**3GPP TSG SA WG4#127 S4-240083**

**Sophia-Antipolis, France, 29th Jan- 2nd Feb 2024**

|  |
| --- |
| *CR-Form-v12.2* |
| **CHANGE REQUEST** |
|  |
|  | **26.998** | **CR** | **0003** | **rev** | **-** | **Current version:** | **18.0.0** |  |
|  |
| *For* ***[HE](http://www.3gpp.org/3G_Specs/CRs.htm%22%20%5Cl%20%22_blank)******[LP](http://www.3gpp.org/3G_Specs/CRs.htm%22%20%5Cl%20%22_blank)*** *on using this form: comprehensive instructions can be found at <http://www.3gpp.org/Change-Requests>.* |
|  |

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

|  |
| --- |
|  |
| ***Title:***  | [FS\_5GSTAR] Updates on Media Formats in AR Scenes of TR 26.998 |
|  |  |
| ***Source to WG:*** | China Mobile Com. Corporation |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** | FS\_5GSTAR, TEI18 |  | ***Date:*** | 2024-01-28 |
|  |  |  |  |  |
| ***Category:*** | C |  | ***Release:*** | Rel-18 |
|  | *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-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** | In current TR 26.998, it only includes explicit 3D media formats for AR content, such as meshes, point-clouds, and voxel grids. Recently, implicit neural representation formats (e,g., NeRF, SDF) have been proposed as alternatives to describe 3D objects and scenes. |
|  |  |
| ***Summary of change:*** | Adds Implicit Neural Representation format for AR content |
|  |  |
| ***Consequences if not approved:*** | Incomplete AR content formats |
|  |  |
| ***Clauses affected:*** | 4.4.4 |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** | **X** |  |  Other core specifications  | TR 26.926 CR ...  |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

# Introduction

# Implicit Neural Representations (INRs) is a novel method for 3D objects or scenes representation (see this document for more details: <https://www.cvlibs.net/publications/Peng2020ECCV_slides.pdf>).

For example, Google's latest research on SMERF(Streamable Memory Efficient Radiance Fields for Real-Time Large-Scene Exploration) demonstrates fully 6DoF navigation within a web browser, and renders real-time on smartphones and laptops.



**===== CHANGE =====**

### 4.4.4 Media Formats/Primitives in AR Scenes

An AR/MR object may be represented in a form of 2D media. One camera or one view frustum in a scene may return a perspective planar projection of the volumetric scene. Such a 2D capture consists of pixels with colour attributes (e.g., RGB).

Each pixel (a) may represent a measure of the distance between the surface of an AR object, point (A) and the camera centre. Conventionally, the distance is represented by the coordinate of the point on the z-axis obtained by the orthogonal projection of the point (A) on this axis, here denoted as the point (A’). The measured distance is thus the length of the segment (CA’) as depicted in Figure 4.4.4-1.



Figure 4.4.4-1: Pixel representation of depth images

This convention is used for commercially available frameworks handling depth images such as the Microsoft Azure KinectTM SDK [7] and the Google ARCoreTM [8]. According to the documentation of the Azure KinectTM SDK, the depth sensor uses the Time-of-Flight (ToF) technique to measure the distance bewteen the camera and a light-reflecting point in the scene. The documentation further specifies that “these measurements are processed to generate a depth map. A depth map is a set of Z-coordinate values for every pixel of the image, measured in units of millimeters”. Similarly, the Google ARCoreTM documentation explains that “when working with the Depth API, it is important to understand that the depth values are not the length of the ray CA itself, but the projection of it” onto the z-axis.

Additionally, sensor API may provide the image from the viewpoint of the depth sensor which is thus not aligned with the viewpoint of RGB camera which is necessarily few millimetres away due to physical constraints. In this case, an alignment operation is necessary in order to guarantee the correspondence between a pixel of the depth image and a pixel of the RGB picture. For instance, the Azure Kinect SDK provides the k4a\_transformation\_depth\_image\_to\_color\_camera() and k4a\_transformation\_color\_image\_to\_depth\_camera() functions which generate a depth image aligned with the colour picture and a colour image aligned with the depth image, respectively. More details and illustrations are provided in [9].

A depth map thus contains pixels with the distance attribute (e.g., depth). Distance is one-dimensional information and may be represented in an absolute/relative or linear/non-linear manner. Metadata to explain the depth map may be provided.

The capturing of a volumetric scene may also be expressed as an omnidirectional image in a spherical coordinate system. Equirectangular Projection (ERP) is an example of projection methods to map a spherical coordinate system into a cylindrical coordinate system. The surface of the cylindrical coordinate system is considered as 2D media.

Capturing of a volumetric scene may be further improved/elevated with hundreds of cameras in an array; High Density Camera Array (HDCA) or lenticular are methods to capture rays of light. Each point on surface of a volumetric scene has countless rays of colours in multiple different directions. Each position of a camera captures a different colour from the same point surface of the volumetric scene. 2D images from the camera array may be packed together to form a larger plenoptic image.

From another perspective, 2D media is the output of the immersive media renderer. One view frustum that represents the user’s viewport is placed in a scene, and in turn, a perspective or an orthogonal projection of the volumetric media may be produced. To minimise motion sickness, a pose corrector function performs a correction of the 2D media at the last stage of presentation. The pose corrector may require additional information such as the estimated or measured user pose that was used for the rendering of the 2D media. For the case that the latest user pose does not match with the estimated user pose, additional information that provides knowledge on the geometry, such as a depth map, may be delivered from immersive media renderer.

Immersive media may be considered as an AR/MR object and may be used to provide an immersive experience to users. The immersive experience may include a volumetric presentation of such media. The volumetric presentation does not bind to a specific display technology. For example, a mobile phone may be used to present either the whole AR media, or a part of the AR media. Users may see a volumetric presentation of a part of the AR media augmented in real space. Therefore, immersive media includes not only volumetric media formats such as omnidirectional visual formatsERP image, 3D meshesPrimitives, point cloudsPrimitives, light fieldsPlenopotic image, scene description, and 3D audio formats, but also 2D video2D image as studied in TR 26.928.

Immersive media also includes Implicit Neural Representations (INRs) formats, such as Signed Distance Function (SDF), Neural Radiance Field (NeRF) and Occupancy Field. These are novel methods for describing 3D objects or scenes representation as continuous and differentiable functions (see this document for more details: [https://www.cvlibs.net/publications/Peng2020ECCV\_slides.pdf).](https://www.cvlibs.net/publications/Peng2020ECCV_slides.pdf%29.)

NeRF, or Neural Radiance Fields, extends the INR into rendering field. Utilizing Multilayer Perceptrons (MLP) and implicit geometric representations such as distance functions, NeRF interprets input images from various perspectives as a set of sampled points. The model takes a five-dimensional vector as input (three-dimensional coordinates along with camera position and orientation), the model outputs include the color and density of each sampled point. By incorporating rendering loss from input images, the MLP weights are continually optimized during training. A trained MLP can take input from any viewpoint and generate color and density outputs, resulting in a complete, high-resolution image.

- Formats for 2D media

Still image formats may be used for 2D media. The 2D media may have metadata for each image or for a sequence of images. For example, pose information describes the rendering parameter of one image. The frame rate or timestamp of each image are typically valid for a sequence of such images.

- Primitives

3D meshes and point clouds consists of thousands and millions of primitives such as vertex, edge, face, attribute and texture. Primitives are the very basic elements in all volumetric presentation. A vertex is a point in volumetric space, and contains position information in terms of three axes in coordinate system. In a Cartesian coordinate system, X, Y, and Z make the position information for a vertex. A vertex may have one or more attributes. Colour and reflectance are typical examples of attributes. An edge is a line between two vertices. A face is a triangle or a rectangle formed by three or four vertices. The area of a face is filled by interpolated colour of vertex attributes or from textures.

In contrast to traditional discrete representations (e.g., 3D meshes and point clouds), INRs leverage a neural network as the continuous mapping function between the signal domain, such as coordinates (e.g., pixel coordinates in an image), and arbitrary positions at that coordinate, representing, for instance, color channels (R, G, B) in the case of images. One advantage of using INRs is their capability to have arbitrary dimensions and resolutions. Additionally, the memory needed to parameterize the signal remains independent of spatial resolution. It scales only with the complexity of the underlying signal, making INRs more memory-efficient.