FINAL DRAFT
International Standard

ISO/IEC
FDIS
21794-6

Information technology — Plenoptic image coding system (JPEG Pleno) —

Part 6:
Learning-based point cloud coding

ISO/IEC JTC 1/SC 29

Secretariat: **JISC**

Voting begins on:
2025-04-17

Voting terminates on:
2025-06-12

Reference number
ISO/IEC FDIS 21794-6:2025(en)

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 or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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. In the IEC, see www.iec.ch/understanding-standards.

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 21794 series can be found on the ISO and IEC websites.

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 and www.iec.ch/national-committees.

Introduction

This document is part of a series of standards for a system known as JPEG Pleno. This set of standards facilitates the capture, representation, exchange and visualization of plenoptic imaging modalities. A plenoptic image modality can be a light field, point cloud or hologram, which are sampled representations of the plenoptic function in the form of, respectively, a vector function that represents the radiance of a discretized set of light rays, a collection of points with position and attribute information, or a complex wavefront. The plenoptic function describes the radiance in time and in space obtained by positioning a pinhole camera at every viewpoint in 3D spatial coordinates, every viewing angle and every wavelength, resulting in a 7D function.

JPEG Pleno specifies tools for coding these modalities while providing advanced functionality at system level, such as support for data and metadata manipulation, editing, random access and interaction, protection of privacy and ownership rights.

The scope of this document is the specification of a learning-based coding standard for point clouds and associated attributes, offering a single-stream, compact compressed domain representation, supporting advanced flexible data access functionalities. In this context, learning-based refers to the use of machine learning technologies to learn an optimal compressed domain representation from supplied training data.

Information technology — Plenoptic image coding system (JPEG Pleno) —

Part 6:
Learning-based point cloud coding

1 Scope

This document defines the JPEG Pleno framework for learning-based point cloud coding.

This document is applicable to interactive human visualization, with competitive compression efficiency compared to state of the art point cloud coding solutions in common use, and effective performance for 3D processing and machine-related computer vision tasks, and has the goal of supporting a royalty-free baseline.

This document specifies a coded codestream format for storage of point clouds. It provides information on the encoding tools. It also defines extensions to the JPEG Pleno File Format and associated metadata descriptors that are specific to point cloud modalities.

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/IEC 6048-1:—[1)], *Information technology — JPEG AI learning-based image coding system — Part 1: Core coding system*

ISO/IEC 15444-1[2)], *Information technology — JPEG 2000 image coding system — Part 1: Core coding system*

ISO/IEC 15444-2[3)], *Information technology — JPEG 2000 image coding system — Part 2: Extensions*

ISO/IEC 21794-1, *Information technology — Plenoptic image coding system (JPEG Pleno) — Part 1: Framework*

ISO/IEC 21794-2, *Information technology — Plenoptic image coding system (JPEG Pleno) — Part 2: Light field coding*

ISO/IEC 21794-3, *Information technology — Plenoptic image coding system (JPEG Pleno) — Part 3: Conformance testing*

ISO/IEC 21794-4, *Information technology — Plenoptic image coding system (JPEG Pleno) — Part 4: Reference software*

ISO/IEC 21794-5, *Information technology — Plenoptic image coding system (JPEG Pleno) — Part 5: Holography*

ISO/IEC 23090-5, *Information technology — Coded representation of immersive media — Part 5: Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC)*

ISO/IEC 23090-9, *Information technology — Coded representation of immersive media — Part 9: Geometry-based point cloud compression*

1) Under preparation. Stage at the time of publication: ISO/IEC PRF 6048-1:2025.

2) Similar to REC. ITU-T T.800 | ISO/IEC 15444-1

3) Similar to REC. ITU-T T.801 | ISO/IEC 15444-2

ISO/IEC 60559, *Information technology — Microprocessor Systems — Floating-Point arithmetic*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 21794-1, ISO/IEC 21794-2, ISO/IEC 21794-3, ISO/IEC 21794-4, ISO/IEC 21794-5, ISO/IEC 23090-5, ISO/IEC 23090-9, ISO/IEC 6048-1, ISO/IEC 15444-1, ISO/IEC 15444-2, ISO/IEC 60559 and the following apply.

ISO and IEC maintain terminology 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/

3.1
point
fundamental element of a point cloud comprising a position specified as 3D spatial coordinates and colour attributes

3.2
point cloud
unordered list of points

3.3
neural network layer
tensor operation that contains trainable parameters which receives and outputs a tensor

3.4
neural network module
set of neural network layers

3.5
neural network model
specified sequence of neural network modules (also called architecture) and corresponding trained parameters

3.6
trainable parameters
parameters of a neural network layer whose values require a training process based on input ground truth data to be set

3.7
trained parameters
parameters of a neural network model whose values have been set by a training process based on input ground truth data

3.8
dense tensor
representation of a 3D block as a regular array with four dimensions: horizontal, vertical, depth and channel dimension

3.9
sparse tensor
representation of a 3D block, where only non-zero elements are represented as a set of indices (or coordinates) C and associated values (or features) F

Note 1 to entry: The set of coordinates C is represented as a matrix $C \in \mathrm{Z}^{P \times 3}$ and the associated features F are represented as a matrix $F \in \mathrm{R}^{P \times N}$, where P is the number of non-zero elements and N is the number of channels. The remaining elements of a sparse tensor are zeros.

Note 2 to entry: Opposed to a dense tensor.

3.10
latent tensor
intermediate representation of point cloud data during encoding or decoding processes, as a sparse tensor

3.11
standard deviations tensor
tensor of unsigned 16 bits integers, used for entropy coding, denoted as σ

3.12
concatenation of sparse tensors
process where the features of two sparse tensors are concatenated

3.13
element-wise addition of sparse tensors
process where the features of two sparse tensors are added element-wise

3.14
sparse convolution layer
three-dimensional sparse convolution denoted as $SpConv\left(K_{ver} \times K_{hor} \times K_{dep}, N_{in}, N_{out}, s\downarrow\right)$

3.15
transposed sparse convolution layer
three-dimensional transposed sparse convolution denoted as $TSpConv\left(K_{ver} \times K_{hor} \times K_{dep}, N_{in}, N_{out}, s\uparrow\right)$

3.16
generative transposed sparse convolution layer
three-dimensional generative transposed sparse convolution denoted as $GTSpConv\left(K_{ver} \times K_{hor} \times K_{dep}, N_{in}, N_{out}, s\uparrow\right)$

3.17
quantized sparse convolution layer
three-dimensional quantized sparse convolution is denoted as $qSpConv\left(K_{ver} \times K_{hor} \times K_{dep}, N_{in}, N_{out}, s\downarrow, d, p\right)$

3.18
quantized generative transposed sparse convolution layer
three-dimensional quantized generative transposed sparse convolution is denoted as $qGTSpConv\left(K_{ver} \times K_{hor} \times K_{dep}, N_{in}, N_{out}, s\uparrow, d, p\right)$

3.19
matrix multiplication
matrix multiplication (denoted as $\times$) receives two two-dimensional arrays $input_1[h_{out}, L]$ and $input_2[L, w_{out}]$.

Note 1 to entry: This module produces a two-dimensional array $output$ of size $[h_{out}, w_{out}]$:

For $i = 0, \ldots, h_{out} - 1$ and $j = 0, \ldots, w_{out} - 1$:

$$output[i, j] = \sum_{l=0}^{L-1} input_1[i, l] \cdot input_2[l, j].$$

3.20
rectified linear unit
rectified linear unit is denoted as $ReLU$

Note 1 to entry: This element-wise function is defined as:

$$ReLU(x) = \begin{cases} x, & if\ x \geq 0, \\ 0, & otherwise. \end{cases}$$

3.21
sigmoid
sigmoid operation is denoted as *Sigmoid*

Note 1 to entry: This element-wise function is defined as:

$$Sigmoid(x) = \frac{1}{1+e^{-x}}.$$

4 Symbols and abbreviated terms

4.1 Symbols

$bias$	bias parameters of a convolutional layer
$BlockPosition$	global coordinates of a point cloud block origin point in the input source point cloud
BS	size of the 3D block in voxels
C	coordinates of sparse tensor
$\boldsymbol{c}$	a specific coordinate within C_{in} or C_{out}
C_{in}	coordinates of sparse tensor input to a process
C_{out}	coordinates of sparse tensor output from a process
Col	colour of original point cloud
$\widehat{Col}$	decoded point cloud colour
$\widehat{Col}^{i}_{knn}$	colour values corresponding to points in $\widehat{Geo}^{i}_{knn}$
$\widehat{Col}_{SR_i}$	colour of point *i* in decoded point cloud with super-resolution
d_j	clipping value for quantized sparse convolution layer *j* in hyper scale decoder network
D	depth of the point cloud volume in voxels
F	features of a sparse tensor
F_{in}	features of a sparse tensor input to a process
F_{out}	features of a sparse tensor output from a process
Geo	geometry of original point cloud
$\widehat{Geo}$	decoded point cloud geometry
$\widehat{Geo}_{SR}$	decoded point cloud geometry with super-resolution
$\widehat{Geo}_{SR_i}$	point *i* in decoded point cloud geometry with super-resolution $\widehat{Geo}_{SR}$
$\widehat{Geo}^{i}_{knn}$	20 nearest neighbouring points of $\widehat{Geo}_{SR_i}$ from decoded point cloud geometry $\widehat{Geo}$
h	height of 2D image produced by 3D to 2D projection module

H	height of point cloud volume in voxels
Im_1	near layer image produced by 3D to 2D projection module
Im_2	far layer image produced by 3D to 2D projection module
k	number of points to be decoded
K	convolutional kernel
K_{ver}	vertical dimension of a convolutional kernel
K_{hor}	horizontal dimension of a convolutional kernel
K_{dep}	depth dimension of a convolutional kernel
m	sampling factor index for super-resolution network
$modelIdx$	index of the point cloud coding model used for encoding and decoding
N	number of points in point cloud
N_{in}	number of input channels of a sparse convolution layer
N_{out}	number of output channels of a sparse convolution layer
N_{σ}	number of scales of the Gaussian distributions for the entropy coding of $\hat{r}_F$
P	number of points in original point cloud block
p_j	de-scaling shift parameter values for quantized sparse convolution layer j in hyper scale decoder network
$\hat{P}$	number of points in decoded point cloud block
QS	user-defined quantisation step
r	residual of subtraction of latent representation y_{QS} and means μ
r_C	coordinates of residual r
r_F	features of residual r
$\hat{r}$	quantized residual r
$\hat{r}_F$	features of $\hat{r}$
s	stride of sparse convolution layers
SF	user-defined down-sampling factor for point cloud block down-sampling
w	width of 2D image produced by 3D to 2D projection module
W	width of point cloud volume in voxels
$weights$	weights of a convolutional layer
x	point cloud block to be transformed by the analysis transform represented as a sparse tensor
x_C	coordinates of x

x_F	features of x
X_{global}	global point cloud coordinates
$X_{internal}$	internal point cloud block coordinates given that each block has its own origin
$\hat{x}$	decoded point cloud block x
$\hat{x}_c$	decoded coordinates of x
$\hat{x}^{SR}$	decoded point cloud block with super-resolution
$\hat{x}_C^{SR}$	coordinates of $\hat{x}^{SR}$
$\hat{x}_F^{SR}$	features of $\hat{x}^{SR}$
y	latent representation of point cloud block x created by the analysis transform
y_C	coordinates of y
y_F	features of y
$\hat{y}$	decoded latent representation y
y_{QS}	latent representation y scaled by QS
$\hat{y}_{QS}$	decoded scaled latent representation y_{QS}
z	hyper latent tensor produced by hyper encoder
z_C	coordinates of z
z_F	features of z
$\hat{z}$	quantized hyper latent tensor z
$\hat{z}_C$	coordinates of $\hat{z}$
$\hat{z}_F$	features of $\hat{z}$
β	ratio of number of points to be decoded over number of points in original point cloud block
μ	means of latent representation prediction created by the hyper mean decoder
σ	standard deviations (or scales) of the Gaussian distributions for the entropy coding of $\hat{r}_F$, created by the hyper scale decoder
σ_{min}	minimum scale value of the Gaussian distributions for the entropy coding of $\hat{r}_F$
σ_{max}	maximum scale value of the Gaussian distributions for the entropy coding of $\hat{r}_F$
Res	residual produced by subtraction of far layer image from near layer image
TRes	trimmed residual image *Res*
Left	left position of *TRes* within *Res*
Top	top position of *TRes* within *Res*

Right	right position of *TRes* within *Res*
Bottom	bottom position of *TRes* within *Res*

4.2 Abbreviated terms

2D	two-dimensional
3D	three-dimensional
CDF	cumulative distribution function
CSV	comma separated values
HTTP	hypertext transfer protocol
IPR	intellectual property rights
IRB	inception-residual blocks
JPEG	Joint Photographic Experts Group
JPL	JPEG Pleno file format
LIRB	lightweight inception-residual block
LSB	least significant bit
MSB	most significant bit
PC	point cloud
rANS	range variant of asymmetric numeral systems
SR	super resolution
XML	extensible markup language

5 Conventions

5.1 Naming conventions for numerical values

Integer numbers are expressed as bit patterns, hexadecimal values or decimal numbers. Bit patterns and hexadecimal values have both a numerical value and an associated length in bits.

Hexadecimal notation, indicated by prefixing the hexadecimal number by "0x", may be used instead of binary notation to denote a bit pattern having a length that is an integer multiple of 4. For example, 0x41 represents an eight-bit pattern having only its second most significant bit and its least significant bit equal to 1. Numerical values that are specified under a "**Code**" heading in tables that are referred to as "code tables" are bit pattern values (specified as a string of digits equal to 0 or 1 in which the left-most bit is considered the most-significant bit). Other numerical values not prefixed by "0x" are decimal values. When used in expressions, a hexadecimal value is interpreted as having a value equal to the value of the corresponding bit pattern evaluated as a binary representation of an unsigned integer (i.e. as the value of the number formed by prefixing the bit pattern with a sign bit equal to 0 and interpreting the result as a two's complement representation of an integer value). For example, the hexadecimal value 0xF is equivalent to the 4-bit pattern '1111' and is interpreted in expressions as being equal to the decimal number 15.

5.2 Operators

NOTE Many of the operators used in this document are similar to those used in the C programming language.

5.2.1 Arithmetic operators

+ addition

– subtraction (as a binary operator) or negation (as a unary prefix operator)

× multiplication

/ division without truncation or rounding

<< left shift; `x<<s` is defined as $x \times 2^s$

>> right shift; `x>>s` is defined as $\lfloor x/2^s \rfloor$

++ increment with 1

-- decrement with 1

umod unsigned modulo operator; x umod a is the unique value y between 0 and a–1 for which y+Na = x with a suitable integer N

& bitwise AND operator; compares each bit of the first operand to the corresponding bit of the second operand
If both bits are 1, the corresponding result bit is set to 1. Otherwise, the corresponding result bit is set to 0.

^ bitwise XOR operator; compares each bit of the first operand to the corresponding bit of the second operand
If both bits are equal, the corresponding result bit is set to 0. Otherwise, the corresponding result bit is set to 1.

5.2.2 Logical operators

|| logical OR

&& logical AND

! logical NOT

a ? b : c if condition a is true, then the result is equal to b; otherwise the result is equal to c

5.2.3 Relational operators

> greater than

>= greater than or equal to

< less than

<= less than or equal to

== equal to

!= not equal to