QoS) in the sense that congestion on the network is much reduced by avoiding transmission of those streams which are not needed by the client at all.

2) *System evaluation* with respect to FTV includes quality of view rendering, bandwidth constraints, prediction accuracy, and fast switching. Based on simultaneous stream transmission methods as discussed earlier, we are now going to evaluate which one is best for FTV and why? In the case of simultaneous transmission of multiple streams, view rendering is performed at the client end, and multiple streams are transmitted from the server. The transfer of various streams has bandwidth constraints which may double or more bandwidth requirements, and it is a big problem in real time video streaming. It provides fast switching in the sense that if the user changes his viewpoint and the required streams for that viewpoint are already available then there will be fast switching.

In the case of simultaneous transmission of the reduced quality stream along with the actual high-quality stream, view rendering is still performed at client end, but bandwidth requirements reduced. As in this method, reduced quality streams are transmitting along with actual streams, so bandwidth requirements are somehow tolerated. Fast switching is also achieved in a better way as compared to the previous one in the sense that if the prediction for streams turns incorrect, then there are already available reduced quality streams which can be decoded. Thus, it avoids stoppage of stereo streaming and provides fast switching.

Lastly, simultaneous transmission of most probable next and actual stream provides much better results as compared to both previous methods. Bandwidth requirements are much reduced by transmitting predicted view along with real view via prediction technique, which provides many accurate results [28]. Since in this method there is no need to transmit streams at the client side so network bandwidth requirements also much reduced. So congestion on the network is also much reduced. As in this method, view rendering is performed at the server side, and the next probable view is transmitted along with the actual view, so, there will be much better results regarding fast switching.

VI. CONCLUSIONS AND FUTURE DIRECTIONS

FTV is the latest technology which facilitates users to switch between multiple streams to watch viewpoint of their own choice. This stream switching should be fast enough so that users should not be annoyed by delaying of video stream switching. This makes the technology more interactive, and users will really enjoy using this service. It is quite effective to use this technology to live sports events. In this paper, we provide comprehensive knowledge about different components

of FTV from several perspectives. We discussed different approaches according to categories like (1) Transmission methods and video coding (2) Stream switching methods, (3) Simultaneous Stream Transmission methods, etc. In addition, we propose a system model for FTV and calculate probable next viewpoint by using linear regression algorithm for fast viewpoint switching. The method of simultaneous stream transmission of actual and probable next stream is best from others.

We suggest implementing stream switching methods as well as simultaneous stream transmission methods based on predictions as future work. We think that the implementation of stream switching methods is quite interesting. Although, some stream switching methods have been implemented and tested already but it is still a quite emerging area. Simultaneous stream transmission methods based on FTV is quite an interesting area in multimedia streaming and a lot of research is in progress. Moreover, FTV with respect to 3DTV could also be an interesting future research work.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCES

- [1] Chae Young Lee, Member IEEE, Chang Ki Hong and Kang Yong Lee, "Reducing channel zapping time in IPTV based on user's channel selection behaviors", IEEE, Transactions on broadcasting, vol. 56, no. 3, September 2010.
- [2] Zhou, Shiwei, Yu Hen Hu, and Hongrui Jiang. "Multiple view image denoising using 3D focus image stacks" IEEE Global Conference on Signal and Information Processing (GlobalSIP). IEEE, 2015.
- [3] Claus Popp Larsen, Rasmus Flodin, Christer Lindqvist, Rickard Lindström, Hemamali Pathirana and Anders Gavler, "Experience with IPTV in the Testbed", Acreo Broadband Communication project report, Y2002-Y2006.
- [4] Brehmer, H. Jacqueline, "Data Localization: The Unintended Consequences of Privacy Litigation", AM. UL Rev. 67 (2017): 927.
- [5] Wang, Shiyu, et al., "Fighting Pollution Attack in Peer-to-Peer Streaming Systems: A Dynamic Reputation Management Approach", Trustworthy Systems and their Applications (TSA), 2Third International Conference on. IEEE, 2016.
- [6] Kardeby, Victor, Ulf Jennehag, and Mikael Gidlund., "Power consumption for global information dissemination in the Internet of Things", In IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), PP. 1-6. IEEE, 2015.
- [7] Shen, Liquan, Kai Li, Guorui Feng, Ping An, and Zhi Liu., "Efficient intra mode selection for depth-map coding utilizing spatiotemporal, inter-component and inter-view correlations in 3D-HEVC", IEEE Transactions on Image Processing 27, no. 9 (2018): 4195-4206.
- [8] De Jesus Ochoa Dominguez, Humberto, et al., "The H. 264 video coding standard", Potentials, IEEE 33.2 (2014): 32-38.
- [9] Bordes, Philippe., "Adapting video compression to new formats", PhD diss., Rennes 1, 2016.
- [10] Özgü Alay, "Fast intra/inter mode decision for a real-time h.264 streaming system", Master Thesis, Middle East Technical University, July 2006.
- [11] Wiegand, Thomas, Gary J. Sullivan, and GB Ajay Luthra., "Overview of the h. 264/avc video coding standard," IEEE Transactions on Circuits and Systems for Video Technology, 2015.
- [12] Daisuke Shirai, Tetsuo Kawano, and Tatsuya Fujii, "6 Gbit/S uncompressed 4k video stream switching on a 10 gbit/s network", IEEE Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems Nov.28-Dec.1, 2007 Xiamen, China.

- [13] D. Shirai, T. Shimizu, et al., "Real time switching and streaming transmission of uncompressed 4K motion pictures", *The International Journal of Grid Computing and Science, Future Generation Computer Systems*, 2009, PP. 192–197.
- [14] Zhou, Mingliang, et al., "Complexity-based Intra Frame Rate Control by Jointing Inter-Frame Correlation for High Efficiency Video Coding", *Journal of Visual Communication and Image Representation*, 2016.
- [15] Ho, Ting-Yu, Yi-Nung Yeh, and De-Nian Yang., "Multi-View 3D Video Multicast for Broadband IP Networks", *arXiv preprint arXiv:1410.3977*, 2014.
- [16] Engin Kurutepe, Student Member IEEE, M. Reha Civanlar, Fellow IEEE, and A. Murat Tekalp, Fellow IEEE, "Client-Driven Selective Streaming of Multiview Video for Interactive 3DTV", *IEEE Transactions on circuits and systems for video technology*, vol. 17, no. 11, November 2007.
- [17] Chakareski, Jacob., "Adaptive multiview video streaming: challenges and opportunities", *Communications Magazine*, IEEE 51.5 (2013): 94-100.
- [18] Ruffini, Marco, Emanuele Di Pascale, and David B. Payne., "Improving high fidelity multimedia distribution in next-generation optical networks", *IEEE 15th International Conference on Transparent Optical Networks (ICTON)*, 2013.
- [19] Memos, Vasileios A., Kostas E. Psannis, Yutaka Ishibashi, Byung-Gyu Kim, and B. Brij Gupta, "An efficient algorithm for media-based surveillance system (EAMSuS) in IoT smart city framework", *Future Generation Computer Systems* 83 (2018): 619-628.
- [20] Barkowsky, Marcus, et al., "Hybrid video quality prediction: Reviewing video quality measurement for widening application scope", *Multimedia Tools and Applications* 74.2 (2015): 323-343.
- [21] A. Albanese, J. Bloemer, J. Edmonds, M. Luby, "Priority encoding transmission", *IEEE Transactions on Information Theory (special issue devoted to coding theory)* 42 (6) (November 1996) 1737-1744.
- [22] Dong, Mianxiong, et al., "Quality-of-Experience (QoE) in emerging mobile social networks", *IEICE Transactions on Information and Systems* 97.10 (2014): 2606-2612.
- [23] Boyce, Jill M., and Alexis M. Tourapis. "Fast efficient channel change [set-top box applications]." In *2005 Digest of Technical Papers. International Conference on Consumer Electronics, 2005. ICCE.*, pp. 1-2. IEEE, 2005.
- [24] Dufaux, Frederic, Béatrice Pesquet-Popescu, and Marco Cagnazzo. "Emerging technologies for 3D video: creation, coding, transmission and rendering.", John Wiley & Sons, 2013.
- [25] Stankiewicz, Olgierd, Krzysztof Wegner, and Marek Domański. "Study of 3D Video Compression Using Nonlinear Depth Representation." *IEEE Access* (2019).
- [26] Ekmekcioglu, Erhan, et al. "Adaptive multi-view video delivery using hybrid networking." (2016).
- [27] Clifford De Raffaele and Carl James Debono, *Applying Prediction Techniques to Reduce Uplink Transmission and Energy Requirements in Mobile Free-Viewpoint Video Applications*, University of Malta, IEEE, *Second International Conferences on Advances in Multimedia*, 2010.
- [28] Zhi Liu, Jingyun Feng, Yusheng Ji, Yongbing Zhang, EAF: Energy-aware adaptive free viewpoint video wireless transmission, *Journal of Network and Computer Applications* Volume 46, Pages 384–394, November 2014.
- [29] Domanski, M., Experiments on acquisition and processing of video for free-viewpoint television, *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, Pages 1-4, 2-4 July 2014.
- [30] Zarb, T., *Broadcasting Free-Viewpoint Television Over Long-Term Evolution Networks*, *Systems Journal, IEEE (Volume:PP , Issue: 99)*, Pages 1-12, 29 July 2015.
- [31] Naim, Muhammad Ahsan, and Yu Zhou. "System and methods for providing multiple voice over IP service modes to a wireless device in a wireless network." U.S. Patent No. 9,565,231. 7 Feb. 2017.
- [32] Han, Junghee, Changryul Kim, and Kailash Joshi. "Analysis of Internet protocol television (IPTV) evolution in Korea: An open innovation perspective." *Journal of Information Technology Case and Application Research* 17.2 (2015): 93-107.
- [33] Tian, Shishun, Lu Zhang, Luce Morin, and Olivier Déforges. "Performance comparison of objective metrics on free-viewpoint videos with different depth coding algorithms." In *Applications of Digital Image Processing XLI*, vol. 10752, p. 107520O. International Society for Optics and Photonics, 2018.
- [34] Van Ma, Linh, Gwanghyun Yu, Jin-Young Kim, Yonggwon Won, and Jinsul Kim. "An efficient transmission method based on HEVC multi-view adaptive video streaming over P2P network in NFV." *The Journal of Supercomputing* 74, no. 12 (2018): 6939-6959.