|  |  |
| --- | --- |
| 3GPP TR 26.956 V0.1.1(2024-11) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Evaluation and Characterization of Beyond 2D Video Formats and Codecs  (Release 19) | |
|  | |
| *5G-logo_175px* | 3GPP-logo_web |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

[Foreword 6](#_Toc21451)

[Introduction 7](#_Toc7683)

[1 Scope 8](#_Toc27557)

[2 References 8](#_Toc23757)

[3 Definitions of terms, symbols and abbreviations 10](#_Toc7767)

[3.1 Terms 10](#_Toc28961)

[3.2 Symbols 10](#_Toc28478)

[3.3 Abbreviations 11](#_Toc7017)

[4 Beyond 2D Video Formats 11](#_Toc17808)

[4.1 Introduction 11](#_Toc29387)

[4.2 Reference Model for Beyond 2D Video 11](#_Toc6986)

[4.2.1 Overview 11](#_Toc10645)

[4.3 Beyond 2D Video Representation Formats 12](#_Toc13260)

[4.3.1 Introduction 12](#_Toc20626)

[4.3.2 Stereoscopic Video 13](#_Toc30702)

[4.3.2.1 Definition 13](#_Toc14591)

[4.3.2.2 Production and Capturing Systems 14](#_Toc32716)

[4.3.2.3 Rendering and Display Systems 15](#_Toc16148)

[4.3.2.4 Supporting Information 15](#_Toc28963)

[4.3.2.5 Benefits and Limitations 15](#_Toc19060)

[4.3.2.5.1 Benefits 15](#_Toc29301)

[4.3.2.5.2 Limitations 16](#_Toc5951)

[4.3.3 Dense Dynamic Point Cloud representation format 16](#_Toc26731)

[4.3.3.1 Definition 16](#_Toc13434)

[4.3.3.2 Production and Capturing Systems 16](#_Toc22279)

[4.3.3.3 Rendering and Display Systems 17](#_Toc19173)

[4.3.3.4 Support Information 18](#_Toc19877)

[4.3.3.4.1 Test and reference sequences 18](#_Toc6024)

[4.3.3.4.2 Uncompressed data size 18](#_Toc15333)

[4.3.3.4.3 Known compression technology 18](#_Toc8045)

[4.3.3.4.4 Conversion from other formats 18](#_Toc16106)

[4.3.3.4.5 Typical quality criteria 18](#_Toc3683)

[4.3.3.5 Benefits and Limitations 19](#_Toc12697)

[4.3.3.5.1 Benefits 19](#_Toc201)

[4.3.3.5.2 Limitations 19](#_Toc24502)

[4.3.4 Multi-view video representation 19](#_Toc14504)

[4.3.4.1 Definition 19](#_Toc31511)

[4.3.4.2 Production and Capturing Systems 21](#_Toc168)

[4.3.4.3 Rendering and Display Systems 22](#_Toc11080)

[4.3.4.4 Supporting Information 22](#_Toc21412)

[4.3.4.4.1 Camera placement 22](#_Toc4934)

[4.3.4.4.2 Spatial resolution 23](#_Toc21996)

[4.3.4.4.2 Objective metrics 23](#_Toc17995)

[4.3.4.4.3 Coding and delivery options 24](#_Toc21868)

[4.3.4.5 Benefits and Limitations 24](#_Toc11196)

[4.3.4.5.1 Benefits 24](#_Toc28766)

[4.3.4.5.2 Limitations 24](#_Toc5045)

[4.3.X Formats under Research 24](#_Toc15914)

[4.3.X.1 Neural Radiance Fields 24](#_Toc2249)

[4.3.X.1.1 Introduction 24](#_Toc29856)

[4.3.X.1.2 Definition 24](#_Toc4336)

[4.3.X.1.3 Production and Capturing Systems 25](#_Toc5046)

[4.3.X.1.4 Rendering and Display Systems 27](#_Toc24248)

[4.3.X.1.5 Supporting Information 27](#_Toc13412)

[4.3.X.1.6 Benefits and Limitations 27](#_Toc15672)

[4.3.X.1.6.1 Benefits 27](#_Toc15206)

[4.3.X.1.6.2 Limitations 27](#_Toc5201)

[5 Overview of existing "Beyond 2D" Video Capabilities in 3GPP 28](#_Toc30428)

[5.1 Introduction 28](#_Toc4164)

[5.2 AR Video Capabilities 28](#_Toc9766)

[5.3 VR Video Profiles 29](#_Toc3747)

[5.4 Messaging Services 29](#_Toc9585)

[6 Evaluation and Characterization Framework 30](#_Toc17448)

[6.1 Overview 30](#_Toc715)

[6.2 Reference Sequences 30](#_Toc27136)

[6.3 Reference Software Tools 30](#_Toc12472)

[6.4 Metrics 30](#_Toc30330)

[6.5 Encoding Constraints 31](#_Toc27300)

[7 Considered Scenarios 31](#_Toc17480)

[7.1 Introduction 31](#_Toc16802)

[7.2 Scenario 1: <tbd> 31](#_Toc25779)

[7.3 Scenario 2: <tbd> 31](#_Toc15442)

[7.4 Scenario 3: <tbd> 31](#_Toc31851)

[7.x Scenario x: <tbd> 31](#_Toc2803)

[8 Common Evaluation Features 31](#_Toc18393)

[9 Evaluation of Selected Scenarios 31](#_Toc6362)

[9.1 Introduction 31](#_Toc2857)

[9.2 Scenario 1: <tbd> 31](#_Toc20213)

[9.2.1 Evaluation Overview 31](#_Toc4955)

[9.2.2 Reference Sequences 32](#_Toc14073)

[9.2.3 Performance Metrics 32](#_Toc22812)

[9.2.4 Candidate Solutions 32](#_Toc28013)

[9.2.4.1 Solution 1: <Name> 32](#_Toc6973)

[9.2.4.1.1 Introduction 32](#_Toc184)

[9.2.4.1.2 Reference Software 32](#_Toc13817)

[9.2.4.1.3 Parameter Settings 32](#_Toc3534)

[9.2.4.1.4 Distribution 32](#_Toc25972)

[9.2.4.1.5 Evaluation Results 32](#_Toc12095)

[9.2.4.1.6 Network Requirements 32](#_Toc21779)

[9.2.4.2 Solution 2: <Name> 32](#_Toc14765)

[9.2.4.2.1 Introduction 32](#_Toc10900)

[9.2.4.2.2 Reference Software 32](#_Toc22766)

[9.2.4.2.3 Parameter Settings 32](#_Toc29472)

[8.2.4.2.4 Distribution 32](#_Toc31605)

[9.2.4.2.5 Evaluation Results 32](#_Toc18856)

[9.2.4.2.6 Network Requirements 32](#_Toc9642)

[9.2.5 Summary of Evaluation 32](#_Toc2368)

[9.3 Scenario 2: <tbd> 32](#_Toc21852)

[9.4 Scenario x: <tbd> 32](#_Toc13153)

[10 Gaps and Optimization Potential 33](#_Toc7541)

[10.1 Identified Gaps and Deficiencies with Video Capabilities 33](#_Toc12603)

[10.2 Potential Requirements for New Video Capabilities 33](#_Toc9365)

[10.3 Potential Network Optimizations 33](#_Toc4212)

[11 Conclusions and Proposed Next Steps 33](#_Toc17730)

[Annex A: Scenario Template 33](#_Toc7946)

[A.1 Introduction 33](#_Toc17607)

[A.2 Template 33](#_Toc10045)

[Annex B: Data Formats and Metrics 35](#_Toc32263)

[B.1 Introduction 35](#_Toc21402)

[B.2 Raw Video Sequences 35](#_Toc1983)

[B.2.1 Overview 35](#_Toc17576)

[B.2.2 JSON Schema 36](#_Toc23059)

[Annex C: Reference Sequences 38](#_Toc27852)

[C.1 Introduction 38](#_Toc21858)

For definitive guidance on drafting 3GPP TSs and TRs, see [3GPP TS 21.801](http://www.3gpp.org/DynaReport/21801.htm) supplemented by the 3GPP web page <http://www.3gpp.org/specifications-groups/delegates-corner/writing-a-new-spec>.

Ensure all blue guidance text is removed before submitting the TS/TR to the TSG for approval.

# Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# Introduction

In recent years, video services are evolving from traditional two-dimensional formats to beyond 2D video, which offer users a more lifelike and immersive experience. Research studies indicate that the beyond 2D market was valued at approximately multi-million USD in 2023 and is anticipated to register a CAGR (Compound Annual Growth Rate) of over 24.5% between 2024 and 2032 [2][3][4].

A variety of beyond 2D video formats and video compression technologies are available and emerging. Therefore, in order to determine appropriate beyond 2D video formats for different services, it is essential to evaluate their feasibility and performance, considering implementation constraints, performance indicators, and interoperability considerations. In addition, advanced network capabilities and service extension also need to be investigated to meet the delay and data rate requirements of beyond 2D-related services.

This document provides an overview of available and emerging beyond 2D video formats and compression technologies, which are mostly related to specific types of capturing systems and display technologies; documents a set of end-to-end reference scenarios and workflows for beyond 2D video; analyzes 3GPP-defined video compression technologies and potential new technologies to support each documented scenario; identifies gaps and offer recommendations to potentially extend 3GPP video specifications and capabilities.

# 1 Scope

The present document collects beyond 2D video formats within 3GPP services, as well as a set of beyond 2D video end-to-end reference scenarios and corresponding workflows. It also documents relevant implementation constraints, performance characteristics, and interoperability requirements of existing 3GPP codecs as well as potentially new codecs to support these scenarios. [The primary scope of the present document includes the following aspects:

1. Identify and document beyond 2D formats, that are market-relevant within the next few years, generated from established and emerging capturing systems (including cameras for spatial video capturing), contribution, and usable on display technologies (smartphones, VR HMDs, AR glasses, autostereoscopic and multiscopic displays).

2. Establish and document a set of beyond 2D video end-to-end reference scenarios, including real-time communication, streaming services, split rendering, and messaging and corresponding workflows (capturing, encoding, packaging, delivery, decoding, rendering, including general constraints on latency, as well as complexity) to support 3GPP network related delivery and devices leveraging the generation or display technologies. This includes identifying and defining relevant beyond 2D formats in the context of above workflows, and representation technologies to support delivery of these formats within 3GPP networks.

3. Prioritize the scenarios and the associated formats based on market relevance for further evaluation.

4. Define concrete evaluation framework per scenario (test conditions, KPIs, Metrics, test sequences, agreed reference signals) based on the above prioritized reference scenarios, and evaluate the feasibility and performance of existing 3GPP codecs as well as potentially new codecs to support the scenarios.

5. Based on the findings in steps 1, 2, and 4 document (i) interoperability requirements, (ii) traffic characteristics and (iii) potential QoS optimizations or requirements, to support the above workflows and evaluate the feasibility of new formats with different services, considering the implementation constraints and performance indicators such as encoding, decoding, and rendering complexity, bandwidth utilization, and interoperability considerations.

6. Based on the findings in steps 1, 2, 4 and 5, identify potential gaps or deficiencies of existing 3GPP codecs, and offer recommendations to potentially extend 3GPP video specifications and capabilities.

1. Identify potential areas for normative work as the next phase and communicate with other 3GPP WGs regarding relevant aspects related to the study to the extent needed.]

Editor’s note: The scope may be updated as study progressed.

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] Allied Market Research, “3D Technology Market Size, Share, Competitive Landscape and Trend Analysis Report by Product, Application : Global Opportunity Analysis and Industry Forecast, 2021-2030.”, [www.alliedmarketresearch.com/3d-technology-market.](http://www.alliedmarketresearch.com/3d-technology-market.)

[3] Mordor Intelligence, “Mobile 3D Market Size & Share Analysis - Growth Trends & Forecasts (2024 - 2029).”, <https://www.mordorintelligence.com/industry-reports/mobile-3d-market.>

[4] Grand View Research, “Immersive Technology Market Size, Share & Trends Analysis Report By Component (Hardware, Software, Services), By Technology, By Application, By Industry, By Region, And Segment Forecasts, 2023 - 2030.”, [https://www.grandviewresearch.com/industry-analysis/immersive-technology-market-report.](https://www.mordorintelligence.com/industry-reports/mobile-3d-market.)

[5] 3GPP TS 26.119: "Media Capabilities for Augmented Reality".

[6] 3GPP TS 26.118: "Virtual Reality (VR) profiles for streaming applications".

[7] 3GPP TS 26.143: "Messaging Media Profiles".

[8] 3GPP TS 26.511: "5G Media Streaming (5GMS); Profiles, codecs and formats".

[9] 3GPP TR 26.966: " Evaluation of new HEVC coding tools".

[10] 3GPP TS 26.265: "Media Delivery: Video Capabilities and Operating Points".

[11] 3GPP TR 26.955: "Video codec characteristics for 5G-based services and applications".

[S1] Apple HEVC Stereo Video - Interoperability Profile (Beta), Version 0.9, June 21, 2023, <https://developer.apple.com/av-foundation/HEVC-Stereo-Video-Profile.pdf>

[S2] Mike Swanson, "Spatial Video", March 7 2024, https://blog.mikeswanson.com/spatial-video/

[S3] Video Contour Map Payload, Version 0.9, June 21, 2023, <https://developer.apple.com/av-foundation/Video-Contour-Map-Metadata.pdf>

[S4] ITU-T H.273 (09/23), Coding-independent code points for video signal type identification

[S5] M. Satya, "3D Image Reconstruction From Multi-View Stereo", <https://medium.com/@satya15july_11937/3d-image-reconstruction-from-multi-view-stereo-782e6912435b>, March, 2023.

[D1] Greg Turk, The Polygon File Format, Stanford University, 1994.

[D2] Volumetric Format Association VFA, <https://www.volumetricformat.org/>

[D3] V-PCC, Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC), ISO/IEC 23090-5 2nd Ed, Nov 2023.

[D4] G-PCC, Geometry-based point cloud compression, ISO/IEC 23090-9, Mar 2023

[D5] Draco Bitstream Specification, <https://google.github.io/draco/spec/>

[N1] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2021. NeRF: representing scenes as neural radiance fields for view synthesis. Commun. ACM 65, 1 (January 2022), 99–106. https://doi.org/10.1145/3503250

[N2] Gao, Kyle et al. “NeRF: Neural Radiance Field in 3D Vision, A Comprehensive Review.” (2022).

[N3] Li, Sicheng et al. “NeRFCodec: Neural Feature Compression Meets Neural Radiance Fields for Memory-Efficient Scene Representation.” ArXiv abs/2404.02185 (2024): n. pag.

[N4] Dong-Ha Kim, Jun Young Jeong, Gwangsoon Lee, and Jae-Gon Kim "Compression method of NeRF model using NNC and VVC", Proc. SPIE 13164, International Workshop on Advanced Imaging Technology (IWAIT) 2024, 131642V (2 May 2024); https://doi.org/10.1117/12.3019533

[N5] G. Lafruit, Y. Liao, and G. Bang, “AhG on Implicit Neural Video Representations (INVR),” ISO/IEC JTC1/SC 29/WG04, M60641, Oct. 2022.G. Lafruit, Y. Liao, and G. Bang, “AhG on Implicit Neural Video Representations (INVR),” ISO/IEC JTC1/SC 29/WG04, M60641, Oct. 2022

[N6] RABBY, AKM SHAHARIAR AZAD and Chengcui Zhang. “BeyondPixels: A Comprehensive Review of the Evolution of Neural Radiance Fields.” ArXiv abs/2306.03000 (2023): n. pag.

[N7] Daniel Duckworth, Peter Hedman, Christian Reiser, Peter Zhizhin, Jean-François Thibert, Mario Lučić, Richard Szeliski, and Jonathan T. Barron. 2024. SMERF: Streamable Memory Efficient Radiance Fields for Real-Time Large-Scene Exploration. ACM Trans. Graph. 43, 4, Article 63 (July 2024), 13 pages. https://doi.org/10.1145/3658193

[N8] Müller, T., Evans, A., Schied, C., & Keller, A. (2022). Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG), 41(4), 1-15..

[R1] OpenCV, <https://opencv.org>.

[R2] Colmap, <https://colmap.github.io/index.html>.

[R3] AliceVision Photogrammetric Computer Vision Framework, <https://alicevision.org>.

[R4] Open Multiple View Geometry (openMVG), <https://github.com/openMVG/openMVG>.

[R5] Immersive Video Depth Estimation (IVDE), <https://gitlab.com/mpeg-i-visual/ivde>.

[R6] Test model for MPEG immersive video, <https://gitlab.com/mpeg-i-visual/tmiv>.

[R7] Reference view synthesizer, <https://gitlab.com/mpeg-i-visual/rvs>.

[R8] Open Realtime Depth Image Based Renderer (OpenDIBR), <https://github.com/IDLabMedia/open-dibr>.

[R9] A. Dziembowski, D. Mieloch, J. Stankowski and A. Grzelka, "IV-PSNR – the objective quality metric for immersive video applications," in IEEE Transactions on Circuits and Systems for Video Technology, doi: [10.1109/TCSVT.2022.3179575](https://doi.org/10.1109/TCSVT.2022.3179575" \t "_blank), software: <https://gitlab.com/mpeg-i-visual/ivpsnr>

[R10] Quality Metrics for Immersive Video (QMIV), <https://gitlab.com/mpeg-i-visual/ivpsnr>.

# 3 Definitions of terms, symbols and abbreviations

This clause and its three subclauses are mandatory. The contents shall be shown as "void" if the TS/TR does not define any terms, symbols, or abbreviations.

## 3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

Definition format (Normal)

**<defined term>:** <definition>.

**example:** text used to clarify abstract rules by applying them literally.

## 3.2 Symbols

For the purposes of the present document, the following symbols apply:

Symbol format (EW)

<symbol> <Explanation>

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

<ABBREVIATION> <Expansion>

# 4 Beyond 2D Video Formats

## 4.1 Introduction

Editor’s note: This clause documents beyond 2D video formats that are market-relevant within next few years.

## 4.2 Reference Model for Beyond 2D Video

### 4.2.1 Overview

In contrast to well-established 2D-based video formats and work flows, for beyond 2D video a variety of emerging formats and reference workflows are under discussion. This aspect makes it more difficult to harmonize specific interop points and formats, also taking into account new developments in the industry and in research. In addition, without systematic and explicit identification of format interop points, beyond 2D scenarios or workflows may look overly complex.

However, basing beyond 2D workflows and scenarios on 2D reference workflows and formats, as for example evaluated in TR 26.955 [11] and extending existing workflows seems to be promising way forward. However, when comparing for example to TR 26.955 [11] for 2D formats or even omnidirectional video formats as defined in TS 26.118 [6], additional aspects may need to be considered for beyond 2D video. To help the situation, a generic reference model for beyond 2D video content is introduced in this sub-clause. This systematic and accurate identification of interoperability points and subcomponents for Beyond 2D video with a high level of abstraction covers the majority of use cases and scenarios.

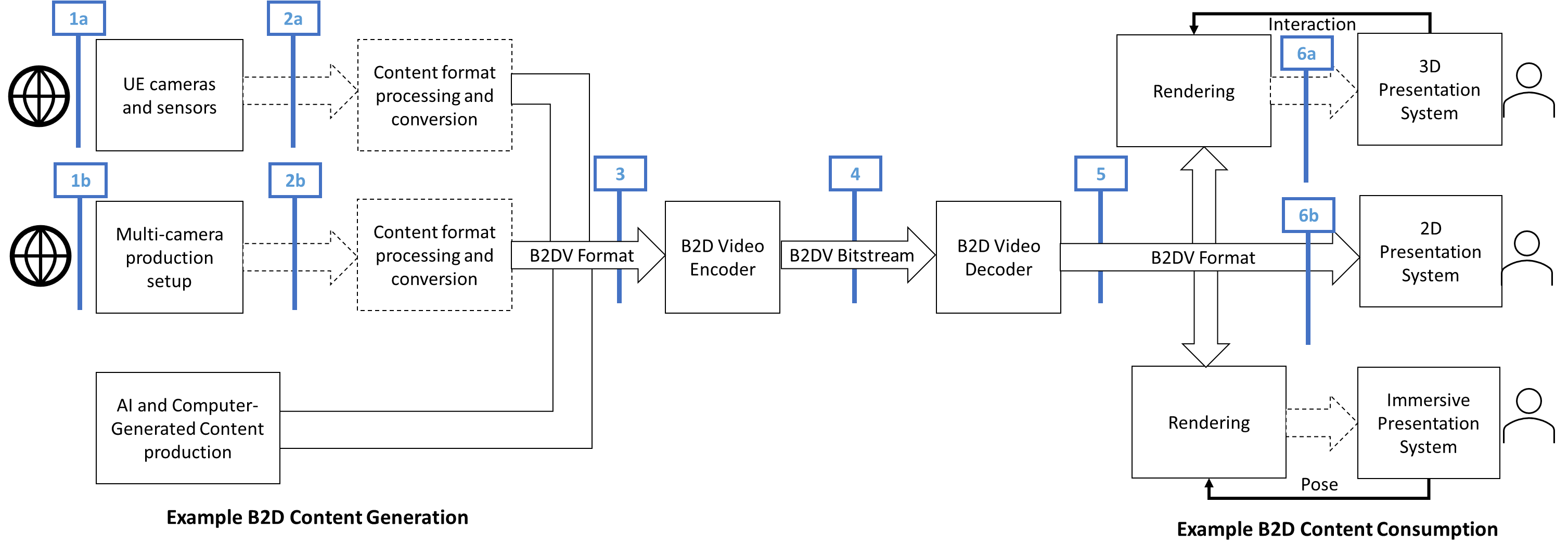


Figure 4.2-1 Beyond 2D Video end-to-end reference model

Figure 4.Y.1-1 illustrates a generic beyond 2D Video end-to-end reference model. For example, it considers three methods of creation of source content. The first apply a naturalistic way to capture sources indicated in reference point (1) and includes for example variants of UE-based cameras and sensor (1a) or a multi-camera production setup (1b). The third option is based on authoring using computer graphics interfacing technologies or other media production technologies. These may sometimes be combined and, possibly with slight variations, these options cover the majority of media production cases.

The capture of content using cameras, for example light fields using sensors is generalized including traditional *passive sensors*, cameras, camera arrays, or plenoptic cameras. For simplicity we also include *active sensors* LiDaR, Time of Flight in this category. These active sensors also transmit a signal before capturing the reflections. Depending on the setup, the collected data may be quite different depending on the capturing system, expressed in reference point (2) with variants (2a) and (2b).

In the general case, some processing based on the captured data would happen to generate a well defined B2DV format, possibly referred to as *sensed data converter*. This step is mainly about converting the multiple digital image formats plus metadata to a well-defined beyond 2D representation or format, referenced with reference identifer (3).

For typical 5G workflows, a compressed digital representation of the B2DV is needed for efficient transmission. The *B2DV Encoder* executes a processing step that will result in the compressed Beyond 2D video bitstream that includes a digitally compressed lossy version of the B2DV format and optional metadata, referred to as reference point (4). The B2DV bitstream is typically delivered through content delivery protocols and 5G radio systems, not shown in Figure 4.2.1-1.

The *B2DV decoder* decodes the B2DV video bitstream and recovers a B2DV format, presented in reference point (5). The recovered signal is forwared to the rendering and display system. In some cases, one viewport of the B2DV format may for example be displayed directly in a 2D Presentation System. In a 3D Presentation System, interacting with the rendering component may allow to generate different views on the content. In an immersive presentation system, pose information may be used to render the views of the content. The format generated by the renderer for the presentation system, indicated in reference point (6), is implementation specific as shown in Figure 4.2.1-1.

Generally, beyond 2D video performance measurement should typically be between interop points (3) and (5) based on the B2DV formats. The last block in the diagram includes the user interactions. Some B2DV scenarios may involve some types of user interactions, such as changing the viewpoint or other interactions. These are captured in the reference diagram for completeness.

## 4.3 Beyond 2D Video Representation Formats

Editor’s note: The documented format should be referenced by at least one scenario in clause 5, and the following aspects may be provided.

### 4.3.1 Introduction

As shown in Figure 4.2.1, beyond 2D video representation formats may originate from different production systems and have to target different rendering systems. This clause collects relevant Beyond 2D Video representation formats and provides a discussion on the relevancy of the formats. In order to assess the relevancy of the formats, for each format different aspects are collected, among others:

- Definition of the format, this is preferably backed by a specification.

- Typical applications of the format, e.g. knowledge about support of the format in workflows (tools, etc.)

- Production options of the representation format

- Rendering of the representation format

- Benefits and limitation of the format

- Supporting information

- Typical quality criteria for evaluating the format

- Existing test and reference sequences

- Conversion from other formats (lossless, lossy)

- Uncompressed data size

- Known compression technologies

- Extensibility of the format

### 4.3.2 Stereoscopic Video

Editor’s Note: Some additional references need to be extracted.

#### 4.3.2.1 Definition

Stereoscopic video presents one image to the user’s left eye and another image (typically correlated) to the user’s right eye to produce the stereopsis effect, defined as "the perception of depth produced by the reception in the brain of visual stimuli from both eyes in combination; binocular vision." [S1].

Stereoscopic video will be defined in TS 26.265 [26265]. However, at this stage it is not yet clear if extensions including camera parameters, depth, etc. will be part of the initial release. It is considered worthwhile to document stereoscopic video in the TR as a beyond 2D format and collect additional metadata that may be useful to support improved rendering.

Stereoscopic video may use projections to left and right eye as follows [S2]:

- rectangular, traditional 3D

- spherically-projected 3D video as defined in TS 26.118 [6].

- either of the two may be extended with additional depth data, also referred to as video contour maps [S3].

In addition, the detailed signal properties of the video each eye needs to be defined:

- Sample aspect ratio for each eye, defined according to the ITU-T H.273 [S4], SampleAspectRatio. Typical parameters are 1:1 (value 1) or 4:3 (value 14).

- Picture aspect ratio for each eye. Typical parameters are 1:1 or 16:9.

- Resolutions per eye of left eye and right eye are

- for picture aspect ratio 1:1: 1080x1080, 1440x1440, 2160x2160, 4320x4320

- for picture aspect ratio 16:9: 1280x720, 1440x1080 (with sample aspect ratio 4:3), 3840x2160, 7680x4320

NOTE: 8K resolution is supported in TS 26.118 [6], and also supported in terms of decoding on modern mobile systems-on-chip. Whether 8K is supported in a full end-to-end workflow is application dependent, but withr appropriate capability negotation, a suitable resolution can be determined.

- Framerates for each eye are: 30 fps, 50fps, 60 fps, 90 fps, 120 fps, 144 fps and possibly fractional variants.

NOTE: 120 and 144 fps are supported in terms of decoding on modern mobile systems-on-chip. Whether such high-frame rates supported in a full end-to-end workflow is application dependent, but withr appropriate capability negotation, a suitable resolution can be determined.

- Signal characteristics

- The video signal is YUV with 4:2:0 chroma subsampling.

- Bit depth: 8 or 10 bits

- Colour primaries, defined according to the ITU-T H.273 [S4], ColourPrimaries. Typical parameters are BT-709 (value 1), and BT-2020/BT-2100 (value 9).

- Transfer characteristics, defined according to the ITU-T H.273 [S4], TransferCharacteristics. Typical parameters are BT-709 (value 1), BT-2020 (value 14), BT-2100 PQ (value 16) and BT-2100 HLG (value 18).

- Matrix coefficients, defined according to the ITU-T H.273 [S4], MatrixCoefficients. Typical parameters are BT-709 (value 1), and BT-2020/BT-2100 non-constant luminance (value 9).

- Typical combined values are BT-709 SDR with (1,1,1), HDR PQ with (9,16,9) and HDR HLG with (9,18,9).

- Projection parameters:

- Projection: rectilinear, fisheye, equirectangular

- Field-of-view and restricted coverage.

NOTE: The parameters may be aligned with TS 26.118 [6]

Additional metadata may be present, either on a static or per frame basis, as follows:

- hero eye: A value that indicates which eye is the primary eye when rendering in 2D.

- camera parameters: camera parameters are typically represented in a 3 × 4 projection matrix called the camera matrix. The extrinsic parameters define the camera pose (position and orientation) while the intrinsic parameters specify the camera image format, specifically:

- extrinsic parameters denote the coordinate system transformations from 3D world coordinates to 3D camera coordinates. For details see: https://en.wikipedia.org/wiki/Camera\_resectioning#Extrinsic\_parameters

- intrinsic parameters describe a specific camera model. These parameters encompass focal length, image sensor format, and camera principal point. For details see: https://en.wikipedia.org/wiki/Camera\_resectioning#Intrinsic\_parameters

- disparity adjustment:

- horizontal disparity adjustment, a value that indicates a relative shift of the left and right images, which changes the zero-parallax plane.

- Disparity/depth map: 10bit, same resolution as source content, monochrome, can possibly be sub-sampled

- Line time (per camera) – rolling shutter readout time, only relevant in poorer quality/reduced functionality camera pipelines typically used on HMD tracking cameras.

- Examples: https://github.com/MPEGGroup/FileFormatConformance/tree/m62054\_exintrinsics/data/file\_features/under\_consideration

#### 4.3.2.2 Production and Capturing Systems

The formats as defined in clause 4.3.2.1 may be captured at least with a reduced set of parameters by mobile devices and Head Mounted Displays (HMD) – for more details refer to the following information:

- <https://techcrunch.com/2023/12/11/apple-releases-spatial-video-recording-on-iphone-15-pro/>

- Spatial Video with 1080p at 30fps

- <https://9to5mac.com/2024/01/04/will-the-iphone-16-be-able-to-record-4k-spatial-video/>

- Spatial Video with 4K is expected to be available

- <https://appleinsider.com/articles/24/03/06/capturing-spatial-video-apple-vision-pro-vs-iphone-15-pro>

- The spatial video captured is in a square 1:1 format at 2200 pixels by 2200 pixels. It is a near-perfect recreation of the passthrough viewed by the user.

- Once stereo is captured on supporting phones, offline postprocess can be used to acquire accompanying depth (using for example Depth-Anything <https://github.com/DepthAnything/Depth-Anything-V2/tree/main> and [ZoeDepth](https://github.com/isl-org/ZoeDepth" \t "_blank" \o "https://github.com/isl-org/zoedepth) https://github.com/isl-org/ZoeDepth or similar).

- Meta Quest™ can record spatial video: <https://360rumors.com/quest-3-3d-videos/>

- After recording, the video or photo is captured in side-by-side format, with a square aspect ratio. Photos will also be side-by-side but they are stretched vertically, and need to be edited to fix that.

- <https://deovr.com/blog/84-record-vr-footage-on-the-meta-quest-3>

- The Meta Quest 3™ features two cameras that deliver full-color passthrough, allowing users to record content in 4K (2k per eye), using the Meta Quest Developer HUB (https://developer.oculus.com/documentation/unity/ts-odh).

- The Quest 3's passthrough cameras record footage that is flat 120-100 (possibly 90) degrees.

NOTE: In TV productions it was known that vthere were issues with visual fatigue, nausea due to bad content production. Guidelines that professional producers can take into account have been provided which minimize these effects. Indications whether this also is an issue for user generated content is for further study.

Beyond user-generated content, an ecosystem is developing around this format including movie production, documentaries and live sports. Examples are mentioned here:

- <https://www.apple.com/newsroom/2024/02/2024-mls-season-kicks-off-today-exclusively-on-mls-season-pass-on-apple-tv/>

- <https://www.apple.com/newsroom/2024/01/apple-previews-new-entertainment-experiences-launching-with-apple-vision-pro/>

- <https://www.macrumors.com/2024/01/08/vision-pro-movies-games/>

#### 4.3.2.3 Rendering and Display Systems

Stereoscopic video with the above parameters can be viewed on different rendering and display systems, including

- Backward-compatible to 2D (just view one eye), hence can be viewed on regular phones. The stereoscopic effect is lost in this case.

- Apple Vision Pro ™

- Meta Quest ™: <https://techcrunch.com/2024/02/01/meta-quest-adds-support-for-apples-spatial-video-ahead-of-vision-pro-launch/>

In addition, OpenXR and WebXR define APIs to render stereoscopic video with additional metadata.

- OpenXR APIs exist

- WebXR APIs exist

For rendering multi-view stereo video, including 3D reconstruction, refer to [S5]. It is shown, how additional metadata as defined in clause 4.3.2.1 can be used to improve rendering.

#### 4.3.2.4 Supporting Information

The baseline video can be encoded using HEVC-based encoding tools:

- framepacking (see for example TS 26.118 [6])

- MV-HEVC (see TR 26.966)

The content can be delivered using regular ISO BMFF based distribution, including streaming with DASH/HLS/CMAF.

Editor’s Note

- Typical quality criteria for evaluating the format

- Existing test and reference sequences

- Conversion from other formats (lossless, lossy)

- Uncompressed data size

- Extensibility of the format

#### 4.3.2.5 Benefits and Limitations

##### 4.3.2.5.1 Benefits

The extended stereoscopic video format has the following benefits:

- Simplicity: The technology is supported by existing content production workflows

- Device Support: The technology is supported by emerging devices on the market

- In device decoding and rendering: The technology generally allows that decoding and rendering can be done in the device, which makes it robust against impaired or lossy network connections.

- Content Industry starts to embrace the format, for details see clause 4.3.2.2

- The format is extensible to add additional metadata, for details see clause 4.3.2.1

- User-generated content production workflows exist.

- Backward-compatible rendering. The content can be rendered on 2D displays.

- Very good B2D user experiences have been reported, when the content is properly produced and suitable devices for playback and rendering are used [S2].

##### 4.3.2.5.2 Limitations

The format is primarily used to support lean-back and seated experiences, typically head movements with 3DOF and 3DOF+ can be supported, but may be extended in the future to address additional degrees freedom.

Editor’s Note: More Benefits and limitations will be added over time

## 4.3.3 Dense Dynamic Point Cloud representation format

There are many applications for point clouds such as representing highly accurate maps of landscapes, buildings, infrastructure, etc… but the format is also used to represent people, animals, objects and scenes composed from these. More precisely, for representing people and objects dense dynamic point clouds are in focus.

### 4.3.3.1 Definition

A point cloud frame is defined as set of (x,y,z) coordinates, where x,y,z have finite precision and dynamic range, , depending on the data type that is used for representing the coordinates. Each (x,y,z) can have multiple attributes associated to it (a1 ,a2, a3 …), where the attributes may correspond to color, reflectance, transparency, normals or other properties of the object/scene that would be associated with a point. Colour is typically represented as RGB and a normal is a normal to a point which can be used by the renderer for handling lighting. Typically, each point in a point cloud frame has the same number of attributes attached to it. Dynamic point clouds consist of several consecutive point cloud frames with the same coordinate system, precisions and attributes. The number of points typically changes from one frame to the other and there is no relation between a point of one frame to the other frame. A dense point cloud contains a high density of points with close neighbors (typically more than 500.000 points per frame for a person or object), where a renderer is able to produce a closed surface allowing for a highly detailed representation.

A simple and often used file format for point clouds is the Polygon File Format (PLY) that has been developed by Greg Turk at Stanford University in 1994 [D1]. Other formats, like the Object File Format (OBJ) can also be used to represent point clouds.

Editor’s Note: Will be completed during study

### 4.3.3.2 Production and Capturing Systems

Professional capturing of volumetric video is typically done with a rig of synchronized cameras aligned around the asset(s) to be captured. Depending on the rig, there can be one or more layers of cameras at different height positions, with each layer consisting of up to 60 cameras. Cameras can be equipped with depth sensors. Hardware such as cameras and depth sensors are typically off the shelf equipment, but the assembly in the rig is vendor dependent and proprietary.

The various camera and depth sensor signals are fed into a production pipeline that produces the asset. Production includes stitching the various signals, filling holes, correcting occlusions, etc. Persons or physical objects (e.g. a ball or an instrument) can be combined in an asset or separate assets can be used for each person or object. The representation format of a produced asset is typically a dense dynamic point cloud or a dynamic mesh.

The Volumetric Format Association (VFA) [D2] aims to “Drive the development of volumetric video as the next revolution for content creation, editing 3D content, distribution of 3D content and creating entirely new ways to tell stories and communicate with each other”. One result of their work is an end-to-end workflow consisting of Volumetric Capturing, Volumetric Processing, Volumetric Encoding and Decode/Render. The workflow can be downloaded from their website in [PDF](https://www.volumetricformat.org/_files/ugd/f2416f_3e1aeca4db234afcae9a8c15ea4f610a.pdf) format. Volumetric Capturing is in line with our description above. Volumetric Processing shows the dynamic point cloud representation format as a central element. First a raw point cloud is created, and which is further processed (e.g. fill holes) and converted to the produced asset. Representation formats for the produced assets is either a dynamic point cloud (in the workflow named as a patch-based format) or a dynamic mesh.

The Volumetric Encoding step includes both options, point cloud and mesh. Once streamed and received on a device, the Decode/Render step includes rendering the mesh, the point-cloud as is or generating mesh or voxels prior to rendering.

### 4.3.3.3 Rendering and Display Systems

The dense dynamic point cloud representation format can be rendered to 2D displays such as in mobile phones, tablets, TV sets but also to HMDs or other 3D type displays.

The visual viewing quality of the point cloud format depends heavily on how voxels are rendered. Just reconstructing voxels in 3D space may bring a limited viewing experience and holes/cracks may become visible. To show the impact of rendering two renderers are investigated:

- MPEG renderer: Each voxel is replaced by a cube of a configurable fixed size. This renderer is deliberately simple for studying the pure impact of compression.

- Representative renderer: Each voxel is replaced by a splat of a size that depends on the viewing distance and some blending is implemented to avoid flickering of points. There are no sophisticated techniques such as lighting or use of normals integrated. It represents a minimum of what a device manufacturer would do to prevent holes or cracks to preserve a good subjective experience. It Is not state-of-the-art or most sophisticated renderer possible.

In the following we give an example of the impact of the renderer on the head of the sequence Thomas with Vox 10 conversion:

A person wearing a hat

Description automatically generatedA person wearing a hat

Description automatically generated

Figure 4.3.3.3-1 Vox 10 MPEG renderer Figure 4.3.3.3-2 Vox 10 Representative renderer

Both snapshots are rendered from the same Vox 10 sequence. In Figure 4.3.3.3-1 we see far more cracks and holes and the borderline of the sequence is less smooth. However, the eyebrows look a bit sharper in Figure 4.3.3.3-2. A high-end industry renderer may do better than the renderers illustrated here.

When evaluating or comparing the point cloud representation format it is essential to select a renderer that is representative of a minimum of what the industry would implement, as holes and cracks in images would influence evaluations negatively.

More sophisticated renderers in products could fill better potential holes, recreate detail and apply lighting depending on the scene. The point cloud representation format supports normals which are useful for lighting the scene. When rendering a point cloud sequence in a scene, correct lighting including shadows and colour alignment can greatly impact the realism of the resulting experience.

POINTS\_GL is the simplest OpenGL [8] primitive type used for rendering (lines and polygons are others that are also commonly used) and a point cloud can be interpreted as a vertex stream that represents points (after ordering of the points). Therefore, a point cloud can be rendered in an extremely straightforward way using native OpenGL vertex shaders. The supported rendering in the standard OpenGL specified by the Khronos consortium implies that point clouds can be rendered on devices that support OpenGL which is rather common today. OpenGL vertex shader renders points size larger than zero, this can be set GL\_PROGRAM\_POINT\_SIZE as a configuration of the rendering.

Specific optimizations for rendering are device manufacturer dependent.

### 4.3.3.4 Support Information

#### 4.3.3.4.1 Test and reference sequences

<TBD>

#### 4.3.3.4.2 Uncompressed data size

The uncompressed data size of a point cloud frame depends on the number of points and the number of attributes. The following table gives data size examples and raw bitrates for the sequence Thomas.

Table 4.3.3.4.2-1 Uncompressed data size and bitrate

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sequence | Quantization | #frames | #points | mean frame size (bytes) | bitrate (mbps) |
| Thomas | Vox10 | 32 | 19012250 | 4010396 | 979.10 |
| Thomas | Vox11 | 32 | 76336020 | 16996692 | 4149.58 |
| Thomas | Vox12 | 32 | 305897397 | 71694702 | 17503.59 |

#### 4.3.3.4.3 Known compression technology

Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC) [D3]

Geometry-based point cloud compression (G-PCC) [D4]

Draco [D5]

#### 4.3.3.4.4 Conversion from other formats

Point clouds can be obtained by sampling from surface-based formats such as meshes. Such transformation is lossy. There are different sampling methods (e.g. methods based on face sampling, on texture map sampling, on ray casting from a grid, etc.) and it’s up to the content provider to select the appropriate sampling method depending on the content and creative intent.

#### 4.3.3.4.5 Typical quality criteria

The visual quality of a point cloud depends on the number of points (density) in the point cloud. For attributes colour is mandatory and there may be reflectance, transparency and normal. Colour is typically in RGB with each in 8 bits. Reflectance, transparency and normal can be used by the renderer when the point cloud is rendered in a scene.

Point clouds of around 1M points/frame allow to watch from a wider distance (e.g. from 3m\*) and 2M points/frame allow to get closer (e.g to around 1.5m distance) at good quality for the target scenario. Emotional facial expressions and buttons and tissue structure of cloths is visible. More points per frame improve the details, but this may not be required for the target scenario. But if a scenario would require it, a professional volumetric video production system is able to capture details from e.g. skin or finer details of tissue and it can be represented with the point cloud representation format.

\* A typical demonstration scenario would be to use e.g. a smartphone or tablet running a volumetric video application showing a real person of e.g. 3m distance on the screen captured by the camera and rendering at the same time a second person rendered from a point cloud next to the first person.

Other scenarios may require the representation of the full detail of a person and the number of required points can be approximated as follows:

Assumptions:

- The visual resolution of the human eye is 1/60 of a degree

- Average human body surface is about 1.9m2

- For simplification the body surface is approximated as a square

Number of points = 1.9/((tan1/60 \* d)^2), where d is the viewing distance from the person.

This leads to the following number of points:

- 1.5m distance: 10 M pixels

- 3m distance: 2.5 M pixels

### 4.3.3.5 Benefits and Limitations

#### 4.3.3.5.1 Benefits

Point cloud representation is simple in structure and representation, has high accuracy and resolution, is faithful to original data, and is easy to acquire from sensors or cameras. Point cloud generation needs less pre-processing as there is no need for surface reconstruction, if sensor data is not so noisy.

A point cloud can be rendered in an extremely straightforward way using native OpenGL vertex shaders.

#### 4.3.3.5.2 Limitations

Point-cloud data does not include information on surfaces and is harder to edit or transform.

Editor’s Note: Will be completed during study.

### 4.3.4 Multi-view video representation

#### 4.3.4.1 Definition

The multi-view video representation consists of multiple frames of multiple synchronized physical or virtual camera views. Each camera view is represented by a colour image (YCbCr), camera intrinsics and camera extrinsics. The combination of video and metadata allows for novel view synthesis (6DoF rendering).

A typical spatial resolution for each of the views is 1920 × 1080 (FullHD). For this representation in this study, we expect resolutions in a range around this number. A typical number of views is 2-4 for real-time capture with range-sensing cameras like the Azure Kinect, and typically 10-20 for offline capture with industrial or professional cameras. Typically, the frame rate is 25 or 30 Hz, and capture beyond 60 Hz is not expected for the coming years.

Optionally there is also a depth image of equal resolution. It is possible to have multi-view content for which some or all views lack depth information. This choice originates from the production and capturing system and thus it is the same for all frames of a view. The depth map image, if present, may also indicate that individual samples are missing. This indication can be used for range-sensing cameras that cannot sense depth in certain situations like object edges, non-reflecting and specular reflecting scene elements. It can also be useful in a production system to remove parts of an image that are not wanted (e.g. revealing camera rigs) or are also present in other views (scene background).

If a view has a depth map, then it must have corresponding depth quantization parameters: quantization type (normalized disparity or linear depth), nearest depth in scene units, furthest depth in scene units, and indication of invalid values. Normalized disparity [m-1] is more commonly used when depth is estimated because it places the code points in a way that correlates with the amount of parallax, and it allows for far away scene elements like the horizon or the star field. Linear depth [m] is commonly used with range-sensing cameras (ToF, LiDAR, etc.) because they often have a limited depth range with equal depth resolution for that entire range.

The camera intrinsics are a model of the projection of points in space in the reference system of the camera to the image sensor (projection plane). Typical parameters include projection type (perspective, fisheye, etc.), and projection-type specific parameters, such as principal point and focal length for perspective projection. Optionally lens distortion parameters may be provided if the camera images are not already corrected for that.

The camera extrinsics model the translation and rotation of a camera in space with respect to the reference system of the scene.

The source format has at least two views. It is expected that most or all test data will have perspective projection (PSP), but test data with equirectangular projection (ERP) may be included.

While this representation allows for 6DoF rendering, it depends on the position and field of view of the cameras, if such a rendering has an acceptable quality. Preferably, the virtual viewpoints are within a viewing space that can be provided as metadata or implicitly derived from the parameters of the set of source views.

Each view has the following video components and metadata:

|  |  |  |
| --- | --- | --- |
| **Component** | **Texture (mandatory)** | **Depth (optional)** |
| **Spatial resolution** | At least 960 × 540  At most 3840 × 2160 | The same as the texture component |
| **Chroma format** | YCbCr | Luma only or YCbCr with chroma planes set to neutral gray |
| **Chroma subsampling** | 4:2:0 | 4:0:0 or 4:2:0 with chroma planes set to neutral gray |
| **Pixel aspect ratio** | 1:1 | 1:1 |
| **Frame rate** | 30, 50, 60 | The same as the texture component |
| **Colour space format** | ITU‑R BT.709 or ITU‑R BT.2100 | Undefined |
| **Transfer characteristics** | Limited range or full range with transfer characteristics matching to the colour space format.  Mastering characteristics such as MDCV (master display colour volume) and CLLI (content light level information) SEI (supplementary enhancement information) messages defined in TS 26.116 Section 4.5.5.7 will be considered. | Full range, linear transfer |
| **Bit depth** | Either 8 bits or 10 bits for all channels | At least 8 bits  At most 16 bits |
| **Metadata** | Camera intrinsics:  Projection type (Perpsective, ERP)  - Projection type (Perpsective, ERP)  - For perspective projection:  - Focal length [px]  - Principal point [px × px]  - For equirectangular projection:  - Latitudinal angle range [rad × rad]  - Longitudinal angle range [rad × rad]  - Lens distortion parameters (optional)  Camera extrinsics:  - Camera position (x, y, z) [m]  - Camera orientation as normalized quaternion (*q* = *iq*x + *jq*y + *kq*y + *q*w) | Depth quantization parameters:  - Quantization type:  - either: normalized disparity  - or: linear depth  - Near depth [m]  - Far depth [m]  - Has invalid pixels flag |

#### 4.3.4.2 Production and Capturing Systems

Multi-view video and multi-view + depth are well-known formats that have many public tools including OpenCV [M1], COLMAP [M2], AliceVision [M3] and OpenMVG [M4]. Also, MPEG has published tools for camera calibration and depth estimation [M5].

There are four typical workflows for multi-view (+ depth):

- Use color cameras to capture multi-view and estimate depth with multi-view consistency.

- Use range-sensing cameras to capture multi-view + depth and refine depth with multi-view consistency.

- Use AI or CG pipelines to raytrace views.

- Combinations of the above.

The beyond 2D video is captured and processed using multiple cameras. Zero or more of those cameras may be range-sensing cameras, and more than one of the cameras has color sensors. In the case of two or more cameras that are not rigidly connected, camera extrinsics are online calibrated. Depth estimation is performed to associate a full depth map with each of the camera views, thus resulting in a multi-view + depth representation.

Additional steps such as object instance segmentation and foreground/background separation may be performed to reduce the sample rate of the representation. This would result in a multi-view + depth + transparency/occupancy representation. All processing may be offline or with a delay of a few seconds.

Figure 1 provides an example processing flow with the following operations:

- Multi Camera Capture: capture of images from multiple cameras

- Intrinsic Calibration: estimation of principal point, focal length and distortion parameters

- Extrinsic Calibration: estimation of camera orientation and translation (e.g. using COLMAP)

- Scene Calibration: estimation of static ground plane geometry and background geometry

- Undistort Images: all images undistorted to one and the same reference intrinsics

- Object Instance Segmentation: determine segments for known objects such as ‘person’/’ball’

- Depth Estimation: determine a dense depth map for each view

- Depth Segmentation: determine sub-instance depth segments consisting of smooth surfaces

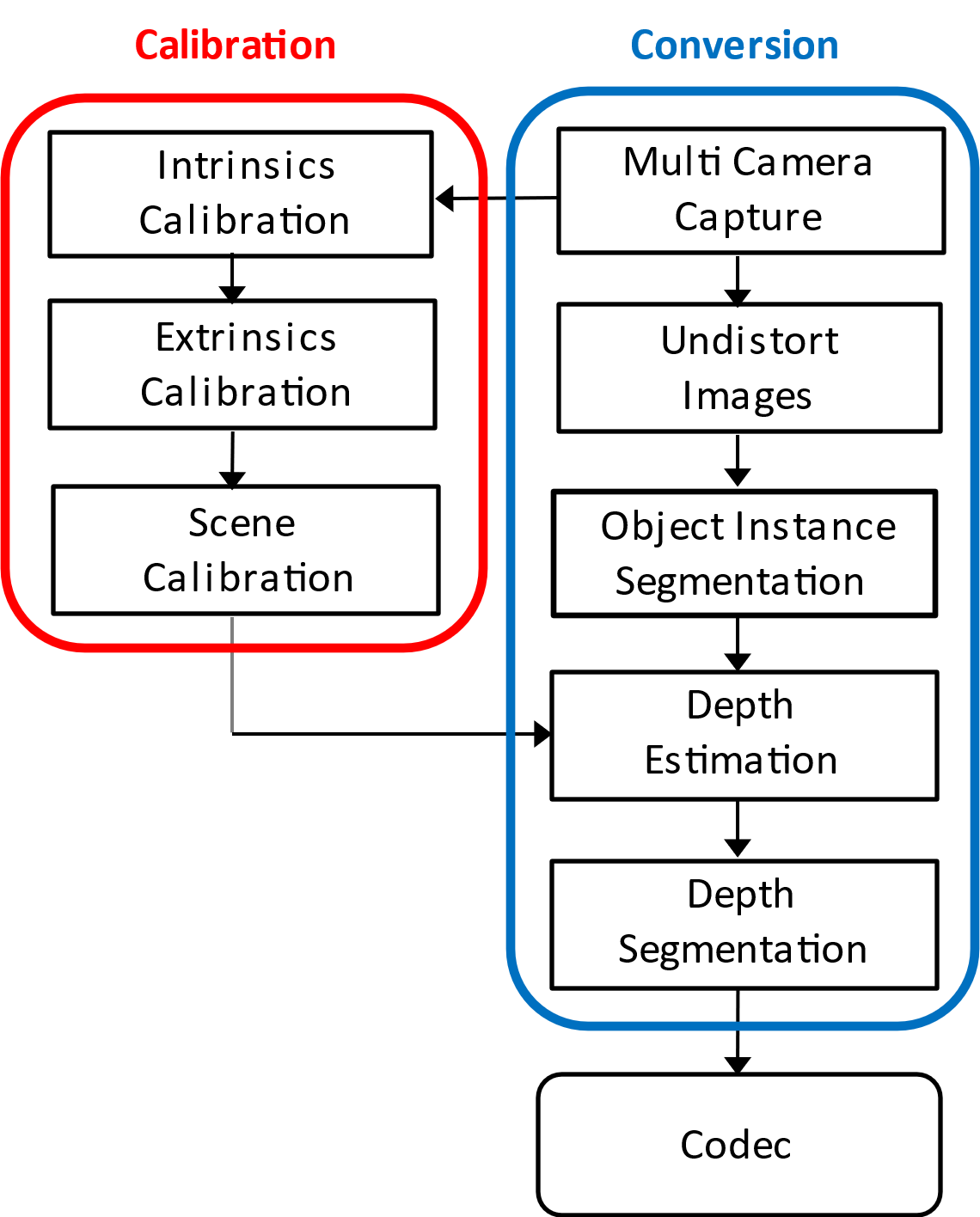


Figure 1: Example processing flow

#### 4.3.4.3 Rendering and Display Systems

Some examples of open source rendering implementations for the multi-view representation are the Reference view synthesizer [M6], Test model for MPEG immersive video [M7], and OpenDIBR [M8]. More implementations exist.

Real-time rendering is typically performed on a GPU without dedicated hardware.

Rendering can be on:

- a device for 2D presentation (fixed viewpoint) such as a phone,

- a device for 3D presentation (multiple viewpoints) such as an autostereoscopic display,

- a device for 6DoF presentation (dynamic viewports) such as an HMD or an autostereoscopic display with eye tracking.

When a viewing space is used, then:

- What is rendered is one or two viewports with perspective projection and with 6 degrees of freedom (3-D position and 3-D orientation).

- The pose of the viewport is within a viewing space that can be signaled or implicitly determined from a decoded frame. A viewing space can limit both position, orientation or both in combination. For instance, it is generally not intended for a viewport to intersect with scene elements.

- When a viewport is rendered that is outside of the viewing space, then the renderer has to perform a mitigation to avoid a viewing experience that is not intended by the content provider.

#### 4.3.4.4 Supporting Information

##### 4.3.4.4.1 Camera placement

For range-sensing cameras a minimum requirement is that scene elements are present in at least one view frustum (seen by at least one camera). This implies that view frustums of adjacent cameras are overlapping at and beyond the nearest object distance.

With multi-view depth estimation, the minimum requirement is that scene elements are present in at least two view frustums (seen by at least two cameras). The additional overlap is needed for stereo correspondence checks.

- Given the above, cameras do not have to be placed on a line, plane or any specific geometry.

##### 4.3.4.4.2 Spatial resolution

The 3-D spatial resolution relates to video resolution and depth map bit depth. Because perspective projection and normalized disparity are most common, equations are provided for this case only.

*Perspective unprojection* maps a sample position ***x***image to a scene position in respect to the camera ***x***camera:

with principal point ***p*** and focal length *f*, both in pixel units. The coordinate system is only a convenient example.

*Normalized disparity expansion* maps sample value *i* to depth range value *r* in scene units, e.g. meters:

One may check that this maps to . Note that nearby objects appear brighter when viewing the depth map directly (as if using a flashlight in a dark room).

In-plane spatial resolution refers to the ability of an imaging system to distinguish between two adjacent points within the same imaging plane. It is a measure of how close two objects can be to each other in the imaging plane while still being resolved as separate entities.

The in-plane spatial resolution can be derived from the first equation:

This indicates that a) the in-plane spatial resolution depends on the focal length in pixel units, and b) the spatial resolution decreases with distance from the camera. As an example, when an object is at 1 meter distance, and the focal length is 1000 pixels, a horizontal or vertical shift of one pixel corresponds to a shift of 1 mm in 3-D space.

Out-of-plane spatial resolution refers to the ability of an imaging system to distinguish between points along the axis perpendicular to the imaging plane (typically the z-axis). It represents the system's capability to resolve depth information or separate structures at different depths.

The out-of-plane spatial resolution can be derived by combining the first and second equations:

With much larger than , this approximates to:

This indicates that a) the out-of-plane spatial resolution depends mainly on the nearest object distance and the depth map bit depth, and b) the spatial resolution decreases quadratically with distance from the camera. As an example, when an object is at meter distance and bit depth is 8 bit (*i*max = 255), then . When instead an object is at 10-meter distance, the step size is about 0.4 m.

##### 4.3.4.4.2 Objective metrics

Objective evaluation on multiview video may be performed by applying 2D video objective metrics (PSNR, SSIM, VMAF, etc.) on each of the source view positions, and averaging them in the correct domain. A higher correlation with subjective evaluation may be obtained by applying immersive video metrics [M9] [M10] that consider that view synthesis may cause pixel shifts that have only a minor influence on subjective scores, but cause PSNR to degrade.

##### 4.3.4.4.3 Coding and delivery options

The content can be encoded using:

- MPEG Immersive Video (MIV)

- MV-HEVC (albeit with some restrictions)

The content can be delivered using regular ISO BMFF based distribution, including streaming with DASH, or delivered in real-time using RTP-based transport.

[Ed.(BK): On acceptance, Philips intend on providing a proposal to rewrite the related scenarios, avoiding overlap with the representation.]

#### 4.3.4.5 Benefits and Limitations

##### 4.3.4.5.1 Benefits

The multiview video representation has the following main benefits:

- Real-time capture is feasible.

- This format is often used as an intermediate step in photogrammetry pipelines such as [ M3].

- The renderings have the appearance of natural video content, as opposed to computer graphics, because all optical effects are baked into the multiple views.

##### 4.3.4.5.2 Limitations

The multiview video representation has the following limitations:

- Need to handle large number of pixels, e.g. by selection.

- For novel view synthesis, multiple views need to be blended for optimal rendering results, to handle non-Lambertian effects.

- Content production depends on the availability of good and efficient depth estimation/refinement tools. Recently, there is a strong progress in the field of computer vision and volumetric approaches specifically, which will benefit applications of this representation.

### 4.3.X Formats under Research

Editor’s Note: Formats in that section will not be part of the evaluation framework of release 19, due to their maturity status, or complexity. However, it is recommended that 3GPP follows the research work on NERF, INVR and GS and awaits stabilization in the industry to commonly agreed formats.

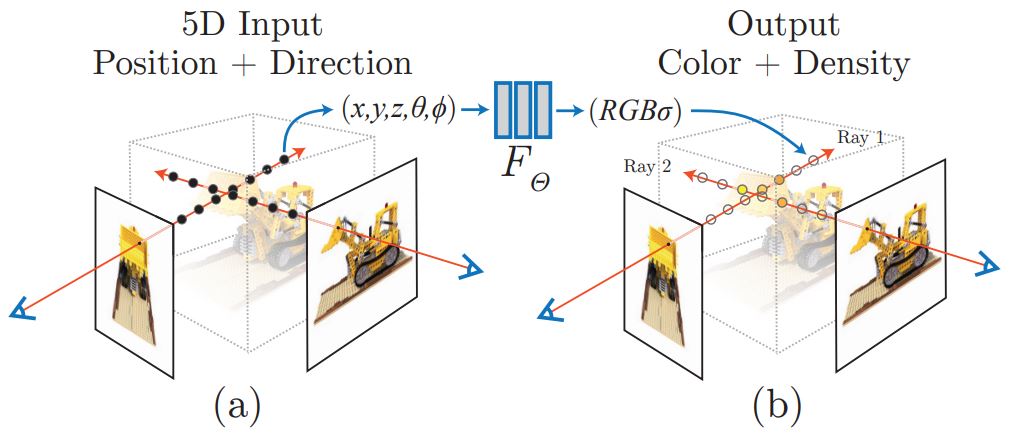
#### 4.3.X.1 Neural Radiance Fields

##### 4.3.X.1.1 Introduction

Neural Radiance Field (NeRF) is a technology at the intersection of Artificial Intelligence (AI) and 3D graphics, and has gained interest based on remarkable progress in computer vision, neural processing units and graphics processing. NeRF was an important research area over the last few years,[but recently the interest in NeRF has declined and more attention is given to other formats documented in the remainder of this clause 4.3.X]. The documentation reflects the state of the art at the time of writing, but the technology has reached a level of maturity.

##### 4.3.X.1.2 Definition

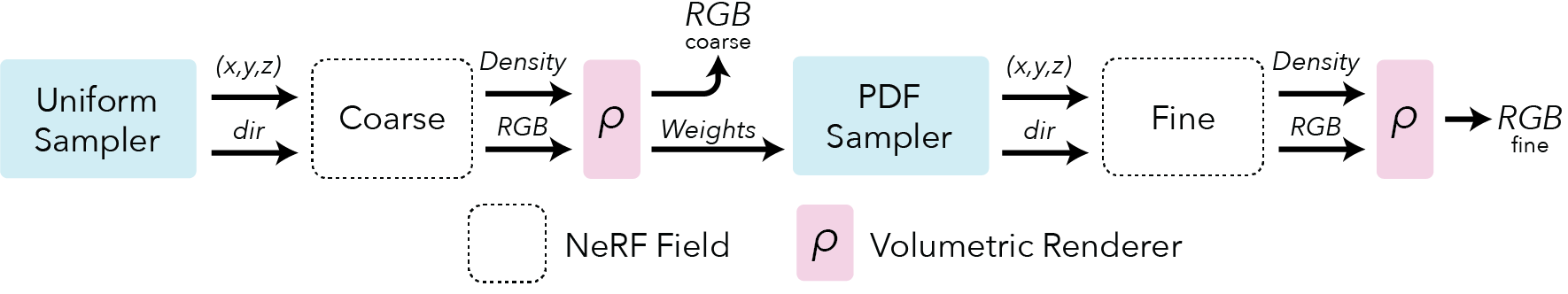
NeRF is the implicit representation of a 3D scene or object using a fully-connected (non-convolutional) deep network, whose input is a single continuous 5D coordinate (spatial location (x, y, z) and viewing direction (Θ,Φ)) and whose output is the volume density (α) and view-dependent emitted radiance (r, g, b) at that spatial location [N1].



**Figure.4.3.X.1.1-1 NeRF representation[N1]**

The key idea behind NeRF is to represent the appearance of a scene as a function of 3D position and viewing direction, known as the radiance field. The radiance field describes how light travels through the scene and interacts with its surfaces and can be used to generate images from arbitrary viewpoints [N6].

The following is an overview pipeline for NeRF:



**Figure.4.3.X.1.1-2 NeRF pipeline (source: https://docs.nerf.studio/nerfology/methods/nerf.html )**

**Field representation:** For each point in space the NeRF represents a view dependent radiance.

**Positional encoding:** The input coordinates (x,y,z,θ,ϕ) need to be encoded to a higher dimensional space prior to being input into the network.

**Rendering**: NeRF rely on classic volumetric rendering techniques to composite the points into a predicted color.

**Sampling:** NeRF use a hierarchical sampling scheme that first uses a uniform sampler and is followed by a PDF sampler.

##### 4.3.X.1.3 Production and Capturing Systems

Mobile apps such as NeRFCapture (https://github.com/jc211/NeRFCapture), Spectacular AI (https://github.com/SpectacularAI), or Record3D (<https://record3d.app/>) are available to capture NeRFs.

A tutorial for capturing NeRFs is provided here: https://github.com/NVlabs/instant-ngp/blob/master/docs/nerf\_dataset\_tips.md.

The NeRFCapture app allows any iPhone™ or iPad™ to quickly collect or stream posed images to InstantNGP. If your device has a LiDAR, the depth images will be saved/streamed as well. It has two modes: Offline and Online. In Offline mode, the dataset is saved to the device and can be accessed in the Files App in the NeRFCapture folder. Online mode uses CycloneDDS to publish the posed images on the network. A Python script then collects the images and provides them to InstantNGP.

The Spectacular AI SDK and apps can be used to capture data from various devices:

- iPhones (with LiDAR)

- OAK-D cameras

- RealSense D455/D435i

- Azure Kinect DK

The Record3D can create a dataset with an iPhone 12 Pro or newer (based on ARKit), a python code is needed to convert the captured data to NeRF (https://github.com/NVlabs/instant-ngp/blob/master/scripts/record3d2nerf.py)

The state-of-art of NeRF at the time of writing includes:

- SMERF (Streamable Memory Efficient Radiance Fields for Real-Time Large-Scene Exploration) is a view synthesis approach that achieves state-of-the-art accuracy among real-time methods on large scenes with footprints up to 300 m2 at a volumetric resolution of 3.5 mm3 [N7] . It enables fully 6DoF navigation within a web browser, and renders real-time on smartphones and laptops.

- Instant Neural Graphics Primitives (Instant-NGP) using multi-resolution hash encoding to split the processing into multiple chunks and using parallel processing using cuda software to effectively change run time from hours to seconds [N8]. Instant-NGP is a method that uses hash-grid and a shallow MLP to accelerate training and rendering. This method reaches speedups of 1000x and train very fast (~6 min) and renders also fast ~3 FPS.

- [NerfStudio](https://docs.nerf.studio/" \t "https://medium.com/@heyulei/_blank) (https://docs.nerf.studio/), which is open-source and combines many radiance fields methods, and supports the storage of NeRF data in a structured format, which includes key elements as follows.

Camera intrinsics:

{

"camera\_model": "OPENCV\_FISHEYE", // camera model type [OPENCV, OPENCV\_FISHEYE]

"fl\_x": 1072.0, // focal length x

"fl\_y": 1068.0, // focal length y

"cx": 1504.0, // principal point x

"cy": 1000.0, // principal point y

"w": 3008, // image width

"h": 2000, // image height

"k1": 0.0312, // first radial distortion parameter, used by [OPENCV, OPENCV\_FISHEYE]

"k2": 0.0051, // second radial distortion parameter, used by [OPENCV, OPENCV\_FISHEYE]

"k3": 0.0006, // third radial distortion parameter, used by [OPENCV\_FISHEYE]

"k4": 0.0001, // fourth radial distortion parameter, used by [OPENCV\_FISHEYE]

"p1": -6.47e-5, // first tangential distortion parameter, used by [OPENCV]

"p2": -1.37e-7, // second tangential distortion parameter, used by [OPENCV]

"frames": // ... per-frame intrinsics and extrinsics parameters

}

Camera extrinsics:

{

// ...

"frames": [

{

"file\_path": "images/frame\_00001.jpeg",

"transform\_matrix": [

// [+X0 +Y0 +Z0 X]

// [+X1 +Y1 +Z1 Y]

// [+X2 +Y2 +Z2 Z]

// [0.0 0.0 0.0 1]

[1.0, 0.0, 0.0, 0.0],

[0.0, 1.0, 0.0, 0.0],

[0.0, 0.0, 1.0, 0.0],

[0.0, 0.0, 0.0, 1.0]

]

// Additional per-frame info

}

]

}

Depth images:

{

"frames": [

{

// ...

"depth\_file\_path": "depth/0001.png"

}

]

}

Masks:

{

"frames": [

{

// ...

"mask\_path": "masks/mask.jpeg"

}

]

}

##### 4.3.X.1.4 Rendering and Display Systems

NeRF heavily relies on the volumetric rendering process to obtain rendered pixels. This rendering function is differentiable, so scene representation can be optimized by minimizing the residual between synthesized and ground truth observed images. The rendering process requires sampling tens to hundreds of points along each ray and inputting them into the neural network to produce the final imaging result. Consequently, rendering a single 1080p image necessitates on the order of 108 neural network forward passes, which often takes several seconds [N2].

Display System: VR HMD, mobile devices.

##### 4.3.X.1.5 Supporting Information

- Typical quality criteria for evaluating the format

- Evaluation metrics such as PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index), and LPIPS (LearnedPerceptual Image Patch Similarity)

- Training iteration, training time, inference speed.

- Conversion from other formats (lossless, lossy)

- Meshes, point clouds

- Uncompressed data size

The original NeRF model has 8 fully connected layers, with a layer width of 256, and each pixel is synthesized based on 128 samplings along the ray. The standard NeRF model demands an impractical 5,600 Terabytes cache size.

- Known compression technologies:

Early research on NeRF compression is ongoing. The MPEG established the ad-hoc group called Implicit Neural Visual Representation (INVR) and is currently exploring the potential standardization of 6 Degree of Freedom (6DoF) video compression using NeRF-based technologies [N5]. The following methods are applied in current research for NeRF compression and encoding:

- Parameter quantization techniques, transform coding, and entropy coding [N3]

- VVC and NNC [N4]

- Extensibility of the format

- Mip-NeRF, Point-NeRF, KiloNeRF, Mega-NeRF and etc [N8].

##### 4.3.X.1.6 Benefits and Limitations

###### 4.3.X.1.6.1 Benefits

- High-quality 3D representation: NeRF can create photo-realistic 3D reconstructions of complex scenes, including fine surface details, reflections and realistic lighting effects.

- Improved view synthesis capabilities: NeRF can synthesize novel views of a scene or object from a small number of input images, allowing rendering from any viewpoint.

- Flexibility: NeRF can handle non-rigid and dynamic scenes, adapting well to varying spatial conditions and changes over time.

- Unsupervised training: NeRF can learn to reconstruct a scene or object without explicit supervision.

###### 4.3.X.1.6.2 Limitations

- More computationally demanding and slower to render compared to photogrammetry and 3D Gaussian Splatting.

- Not reductionistic: The entire scene is encoded in a single NeRF function, which makes it challenging to segment the scene into parts, edit individual objects within the scene, or combine different NeRF scenes into one.

- Currently, NeRF representation formats do not seem to effectively handle dynamic content within 3D scenes.

# 5 Overview of existing "Beyond 2D" Video Capabilities in 3GPP

Editor’s note: This clause summarized existing beyond 2D video capabilities in 3GPP from at least TS.26.119 and TS.26.118.

## 5.1 Introduction

<TBD>

## 5.2 AR Video Capabilities

3GPP TS 26.119 [5] specifies the mandatory and optional media capabilities and profiles to be supported for each XR device type. These media capabilities include support for video codecs (AVC and HEVC), audio codecs (EVS, IVAS and AAC-ELDv2), scene description formats, and XR system capabilities. Table 5.2-1 summarized the Beyond 2D video capabilities defined in clause 7 of TS 26.119 [5].

NOTE: The definition of concurrent video decoder instances can be found in clause 7.1.2.1 of TS 26.119 [5].

Table 5.2-1: Summary of Operation Points

|  |  |  |
| --- | --- | --- |
| Operation Point Name | Max Concurrent Video Decoder Instances | Decoding Capabilities |
| AVC-FullHD-Dec-2 | 2 | Aggregate decoding capabilities of H.264/AVC HP@L4.0 |
| AVC-UHD-Dec-4 | 4 | Aggregate decoding capabilities of H.264/AVC HP@L5.1 |
| HEVC-UHD-Dec-4 | 4 | Aggregate decoding capabilities of H.265/HEVC MP10@L5.1 |
| UHD-Dec-4 | 4 | Aggregate capabilities of *AVC-UHD-Dec-4* |
| Aggregate capabilities of *HEVC-UHD-Dec-4* |
| Decoding up to 4 bitstreams, each not exceeding the capabilities of H.264/AVC HP@L4.0 or H.265/HEVC MP10@L4.1. |
| AVC-8K-Dec-8 | 8 | Aggregate capabilities of H.264/AVC HP@L6.1 |
| HEVC-8K-Dec-8 | 8 | Aggregate capabilities of H.265/HEVC MP10@L6.1 |
| 8K-Dec-8 | 8 | Aggregate capabilities of *AVC-8K-Dec-8* |
| Aggregate capabilities of *HEVC-8K-Dec-8* |
| Decoding up to 8 bitstreams, each not exceeding the capabilities of H.264/AVC HP@L4.0 or H.265/HEVC MP10@L4.1. |
| Decoding up to 4 bitstreams, each not exceeding the capabilities of H.264/AVC HP@L5.1 or H.265/HEVC MP10@L5.1. |

## 5.3 VR Video Profiles

The VR profiles for streaming services are defined in TS 26.118 [6], specifying the coded representation and media profile of 360 VR distribution signals. Table 5.3-1 provides an overview of the 360 VR relevant formats considered in the context of 3GPP VR Profiles.

For restrictions on source formats such as resolution and frame rates, content generation and encoding guidelines, refer to TS 26.118 [6], Annex A.

Table 5.3-1: High-level Summary of Operation Points

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Operation Point name | Decoder | Bit depth | Typical  Original Spatial Resolution | Frame Rate | Colour space format | Transfer  Characteristics | Projection | Rotation | RWP | Stereo |
| Basic H.264/AVC | H.264/AVC [HP@L5.1](mailto:HP@L5.1) | 8 | Up to 4k | Up to 60 Hz | BT.709 | BT.709 | ERP w/o padding | No | No | No |
| Main H.265/HEVC | H.265/HEVC [MP10@L5.1](mailto:MP10@L5.1) | 8, 10 | Up to 6k in mono and 3k in stereo | Up to 60 Hz | BT.709  BT.2020 | BT.709 | ERP w/o padding | No | Yes | Yes |
| Flexible H.265/HEVC | H.265/HEVC [MP10@L5.1](mailto:MP10@L5.1) | 8, 10 | Up to 8k in mono and 3k in stereo | Up to 120 Hz | BT.709  BT.2020 | BT.709,  BT.2100 PQ, BT.2100 HLG | ERP w/o padding CMP | No | Yes | Yes |
| Main 8K H.265/HEVC | H.265/HEVC MP10@L6.1 | 10 | Up to 8k in mono and 6k in stereo | Up to 60 Hz for 8K and 120 Hz for 4k | BT.709  BT.2020 | BT.709, BT.2100 PQ, BT.2100 HLG | ERP w/o padding | No | Yes, but restricted to coverage | Yes |

Table 5.3-2 summarizes the video operation point, sample entry, and DASH integration associated with each video media profiles defined in clause 5.2 of TS 26.118 [6].

Table 5.3-2 Video Media Profiles

|  |  |  |  |
| --- | --- | --- | --- |
| Media Profile | Operation Point | Sample Entry | DASH Integration |
| Basic Video | Basic H.264/AVC | resv  avc1 | Single Adaptation Set  Single Representation streaming |
| Main Video | Main H.265/HEVC or Main 8K H.265/HEVC | resv  hvc1 | Single or Multiple independent Adaptation Sets offered  Single Representation streaming |
| Advanced Video | Flexible H.265/HEVC | resv  hvc1, hvc2 | Single or Multiple dependent Adaptation Sets offered  Single or Multiple representation streaming |

## 5.4 Messaging Services

3GPP TS 26.143 [7] specifies the media types, formats, codecs capabilities and profiles for the messaging applications used over the 5G System. The document extends to codecs for speech, audio, video, still images, bitmap graphics, 3D scenes and assets, and other media in general, as well as scene description.

Specifically, the 2D video capabilities defined in TS 26.143 [7] clause 6.2 are fully aligned with 5G Media Streaming in 3GPP TS 26.511 [8]:

- **AVC with HD** and **Full-HD resolutions**

- **HEVC with HD**, **Full-HD** and **UHD resolutions**

For Beyond 2D video capabilities, as HEVC simulcast and HEVC frame packing already been included in SA4 specifications and given the coding benefits MV-HEVC provides compared to these solutions, the support for stereoscopic MV-HEVC for low delay applications of stereoscopic 3D video was recommended by TR 26.966 [9]. This aspect is being addressed in a Rel-19 work TS 26.265 [10].

# 

# 6 Evaluation and Characterization Framework

## 6.1 Overview

Generally, the test and characterization framework as documented in TR 26.955, clause 5 also applies to this document. This clause only documents differences and extensions that are needed for beyond 2D Evaluation and characterization framework.

The overview of the evaluation framework for the B2D messaging is presented in Figure 6.1-1. Representative reference sequences are collected and stored in a well-defined B2D format. For a video encoder, the configuration is provided that matches the application constraints. The resulting video streams are “pseudo”-packaged in order to determine the file size/bitrate. The data is then unpackaged, and a B2D video decoder is used to reconstruct data in the B2D format again. The data is stored. The original sequence and the recovered sequence are used determine metrics. The sequences may also be inspected subjectively.

A diagram of a computer

Description automatically generated

Figure 6.1-1 B2D Evaluation framework

## 6.2 Reference Sequences

This document provides reference sequences that are used to generate anchors and are also made available in order to generate test bitstreams for other codecs. Reference sequences are selected to be representative for a scenario.

Reference sequences are described in Annex C of this document along with their properties and their licenses. A format for raw reference sequences based on a JSON schema is defined in clause B.2.

Annex D describes how to upload new proposed reference sequences and how to download the reference sequences.

## 6.3 Reference Software Tools

Editor’s Note: For further study

## 6.4 Metrics

Editor’s Note: For further study

The metrics in clause 5.5 of TR 26.955 also apply for this report.

In addition, the following is defined

Editor’s Note: For further study

For metrics reporting, the following csv scheme is defined: tbd

## 6.5 Encoding Constraints

The encoding constraint definition in clause 5.6 of TR 26.955 also apply for this report.

In addition, the following is defined:

- Equal Quality Views: equal quality views refers to the encoding such that each view when decoded has the same quality target, typically applying the same QP.

Editor’s Note: More details are to be defined

# 7 Considered Scenarios

## 7.1 Introduction

Editor’s note: This clause collects end-to-end scenarios and corresponding workflows for beyond 2D video, based on the template defined in Annex A. Alignment with the generalized media delivery architecture defined in TS 26.501/506 is expected, primarily addressing reference points M2 and M4.

## 7.2 Scenario 1: <tbd>

## 7.3 Scenario 2: <tbd>

## 7.4 Scenario 3: <tbd>

## 7.x Scenario x: <tbd>

# 8 Common Evaluation Features

Editor’s note: Documents common metrics, software, etc..

# 9 Evaluation of Selected Scenarios

Editor’s note: This clause defines test conditions and parameters, KPIs, Metrics, test sequences, agreed reference signals per scenario.

## 9.1 Introduction

Editor’s note: Identifies the preferred scenarios

## 9.2 Scenario 1: <tbd>

### 9.2.1 Evaluation Overview

Editor’s note: Based on scenario in clause 6, summarizes the source formats parameters used for evaluation, the encoding and decoding constraints, interoperability considerations and the general idea of the performance metrics.

### 9.2.2 Reference Sequences

### 9.2.3 Performance Metrics

### 9.2.4 Candidate Solutions

#### 9.2.4.1 Solution 1: <Name>

##### 9.2.4.1.1 Introduction

##### 9.2.4.1.2 Reference Software

##### 9.2.4.1.3 Parameter Settings

##### 9.2.4.1.4 Distribution

##### 9.2.4.1.5 Evaluation Results

##### 9.2.4.1.6 Network Requirements

Editor’s note: Documents required bitrates as well as possibly other aspects.

#### 9.2.4.2 Solution 2: <Name>

##### 9.2.4.2.1 Introduction

##### 9.2.4.2.2 Reference Software

##### 9.2.4.2.3 Parameter Settings

##### 8.2.4.2.4 Distribution

##### 9.2.4.2.5 Evaluation Results

##### 9.2.4.2.6 Network Requirements

### 9.2.5 Summary of Evaluation

## 9.3 Scenario 2: <tbd>

## 9.4 Scenario x: <tbd>

# 10 Gaps and Optimization Potential

## 10.1 Identified Gaps and Deficiencies with Video Capabilities

## 10.2 Potential Requirements for New Video Capabilities

## 10.3 Potential Network Optimizations

# 11 Conclusions and Proposed Next Steps

Editor’s note: This clause provides conclusion and potential areas for normative work as the next phase.

# Annex A: Scenario Template

## A.1 Introduction

This annex provides a proposed template to introduce a Scenario for Beyond 2D Video. This template has been used to collect the scenarios in this report. The text in blue corresponds to guidelines on the information to be provided with a scenario proposal.

## A.2 Template

The following aspects are considered for a scenario:

1. **Scenario name**
2. **Motivation for the scenario**

*What is the market relevance of the proposed scenario within the next few years? Are there any commercially available or pre-released products or prototypes?*

*Market relevance key indicators:*

1. *Technology evaluation on the market*

*Are there indications of pre-evaluation by service providers, device manufacturers, and/or network operators?*

1. *Industry activities*

*Is there relevant work in 3GPP MRPs, industry collaborations or among market stakeholders?*

1. *Production tools/companies*

*What is the availability of capturing setups, and production software? Are there endorsed formats for representation, contribution, compression, and storage? Is there an ecosystem of content creators?*

1. *Delivery solutions*

*Which delivery type is expected to be used? What are the expected transport formats? Is there SW or HW support and providers?*

1. *Content decoding and rendering*

*Is there decoding SW/HW support, and providers? Are there rendering devices and displays available yet?*

1. **Description of the scenario**

*This provides a description of beyond 2D video end-to-end workflows, which includes identifying and defining beyond 2D formats being used in the context and representation technologies to delivery these formats. The following aspects may be considered for each workflow:*

1. *Capturing and processing*
2. *Encoding*
3. *Packaging and delivery*
4. *Decoding*
5. *\*Post-processing*
6. *Rendering*
7. *General constraints on latency, bandwidth, reliability and complexity*
8. **Supporting companies and 3GPP members**
9. *This documents the 3GPP members that support this scenario in terms of providing the information, test material, test requirements and the characterization for the tests. For each of the identified necessities, a tick box is created in the template.*
10. *Preferably several 3GPP members are included in the support, and in addition a video service provider may be included (not necessarily a 3GPP member).*
11. *Cross-verification is preferably done by the supporters of the scenario.*
12. **Source format properties**

*This defines a clear range of the considered and relevant source formats, including the signal properties, but also the characteristics of the content. As an example, the texture and depth format properties of the source may be used which include:*

1. *Spatial resolutions*
2. *Chroma Format*
3. *Chroma Subsampling*
4. *Aspect ratios*
5. *Frame rates*
6. *Colour space formats*
7. *Transfer Characteristics*
8. *Bit depth*
9. *Viewpoints*
10. *Other signal properties*
11. **Encoding and decoding constraints and settings**

*Typical encoding constraints and settings such as:*

* 1. *Relevant Codec and Codec Profile/Levels according to 3GPP TS (e.g., TS 26.119),*
  2. *Random access frequency*
  3. *Error resiliency requirements*
  4. *Bitrates and quality requirements*
  5. *Bitrate parameters (CBR, VBR, CAE, HRD parameters)*
  6. *ABR encoding requirements (switching frequency, etc.)*
  7. *Latency requirements and specific encoding settings*
  8. *Encoding context: real-time encoding, on device encoding, cloud-based encoding, offline encoding, etc.*
  9. *Required decoding capabilities*
  10. *Synchronization requirements*

1. **Performance Metrics and Requirements**
2. *A clear definition on how the performance needs to be evaluated including metrics, etc addressing the main KPIs of the scenario.*
3. *Objective measures such as PSNR, VMAF, etc, may be used*
4. *Justification on whether objective metrics are sufficient and representative of the subjective performance.*
5. **Interoperability Considerations for the application**
6. *Streaming with DASH/HLS/CMAF/QUIC*
7. *RTP based delivery*
8. **Test Sequences**

*A set of selected test sequences that are provided by the proponents in order to do the evaluation. They should cover a set of source format properties*

1. **Detailed test conditions**

*Provides a proposal for detailed test conditions, for example based on a reference software together with the sequences and configuration parameters.*

1. **External Performance data**

*References to external performance data that can be added, for example other SDOs, public documents and so on.*

1. **Additional Information**
2. *Industry activities*

*Is there Relevant work in industry forums?*

1. *Implementation constraints*

*Are there any indications about scalability of the technology with regards to network and devices?*

1. *Innovation*

*Does the technology address a current or a future need on the market? Can it potentially disrupt existing markets?*

# Annex B: Data Formats and Metrics

## B.1 Introduction

<TBD>

## B.2 Raw Video Sequences

### B.2.1 Overview

<TBD>

### B.2.2 JSON Schema

JSON schema for the raw format is here

< [https://dash-large-files.akamaized.net/WAVE/3GPP/Beyond2D/ReferenceSequence](https://dash-large-files.akamaized.net/WAVE/3GPP/5GVideo/Beyond2D)/raw-schema.json>

1. {
2. "Sequence": {
3. "Name": "Example",
4. "Background": "This is a B2DV format example",
5. "Scenario": "On-demand",
6. "Key": "Identifier",
7. "TR26.956": "Annex X.Y.Z"
8. },
9. "Views": [
10. {
11. "ViewId": "v0",
12. "Extrinsics": {
13. "orientation": {
14. "qw": 0.9999915361,
15. "qx": 0.0024327517,
16. "qy": 0.0024349121,
17. "qz": -0.0022688841
18. },
19. "position": [
20. -0.0006123598,
21. 0.3035059273,
22. 0.0012498678
23. ]
24. },
25. "Intrinsics": {
26. "focalLength": 1002.349976,
27. "principalPoint": {
28. "horizontalNorm": 960.0,
29. "vertical": 540.0
30. }
31. },
32. "ProjectionPlaneSize": {
33. "columnCount": 1920,
34. "rowCount": 1080
35. },
36. "Quantization": {
37. "highNormDisp": 2.000000,
38. "lowNormDisp": 0.200000
39. },
40. "Components": [
41. {
42. "ComponentId": "texture",
43. "Data": {
44. "URI": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.yuv",
45. "md5": "e537665c18e32bbaf8e5e9d63e18dd2c",
46. "thumbnail": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.png",
47. "preview": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.mp4",
48. "size": 7962624000,
49. "md5-10": "1c3550197120f95502c4add38d7ebd33"
50. },
51. "Properties": {
52. "width": 1920,
53. "height": 1080,
54. "format": "yuv",
55. "packing": "planar",
56. "scan": "progressive",
57. "subsampling": "420",
58. "bitDepth": 8,
59. "frameRate": 30,
60. "colourPrimaries": "1",
61. "transferCharacteristics": "1",
62. "matrixCoefficients": "1",
63. "sampleAspectRatio": "1",
64. "duration": 10,
65. "frameCount": 600,
66. "startFrame": 1,
67. "videoFullRangeFlag": "0",
68. "chromaSampleLocType": "0"
69. }
70. },
71. {
72. "ComponentId": "depth",
73. "Data": {
74. "URI": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.yuv",
75. "md5": "e537665c18e32bbaf8e5e9d63e18dd2c",
76. "thumbnail": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.png",
77. "preview": "https://dash-large-files.akamaized.net/WAVE/3GPP/some/url/file.mp4",
78. "size": 7962624000,
79. "md5-10": "1c3550197120f95502c4add38d7ebd33"
80. },
81. "Properties": {
82. "width": 1920,
83. "height": 1080,
84. "format": "yuv",
85. "packing": "planar",
86. "scan": "progressive",
87. "subsampling": "420",
88. "bitDepth": 16,
89. "frameRate": 30,
90. "colourPrimaries": "2",
91. "transferCharacteristics": "8",
92. "matrixCoefficients": "0",
93. "sampleAspectRatio": "1",
94. "duration": 10,
95. "frameCount": 600,
96. "startFrame": 1,
97. "videoFullRangeFlag": "1",
98. "chromaSampleLocType": "0"
99. }
100. }
101. ]
102. }
103. ],
104. "copyRight": "Conditions that are suitable for this study",
105. "Contact": {
106. "Name": "Bart Kroon",
107. "Company": "Philips",
108. "e-mail": "bart.kroon@philips.com",
109. "generation": "provided by contact"
110. }
111. }

# Annex C: Reference Sequences

## C.1 Introduction

This annex provides a summary of candidate reference sequences that where discussed to be potentially suitable for one or multiple of the scenarios introduced in clause 6 of this Technical Report. For each candidate reference sequence, at least the following information is provided.

- A summary of the sequence characteristics

- A screenshot of the sequence

- Source sequence properties

- Information where the source sequence is hosted

- Copyright and license information

The content is provided in JSON files here: [https://dash-large-files.akamaized.net/WAVE/3GPP/Beyond2D/ReferenceSequence](https://dash-large-files.akamaized.net/WAVE/3GPP/5GVideo/Beyond2D). The format of the reference sequences follows the proposed format in Annex B.2.

The sequences are summarized here: https://dash-large-files.akamaized.net/WAVE/3GPP/Beyond2D/ReferenceSequences/sequences.csv.

Annex <X> (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2024-04 | SA4#127-bis | S4-240825 |  |  |  | Initial Version | 0.0.1 |
| 2024-05 | SA4#128 | S4-240947 |  |  |  | Updated version based on SA4-post 127-bis, 24,May,2024 | 0.0.2 |
| 2024-05 | SA4#128 | S4-241319 |  |  |  | Update style and include agreed content in S4-241266, S4-241336 and S4-241318 | 0.0.3 |
| 2024-08 | SA4#129-e | S4-241491 |  |  |  | Updated version based on agreed Tdoc S4aV240023, S4aV240040  In SA4-post 128 meeting. | 0.0.4 |
| 2024-08 | SA4#129-e | S4-241721 |  |  |  | Updated version based on agreed Tdoc S4-241708, S4-241709 and S4-241710 during SA4#129-e meeting. | 0.1.0 |
| 2024-11 | SA4#130 |  |  |  |  | Updated version based on agreed Tdoc S4aV240062 during SA4-post 129-e meeting. | 0.1.1 |