PHILIPS

# MIV tutorial
# IEEE VCIP 2021

Part 1

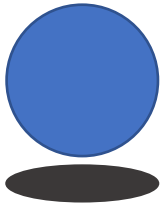
Bart Kroon, bart.kroon@philips.com

# What is immersive video?

# Immersive video gives 6 degrees of freedom

**Viewer position**

- The viewer may:
  - Translate left/right
  - Translate up/down
  - Translate forward/back
- Within a *3D viewing space*



**Viewer orientation**

- The viewer may:
  - Yaw left/right
  - Pitch up/down
  - Roll left/right
- Within the *field of view*


Immersive → *6DoF viewing space*

Video → *time axis*

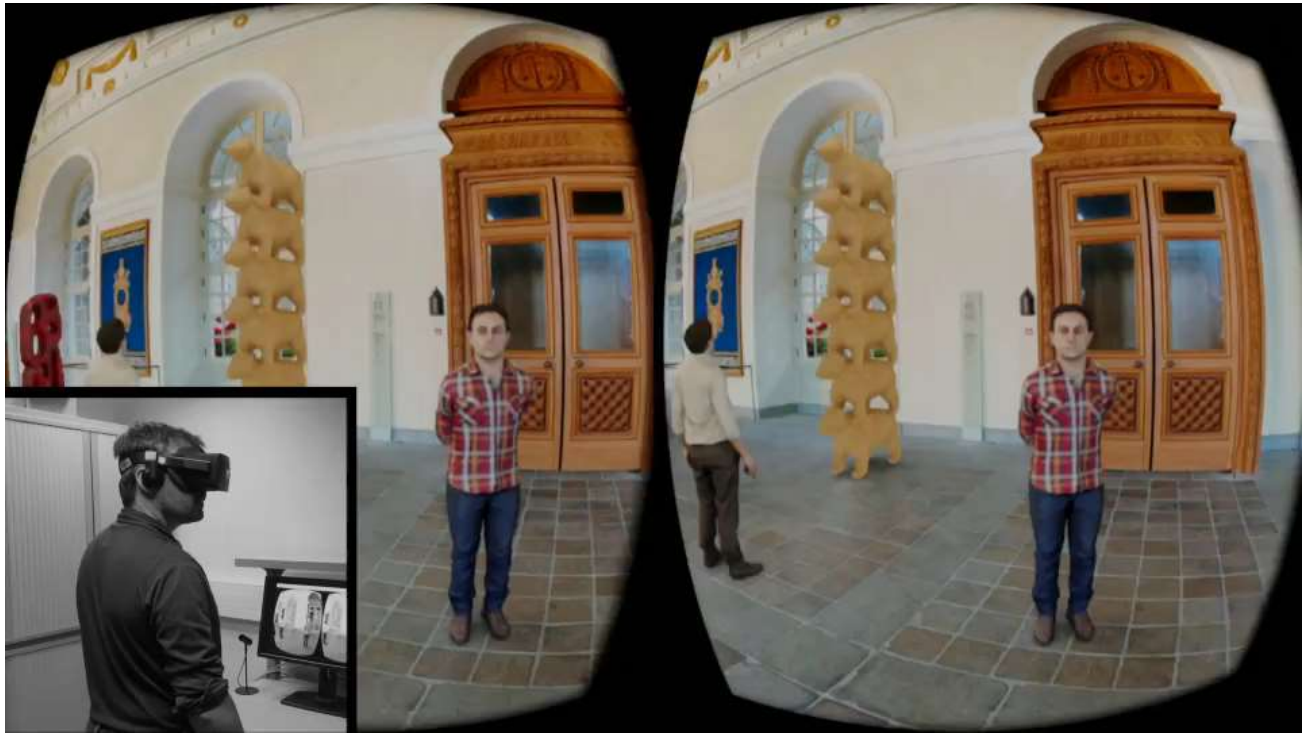
# Use case:
# Virtual reality

**Applications**: Cinematic, Educational, Training, Medical, Professional, Telepresence

**Level of immersion**: Sitting/standing, Some steps

**Experience**: be somewhere else

**Client device**: Closed HMD, Eye-tracked 3D



# Use case:
# Free navigation

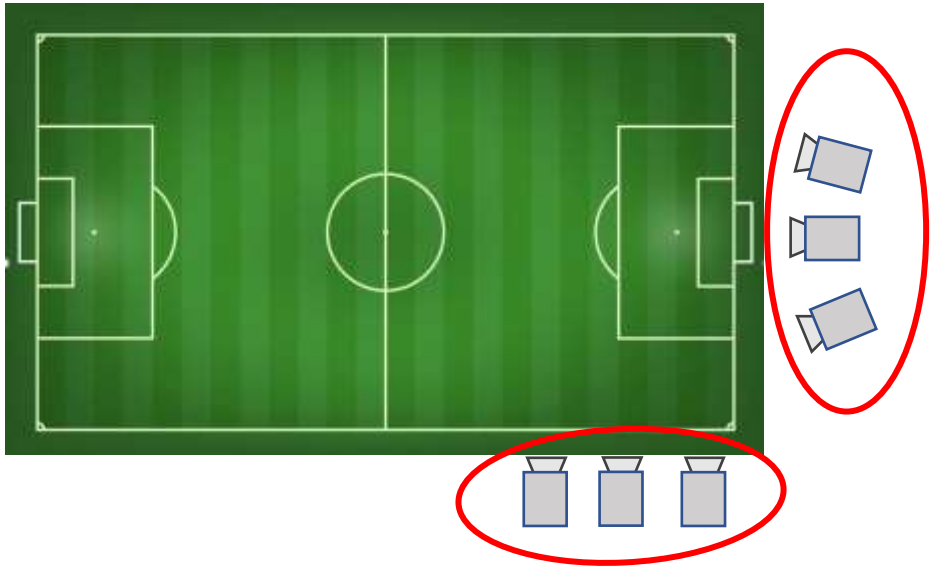
**Applications**: Sports (live, replays), Educational, Training, Medical, Professional, Conferencing

**Level of immersion**: Sitting/standing, Virtual window, Outside-in

**Experience**: Ability for a user to change the viewpoint

**Client device**: Phone, Tablet, Laptop/PC, TV, Eye-tracked 3D

*Possible camera rig on basketball field*



*Possible camera rig positioning near goal on soccer field*

# Use case:
# Light-field display

**Applications**: Cinematic, Educational, Training, Medical, Professional, Telepresence, Conferencing

**Level of immersion**: Virtual window

**Experience**: Be somewhere else, Ability to change viewpoint

**Client device**: Eye-tracked 3D, Light-field display

# Use cases of immersive video

## Application domains

**Sports (live, replays)**
**Cinematic**
**Educational**
**Training**
**Medical**
**Professional**
**Telepresence**
**Conferencing**

## Level of immersion

**Sitting/standing** [VR]

**Virtual window** [VR, FN]
**Some steps** [VR, FN]
**Outside-in** [AR, XR, FN]

**Full 6DoF** [VR, XR, FN]

## Experience

**Virtual reality** (VR)
(be somewhere else)

**Augmented reality** (AR)
(bring something here)

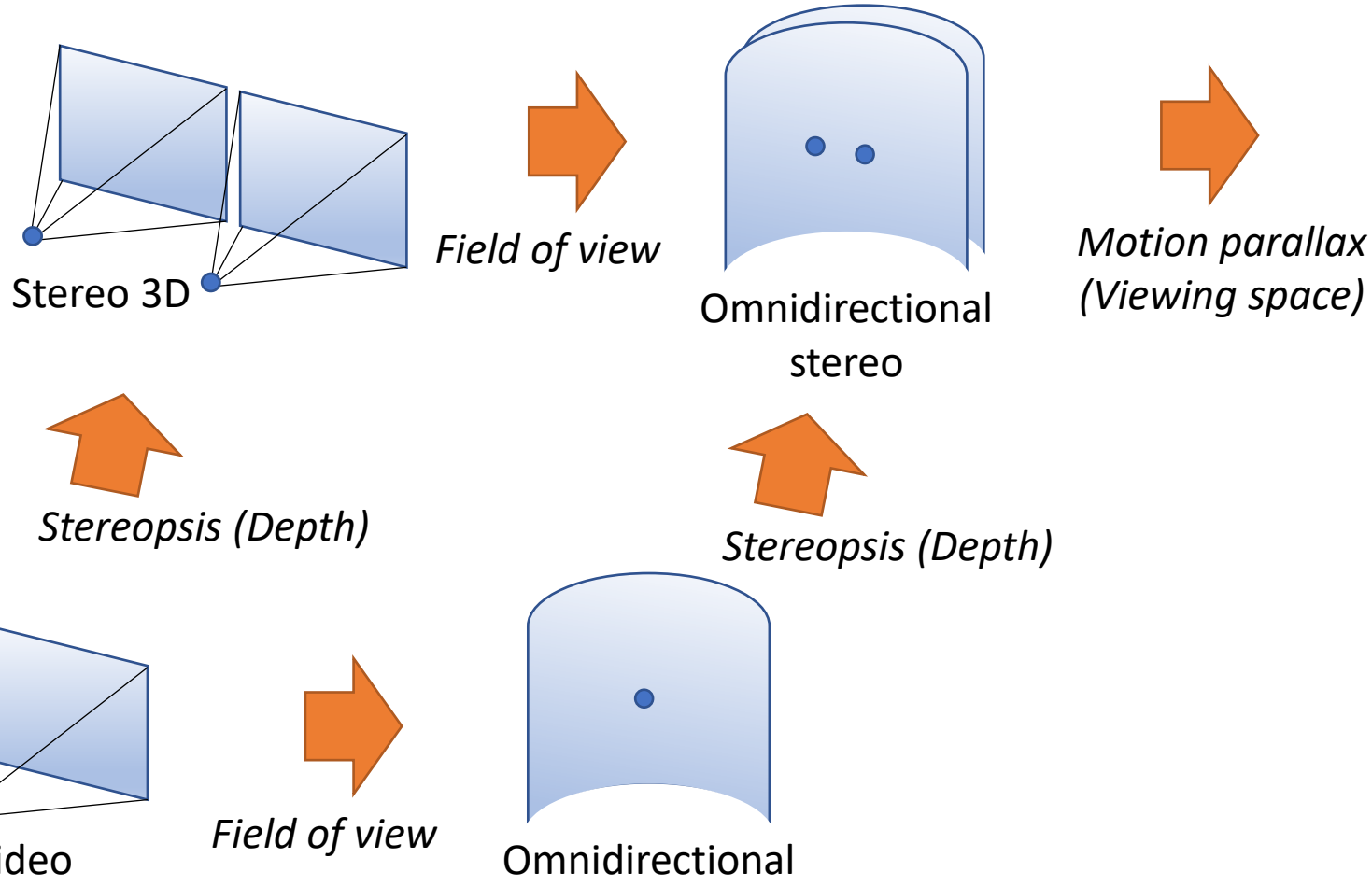
**Mixed reality** (XR)
(interaction between physical and virtual scene elements)

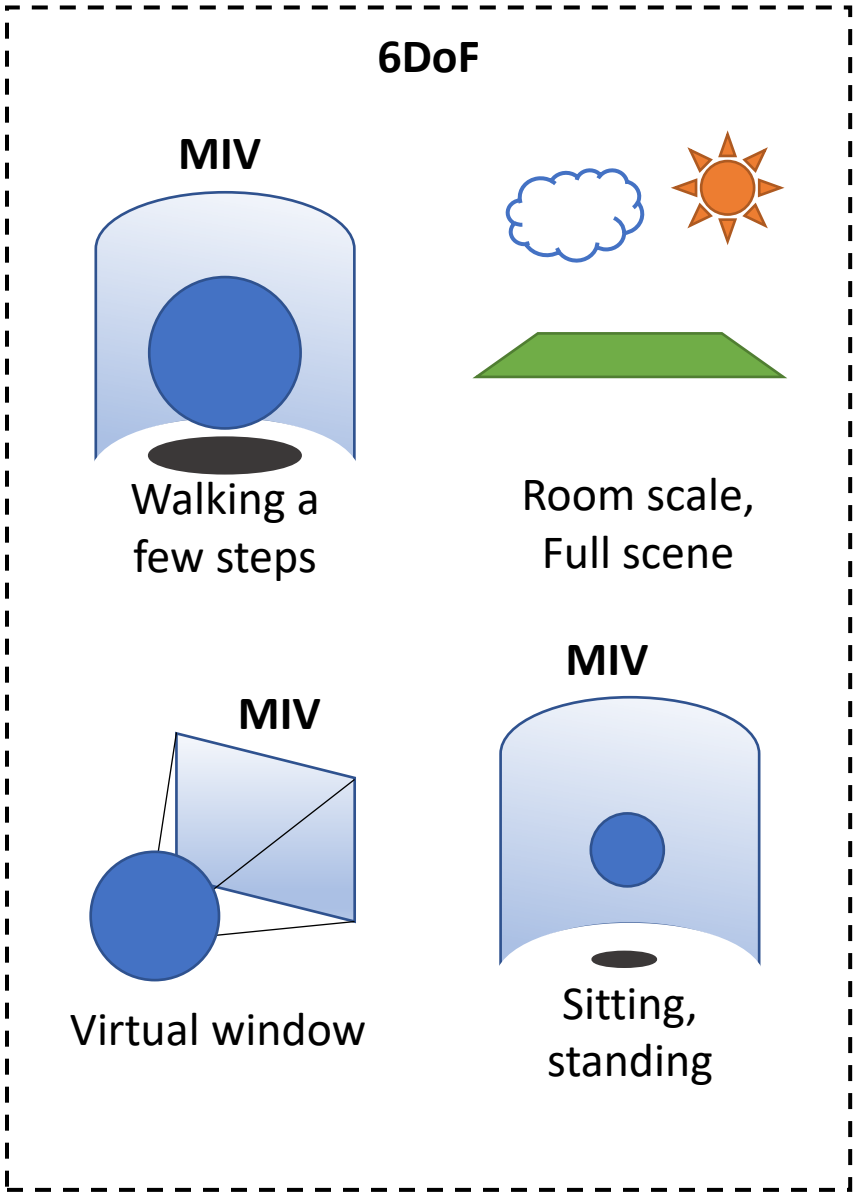
**Free navigation** (FN)
(ability for a user to change the viewpoint)

## Client device

**Phone** [AR, XR, FN]
**Tablet** [AR, XR, FN]
**Laptop/PC** [FN]
**Television** [FN]
**Closed HMD** [VR]
**See-trough HMD** [XR]
**Eye-tracked 3D** [VR, FN]
**Light-field display** [VR, FN]

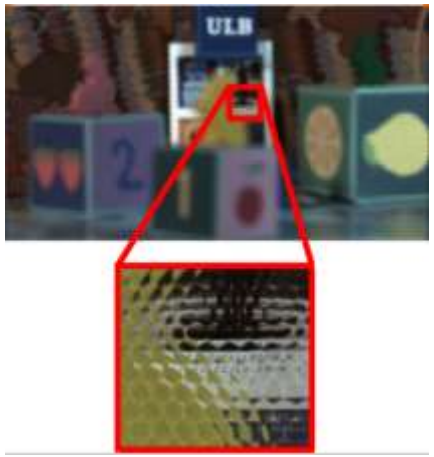
# Levels of immersion



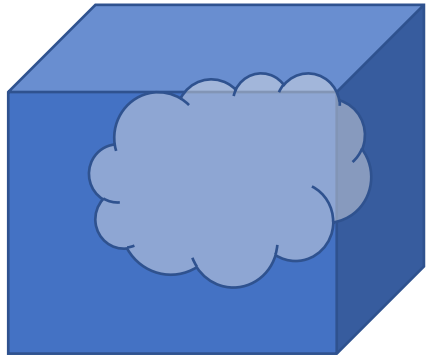
# Possible representations for immersive video

**Densely sampled light field** [rays]



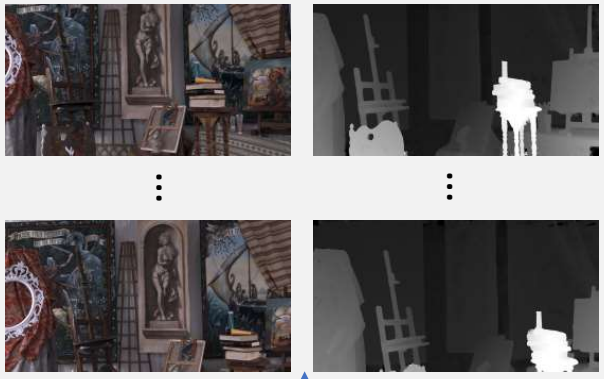

Source: Université Libre Bruxelles

**Densely sampled volume** [voxels]



**MIV**

**Multiview + depth (MVD)** [pixels]

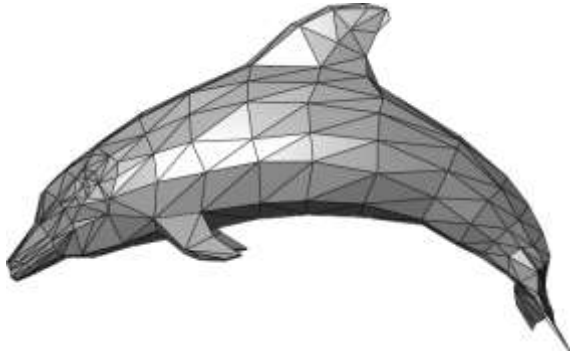


**Multi-planar (MPI / MSI)** [pixels]

Source: Interdigital

**Point cloud** [points]

**Mesh** [polygons]

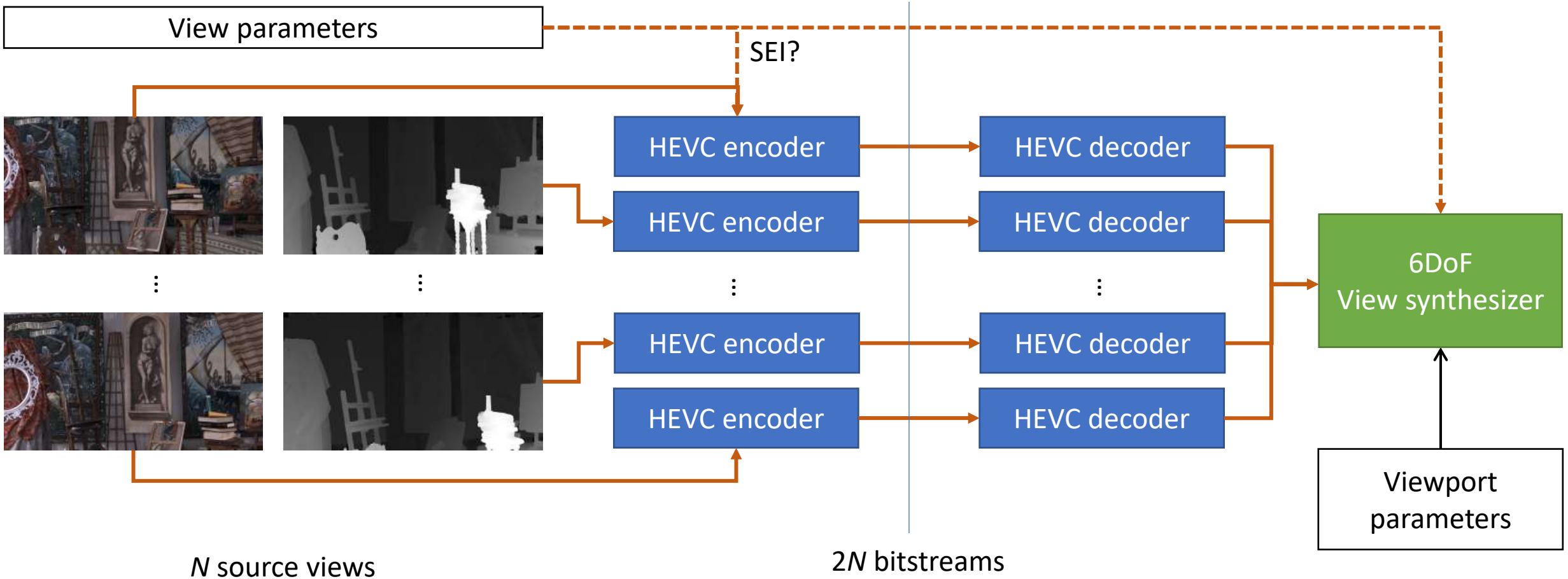


https://en.wikipedia.org/wiki/File:Dolphin_triangle_mesh.png

# Existing codecs

# Multicast HEVC

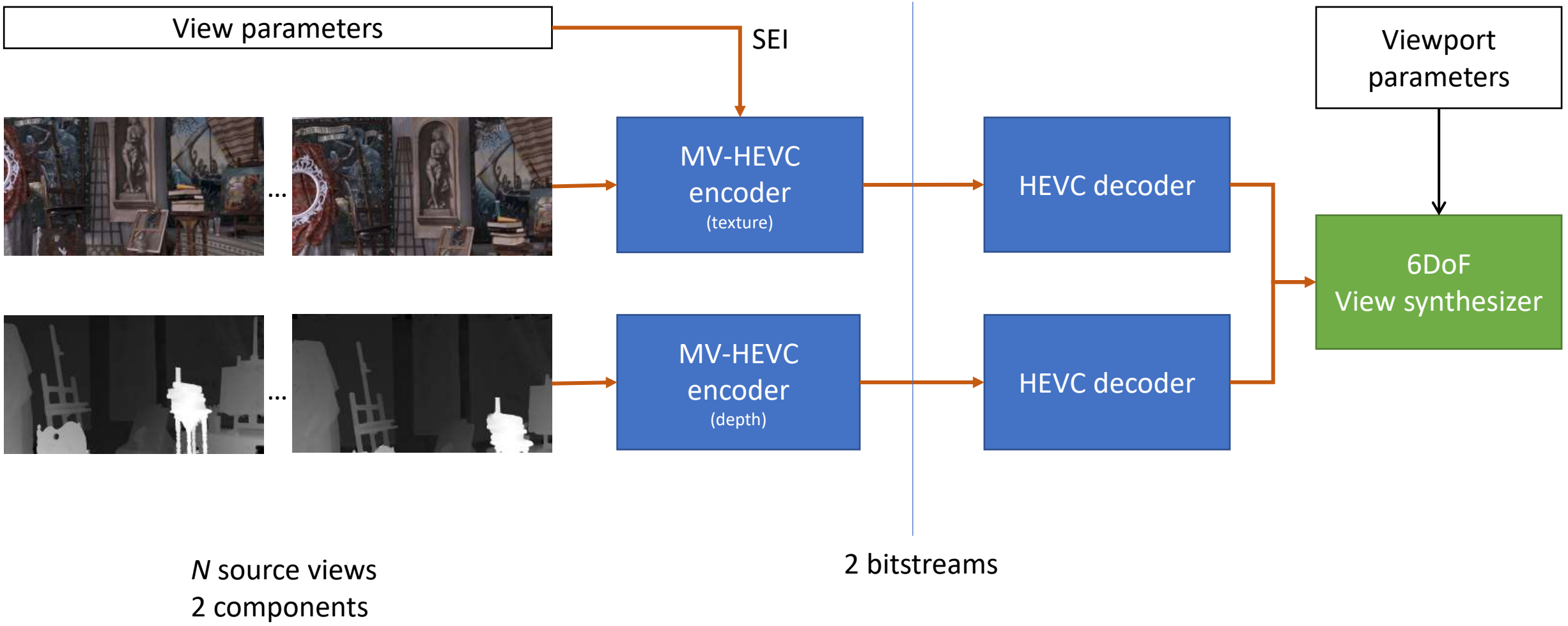
ISO/IEC 23008-2 | ITU-T H.265, also other 2D video codecs




SEI = Supplemental Enhancement Information

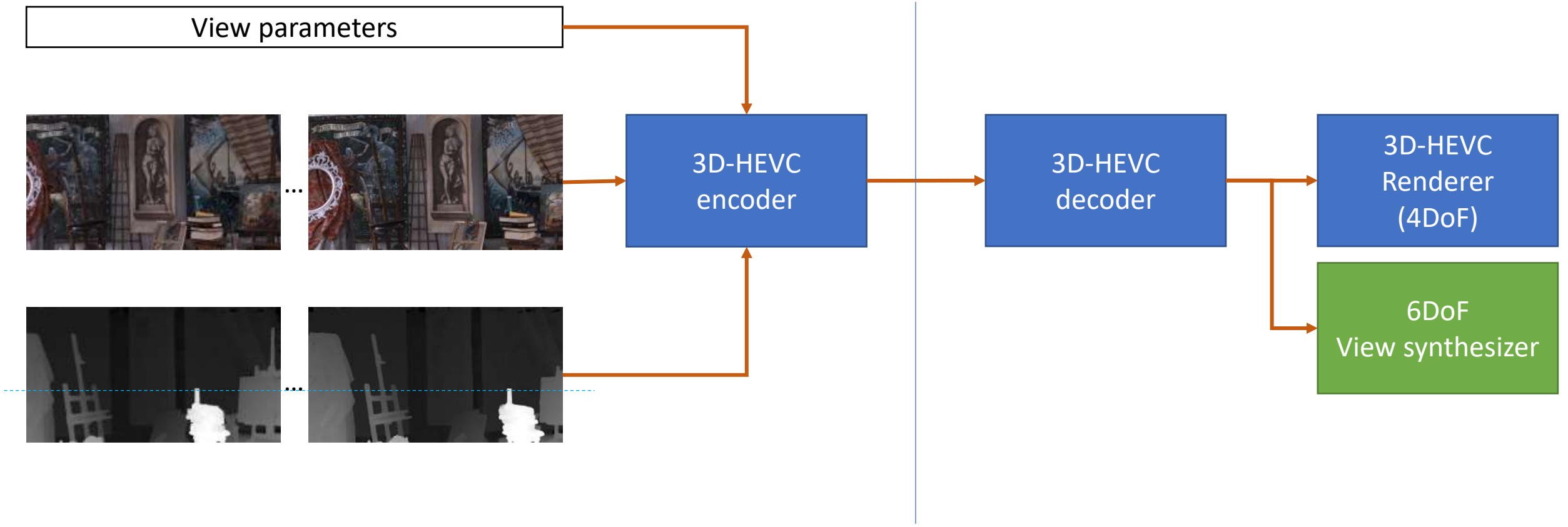
# Multiview HEVC (MV-HEVC)

## Annex G of ISO/IEC 23008-2 | ITU-T H.265



# 3D-HEVC

## Annex I of ISO/IEC 23008-2 | ITU-T H.265



# MPEG Immersive video (MIV)

ISO/IEC 23090-12

# How it started…

- MPEG-I Phase 1b requirements: **October 2018**
  - Head motion parallax
  - Small viewing space (sitting/standing)
- “pixels” → Investigation by MPEG video coding **January 2018**
- Creating of an anchor based on multicast HEVC and a purposely built view synthesizer (RVS)
- Call for Evidence → provided by Technicolor
- Call for Proposals → 5 proposals from 7 organizations **March 2019** (ZJU, PUT/ETRI, Technicolor/Intel, Nokia, Philips)

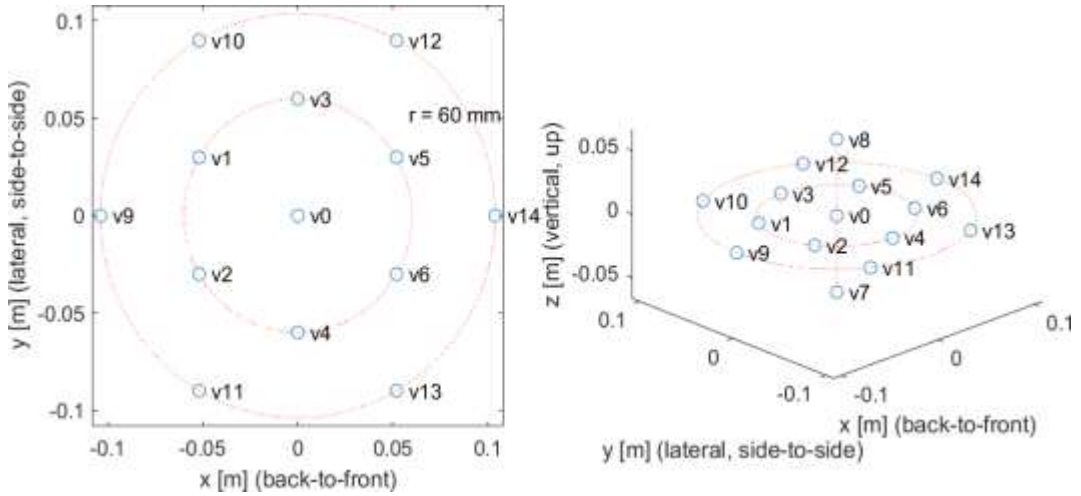
# Introduction & scope

**Introduction** [MIV] was developed to support compression of immersive video content, in which a real or virtual 3D scene is captured by multiple real or virtual cameras. [MIV] enables storage and distribution of immersive video content over existing and future networks, for playback with 6 degrees of freedom of view position and orientation.

**Scope** [MIV] specifies the syntax, semantics and decoding processes for MPEG Immersive video (MIV), as an extension of [V3C]. It provides support for playback of a three-dimensional (3D) scene within a limited range of viewing positions and orientations, with 6 Degrees of Freedom (6DoF).

# Example encoder sources

(Rendering of intermediate views)

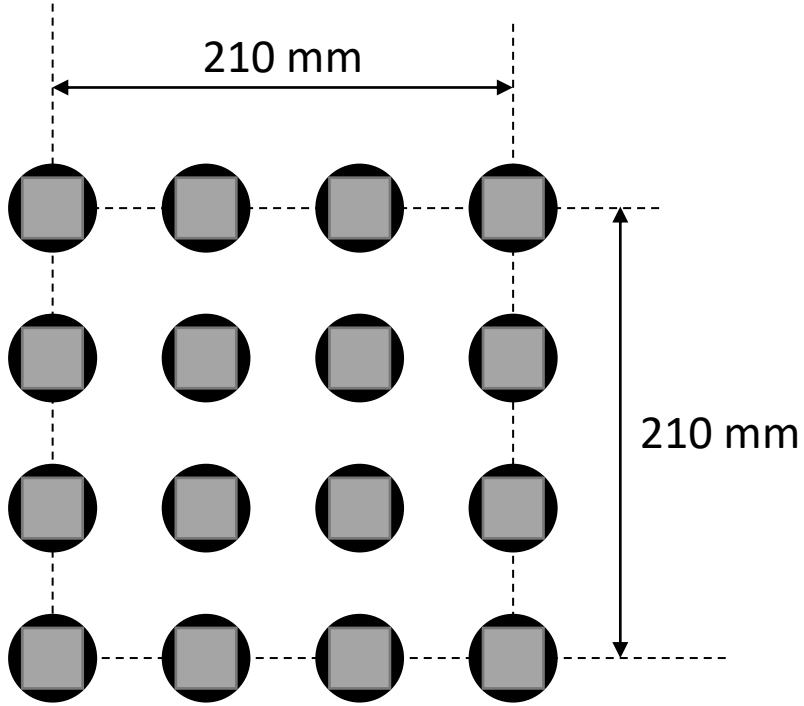



15 views (photorealistic):

- 4K × 2K
- 360° equirectangular projection (ERP)
- Geometry (=depth range)
- Texture attribute (YCbCr)

# Example encoder sources

(Rendering of intermediate views)

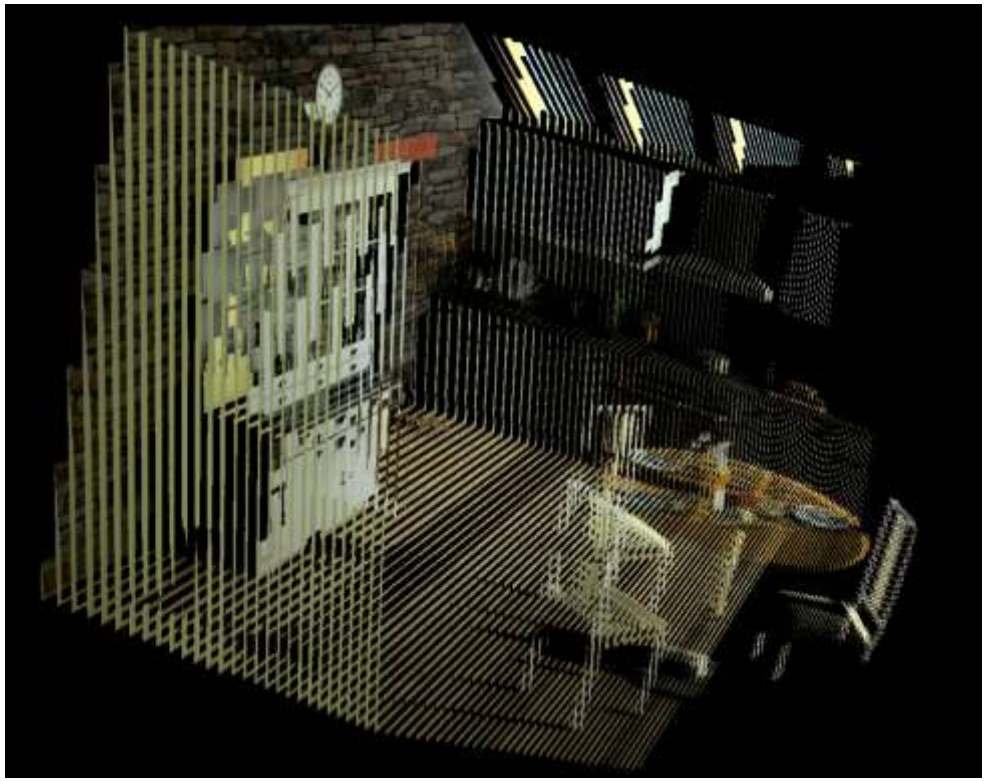



16 physical cameras:

- 2K × 1K
- Perspective projection
- Geometry (=depth range)
- Texture attribute (YCbCr)

# Example encoder sources

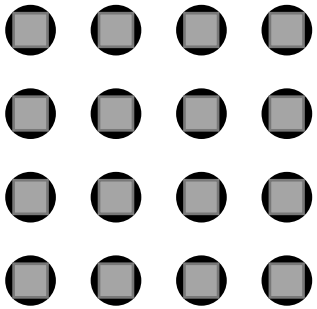
**Multi-plane image (MPI)**



**Multi-sphere image (MSI)**

# MIV codec model

Transform the source to leverage any existing 2D video codec for 6DoF video



Original

MIV encoder

V3C bitstream

- Video data
- Atlas data
- View params

MIV decoder/ renderer

Reconstruction

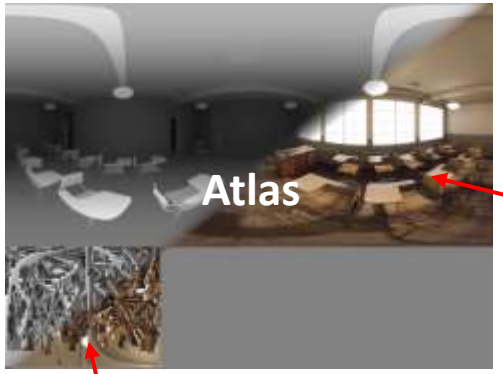
Viewing space

Multi-view or multi-plane video

- Geometry (G) or Transparency
- Texture attribute (T)
- View parameters

T

G



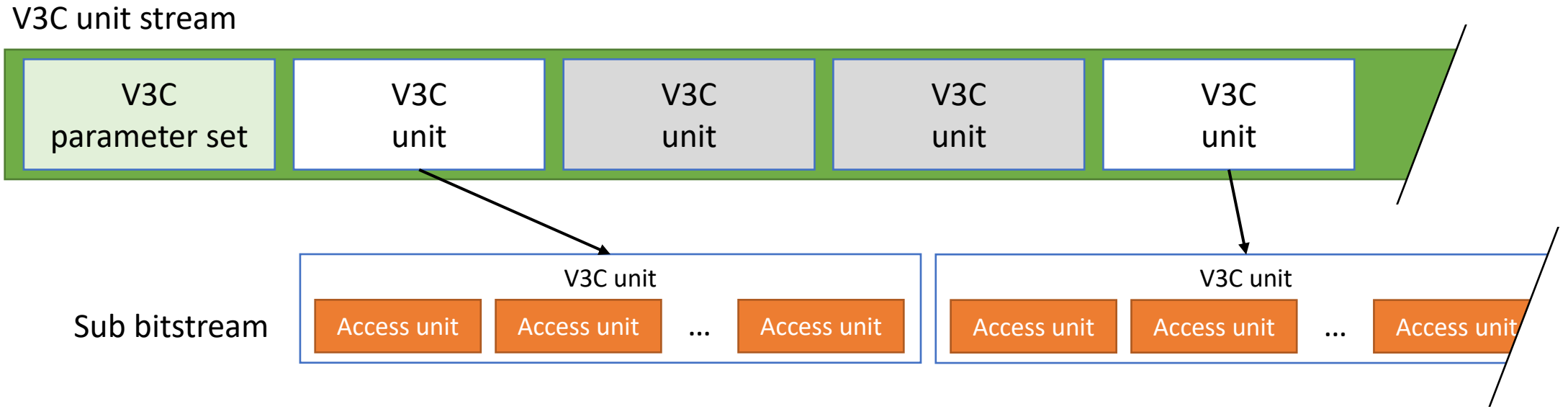

Complete (basic) view

Patches from additional views

Viewport video

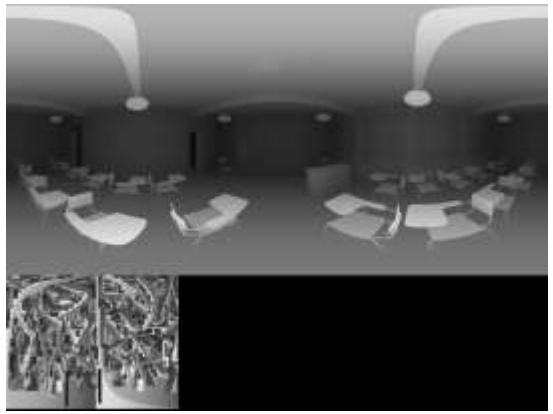
- (Geometry)
- Texture attribute

# Bitstream structure




Sub bitstreams:

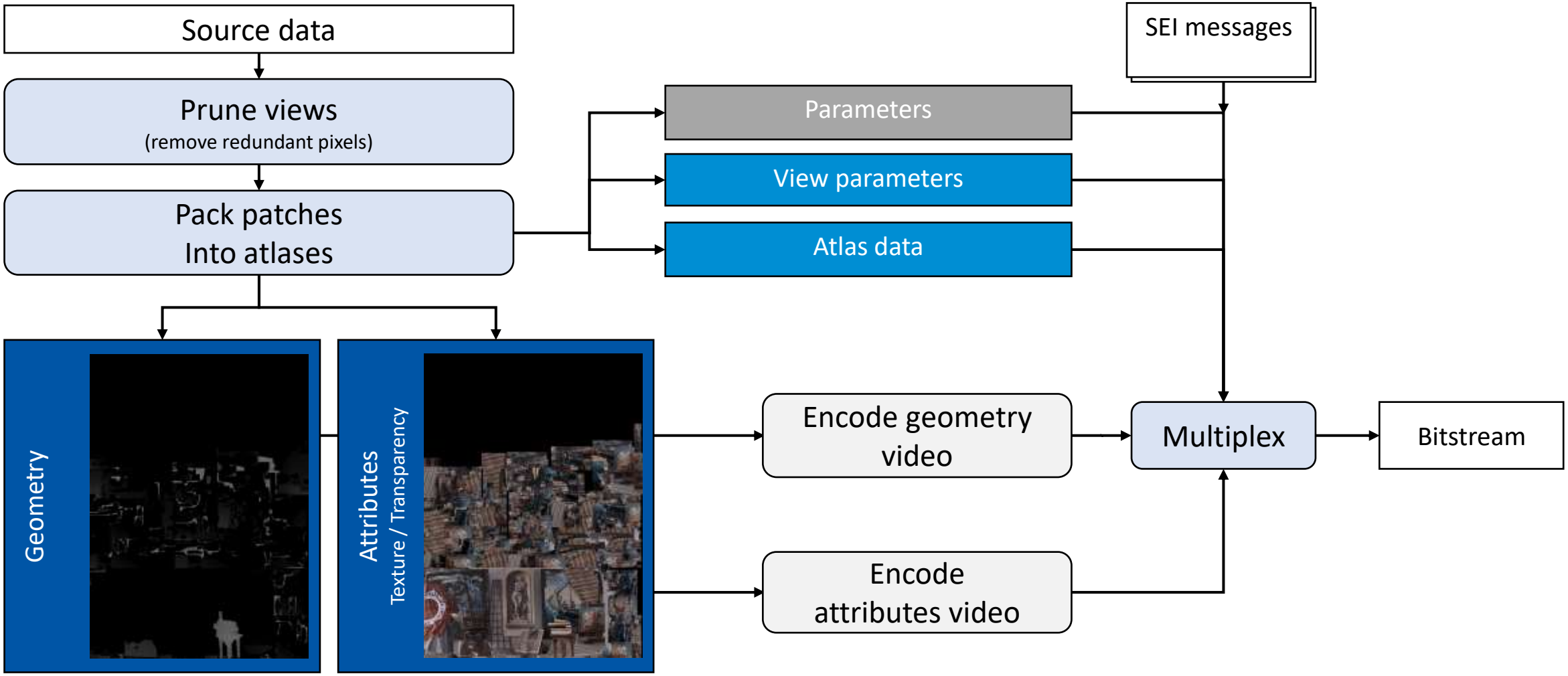
- Common atlas data (has view parameters)
- Multiple atlases:
  - Geometry video data
  - Attribute video data
  - Atlas data (has patch parameters)



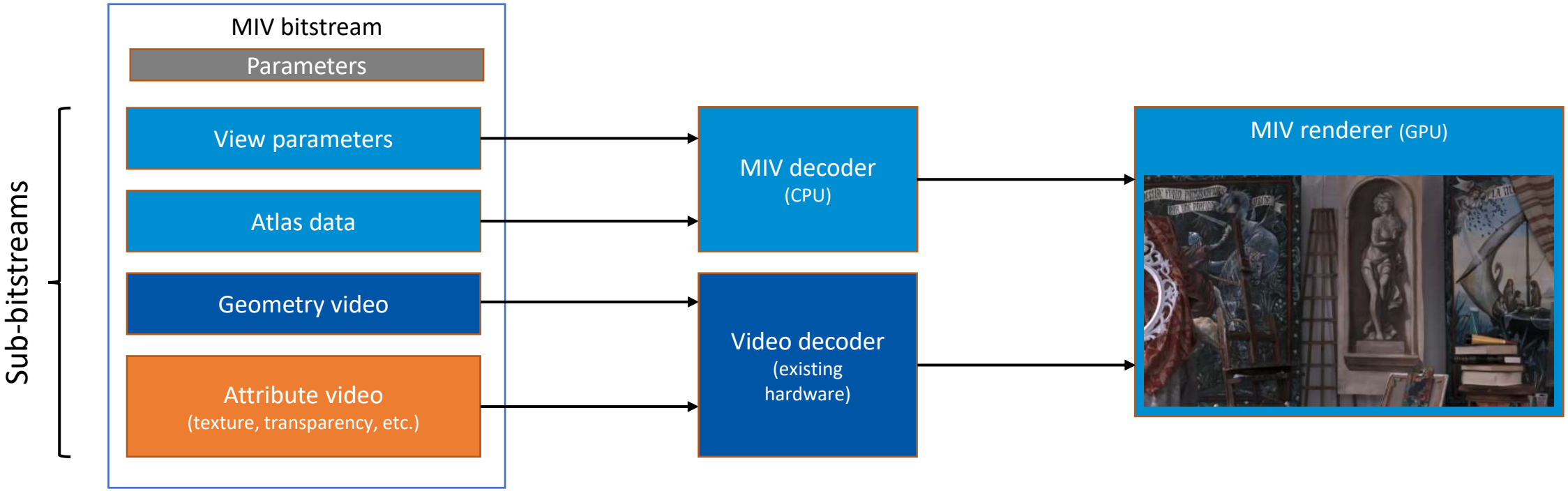
# MIV test model

- Test Model for Immersive Video (TMIV)
- Public software: https://gitlab.com/mpeg-i-visual/tmiv
- Includes:
  - *Example* of an MIV encoder
  - Conforming MIV decoder
  - *Example* of an MIV renderer
  - Extra tools: multiplexer, parser, bitrate report
  - Multiple configurations for experimentation purposes

# TMIV encoder model



# TMIV decoder/renderer model

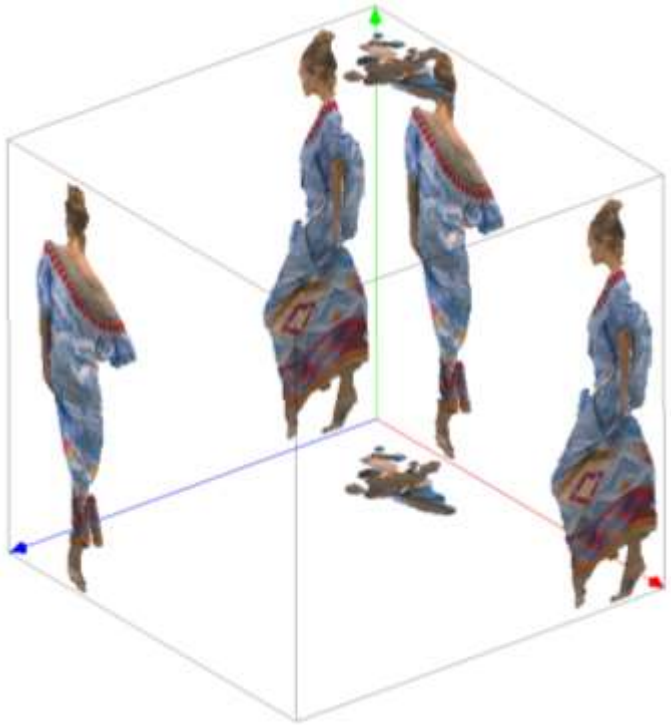


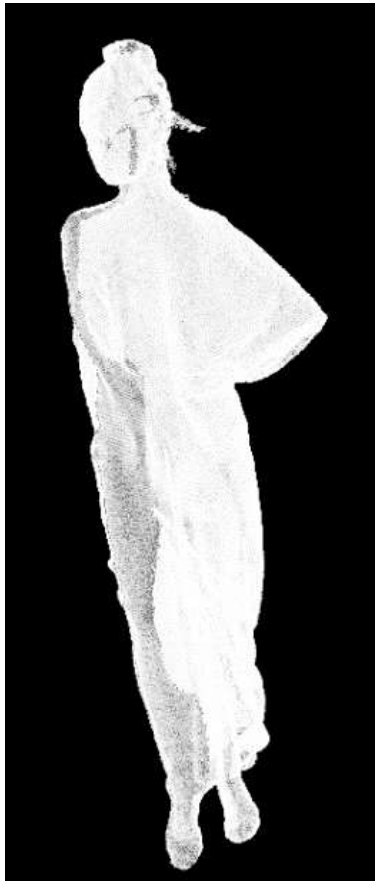
# Related standards

# Video-based Point Cloud Coding (V-PCC)

ISO/IEC 23090-5 Annex H

- Single atlas, but may be tiled
- Point reconstruction information
- Six orthographic projection planes

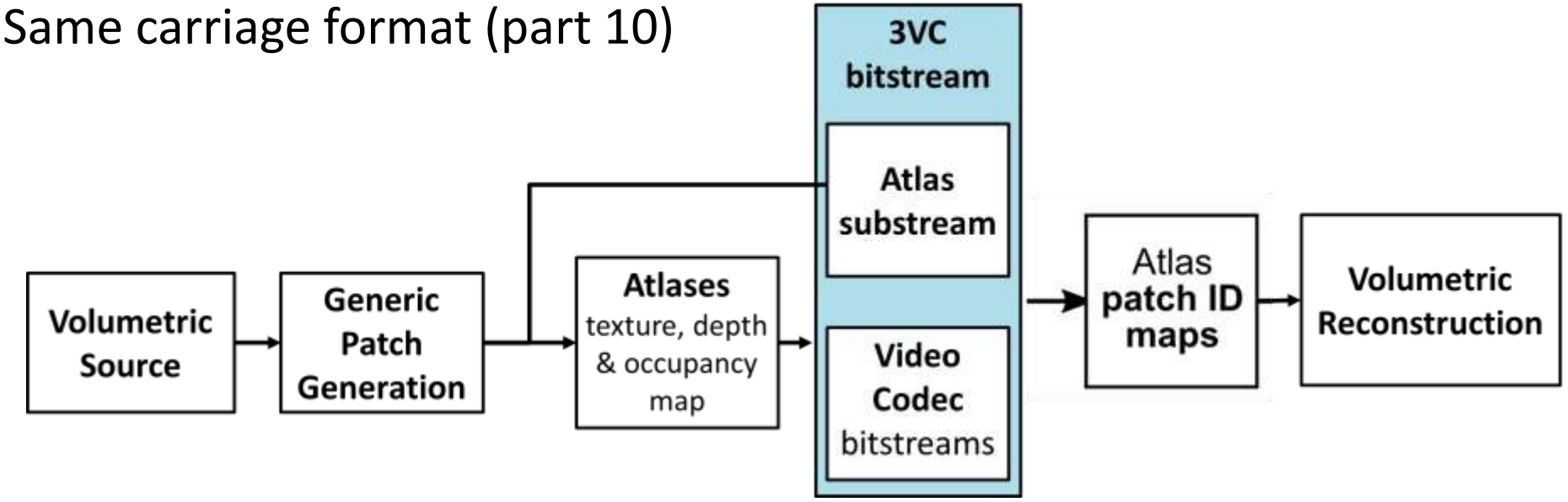




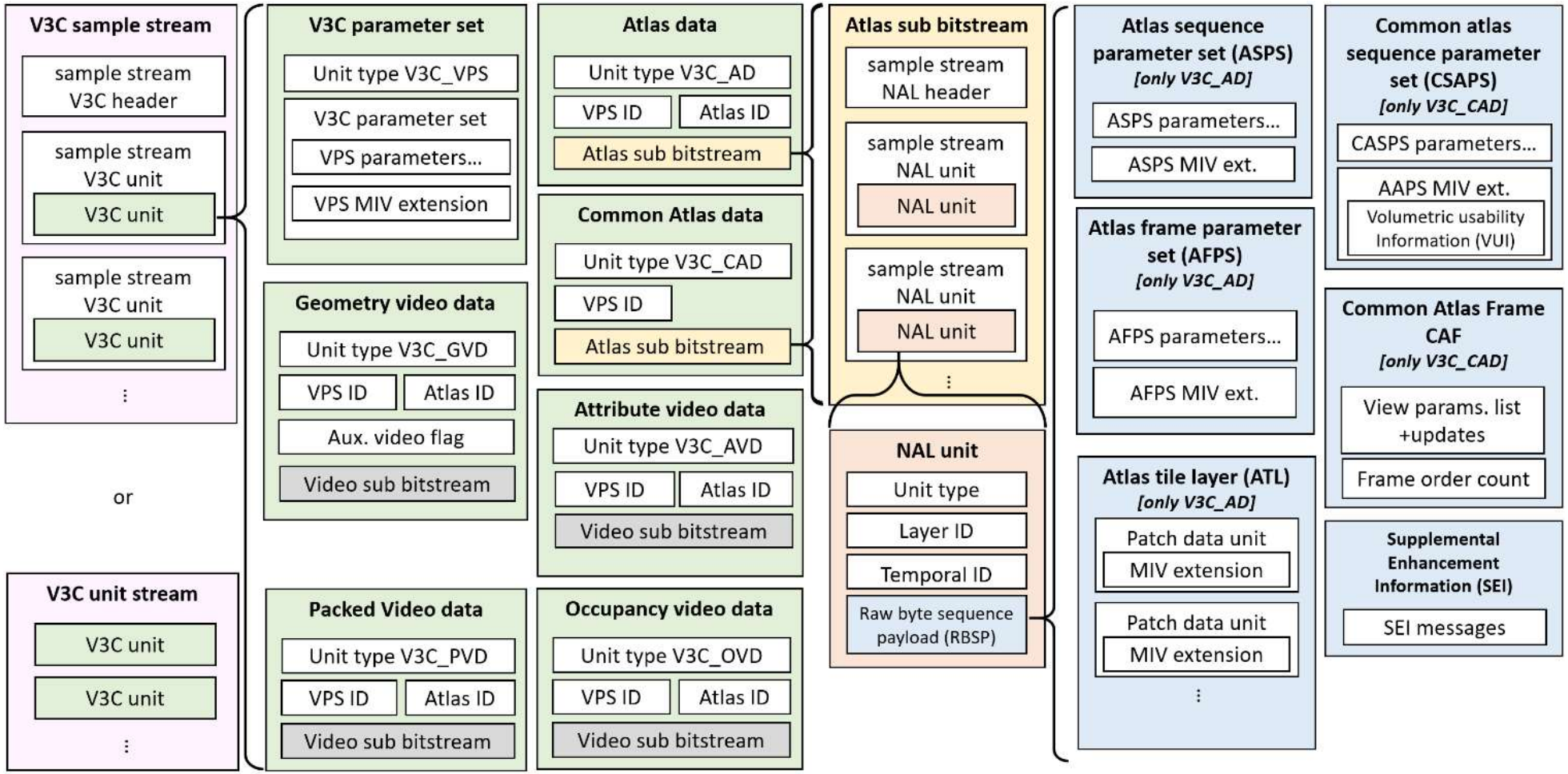
# Video-based Visual Volumetric Coding (V3C)

ISO/IEC 23090-5

- V3C is the (large) intersection of MIV and V-PCC
  - Avoid duplicate specification
  - Reduce implementation effort
  - Same bitstream format
  - Same carriage format (part 10)



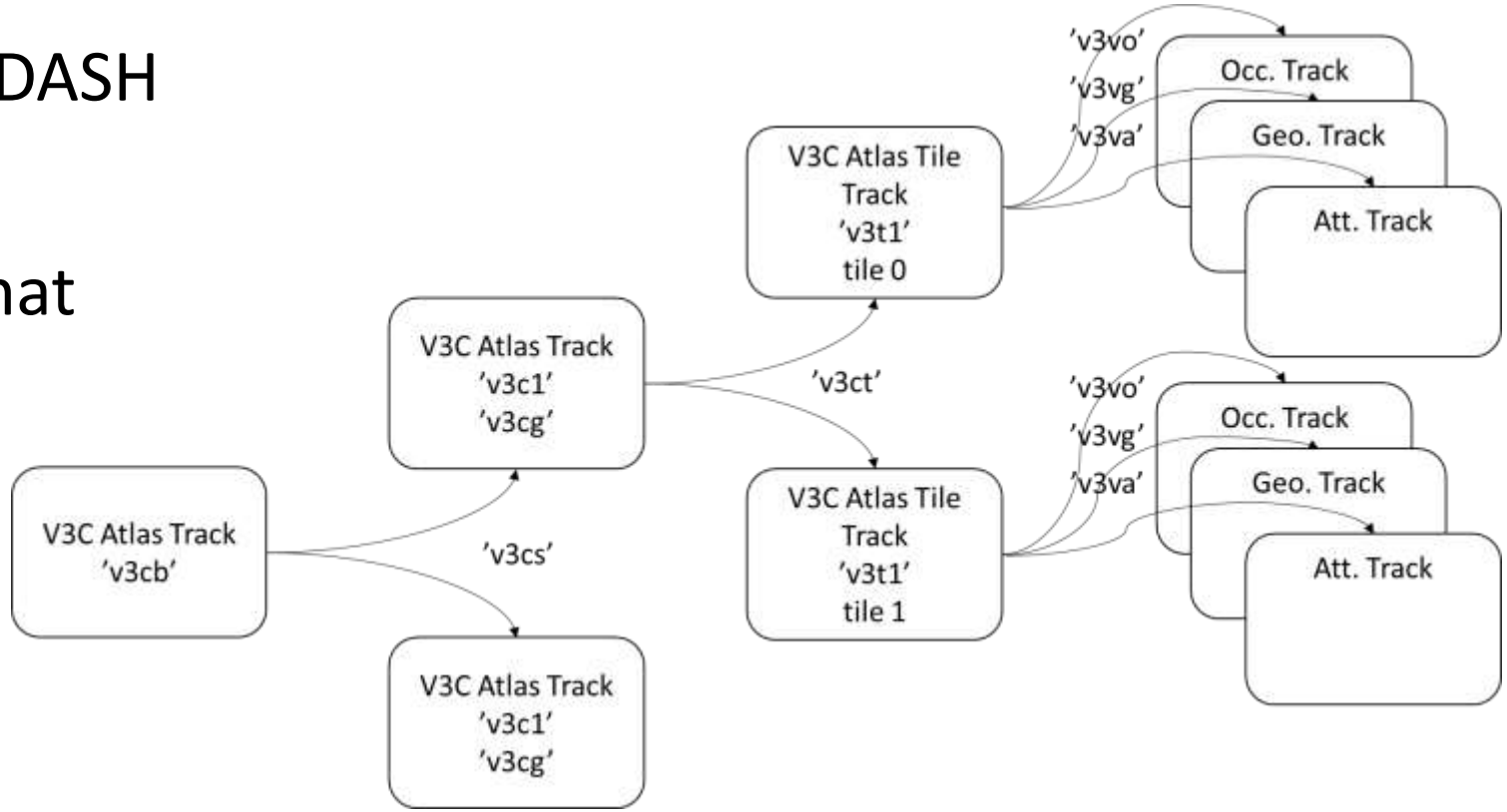
# MIV: Restrictions and extensions on V3C



# Carriage of V3C Data

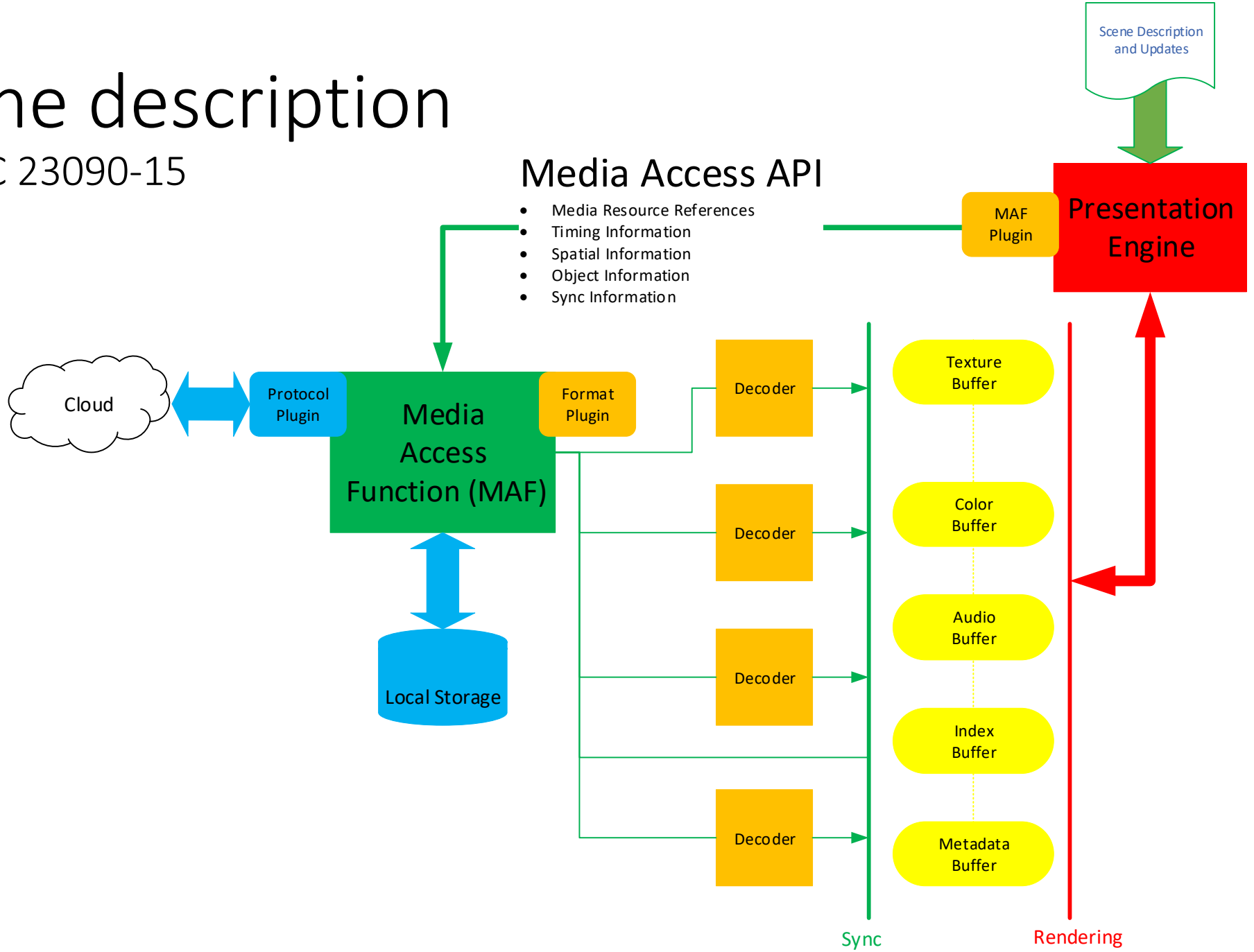
ISO/IEC 23090-10

- Single/multi-track format
- Partial access
- Encapsulation in MPEG DASH
- Encapsulation in MMT
- ISO base media file format



# Scene description

## ISO/IEC 23090-15



# Timeline

| MPEG meeting | V3C + V-PCC 1st edition | V3C 2nd edition | MIV 1st edition | Carriage of V3C 1st edition |
|---|---|---|---|---|
| 131: Jul. 2020 | Final draft (FDIS) | | | |
| 132: Oct. 2020 | | | | |
| 133: Jan. 2021 | | | Draft (DIS) | |
| 134: Apr. 2021 | Publication | | | Final draft (FDIS) |
| 135: Jul. 2021 | | Draft (DIS) | Final draft (FDIS) | |
| 136: Oct. 2021 | Conformance & Verification tests | | | Publication |
| 137: Jan. 2022 | | Final draft (FDIS) | Publication | |
| 138: Apr. 2022 | | | Conformance & Verification tests | |

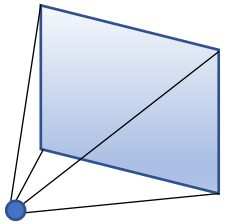
DIS = Draft international standard
FDIS = Final draft international standard

# Features of the standard

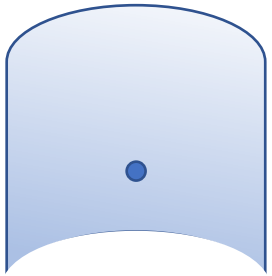
# View parameters

**Projection plane**

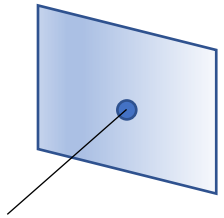

**Intrinsics**



Perspective projection
$(f_u, f_v, p_u, p_v)$

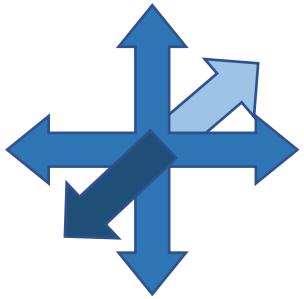


Equirectangular projection
$(\theta_1, \theta_2, \phi_1, \phi_2)$

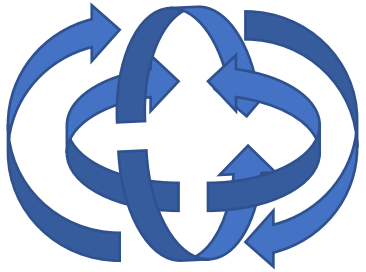


Orthographic projection
(w, h)

**Extrinsics**

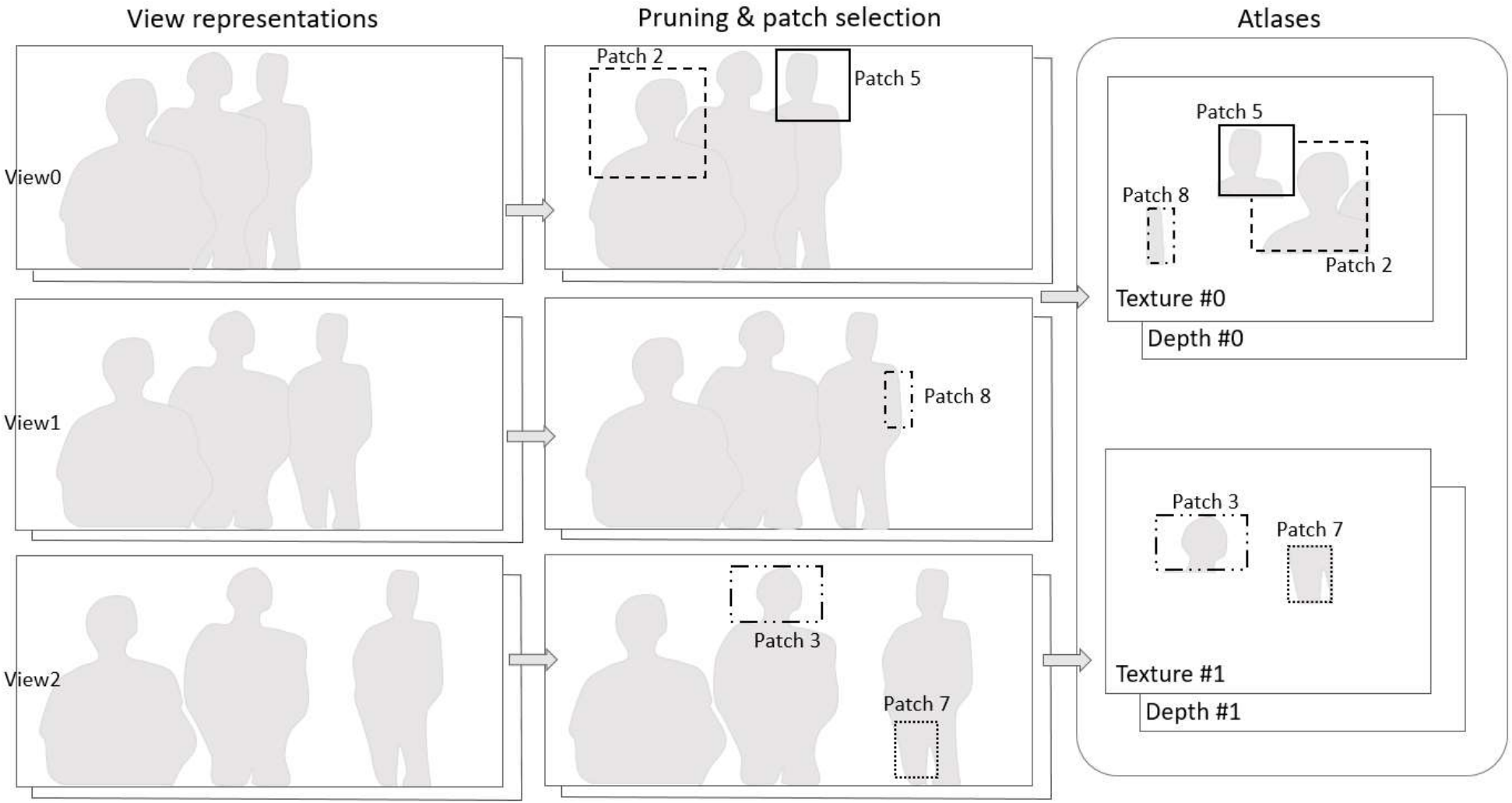


Position
(x, y, z)



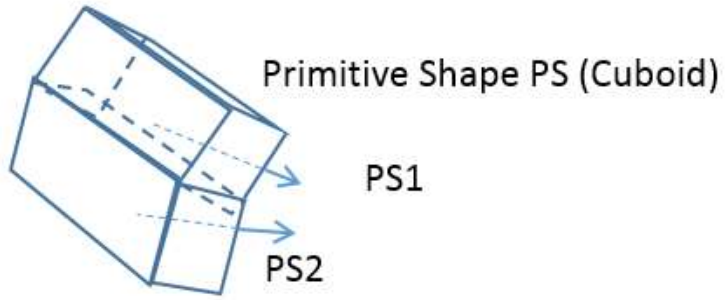
Orientation
$(\alpha, \beta, \gamma)$

# Patch parameters

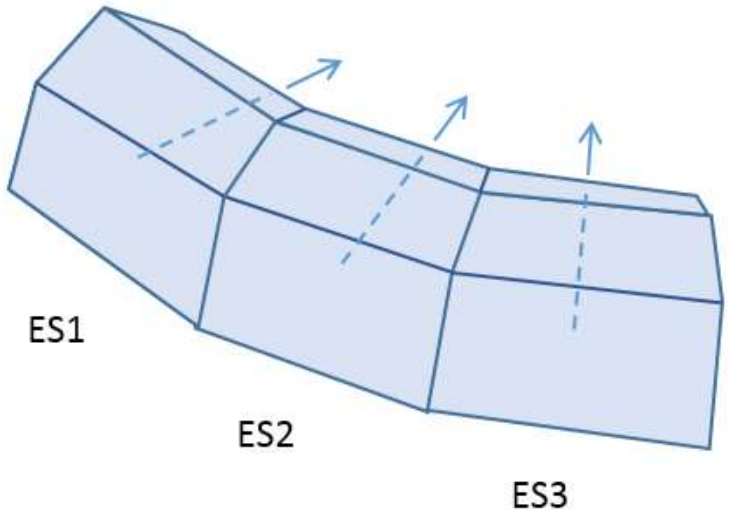


# Viewing space definition

- Definition of a viewing space
  - Spatial (3D)
  - Spatial + field of view (5D)
- Can be as simple as a box or a sphere
- Constructive solid geometry (CSG)
- Signed distance function
  - < 0: Inside
  - = 0: Edge
  - > 0: Outside




Elementary Shape ES= PS1 ∪ PS2

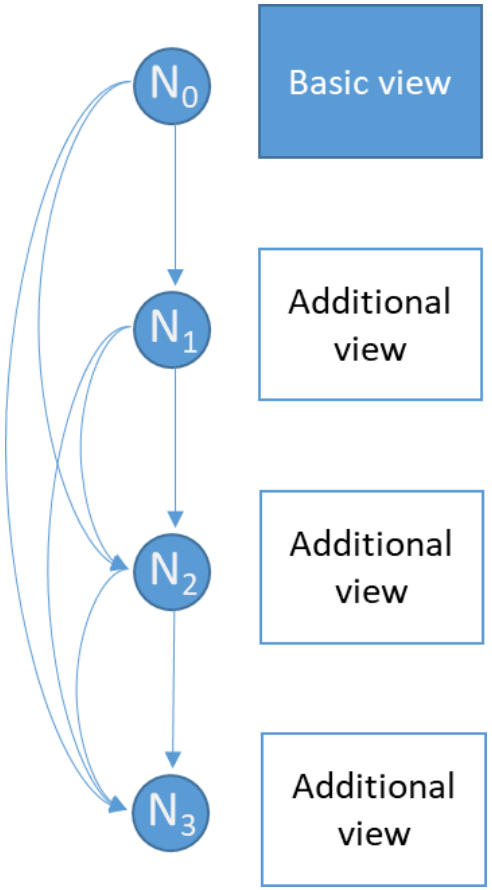



Viewing Space VS= ES1 ∪ ES2 ∪ ES3

# Viewing space handling

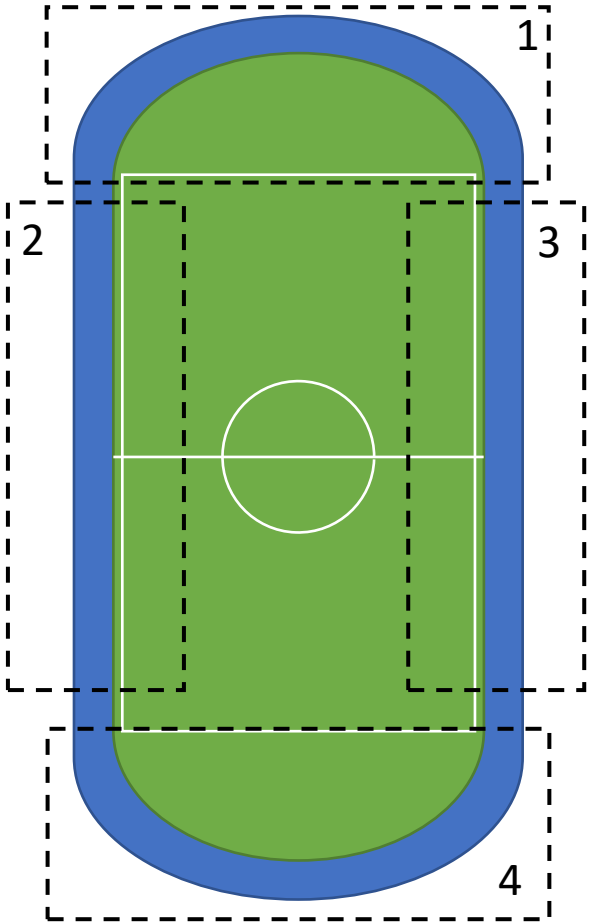
- SEI message to convey the creative intent regarding the viewing space
  - For device: all, HMD, phone, 3D display
  - For application: all, AR, VR, web, scene description
  - What happens when the viewer moves outside the viewing space:
    - Default client behaviour
    - Always render, even when there would be artifacts
    - Fade out
    - Extrapolate scene colours
    - Reset the viewer position
    - Stretch viewer pose to stay within the viewing space
    - Rotate the viewer to stay within the field of view

# View pruning graph




- Provide information on which view was used to prune which other views.
- Useful for rendering algorithms that rely on view blending.

# Group-based encoding




MIV encoder 1

MIV encoder 2

MIV encoder 3

MIV encoder 4

Merge bitstreams

Transmission over network

Select bitstreams

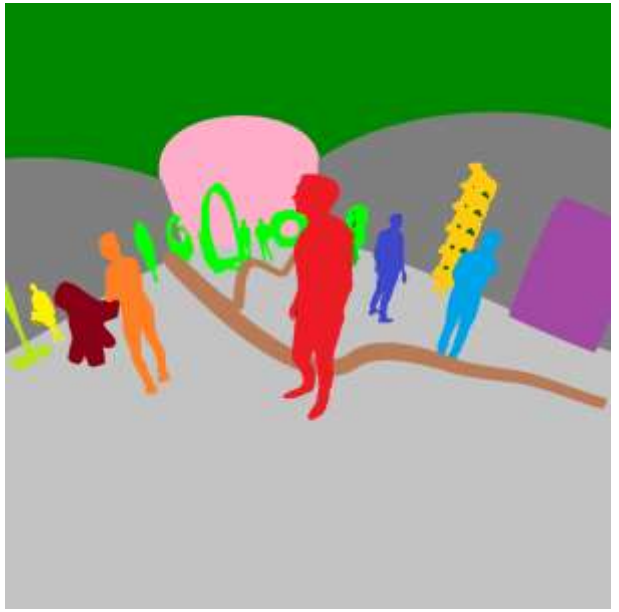
Decode & render

Request video track

# Entity-based coding

Texture attribute

Geometry
(Normalized disparity)



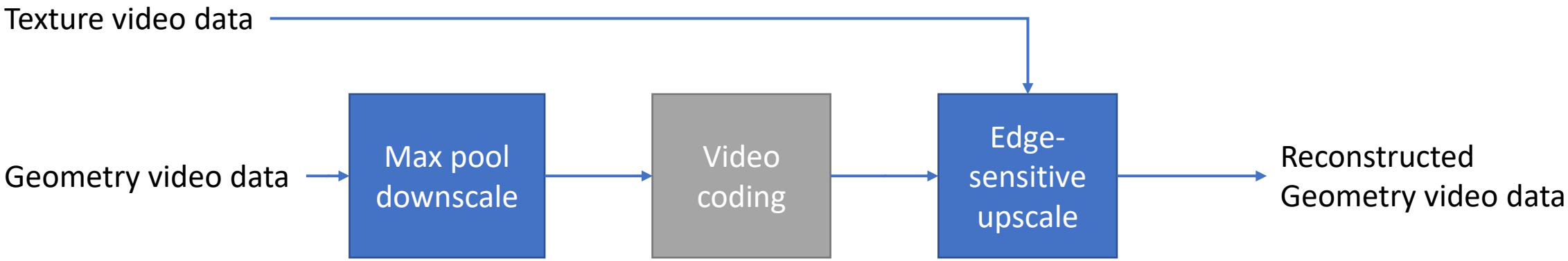
Entity ID map

Not coded as a separate video component → Each patch has an entity ID

# Server-side inpainting

- Move inpainting to the server (more info, more resources)
- Inpaint patches or whole views
- The renderer may use them to fill up disocclusion zones.
- TMIV encoder renders a low-res background from the source views

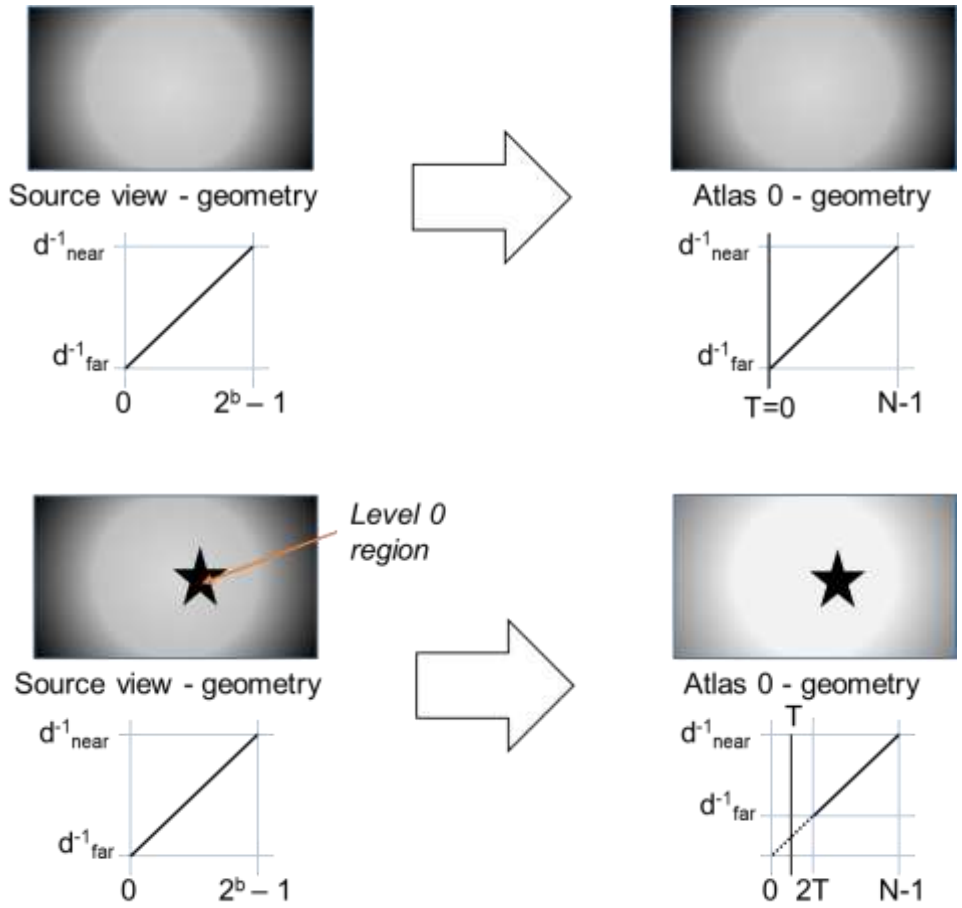
# Geometry scaling




- Video encoders are not designed to encode geometry
- Geometry downscaling enables lower quantization (QP) parameters for a given bitrate
- Preserve thin foreground edges

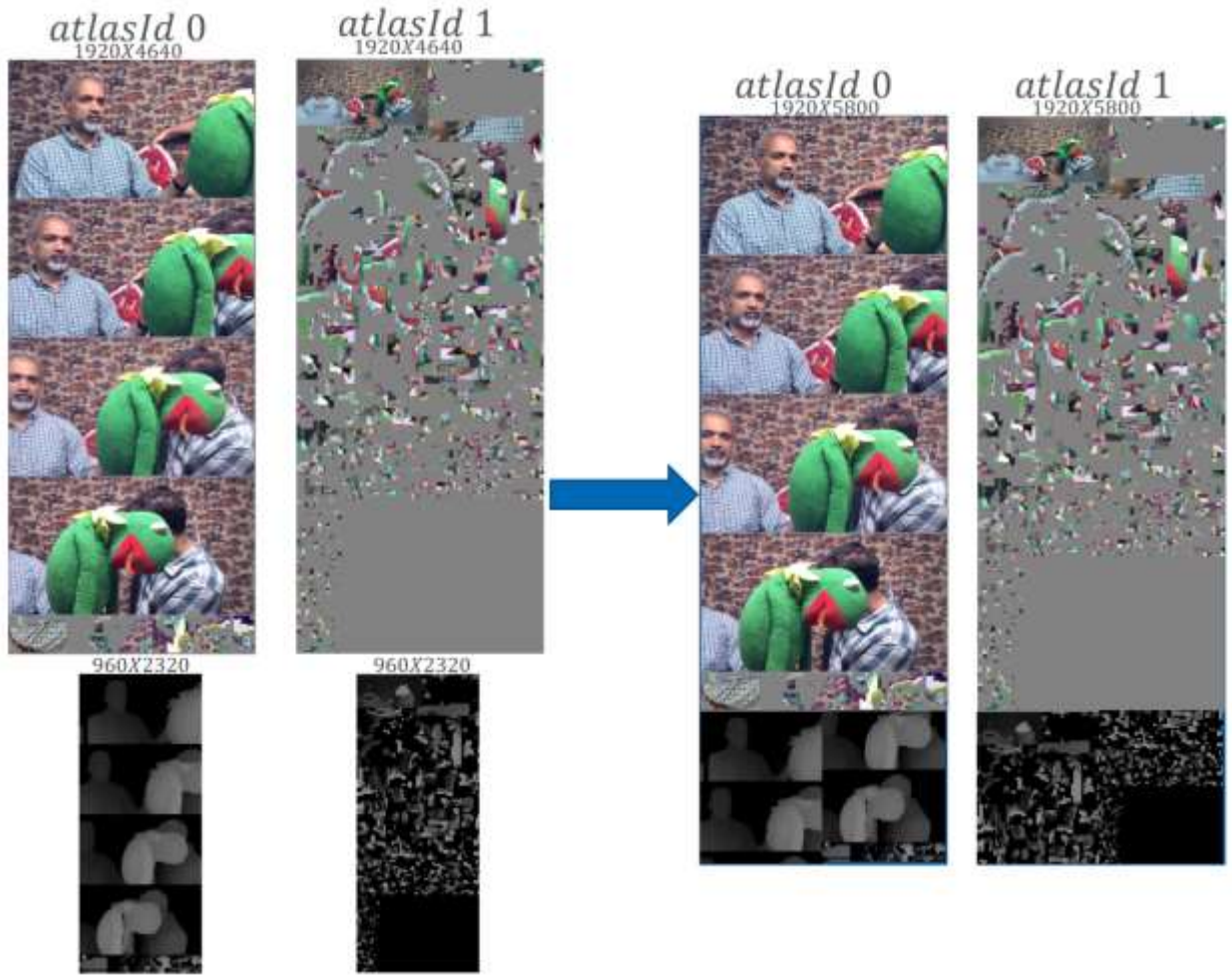
# Representation of occupancy information

- **Full patch occupancy**
  All sample values of a patch are occupant
- **Embedded occupancy**:
  Occupancy embedded in geometry video data
- **Explicit occupancy:**
  Occupancy video data
  Full resolution or subsampled



# Frame packing

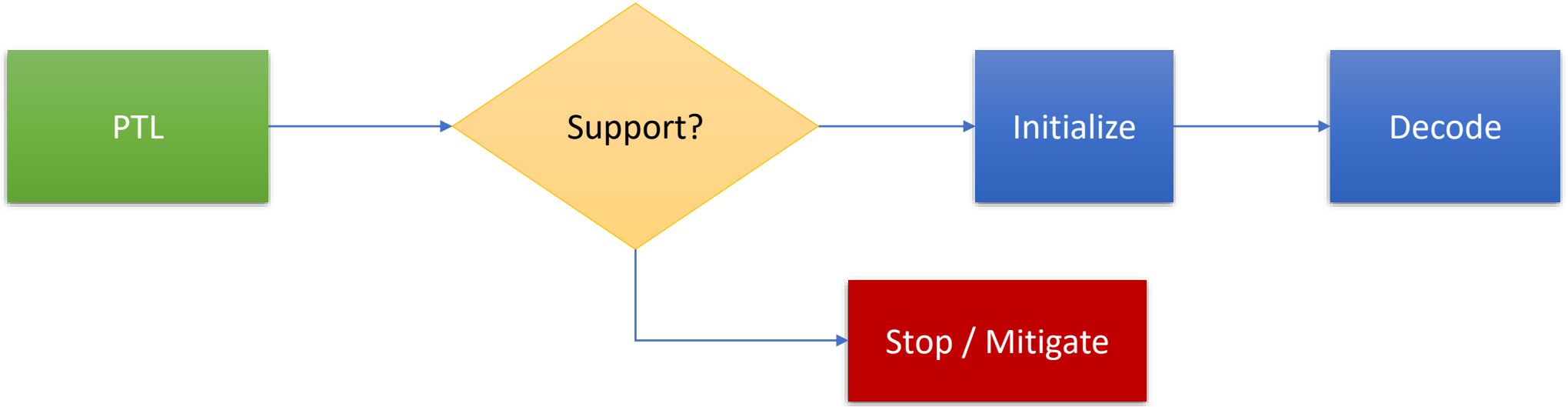
- Reduce the number of video decoder instantiations by packing multiple components into one video frame.
- Works well with (but does not require) HEVC MCTS or VVC sub-pictures.



# Profile and sub-profiles

# Profile, tier and level information

- Profile = subset of the standard
  - Codec group: MPEG-4 AVC, HEVC, VVC, MP4RA
  - Toolset: V-PCC …, MIV …
- Tier = main (to client device), high (production, archival)
- Level = limits on samples, pixels, bits, etc.



# Profile: MIV Main

- Useful for multiview + depth:
  - Computer graphics
  - Natural content with depth est.
- **Geometry video data**
- **Texture video data** (opt.)
- Full or **embedded occupancy**
- Patches and full views

# Real-time immersive video on Android phone

Martijn Govers, Bart Sonneveldt, Hugo Habets, Chris Varekamp, Andy Willems, Bart Kroon

- MIV Main
- Two HEVC Main10 video bitstreams
  - Texture
  - Geometry
- Full views + inpainted background
- Computer graphics and natural content
- One V3C sample stream per second

- Phone with interactivity

# Profile: MIV Extended

- Full or embedded occupancy, or **occupancy video data**
- **Constant depth per patch** or geometry video data
- Texture video data (opt.)
- **Transparency video data** (opt.)
- **Packed video data** (with any of the above video components)

# Immersive video playback on Intel Xe Max

Max Dmitrichenko, Basel Salahieh, Mengyu Chen, Penne Lee, Jill Boyce

- MIV Extended
- Single HEVC Main10 video bitstream
- Packed video with texture and geometry
- Only full views
- Atlas data carried in vendor-specific SEI message

- 2D display with eye tracking

# MIV Extended with Restricted Geometry

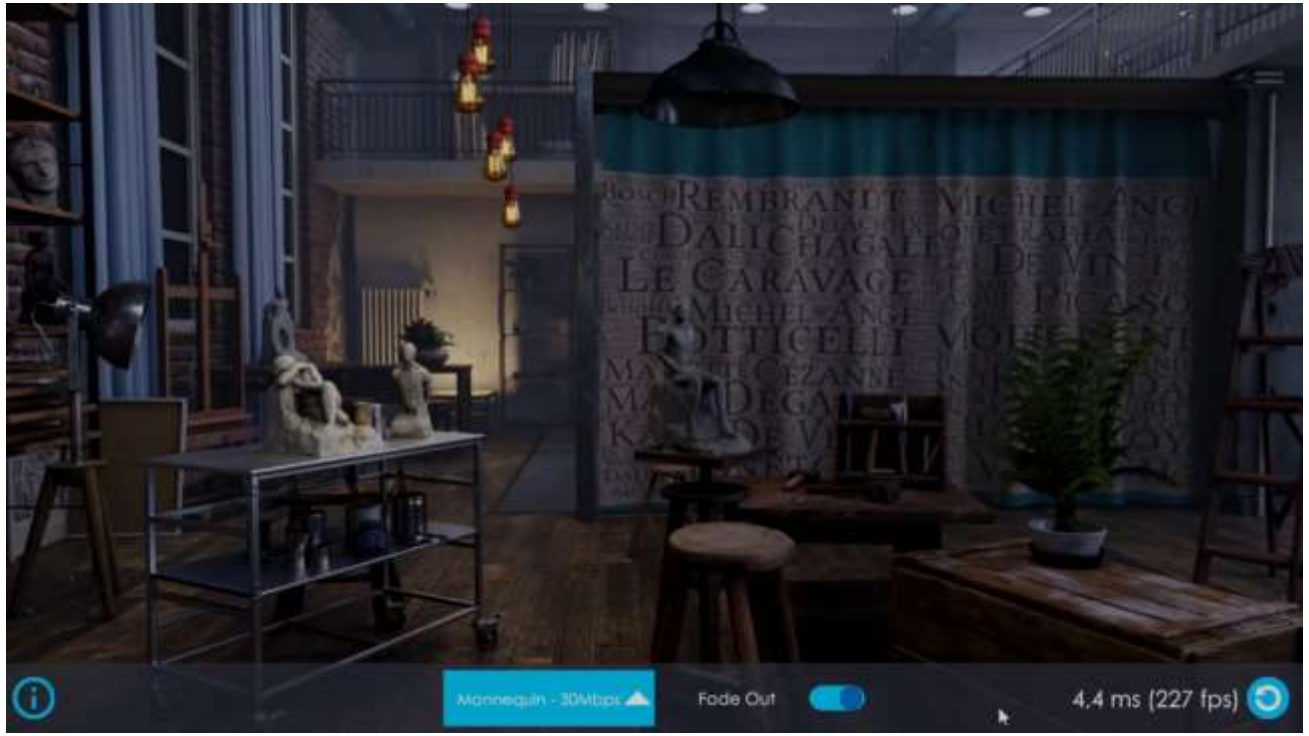
- Simplified rendering due to non-intersecting patches (inverse painters algorithm)
  - Full occupancy
  - Constant depth per patch
  - Texture video data
  - Transparency video data
  - Packed video data (opt.)

# Multi-Plane Images for Free Viewpoint

Julien Fleureau, Renaud Doré

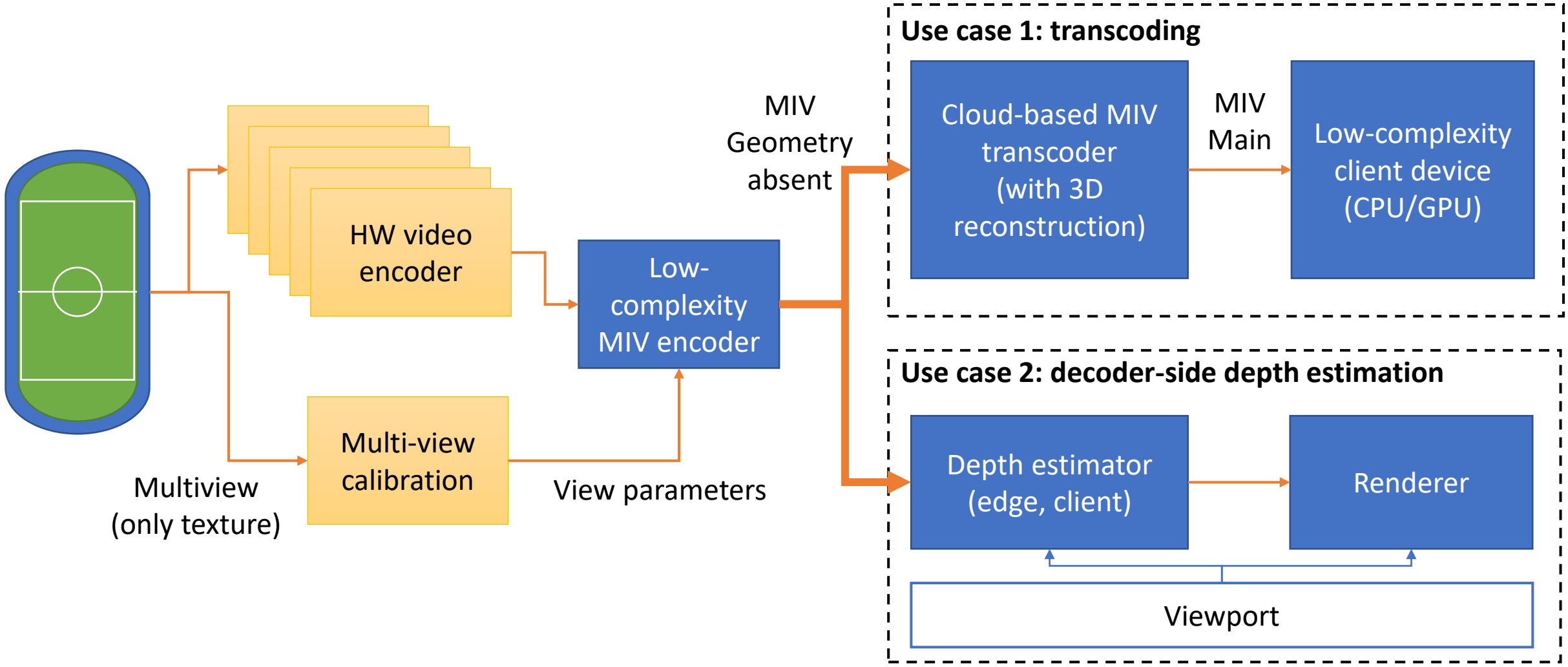
- MIV Extended w. Restricted Geometry
- Two HEVC Main10 video bitstreams
  - Texture
  - Transparency
- Multi-plane image strips packed in atlas

- Mobile, PC



# Profile: MIV Geometry absent

Postpone 3D reconstruction



# Preparation of MIV edition-2 has started

Please participate

**Preliminary schedule**

- 21Q4 Draft requirements
- 22Q1 Final requirements (WD)
- 22Q2 138 WD
- 22Q3 139 WD
- 22Q4 140 CD
- 23Q1 141 DIS
- 23Q3 143 FDIS

**Draft use cases and requirements**

- Evolution of MIV 1
- Larger viewing spaces
- Non-Lambertian content
- Scenes with static and dynamic elements
- Object-based coding
- Bullet time

# MIV tutorial

- MIV website: https://mpeg-miv.org
- Public software:
  - Test model: https://gitlab.com/mpeg-i-visual/tmiv
  - IV-PSNR: https://gitlab.com/mpeg-i-visual/ivpsnr
  - IVDE: https://gitlab.com/mpeg-i-visual/ivde
  - RVS: https://gitlab.com/mpeg-i-visual/rvs
- Demo videos:
  - Freeport player: https://youtu.be/UeT_Xm1jBGs
  - Multi-plane images: https://youtu.be/DPMMzj2I6Yw
  - Real-time RVS: https://lisaserver.ulb.ac.be/rvs/
- Test material (14+ sequences) available on request

# Q&A

End of part 1