INTERNATIONAL STANDARD

ISO/IEC 23090-6

First edition
2021-06

Information technology — Coded representation of immersive media —

Part 6:
Immersive media metrics

Reference number
ISO/IEC 23090-6:2021(E)

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23090 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

The immersive media metrics and measurement framework provide interoperability for consistent logging and monitoring of immersive media quality and experiences.

Annex A provides an illustration of immersive media metrics measurement.

Information technology — Coded representation of immersive media —

Part 6: Immersive media metrics

1 Scope

This document specifies immersive media metrics and the measurement framework. The immersive media metrics can be collected by service providers and used to enhance the immersive media quality and experiences. This document also includes a client reference model with observation and measurement points for collection of the metrics.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 23009-1:2019, *Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats*

ISO 23090-2, *Coded representation of immersive media — Part 2: Omnidirectional media format*

3 Terms and definitions

No terms and definitions are listed in this document.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

4 Abbreviated terms

2D	two-dimensional
2DQR	two-dimensional quality ranking
DASH	dynamic adaptive streaming over http
ER	effective resolution
ERT	effective resolution threshold
FOV	field of view
OMAF	omnidirectional media format
MCR	metrics computing and reporting

MPD media presentation description

OP observation point

PPI pixels per inch

QR quality ranking

QRT quality ranking threshold

SRQR spherical-region quality ranking

VR virtual reality

5 Arithmetic operators and mathematical functions

+ addition

− subtraction (as a two-argument operator) or negation (as a unary prefix operator)

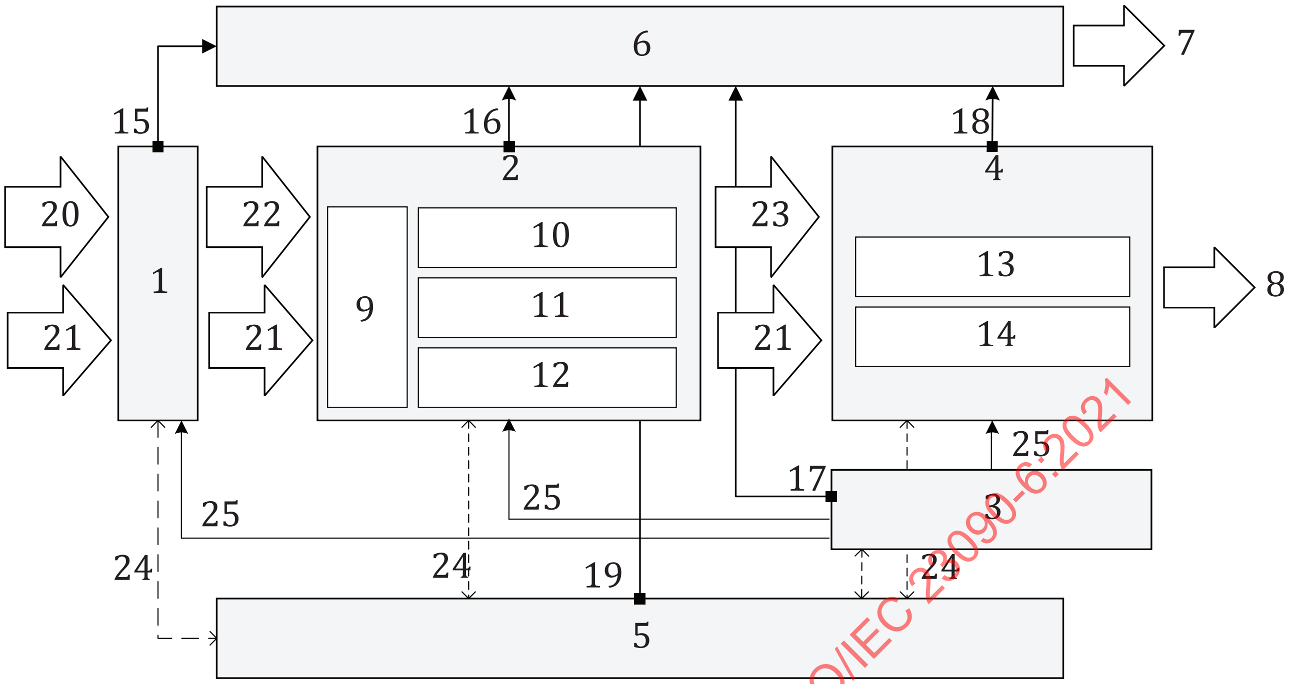
* multiplication, including matrix multiplication

$\sum_{i=x}^{y} f(i)$ summation of $f(\ i\ )$ with i taking all integer values from x up to and including y

6 Immersive media metrics client reference model

6.1 Overview

A generic immersive media client reference model is shown in Figure 1 with observation points for metrics measurement. The model consists of key functional modules including network access, media processing, sensor, media renderer, and immersive media application. A VR client may be an OMAF player for file/segment reception or file access, file/segment decapsulation, decoding of audio, video, or image bitstreams, audio and image rendering, and viewport selection. The metrics computing and reporting (MCR) module queries the measurable data from various functional modules and calculates the specified metrics. The MCR module may reside inside or outside of the VR client. The specified metrics may then be reported to an analytics server or other entities interested and authorized to access such metrics. The analytics server or other entities may use the metrics data to analyse the end user experience, assess client device capabilities, and evaluate the immersive system performance in order to enhance the overall immersive service experience across network, platform, device, applications and services.



Key

1	network access	14	audio rendering
2	media processing	15	OP1
3	sensor	16	OP2
4	media renderer	17	OP3
5	immersive media application	18	OP4
6	MCR	19	OP5
7	immersive media metrics	20	media segment
8	immersive presentation	21	metadata
9	file/segment decapsulation	22	media track
10	video decoding	23	media data
11	image decoding	24	control/config
12	audio decoding	25	sensor data
13	image rendering		

Figure 1 — Immersive media metrics client reference model

6.2 Definition of observation points

6.2.1 General

This clause defines the observation points as depicted in Figure 1.

6.2.2 Observation point 1

The network access module issues media file/segment requests and receives media files or segment streams from the network. The interface from the network access element towards MCR is referred to as observation point 1 (OP1). This observation point is equivalent to ISO/IEC 23009-1 observation point 1 as defined in ISO/IEC 23009-1:2019, D.3.2.

6.2.3 Observation point 2

The media processing module processes the file or the received media track, extracts the coded bitstreams, parses the media and metadata, and decodes the media. The interface from the media processing module towards MCR is referred to as observation point 2 (OP2).

The collectable data of OP2 includes parameters such as:

- MPD information, for example:
  - media type;
  - media codec;
  - adaptation set, representation, and preselection IDs;
- OMAF metadata, for example:
  - omnidirectional video projection;
  - omnidirectional video region-wise packing;
  - omnidirectional viewport;
- Other media metadata, for example:
  - frame packing;
  - colour space:
  - dynamic range.

6.2.4 Observation point 3

The sensor module acquires the user's viewing orientation, position and interaction. The interface from the sensor towards MCR is referred to as observation point 3 (OP3). The sensor data may be used by network access, media processing and media renderer module to retrieve, process and render VR media elements. For example, the current viewing orientation may be determined by the head tracking and possibly also eye tracking functionality. Besides being used by the renderer to render the appropriate part of decoded video and audio signals, the current viewing orientation may also be used by the network access for viewport dependent streaming and by the video and audio decoders for decoding optimization.

OP3 for example provides information of collectable sensor data for:

- the centre point of the current viewport;
- head motion tracking;
- eye tracking.

6.2.5 Observation point 4

The media renderer module synchronizes and playbacks the different VR media components to provide a fully immersive VR experience to the user. The decoded pictures are projected onto the screen of a head-mounted display or any other display device based on the current viewing orientation or viewport based on the metadata that includes information on region-wise packing, frame packing, projection, and sphere rotation as defined in ISO/IEC 23090-2. Likewise, decoded audio is rendered (e.g. through headphones) according to the current viewing orientation. The media renderer module may support colour conversion, projection, and media composition for each VR media component. The interface from the media renderer towards MCR is referred to as observation point 4 (OP4).

This observation point is equivalent to ISO/IEC 23009-1 observation point 3 as defined in ISO/IEC 23009-1:2019, D.3.4.

The collectable data from OP4 may, for example, include:

— the media type:

  — the media sample presentation timestamp:

  — wall clock time:

  — actual rendered viewport:

  — actual media sample rendering time:

  — actual rendering frame rate.

6.2.6 Observation point 5

The immersive media application manages the application configurations such as display resolution, frame rate, field of view (FOV), lens separation distance, etc. The interface from the immersive media application towards MCR is referred to as observation point 5 (OP5).

OP5 consists of client capability and configuration parameters, and the collectable data from OP5 includes, for example:

— display resolution;

— display density, in units of pixels per inch (PPI);

— horizontal and vertical FOV, in units of degrees;

— media format and codec support;

— OS support.

7 Metrics

7.1 General

This clause specifies specific immersive media metrics. The syntax for the DASH metrics as specified in ISO/IEC 23009-1:2019, D.4.1 is used for immersive media metrics with the following addition:

A new data type, `ViewportDataType`, is defined as shown in Table 1. `ViewportDataType` is an object with six integer keys that identify a viewport. The six keys are: `viewpoint_id`, `centre_azimuth`, `centre_elevation`, `centre_tilt`, `azimuth_range` and `elevation_range`.

Table 1 — ViewportDataType

Key		Type	Description
ViewportDataType		Object	
	viewpoint_id	Integer	Specifies the identifier of the viewpoint to which the viewport belongs.
	centre_azimuth	Integer	Specifies the azimuth of the centre of the viewport in units of 2^{-16} degrees. The value shall be in the range of −180 * 2^{16} to 180 * 2^{16} – 1, inclusive.
	centre_elevation	Integer	Specifies the elevation of the centre of the viewport in units of 2^{-16} degrees. The value shall be in the range of −90 * 2^{16} to 90 * 2^{16}, inclusive.
	centre_tilt	Integer	Specifies the tilt angle of the viewport in units of 2^{-16} degrees. The value shall be in the range of −180 * 2^{16} to 180 * 2^{16} – 1, inclusive.
	azimuth_range	Integer	Specifies the azimuth range of the viewport through the centre point of the viewport, in units of 2^{-16} degrees.
	elevation_range	Integer	Specifies the elevation range of the viewport through the centre point of the viewport, in units of 2^{-16} degrees.

7.2 Rendered FOV set metric

The `RenderedFOVSet` metric reports a set of FOVs rendered by VR client devices, as specified in Table 2.

Table 2 — RenderedFOVSet

Key			Type	Description
RenderedFovSet			Set	set of rendered FOVs
	Entry		Object	
		renderedFovH	Integer	The horizontal element of the rendered FOV, in units of 2^{-16} degrees, the value shall be in the range of 0 to 360 * 2^{16}, inclusive.
		renderedFovV	Integer	The vertical element of the rendered FOV, in units of 2^{-16} degrees, the value shall be in the range of 0 to 360 * 2^{16}, inclusive.

7.3 Display information set metric

The `DisplayInfoSet` metric reports a set of display resolution, pixel density and refresh rate values used by VR clients for rendering the VR video, as specified in Table 3.

Table 3 — DisplayInfoSet

Key			Type	Description
DisplayInfoSet			Set	set of display information
	Entry		Object	
		displayResolution	String	display resolution, in units of pixels
		displayPixelDensity	Integer	display pixel density, in units of PPI
		displayRefreshRate	Integer	display refresh rate, in units of Hz

This is a generic metric that also applies to other visual media applications than VR applications.

7.4 Rendered viewports metric

The `RenderedViewports` metric reports a list of viewports that have been rendered at particular intervals of media presentation times, as specified in Table 4.

Table 4 — RenderedViewports

Key			Type	Description
RenderedViewports			List	list of rendered viewports
	Entry		Object	
		startTime	Media-Time	Specifies the media presentation time of the first played out media sample when the viewport cluster indicated in the current entry is rendered starting from this media sample.
		duration	Integer	The time duration, in units of milliseconds, of the continuously presented media samples when the viewport cluster indicated in the current entry is rendered starting from the media sample indicated by `startTime`. "Continuously presented" means that the media clock continued to advance at the playout speed throughout the interval.
		viewport	ViewportDataType	Indicates the average region of the omnidirectional media corresponding to the viewport cluster that is rendered with the longest time starting from the media sample indicated by `startTime`.

7.5 Comparable quality viewport switching latency metric

The comparable quality viewport switching latency metric (`CQViewportSwitchingLatency`) reports the latency experienced by the user when switching from a first viewport to a second viewport until the presentation quality of the second viewport reaches a comparable presentation quality as the first viewport.

A new data type, `Viewport-Item`, is defined as shown in Table 5. `Viewport-Item` is an object specifying a list of regions with associated quality covering a viewport.

Table 5 — Viewport-Item

Key				Type	Description
Viewport-Item				Object	
	Position			ViewportDataType	identifies the viewport
	QualityLevels			List	list of different quality levels regions within the viewport
		Coverage		Float	percentage of the viewport area covered by this region
		QR		Integer	quality ranking (QR) value of this region
		Resolution		Object	resolution for this region
			Width	Integer	horizontal resolution for this region
			Height	Integer	vertical resolution for this region

The comparable quality viewport switching latency metric is specified in Table 6.

Table 6 — CQViewportSwitchingLatency

Key					Type	Description
CQViewportSwitchingLatency					List	List of comparable quality viewport switching latencies
	Entry				Object	
		firstViewport			ViewportItem	Specifies information about the first viewport (i.e., before the switching).
		secondViewport			ViewportItem	Specifies information about the second viewport (i.e., after the switching).
		worstViewport			ViewportItem	Specifies information about the worst viewport seen during the switch duration.
		time			Real-Time	Wall-clock time when switch starts.
		Mtime			Media-Time	Media presentation time when switch starts.
		latency			Integer	Specifies the interval in milliseconds between the time a user movement from first viewport to second viewport when the presentation quality of the second viewport reaches a comparable presentation quality as the first viewport.
		Accuracy			Integer	Specifies the estimated accuracy of the latency metric in milliseconds.
		Cause			List	Specifies a list of possible causes for the latency.
			Entry		Object	
				code	Enum	A possible cause for the latency. The value is equal to one of the following: — 0: Segment duration — 1: Buffer fullness — 2: Availability of comparable quality segment — 3: Timeout

8 Metric measurement process

8.1 General

This clause specifies the immersive media metric measurement process to provide interoperability for consistent logging and monitoring of immersive media quality and experiences.

8.2 Rendered viewport measurement

The `RenderedViewports` metric reports a list of viewports that have been rendered during the media presentation.

The client shall evaluate the current viewport every X milliseconds and potentially add the viewport to the rendered viewport list. To enable frequent viewport evaluations without necessarily increasing the report size too much, consecutive viewports which are close to each other may be grouped into clusters, where only the average cluster viewport data is reported.

The viewport clustering is controlled by an angular distance threshold *D*. If the centre (i.e. the azimuth and the elevation) of the current viewport is closer than the distance *D* to the current cluster centre (i.e.

the average cluster azimuth and elevation), the viewport is added to the cluster and the current cluster centre is updated.

If the distance to the cluster centre is instead equal to or larger than *D*, the current cluster centre, `azimuth_range` and `elevation_range`, the start time and duration is added to the viewport list and a new cluster is started and is set to be the current viewport.

Before reporting a viewport list, a filtering based on viewport duration shall be done. Each entry in the viewport list is first assigned an "aggregated duration" equal to the duration of that entry. Then, for each entry *E*, the other entries in the viewport list are checked. The duration for a checked entry is added to the aggregated duration for entry *E*, if the checked entry is both less than *t* milliseconds away from *E*, and closer than the angular distance *D* from *E*.

After all viewport entries have been evaluated and have received a final aggregated duration, all viewport entries with an aggregated duration of less than *t* are deleted from the viewport list (and thus not reported).

The observation points needed to calculate the metrics are:

— OP3 sensor: Gaze information;

— OP5 VR application: Field-of-view information of the device.

The configuration of values of viewport sample interval *X* (in milliseconds), the distance threshold *D* (in degrees), and the duration threshold *t* (in milliseconds) is not part of this document and may be set by external means.

8.3 Comparable quality viewport switching latency measurement

The `CQViewportSwitchingLatency` metric reports the latency and the quality-related factors when viewport movement causes quality degradations, such as when low-quality background content is briefly shown before the normal higher-quality is restored.

The viewport quality is represented by two factors; the quality ranking (QR) value, and the pixel resolution of one or more regions within the viewport. The resolution is defined by the `orig_width` and `orig_height` values in ISO/IEC 23090-2 in SRQR (spherical-region quality ranking) or 2DQR (2-dimensional quality ranking). The resolution corresponds to the monoscopic projected picture from which the packed region covering the viewport is extracted.

In order to determine whether two viewports have a comparable quality, if more than one quality ranking region is visible inside the viewport, the aggregated viewport quality factors are calculated as the area-weighted average for QR and the area-weighted (effective) pixel resolution, respectively, as specified in A.4.

When a viewport is moved so that the current viewport includes at least one new quality ranking region (i.e. a quality ranking region not included in the previous viewport), a switch event is started. The list quality factors related to the last evaluated viewport quality before the switch are assigned to the `firstViewport` log entry. The start time, `time`, of the switch is also set to the time of the last evaluated viewport before the switch.

The end time for the switch is defined as when both the average QR and the effective resolution for the viewport reach values comparable to the ones before the switch. A value is comparable if it is not more than QRT% (QR threshold) or ERT% (effective resolution threshold) worse than the corresponding values before the switch. If comparable values are not achieved within *N* milliseconds, a timeout occurs (for instance if an adaptation to a lower bitrate occurs, and the viewport never reaches comparable quality).

The list quality factors related to the viewport which fulfils both thresholds are assigned to the `secondViewport` log entry, and the latency (end time minus start time) is assigned to the latency log entry. In case of a timeout, this is indicated under the `cause` log entry.

During the duration of the switch the worst evaluated viewport is also stored and assigned to the `worstViewport` log entry.

If a new viewport switching event occurs (e.g. yet another new region becomes visible) before an ongoing switch event has ended, only the *N* milliseconds timeout is reset. The ongoing measurement process continues to evaluate the viewport quality until a comparable viewport quality value is achieved (or a timeout occurs). The observation points needed to calculate the metrics are:

— OP2 file decoder: SRQR/2DQR information;

— OP3 sensor: Gaze information;

— OP4 VR renderer: Start of switch event detection (alternatively, region coverage information from SRQR/2DQR can be used);

— OP5 VR application: Field-of-view information of the device.

The accuracy of the measured latency depends on how the client implements the viewport switching monitoring. As this might differ between clients, the client shall report the estimated `accuracy`.

The thresholds QRT, ERT, and the timeout *N*, can be specified during metrics configuration. Their configuration is not part of this document and may be set by external means.

Annex A
(informative)

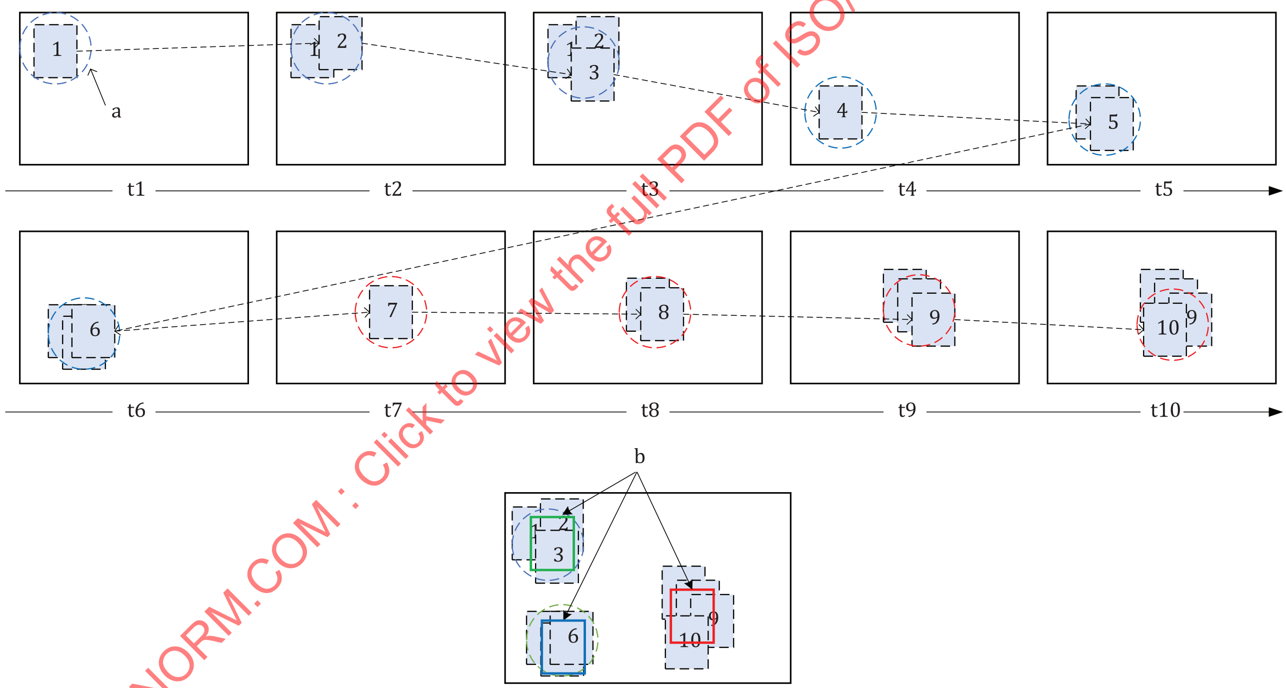
Illustration of implementation

A.1 General

This annex illustrates the calculation of immersive media metrics measurement.

A.2 Rendered viewport cluster

A viewport clustering is formed with a number of consecutive viewports close to each other. If the centre (i.e. the azimuth and the elevation) of the current viewport is closer than the distance *D* to the current cluster centre (i.e. the average cluster azimuth and elevation), the viewport is added to the cluster and the current cluster centre is updated.



Key

a radius = D

b average cluster viewports

Figure A.1 — Example of a rendered viewport cluster

Figure A.1 illustrates an example of the cluster and the associated rendered viewports. Rendered viewports #1, #2 and #3 form a first cluster as these viewports are all within the distance *D* of the first cluster. Viewport #4 starts a second cluster as the distance between the first cluster and viewport #4 is larger than *D*, viewports #4, #5 and #6 form a second cluster as they are all within distance *D* of the second cluster. Similarly, viewports #7, #8, #9 and #10 form a third viewport cluster. The centre position of each cluster is derived by averaging the centre position of all associated viewports of the cluster, and the size of the viewport cluster is derived by averaging the size of all associated viewports of the cluster.

For each cluster j, the final averaged viewport parameters can be derived as follows, assuming there are N viewports in the j-th cluster. Note that the centre azimuth and tilt averaging also needs to handle the special case around −180/180 degrees, as some values might be positive (e.g. 176 degrees), while others might be negative (e.g. −178 degrees). This special case is not shown in the following:

$$\alpha[j]=\sum_{i=1}^{N}a[i]/N$$

$$\beta[j]=\sum_{i=1}^{N}b[i]/N$$

$$\Upsilon[j]=\sum_{i=1}^{N}c[i]/N$$

$$\mu[j]=\sum_{i=1}^{N}d[i]/N$$

$$v[j]=\sum_{i=1}^{N}e[i]/N$$

where

α is the average_centre_azimuth

β is the average_centre_elevation

Υ is the average_centre_tilt

μ is the average_azimuth_range

v is the average_elevation_range

a is the centre_azimuth

b is the centre_elevation

c is the centre_tilt

d is the azimuth_range

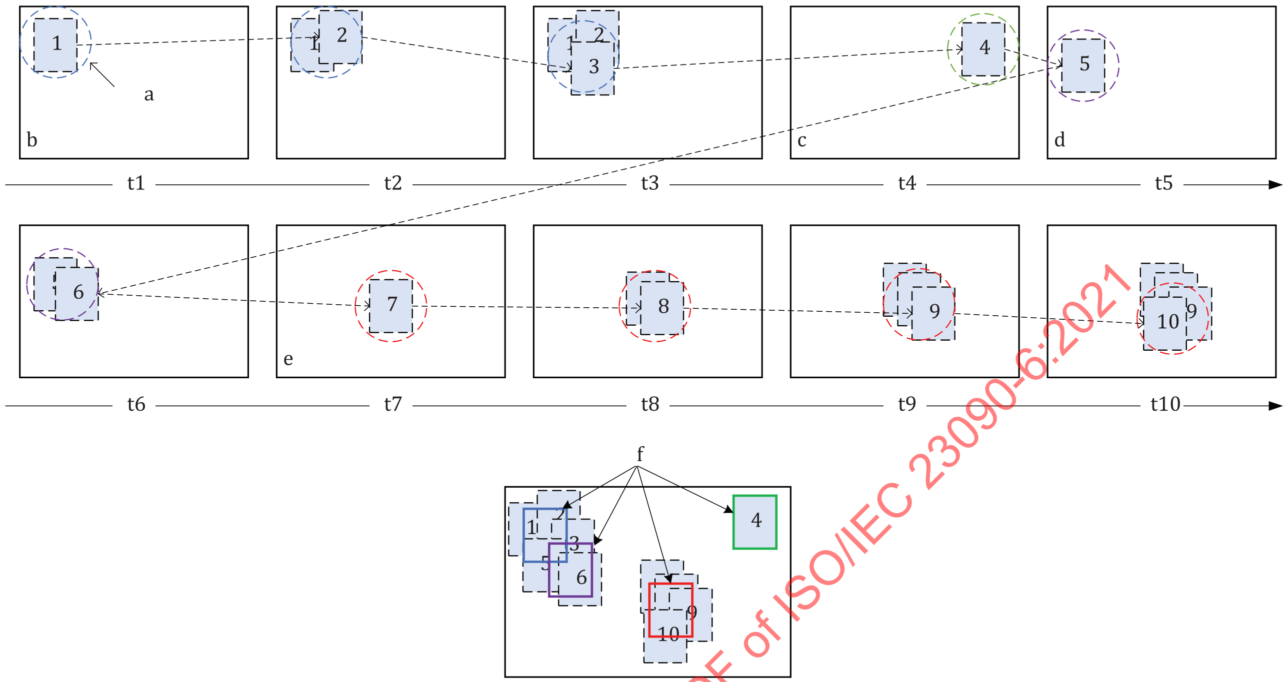
e is the elevation_range

Note also that the azimuth and elevation range (i.e. the visible coverage of the viewport) might often be the same for every viewport, unless the user explicitly changes the field-of-view for the device. For consistency, and to catch any during-session field-of-view changes, these two parameters should still be averaged.

A.3 Rendered viewport filtering

Figure A.2 illustrates an example of the duration filtering. The user starts by looking at the upper left part of the media (viewports #1 to #3), then make a very brief glance to the right (viewport #4), and then moves back to the upper-left again (viewports #5 and #6). Then the user moves his gaze to the lower-right part (viewports #7 to #10).

Assume that the duration t is set to 4 times the value of the viewport sample rate X (i.e. a cluster needs to have a duration corresponding to at least four viewports to be reported). Here four clusters are formed, but before filtering only cluster #4 would be reported. After filtering, clusters #1 and #3 are close enough both in time and distance to add to each other's aggregated duration, so each of them will be assigned an aggregated duration of 5, and thus be reported. Cluster #2, the quick gaze up to the right, has too short duration and will not be reported.



Key

a radius = *D*

b cluster #1 starts

c cluster #2 starts

d cluster #3 starts

e cluster #4 starts

f average cluster viewports

Figure A.2 — Example of rendered viewport cluster filtering

A.4 Viewport quality

This clause illustrates how the weighted average QR value and the effective resolution can be calculated.

The quality level of each region is determined with its respective QR value. A viewport can be covered with multiple regions. A quality level value for the viewport can be derived as weighted average of the QR values of the regions covering the viewport. The weight of each region is defined as the percentage of the viewport area covered by the corresponding region. The viewport quality level can be calculated by:

$$V = \sum_{i=1}^{N} (Q[i] * C[i] / 100)$$

where

V is the quality (viewport)

N is the number of regions covering the viewport

$Q[i]$ is the QR value of i-th quality ranking region

$C[i]$ is the viewport coverage value (in percent) of i-th quality ranking region