(12) UK Patent (19) GB (11) 2539488 (13) B

(45) Date of B Publication **27.03.2019**

(54) Title of the Invention: # Encoder, decoder and method employing palette utilization and compression

(51) INT CL: ***H04N 19/90*** (2014.01) ***H03M 7/30*** (2006.01) ***H04N 19/13*** (2014.01) ***H04N 19/147*** (2014.01) ***H04N 19/17*** (2014.01) ***H04N 19/70*** (2014.01)

(21) Application No: **1510768.3**

(22) Date of Filing: **18.06.2015**

(43) Date of A Publication **21.12.2016**

(56) Documents Cited:

GB 2371730 A
WO 2007/029945 A1
US 20150110181 A1
US 20150016501 A1
US 20140147040 A1
US 20070053426 A1
(JCT-VC R1014) Screen content coding test model 2, October 2014
(JCT-VC R1005) HEVC screen content coding draft text 1, August 2014.

(58) Field of Search:
As for published application 2539488 A viz:
INT CL **H04N**
Other: **EPODOC; INSPEC; INTERNET; WPI**
updated as appropriate

Additional Fields
Other: **None**

(72) Inventor(s):
Ossi Kalevo

(73) Proprietor(s):
Gurulogic Microsystems Oy
Linnankatu 34, Turku FI-20100, Finland

(74) Agent and/or Address for Service:
Basck Ltd
16 Saxon Road, CAMBRIDGE, Cambridgeshire, CB5 8HS, United Kingdom

GB 2539488 B

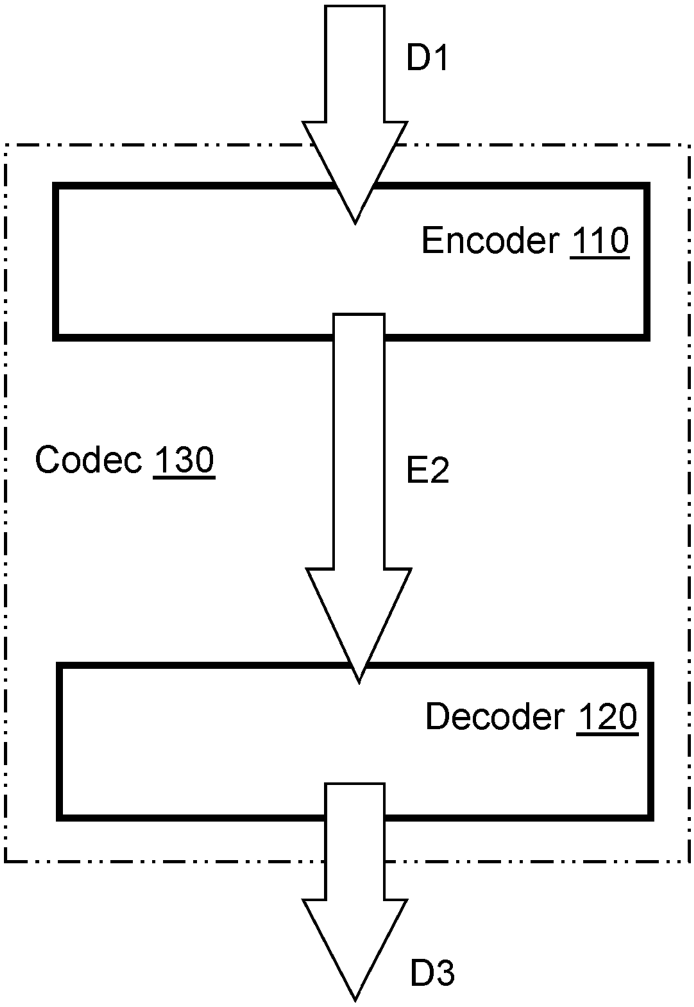


FIG. 1

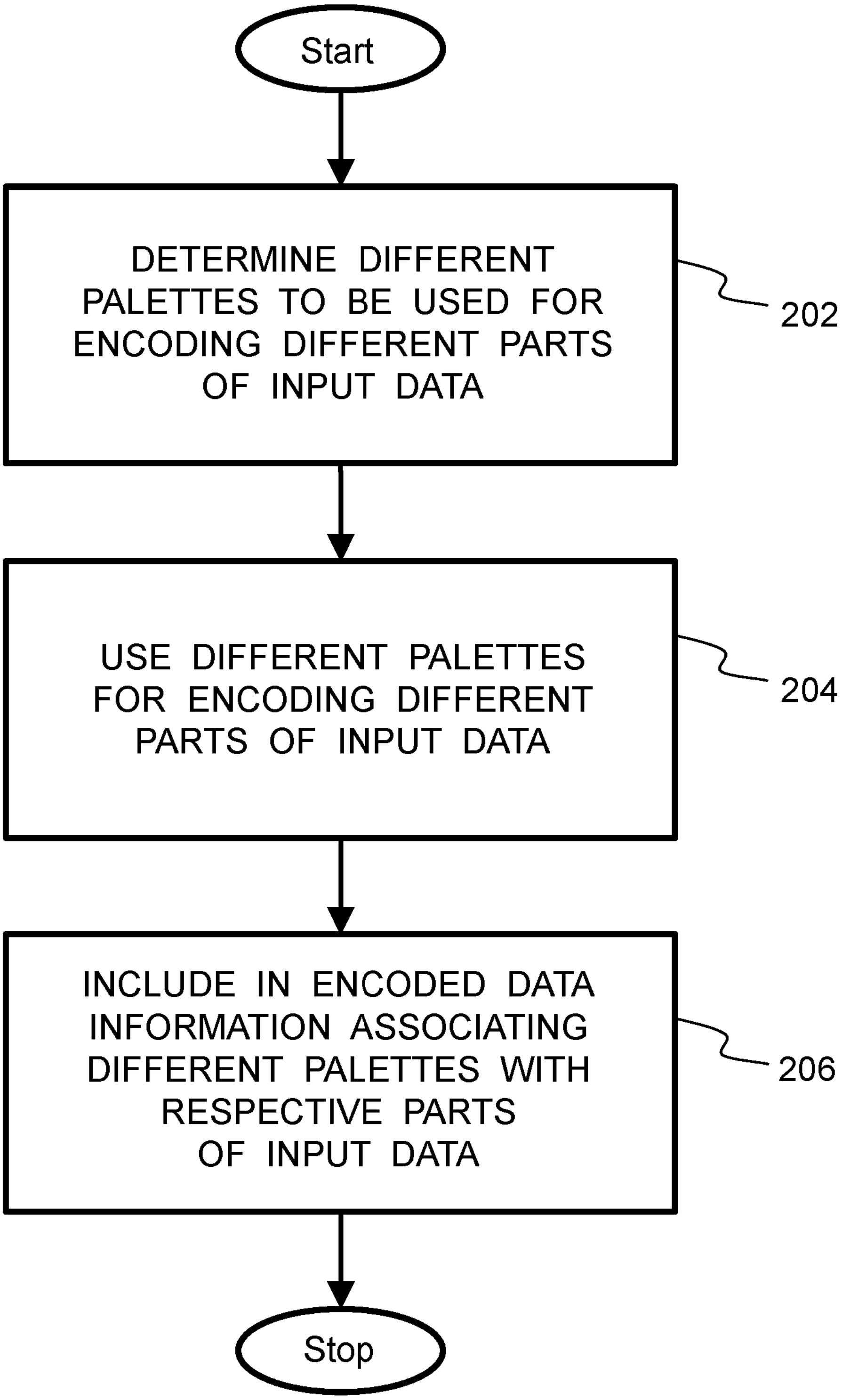

Start
DETERMINE DIFFERENT PALETTES TO BE USED FOR ENCODING DIFFERENT PARTS OF INPUT DATA
202
USE DIFFERENT PALETTES FOR ENCODING DIFFERENT PARTS OF INPUT DATA
204
INCLUDE IN ENCODED DATA INFORMATION ASSOCIATING DIFFERENT PALETTES WITH RESPECTIVE PARTS OF INPUT DATA
206
Stop

FIG. 2

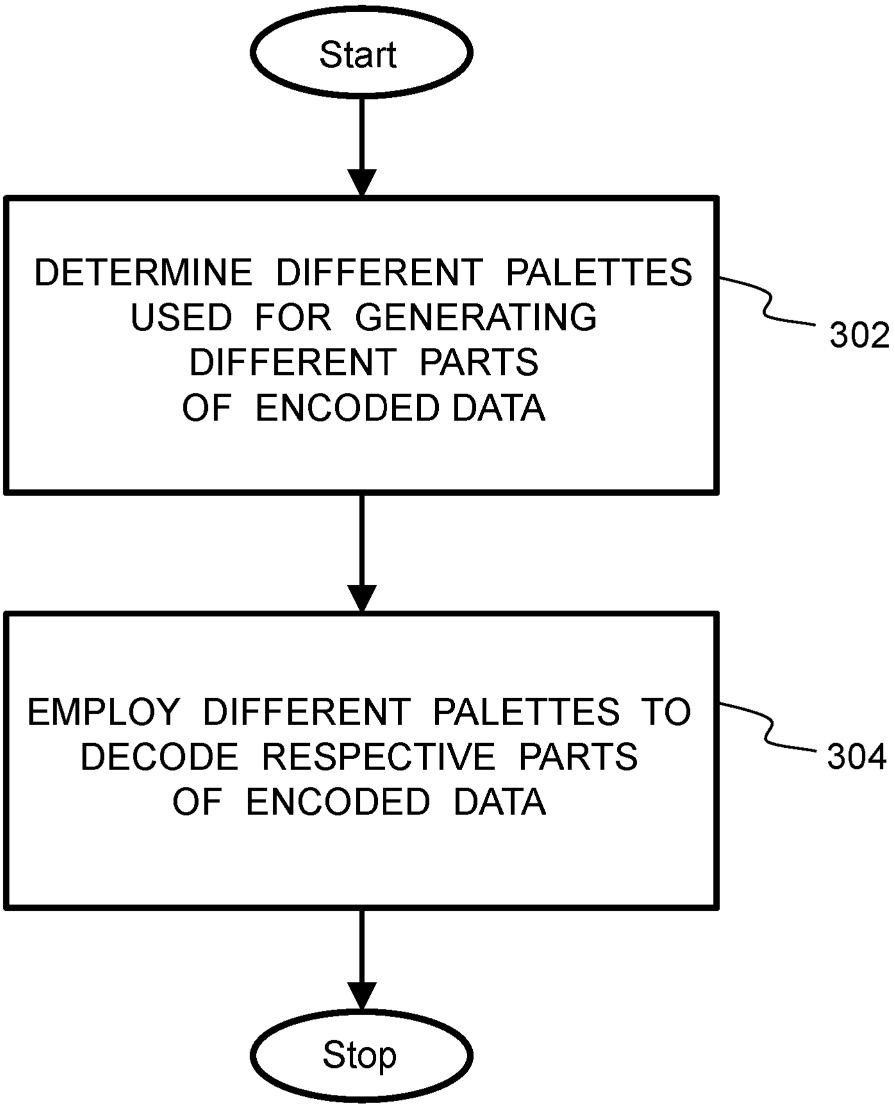

Start
DETERMINE DIFFERENT PALETTES USED FOR GENERATING DIFFERENT PARTS OF ENCODED DATA
302
EMPLOY DIFFERENT PALETTES TO DECODE RESPECTIVE PARTS OF ENCODED DATA
304
Stop

FIG. 3

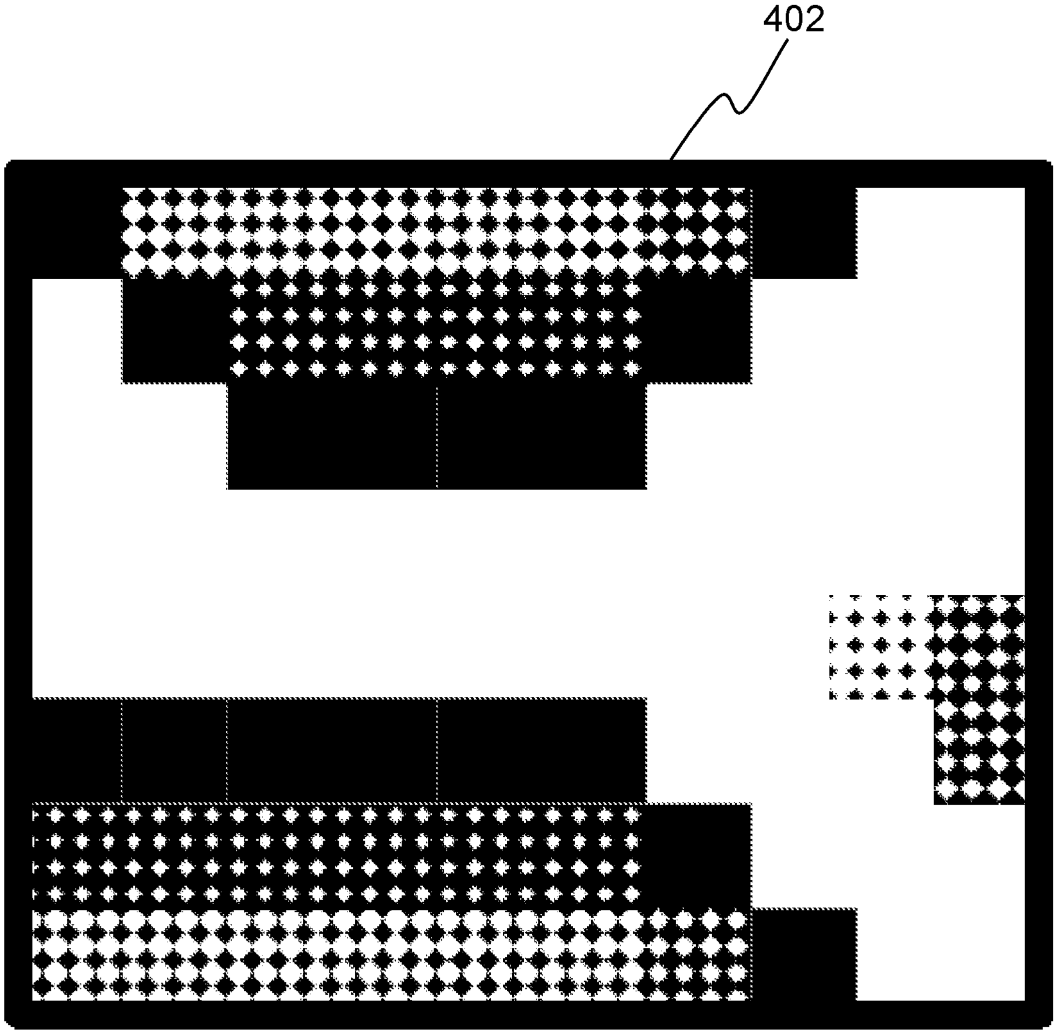


FIG. 4A

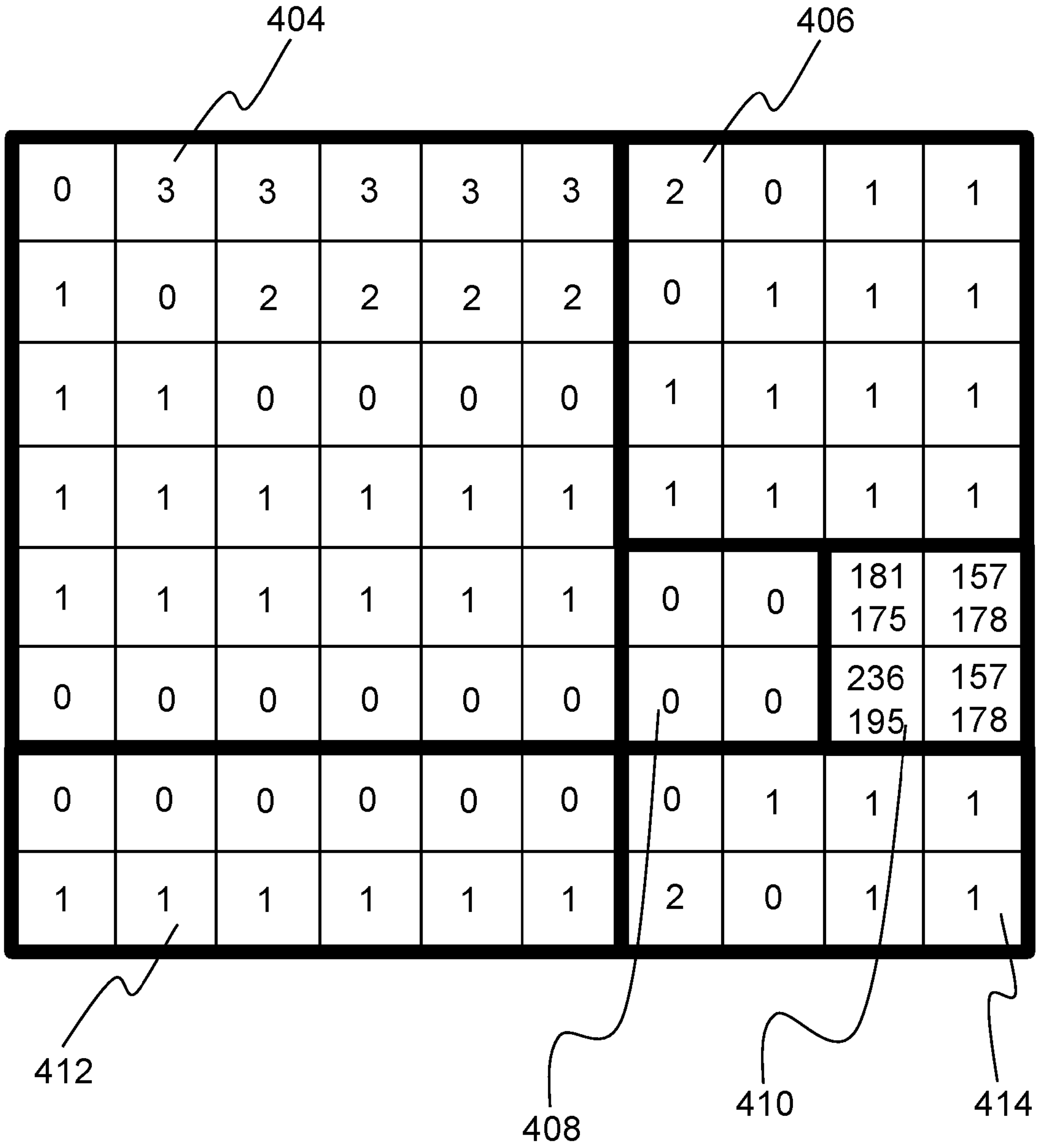

404
406
0 3 3 3 3 3 2 0 1 1
1 0 2 2 2 2 0 1 1 1
1 1 0 0 0 0 1 1 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 0 0 181 175 157 178
0 0 0 0 0 0 0 0 236 195 157 178
0 0 0 0 0 0 0 1 1 1
1 1 1 1 1 1 2 0 1 1
412
408
410
414

FIG. 4B

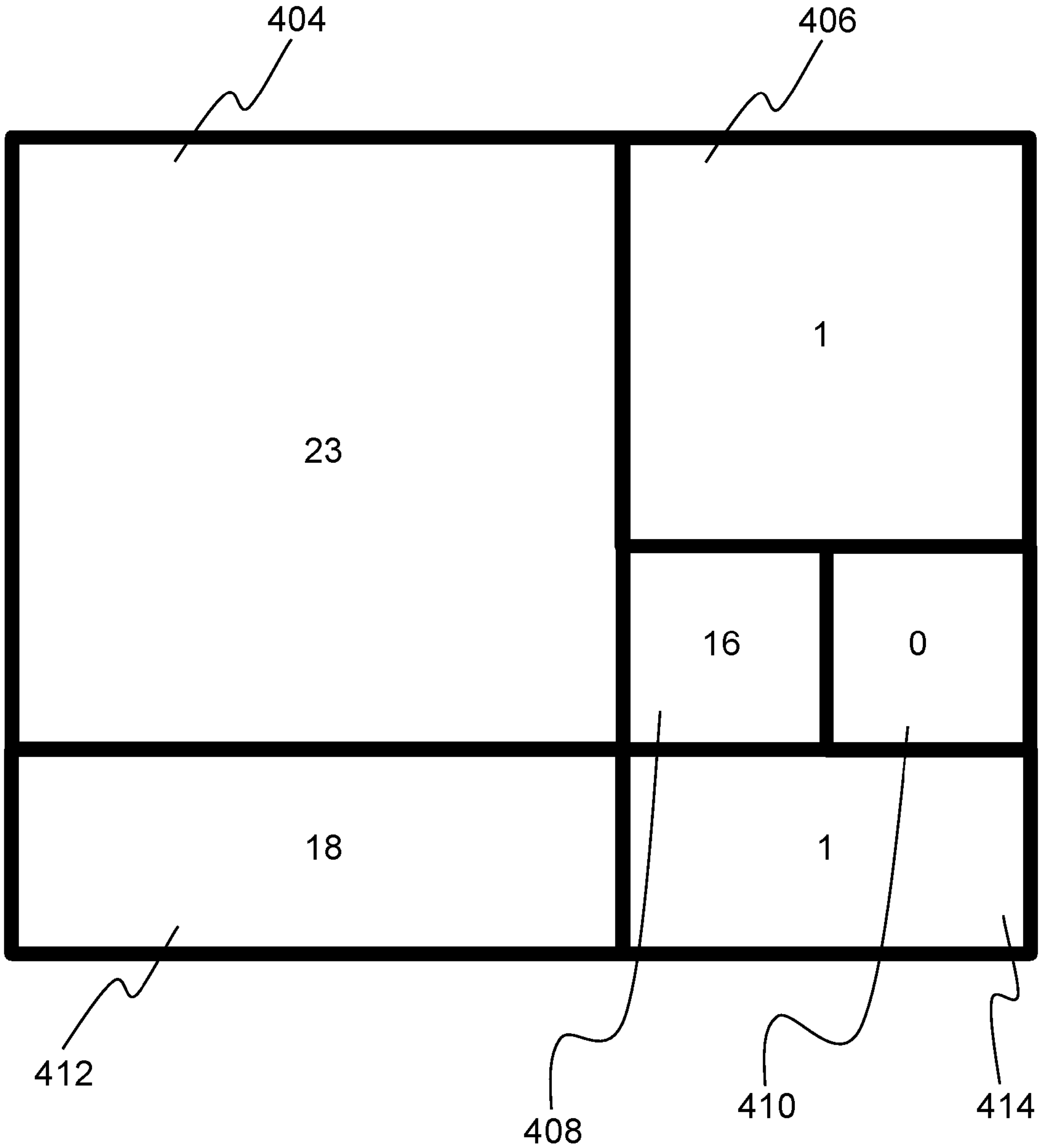


FIG. 4C

A B	126 70	167 152	167 152	167 152	167 152	167 152	152 138	126 70	236 195	236 195
A B	236 195	126 70	145 168	145 168	145 168	145 168	126 70	236 195	236 195	236 195
A B	236 195	236 195	126 70	126 70	126 70	126 70	236 195	236 195	236 195	236 195
A B	236 195	236 195	236 195	236 195	236 195	236 195	236 195	236 195	236 195	236 195
A B	236 195	236 195	236 195	236 195	236 195	236 195	236 195	236 195	181 175	157 178
A B	126 70	126 70	126 70	126 70	126 70	126 70	236 195	236 195	236 195	157 178
A B	145 168	145 168	145 168	145 168	145 168	145 168	126 70	236 195	236 195	236 195
A B	167 152	167 152	167 152	167 152	167 152	167 152	152 138	126 70	236 195	236 195

FIG. 4D

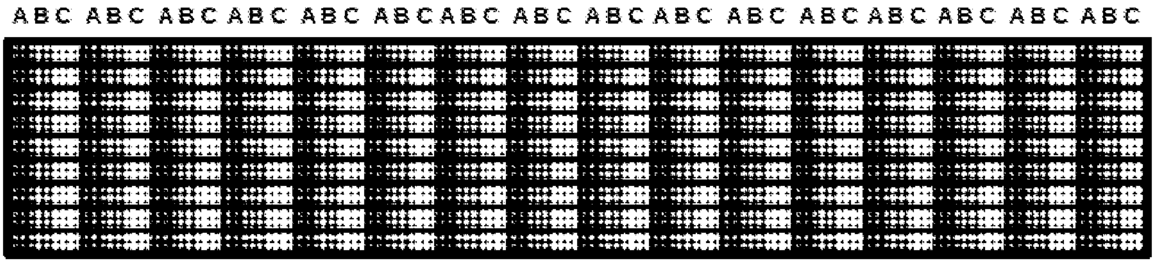

ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC ABC

FIG. 5A

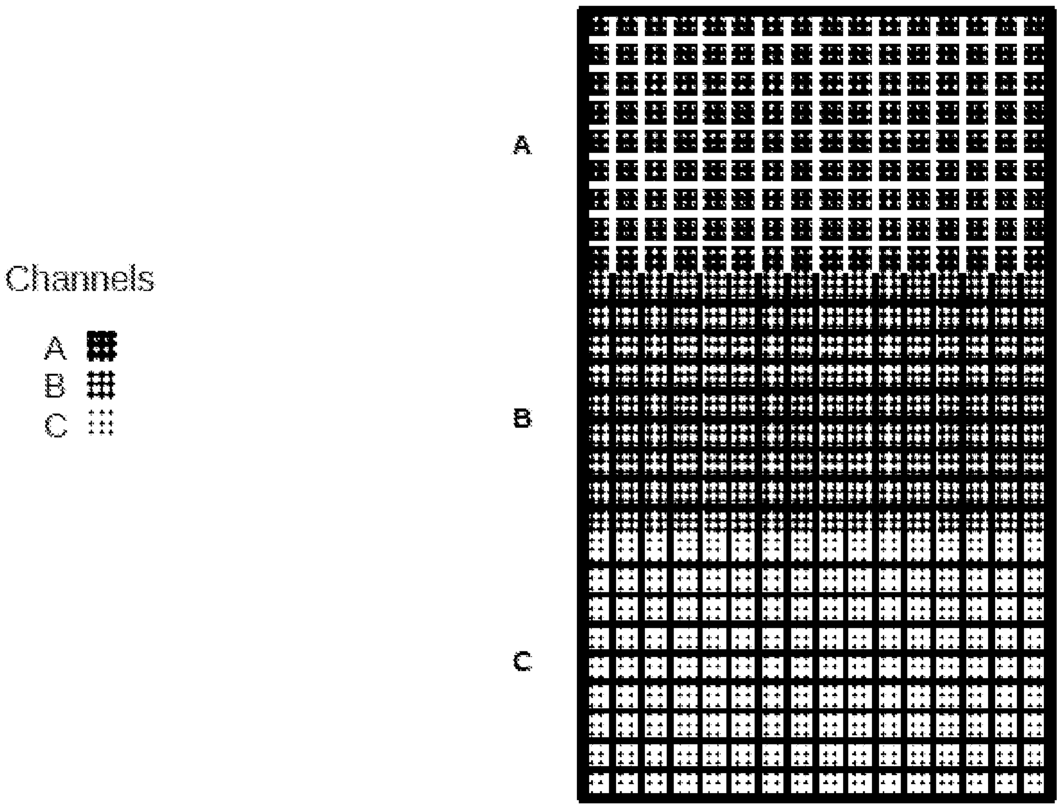

Channels
A
B
C
A
B
C

FIG. 5B

A

B

C

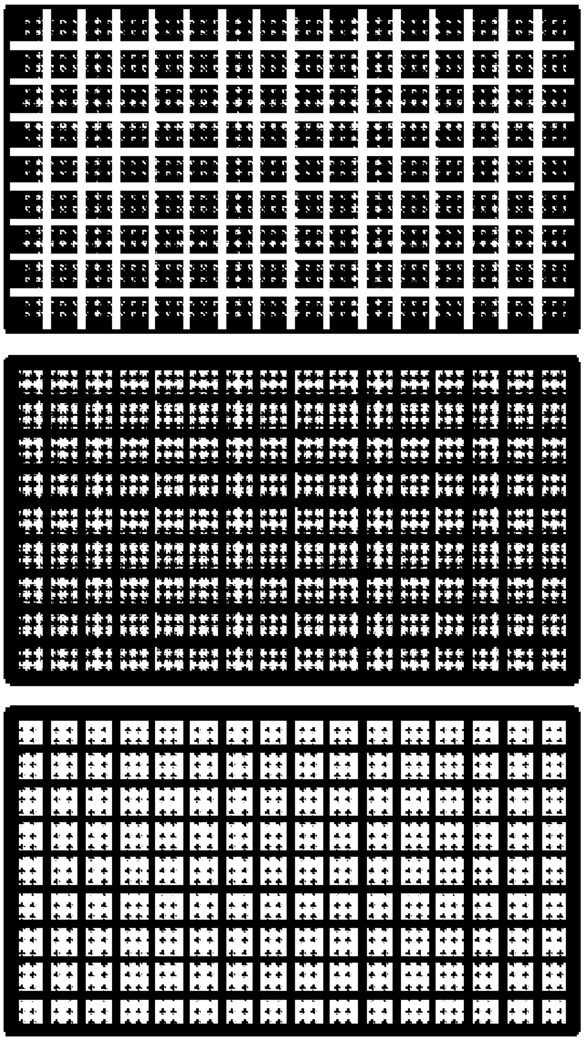
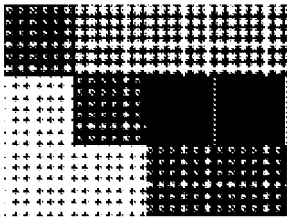


FIG. 5C

Y



Plane

Y	166	102	102	102
	97	166	71	71
	97	97	166	166

1D-LUT

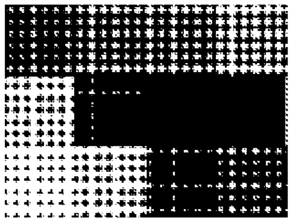
	0	1	2	3
Y	71	97	102	166

Indexes

Y	3	2	2	2
	1	3	0	0
	1	1	3	3

FIG. 6

ABC



A	B	C	A	B	C	A	B	C	A	B	C
126	165	60	1	101	205	2	104	203	1	103	205
234	97	62	127	168	60	126	70	205	124	71	203
236	98	61	234	98	63	124	167	60	123	165	62

FIG. 7A

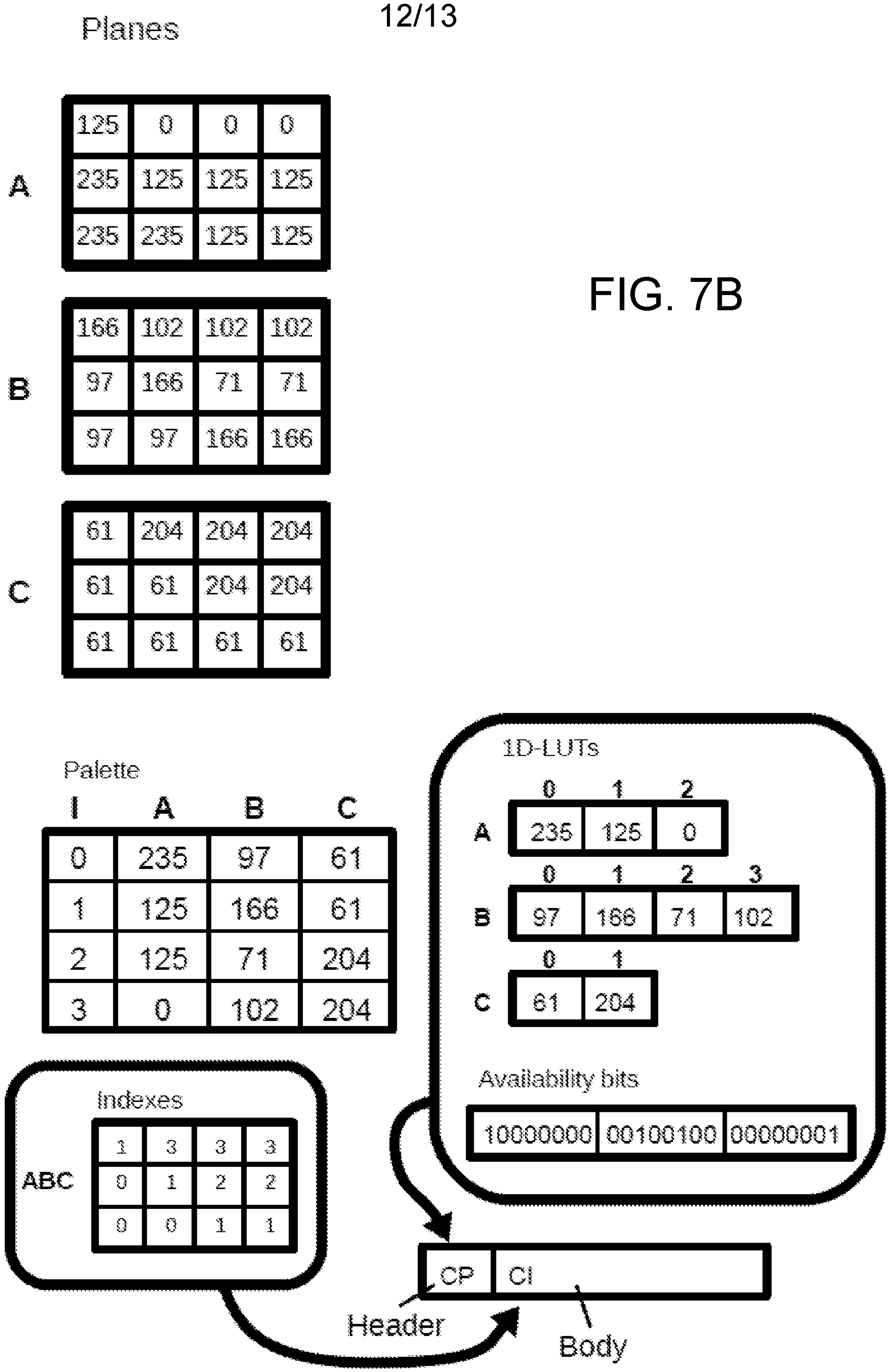
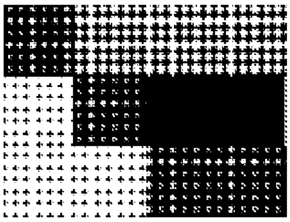

Planes
A
125 0 0 0
235 125 125 125
235 235 125 125
B
166 102 102 102
97 166 71 71
97 97 166 166
C
61 204 204 204
61 61 204 204
61 61 61 61
Palette
I A B C
0 235 97 61
1 125 166 61
2 125 71 204
3 0 102 204
1D-LUTs
A
0 1 2
235 125 0
B
0 1 2 3
97 166 71 102
C
0 1
61 204
Availability bits
10000000 00100100 00000001
Indexes
ABC
1 3 3 3
0 1 2 2
0 0 1 1
CP
CI
Header
Body

FIG. 7B

ABC



A	B	C	A	B	C	A	B	C	A	B	C
125	166	61	0	102	204	0	102	204	0	102	204
235	97	61	125	166	61	125	71	204	125	71	204
235	97	61	235	97	61	125	166	61	125	166	61

FIG. 7C

The following terms are registered trade marks and should be read as such wherever they occur in this document:

Hesburger
Google

ENCODER, DECODER AND METHOD EMPLOYING PALETTE UTILIZATION AND COMPRESSION

TECHNICAL FIELD

The present disclosure relates to encoders for encoding input data (D1) to generate corresponding encoded data (E2), and corresponding methods of encoding input data (D1) to generate corresponding encoded data (E2). Moreover, the present disclosure relates to decoders for decoding encoded data (E2) to generate corresponding decoded data (D3), and corresponding methods of decoding encoded data (E2) to generate corresponding decoded data (D3). Furthermore, the present disclosure is concerned with computer program products comprising a non-transitory computer-readable storage medium having computer-readable instructions stored thereon, the computer-readable instructions being executable by a computerized device comprising processing hardware to execute aforesaid methods. Additionally, the present disclosure concerns codecs including at least one aforementioned encoder and at least one aforementioned decoder.

BACKGROUND

Palettes, also known as “*look-up-tables*” (LUT’s) or “*color look-up-tables*” (CLUT’s), are conventionally used to describe some sort of information in some other form. As an example, in a conventional Graphics Interchange Format (GIF) file, a color palette is used to describe, for example, 32/64/128/256 different index values that are used to present original information of the GIF file in a lossy or lossless form. Likewise, conventional Portable Network Graphics format (PNG) files also contain palette information. Typically, only one palette is used for a given image file.

Each index value in a GIF file describes, for example, 24-bit Blue-Green-Red (BGR) color values, in a manner that some specific combinations of 8-bit color values are used for describing pixel values of the GIF file in red, green and blue channels. If a given color palette includes 256 index values, 768 bytes (= 3 x 256 bytes) are used in a GIF file to describe the given color palette. It will be appreciated here that only a limited number of different 24-bit color values can be described by using a limited number of different index values.

GIF employs two techniques for rendering more than 256 colors in a color image. A first technique involves splitting a given image into partial images, namely into blocks that are then processed as separate images would be, and issuing the partial images their own palettes. The first technique is rather inefficient, because each partial image, for example, has its own header, including information about its size, a format used and a palette used. Moreover, many software applications that are generally able to open GIF images do not understand these partial images correctly. Such software applications either make an animation from the partial images, or open only a first partial image.

A second technique for rendering more than 256 colors in a color image involves using transparency information to add more colors, in addition to the palette used. Considerable planning and optimization are required to make the second technique work even fairly efficiently. Moreover, the second technique yields insufficient efficiency when encoding data.

In a published US patent US2015110181A1 ("*Methods for palette prediction and intra block copy padding"*, Samsung Electronics Co. Ltd."), a method is provided that includes receiving a bitstream. The method also includes parsing the bitstream for a flag indicating whether a palette was used from a first coding unit or a second coding unit. The aforementioned method also includes decoding the first coding unit using the palette from the first or second coding unit indicated by the flag. The palette is determined based on which palette of the first or second coding unit improves compression performance. Moreover, a method is provided that includes receiving a bitstream with a predicted pixel. A coding unit and a reference unit are identified. A number of pixels of the coding unit and the reference unit overlap. A set of available pixels and a set of unavailable pixels of the reference unit are identified. The predicted pixel of the set of unavailable pixels is estimated as a pixel of the set of available pixels.

In a published US patent US20070053426A1 ("*Method and apparatus for enhancing performance of entropy coding, video coding method and apparatus using the method*", BAE-KEUN LEE *et al.*), there is described an apparatus for enhancing the performance of entropy coding in a multilayer-based codec and an entropy coding method which includes obtaining a distribution of a second coefficient included in an area of a base layer corresponding to the block, selecting a lookup table which is appropriate to the obtained distribution among a plurality of lookup tables, and transforming the first coefficient into a value mapped to the selected lookup table.

In a published technical document ("*Screen content coding test model 2 (SCM 2)*", Rajan Joshi *et al.*), there is described, from an encoder-side perspective, a HEVC Screen content coding test model 2 (SCM 2), which serves as a tutorial for an encoding model implemented in HM-15.0+RExt- 8.0+SCM-2.0 software. The published document has a purpose to share a common understanding of reference encoding methods supported in the HM-15.0+RExt-8.0+SCM-2.0 software, in order to facilitate an assessment of a technical impact of proposed new technologies during standardization processes.

In a published technical document ("*Palette-based Coding in the Screen Content Coding Extension of the HEVC Standard*", Rajan Joshi *et al.*), a technical overview of palette-based coding is described, wherein the palette-based coding is susceptible to being adopted into a test model for a screen content coding (SCC) extension of the known High Efficiency Video Coding (HEVC) standard. Key techniques that enable a palette mode to deliver significant coding gains for screen contents are highlighted, including palette table generation, palette table coding, and the coding methods for palette indices and escape colors (colours). Proposed and adopted techniques up to the first version of the working draft of HEVC SCC extension and test model SCM-2.0 are presented. Experimental results are provided to evaluate the performance of the palette mode in the second extension of HEVC.

In a published US patent US20140147040A1 (*"Image encoding device, image decoding device, image encoding method, and image decoding method"*, Tatsuya Tanaka *et al.*) , there is described an image encoding device which includes a deciding unit, an assigning unit, and an encoding unit. The deciding unit is configured to determine representative colors for expressing each of pixel blocks into which image data are divided. The assigning unit is configured to assign an index for identifying the representative color to each pixel in the pixel block. The encoding unit is configured to encode indices and the representative colors, the indices and the representative colors in each pixel box being arranged alternately so that two representative colors are discontinuously encoded.

In a published GB patent GB2371730A (*"Image data storage",* NEC Corp.), a method is described that if a user wants to store an image acquired, and it is determined that the data size of the image acquired from the site is larger than a designated storage frame in a memory region, then it is determined whether or not the number of colours in the palette of the acquired image data can be reduced. If it is possible to achieve a reduction, the number of palette colors is reduced to compress thereby the image, and the image data is updated based on the compressed palette. Conversely, if it is determined that the number of palette colours cannot be reduced, a control section determines that the image data cannot be stored and abandons the data, and a display section displays '*storage error*'.

In a published US patent US2015/0016501 A1 (*"Palette prediction in palette-based video coding», Liwei GUO et al.*), a method of in-palette-based coding is described, in which a video coder may form a so-called "*palette*" as a table of colours representing the video data of a given block. The video coder may code index values for one or more pixels values of a current block of video data, where the index values indicate entries in the palette that represent the pixel values of the current block. According to the method, a video coder determines one or more palette entries in a predictive palette that are copied to the current palette, and a number of new palette entries not in the predictive palette that are included in the current palette. The video coder calculates a size of the current palette equal to the sum of the number of the copied palette entries and the number of the new palette entries, and generates the current palette including the copied palette entries and the new palette entries.

In a published WO patent WO2007/029945 A1 (*"Method and apparatus for enhancing performance of entropy coding, video coding method and apparatus using the method"*, Samsung Electronics Co. Ltd.), an apparatus is described for enhancing the performance of entropy coding in a multilayer-based codec and an entropy coding method is described which includes obtaining a distribution of a second coefficient included in an area of a base layer corresponding to the block, selecting a lookup table which is appropriate to the obtained distribution among a plurality of lookup tables, and transforming the first coefficient into a value mapped to the selected lookup table.

SUMMARY

The present disclosure seeks to provide an improved encoder for encoding input data (D1) to generate corresponding encoded data (E2).

Moreover, the present disclosure seeks to provide an improved decoder for decoding encoded data (E2) to generate corresponding decoded data (D3).

A further aim of the present disclosure is to at least partially overcome at least some of the problems of the prior art, as discussed above.

In a first aspect, embodiments of the present disclosure provide a method of encoding input data (D1) to generate corresponding encoded data (E2), wherein the method includes encoding the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that the method further includes:

determining mutually different palettes to be used for encoding mutually different parts of the input data (D1);

using the mutually different palettes for encoding the mutually different parts of the input data (D1); and

including, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2) wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each of the mutually different parts of input data (D1) represents a block.

Optionally, the method includes using the palette selection indices to indicate at least one of:

(a) no palette is used;
(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;
(c) pre-defined palettes;
(d) dynamic palettes;
(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

Optionally, the method includes reusing or using a given palette for a plurality of parts of the input data (D1). In an example, the given palette is optionally an already known palette.

Optionally, the method includes using the same palette for a plurality of parts of the input data (D1).

Optionally, the method includes employing a Rate Distortion (RD) method to determine whether or not a given palette is suitable for encoding a given part of the input data (D1).

Optionally, the method determining of the mutually different palettes depends upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: colour, video content, audio content, image content, measurement data, genomic data, a statistical parameter of data blocks or data block sizes present in the input data (D1).

Optionally, the method determining of the mutually different palettes depends upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2).

Optionally, the method determining the mutually different palettes is performed on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks.

Optionally, the method includes employing change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block.

Optionally, the method includes delivering one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

More optionally, the method includes compressing the one or more of the mutually different palettes using mutually different compression methods.

Optionally, the method includes delivering one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

Optionally, the method includes generating compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

More optionally, the method includes communicating the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data.

More optionally, the method includes communicating the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

More optionally, the method includes employing at least one entropy-encoding method to compress the one channel or mutually different channels in the interleaved format.

More optionally, the method includes compressing the one or mutually different channels together or separately in the planar format.

More optionally, the method includes compressing data of a given channel by using one-dimensional look-up table coding.

In a second aspect, embodiments of the present disclosure provide a method of decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that the method includes:

determining mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of encoded data (E2) represents a block;

employing the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

Optionally, the palette selection indices indicate at least one of:

(a) no palette is used;
(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;
(c) pre-defined palettes;
(d) dynamic palettes;
(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

Optionally, the method includes reusing a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2).

Optionally, the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

Optionally, the method includes receiving one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

Optionally, the method includes receiving one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

Optionally, the method includes decompressing compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one channel or mutually different channels of the palette that are compressed:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

More optionally, the method includes employing an inverse of at least one entropy-encoding method employed to decompress the compressed palette data, when the one channel or mutually different channels are compressed in the interleaved format.

More optionally, the method includes employing an inverse of an encoding method employed to decompress the one or mutually different channels together or separately, when the one channel or mutually different channels are compressed in the planar format.

More optionally, characterized in that the method includes generating the palette from the different index values for the mutually different channels and the availability information indicative of the combinations of index values used in the palette.

In a third aspect, embodiments of the present disclosure provide an encoder for encoding input data (D1) to generate corresponding encoded data (E2), wherein the encoder is operable to encode the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that:

the encoder is operable to determine mutually different palettes to be used for encoding mutually different parts of the input data (D1);

the encoder is operable to use the mutually different palettes for encoding the mutually different parts of the input data (D1); and

the encoder is operable to include, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of input data (D1) represents a block.

Optionally, the encoder is operable to reuse a given palette for a plurality of parts of the input data (D1) when encoding the input data (D1).

Optionally, the encoder indicates in the palette selection indices at least one of:

(a) no palette is used;
(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;
(c) pre-defined palettes;
(d) dynamic palettes;
(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

Optionally, the encoder is operable to reuse or use a given palette for a plurality of parts of the input data (D1) when encoding the input data (D1).

Optionally, the encoder is operable to employ a Rate Distortion (RD) method to determine whether or not a given palette is suitable for encoding a given part of the input data (D1).

Optionally, the encoder is operable to determine the mutually different palettes depending upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: colour, video content, audio content, image content, measurement data, genomic data, a statistical parameter of data blocks or data block sizes present in the input data (D1).

Optionally, the encoder is operable to determine the mutually different palettes depending upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2).

Optionally, the encoder is operable to determine the mutually different palettes on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks.

Optionally, the encoder is operable to employ change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block.

Optionally, the encoder is operable to deliver one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

More optionally, the encoder is operable to compress the one or more of the mutually different palettes using mutually different compression methods.

Optionally, the encoder is operable to deliver one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

Optionally, the encoder is operable to generate compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

More optionally, the encoder is operable to communicate the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data.

More optionally, the encoder is operable to communicate the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

More optionally, the encoder is operable to employ at least one entropy-encoding method to compress the one channel or mutually different channels in the interleaved format.

More optionally, the encoder is operable to compress the one channel or mutually different channels together or separately in the planar format.

More optionally, the encoder is operable to compress data of a given channel by using one-dimensional look-up table coding.

In a fourth aspect, embodiments of the present disclosure provide a decoder for decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that:

the decoder is operable to determine mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2) wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of encoded data (E2) represents a block; and

the decoder is operable to employ the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

Optionally, the palette selection indices indicate at least one of:

(a) no palette is used;

(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;

(c) pre-defined palettes;

(d) dynamic palettes;

(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

Optionally, the decoder is operable to reuse a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2).

Optionally, the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

Optionally, the decoder is operable to receive one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

Optionally, the decoder is operable to receive one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

Optionally, the decoder is operable to decompress compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one channel or mutually different channels of the palette that are compressed:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

More optionally, the decoder is operable to employ an inverse of at least one entropy-encoding method employed to decompress the compressed palette data, when the one channel or mutually different channels are compressed in the interleaved format.

More optionally, the decoder is operable to employ an inverse of an encoding method employed to decompress the one or mutually different channels together or separately, when the one channel or mutually different channels are compressed in the planar format.

More optionally, the decoder is operable to generate the palette from the different index values for the mutually different channels and the availability information indicative of the combinations of index values used in the palette.

In a fifth aspect, embodiments of the present disclosure provide a codec including at least one encoder for encoding input data (D1) to generate corresponding encoded data (E2) pursuant to embodiments of the present disclosure, and at least one decoder for decoding the encoded data (E2) to generate corresponding decoded data (D3) pursuant to embodiments of the present disclosure.

In a sixth aspect, embodiments of the present disclosure provide a computer program product comprising a non-transitory computer-readable storage medium having computer-readable instructions stored thereon, the computer-readable instructions being executable by a computerized device comprising processing hardware to execute any of the aforementioned methods pursuant to embodiments of the present disclosure.

Additional aspects, advantages, features and objects of the present disclosure would be made apparent from the drawings and the detailed description of the illustrative embodiments construed in conjunction with the appended claims that follow.

It will be appreciated that features of the present disclosure are susceptible to being combined in various combinations without departing from the scope of the present disclosure as defined by the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The summary above, as well as the following detailed description of illustrative embodiments, is better understood when read in conjunction with the appended drawings. For the purpose of illustrating the present disclosure, exemplary constructions of the disclosure are shown in the drawings. However, the present disclosure is not limited to specific methods and apparatus disclosed herein. Moreover, those in the art will understand that the drawings are not to scale. Wherever possible, like elements have been indicated by identical numbers.

Embodiments of the present disclosure will now be described, by way of example only, with reference to the following diagrams wherein:

FIG. 1 is a schematic illustration of an encoder for encoding input data (D1) to generate corresponding encoded data (E2) and a decoder for decoding the encoded data (E2) to generate corresponding decoded data (D3), wherein the encoder and the decoder collectively form a codec, in accordance with an embodiment of the present disclosure;

FIG. 2 is a schematic illustration of a flow chart depicting steps of a method of encoding input data (D1) to generate corresponding encoded data (E2), in accordance with an embodiment of the present disclosure;

FIG. 3 is a schematic illustration of a flow chart depicting steps of a method of decoding encoded data (E2) to generate corresponding decoded data (D3), in accordance with an embodiment of the present disclosure;

FIGs. 4A, 4B, 4C and 4D collectively are a schematic illustration of an example of how palettes are determined for a given input data (D1), in accordance with an embodiment of the present disclosure;

FIGs. 5A, 5B and 5C are a group of illustrations of a data or palette including mutually different channels of the data or palette that will be compressed in various ways, wherein:

FIG. 5A is an illustration of an interleaved format, wherein the channels are to be compressed together, in accordance with an embodiment of the present disclosure;

FIG. 5B is an illustration of a planar format, wherein the channels are to be compressed together, in accordance with an embodiment of the present disclosure;

FIG. 5C is an illustration of a planar format, wherein the channels are to be compressed separately, in accordance with an embodiment of the present disclosure;

FIG. 6 is an illustration of data that is modified with 1-dimensional (1-D) look-up-table (LUT) for compression, in accordance with an embodiment of the present disclosure; and

FIGs. 7A, 7B and 7C collectively are an illustration of data, compressed palette based upon the data, the palette, 1-D LUT for each channels and availability bits of different available 1-D LUT value combinations, wherein some of the 1-D LUT value combinations are used in the palette, in accordance with an embodiment of the present disclosure.

In the accompanying diagrams, an underlined number is employed to represent an item over which the underlined number is positioned or an item to which the underlined number is adjacent. A non-underlined number relates to an item identified by a line linking the non-underlined number to the item.

DETAILED DESCRIPTION OF EMBODIMENTS

The following detailed description illustrates embodiments of the present disclosure and ways in which they can be implemented. Although some modes of carrying out the present disclosure have been disclosed, those skilled in the art would recognize that other embodiments for carrying out or practising the present disclosure are also possible.

In a first aspect, embodiments of the present disclosure provide a method of encoding input data (D1) to generate corresponding encoded data (E2), wherein the method includes encoding the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that the method further includes:

determining mutually different palettes to be used for encoding mutually different parts of the input data (D1);

using the mutually different palettes for encoding the mutually different parts of the input data (D1); and

including, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2).

Optionally, the method includes reusing a given palette for a plurality of parts of the input data (D1).

It will be appreciated here that the mutually different palettes are used for encoding the input data (D1) as a whole, but with regard to the different parts of the input data (D1), a palette used for a given part of the input data (D1) is optionally used or reused for one or more other parts of the input data (D1) if, for example, the given part and the one or more other parts have substantially similar content. In other words, a same given palette is optionally used or reused for substantially mutually similar parts of the input data (D1).. It will be appreciated that, by employing an expression "*substantially mutually similar*", it is meant that data content of these parts of the input data (D1) is constructed of similar values in this context. Moreover, optionally, a subset of a given palette is used for two or more different parts of the input data (D1), where data content of the input data (D1) contains many similar values, but some other values are different. In other words, this means that a major part of the data content of the input data (D1) is overlapping between those different parts of the input data (D1).

Moreover, it will be appreciated here that palettes can be used for mutually different types of data, for example, such as image data, video data, audio data, genomic data, measurement data, deoxyribonucleic acid (DNA) data, ribonucleic acid (RNA) data, biometric data, and so forth. As an example, an audio palette can be used to express an amplitude of an audio signal in a given frequency band. However, it will be appreciated that, contemporarily, color palettes are commonly used for images and videos, and represent an example application of embodiments of the present disclosure. Moreover, it is to be noted that the aforementioned method can be used with master palettes as well as adaptive palettes (see reference [1]).

According to an embodiment of the present disclosure, determining the mutually different palettes includes:

(i) selecting, from amongst a plurality of known palettes, a given palette that is suitable for encoding at least one part of the input data (D1); and/or

(ii) creating a new palette for encoding at least one part of the input data (D1).

Optionally, the new palette is selected based upon a rule-based analysis and/or a neural network analysis of the input data (D1), for example based upon earlier palettes that have been employed when encoding earlier data; namely, the neural network is trained on earlier input data (D1) and corresponding selection of palettes to determine when to employ a given example palette. Similarly, the neural network analysis is optionally employed when devising new types of palettes, for example by way of an iterative process, wherein palettes are progressively modified until they are substantially optimal for a part of the input data (D1) to be encoded. Such neural network analysis is optionally executed using contemporary high-speed reduced-instruction-set computing (RISC) processors. An example of a neural network capable of being employed is a feedforward neural network (see reference [8]).

As an example, the input data (D1) can be received as an 8-bit palette index image using a known palette. Optionally, in this regard, the method includes:

(i) determining whether or not the known palette is suitable for encoding at least one part of the input data (D1);

(ii) utilizing the known palette for that at least one part of the input data (D1) for which the known palette is determined to be suitable; and

(iii) creating one or more new palettes for remaining parts of the input data (D1).

Optionally, in this regard, the method includes employing a Rate Distortion (RD) method for determining whether or not a known palette is suitable for encoding a given part of the input data (D1).

For illustration purposes only, there will now be considered an example of how to determine whether or not a known palette is suitable for encoding a given part of the input data (D1), for example, such as a block or sequence of data (hereinafter referred to as a "*data block*", for the sake of clarity only). An RD value produced by encoding the data block using the known palette is calculated as a distortion between original data of the data block and corresponding decoded data, incremented by "*lambda*" times a rate required for palette data and indexed data, wherein the decoded data is reconstructed by using the known palette for each individual original data value and "*lambda*" is a Lagrange multiplier. Optionally, the distortion between the original data and the decoded data is calculated as a Sum of Absolute Difference (SAD), a Sum of Squared Difference (SSD), or some other difference between the original data and the decoded data. Moreover, optionally, the rate required for the palette data and the indexed data includes all bits that are required for encoding the data block, for example, including bits required for palette selection and optionally bits required for compressed palette delivery from a given encoder to a given decoder, and bits required for encoding the data block using the selected palette, for example selected known palette.

If the RD value produced by encoding the data block using the known palette is smaller than RD values produced by encoding the data block using other known or created palettes, or it is smaller than a pre-defined threshold value, then the known palette can be used to encode the data block. Optionally, the pre-defined threshold value depends on a desired quality parameter, for example for controlling whether the encoding is substantially lossless or lossy.

It will be appreciated here that the term "*known palette*" optionally refers to a generally known palette, or a palette that has been otherwise defined as fixed, or a palette that has been used previously for a same data block or some other data block and that has been indexed for reuse in embodiments of the present disclosure.

Moreover, regardless of whether or not a known palette is suitable, new palettes are optionally generated, either with or without quantization. In such a case, a best alternative, namely a palette to be used for encoding, is selected based upon one or more criteria, for example, based upon RD values, or in case of lossless coding, based upon which alternative would require a least amount of bits. It will be appreciated that it is sometimes advantageous to create a new palette and use the new palette, even if the new palette is slightly more inefficient for a particular block or sequence of data, if it is otherwise known that the new palette might be useful in encoding some other block or sequence of data. In other words, sometimes, a palette that is a best alternative for use when encoding data, for example based upon an RD value, is not necessarily selected. Optionally, with regard to the whole input data (D1) to be encoded, it is better to select an alternative palette that does not optimize the RD value of a given data block in question, but results in an optimized encoding solution as a whole or, for example, an alternative that results in faster subsequent decoding of data in a decoder. Moreover, a palette, or a subset of a palette, that is not a best alternative for certain blocks whose content overlaps with other blocks is sometimes, in practice, suitable to be used for using the same palette for them both.

According to an embodiment of the present disclosure, determining the mutually different palettes depends upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: color, video content, audio content, image content, measurement data, genomic data and/or a statistical parameter of data blocks or data block sizes present in the input data (D1). For example, there are optionally employed different palettes with similar indices that are able to be employed separately, based upon the content of the input data (D1); in such a situation, a palette to be used is optionally selected from amongst the different palettes that are known, or that are created as new, based upon various criteria, as described in the foregoing, and/or using any known method, such as described in reference [9]. As an example of how size can affect the palette selection, when data blocks are small, then typically palettes are also small, whereas when data blocks are large, there can be both large and small palettes.

It will be appreciated here that the determination of the palettes is not always dependent on the content and/or the type of the input data (D1), but is also optionally dependent upon data values or index values. Sometimes, when the palette determination is performed based upon indices of a given palette, information relating to the statistical parameter of data blocks or data block sizes is not required to be delivered separately from a given encoder to a given decoder. However, in another case, the palette determination is performed from the original input data (D1), wherein information relating to the statistical parameter of data blocks or data block sizes is required to be delivered from the given encoder to the given decoder.

According to an embodiment of the present disclosure, determining the mutually different palettes depends upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2); such characteristics, for example, can be indicative of whether or not reformatting or transcoding the data is required when decoding the encoded data (E2) to generate corresponding decoded data (D3). This dependence is potentially particularly advantageous in real-time streaming of data or interactive conferencing, for example, in a situation where it is known that a certain decoder to which the encoded data (E2) is to be delivered has certain unique properties, or is lacking certain properties, that are normally available. Moreover, this is potentially particularly advantageous when transcoding data, as aforementioned, wherein a transcoder takes into account properties of, for example, a display of a given decoder, namely supported palettes or a bit depth of colours of the display. For example, in such situations where the data is needed only for the display or for transcoding purposes, there are optionally selected one or more palettes that are much simpler than palettes required in a lossless coding. Moreover, this can be done when the decoder is lacking a property of supporting lossless decoding.

According to an embodiment of the present disclosure, determining the mutually different palettes is performed on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks. Optionally, in this regard, data partitioning can be implemented by splitting the input data (D1) into data blocks and/or by combining data blocks present in the input data (D1) into groups of data blocks.

Moreover, the data blocks and/or groups of data blocks are determined based upon an encoding of the input data (D1). Optionally, in this regard, a given data block and/or group of data blocks can be described, for example, by using a fixed block size or split/combine bits (see references [2] & [3]) for a given section of the input data (D1) or the entire input data (D1). Thus, boundaries of the data blocks and/or groups of data blocks can be delivered efficiently. In other words, “boundaries” as a term means that the data values inside the block can be known and bordered.

It will be appreciated here that not all individual data blocks require a separate palette to be determined. In other words, a given palette can be used by more than one data block and/or group of data blocks. Optionally, some data blocks might not use a palette at all, for example they are encoded in another manner completely, for example using another type of transformation that is not palette-based. .

Moreover, the aforementioned method can be used in combination with multiple mutually different encoding methods and standards. As an example, the aforementioned method can be used with a data block encoder described in reference [2]. As another example, the aforementioned method can be used with the Graphics Interchange Format (GIF), the Portable Network Graphics format (PNG) and the like.

According to an embodiment of the present disclosure, the method includes referencing the mutually different palettes on a block-by-block basis via use of palette selection indices in the encoded data (E2). Thus, the palette selection indices associate the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2), namely their respective data blocks and/or groups of data blocks. In other words, the mutually different palettes are indicated and delivered from a given encoder to a corresponding given decoder via the use of the palette selection indices.

As an example, different palettes and their respective palette selection indices can be used for different parts of video data, for example, such as video clips, video frames, views, channels, macro blocks, blocks, sections, and so forth.

Optionally, when a pre-defined or previously-used palette is used for encoding a given data block and/or group of data blocks, the used palette is referred to with a palette selection index.

Additionally, optionally, when a new palette is used for encoding a given data block and/or group of data blocks, the new palette is delivered together with the given data block and/or group of data blocks. Optionally, the new palette is then added, for example as a new dynamic palette, and is assigned a palette selection index, so as to enable reuse of the new palette in future. For example, it is occasionally feasible to encode two parts of the input data (D1) using a similar palette, thereby reducing the number of palettes, or their identification indices, whose identities or palette information have to be delivered from a given encoder to a given corresponding decoder.

Moreover, optionally, the aforementioned palette selection indices are defined in a following manner, wherein there are at least two palette selection indices employed:

(i) one or more of the palette selection indices are used to indicate different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed, and/or

(ii) one or more of the palette selection indices are reserved for pre-defined palettes, namely palettes that are generally known, for example, such as “master” palettes or a grayscale palette, and/or

(iii) one or more of the palette selection indices are used for reusable palettes, namely palettes that are used and delivered previously and are referred to for subsequent data blocks and/or groups of data blocks, and/or

(iv) one or more of the palette selection indices are used for palettes that are compressed using different compression methods and that are delivered together with their respective data blocks and/or groups of data blocks to which these palettes belong, and/or

(v) at least one of the palette selection indices is used to indicate that no palette is used.

It will be appreciated here that a given palette can be either reused for several data blocks or used for only a single data block. Furthermore, it will also be appreciated that it is not necessary to use all different palette selection index alternatives when the input data (D1) is encoded.

Optionally, a given palette selection index is used to describe a given palette that is already known or is delivered separately, for example after another palette selection index that indicates a delivery method used. Optionally, a given palette selection index also expresses that its corresponding palette has been delivered within a header in the encoded data (E2), or can be obtained or retrieved, for example, from a certain server, or from a default server by using a reference index for accessing the palette at the default server.

As an example, 256 palette selection indices can be defined, wherein:

(i) a palette selection index ‘0’ is used to indicate that no palette is used;

(ii) palette selection indices ‘1’ to ‘15’ are used to indicate different palette delivery and/or compression methods;

(iii) palette selection indices ‘16’ to ‘127’ are used for pre-defined palettes, for example, such as fixed static palettes (hereinafter referred to as “fixed palette selection indices”);

(iv) palette selection indices ‘128’ to ‘254’ are used for dynamic palettes, namely palettes that can be added and/or updated, for example based upon already-delivered palettes (hereinafter referred to as “dynamic palette selection indices”); and

(v) a palette selection index ‘255’ is reserved for future use to enable addition of more palettes or different palette delivery methods.

It is to be noted here that the number of different palettes used and their palette selection indices are not limited to a certain number, and can vary depending on the different parts of the input data (D1).

Optionally, different palettes can be used for different contents or for different users or companies. At least some of the fixed static palettes can be content-specific or user-specific or company-specific. Thus, the number of fixed static palettes potentially varies, for example based upon a user, a company and/or a type of the content being processed.

As an example, a particular company could determine its own fixed color palette containing colors that exist in its registered logo, which is typically used in documents or images prepared by that particular company. Similarly, a company that sequences genome data could determine a palette that describes groups having resulted for such sequencing efficiently. An advantage of using such company-specific palettes is that these company-specific palettes are not required to be separately delivered or included within the encoded data (E2). Examples of such companies that optionally utilize palettes of their own are, for example, Finnish *Hesburger* with its logo with red, white and blue colors as well as Google with variety of different logos it uses at its search page.

Use of specific palettes enables a higher compression ratio or a better reconstruction quality to be achieved for some company-specific or user-specific content. Thus, the fixed static palettes provide a business opportunity to sell fixed palette selection indices.

Moreover, optionally, the fixed static palettes include color palettes, for example, as described in reference [4].

On the other hand, the dynamic palettes include palettes that have been delivered during the encoding, or have been delivered in previous known encoding sessions. Thus, dynamic palette selection indices enable reuse of already delivered palettes, namely previously-used palettes.

The aforementioned method enables utilization of different palettes for different parts of the input data (D1) by making use of an efficient palette delivery method, for example, as will be described later. These palette delivery methods make it possible to achieve a higher compression ratio, or a better signal-to-noise ratio between the input data (D1) and decoded data (D3) generated from the encoded data (E2) compared to known contemporary methods.

It will be appreciated that the term “palette selection indices” does not refer to the concept of palette index values used in, for example, color palettes that indicate an array index of a color triplet.

It is hitherto not known to utilize different palettes for different parts of input data (D1) by using palette selection indices that describe the palettes used for the different parts between a given encoder and a corresponding given decoder. Moreover, in embodiments of the present disclosure, only one palette selection index is used to describe a palette used for a given data block and/or group of data blocks.

Moreover, optionally, the palette selection indices are delivered as they are, for example, in a form a data stream. Alternatively, optionally, the palette selection indices are compressed using a suitable encoding method, prior to delivery.

Moreover, optionally, the palettes are delivered from the given encoder to the given decoder by using one or more data files, or by streaming from the given encoder to the given decoder. Optionally, a palette used for encoding a given data block into an encoded data block is delivered within a header of the encoded data block or within the encoded data block itself.

It will be appreciated that the input data (D1) can be encoded either in an interleaved manner or in a planar format. A planar format potentially results in a relatively simpler data structure in the encoded data (E2), but also potentially requires less computing effort to implement. In contradistinction, the aforementioned interleaved format is potentially more robust to unauthorized eavesdropping of the encoded data (E2), for example in situations where data security is important. Moreover, for example when limited computational resources are available at decoding devices, it is optionally feasible to deliver only a part of the input data (D1) encoded via use of palettes in the encoded data (E2).

Moreover, in some examples, only a part of the input data (D1) is encoded and/or delivered using one or more palettes. As an example, pixel values of a YUV image can be delivered with two data values, wherein a first data value describes a luminance value, namely a data value for the Y channel, and a second data value describes a chrominance value, namely a data value for the U and V channels, along with a palette selection index of a used color palette. In this example, the luminance value can be delivered with or without a one-dimensional Look-Up Table (1-D LUT). The used color palette corresponds to two channels, namely the U and V channels, and can also be called a “chroma” palette or a “chrominance” palette. The chrominance palette is particularly beneficial to use with I420 (12 bits per pixel) and I422 (16 bits per pixel) pixel formats, namely “*data format*”, but can also be used, for example, for I444 (24 bits per pixel), BGR, I444A and BGRA pixel formats. The chrominance palette potentially prevents a size of the used color palette from growing too large, while efficiently compressing redundancy of data of the U and V channels, or alternatively, of I444 or BGR channels in the I444A or BGRA images and/or videos.

Furthermore, according to an embodiment of the present disclosure, the method includes employing change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block. If the palette used for encoding the given data block has changed, a palette selection index describing the palette is also delivered for the given data block. Otherwise, if the palette has not changed, no additional information is required to be delivered for the given data block as regards the palette used.

Moreover, optionally, the change bits are written and/or delivered as a bit string. Optionally, the bit string of the change bits is compressed, for example using range coding with or without entropy modification (EM). In a case where a lot of changes occur or a case where only a few changes occur, a probability of bits ‘0’ and ‘1’ in the bit string are very different, and an associated entropy of bits is small. In both of these cases, the bit string can be compressed efficiently, for example, by using range coding.

Furthermore, there are often situations when a given palette is not already known between a given encoder and a corresponding given decoder. Such palettes are described and delivered from the given encoder to the given decoder.

Optionally, in this regard, the method includes delivering one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

Optionally, in this regard, the method includes compressing the one or more of the mutually different palettes using mutually different compression methods. Optionally, these different compression methods are assigned their own palette selection indices. Such a palette selection index indicates to a given decoder which compression method has been employed for compressing a given palette.

Optionally, in this regard, the method includes employing a palette delivery method that utilizes a most suitable compression method for a given palette. Optionally, the palette delivery method employs a server from where the given palette can be obtained or retrieved. This enables a significant decrease in data size in comparison to conventional methods that describe a given palette with all original index values, and optionally, for all data blocks by using raw interleaved channel data values, for example represented by three bytes for each palette index of an RGB palette.

According to an embodiment of the present disclosure, the method includes delivering one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette. In other words, a difference between a used palette and a known palette is delivered instead of the used palette. The use of delta values enables switching between different palettes when encoding the different parts of the input data (D1) into the encoded data (E2).

Moreover, according to an embodiment of the present disclosure, the method includes generating compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format (namely together);

(ii) in a planar format (namely together or separately); and/or

(iii) in a format that indicates different index values, for example as a 1-dimensional look-up-table (1-D LUT), for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

Throughout the present disclosure, the one or mutually different channels of the palette are hereinafter interchangeably referred to as "*palette data*".

According to an embodiment of the present disclosure, the method includes communicating the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data. According to another embodiment, the method includes communicating the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

The aforementioned method can be implemented via a given encoder. Optionally, the compressed palette data is delivered from the given encoder to a corresponding given decoder by using one or more data files, or by streaming from the given encoder to the given decoder. Optionally, the compressed palette data is delivered within a header or within the encoded data (E2).

According to an embodiment of the present disclosure, when compressing in the interleaved format, the method includes employing at least one entropy-encoding method to compress the one channel or mutually different channels in the interleaved format. In the interleaved format, index values of all the mutually different channels of the given palette are compressed together. Optionally, in this regard, the at least one entropy-encoding method is used to compress index values of the mutually different channels in a sequence in which these index values occur in the interleaved format. As an example, the mutually different channels can be compressed using Huffman coding, Variable-Length Coding (VLC), range coding, or via a reference to a database, such as in database coding disclosed in the United Kingdom patent application GB2509055A (see reference [7]). Optionally, the encoded data (E2) is also encrypted.

According to an embodiment of the present disclosure, when compressing in the planar format, the method includes compressing the one or mutually different channels together or separately. As index values of one channel are typically independent of index values of other channels, the mutually different channels can be compressed as planar channels using optionally different encoding methods.

For a given channel, a suitable encoding method can be selected based upon an inspection of data or index values of the given channel. As an example, if the given channel includes large index values that occur repeatedly, VLC can be used to compress the data or index values of the given channel. As another example, if the given channel includes small index values that occur sequentially, Delta coding or ODelta coding can be used to compress the data or index values of the given channel. Herein, the term "*Delta coding*" refers to a method of storing or transmitting data in a form of differences between sequential data rather than complete data files or data values, while the term "*ODelta coding*" refers to a differential form of encoding based upon wraparound in a binary or integer counting regime, for example as described in the United Kingdom patent document GB 1412937.3 (see reference [6]).

Moreover, optionally, when compressing palette data, or other types of data in the planar format, the method includes compressing data of a given channel by using 1-D LUT coding. In the 1-D LUT coding, original index values of the given channel are replaced by new index values specified in a 1-D LUT. The 1-D LUT coding is particularly beneficial in a case where the original index values are numbers that are sparsely distributed in a whole dynamic range of the index values, or are not numbers that are replaced by indexing to numbers.

Beneficially, the new index values are relatively small as compared to the original index values. Therefore, optionally, Delta or ODelta coding is used to compress these new index values. As aforementioned, the term "*Delta coding*" refers to a method of storing or transmitting data in a form of differences between sequential data rather than complete data files, while the term "*ODelta coding*" refers to a differential form of encoding based upon wraparound in a binary counting regime, for example as described in the aforementioned United Kingdom patent document GB 1412937.3 (see reference [6]).

Moreover, optionally, a total number of different index values is reduced when the 1-D LUT coding is used with linear or non-linear quantization. In this regard, a count of original index values is taken into account when quantized index values are defined for a 1-D LUT, so as to minimize distortion due to the quantization. When lossy compression is used, it is beneficial to take into account how much distortion will be caused to the original index values when the original index values are replaced with quantized index values and reconstructed.

Optionally, in this regard, the method includes optimizing the compression of the data of the given channel by using a Rate Distortion (RD) method. This enables a high compression ratio to be achieved, while also enabling a high quality reconstruction to be achieved when the encoded data (E2) is subsequently decoded at a given decoder. As an example, a squared Euclidean distance can be used to represent the distortion caused by lossy compression of palette data. The compression can then be efficiently optimized by using an RD method, wherein an RD value is calculated as a distortion caused by compression (namely, quantization) added by lambda times a rate required for the palette data and indexed data.

It is to be noted here that various different quantization methods can be used when original index values are quantized.

A quantized index value for the 1-D LUT can be one of the original index values, or can be any other value that lies between a range of the original index values, for example an average, a weighted average, a median, or a mode.

According to an embodiment of the present disclosure, the method includes compressing the mutually different channels by delivering different index values, for example as a 1-D LUT, for each channel (hereinafter referred to as "*channel data*") separately, and delivering availability information indicative of combinations of index values that are used in the given palette. This is particularly beneficial when the given palette is large, but each channel contains only a few different index values, namely originally or after quantization or using a 1-D LUT.

Optionally, the availability information is delivered using availability bits. Delivering the availability information is particularly beneficial when all possible combinations of the index values are not available in the given palette.

It will be appreciated that an order of combinations is also defined, so as to enable a given encoder and a corresponding given decoder to create a similar palette based upon the channel data and the availability bits. Optionally, the order of combinations is pre-defined and fixed. Alternatively, optionally, information indicative of the order of combinations is delivered from the given encoder to the corresponding given decoder.

Moreover, optionally, the order of combinations is changed, so as to further improve the compression of the palette data. It is to be noted here that the order of combinations has a large influence on the compression of the input data (D1). Therefore, when the order of combinations is changed, it is beneficial to take into account both a total number of bits required to deliver the encoded data (E2) and a total number of bits required to deliver the compressed palette data.

Moreover, optionally, if amounts of the used and unused combinations are very different, the availability bits can be compressed by employing, for example, range coding with, namely as symbols, or without, namely as bits or symbols, an Entropy Modifying (EM) encoding method that is described in reference [5].

Moreover, it will be appreciated that when mutually different channels of a given palette contain different numbers of mutually different index values, the number of different index values for each channel is delivered separately. Moreover, the number of availability bits depends on a multiplication of the number of different index values in each channel. Optionally, in this regard, 1-D LUT coding with quantization is particularly beneficial to use. The 1-D LUT coding can be used with quantization to reduce the number of different index values to a smaller number. As a result, the number of possible combinations for the channels of the given palette decreases considerably, which, in turn, reduces the number of availability bits required for describing combinations actually used in the given palette.

In a second aspect, embodiments of the present disclosure provide a method of decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that the method includes:

determining mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2); and

employing the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

Optionally, the method includes reusing a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2).

According to an embodiment of the present disclosure, the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis. Additionally, optionally, the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

According to an embodiment of the present disclosure, the method includes receiving one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

According to an embodiment of the present disclosure, the method includes receiving one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette. The use of delta values enables switching between different palettes when decoding the encoded data (E2) to generate the decoded data (D3).

According to an embodiment of the present disclosure, the method includes decompressing compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one or mutually different channels of the palette that are compressed:

(i) in an interleaved format (namely together);

(ii) in a planar format (namely together or separately); and/or

(iii) in a format that indicates different index values, for example as a 1-dimensional look-up-table (1-D LUT), for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

The aforementioned method can be implemented via a given decoder. Moreover, the aforementioned method can be used in combination with multiple mutually different decoding methods and standards. As an example, the aforementioned method can be used with a data block decoder described in reference [3]. As another example, the aforementioned method can be used with GIF, PNG and the like.

Optionally, the compressed palette data is received via one or more data files, or via streaming from a given encoder to the given decoder, or is accessed from a database.

According to an embodiment of the present disclosure, when the one or mutually different channels have been compressed in the interleaved format, the method includes employing an inverse of at least one entropy-encoding method that was employed at the given encoder to decompress the compressed palette data. As an example, the compressed palette data can be decompressed using Huffman decoding, variable-length decoding, range decoding, or via a reference to a database, such as in database coding disclosed in the United Kingdom patent application GB2509055A (see reference [7]).

According to an embodiment of the present disclosure, when the one or mutually different channels of palette data, or other types of data, have been compressed in the planar format, the method includes employing an inverse of an encoding method that was employed at the given encoder to decompress the one or mutually different channels together or separately.

According to an embodiment of the present disclosure, the method includes generating the palette from the different index values, for example as a 1-D LUT, for the mutually different channels and the availability information indicative of the combinations of index values used in the palette. Optionally, in this regard, the method includes receiving information indicative of an order of the combinations from the given encoder.

Furthermore, optionally, the generated palette is utilized directly during decoding of the encoded data (E2) to reconstruct the decoded data (D3).

As an example, 8-bit palette indices of an encoded image data are replaced by 24-bit RGB color values to create a color image during reconstruction.

As another example, the generated palette is used when the decoded data (D3) is displayed or printed, for example when a display device or a printer device, respectively, supports large palettes or color look-up-tables (CLUT's). It is to be noted here that a CLUT is conventionally used instead of or together with a color palette. Typically, a CLUT describes an LUT that enables efficient transformation of used color index values to true colors or vice versa.

In a third aspect, embodiments of the present disclosure provide an encoder for encoding input data (D1) to generate corresponding encoded data (E2), wherein the encoder is operable to encode the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that:

the encoder is operable to determine mutually different palettes to be used for encoding mutually different parts of the input data (D1);

the encoder is operable to use the mutually different palettes for encoding the mutually different parts of the input data (D1); and

the encoder is operable to include, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2).

Optionally, the encoder is operable to reuse a given palette for a plurality of parts of the input data (D1) when encoding the input data (D1), as described earlier.

According to an embodiment of the present disclosure, when determining the mutually different palettes, the encoder is operable to:

(i) select, from amongst a plurality of known palettes, a given palette that is suitable for encoding at least one part of the input data (D1); and/or

(ii) create a new palette for encoding at least one part of the input data (D1).

Moreover, according to an embodiment of the present disclosure, the encoder is operable to employ an RD method to determine whether or not a known palette is suitable for encoding a given part of the input data (D1), as described earlier.

Moreover, regardless of whether or not a known palette is suitable, new palettes are optionally generated, either with or without quantization. In such a case, a best alternative, namely a palette to be used for encoding, is selected based upon one or more criteria, for example, based upon RD values, or in case of lossless coding, based upon which alternative would require a least amount of bits.

According to an embodiment of the present disclosure, the encoder is operable to determine the mutually different palettes depending upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: color, video content, audio content, image content, measurement data, genomic data and/or a statistical parameter of data blocks or data block sizes present in the input data (D1).

According to an embodiment of the present disclosure, the encoder is operable to determine the mutually different palettes depending upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2).

According to an embodiment of the present disclosure, the encoder is operable to determine the mutually different palettes on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks.

Moreover, the data blocks and/or groups of data blocks are determined based upon an encoding of the input data (D1). Optionally, in this regard, a given data block and/or group of data blocks can be described, for example, by using a fixed block size or split/combine bits (see references [2] & [3]) for a given section of the input data (D1) or the entire input data (D1). Thus, boundaries of the data blocks and/or groups of data blocks can be delivered efficiently.

Moreover, the aforementioned encoder can be used together with multiple mutually different encoders. As an example, the aforementioned encoder can be used with a data block encoder described in reference [2]. As another example, the aforementioned encoder can be used with GIF, PNG and the like.

According to an embodiment of the present disclosure, the encoder is operable to reference the mutually different palettes on a block-by-block basis via use of palette selection indices in the encoded data (E2), as described earlier.

According to an embodiment of the present disclosure, the encoder is operable to employ change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block.

Optionally, the encoder is operable to compress a bit string of these change bits, for example using range coding with or without entropy modification (EM), as aforementioned.

According to an embodiment of the present disclosure, the encoder is operable to deliver one or more of the mutually different palettes via the encoded data (E2) in a compressed form. Optionally, in this regard, the encoder is operable to compress the one or more of the mutually different palettes using mutually different compression methods.

According to an embodiment of the present disclosure, the encoder is operable to deliver one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

According to an embodiment of the present disclosure, the encoder is operable to generate compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format (namely together);

(ii) in a planar format (namely together or separately); and/or

(iii) in a format that indicates different index values, for example as a 1-D LUT, for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

Optionally, these different compression methods are assigned their own palette selection indices. Such a palette selection index indicates to a given decoder which compression method has been employed for compressing a given palette in respect of generating the encoded data (E2).

Optionally, the encoder is operable to deliver the compressed palette data to a corresponding given decoder by using one or more data files, or by streaming to the given decoder. Optionally, the encoder is operable to deliver the compressed palette data within a header or within the encoded data (E2).

According to an embodiment of the present disclosure, the encoder is operable to communicate the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data. According to another embodiment, the encoder is operable to communicate the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

According to an embodiment of the present disclosure, when compressing in the interleaved format, the encoder is operable to employ at least one entropy-encoding method to compress the mutually different channels in the interleaved format. Optionally, in this regard, the encoder is operable to use the at least one entropy-encoding method to compress index values of the mutually different channels in a sequence in which these index values occur in the interleaved format. As an example, the encoder is operable to compress the mutually different channels using Huffman coding, VLC, range coding, or via a reference to a database, such as in database coding disclosed in the United Kingdom patent application GB2509055A (see reference [7]).

According to an embodiment of the present disclosure, when compressing in the planar format, the encoder is operable to compress the one channel or mutually different channels together or separately, as described earlier.

Moreover, optionally, when compressing in the planar format, the encoder is operable to compress data of a given channel by using 1-D LUT coding, as described earlier. Optionally, in this regard, the encoder is operable to optimize the compression of the data of the given channel by using a rate distortion method.

According to an embodiment of the present disclosure, the encoder is operable to compress the palette data by delivering different index values, for example as a 1-D LUT, for each channel separately, and delivering availability information indicative of combinations of index values that are used in the palette. This is particularly beneficial when the palette is large, but each channel contains only a few different index values, namely originally or after quantization or using a 1-D LUT.

Optionally, the availability information is delivered using availability bits. Delivering the availability information is particularly beneficial when all possible combinations of the index values are not available in the palette.

It will be appreciated that an order of combinations is also defined, so as to enable the encoder and a corresponding decoder to create a similar palette based upon the channel data and the availability bits. Optionally, the order of combinations is pre-defined and fixed. Alternatively, optionally, information indicative of the order of combinations is delivered from the encoder to the corresponding decoder.

In a fourth aspect, embodiments of the present disclosure provide a decoder for decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that:

the decoder is operable to determine mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2); and

the decoder is operable to employ the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

Optionally, the decoder is operable to reuse a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2), as described earlier.

According to an embodiment of the present disclosure, the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis.

Moreover, the aforementioned decoder can be used together with multiple mutually different decoders. As an example, the aforementioned decoder can be used with a data block decoder described in reference [3]. As another example, the aforementioned decoder can be used with GIF, PNG and the like.

According to an embodiment of the present disclosure, the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

According to an embodiment of the present disclosure, the decoder is operable to receive one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

According to an embodiment of the present disclosure, the decoder is operable to receive one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

According to an embodiment of the present disclosure, the decoder is operable to decompress compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one channel or mutually different channels of the palette that are compressed:

(i) in an interleaved format (namely together);

(ii) in a planar format (namely together or separately); and/or

(iii) in a format that indicates different index values, for example as a 1-D LUT,

for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

Optionally, the decoder is operable to receive the compressed palette data via one or more data files, or via streaming from a given encoder, or to access the compressed palette data from a database, for example as aforementioned.

According to an embodiment of the present disclosure, when the palette data has been compressed in the interleaved format, the decoder is operable to employ an inverse of at least one entropy-encoding method that was employed at a given encoder to decompress the compressed palette data. As an example, the decoder is operable to decompress the compressed palette data using Huffman decoding, variable-length decoding, range decoding, or via a reference to a database, such as in database coding disclosed in the United Kingdom patent application GB2509055A (see reference [7]).

According to an embodiment of the present disclosure, when the palette data has been compressed in the planar format, the decoder is operable to employ an inverse of an encoding method that was employed at the given encoder to decompress the mutually different channels together or separately.

According to an embodiment of the present disclosure, the decoder is operable to generate the palette from the different index values, for example as a 1-D LUT, for the mutually different channels and the availability information indicative of the combinations of index values used in the palette. Optionally, in this regard, the decoder is operable to receive information indicative of an order of the combinations from the given encoder.

In a fifth aspect, embodiments of the present disclosure provide a codec including at least one encoder for encoding input data (D1) to generate corresponding encoded data (E2) pursuant to embodiments of the present disclosure, and at least one decoder for decoding the encoded data (E2) to generate corresponding decoded data (D3) pursuant to embodiments of the present disclosure. Such a codec is, for example, employed in scientific instruments for sensing a given region, for generating the input data (D1), for encoding the input data (D1) to generate corresponding encoded data (E2), for example for storage in data memory of the scientific instruments, and for providing review of the encoded data (E2), for example replay of the encoded data (E2), spatially locally at the scientific instruments. The scientific instruments include, for example medical ultrasonic sensing apparatus, MRI imagers, endoscopic inspection devices, dental X-ray (Röntgen) apparatus, and so forth, but not limited thereto.

An example codec has been provided in conjunction with FIG. 1 as explained in more detail below. The codec includes at least one encoder and at least one decoder.

The at least one encoder employs a method of encoding input data (D1) as described in the foregoing including utilizing mutually different palettes and compression of palette data. The at least one encoder is thus operable to encode the input data (D1) to generate corresponding encoded data (E2).

Moreover, the at least one decoder is operable to perform an inverse of operations executed in the at least one encoder, to decode the encoded data (E2) to generate corresponding decoded data (D3).

Optionally, the codec includes fixed static palettes and dynamic palettes. Optionally, the fixed static palettes are referenced via one or more palette databases, and are downloaded from a dedicated server.

Optionally, the decoded data (D3) is exactly similar to the input data (D1), as in a lossless mode of operation. Alternatively, optionally, the decoded data (D3) is substantially similar to the input data (D1), as in a lossy mode of operation. Yet alternatively, optionally, the decoded data (D3) is different to the input data (D1), for example by way of transcoding or via use of one or more transformations, for example, such as a color conversion, a format conversion, an upscaling conversion, a downscaling conversion, a cropping conversion, a rotation conversion, a flipping conversion, but not limited thereto, but retains substantially similar information present in the input data (D1); for example, the decoded data (D3) is usefully made different to the input data (D1) when reformatting of the decoded data (D3) is also required, for example to be compatible with different types of communication platforms, software layers, communication devices, display devices and so forth.

A color conversion can be deemed to be optimal for the following reasons:

(i) so as to achieve as good a reconstruction as is possible; and/or

(ii) so as to execute the optimal color conversion as regards, for example, properties of a display device to be used for presenting the decoded data (D3).

However, an essential issue is the following: if a palette exists that contains values, for example in the RGB color space, and it is desired to reconstruct an image in the YUV color space, then it is considerably more efficient firstly to transform the palette values from the RGB color space to the YUV color space and then to map those YUV values from the transformed palette, than it would be if the values were firstly mapped from the original RGB palette and then a thereby generated image was transformed into the YUV color space. It will be appreciated that the amount of necessary operations is considerably smaller in the first option here.

The at least one encoder includes a data processing arrangement for processing the input data (D1) to generate the corresponding encoded data (E2) pursuant to embodiments of the present disclosure. Optionally, the data processing arrangement of the at least one encoder is implemented by employing at least one Reduced Instruction Set Computing (RISC) processor that is operable to execute program instructions as elucidated earlier. RISC processors are operable, for example, to implement relatively simple data manipulations at a very great speed, whilst simultaneously utilizing relatively little electrical power, namely a characteristic that is highly advantageous when embodiments of the present disclosure are implemented in mobile battery-powered apparatus, for example “*black box*” flight recorders for aircraft.

Furthermore, optionally, the at least one encoder is operable to communicate the encoded data (E2) to a data server and/or data storage for storing in a database. The data server and/or data storage is arranged to be accessible to the at least one decoder, which is beneficially compatible with the at least one encoder, for subsequently decoding the encoded data (E2).

In some examples, the at least one decoder is optionally operable to access the encoded data (E2) from the data server and/or data storage.

In alternative examples, the at least one encoder is optionally operable to stream the encoded data (E2) to the at least one decoder, either via a data carrier or a data communication network or via a direct connection. Moreover, it is to be noted that a device equipped with a hardware-based or software-based encoder can also communicate directly with another device equipped with a hardware-based or software-based decoder.

In yet other alternative examples, the at least one decoder is optionally implemented so as to retrieve the encoded data (E2) from a non-transitory (namely non-transient) computer-readable storage medium, such as a hard drive and a Solid-State Drive (SSD).

The at least one decoder includes a data processing arrangement for processing the encoded data (E2) to generate the corresponding decoded data (D3) pursuant to embodiments of the present disclosure. Optionally, the data processing arrangement of the at least one decoder is implemented by employing at least one RISC processor that is operable to execute program instructions as elucidated earlier; such a RISC processor is capable of performing relatively simpler concatenated operations at a very high speed, and is suitable for decoding data provided in a streamed format, for example in real-time. RISC processors are contemporarily employed in smart phones for performing data processing of heterodyned wireless signals at high speed, and enable embodiments of the present disclosure to be conveniently implemented on such smart phones, for example.

Optionally, in this regard, the data processing arrangement of the at least one encoder is operable to execute the aforementioned method of encoding in parallel for different parts of the input data (D1). Correspondingly, optionally, the data processing arrangement of the at least one decoder is operable to execute the aforementioned method of decoding in parallel for different parts of the encoded data (E2), to reconstruct different parts of the decoded data (D3).

When embodiments of the present disclosure are implemented in a multicasting manner, there is a plurality of such decoders that are employed.

Optionally, the codec is implemented within a single device. Alternatively, optionally, the codec is effectively implemented between multiple devices. Optionally, the codec is implemented as custom-design digital hardware, for example via use of one or more Application-Specific Integrated Circuits (ASIC's). Alternatively or additionally, optionally, the codec is implemented using computing hardware that is operable to execute program instructions, for example provided to the computing hardware on a non-transient (non-transitory) machine-readable data carrier.

As an example, the at least one encoder and/or the at least one decoder can be beneficially employed in consumer electronics apparatus, wireless communication apparatus and associated systems, cameras, smart phones, tablet computers, personal computers, scientific measuring apparatus, communications equipments, display devices, videoconferencing equipments, satellites, but not limited thereto.

In a sixth aspect, embodiments of the present disclosure provide a computer program product comprising a non-transitory computer-readable storage medium having computer-readable instructions stored thereon, the computer-readable instructions being executable by a computerized device comprising processing hardware to execute any of the aforementioned methods pursuant to embodiments of the present disclosure.

Optionally, the computer-readable instructions are downloadable from a software application store, for example, from an "*App store*" to the computerized device.

Next, embodiments of the present disclosure will be described with reference to figures.

Referring to FIG. 1, embodiments of the present disclosure concern:

(i) an encoder **110** for encoding input data (D1) to generate corresponding encoded data (E2), and corresponding methods of encoding the input data (D1) to generate the encoded data (E2);

(ii) a decoder **120** for decoding the encoded data (E2) to generate corresponding decoded data (D3), and corresponding methods of decoding the encoded data (E2) to generate the decoded data (D3); and

(iii) a codec **130** including a combination of at least one encoder and at least one decoder, for example a combination of the encoder **110** and the decoder **120**.

FIG. 1 is merely an example, which does not unduly limit the scope of the claims herein. It is to be understood that the specific designation for the codec **130** is provided as an example and is not to be construed as limiting the codec **130** to specific numbers, types, or arrangements of encoders and decoders. A person skilled in the art will recognize many variations, alternatives, and modifications of embodiments of the present disclosure.

Referring now to FIG. 2, there is provided a flow chart depicting steps of a method of encoding input data (D1) to generate corresponding encoded data (E2), in accordance with an embodiment of the present disclosure. The method is depicted as a collection of steps in a logical flow diagram, which represents a sequence of steps that can be implemented in hardware, software, or a combination thereof, for example as aforementioned.

At a step **202**, mutually different palettes to be used for encoding mutually different parts of the input data (D1) are determined.

At a step **204**, the mutually different palettes are used for encoding the mutually different parts of the input data (D1). However, optionally, it will be appreciated that some of the palettes are reused or used for encoding a plurality of parts for which the palettes are suitable.

The step **204** can be executed in parallel for the different parts of the input data (D1), to generate different parts of the encoded data (E2).

At a step **206**, information associating the mutually different palettes with their respective parts of the input data (D1) is included within the encoded data (E2).

The steps **202** to **206** are only illustrative and other alternatives can also be provided where one or more steps are added, one or more steps are removed, or one or more steps are provided in a different sequence without departing from the scope of the claims herein.

Referring now to FIG. 3, there is provided a flow chart depicting steps of a method of decoding encoded data (E2) to generate corresponding decoded data (D3), in accordance with an embodiment of the present disclosure. The method is depicted as a collection of steps in a logical flow diagram, which represents a sequence of steps that can be implemented in hardware, software, or a combination thereof, for example as aforementioned.

At a step **302**, mutually different palettes that are used for generating mutually different parts of the encoded data (E2) are determined by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2).

At a step **304**, the mutually different palettes are employed to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

The step **304** can be executed in parallel for the different parts of the encoded data (E2), to reconstruct different parts of the decoded data (D3).

The steps **302** to **304** are only illustrative and other alternatives can also be provided where one or more steps are added, one or more steps are removed, or one or more steps are provided in a different sequence without departing from the scope of the claims herein.

FIGs. 4A, 4B, 4C and 4D are collectively schematic illustrations of an example of a manner in which palettes are determined for a given input data (D1), in this example image data, in accordance with an embodiment of the present disclosure.

In FIG. 4A, there is shown an image **402** that includes eighty value pairs arranged in a 10 X 8 array.

Pursuant to embodiments of the present disclosure, the image **402** is split into six data blocks, depicted as data blocks **404**, **406**, **408**, **410**, **412** and **414** in FIG. 4B. Suitable palettes are then determined for the data blocks **404**, **406**, **408**, **410**, **412** and **414**. Optionally, in this regard, original value pairs are replaced by palette index values using selected palettes.

Thus, a first palette employed for the data block **404** is represented as follows:

Original Value Pair	Palette Index Value
(126, 70)	0
(236,195)	1
(145,168)	2
(167,152)	3

A second palette employed for the data block **406** and **414** is represented as follows:

Original Value Pair	Palette Index Value
(126, 70)	0
(236,195)	1
(152,138)	2

A third palette employed for the data block **408** is represented as follows:

Original Value Pair	Palette Index Value
(236,195)	0

A fourth palette employed for the data block **412** is represented as follows:

Original Value Pair	Palette Index Value
(145,168)	0
(167,152)	1

With reference to FIG. 4B, the data block **410** is not encoded by use of palettes, and the original value pairs are therefore delivered. The original value pairs of the data block **410** are optionally delivered, for example, as a stream of data as herewith represented:

181, 175, 157, 178, 236, 195, 157, 178

In FIG. 4C, there are shown palette selection indices associated with the data blocks **404**, **406**, **408**, **410**, **412** and **414**, namely palette selection indices ‘23’, ‘1’, ‘16’, ‘0’, ‘18’, and ‘128’, respectively. These palette selection indices are optionally delivered on a block-by-block basis, for example, as a stream as herewith represented:

23, 1, 16, 0, 18, 128

It should be noted that even if there is no same palette selection index used for plurality of blocks in the abovementioned example, it is possible to use same palette selection index for multiple blocks.

Moreover, the palette index values are optionally delivered on a block-by-block basis, for example, as a stream of data values as herewith represented:

0, 3, 3, 3, 3, 3, 1, 0, 2, 2, 2, 2, 1, 1, 0, 0, 0, 0 ...

Furthermore, a table below provides palette selection indices defined for the illustrated example.

Palette Selection Index	Definition or content of palette
0	No palette
1	Compression method 1
2	Compression method 2
…	…
15	Compression method 15
16	(236,195)
…	…
18	(145,168), (167,152)
…	…
23	(126,70),(236,195),(145,168),(167,152)
…	…
128	(126,70), (236,195), (152,138)

It will be appreciated that a palette or look-up-table (LUT) may contain 1 to N values, wherein N is an integer, which again may be elements consisting of one or more physical data values. For example, RGBA values of a 2x2 block can be described using one palette index, in which case N = 16 (4x4); for example, the RGBA values are generated from an imaging sensor, for example a CCD or CMOS sensor, which is directed towards a given scene. The data values of a generic palette can have any conceivable bit depth, for example 1, 2, 4, 8, 16, 24, 32. A generic palette may contain 1 to M indices, wherein M is an integer, and wherein M is typically a small value, for example less than or equal to a value 256 as in GIF or PNG representations. However, M in the index is optionally used to describe even all the different color values in the 32-bit data, which would mean that the palette index would be equal to interleaved 32-bit data values, and M thus has a value 2^{32}. In practice, it is often not appropriate to use such big palettes, and typically the value of M is less than or equal to 65536 (16-bit value), and often it is less than or equal to 256 (8-bit value). The size of the palette is always advantageously selected, based upon how many combinations there are in the data, or can be or needs to be, for example selected depending upon how much data there is to be processed.

In FIG. 4D, there are shown palette index values of these eighty value pairs, namely palette index values of a two-channel palette employed in the image **402**. For the sake of clarity only, channels of the two-channel palette have been marked as ‘A’ and ‘B’ in FIG. 4D.

FIGs. 4A to 4D are merely examples, which do not unduly limit the scope of the claims herein. A person skilled in the art will recognize many variations, alternatives, and modifications of embodiments of the present disclosure.

Referring next to FIG. 5A, there is shown an illustration of example data or palette expressed via use of channels A, B and C in an interleaved format, wherein the data or the palette associated with the channels A, B and C are to be compressed mutually together, in accordance with an embodiment of the present disclosure. Moreover, in FIG. 5B, there is shown an illustration of example data or palette expressed via use of channels A, B and C in a planar format, wherein the data or the palette associated with the channels A, B, and C are to be compressed mutually together, in accordance with an embodiment of the present disclosure. Furthermore, in FIG. 5C, there is shown an illustration of example data or palette, wherein the data or the palette associated with the channels A, B and C are in a planar format and are to be compressed mutually separately, in accordance with an embodiment of the present disclosure. The example data represents, for example, image data, audio data, sensor data, genetic data, but not limited thereto.

Referring next to FIG. 6, data denoted by “Y” is, for example, color-defining data associated with one or more images. When expressed in a planar format, the data “Y” has integer values, as shown, in a range of 71 to 166. The integer values are expressible in a 1-dimensional look-up-table (1-D LUT), having values 71, 97, 102 and 166, which are referenced by index values 0, 1, 2 and 3, respectively. Using the index values, the color-defining values can also be defined efficiently in a planar format, by using fewer bits, with reference to the 1-D LUT. There is thereby provided an efficient method of encoding the data denoted by “Y”, in accordance with an embodiment of the present disclosure.

In FIG. 7A, an example of representing data to be encoded in an interleaved format for channels A, B and C is shown at bottom and an image is shown at top.

An example of a lossy method of encoding data, which is illustrated in FIG. 7A, is provided in FIG. 7B, in accordance with an embodiment of the present disclosure. Referring next to FIG. 7B, the data to be encoded is represented in three planes associated with channels A, B and C. For the values present in the data to be encoded, it is feasible to generate a lossy palette defined by an index parameter value I, and values for the channels A, B and C, as illustrated. The palette can, for example, be described by a collection of 1-dimensional look-up-tables (1-D LUT), wherein each of the channels A, B and C has its associated 1-D LUT as shown. By employing the palette and its index parameter values I, it is feasible to represent the data to be encoded as lossy encoded data, wherein the encoded data includes a header portion that includes compressed palette, denoted by "*CP*", and a body portion that includes compressed data as compressed index values based upon palette, denoted by "*CI*". Different portions of CP and/or CI data are optionally further encoded by using entropy encoding methods, for example, such as range coding or VLC coding.

An example of a decoded data generated from the lossy encoded data of FIG. 7B is illustrated in FIG. 7C, wherein the decoded data is represented, for channels A, B and C, in an interleaved format shown at bottom and as an image shown at top.

Methods associated with FIG. 5A to FIG. 7B are usefully employed in an encoder for encoding input data (D1) to generate corresponding encoded data (E2), and an inverse of the methods is usefully employed when decoding the encoded data (E2) to generate corresponding decoded data (D3). One example of decoded data is shown in FIG. 7C. In FIG. 6, two upper images show the data to be encoded and corresponding decoded data as per a lossless mode of operation.

FIGs. 5A to 7C are merely examples, which do not unduly limit the scope of the claims herein. A person skilled in the art will recognize many variations, alternatives, and modifications of embodiments of the present disclosure.

Modifications to embodiments of the invention described in the foregoing are possible without departing from the scope of the invention as defined by the accompanying claims. Expressions such as "including", "comprising", "incorporating", "consisting of", "have", "is" used to describe and claim the present invention are intended to be construed in a non-exclusive manner, namely allowing for items, components or elements not explicitly described also to be present. Reference to the singular is also to be construed to relate to the plural. Numerals included within parentheses in the accompanying claims are intended to assist understanding of the claims and should not be construed in any way to limit subject matter claimed by these claims.

REFERENCES

[1] Palette (computing) - Wikipedia, the free encyclopedia
(accessed May 8, 2015)
URL: http://en.wikipedia.org/wiki/Palette_%28computing%29

[2] Encoder and Method
(accessed May 8, 2015);
URL: http://gurulogic.com/files/US8675731B2.pdf

[3] Decoder and Method
(accessed May 8, 2015)
URL: http://gurulogic.com/files/US2014044191A1.pdf

[4] List of color palettes - Wikipedia, the free encyclopedia
(accessed May 8, 2015)
URL: http://en.wikipedia.org/wiki/List_of_color_palettes

[5] Entropy Modifier and Method
(accessed May 8, 2015)
URL: http://gurulogic.com/files/US8754791B1.pdf

[6] ODelta application, as described in patent document WO2014/131517A1

[7] Database coding, as described in patent document GB1222240.2 and EP13005263.2

[8] Feedforward neural network - Wikipedia, the free encyclopedia
(accessed May 8, 2015)
URL: http://en.wikipedia.org/wiki/Feedforward_neural_network

[9] Pictaculous – a color palette generator
(accessed May 8, 2015)
URL: http://www.pictaculous.com/

CLAIMS

We claim:

1. A method of encoding input data (D1) to generate corresponding encoded data (E2), wherein the method includes encoding the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that the method further includes:

determining mutually different palettes to be used for encoding mutually different parts of the input data (D1);

using the mutually different palettes for encoding the mutually different parts of the input data (D1); and

including, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each of the mutually different parts of input data (D1) represents a block.

2. The method of claim 1, characterized in that the palette selection indices indicate at least one of:

(a) no palette is used;

(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;

(c) pre-defined palettes;

(d) dynamic palettes;

(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

3. The method of claim 1 or 2, characterized in that the method includes reusing or using a given palette for a plurality of parts of the input data (D1).

4. The method of claim 1, 2 or 3, characterized in that the method includes employing a Rate Distortion (RD) method to determine whether or not a given palette is suitable for encoding a given part of the input data (D1).

5. The method of any one of the preceding claims, characterized in that the determining of the mutually different palettes depends upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: colour, video content, audio content, image content, measurement data, genomic data, a statistical parameter of data blocks or data block sizes present in the input data (D1).

6. The method of any one of the preceding claims, characterized in that the determining of the mutually different palettes depends upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2).

7. The method of any one of the preceding claims, characterized in that the determining of the mutually different palettes is performed on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks.

8. The method of any one of the preceding claims, characterized in that the method includes employing change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block.

9. The method of any one of the preceding claims, characterized in that the method includes delivering one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

10. The method of claim 9, characterized in that the method includes compressing the one or more of the mutually different palettes using mutually different compression methods.

11. The method of any one of the preceding claims, characterized in that the method includes delivering one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

12. The method of any one of the preceding claims, characterized in that the method includes generating compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

13. The method of claim 12, characterized in that the method includes communicating the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data.

14. The method of claim 12, characterized in that the method includes communicating the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

15. The method of claim 12, 13 or 14, characterized in that the method includes employing at least one entropy-encoding method to compress the one channel or mutually different channels in the interleaved format.

16. The method of claim 12, 13 or 14, characterized in that the method includes compressing the one or mutually different channels together or separately in the planar format.

17. The method of claim 16, characterized in that the method includes compressing data of a given channel by using one-dimensional look-up table coding.

18. A method of decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that the method includes:

determining mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of encoded data (E2) represents a block; and

employing the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

19. The method of claim 18, characterized in that the palette selection indices indicate at least one of:

(a) no palette is used;

(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;

(c) pre-defined palettes;

(d) dynamic palettes;

(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

20. The method of claim 18 or 19, characterized in that the method includes reusing or using a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2).

21. The method of claim 18, 19 or 20, characterized in that the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

22. The method of claim 18, 19, 20 or 21, characterized in that the method includes receiving one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

23. The method of any one of claims 18 to 22, characterized in that the method includes receiving one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

24. The method of any one of claims 18 to 23, characterized in that the method includes decompressing compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one channel or mutually different channels of the palette that are compressed:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

25. The method of claim 24, characterized in that the method includes employing an inverse of at least one entropy-encoding method employed to decompress the compressed palette data, when the one channel or mutually different channels are compressed in the interleaved format.

26. The method of claim 24, characterized in that the method includes employing an inverse of an encoding method employed to decompress the one or mutually different channels together or separately, when the one channel or mutually different channels are compressed in the planar format.

27. The method of claim 24, 25 or 26, characterized in that the method includes generating the palette from the different index values for the mutually different channels and the availability information indicative of the combinations of index values used in the palette.

28. An encoder (110) for encoding input data (D1) to generate corresponding encoded data (E2), wherein the encoder (110) is operable to encode the input data (D1) into one or more symbols in the encoded data (E2), wherein the one or more symbols represent data as defined by a plurality of palettes indicated and/or included in the encoded data (E2), characterized in that:

the encoder (110) is operable to determine mutually different palettes to be used for encoding mutually different parts of the input data (D1);

the encoder (110) is operable to use the mutually different palettes for encoding the mutually different parts of the input data (D1); and

the encoder (110) is operable to include, within the encoded data (E2), information associating the mutually different palettes with their respective parts of the input data (D1) encoded into the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of input data (D1) represents a block.

29. The encoder (110) of claim 28, characterized in that the palette selection indices indicate at least one of:

(a) no palette is used;

(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;

(c) pre-defined palettes;

(d) dynamic palettes;

(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

30. The encoder (110) of claim 28 or 29, characterized in that the encoder (110) is operable to reuse or use a given palette for a plurality of parts of the input data (D1) when encoding the input data (D1).

31. The encoder (110) of claim 28, 29 or 30, characterized in that the encoder (110) is operable to employ a Rate Distortion (RD) method to determine whether or not a given palette is suitable for encoding a given part of the input data (D1).

32. The encoder (110) of any one of claims 28 to 31, characterized in that the encoder (110) is operable to determine the mutually different palettes depending upon content and/or type of the input data (D1), wherein the content and/or the type of the input data (D1) relate to at least one of: colour, video content, audio content, image content, measurement data, genomic data, a statistical parameter of data blocks or data block sizes present in the input data (D1).

33. The encoder (110) of any one of claims 28 to 32, characterized in that the encoder (110) is operable to determine the mutually different palettes depending upon characteristics describing one or more decoders that are to be used for decoding the encoded data (E2).

34. The encoder (110) of any one of claims 28 to 33, characterized in that the encoder (110) is operable to determine the mutually different palettes on a block-by-block basis, for data blocks present in the input data (D1) and/or data blocks derived from the input data (D1) by combining and/or splitting data blocks present in the input data (D1) and/or groups of data blocks.

35. The encoder (110) of any one of claims 28 to 34, characterized in that the encoder (110) is operable to employ change bits on a block-by-block basis, to express whether or not a palette used for encoding a given data block is changed with respect to a palette used for encoding a preceding data block.

36. The encoder (110) of any one of claims 28 to 35, characterized in that the encoder (110) is operable to deliver one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

37. The encoder (110) of claim 36, characterized in that the encoder (110) is operable to compress the one or more of the mutually different palettes using mutually different compression methods.

38. The encoder (110) of any one of claims 28 to 37, characterized in that the encoder (110) is operable to deliver one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

39. The encoder (110) of any one of claims 28 to 38, characterized in that the encoder (110) is operable to generate compressed palette data by compressing one channel or mutually different channels of a given palette:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the given palette.

40. The encoder (110) of claim 39, characterized in that the encoder (110) is operable to communicate the one or more symbols via a mutually different communication channel to that employed for communicating the compressed palette data.

41. The encoder (110) of claim 39, characterized in that the encoder (110) is operable to communicate the one or more symbols via a mutually similar communication channel to that employed for communicating the compressed palette data.

42. The encoder (110) of claim 39, 40 or 41, characterized in that the encoder (110) is operable to employ at least one entropy-encoding method to compress the one channel or mutually different channels in the interleaved format.

43. The encoder (110) of claim 39, 40 or 41, characterized in that the encoder (110) is operable to compress the one channel or mutually different channels together or separately in the planar format.

44. The encoder (110) of claim 43, characterized in that the encoder (110) is operable to compress data of a given channel by using one-dimensional look-up table coding.

45. A decoder (120) for decoding encoded data (E2) to generate corresponding decoded data (D3), characterized in that:

the decoder (120) is operable to determine mutually different palettes that are used for generating mutually different parts of the encoded data (E2) by analyzing information, included within the encoded data (E2), associating the mutually different palettes with their respective parts of the encoded data (E2), wherein the information includes palette selection indices for referencing the mutually different palettes on a block-by-block basis, wherein each block is associated with one palette selection index, wherein each part of the mutually different parts of encoded data (E2) represents a block; and

the decoder (120) is operable to employ the mutually different palettes to decode their respective parts of the encoded data (E2) to generate the decoded data (D3).

46. The decoder (120) of claim 45, characterized in that the palette selection indices indicate at least one of:

(a) no palette is used;

(b) different palette delivery and/or compression methods by which associated palettes are being delivered and/or compressed;

(c) pre-defined palettes;

(d) dynamic palettes;

(e) reservation for future use to enable addition of more palettes or different palette delivery methods.

47. The decoder (120) of claim 45 or 46, characterized in that the decoder (120) is operable to reuse or use a given palette for a plurality of parts of the encoded data (E2) when decoding the encoded data (E2).

48. The decoder (120) of claim 45, 46 or 47, characterized in that the information includes change bits that express whether or not a palette used for generating a given encoded data block is changed with respect to a palette used for generating a preceding encoded data block.

49. The decoder (120) of claim 45, 46, 47 or 48, characterized in that the decoder (120) is operable to receive one or more of the mutually different palettes via the encoded data (E2) in a compressed form.

50. The decoder (120) of any one of claims 45 to 49, characterized in that the decoder (120) is operable to receive one or more of the mutually different palettes via use of delta values relative to a pre-defined or previously-used palette.

51. The decoder (120) of any one of claims 45 to 50, characterized in that the decoder (120) is operable to decompress compressed palette data indicated and/or included in the encoded data (E2) to generate a palette, wherein the compressed palette data includes one channel or mutually different channels of the palette that are compressed:

(i) in an interleaved format;

(ii) in a planar format; and/or

(iii) in a format that indicates different index values for each of the mutually different channels, together with availability information indicative of combinations of index values used in the palette.

52. The decoder (120) of claim 51, characterized in that the decoder (120) is operable to employ an inverse of at least one entropy-encoding method employed to decompress the compressed palette data, when the one channel or mutually different channels are compressed in the interleaved format.

53. The decoder (120) of claim 51, characterized in that the decoder (120) is operable to employ an inverse of an encoding method employed to decompress the one or mutually different channels together or separately, when the one channel or mutually different channels are compressed in the planar format.

54. The decoder (120) of claim 51, 52 or 53, characterized in that the decoder (120) is operable to generate the palette from the different index values for the mutually different channels and the availability information indicative of the combinations of index values used in the palette.

55. A codec (130) including at least one encoder (110) of any of claims 28 to 44 for encoding input data (D1) to generate corresponding encoded data (E2),

and at least one decoder (120) of any of claims 45 to 54 for decoding the encoded data (E2) to generate corresponding decoded data (D3).

56. A computer program product comprising a non-transitory computer-readable storage medium having computer-readable instructions stored thereon, the computer-readable instructions being executable by a computerized device comprising processing hardware to execute a method as claimed in any of claims 1 to 27.