**3GPP SA4#116-eS4-211611**

**10-19 Nov 2021**

|  |
| --- |
| *CR-Form-v12.0* |
| **Pseudo CHANGE REQUEST** |
|  |
|  | **26.862** | **CR** | **<CR#>** | **rev** | **-** | **Current version:** | **0. 0.3** |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | [ITT4RT] Requirement for occlude-free areas in 360 video |
|  |  |
| ***Source to WG:*** | Tencent |
| ***Source to TSG:*** | SA4 |
|  |  |
| ***Work item code:*** | ITT4RT |  | ***Date:*** | 2021-11-14 |
|  |  |  |  |  |
| ***Category:*** | **B** |  | ***Release:*** | Rel-17 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)Rel-12 (Release 12)**Rel-13 (Release 13)Rel-14 (Release 14)Rel-15 (Release 15)Rel-16 (Release 16)* |
|  |  |
| ***Reason for change:*** | Adds a feature for improving the user experience |
|  |  |
| ***Summary of change:*** | Adding a requirement to signal the important areas of the 360 video that should not be covered by overlay or other 2-d images or video by the receiver during the rendering |
|  |  |
| ***Consequences if not approved:*** | Key topic not addressed |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  | **X** |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
| ***56***  |  |
| ***This CR's revision history:*** |  |

**===== CHANGE 1 =====**

# 5 Requirements

* Multiple single-user participants are supported.
* Communications between the single users can be conventional MTSI/Telepresence communications. MSMTSI could be used, and if that is used, then media data can be transmitted in separate media streams, and the layout of different participants is up to the client application/implementation.
* One 360 camera per location in multi-party conference scenarios involving multiple physical locations are allowed.
* Both in-camera stitching and network-based stitching are supported.
	+ In case of camera stitching, stitched immersive video is sent from the conference room to the conferencing server (e.g., MSMTSI MRF or any other media gateway) and then from the conferencing server to the remote participants. If this is a one-to-one conversational session between the conferencing room and the remote participant, a media gateway in the middle may not be necessary.
	+ In case of network-based stitching, different 2D captures are sent from the conference room to the conferencing server and the conferencing server performs decoding, stitching, and re-encoding to produce the immersive video, which is then distributed to the remote participants.
* It is recommended that MTSI and IMS Telepresence endpoints support codec, protocol and transport capabilities relevant for encoding, delivery and consumption of immersive speech/audio and immersive video.
* Capability for the party that sends 360-degree video to send viewport-dependent and/or viewport-independent streams.
* Timely delivery of the changes in viewport orientation from the remote participants, and appropriate low-delay actions to update the viewport-dependent streams. Any changes in viewport orientation should not lead to latency-prone signalling, such as SIP renegotiations.
* Capability to create viewport-dependent streams for individual UEs including an larger area of the original viewport for safe playback in the UE.
* A suitable coordinate system to be used as the standard way of communicating the orientation of the viewport.
* Given possible end device limitations as well as potential constraints on the conference room equipment, network-based processing should be considered for media workloads involving both conference room and remote participants, e.g., stitching of captured streams from the conference room, media composition, transcoding and prerendering for the remote participant, etc.
* The following parameters need to be signalled in the SDP during call setup in addition to normal MTSI call signaling [1].
	1. Initial viewport orientation. It is the default orientation from which to start the view at the receivers’ side.
	2. Decoding/Rendering metadata, e.g., region-wise packing information, projection mapping information, frame packing information, etc. It is subject of discussion whether this information is signaled via SDP and/or within SEI messages with the media stream.
	3. Capture Field-of-View (CFoV): as discussed during the definition of the use case, the system supports transmission of 360-degree video. However, the range of the FoV may be restricted in order to enhance user experience. The negotiation requires signaling the capture FoV of the capture device, and a response carrying the receiver’s preferred FoV (PFoV) depending on the remote UE, where the preferred FoV will be less than or equal to the captured FoV.
	4. Codec negotiation

The high level signaling flows are depicted in Figure 4.1. The user C is not represented here for simplicity, but this is not a restriction for our reasoning. In this example MRF/MCU is not used.

 

Figure 5.1 Signalling flow for a 360-degree conference call with unidirectional 360- degree video from A to B.

* Once the call has been established, remote parties (B or C) can send viewport orientation information using RTCP reports with yaw, pitch and roll data. These may be sent at fixed intervals or event-based, triggered by changes in viewport orientation. or hybrid combination of fixed interval and event-based triggers. When hybrid reporting scheme is used, the event-based feedback is triggered by any changes in viewport whereas the regular RTCP interval provides the sender with a regular update, in fixed intervals, of the viewport even if the event-based feedback is not received. The most efficient RTCP reporting scheme for viewport orientation information is for further study.
* Capability to support the interaction where all media types will be presented to certain users and a subset of the media types are presented to the others.
* Capability for the participant in room A with his or her own display device to receive a viewport independent or viewport dependent video from omnidirectional camera in room A.
* Capability for the remote party to share a viewport dependent video stream with embedded viewport metadata to another remote participant.
* Capability for the participant in room A with his or her own display device to follow remote participant viewport presentation.
* The capability to place overlays in the 360-degree video either within the device or pre-rendered through a network element.
* Transmission from sender to receiver of the coordinates of the location of the overlay (e.g. a presentation): this is a necessary and basic requirement that will give flexibility in the overlay placement at the receiver’s side. By sender/receiver it is meant either one of the parties or the MRF/MCU.
* Avoid that the overlaid background content is transmitted unnecessarily at high quality within the user viewport: this is a basic issue that overlays cause to viewport-dependent streaming. The content in the viewport is always streamed at higher quality. However, when an overlay with different content is sticked on top of (part of) the viewport, the content behind the overlay does not need to be sent ay higher quality. This allows saving bandwidth or increase the quality in the viewport for the non overlaid parts.
* Enable some form of interaction with the overlay (e.g., moving or rotating the overlay, resizing it, switching it on/off, etc.): these are basic and simple ways to interact with the overlay, to increase flexibility and utility of an overlay.
* Capability for users to receive an incoming interaction message (e.g., SMS, chat message, voice call or audio-visual call) from other users as an overlay: this is a good way to allow integration of other 3GPP services and applications into ITT4RT/MTSI applications in order to increase the value of the first VR applications for 3GPP.
* To facilitate network-based stitching, it is possible to signal camera calibration parameters for each 2D video capture (i.e., each camera lens) transported from the conference room to the conferencing server at the beginning of each session. Relevant intrinsic and extrinsic camera parameters can include lens numbers, layouts, positions, angles, radius, distortion, entrance pupil and resolutions.
* An RTP receiver should be able to signal higher-level metrics such as Motion to High-Quality Delay to the sender to assist in bandwidth adaptation and monitoring.
* Allow still background images to be used when network conditions do not permit transmitting a video stream for the area outside the viewport.
* A sender can offer support for conditional overlays in session signalling and receiver can understand it.
* To activate/deactivate conditional overlays, a receiver is capable to signal one or more regions and their associated conditions using RTCP feedback/SDP to the sender during a media session.
* Capability to identify the location of the presentation and where to insert an overlay of the alternative presentation into the omnidirectional content:
* while stitching the different camera images together in the network
* after the stitching of the camera images into the omnidirectional video (e.g. by reencoding the omnidirectional video in the network)
* after receiving the stitched omnidirectional video and the overlay by the receiving client
* If the overlay is represented in video format, it shall be delivered over RTP. If the overlay is represented in formats other than video (e.g., images), then the overlay may be delivered considering the following formats and protocols:
	+ the overlay may be delivered using the WebRTC data channel (SCTP/DTLS) [15], which is currently already defined for both MTSI in TS 26.114 and IMS-based Telepresence in TS 26.223.
	+ the overlay may be represented according to the HEVC image format as specified in Annex B of [26]. Accordingly HEIF image items and image sequences are described as regular HEVC media streams. It should be possible to use the HEVC Payload format as described in RFC 7798 for the delivery of the HEVC images, image collections, and image sequences (details TBD). All NAL units of an HEVC image stored as meta items are extracted and transmitted as HEVC access units with the same presentation time stamp.
* Allow a sender to signal the regions of its 360 video is considered as important foreground (like head and shoulder of the participants in the room) that should not be completely or partially occluded at the receiver by rendering one or more overlay images or videos on top of them. The provided information is used when a receiver is viewing its desired viewport of the 360 video and likes to render the overlays and other 2-d videos from this or other senders on top of the view where the default location of the overlay is outside of the viewport. The sent information describes one or more regions that are not supposed to be occluded by other visual media and the location, size, and number of these regions may need to be updated due to the change of the number of participants, or their location in the 360 video during the video conferencing session.