Computers, Materials & Continua CMC, vol.65, no.3, pp.2571-2589, 2020

Bandwidth-Efficient Transmission Method for User View-Oriented Video Services

Minjae Seo[1] and Jong-Ho Paik[2, *]

Abstract: The trend in video viewing has been evolving beyond simply providing a multi-view option. Recently, a function that allows selection and viewing of a clip from a multi-view service that captures a specific range or object has been added. In particular, the free-view service is an extended concept of multi-view and provides a freer viewpoint. However, since numerous videos and additional data are required for its construction, all of the clips constituting the content cannot be simultaneously provided. Only certain clips are selected and provided to the user. If the video is not the preferred video, change request is made, and a delay occurs during retransmission from the server. Delays due to frequent re-requests degrade the overall quality of service. For free-view services, selectively transmitting the video according to the user's desired viewpoint and region of interest within the limited network of available videos is important. In this study, we propose a method of screening and providing the correct video based on objects in the contents. Based on the method of recognizing the object in each clip, we designed a method of setting its priority based on information about the object's location for each viewpoint. During the transmission and receiving process using this information, the selected video can be rapidly recognized and changed. Herein, we present a service system configuration method and propose video selection examples for free-view services.

Keywords: Free-viewpoint video, multi-view video coding, scene change, object co-detection, transmission method.

1 Introduction

Owing to the development of video processing technology, camera prices have dropped, leading to the development of diverse display devices. Contents are also being provided in more realistic formats, and users can now enjoy watching clips more actively. Hence, multi-view services that provide videos from multiple viewpoints relative to the same content are being researched actively. Owing to the demand in this market where streaming is preferred over stored media, various methods for streaming these multi-view videos have been proposed.

[1] Department of Computer, Seoul Women's University, Seoul, 01797, Korea.

[2] Department of Software Convergence, Seoul Women's University, Seoul, 01797, Korea.

[*] Corresponding Author: Jong-Ho Paik. Email: paikjh@swu.ac.kr.

Received: 02 May 2020; Accepted: 04 July 2020.

Object-tracking technology was investigated while methods that efficiently stream multi-view media were researched [Lou, Cai and Li (2005)]. Assuming that the user's viewing flow moves around a specific event in the content, a technique for tracking and providing an object has been proposed [Zhang, Toni, Frossard et al. (2018)]. However, implementing this technique involves challenges, such as computational complexity and requirement for strict pre-processing procedures. Thus, it has been difficult to obtain practical implementation and application results. Unlike when the first multi-view service was proposed, the trends are moving toward selecting and watching narrow set videos covering a specific range or object by providing an overall screenshot. For example, videos such as "Fan cam," which capture a specific player in the stadium or a specific member of a group of singers on a stage, are actively uploaded and viewed on various service platforms, including YouTube. Research on how to request a clip based on the movement in the content is increasingly important.

Recently, a service called the free-viewpoint video (FVV) has emerged. FVV is an extended concept of multi-view video [Hamza and Hefeeda (2016)]. It not only provides pre-recorded multi-view media but also creates and provides intermediate views between originally separate views through view synthesis. In addition, FVV supports zoom in–zoom out features at each view, thus allowing free and realistic viewing.

However, FVV requires many clips and a large amount of additional data for service configuration. Therefore, multiple problems must be solved before being able to stream a large-capacity FVV service smoothly. Currently, simultaneously providing all videos that comprise the content is impossible; only certain videos can be provided to the users at one time. Furthermore, if the provided video is not the video preferred by the user, the user requests another one, which causes transmission delay. For example, assume that a user wants to watch a soccer game. If they attempt to move the point of view along a specific team or player's movement, the view display should follow that movement. Delays from frequent re-requests degrade the user's overall quality of service (QoS). For FVV service users, transferring videos selectively is essential according to the user's desired view flow within the limited network of available videos.

In this study, we propose a method of providing video based on objects in the scene while achieving a smooth viewing flow for the user. Object co-detection recognizes objects in a video, even if they are viewed from different angles. Objects in each clip can be recognized, and the priority of the clip to be provided can be set according to the recognition rate of the object located from each viewpoint. This object co-detection technique is referenced as a standard for selecting a clip. Herein, we present a service system configuration method and propose video selection examples for FVV services.

2 Related works

2.1 Free-viewpoint video (FVV)

Free-viewpoint video is a complementary service to multi-view. While maintaining the characteristics of multi-view, which provides a variety of viewpoints, FVV supports functions such as zoom in-zoom out, thereby allowing users to enjoy more realistic and active viewing [Smolic (2011)]. The FVV streaming system is provided with an initial view clip of the content set for the user when streaming starts. At any point during playback, the

user can request a clip from a different viewing angle. The method of changing the viewpoint can be provided differently based on the user's device. For example, if the user's input device is a keyboard, then an arrow key can be used; or if using a head-mounted display (HMD), the viewpoint can be changed through head tracking.

The client on the user's side must respond quickly to a viewer switching request and be able to switch smoothly between different views. To provide the user with a natural and smooth viewpoint movement screen, the client should be able to provide as many viewpoints as possible. Since there is high interdependency between views for FVV streaming, the overall quality of service (QoS) may deteriorate if relationship information is ignored. When a client receives only one or two clips, bandwidth can be wasted. More efficient solutions are adopted by the system to capture a scene using an array of cameras and create a virtual view at the receiver using a subset of the captured views [Hamza and Hefeeda (2014)]. This enables greater efficiency while reducing the amount of data to be transmitted from the server.

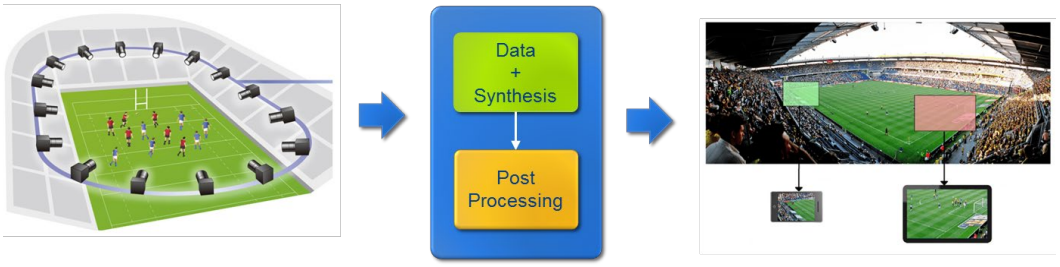

Figure 1: Free-viewpoint video

In this study, a limited set of conditions was assumed for providing FVV. The conditions of omnidirectional video are considered; this is a research area based on the main technologies of FVV. Omnidirectional video is an immersive video system (known as MPEG-I in MPEG) that limits the movement conditions from six degrees of freedom (6 DoF) to a range of several steps [Lee, Jeong, Shin et al. (2019)]. This recognizes and reduces the otherwise impossible scope of implementation during the process specified in the standardization meeting. In this study, the service considered was limited to a range similar to the 6 DoF used in sensation media over a vast range of free-viewpoint video services that are currently difficult to implement. When the service was initially proposed, there were no limitations on the scope of delivery; therefore, studies dealing with a large amount of content were conducted. However, it is difficult to implement or commercialize this, so various restrictions have been recently introduced. Studies on such services have become more specific. We used a method for providing each clip as a layer for each image in a limited zoom in-zoom out section. However, by creating a scene of overlapping sections separately, we invested effort in compensating for possible problems.

2.2 Object co-detection

Object-tracking technology has been proposed as one of the methods for streaming multi-view services [Lou, Cai and Li (2005)]. Under the assumption that the user's viewing flow will follow certain events in the content, the need to apply techniques to track and

provide objects has been emphasized. Object-tracking technology is more efficient than the conventional random transmission method because it uses the characteristics of the content; therefore, the scene in the video content can be played based on the user's interest. However, problems such as pre-processing and computational complexity must be addressed to implement object tracking within multi-viewpoint. Thus, the information necessary for the implementation phase cannot be adequately specified. Accordingly, herein, object detection technology is applied to FVV to enable object-oriented transmission. Object detection refers to a technique that detects objects based on photographs of objects taken from various angles to recognize them [Bao, Xiang and Savarese (2012)]. For object detection, various techniques have been studied for modeling to recognize information about objects in each image.

(a) Input Image #1 (b) Input Image #2

Figure 2: Object co-detection for two images [Bao, Xiang and Savarese (2012)]

The application of object co-detection is also important, but the criteria for object recognition can be used to judge the presence or absence of an object in a scene. To recognize an object in various scenes, a certain ratio of precision must be passed. In object co-detection, the average precision of objects between each clip was calculated at about 53.8% [Bao, Xiang and Savarese (2012)]. Referring to this, we considered the object to be recognized only for clips exceeding this ratio. In actual FVV, one object may be spread across multiple screens during acquisition, such as when there is an interval between cameras. This was applied because the interval between views varies during zoom in-zoom out.

The present research does not consider object tracking, which requires complex pre-processing from the image-capturing stage. Instead, the object is assumed to be recognized in the images already acquired and processed by applying the technique of detecting the object in each angle through a similar image. In this study, alongside object detection technologies, object co-detection checks whether objects in each clip are related and classifies them. If there is a set of objects observed from multiple images, the identity of each object can be set. When this is complete, the object's viewpoint transformation and movement can be read in the image, and the recognition rate for object detection can be increased.

2.3 Spatial relationship description (SRD)

To transmit with dynamic adaptive streaming over HTTP (DASH), a manifest file media presentation description (MPD) is required. MPD is an XML file that shows information about a stream. It contains addresses for each video and is encoded for each image quality.

Furthermore, through the MPD file, it is possible to achieve service without interruption by actively performing image quality conversion based on the network scenario [D'Acunto, Van den Berg, Thomas et al. (2016)]. However, this is only a consideration for image quality and does not determine where each video is presented in the display. To determine the position of the clip, MPD screen composition information, such as the spatial relationship description (SRD), has been added to the form [D'Acunto, Van den Berg, Thomas et al. (2016)]. The SRD is a document for explaining spatial relationships; it uses a structure suitable for tiled media in which individual clips constitute parts of the overall content. Spatial information is essential in FVV because regions of interest (RoIs) must be applied for zoom and panning functions to be implemented properly.

In this study, an SRD is applied to the screen composition information of the media. In addition, a scene was created considering the overlapping space while constructing the video for various image qualities. If the temporary point of view movement of the streaming service is an area where four FHD videos overlap, then there is no problem in providing the desired one; but if the point to be continuously provided requires multiple videos, then one 4K and four FHD videos are required. This is because video transmission is efficient. Creating such alternate information can also avoid synchronization problems that can occur when playing multiple videos. Through the implementation of the playback player, the zoom in–zoom out function within a single video can be displayed. Nevertheless, constructing an FVV that is realistically completely free at this point remains difficult. Object co-detection and application of the SRD in FVV are described in detail in Section 3.

3 Design of a bandwidth-efficient transmission method for user view-oriented video services

To provide an FVV service, the entire system from video acquisition to display should be organically connected owing to the nature of the content.

3.1 Free-viewpoint video service scenario

The user is provided with a service through a communication network in a fixed environment such as a home. The user's network environment can receive up to 8K content. The user intends to watch dynamic videos such as sports events or music concerts, and when watching free-view videos, the user wants to focus on specific people in a group. The service content provided to the user supports up to 8K quality when zooming out and FHD quality when zooming in. The user wants to watch 4K mid-level image quality moving continuously around a specific person while being provided with the service; then, in a specific event scene, the user wants to zoom in at FHD. The user can tolerate a certain level of delay when the screen size changes; however, a natural screen change within the same image quality is expected. The service requirements for this scenario can be defined as follows.

1. The service should be able to support watching people or events.

2. Regardless of the location within the content, the service should be able to support the zoom in-zoom out function.

3. A delay when changing screen sizes is permissible, but delays caused by movement within the same image quality should be adjustable.

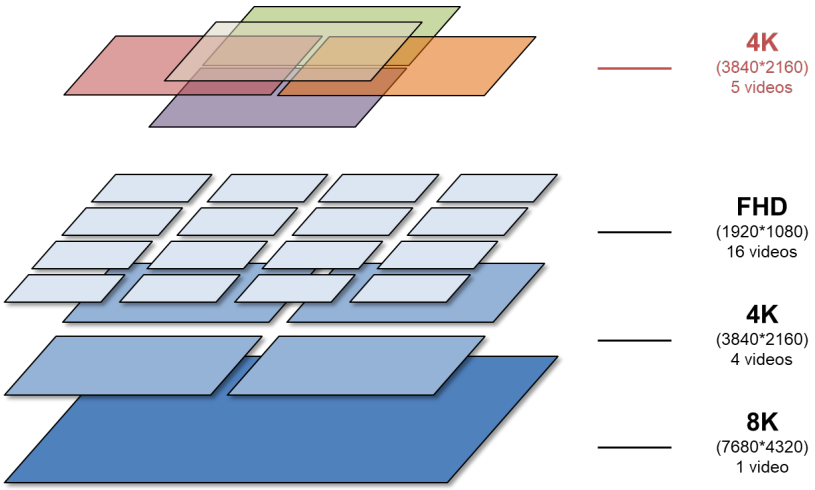

Figure 3: Video composition for FVV service

When the user watches, for example, an athletic event or a music concert, the video flow should move to the object or event that the user is watching so that the content can be enjoyed more realistically. Therefore, the service must be able to move along with object movement. In this study, it is assumed that object co-detection is applied. In addition, to provide the FVV service, the zoom in-zoom out function should be included; hence, RoIs must be supported. A scalability function can be applied while mapping videos of various image qualities by location. Furthermore, to reduce the delay for size changes and movement with respect to image quality, the system is designed such that request-related videos can be additionally provided in advance based on object position.

To provide scalability, we designed a layer for each image quality in the video, as shown in Fig. 3. It consists of 16 FHD videos for zoomed-in video and four 4K videos for zoomed out. However, the object of interest may not be located at the center of the four videos. To show the video centering on the object, we create five additional videos. Even considering constant overlap, it was judged that it is more efficient to generate multiple videos and present them as one than to receive two 4K videos and show an intermediate viewpoint.

In addition, to reduce delays with size changes and movement with regard to image quality, the system is designed to request that related clips be provided additionally in advance according to the object's position. We considered a method of providing information according to each segment for cases that stream videos through services such as DASH.

3.2 Video combination for transmission

The efficiency in transmission bandwidth obtained by dividing and providing the video content is significant, as shown by a simple calculation. The bandwidth available to

implement the proposed service is based on one 8K video constituting the entire screen. Since about 100 Mbps are required to transmit 8K video, it is assumed that the provided environment has a maximum of 100 Mbps. Since it requires approximately 112-128 Mbps to send 16 FHD videos, the entire FHD video content clearly cannot be transmitted. Eventually, a rule is needed for selecting only certain videos to send. However, four divided 4K videos can be transmitted, but this may exceed the transmission capacity. In this case, the transmission may not be performed properly; this may damage the QoS of the entire service.

In this study, three types of transmission are proposed. There may be a way to receive one 4K video and transmit all 12 videos surrounding it. This corresponds to a method of receiving videos characterized as up, down, left, and right when receiving 4K video corresponding to a viewpoint located in the middle. There are combinations that offer two 4K videos and four videos on either side. This is a method of transmitting the eight extra videos when watching a clip in the middle of two 4K videos; this may arise, for instance, if changing the view from side to side. Finally, we can consider how to receive three 4K videos and four FHDs. The division method may be determined according to a user's network situation, whether an object exists in the content, or the number of objects. Section 3.3 presents the details.

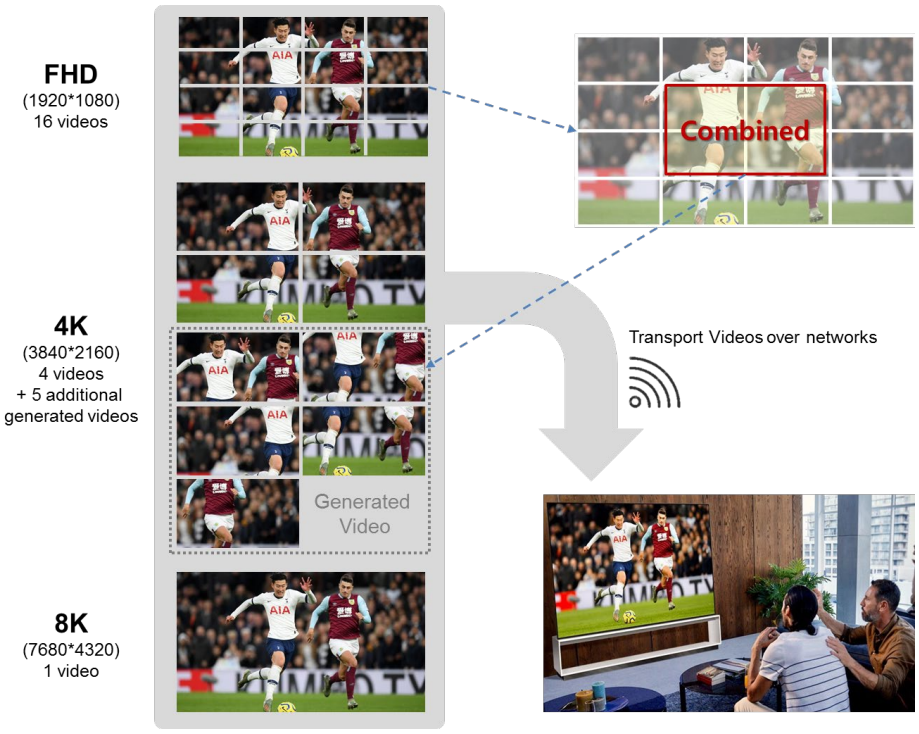

Figure 4: Video composition for the FVV service

Table 1: Video bitrate of each resolution

Resolution	Typical Bitrate	Video Count	Total Bitrate
8K video	100 Mbps	1	100 Mbps
FHD	6-7 Mbps	16	112-128 Mbps
4K video	25-30 Mbps	4	100-120 Mbps
4K video	25 Mbps	9	225 Mbps

Table 2: Proposed method of video combination

No.	Resolution	Video Count	Resolution	Video Count	Total Bitrate
1	4K video	1	FHD	12	97-114 Mbps
2	4K video	2	FHD	8	98-116 Mbps
3	4K video	3	FHD	4	99-118 Mbps

3.3 Design of a bandwidth-efficient transmission method for user view-oriented video services

In applying the configuration described in Section 3.2, a method is needed for constructing an efficient transmission to a client within the available transmission capacity. We considered changing the selection for movement in the composition within the content, as shown in Fig. 5. The yellow tile shows a scene in which the object the user wants to watch is currently located. The shift of the yellow tile shows that the selected position varies according to the movement of the object. If only the top tile corresponding to FHD image quality is considered, then the composition of the 4K video is not significantly different because the change is only between adjacent views. However, depending on the location or size of the object, 4K video tiles showing the object as the focus may also be different. In the case of zooming in, the adjacent clip related to the object is first selected based on the FHD video that the user currently wants to watch. If the entire object-related clip has been selected in FHD quality and has not exceeded the maximum transferable amount, the 4K video is selected to prepare for the zoomed-out screen selection. At this time, the 4K video selected according to the object's movement may be different.

For a detailed description, a total of three clip combinations were prepared as examples. To show various combinations, one, two, and three objects were classified. With one object, the tree is structured, as shown on the left side of Fig. 6, and the clip can be selected, as shown on the right side. The user is watching screen 6 when zoomed in to the FHD screen. In that case, tiles 6, 7, 10, and 11 are additionally transmitted to the adjacent clip that is closely related to the object using the co-detection value. Tiles 2, 3, 14, and 15 may also be selectable depending on the screen configuration or network conditions. However, Fig. 6 presents an example, and focuses on the clip that is most relevant to one object; three 4K clips corresponding to zooming out are selected accordingly. The 4K numbering can be seen as four bundles of tiles divided by an FHD tile, and the numbered tiles are tiled 2×2 based on the top left. Therefore, the 4K video in Fig. 6 is selected with 5, 7, and 9. As only one object exists in the composition of the screen and the zoomed-in

scene was not a meaningful scene, it was transmitted to ensure the zoom out capability.

Fig. 7 shows a video screen with two objects. Looking at the FHD screen, since the user is watching tile 10, object 1, located on the left, is relatively more important than object 2, located on the right. Hence, all the clips related to object 1, that is, tiles 5, 6, 9, and 10, are transmitted; only part of object 2 is transmitted. Magnification should also be considered for each object within a given bandwidth. The 4K video related to object 1, tile 6, should be selected first; the main zoomed-out 4K video for object 2, tile 8, is also selected.

Fig. 8 shows the case where there are three or more objects. As the scene being watched by the user is FHD tile 6, all tiles related to the object are selected, that is 5, 6, 9, 10, 13, and 14. Also, tiles 2, 3, 10, and 14 are additionally transmitted because another object, object 2, must also be considered. To prepare for other objects, tiles 7, 8, 11, and 12 are also selected within the remaining bandwidth. Considering object 3, we select 4K video and FHD video of object 3 within the remaining bandwidth, assuming that we selected 4K tiles 5 and 6, which are zoomed-out screens for objects 1 and 2. However, even for important objects, if the scene does not have a sufficiently meaningful value compared to the value of object co-detection, the selection may vary depending on the number of objects or network conditions. In this example, rather than selecting a related clip with a small value, a method of selecting options to ensure at least minimum viewing of each object is considered.

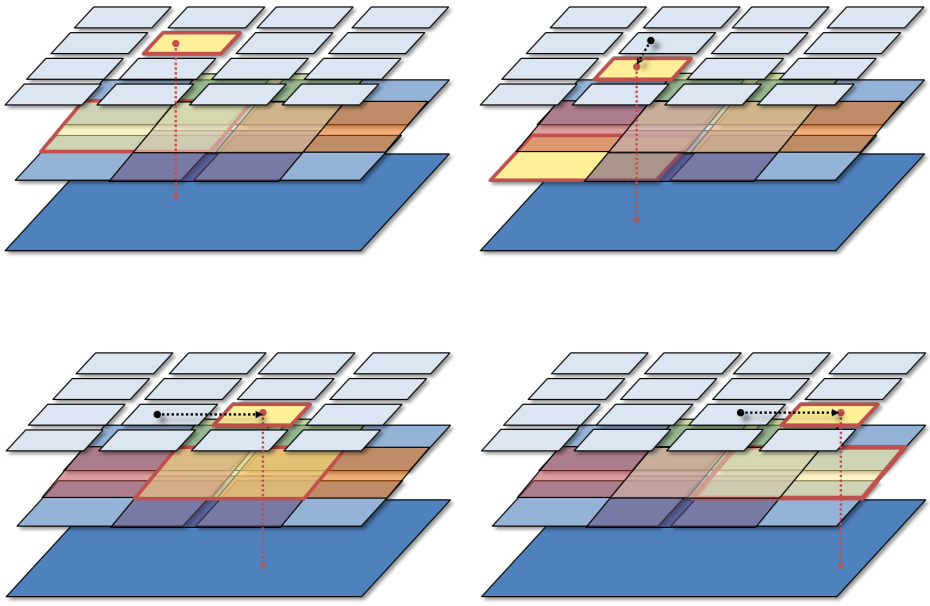

Figure 5: Object movement in a tiled video

To load information for changing the selection, considering when to update the information is also necessary. In this study, information about the video is updated and checked at intervals of 1 s. Generally, when a streaming service is provided in DASH, the video is segmented at intervals of about 4 s. However, in the case of dynamic video,

some tiles of the FHD video may be moved and changed, even for intervals of 4 s. Basically, considering 4K video and assuming that it changes frequently, the group of pictures (GoP) using 4K video should be considered. 4K UHD video is composed of about 60 frames per second. For example, to update at 0.5 s intervals such as in signaling, the change must be executed at about 30 frames per second, but I frame cannot be guaranteed. To play a new scene, I frame should be accessed first. Therefore, to match the clip playback, configuring the screen composition so that it can be changed at 1 s intervals is reasonable. Even if the length becomes longer or shorter depending on the segment configuration, applying information at intervals of 1 s is convenient. Information can be provided in various forms, but, in this study, we consider providing it with a video segment as a tree-type XML document.

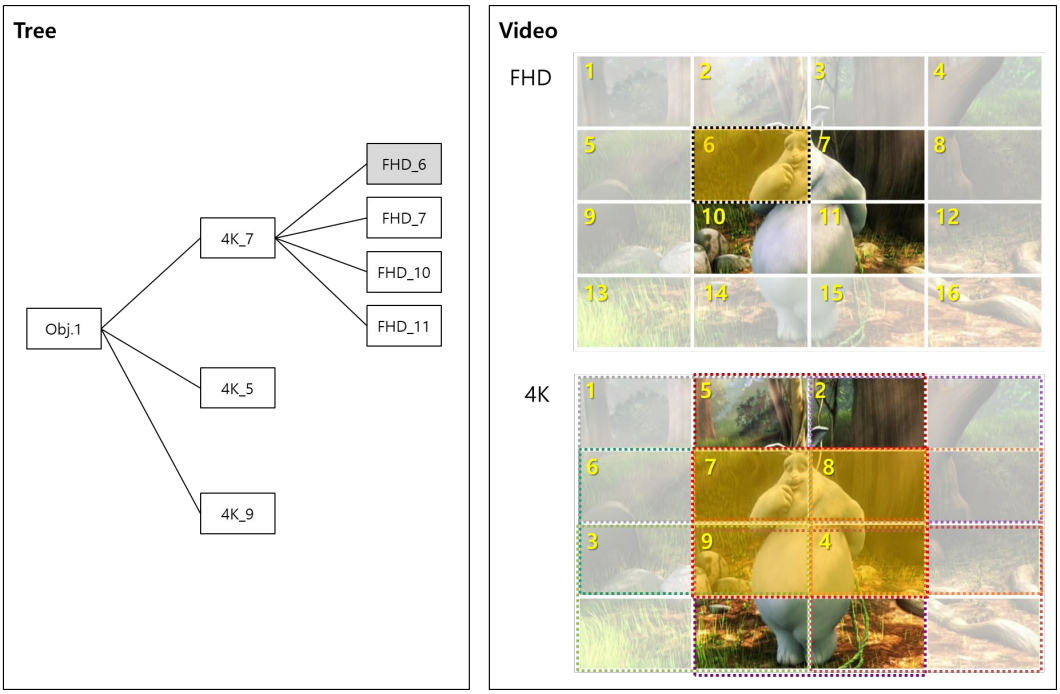

Figure 6: Example of tree structure and clip selection for one object

To provide an FVV service, covering the entire process is necessary, from clip collection to display. The free-viewpoint service system structure proposed in this study is as follows. After the clips are acquired, they are used to acquire each object's information in the image files. Each scene is treated by the object information processor. Then, via object co-detection, the objects in the media are detected and the object is recognized in another image at the same distance. As mentioned previously, we use the object co-detection method proposed in Bao's paper [Bao, Xiang and Savarese (2012)]. Therefore, through the detection obtained from each image, objects can be tracked by subsequent matching. However, for object information processing, information on the arrangement of each video is required; otherwise, the positional relationships between videos are unknown. This information can be provided through the naming rules for each clip

without processing a separate document.

Based on the information obtained through this processing, each clip is collected and encoded at 8K, 4K, and FHD quality. In each process, the video is encoded into one 8K video, nine 4K videos, and sixteen FHD videos. The 4K videos comprises of the 8K video divided into four videos. However, when considering the zoom in-zoom out function, the 4K video may be the default media quality. In case the object is located between the four 4K video intervals where each 8K screen is divided, five additional 4K videos are generated so that the object can be presented as one. Since the minimum quality of the media in the content was viewed in FHD, the FHD video is created with 16 videos filling the entire 8K. It encodes the video and encapsulates it in MP4 for transmission.

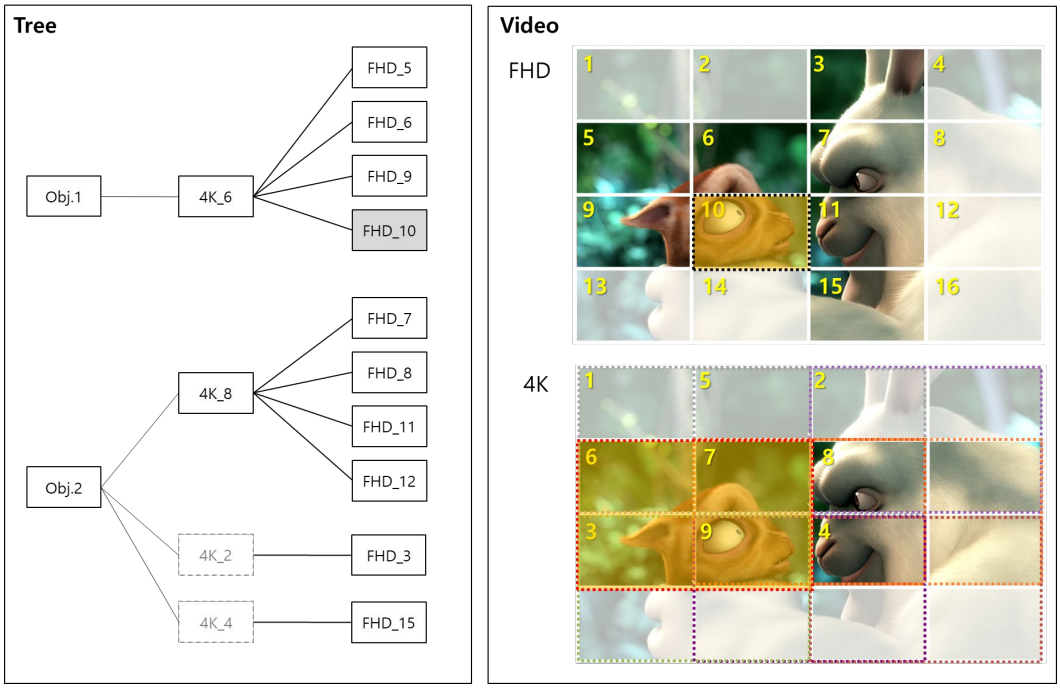

Figure 7: Example of tree structure and clip selection for two objects

An SRD document is then generated. An example of the SRD document structure is as follows. For an 8K video constituting the entire screen, the media is processed as *EssentialProperty*, and the information necessary for segmenting the rest of the media is written. *SupplementalProperty* is classified for additional information that is not essential. The information for the entire 8K screen is configured, and the ID value is set to 1 to provide configuration information for the 4K screen. This is composed of a total of nine 4K videos to provide the required identification. With the configuration of 16 FHD videos, each video is composed of a 4×4 matrix from the top left to the bottom right.

The lower part additionally shows a screen configuration that temporarily provides four FHD videos for a four-split screen. Even if the user is watching one FHD video through the zoom-in function, if the serviceable equipment and network situation provide considerably more space, then the video related to the current video is also transmitted

rather than sending only one video. When the user moves the viewpoint, it is intended to ensure the QoS of the service by allowing users to pre-select and respond to the provided files. To provide 4K video when a user zooms out in the FHD video, it is best to provide the 4K video that was previously delivered to the client. If the network condition is not good, the corresponding 4K video is requested at the time of the zoom out request. Instead, we considered how to configure the screen: first with a combination of the FHD video being provided and then with related FHD video received before the corresponding 4K video arrived. Finally, we exchange the 4K video upon arrival.

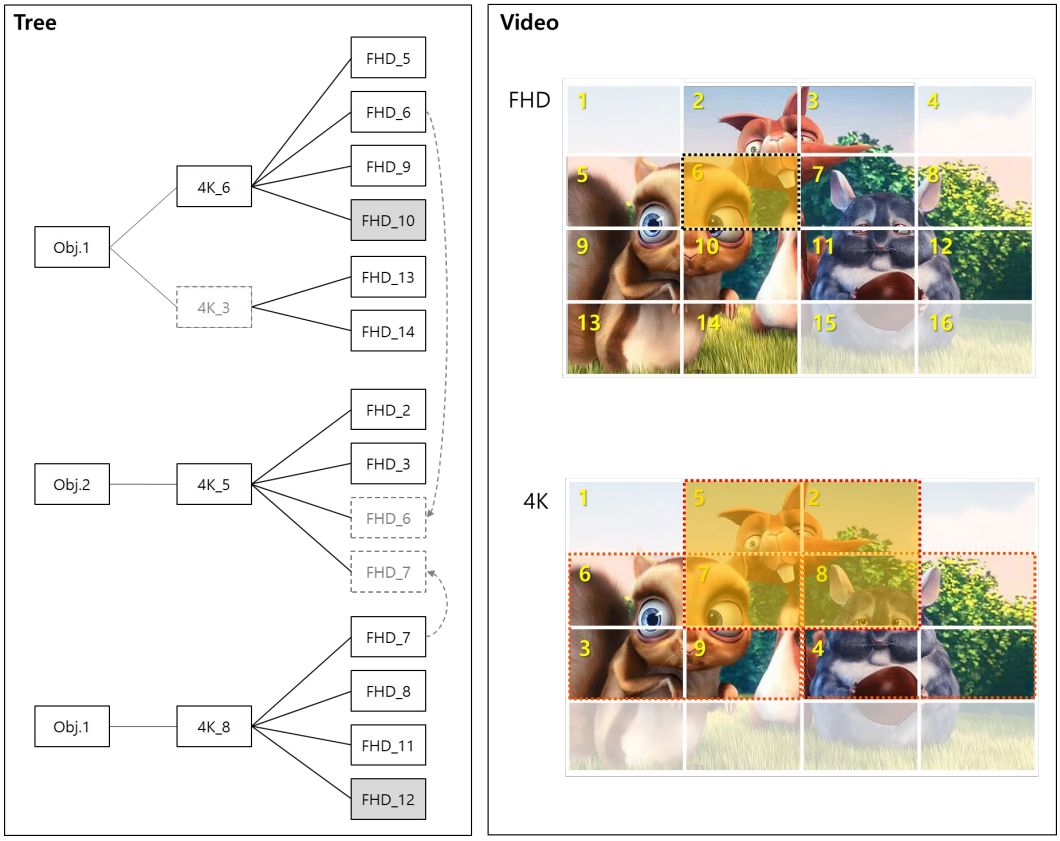

Figure 8: Example of tree structure and video selection for three or more objects

However, information about how the zoomed in–zoom out scenes are related and presented on the same screen cannot be known from the above screen composition information. This information must be mapped and included in the service. Under the premise that the service map information is streamed through a communication network, a format, such as MPEG media transport presentation information (MMT-PI), is adopted here. The MMT-PI represents scene composition information proposed in the MMT standard. MMT-PI defines the entire program for the media required in the content and can easily represent the spatial and temporal relationships between media [Paik, Seo and Yu (2019)]. Therefore, by configuring the service map information, the overall configuration of the service can be

managed. When such information is configured, media and service map information is loaded in a practical server called a service manager.

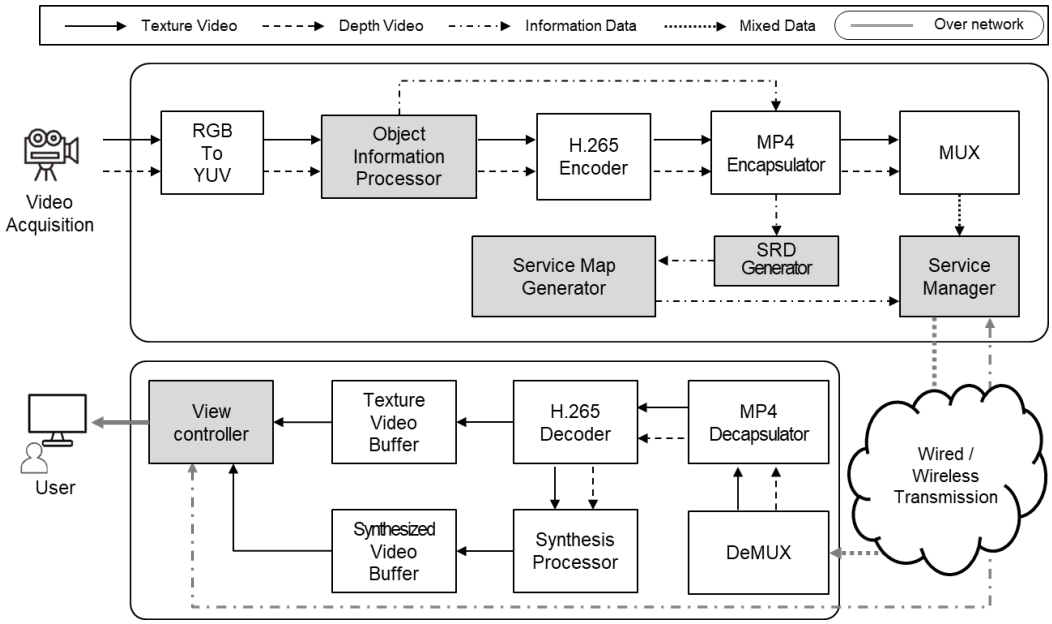

Figure 9: Proposed FVV streaming system

Meanwhile, the media and service map information provided by default is transmitted to the client device through a wired or wireless communication network. The view controller can then identify where the currently provided video is located and which scene is adjacent to the current scene through the provided service map information. The video is decoded after de-multiplexing and subsequently presented to the user. If the user requests a view that does not exist, or if synthesis is required in the process of moving the view, processing related to the synthesis is performed.

In this study, content related to synthesis in the video is not discussed in detail. With a 4K screen, compositing—a technique for providing a scene that cannot be seen at each captured time—is employed. However, since we researched the best approach for placing objects in the center of the screen and providing core scenes, we conducted the study under the assumption that the important scenes and objects were already captured. However, the range of angles may increase or decrease in the process of zooming in-zooming out. In the existing scene, additional data necessary for synthesis are provided, and until the synthesis is provided, the receiver's policy can suggest the best view in the currently provided scene. Thus, media data can be provided considering the additional data related thereto. However, there may be a case where a specific object is not at the location of a video at the time the user wants to watch. For this case, we should consider viewpoint synthesis.

```
<xml version="1.0" encoding="UTF-8" standalone="no"?>
...
<Period>
  <!-- full view contents -->
  <AdaptationSet>
       <SupplementalProperty schemeIdUri="urn:mpeg:dash:swu:2020"  value="0,0,0,7680,4320,7680,4320"/>
    <Representation id="1" width="3840" height="2160"  codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
    <BaseURL>4k_video_1.mp4</BaseURL>
    </Representation>
    ...
    <Representation id="25" width="1920" height="1080"  codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
    <BaseURL>FHD_video_16.mp4</BaseURL>
    </Representation>
  </AdaptationSet>

  <!-- part view contents -->
   <AdaptationSet>
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,0,0,3840,2160,7680,4320"/>
        <Representation id ="26" width="1920" height="1080"  codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
     <BaseURL>FHD_video_1.mp4</BaseURL>
    </Representation>
    .....
    <Representation id ="34" width="1920" height="1080"  codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
     <BaseURL>FHD_video_11.mp4</BaseURL>
    </Representation>
  </AdaptationSet>
  ......
  <AdaptationSet>
     <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,3840,2160,7680,4320,7680,4320"/>
       <Representation id ="54" width="1920" height="1080" codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
     <BaseURL>FHD_video_6.mp4</BaseURL>
    </Representation>
    .....
    </SegmentList>
    </Representation>
    <Representation id ="62" width="1920" height="1080" codecs="hevc" mimeType="video/mp4" startWithSAP="0" >
     <BaseURL>FHD_video_16.mp4</BaseURL>
    </Representation>
  </AdaptationSet>
</Period>
```

Figure 10: Example of the proposed SRD structure

4 Experimental results

To confirm the results of configuring the service, as suggested in this study, we conducted an experiment. We tested the performance of the method based on available 8K images. The example was a video of a group of singers where the lighting of the background may change slightly, but the main change is in the movement of the members of the group. The video was close to being a dynamic video. We attempted to extract object information for each person and apply a video request.

Figure 11: Example of the test video

Table 3: Video encoding information

GoP Size	FPS	CRF	Total Frame Count	I Frame Count
30	60	24	7264	42

Table 4: Average bitrate for each resolution

Resolution	Average Bitrate
8K (7680×4320)	96 Mbps
4K (3840×2160)	31 Mbps
FHD (1920×1080)	8 Mbps

The content was encoded according to Tab. 3 and each video was encoded in 8K, 4K, and FHD. The average bit rate for each resolution was as specified in Tab. 4. As this work assumed that the environment can support up to 100 Mbps, the encoding specification was adjusted to support 8K accordingly. We used FHD quality for the zoomed-in image, 8K for the zoomed-out image, and supported video in the default 4K image.

To clearly show the results of the experiment, each member of the group was extracted as an object, but the experiment was conducted with the most identifiable characters. Based on Fig. 12, when the user selects tile 5 for viewing, FHD content of tile 5 was provided. In addition, FHD images, including objects 9 and 13, were provided. In addition, one scene corresponding to a 4K video starting from tile 5 and one 4K video starting from tile 6, including all the remaining objects, were provided. Additionally, if FHD was available among tiles 6, 7, 10, and 11 in the remaining band, an additional one was provided to receive a total of 94 Mbps. In this case, efficient viewing was possible rather than providing all of tiles 1, 2, 3, 4, 8, 12, 14, 15, and 16.

On the other hand, the case of an image where the objects are not collected can be confirmed in Fig. 13. In this example, it was decided to extract the object located in the same video: tile 8. FHD video provides tiles 8, 12, and 16 as standards. Subsequently, a 4K image starting at tile 7 is additionally selected, and a 4K image starting at tile 5 is also selected. In the case of the object located at tile 5, the probability of selection is very low in this study. However, to guarantee the presence of an object as much as possible, the remaining two objects can be viewed with 4K video, and the video related to tile 6 can be guaranteed through tiles 5 and 7 (4K video). If the user selects an object other than tile 8, this is a change that occurs when the selected object is changed. Hence, the video is requested with a slight delay.

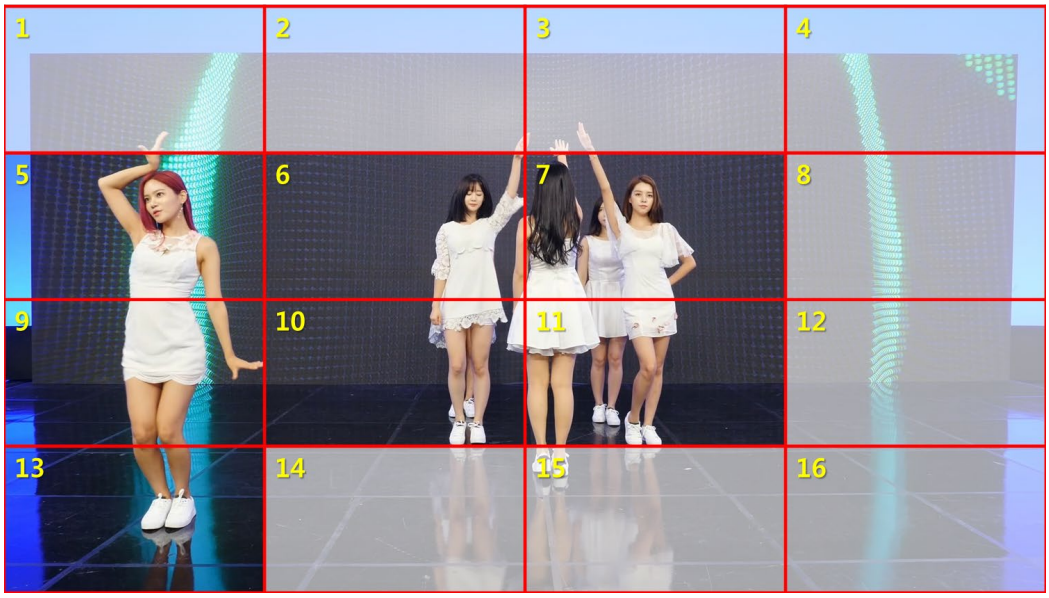

Figure 12 : Example of video applying our methods

Figure 13 : Example of video applying our methods

When the object-related information was provided, the video was selected as described above so that the video about the object was guaranteed as much as possible. In this way, since the video was prepared in advance, it was confirmed through experiments that there was no delay when moving the viewpoint related to the selected object.

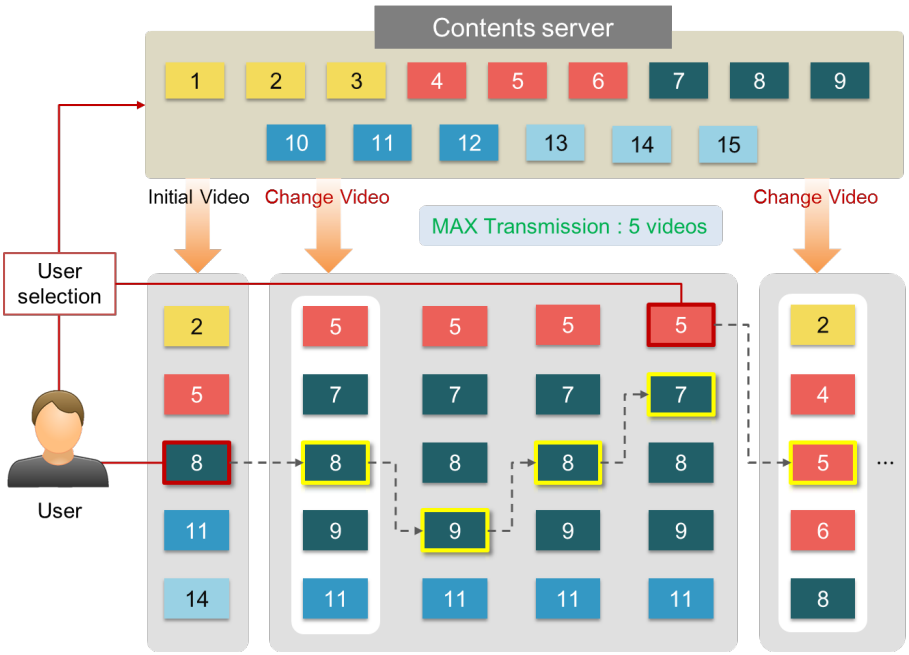

Figure 14 : FVV video reception expectation scenario

Fig. 14 shows the expected results when applying the proposed technique. It can be assumed that there are 15 videos in total, and three videos each are related to each object. In this case, it is important when the user selects a specific object after initially providing individual object information. According to the conventional method, to view a video related to an object, additional video requests and responses must be made even though the viewpoint is adjacent. However, we were able to ensure that the content that the user focused on was guaranteed to the maximum in the bandwidth, and that resulted in stable results.

5 Conclusions

In this study, we proposed a method of screening and providing video based on objects so that the user's viewing flow can move naturally. Object co-detection was referenced as a standard to apply the function of screening the video through an object-based approach. This made it possible to recognize the objects present in each clip and to set the transmission priority of each scene according to the recognition rate of the objects located at each viewpoint. Additionally, by adding information related to objects in the video, the video can be rapidly recognized and changed during the transmitting and receiving processes. We proposed the video service system configuration method and the structure of the manifest document for use with this approach and presented an example of the expected results attained through estimation.

The proposed method does not save much in terms of transmission volume, but it prioritizes and provides videos based on when the user wants to receive them. Therefore, at least the video of interest at the current time can be reliably guaranteed, subject to the network status. Through which, more natural viewpoint movement can be achieved. As a

future study, the service system will be implemented practically, and the effectiveness of the proposed design will be confirmed through experimental results. We will also proceed with experiments investigating what priority should be applied and guaranteed when using additional data up to the synthesis time point. Through this future research, it is expected that smoother and more natural FVV services will be possible.

Acknowledgment: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1F1A1061635) and by a research grant from Seoul Women's University (2020-0213).

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Bao, S. Y.; Xiang, Y.; Savarese, S. (2012): Object co-detection. *European Conference on Computer Vision*, pp. 86-101.

Chen, Y. T.; Xia, R. L.; Wang, Z.; Zhang, J. M.; Yang, K. et al. (2019): The visual saliency detection algorithm research based on hierarchical principle component analysis method. *Multimedia Tools and Applications*, https://doi.org/10.1007/s11042-019-07756-1.

D'Acunto, L.; Van den Berg, J.; Thomas, E.; Niamut, O. (2016): Using MPEG DASH SRD for zoomable and navigable video. *Proceedings of the 7th International Conference on Multimedia Systems*, pp. 1-4.

Gui, Y.; Zeng, G. (2020): Joint learning of visual and spatial features for edit propagation from a single image. *The Visual Computer*, vol. 36, no. 3, pp. 469-482.

Hamza, A.; Hefeeda, M. (2014): A DASH-based free viewpoint video streaming system. *Proceedings of Network and Operating System Support on Digital Audio and Video Workshop*, pp. 55-60.

Hamza, A.; Hefeeda, M. (2016): Adaptive streaming of interactive free viewpoint videos to heterogeneous clients. *Proceedings of the 7th International Conference on Multimedia Systems*, pp. 1-12.

He, S. M.; Xie, K.; Xie, K. X.; Xu, C.; Wang, J. (2019): Interference-aware multisource transmission in multiradio and multichannel wireless network. *IEEE Systems Journal*, vol. 13, no. 3, pp. 2507-2518.

Lee, G. S.; Jeong, J. Y.; Shin, H. C.; Seo, J. I. (2019): Standardization trend of 3DoF+ video for immersive media. *Electronics and Telecommunications Trends*, vol. 34, no. 6, pp. 156-163.

Lee, J. M.; Lee, J. H.; Lim, J. Y.; Kim, M. R. (2019): Bandwidth-efficient live virtual reality streaming scheme for reducing view adaptation delay. *KSII Transactions on Internet and Information Systems*, vol. 13, no. 1, pp. 291-304.

Li, Y.; Yang, G. B.; Zhu, Y. P.; Ding, X. L.; Song, Y. et al. (2019): Hybrid stopping model-based fast PU and CU decision for 3D-HEVC texture coding. *Journal of Real-Time Image Processing*, https://doi.org/10.1007/s11554-019-00876-9.

Lou, J. G; Cai, H.; Li, J. (2005): A real-time interactive multi-view video system. *Proceedings of the 13th annual ACM international conference on Multimedia*, pp. 161-170.

Niamut, O. A.; Thomas, E.; D'Acunto, L.; Concolato, C.; Denoual, F. et al. (2016): MPEG DASH SRD: spatial relationship description. *Proceedings of the 7th International Conference on Multimedia Systems*, pp. 1-8.

Paik, J. H.; Seo, M.; Yu, K. A. (2019): Design and implementation of transmission scheduler for terrestrial UHD contents. *Journal of Broadcast Engineering*, vol. 24, no. 1, pp. 118-131.

Smolic, A. (2011): 3D video and free viewpoint video-from capture to display. *Pattern Recognition*, vol. 44, no. 9, pp. 1958-1968.

Xiang, L. Y.; Shen, X. B.; Qin, J. H.; Hao, W. (2019): Discrete multi-graph hashing for large-scale visual search. *Neural Processing Letters*, vol. 49, no. 3, pp.1055-1069.

Zhang, D. Y.; Liang, Z. S.; Yang, G. B.; Li, Q. G.; Li, L. D. et al. (2018): A robust forgery detection algorithm for object removal by exemplar-based image inpainting. *Multimedia Tools and Applications*, vol. 77, no. 10, pp. 11823-11842.

Zhang, X.; Toni, L.; Frossard, P.; Zhao, Y.; Lin, C. (2018): Adaptive streaming in interactive multiview video systems. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 1130-1144.