

ENCODING AND DECODING VIDEO SEQUENCES COMPRISING REFERENCE PICTURE SETS

Technical field

5

The invention relates to a method of encoding a video sequence comprising Reference Picture Sets (RPSs), a method of decoding a video sequence comprising RPSs, corresponding computer programs and computer program products, a corresponding encoder, and a corresponding
10 decoder.

Background

High Efficiency Video Coding (HEVC) is a new video coding standard
15 currently being developed in the Joint Collaborative Team - Video Coding (JCT-VC). JCT-VC is a collaborative project between the Moving Picture Experts Group (MPEG) and the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T). Currently, a committee draft is defined that includes a number of new tools which makes HEVC
20 considerably more efficient than prior art video coding standards, in particular H.264/AVC.

HEVC is a hybrid codec that uses multiple reference pictures for inter-prediction. It includes a picture marking process in which reference pictures can be marked as "used for short-term reference", "used for long-term
25 reference", or "unused for reference". If marked "unused for reference", the reference picture is turned into a non-reference picture and cannot be used for inter-prediction any more. A picture marked "unused for reference" cannot be re-marked later to be used for short-term or long-term reference.

The marking process in HEVC is controlled by RPSs. An RPS is a set
30 of picture identifiers which identify reference pictures. The RPS is sent in

each slice, and reference pictures which are indicated in the RPS will be kept in the Decoder Picture Buffer (DPB) and marked as "used for short-term reference" or "used for long-term reference".

As an example, the RPS information may contain the values "-4, -6, 4". This means that the current picture can predict from, i.e., copy pixels from, the picture four frames back (in display order) since the figure -4 is present. It will also be able to predict from the picture six frames back and even from the picture four frames in the future. Thus, the decoder can discard all the images in its buffer except for the three described above. This is a robust way for the decoder to discard pictures. In practice, the decoder may have to keep these images until they are displayed, but they will not be used for inter-prediction again. For simplicity, in the remainder of this disclosure this situation will be considered as if the decoder can discard these images.

Sometimes the information comprised in an RPS can be rather lengthy. As an example, the following RPS is used for test purposes during HEVC standardization: "-3, -2, 1, 2, 5, 6". Encoding this example RPS may require up to 33 bits, which is a noticeable amount, in particular at very low bit rates and small image sizes.

One key observation is that the RPSs are typically not completely random. Rather, they can be reused over and over again. As an example, we consider a sequence of 18 images from a configuration file used for test purposes in HEVC standardization.

Pictures in HEVC are identified by their Picture Order Count (POC) values (PicOrderCntVal), also known as full POC values. These numbers represent the output order, also referred to as the display order, of the pictures. That is, a picture with POC = 57 will be displayed directly after a picture with POC = 56. However, the images are not always transmitted in the order they are displayed. For instance, the encoder may first transmit the picture with POC = 0, followed by POC = 8, followed by POC = 4, and so

forth. The decoder has to keep track of the pictures and display them in the correct order. In the example from HEVC standardization, the 18 pictures will be transmitted in the order indicated in Fig. 1.

As can be seen in Fig. 1, some RPSs are sent several times. For instance, the RPS sent for POC=6 (-2, -4, -6, 2) is the same as for POC=14. Therefore, the standard allows RPSs to be sent in Sequence Parameter Sets (SPSs) in addition to sending RPSs in slice headers. SPSs comprise data which does not need to be sent for every slice. Typically, SPSs are sent only once per sequence, or as often as the possibility of random access is desired. For instance, if the bit stream is broadcasted, it may be sufficient to send the SPS every second, since this would make it possible to switch channel every second. It should be noted that, for it to be possible to switch channel every second, a Clean Random Access (CRA) picture, or an Instantaneous Decoding Refresh (IDR) picture, also needs to be sent every second. A CRA picture is a picture that is not predicted from any previous picture, and no picture which follows the CRA picture in output order predicts from any picture that precedes the CRA picture in output order. An IDR picture is a CRA for which no picture which follows the IDR in decoding order may refer to any picture that precedes the IDR in decoding order.

In the SPS it is possible to specify the eight recurring RPSs of the example shown in Fig. 1 and assign indices to them, as is shown in Fig. 2. The information sent in the slice header now only has to refer to an RPS index in the SPS, as is illustrated in Fig. 3. Using RPS indices requires fewer bits than sending the RPSs themselves.

To this end, using RPS indices, as is illustrated in Fig. 3, considerably reduces the amount of bits required for sending RPSs since the bulk of the data is sent in the SPS instead, which is sent less frequently. Still, it turns out that it is possible to compress the RPS information even further. By comparing two rows in Fig. 1 one can notice a similarity between them. For instance, every number in the RPS for POC=1 is equal to a corresponding

number in the RPS for POC=6 if “5” is added to it. That is the first value “-1” in the RPS for POC=1 is equal to $-6 + 5$. The second value “1” is equal to $-4 + 5$. The third value “3” is equal to $-2 + 5$. The only exception to this rule is the second to last number “5” in the RPS for POC=1. It would need a “0” in the RPS for POC=6, but an image cannot predict from itself.

As it turns out, every RPS in Fig. 1 can be predicted from another RPS. This leads to the following way of describing RPS data in an SPS, e.g., the RPS data in Fig. 1:

- For each row, i.e., RPS, it is specified from which other RPS inter-prediction should be made. For instance, predicting from the preceding RPS is indicated by sending the value “-1” in `delta_idx_minus1`, which is a parameter in the SPS (see short-term RPS syntax Fig. 4).
- Then, the value to add, “5” in the example, is transmitted using the values `delta_rps_sign` and `abs_delta_rps_minus1`, which are parameters in the SPS (see short-term RPS syntax Fig. 4).

By sending RPS information in this way a lot of bits can be saved. The amount of saved bits is about 50% of the bits used for sending RPSs in the SPS, measured for an older version of the configuration files used for testing HEVC reference picture structures. Since SPS data is a very small part of the total video bit stream data, the overall effect is less than 50%, but compressing data efficiently is still important.

Typically, an RPS is sent once, in an SPS, and subsequent slices simply indicate which RPS should be used, by using an RPS index. In some situations, however, the encoder may want to use an RPS which is not in the SPS. The encoder has then the option of sending the RPS explicitly, i.e., in a slice, as is described above. Whether inter-prediction is used or the RPS is encoded value-by-value is signaled for each RPS using the parameter `inter_ref_pic_set_prediction_flag`, which is part of the short-term RPS shown in Fig. 4. If `inter_ref_pic_set_prediction_flag` is equal to zero, then the value-

by-value method of transmitting RPSs is used, otherwise inter-prediction is used for signaling RPSs.

Summary

5

It is an object of the invention to provide an improved alternative to the above techniques and prior art.

More specifically, it is an object of the invention to provide an improved way of signaling RPSs.

10 These and other objects of the invention are achieved by means of different aspects of the invention, as defined by the independent claims. Embodiments of the invention are characterized by the dependent claims.

 According to a first aspect of the invention, a method of encoding a video sequence is provided. The video sequence comprises RPSs, in
15 particular short-term RPSs. The method comprises arranging the RPSs in transmission order in a data structure, determining whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and encoding information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS
20 transmission is used.

 According to a second aspect of the invention, a computer program is provided. The computer program comprises computer program code. The computer program code is adapted, if executed on a processor, to implement the method according to the first aspect of the invention.

25 According to a third aspect of the invention, a computer program product is provided. The computer program product comprises a computer readable storage medium. The computer readable storage medium has the computer program according to the second aspect of the invention embodied therein.

According to a fourth aspect of the invention, a method of decoding a video sequence is provided. The video sequence comprises RPSs, in particular short-term RPSs. The method comprises receiving a data structure comprising the RPSs arranged in transmission order, determining whether
5 explicit RPS transmission is used for an RPS of a current picture of the video sequence, and receiving information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

According to a fifth aspect of the invention, a computer program is
10 provided. The computer program comprises computer program code. The computer program code is adapted, if executed on a processor, to implement the method according to the fourth aspect of the invention.

According to a sixth aspect of the invention, a computer program product is provided. The computer program product comprises a computer
15 readable storage medium. The computer readable storage medium has the computer program according to the fifth aspect of the invention embodied therein.

According to a seventh aspect of the invention, an encoder for encoding a video sequence is provided. The video sequence comprises
20 RPSs, in particular short-term RPSs. The encoder comprises a processor and a memory. The memory comprises instructions executable by the processor, whereby the encoder is operative to arrange the RPSs in transmission order in a data structure, determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence,
25 and encode information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

According to an eighth aspect of the invention, a decoder for decoding a video sequence is provided. The video sequence comprises RPSs, in
30 particular short-term RPSs. The decoder comprises a processor and a

memory. The memory comprises instructions executable by the processor, whereby the decoder is operative to receive a data structure comprising the RPSs arranged in transmission order, determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and receive information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

The invention makes use of an understanding that coding and signaling of RPSs, in particular short-term RPSs, can be performed more efficiently by using inter-prediction of RPSs. To this end, the RPSs are arranged in transmission order in a data structure, such as an SPS. Only if explicit transmission is used for an RPS of a current picture, information indicating which of the RPSs comprised in the data structure is used for predicting the RPS of the current picture is signaled from the encoder to the decoder by encoding such information. In other words, the information which describes which RPS of the RPSs arranged in transmission order should be used for prediction is only signaled if it is required for decoding the video sequence. The information may be encoded into an RPS, in particular a short-term RPS, of the current picture. Embodiments of the invention are advantageous in that the RPS of the current picture can be predicted from RPSs signaled in an SPS, thereby reducing the number of bits required for signaling RPSs and, consequently, the bitrate of the encoded video sequence. In addition, the proposed way of encoding and signaling RPSs is less complex than what is known in the art.

Throughout this disclosure, it is to be understood that information, such as information elements, parameters, values, and the like, which is encoded and signaled, or transmitted, from an encoder to a decoder over a communications network. Such information is consequently received at the decoder and may be decoded and subsequently be used in the decoding process.

According to an embodiment of the invention, the method further comprises determining whether inter-prediction is used for the RPS of the current picture, and wherein the information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture is only encoded if explicit RPS transmission and inter-prediction are used. That is, information specifying which RPS comprised in the data structure is used for inter-prediction is signaled to the decoder only if inter-prediction is used for an explicitly transmitted RPS. In the event that an explicitly transmitted RPS is encoded value-by-value, i.e., not using inter-prediction, that information is not encoded and signaled. Whether or not inter-prediction is used may, e.g., be signaled from the encoder to the decoder by means of a parameter. This is advantageous in that all alternatives for signaling RPSs are taken into consideration, and the number of bits required for signaling RPSs is reduced.

According to an embodiment of the invention, the encoding the data structure comprises predicting at least one RPS from an RPS which is directly preceding the at least one RPS in transmission order. In particular, this applies if explicit RPS transmission is not used. Predicting an RPS comprised in the data structure from another RPS comprised in the data structure, such as an RPS preceding the RPS in transmission order, is advantageous in that the number of bits required for signaling RPSs is reduced. It will be appreciated that inter-prediction cannot be used for the first RPS in transmission order. As an alternative to using inter-prediction, RPSs may be encoded value-by-value.

According to an embodiment of the invention, the method further comprises assigning an index to each RPS comprised in the data structure, and wherein the information indicating an RPS to be used for predicting the RPS of the current picture specifies the difference between an index of the RPS of the current picture and an index of the RPS used for prediction. This way of coding the information indicating which RPS to use for inter-prediction

is advantageous in that the number of bits required for conveying that information is reduced, in particular if Variable Length Coding (VLC) is employed. For instance, the information may be encoded using the HEVC parameter `delta_idx_minus1` which is part of the short-term RPS syntax
5 shown in Fig. 4.

According to an embodiment of the invention, the data structure is an SPS.

According to an embodiment of the invention, the determining whether explicit RPS transmission is used comprises determining whether an index of
10 the RPS of the current picture is equal to a number of RPSs comprised in the SPS. This is an efficient way of determining whether explicit RPS transmission is used, since the number of RPSs, in particular short-term RPSs, comprised in the SPS is signaled in the SPS and therefore known at both encoder and decoder.

According to an embodiment of the invention, the determining whether explicit RPS transmission is used comprises determining whether the RPS of the current picture is encoded in a slice header of the current picture. This is an alternative way of determining whether explicit RPS transmission is used. To this end, the slice header is inspected to check whether an RPS is
20 encoded into the slice header.

Even though advantages of the invention have in some cases been described with reference to embodiments of the first or fourth aspects of the invention, corresponding reasoning applies to embodiments of other aspects of the invention.

Further objectives of, features of, and advantages with, the invention will become apparent when studying the following detailed disclosure, the drawings and the appended claims. Those skilled in the art realize that different features of the invention can be combined to create embodiments other than those described in the following.

Brief description of the drawings

The above, as well as additional objects, features and advantages of the invention, will be better understood through the following illustrative and non-limiting detailed description of embodiments of the invention, with
5 reference to the appended drawings, in which:

Fig. 1 is an example of RPSs.

Fig. 2 illustrates coding the example RPSs in an SPS.

Fig. 3 illustrates coding the example RPSs in a slice header.

10 Fig. 4 shows the known syntax of a short-term RPS.

Fig. 5 shows an example sequence of pictures in strict transmission order, in accordance with an embodiment of the invention.

Fig. 6 shows the example sequence of pictures in less strict transmission order, in accordance with an embodiment of the invention.

15 Fig. 7 illustrates coding the example RPSs in an SPS in transmission order, in accordance with an embodiment of the invention.

Fig. 8 shows the proposed short-term RPS syntax, in accordance with an embodiment of the invention.

20 Fig. 9 shows a bitstream, in accordance with an embodiment of the invention.

Fig. 10 shows a method of encoding RPSs, in accordance with an embodiment of the invention.

Fig. 11 shows a method of decoding RPSs, in accordance with an embodiment of the invention.

25 Fig. 12 shows an encoder and a decoder, in accordance with embodiments of the invention.

All the figures are schematic, not necessarily to scale, and generally only show parts which are necessary in order to elucidate the invention, wherein other parts may be omitted or merely suggested.

Detailed description

The invention will now be described more fully herein after with reference to the accompanying drawings, in which certain embodiments of the invention are shown. This invention may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein. Rather, these embodiments are provided by way of example so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those skilled in the art.

10 The proposed way of signaling RPSs is based on arranging the RPSs into a data structure, such as an SPS, in transmission order. Different types of transmission order exists, in particular strict transmission order, Group Of Picture (GOP)-wise transmission order, and general transmission order. In the following, strict transmission order is described with reference to Fig. 5, 15 which shows a sequence of 16 pictures.

 In strict transmission order, the RPSs are encoded in exactly the same order as they are be used, i.e., in the order shown in Fig. 5, which is the order in which the images are decoded, i.e., decoding order, until there are no more RPSs to send (the picture with POC=17 will reuse the RPS in 20 row 12, POC=18 will reuse the RPS in row 13, POC=19 will reuse the RPS in row 14, POC=20 will reuse the RPS in row 15, POC=21 will reuse the RPS in row 12, and so forth). In this case we are guaranteed that the prediction from the preceding row, i.e., RPS, will work.

However, this is a sufficient, but not a necessary condition. There are other, less strict, transmission orders which also allow prediction from the preceding RPS row. One way is to permute the GOPs. Since the pictures available for prediction are mainly determined by the position within a GOP, it is sometimes possible to predict the first RPS of a certain GOP from the last RPS of a preceding GOP. For instance, the order illustrated in Fig. 6 would be possible. In Fig. 6, the second GOP group in the SPS has a lower GOP number than the preceding (first) GOP group, and the row with index 4 (GOP=0) can therefore predict from the row with index 3.

10 An alternative definition of transmission in transmission order is the following: An RPS A is said to be transmitted in transmission order relative an RPS B if RPS B is transmitted next after A and there exist a picture X that uses RPS A and a picture Y that uses RPS B, and picture Y is the picture that follows picture X in decoding order.

15 A similar definition, but of storage in transmission order is the following: An RPS A is said to be stored in transmission order relative an RPS B if the index of RPS B is equal to the index of RPS A plus one, and there exists a picture X that uses RPS A and a picture Y that uses RPS B, and picture Y is the picture that follows picture X in decoding order.

20 Further, the process of encoding and signaling RPSs is dependent on if the RPS is sent explicitly or implicitly. If the RPSs are sent once per sequence or once per second in a data structure such as an SPS, then each slice, which in most cases refers to a frame, can refer to an RPS row, i.e., an RPS index, in the SPS. This is referred to as implicit RPS transmission.

25 An alternative is to transmit the RPSs directly in the slice header. This is referred to as explicit RPS transmission and can be used in situations when the decoder cannot use an RPS in the SPS, or, wants to use an RPS which is not comprised in the active SPS.

30 As is described hereinbefore, there are two alternatives for encoding an RPS if explicit RPS transmission is used, encoding using inter-prediction

or the value-by-value approach. If inter-prediction is used, it is important to specify which RPS, or row in the list of RPSs in the data structure, to predict from. Since the encoder can choose to explicitly encode the RPS for any frame, the decoder cannot know which row is more likely. Therefore, if inter-
5 prediction is used for explicit RPS transmission, information identifying the one of the RPS comprised in the data structure, i.e., the SPS, is required. That information may be signaled in the parameter `delta_idx_minus1`.

However, a non-obvious observation is that this is not the case for encoding RPSs in an SPS. Under the condition that RPSs are sent in
10 transmission order, for each RPS, apart from the first RPS in the SPS, an RPS directly preceding the RPS must include all images which can be used for the RPS, except for the previous frame itself. Consequently, when encoding RPSs in an SPS, the variable `delta_idx_minus1` is equal to zero and does not need to be transmitted.

15 To elucidate this observation further, the RPSs of Fig. 2 are again shown in Fig. 7, albeit with the POC numbers written out, as well as the delta POC, i.e., the difference between the previous and the current POC number.

With reference to Fig. 7, the encoder would first have to send the first picture with $\text{POC}=0$, an intra-picture which cannot predict from any previous
20 picture. The second picture, i.e., the first row in Fig. 7, would have $\text{POC}=8$ and $\text{deltaPOC}=0-8 = -8$. Since there is no other picture transmitted than $\text{POC}=0$, the only picture that $\text{POC}=8$ can refer to is $\text{POC}=0$. Hence, -8 is the only possible value in the RPS.

Now, considering the second transmitted picture with $\text{POC}=4$. The
25 only two pictures which are available are the previous image $\text{POC}=8$, represented by $\text{deltaPOC}=4$, and the first image with $\text{POC}=0$. If it is not the previous picture, it must be available in the preceding RPS, otherwise the decoder would have discarded it. The second picture must therefore have a POC value which is derivable from the preceding RPS, which only contained

-8. Adding deltaPOC=4 to that value yields -4, which is the second value in the RPS.

Hence, it should be clear that if the RPSs are stored in transmission order, then all the RPSs must either a) be reconstructable, or predictable, from the preceding RPS adding deltaPOC to the values in the preceding RPS, or b) it must be deltaPOC itself.

If this is not the case, i.e., if a value is neither a) nor b), the value is not predictable from any value in the preceding RPS and the decoder has discarded the picture and it cannot be referenced. The only exceptions are the preceding picture, which is not in the preceding RPS since the picture does not predict from itself, and that is exactly b).

Based on this reasoning it should be clear that, if RPSs are arranged and encoded in transmission order in an SPS, the parameter delta_idx_minus1 may be omitted, saving both complexity and bits. However, when transmitting RPSs explicitly, access to any RPS in the SPS is desirable in order to use that RPS for inter-prediction. In this case, the RPS to predict from is signaled using delta_idx_minus1. Therefore, it is proposed to check whether explicit RPS transmission is used in connection with inter-prediction or not. If explicit RPS transmission is used, and inter-prediction is performed, delta_idx_minus1 will be transmitted, otherwise it is omitted.

It is noted that the issue of whether to signal delta_idx_minus1 or not arises only if inter-prediction is utilized, as can be seen from the short-term RPS syntax shown in Fig. 4. The if-statement in row 3,

if(inter_ref_pic_set_prediction_flag)

checks whether inter-prediction is used or not. To this end, delta_idx_minus1 is signaled only if the if-statement evaluates true, i.e., only if inter-prediction is used.

The proposed way of encoding and signaling RPSs, in particular with respect to explicit or implicit transmission, is reflected in the proposed short-term RPS syntax shown in Fig. 8.

In the short-term RPS syntax shown in Fig. 8, a new if-statement is
5 introduced:

```
if( idx == num_short_term_ref_pic_sets )
```

If this statement evaluates to “true”, explicit RPS transmission is used
10 for the current picture, and the decoder needs to decode delta_idx_minus1. Otherwise, if the statement evaluated to false, the RPS is one of the RPSs signaled in the SPS, and delta_idx_minus1 is inferred to be equal to 0, i.e., inter-prediction from the directly preceding RPS is performed. In
15 correspondence with the known short-term RPS shown in Fig. 4, the issue of whether to signal delta_idx_minus1 or not arises only if inter-prediction is used. This can be seen from the if-statement in row 3 of the proposed short-term RPS shown in Fig. 8, which is identical to the if-statement of the known short-term RPS shown in Fig. 4.

More specifically, processing in accordance with the syntax shown in
20 Fig. 8 is performed both when the SPS is created, e.g., once per sequence or once per second, and, optionally for a slice header of the current picture in the event that the encoder employs explicit RPS transmission. In the first case, the value of idx will vary from 0 to the value of
num_short_term_ref_pic_sets - 1, i.e., the number of short-term RPSs
25 signaled in the SPS, minus one. In the second case, when explicit RPS transmission is used, the value of idx will be equal to
num_short_term_ref_pic_set. Thus, one can determine whether explicit RPS transmission is used by comparing idx to num_short_term_ref_pic_set, as is done by the if-statement.

As an alternative, one may envisage embodiments of the invention which are based on another parameter `other_parameter` which is equal to 0 when called during the SPS creation, i.e., the first case described above, and equal to 1 when called during creation of a slice header. In such case, the
5 corresponding if-statement reads

`if( other_parameter ).`

As yet a further alternative, one function for creating the short-term
10 RPS may be executed when the SPS is created, and another function when the slice header is created. To this end, the first function would not transmit/receive the parameter `delta_idx_minus1`, but the second function would transmit/receive the parameter `delta_idx_minus1`.

At first glance, the proposed short-term RPS syntax shown in Fig. 8
15 appears to have increased complexity as compared to the current standard shown in Fig. 4, partly due to extra if-statement on row 4. However, it should be noted that decoding the value `delta_idx_minus1` is more complex than just evaluating the if-statement. Also, since non-explicit RPS transmission is used for most of the pictures, the additional if-statement is rarely executed.

20 In order to elucidate the invention even further, an illustrative example of an embodiment of the invention is now described with reference to Fig. 9, which depicts a bit stream 900 comprising an SPS 901 and three slices, "slice 2" 902, "slice 3" 903, and "slice 4" 904. The bit stream 900 may comprise additional elements which are omitted in Fig. 9.

25 The SPS 901 comprises, among other data, five RPSs 905. An RPS can be transmitted in an SPS by means of value-by-value coding, which in many cases is the most efficient way if the number of values in the RPS is small. For instance, the first RPS in the list of RPSs 905 (`{-8}`) can be sent value-by-value as:

30

```

inter_ref_set_prediction_flag = 0
num_negative_pics = 1
num_positive_pics = 0
delta_poc_s0_minus1[0] = 8
5   used_by_cur_flag[0] = 1

```

As an alternative, an RPS comprises in an SPS can also be transmitted using inter-prediction, e.g., the second RPS in the list of RPSs 905 ({-4, 4}). Since the RPS is signaled in the SPS 901, i.e., not explicitly, no information describing which RPS to predict from needs to be signaled, since inter-prediction is performed from the previous row, which in the present example comprises only a single value {-8}. Hence, the second RPS is transmitted as:

```

15   inter_ref_set_prediction_flag = 1
      delta_rps_sign = 0
      abs_delta_rps_minus1 = 3

```

The decoder will be able to reconstruct the second RPS as follows.

20 First, $\text{deltaRps} = (1 - 2 * \text{delta_rps_sign}) * (\text{abs_delta_rps_minus1} + 1) = 4$ is calculated and added to the value {-8} (first RPS), resulting in -4, which is the first value of the second RPS. The second value of the second RPS is the delta_rps-value itself (delta_rps = 4). Thus, the second RPS is reconstructed as {-4, 4}. It is emphasized here that delta_idx_minus1 is not transmitted for the second RPS since it is inferred to be equal to zero.

Correspondingly, the remaining RPSs in the list of RPSs 905 in the SPS 901 can be sent either value-by-value or using inter-prediction, as is described hereinbefore. For instance, the third RPS {-2, -4, -6, 2} may be signaled using inter-prediction whereas the fourth RPS {-1, 1, 3, 5, 7} may be signaled value-by-value.

Further with reference to the bitstream 900 illustrated in Fig. 9, slice 2 902 illustrates implicit transmission of an RPS. To this end, slice 2 902 comprises information indicating 909 which RPS in the list 905 of RPSs to use, such as an RPS index “#1”. Thus, for slice 2 902 the encoder transmits:

5

```
short_term_ref_pic_set_sps_flag = 1
```

to indicate that an RPS from the list 905 of RPSs in the SPS 901 should be used, and then transmits

10

```
short_term_ref_pic_set_idx = 1
```

to indicate that the second RPS in the list 905, having index “#1”, should be used.

15

As a further example, it is assumed that slice 3 903, which is transmitted in the bitstream 900, cannot use any RPS from the list 905 of RPSs signaled in the SPS 901, but instead signals an RPS 907 explicitly. There are two ways of signaling explicitly, and in the present example slice 3 903 uses value-by-value coding. Thus, the encoder transmits:

20

```
inter_ref_set_prediction_flag = 0
```

```
num_negative_pics = 1
```

```
num_positive_pics = 2
```

```
delta_poc_s0_minus1[0] = 1
```

25

```
used_by_cur_flag[0] = 1
```

```
delta_poc_s1_minus1[0] = 1
```

```
used_by_cur_flag[0] = 1
```

```
delta_poc_s1_minus1[1] = 3
```

```
used_by_cur_flag[1] = 1
```

30

to signal the RPS 907 ({-2, 2, 6}). The number of negative pics in this set is one (-2), and the number positive ones is two (2 and 6). The negative value is coded relative to the zero ($\text{delta_poc_s0}[0] = 2$), and the first positive value is coded likewise. Finally the difference between the two positive values (6-2=4) is encoded ($\text{delta_poc_s1}[1] = 4$).

The other way of encoding explicitly is illustrated with reference to slice 4 904 of the bitstream 900. Slice 4 904 uses inter-prediction, but since the RPS it is transmitted explicitly, information 908 indicating 910 from which RPS in the list 905 of RPSs to predict from must be provided. In the present example, this is indicated by RPS index "#2" 908. Thus, the following is transmitted in slice 4 904:

```

inter_ref_set_prediction_flag = 1
delta_idx_minus1 = 2
15  num_negative_pics = 1
    num_positive_pics = 0
    delta_poc_s0_minus1[0] = 8
    used_by_cur_flag[0] = 1

```

20 The decoder can now calculate the RefRpsIdx as

$$\text{RefRpsIdx} = \text{stRpsIdx} - (\text{delta_idx_minus1} + 1).$$

In the present example, the list 905 of RPSs in the SPS 901 comprises five RPSs. Thus, stRpsIdx will be equal to five, and RefRpsIdx is evaluated to $5 - (2 + 1) = 2$, which is the index of the third RPS ({-2, -4, -6, 2}). In the case of slice 4 904, delta_idx_minus1 must be sent in order to indicate which RPS to predict from. In addition to the RPS index 908, information describing how to predict is signaled ("prediction data" 908).

In the following, the example shown in Fig. 9 is described for a decoder. The decoder first receives the SPS 901 comprising a list 905 of RPSs. The RPSs comprised in list 905 can either be received value-by-value or using inter- prediction. For instance, the first RPS ({-8}) is received value-
 5 by-value by receiving

```

inter_ref_set_prediction_flag = 0
num_negative_pics = 1
num_positive_pics = 0
10 delta_poc_s0_minus1[0] = 8
used_by_cur_flag[0] = 1

```

On the other hand, the second RPS ({-4, 4}) may be received using inter-prediction. However, since the RPS is comprised in the SPS 901, the
 15 RPS is not received explicitly, and delta_idx_minus1 is therefore not received. Hence, the following is received by the decoder for the second RPS:

```

inter_ref_set_prediction_flag = 1
20 delta_rps_sign = 0
abs_delta_rps_minus1 = 3

```

The decoder will be able to reconstruct $\text{deltaRps} = (1 - 2 * \text{delta_rps_sign}) * (\text{abs_delta_rps_minus1} + 1) = 4$. deltaRps is then
 25 added to the value {-8} (from the first RPS), resulting in -4, which is the first value of the second RPS. The second value of the second RPS is the delta_rps-value itself (4), and the RPS is therefore predicted as {-4, 4}. The other RPSs signaled in the SPS 901 are received value-by-value or using inter-prediction, as is described above.

For slice 2 902, the RPS is received implicitly by receiving an RPS index 906 (“#1”) which points 909 to the second RPS in the list 905 of RPSs in the SPS 901. Slice 3 903 transmits its RPS 907 explicitly, coded value-by-value.

- 5 The RPS for slice 4 904 is received explicitly using inter-prediction. Therefore, information 908 indicating an RPS in the list 905 of RPSs to be used for inter-prediction (RPS index “#2”) is received, in addition to information describing how to predict (“prediction data” 908). In this case, it is necessary to receive `delta_idx_minus1`, and the following information is
10 received by the decoder:

```
inter_ref_set_prediction_flag = 1
delta_idx_minus1 = 2
num_negative_pics = 1
15 num_positive_pics = 0
delta_poc_s0_minus1[0] = 8
used_by_cur_flag[0] = 1
```

- It is emphasized here that, in order to receive the RPS of slice 4 904
20 explicitly, using inter-prediction, it is necessary to receive `delta_idx_minus1`.

- In Fig. 10 an embodiment of the method of encoding a video sequence comprising RPSs is shown. The method 1000 starts with arranging 1001 the RPSs in transmission order in a data structure, such as an SPS, and sending 1001 the data structure. Subsequently, it is
25 determined 1002 whether inter-prediction is used. This may, e.g., be achieved by using `inter_ref_set_prediction_flag`, as is described hereinbefore and as is known in the art. If inter-prediction is used (“YES” in Fig. 10), it is determined 1002 whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, i.e., whether the RPS is signaled in a
30 slice header of the current picture. If explicit transmission is used, i.e., if the

RPS is signaled in a slice header of the current picture, information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as `delta_idx_minus1`, is encoded and sent 1004.

The information is only transmitted 1004 if explicit RPS transmission is used, otherwise it is not transmitted 1005. Likewise, if inter-prediction is not used 1002 ("NO" in Fig. 10), information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as `delta_idx_minus1`, is not signaled. It will be appreciated that embodiments of the invention are not limited to the exact order of steps shown in Fig. 10.

For instance, embodiments of the invention may be envisaged which first determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, i.e., whether the RPS is signaled in a slice header of the current picture, and subsequently determine whether inter-prediction is used.

In Fig. 11 an embodiment of the method of decoding a video sequence comprising RPSs is shown. The method 1100 comprises receiving 1101 a data structure comprising the RPSs arranged in transmission order and determining 1102 whether inter-prediction is used. This may, e.g., be achieved by inspecting `inter_ref_set_prediction_flag`, as is described hereinbefore. If inter-prediction is used ("YES" in Fig. 11), it is determined 1103 whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, i.e., whether the RPS is signaled in a slice header of the current picture. If explicit transmission is used, i.e., if the RPS is signaled in a slice header of the current picture, information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as `delta_idx_minus1`, is received 1104. The information is only received 1104 if explicit RPS transmission is used, otherwise it is not received 1105. If the information is `delta_idx_minus1`, specifying the difference between an index of the RPS of the current picture and an index of the RPS used for prediction, `delta_idx_minus1` is instead

interpreted 1105 to be equal to zero, i.e., the preceding RPS is used for inter-prediction. If inter-prediction is not used 1102 ("NO" in Fig. 10), information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as delta_idx_minus1, is not received. It
5 will be appreciated that embodiments of the invention are not limited to the exact order of steps shown in Fig. 11. For instance, embodiments of the invention may be envisaged which first determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, i.e., whether the RPS is signaled in a slice header of the current picture, and
10 subsequently determine whether inter-prediction is used.

In Fig. 12 embodiments of an encoder for encoding a video sequence comprising RPSs and embodiments of a decoder for decoding a video sequence comprising RPSs are shown.

The encoder 1200 comprises a processor 1201 and a memory 1202
15 comprising instructions 1203 executable by the processor 1201, whereby the encoder 1200 is operative to arrange the RPSs in transmission order in a data structure, determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and encode information 1220 indicating an RPS comprised in the data structure to be used for predicting
20 the RPS of the current picture, such as delta_idx_minus1, only if explicit RPS transmission is used.

The decoder 1210 comprises a processor 1211 and a memory 1212 comprising instructions 1213 executable by the processor 1211, whereby the decoder 1210 is operative to receive a data structure comprising the RPSs
25 arranged in transmission order, determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and receive information 1220 indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as delta_idx_minus1, only if explicit RPS transmission is used.

It should be noted that the embodiments described herein relate to video encoders and decoders, as well as bitstream parsers, transcoders, network nodes, and so forth. Embodiments of the encoder and the decoder can be implemented in devices such as video cameras, displays, tablets,
5 digital television receivers, network nodes, and so forth. Although it is shown in Fig. 12 that the parameter `delta_idx_minus1` is sent, it is to be understood that `delta_idx_minus1` is only sent only if explicit RPS transmission is used, in accordance with embodiments of the invention.

It will also be appreciated that the use of `delta_idx_minus1` throughout
10 this document it is only an example for how to indicate which RPS should be used for inter-prediction, and it should be obvious the reader that other signaling schemes can be used. In this respect, any reference to `delta_idx_minus1` should be understood as an indicator of which RPS to predict from.

15 It should further be noted that, although the embodiments herein describe RPSs to be part of an SPS, RPSs may also be included in other data structures, e.g., PPSs or any other parameter set.

As an alternative to the encoder 1200 described with reference to Fig. 12, an embodiment of the encoder may also comprise means being
20 configured for arranging the RPSs in transmission order in a data structure, determining whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and encoding information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as `delta_idx_minus1`, only if explicit RPS transmission is
25 used.

As yet a further alternative, an embodiment of the encoder may also comprise means for arranging the RPSs in transmission order in a data structure, means for determining whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and means for
30 encoding information indicating an RPS comprised in the data structure to be

used for predicting the RPS of the current picture, such as delta_idx_minus1, only if explicit RPS transmission is used.

As an alternative to the decoder 1210 described with reference to Fig. 12, an embodiment of the decoder may also comprise means being
5 configured for receiving a data structure comprising the RPSs arranged in transmission order, determining whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and receiving information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as delta_idx_minus1, only if
10 explicit RPS transmission is used.

As yet a further alternative, an embodiment of the decoder may also comprise means for receiving a data structure comprising the RPSs arranged in transmission order, means for determining whether explicit RPS
transmission is used for an RPS of a current picture of the video sequence,
15 and means for receiving information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture, such as delta_idx_minus1, only if explicit RPS transmission is used.

The person skilled in the art realizes that the invention by no means is limited to the embodiments described above. On the contrary, many
20 modifications and variations are possible within the scope of the appended claims.

CLAIMS

1. A method (1000) of encoding a video sequence comprising Reference Picture Sets, RPSs, the method comprising:
 - 5 arranging (1001) the RPSs in transmission order in a data structure, determining (1003) whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and encoding (1004) information (1220) indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture
 - 10 only if explicit RPS transmission is used.
2. The method according to claim 1, further comprising determining (1002) whether inter-prediction is used for the RPS of the current picture, and wherein the information indicating an RPS comprised in the data
15 structure to be used for predicting the RPS of the current picture is only encoded (1004) if explicit RPS transmission and inter-prediction are used.
3. The method according to claims 1 or 2, wherein at least one RPS of the RPSs arranged in the data structure is predicted from an RPS which is
20 directly preceding the at least one RPS in transmission order.
4. The method according to any one of claims 1 to 3, further comprising assigning an index to each RPS comprised in the data structure, and wherein the information indicating an RPS to be used for predicting the
25 RPS of the current picture specifies the difference between an index of the RPS of the current picture and an index of the RPS used for prediction.
5. The method according to any one of claims 1 to 4, wherein the data structure is a Sequence Parameter Set, SPS.

6. The method according to claim 5, wherein the determining (1003) whether explicit RPS transmission is used comprises determining whether an index of the RPS of the current picture is equal to a number of RPSs comprised in the SPS.

5

7. The method according to any one of claims 1 to 5, wherein the determining (1003) whether explicit RPS transmission is used comprises determining whether the RPS of the current picture is encoded in a slice header of the current picture.

10

8. A computer program (1203) comprising computer program code, the computer program code being adapted, if executed on a processor (1201), to implement the method according to any one of the claims 1 to 7.

15

9. A computer program product comprising a computer readable storage medium (1202), the computer readable storage medium having the computer program (1203) according to claim 8 embodied therein.

20

10. A method (1100) of decoding a video sequence comprising Reference Picture Sets, RPSs, the method comprising:

receiving (1101) a data structure comprising the RPSs arranged in transmission order,

25

determining (1103) whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and

receiving (1104) information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

11. The method according to claim 10, further comprising determining (1102) whether inter-prediction is used for the RPS of the current picture, and wherein the information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture is only
5 received (1104) if explicit RPS transmission and inter-prediction are used.

12. The method according to claims 10 or 11, further comprising predicting at least one RPS from an RPS which is directly preceding the RPS in transmission order.

10

13. The method according to any one of claims 10 to 12, wherein an index is assigned to each RPS comprised in the data structure, and wherein the information indicating an RPS to be used for predicting the RPS of the current picture specifies the difference between an index of the RPS of the
15 current picture and an index of the RPS used for prediction.

14. The method according to any one of claims 10 to 13, wherein the data structure is a Sequence Parameter Set, SPS.

20 15. The method according to claim 14, wherein the determining (1103) whether explicit RPS transmission is used comprises determining whether an index of the RPS of the current picture is equal to a number of RPSs comprised in the SPS

25 16. The method according to any one of claims 10 to 14, wherein the determining (1103) whether explicit RPS transmission is used comprises determining whether the RPS of the current picture is encoded in a slice header of the current picture.

17. A computer program (1213) comprising computer program code, the computer program code being adapted, if executed on a processor (1211), to implement the method according to any one of the claims 10 to 16.

5

18. A computer program product comprising a computer readable storage medium (1212), the computer readable storage medium having the computer program (1213) according to claim 17 embodied therein.

10 19. An encoder (1200) for encoding a video sequence comprising Reference Picture Sets, RPSs, the encoder comprising a processor (1201) and a memory (1202) comprising instructions executable by the processor, whereby the encoder is operative to:

arrange the RPSs in transmission order in a data structure,

15 determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and

encode information (1220) indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

20

20. The encoder according to claim 19, being further operative to determine whether inter-prediction is used for the RPS of the current picture, and to encode the information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if
25 explicit RPS transmission and inter-prediction are used.

21. The encoder according to claims 19 or 20, wherein at least one RPS of the RPSs arranged in the data structure is predicted from an RPS which is directly preceding the at least one RPS in transmission order.

30

22. The encoder according to any one of claims 19 to 21, being further operative to assign an index to each RPS comprised in the data structure, and wherein the information indicating an RPS to be used for predicting the RPS of the current picture specifies the difference between an index of the RPS of the current picture and an index of the RPS used for prediction.

23. The encoder according to any one of claims 19 to 22, wherein the data structure is a Sequence Parameter Set, SPS.

24. The encoder according to claim 23, being further operative to determine whether explicit RPS transmission is used by determining whether an index of the RPS of the current picture is equal to a number of RPSs comprised in the SPS.

25. The encoder according to any one of claims 19 to 24, being further operative to determine whether explicit RPS transmission is used by determining whether the RPS of the current picture is encoded in a slice header of the current picture.

26. A decoder (1210) for decoding a video sequence comprising Reference Picture Sets, RPSs, the decoder comprising a processor (1211) and a memory (1212) comprising instructions executable by the processor, whereby the decoder is operative to:

receive a data structure comprising the RPSs arranged in transmission order,
determine whether explicit RPS transmission is used for an RPS of a current picture of the video sequence, and
receive information (1220) indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission is used.

27. The decoder according to claim 26, being further operative to determine whether prediction is used for the RPS of the current picture, and to receive the information indicating an RPS comprised in the data structure to be used for predicting the RPS of the current picture only if explicit RPS transmission and prediction are used.

28. The decoder according to claims 26 or 27, being further operative to predict at least one RPS from an RPS which is directly preceding the RPS in transmission order.

29. The decoder according to any one of claims 26 to 28, wherein an index is assigned to each RPS comprised in the data structure, and wherein the information indicating an RPS to be used for predicting the RPS of the current picture specifies the difference between an index of the RPS of the current picture and an index of the RPS used for prediction.

30. The decoder according to any one of claims 26 to 29, wherein the data structure is a Sequence Parameter Set, SPS.

31. The decoder according to claim 30, being further operative to determine whether explicit RPS transmission is used by determining whether an index of the RPS of the current picture is equal to a number of RPSs comprised in the SPS

32. The decoder according to any one of claims 26 to 30, being further operative to determine whether explicit RPS transmission is used by determining whether the RPS of the current picture is encoded in a slice header of the current picture.