Claims

1. A method for decoding a video, the method comprising:

decoding, from a bitstream of the video, an additional bit count M indicating a quantity of additional general constraints information (GCI) bits included in the bitstream of the video, wherein the additional bits comprise flag bits indicating respective additional coding tools to be constrained for the video, and wherein an expected value of the additional bit count is 0, 6, or greater than 6;

in response to determining that the decoded additional bit count M is greater than 6, decoding M-6 bits that follow six flag bits in the bitstream; and

decoding a remaining portion of the bitstream of the video into images independent of the decoded M-6 bits and based, at least in part, upon constraints specified for the respective additional coding tools by the six flag bits.

2. The method of claim 1, further comprising:

decoding, from the bitstream of the video and prior to decoding the M-6 bits, the six flag bits representing six respective flags indicating six respective additional coding tools to be constrained for the video; and

decoding a remaining portion of the bitstream of the video into images based, at least in part, upon the constraints for the six additional coding tools indicated by the six flags.

3. The method of claim 2, wherein the six flags comprise:

a first flag indicating a restriction on pictures of the video to be either intra random access point (IRAP) pictures or gradual decoder refresh (GDR) pictures;

a second flag indicating whether an extended transform precision is constrained;

a third flag indicating whether an explicit Rice parameter signaling is constrained;

a fourth flag indicating an alternative Rice parameter derivation for binarization of quantization residuals of the video;

a fifth flag indicating whether to initialize Rice parameter derivation for binarization based on previous transform units; and

a sixth flag indicating whether to impose a constraint on pictures in an in-scope output layer set OlsInScope when decoding a position of a last non-zero level in a transform unit.

4. The method of claim 3, wherein decoding the remaining portion of the bitstream of the video into images based, at least in part, upon the constraints for the six additional coding tools indicated by the six flags comprises one or more of:

determining all pictures in one or more output layer sets are GDR pictures with ph_recovery_poc_cnt equal to 0 or IRAP pictures based on the first flag being one, and decoding the GDR pictures or IRAP pictures in the one or more output layer sets;

determining that an extended transform precision is constrained based on the second flag being one and decoding the remaining portion of the bitstream of the video by setting sps_extended_precision_flag for pictures in the in-scope output layer set OlsInScope be equal to 0 so that extended dynamic range is not used;

determining that an explicit Rice parameter signaling is constrained based on the third flag being one, and decoding the remaining portion of the bitstream of the video by disabling alternative Rice parameter signaling for pictures in the in-scope output layer set OlsInScope;

determining that an alternative Rice parameter derivation for binarization of quantization residuals of the video is constrained based on the fourth flag being one, and decoding the remaining portion of the bitstream of the video by disabling alternative Rice parameter signaling for pictures in the in-scope output layer set OlsInScope;

determining that an initialization of Rice parameter derivation for binarization based on previous transform unit state is constrained based on determining the fifth flag being one, and decoding the remaining portion of the bitstream of the video without initializing Rice parameter based on previous transform unit state for pictures in the in-scope output layer set OlsInScope; or

determining that coordinates of a last significant coefficient are coded relative to a top-left corner for each transform block of a slice based on the sixth flag being one and decoding the remaining portion of the bitstream of the video by interpreting decoded coordinates of the last significant coefficient as being relative to the top-left corner for each transform block of the slice.

5. The method of claim 3, further comprising one or more of:

determining that the first flag is not present in the bitstream and inferring a value of the first flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the second flag is not present in the bitstream and inferring a value of the second flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the third flag is not present in the bitstream and inferring a value of the third flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the fourth flag is not present in the bitstream and inferring a value of the fourth flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the fifth flag is not present in the bitstream and inferring a value of the fifth flag to be 0 indicating no constraint is imposed for a corresponding coding tool; or

determining that the sixth flag is not present in the bitstream and inferring a value of the sixth flag to be 0 indicating no constraint is imposed for a corresponding coding tool.

6. The method of claim 1, further comprising, prior to decoding the additional bit count M:

decoding a general constraints information (GCI) flag from the bitstream of the video; and

determining that one or more general constraints are imposed for the video based on a value of the GCI flag.

7. The method of claim 6, wherein the GCI flag is decoded from a network packet of the video, a video parameter set of the video, or a sequence parameter set of the video.

8. A method for encoding a video, the method comprising:

determining an additional bit count M indicating a quantity of additional general constraints information (GCI) bits, wherein the additional bits comprise flag bits indicating respective additional coding tools to be constrained for the video, and wherein an expected value of the additional bit count is 0, 6, or greater than 6;

encoding the additional bit count M into a bitstream of the video; and

in response to the additional bit count M being greater than 6, encoding M-6 bits that follow six flag bits into the bitstream of the video,

wherein a remaining portion of the bitstream of the video is based, at least in part, upon constraints specified for the respective additional coding tools by the six flag bits.

9. The method of claim 8, further comprising:

determining six flag bits representing six respective flags indicating six respective additional coding tools to be constrained for the video; and

encoding a remaining portion of the bitstream of the video based, at least in part, upon the constraints for the six additional coding tools indicated by the six flags,

wherein the six flags comprise:

a first flag indicating a restriction on pictures of the video to be either intra random access point (IRAP) pictures or gradual decoder refresh (GDR) pictures;

a second flag indicating whether an extended transform precision is constrained;

a third flag indicating whether an explicit Rice parameter signaling is constrained;

a fourth flag indicating an alternative Rice parameter derivation for binarization of quantization residuals of the video;

a fifth flag indicating whether to initialize Rice parameter derivation for binarization based on previous transform units; and

a sixth flag indicating whether to impose a constraint on pictures in an in-scope output layer set OlsInScope when determining a position of a last non-zero level in a transform unit.

10. The method of claim 9, wherein encoding the remaining portion of the bitstream of the video based, at least in part, upon the constraints for the six additional coding tools indicated by the six flags comprises one or more of:

determining all pictures in one or more output layer sets are GDR pictures with ph_recovery_poc_cnt equal to 0 or IRAP pictures based on the first flag being one, and encoding the GDR pictures or IRAP pictures in the one or more output layer sets;

determining that an extended transform precision is constrained based on the second flag being one and encoding the remaining portion of the bitstream of the video by setting sps_extended_precision_flag for pictures in the in-scope output layer set OlsInScope be equal to 0 so that extended dynamic range is not used;

determining that an explicit Rice parameter signaling is constrained based on the third flag being one, and encoding the remaining portion of the bitstream of the video by disabling alternative Rice parameter signaling for pictures in the in-scope output layer set OlsInScope;

determining that an alternative Rice parameter derivation for binarization of quantization residuals of the video is constrained based on the fourth flag being one, and encoding the remaining portion of the bitstream of the video by disabling alternative Rice parameter signaling for pictures in the in-scope output layer set OlsInScope;

determining that an initialization of Rice parameter derivation for binarization based on previous transform unit state is constrained based on determining the fifth flag being one, and encoding the remaining portion of the bitstream of the video without initializing Rice parameter based on previous transform unit state for pictures in the in-scope output layer set OlsInScope; or

determining that coordinates of a last significant coefficient are coded relative to a top-left corner for each transform block of a slice based on the sixth flag being one and encoding the remaining portion of the bitstream of the video by interpreting coordinates of the last significant coefficient as being relative to the top-left corner for each transform block of the slice,

11. The method of claim 9, wherein the method further comprises:

determining that the first flag is not present in the bitstream and inferring a value of the first flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the second flag is not present in the bitstream and inferring a value of the second flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the third flag is not present in the bitstream and inferring a value of the third flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the fourth flag is not present in the bitstream and inferring a value of the fourth flag to be 0 indicating no constraint is imposed for a corresponding coding tool;

determining that the fifth flag is not present in the bitstream and inferring a value of the fifth flag to be 0 indicating no constraint is imposed for a corresponding coding tool; or

determining that the sixth flag is not present in the bitstream and inferring a value of the sixth flag to be 0 indicating no constraint is imposed for a corresponding coding tool.

12. A computer-readable storage medium having stored thereon a computer program and a bitstream, wherein when the computer program is executed by a processor, the computer program implements the steps of the method of any one of claims 9 to 11 to generate the bitstream.