TITLE

METHOD AND APPARATUS FOR CHROMINANCE QUANTIZATION PARAMETERS SIGNALLING

CROSS-REFERENCE TO RELATED APPLICATIONS

This patent application claims priority of international patent application PCT/RU2019/000664 filed on 23.09.2019. ~~The disclosure of the aforementioned patent application is hereby incorporated by reference in its entirety.~~

TECHNICAL FIELD

Embodiments of the present application (disclosure) generally relate to the field of picture processing and more particularly to Method and apparatus for Chrominance quantization parameters signaling.

BACKGROUND

Video coding (video encoding and decoding) is used in a wide range of digital video applications, for example broadcast digital TV, video transmission over internet and mobile networks, real-time conversational applications such as video chat, video conferencing, DVD and Blu-ray discs, video content acquisition and editing systems, and camcorders of security applications.

The amount of video data needed to depict even a relatively short video can be substantial, which may result in difficulties when the data is to be streamed or otherwise communicated across a communications network with limited bandwidth capacity. Thus, video data is generally compressed before being communicated across modern day telecommunications networks. The size of a video could also be an issue when the video is stored on a storage device because memory resources may be limited. Video compression devices often use software and/or hardware at the source to code the video data prior to transmission or storage, thereby decreasing the quantity of data needed to represent digital video images. The compressed data is then received at the destination by a video decompression device that decodes the video data. With limited network resources and ever-increasing demands of higher video quality, improved compression and decompression techniques that improve compression ratio with little to no sacrifice in picture quality are desirable.

CLAIMS

1. A method for inverse quantization of a current block of a picture, wherein the method is performed by a decoder, and the method comprising:

receiving a bitstream;

obtaining a joint chrominance component residual, JCCR, control flag from the bitstream;

obtaining a chrominance mapping information from the bitstream based on the JCCR control flag;

obtaining at least one chrominance quantization parameter, QP, offset from the bitstream based on the JCCR control flag;

obtaining a QP value for the current chrominance block based on the obtained chrominance mapping information and the at least one obtained chrominance QP offset;

performing inverse quantization on the current chrominance block by using the determined QP value.

2. The method according to claim 1, wherein the bitstream comprises an SPS level syntax, and the JCCR control flag is obtained from the SPS level syntax.

3. The method according to claim 1 or 2, wherein the JCCR control flag is the sps_joint_cbcr_enabled_flag.

4. The method according to claim 3, wherein if the value of the sps_joint_cbcr_enabled_flag is 1, the at least one obtained chrominance QP offset is specified by slice_joint_cbcr_qp_offset.

~~5. The method according to any one of claims 1 to 4, wherein the chrominance mapping information comprises delta_qp_in_val_minus1[i][j] and delta_qp_out_val[i][j], and the chrominance mapping information is obtained from an SPS level syntax comprised in the bitstream.~~

~~6~~5. The method according to any one of claims 2 to ~~5~~4, wherein the SPS level syntax comprises the following structure:

seq_parameter_set_rbsp( ) {	**Descriptor**
…	
same_qp_table_for_chroma	**u(1)**

for(i = 0; i < same_qp_table_for_chroma ? 1 : sps_joint_cbcr_enabled_flag ? 3 : 2; i++) {	
num_points_in_qp_table_minus1[i]	**ue(v)**
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {	
delta_qp_in_val_minus1[i][j]	**ue(v)**
…	**ue(v)**
}	
}	
}	
…	
}	

.

~~7~~6. The method according to any one of claims 1 to ~~6~~5, wherein the obtaining the at least one chrominance QP offset from the bitstream based on the JCCR control flag comprises:

obtaining, based on the JCCR control flag, the at least one chrominance QP offset from a picture parameter set, PPS, level syntax of the bitstream.

~~8. The method according to claim 7, wherein the PPS level syntax comprises the following structure:~~

~~pic_parameter_set_rbsp( ) {~~	~~Descriptor~~
~~…~~	
~~**pps_cb_qp_offset**~~	~~**se(v)**~~
~~**pps_cr_qp_offset**~~	~~**se(v)**~~
~~**if(sps_joint_cbcr_enabled_flag)**~~	
~~**pps_joint_cbcr_qp_offset**~~	~~**se(v)**~~
~~**cu_chroma_qp_offset_enabled_flag**~~	~~**u(1)**~~
~~if(pps_cu_chroma_qp_offset_list_enabled_flag) {~~	
~~**cu_chroma_qp_offset_subdiv**~~	~~**ue(v)**~~
~~**chroma_qp_offset_list_len_minus1**~~	~~**ue(v)**~~
~~for(i = 0; i <= chroma_qp_offset_list_len_minus1; i++) {~~	
~~**cb_qp_offset_list[i]**~~	~~**se(v)**~~
~~**cr_qp_offset_list[i]**~~	~~**se(v)**~~
~~**if(sps_joint_cbcr_enabled_flag)**~~	
~~**joint_cbcr_qp_offset_list[i]**~~	~~**se(v)**~~
~~}~~	
~~}~~	
~~…~~	
~~}~~	

~~9. A method for inverse quantization of a current block of a picture, wherein the method is performed by a decoder, and the method comprising:~~

~~receiving a bitstream, wherein the bitstream comprises a Slice Header syntax and a PPS syntax;~~

~~obtaining syntax elements from the PPS syntax, wherein the obtained syntax elements comprise at least one chrominance quantization parameter, QP, offset;~~

~~obtaining chrominance QP offset information from the Slice Header, wherein the QP offset information is obtained independently of any PPS syntax elements in the PPS syntax;~~

~~determining a QP value for the current chrominance block depending on the at least one chrominance QP offset obtained from the PPS syntax and the chrominance QP offset information obtained from the Slice Header syntax;~~

~~performing inverse quantization on the current chrominance block by using the determined QP value.~~

~~10~~7. A method for encoding a current block of a picture, wherein the method is performed by an encoder, the method comprising:

encoding a joint chrominance component residual, JCCR, control flag into a bitstream;

encoding a chrominance mapping information into the bitstream based on the JCCR control flag;

encoding at least one chrominance quantization parameter, QP, offset into the bitstream based on the JCCR control flag;

providing the bitstream.

~~11~~8. The method according to claim ~~10~~7, wherein the bitstream comprises an SPS level syntax, and the JCCR control flag is encoded into the SPS level syntax.

~~12~~9. The method according to claim ~~10~~ 7 or ~~11~~8, wherein the JCCR control flag is the sps_joint_cbcr_enabled_flag.

~~13~~10. The method according to claim ~~12~~9, wherein if the value of the sps_joint_cbcr_enabled_flag is 1, the at least one encoded chrominance QP offset is specified by slice_joint_cbcr_qp_offset.

~~14. The method according to any one of claims 10 to 13, wherein the chrominance mapping information comprises delta_qp_in_val_minus1[i][j] and delta_qp_out_val[i][j], and the chrominance mapping information is encoded into an SPS level syntax comprised in the bitstream.~~

~~15. The method according to any one of claims 10 to 14, wherein the SPS level syntax comprises the following structure:~~

~~seq_parameter_set_rbsp( ) {~~	~~**Descriptor**~~
~~…~~	
~~**same_qp_table_for_chroma**~~	~~**u(1)**~~
~~for(i = 0; i < same_qp_table_for_chroma ? 1 : sps_joint_cbcr_enabled_flag ? 3 : 2; i++) {~~	
~~**num_points_in_qp_table_minus1[i]**~~	~~**ue(v)**~~
~~for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {~~	
~~**delta_qp_in_val_minus1[i][j]**~~	~~**ue(v)**~~
~~**…**~~	~~**ue(v)**~~
~~}~~	
~~}~~	
~~}~~	
~~…~~	
~~}~~	

~~16~~11. The method according to any one of claims ~~10~~ 7 to ~~15~~10, wherein the encoding the at least one chrominance QP offset into the bitstream based on the JCCR control flag comprises:

encoding, based on the JCCR control flag, the at least one chrominance QP offset into a picture parameter set, PPS, level syntax of the bitstream.

~~17. The method according to claim 16, wherein the PPS level syntax comprises the following structure:~~

~~pic_parameter_set_rbsp( ) {~~	~~Descriptor~~
~~…~~	
~~**pps_cb_qp_offset**~~	~~**se(v)**~~
~~**pps_cr_qp_offset**~~	~~**se(v)**~~
~~**if(sps_joint_cbcr_enabled_flag)**~~	
~~**pps_joint_cbcr_qp_offset**~~	~~**se(v)**~~
~~**cu_chroma_qp_offset_enabled_flag**~~	~~**u(1)**~~
~~if(pps_cu_chroma_qp_offset_list_enabled_flag) {~~	

~~cu_chroma_qp_offset_subdiv~~	~~ue(v)~~
~~chroma_qp_offset_list_len_minus1~~	~~ue(v)~~
~~for(i = 0; i <= chroma_qp_offset_list_len_minus1; i++) {~~	
~~cb_qp_offset_list[i]~~	~~se(v)~~
~~cr_qp_offset_list[i]~~	~~se(v)~~
~~if(sps_joint_cbcr_enabled_flag)~~	
~~joint_cbcr_qp_offset_list[i]~~	~~se(v)~~
~~}~~	
~~}~~	
~~…~~	
~~}~~	

~~18. A method for encoding a current block of a picture, wherein the method is performed by an encoder, the method comprising:~~

~~encoding syntax elements of a PPS syntax into a bitstream, wherein the syntax elements comprise at least one chrominance quantization parameter, QP, offset;~~

~~encoding chrominance QP offset information of a Slice Header syntax into the bitstream, wherein the QP offset information is encoded independently of any PPS syntax elements in the PPS syntax;~~

~~providing the bitstream.~~

~~19. A decoder (30) comprising processing circuitry for carrying out the method according to any one of claims 1 to 9.~~

~~20. An encoder (20) comprising processing circuitry for carrying out the method according to any one of claims 10 to 18.~~

~~21~~12. A computer program product comprising program code for performing the method according to any one of the preceding claims 1 – ~~18~~ 11 when executed on a computer or a processor.

~~22~~13. A decoder, comprising:

one or more processors; and

a non-transitory computer-readable storage medium coupled to the processors and storing programming for execution by the processors, wherein the programming, when executed by the processors, configures the decoder to carry out the method according to any one of the preceding claims 1 - ~~9~~6.

~~23~~14. ~~——~~An encoder, comprising:

one or more processors; and

a non-transitory computer-readable storage medium coupled to the processors and storing programming for execution by the processors, wherein the programming, when executed by the processors, configures the encoder to carry out the method according to any one of the preceding claims ~~10~~ 7 – ~~18~~11.

~~24~~15. A non-transitory computer-readable medium comprising a bitstream decoded by the method of any one of the claims 1-6.~~carrying a program code which, when executed by a computer device, causes the computer device to perform the method of any one of the preceding claims 1 - 18.~~

~~25. A decoder (30), comprising~~

~~a receiving unit configured to receive a bitstream;~~

~~a first obtaining unit configured to obtain a joint chrominance component residual, JCCR, control flag from the bitstream;~~

~~a second obtaining unit configured to obtain a chrominance mapping information from the bitstream based on the JCCR control flag;~~

~~a third obtaining unit configured to obtain at least one chrominance quantization parameter, QP, offset from the bitstream based on the JCCR control flag;~~

~~a fourth obtaining unit configured to obtain a QP value for the current chrominance block based on the obtained chrominance mapping information and the at least one obtained chrominance QP offset;~~

~~an inverse quantizing unit configured to perform inverse quantization on the current chrominance block by using the determined QP value.~~

~~26. The decoder according to claim 25, wherein the bitstream comprises an SPS level syntax, and the JCCR control flag is obtained from the SPS level syntax.~~

~~27. The decoder according to claim 25 or 26, wherein the JCCR control flag is the sps_joint_cbcr_enabled_flag.~~

28. The decoder according to claim 27, wherein if the value of the sps_joint_cbcr_enabled_flag is 1, the at least one obtained chrominance QP offset is specified by slice_joint_cbcr_qp_offset.

29. The decoder according to any one of claims 25 to 28, wherein the chrominance mapping information comprises delta_qp_in_val_minus1[i][j] and delta_qp_out_val[i][j], and the chrominance mapping information is obtained from an SPS level syntax comprised in the bitstream.

30. The decoder according to any one of claims 26 to 29, wherein the SPS level syntax comprises the following structure:

seq_parameter_set_rbsp( ) {	**Descriptor**
…	
same_qp_table_for_chroma	**u(1)**
for(i = 0; i < same_qp_table_for_chroma ? 1 : sps_joint_cbcr_enabled_flag ? 3 : 2; i++) {	
num_points_in_qp_table_minus1[i]	**ue(v)**
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {	
delta_qp_in_val_minus1[i][j]	**ue(v)**
…	**ue(v)**
}	
}	
}	
…	
}	

.

31. The decoder according to any one of claims 25 to 30, wherein the obtaining the at least one chrominance QP offset from the bitstream based on the JCCR control flag comprises:

obtaining, based on the JCCR control flag, the at least one chrominance QP offset from a picture parameter set, PPS, level syntax of the bitstream.

32. The decoder according to claim 31, wherein the PPS level syntax comprises the following structure:

pic_parameter_set_rbsp( ) {	Descriptor
…	
pps_cb_qp_offset	**se(v)**
pps_cr_qp_offset	**se(v)**

~~if(sps_joint_cbcr_enabled_flag)~~	
~~pps_joint_cbcr_qp_offset~~	~~se(v)~~
~~cu_chroma_qp_offset_enabled_flag~~	~~u(1)~~
~~if(pps_cu_chroma_qp_offset_list_enabled_flag) {~~	
~~cu_chroma_qp_offset_subdiv~~	~~ue(v)~~
~~chroma_qp_offset_list_len_minus1~~	~~ue(v)~~
~~for(i = 0; i <= chroma_qp_offset_list_len_minus1; i++) {~~	
~~cb_qp_offset_list[i]~~	~~se(v)~~
~~cr_qp_offset_list[i]~~	~~se(v)~~
~~if(sps_joint_cbcr_enabled_flag)~~	
~~joint_cbcr_qp_offset_list[i]~~	~~se(v)~~
~~}~~	
~~}~~	
~~…~~	
~~}~~	

~~33. A decoder (30), comprising:~~

~~a receiving unit configured to receive a bitstream, wherein the bitstream comprises a Slice Header syntax and a PPS syntax;~~

~~a first obtaining unit configured to obtain syntax elements from the PPS syntax, wherein the obtained syntax elements comprises chrominance quantization parameter, QP, offsets;~~

~~a second obtaining unit configured to obtain chrominance QP offset information from the Slice Header, wherein the QP offset information is obtained independently of any PPS syntax elements in the PPS syntax;~~

~~a determining unit configured to determining a QP value for the current chrominance block depending on the chrominance QP offset obtained from the PPS syntax and the chrominance QP offset information obtained from the Slice Header syntax;~~

~~an inverse quantization unit configured to perform inverse quantization on the current chrominance block by using the determined QP value.~~

~~34. An encoder (20), comprising:~~

~~a first encoding unit configured to encode a joint chrominance component residual, JCCR, control flag into a bitstream;~~

~~a second encoding unit configured to encode a chrominance mapping information into the bitstream based on the JCCR control flag;~~

a third encoding unit configured to encode at least one chrominance quantization parameter, QP, offset into the bitstream based on the JCCR control flag;

a providing unit configured to provide the bitstream.

35. The encoder according to claim 34, wherein the bitstream comprises an SPS level syntax, and the JCCR control flag is encoded into the SPS level syntax.

36. The encoder according to claim 33 or 35, wherein the JCCR control flag is the sps_joint_cbcr_enabled_flag.

37. The encoder according to claim 36, wherein if the value of the sps_joint_cbcr_enabled_flag is 1, the at least one encoded chrominance QP offset is specified by slice_joint_cbcr_qp_offset.

38. The encoder according to any one of claims 33 to 37, wherein the chrominance mapping information comprises delta_qp_in_val_minus1[i][j] and delta_qp_out_val[i][j], and the chrominance mapping information is encoded into an SPS level syntax comprised in the bitstream.

39. The encoder according to any one of claims 33 to 38, wherein the SPS level syntax comprises the following structure:

seq_parameter_set_rbsp( ) {	**Descriptor**
…	
same_qp_table_for_chroma	**u(1)**
for(i = 0; i < same_qp_table_for_chroma ? 1 : sps_joint_cbcr_enabled_flag ? 3 : 2; i++) {	
num_points_in_qp_table_minus1[i]	**ue(v)**
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {	
delta_qp_in_val_minus1[i][j]	**ue(v)**
…	**ue(v)**
}	
}	
}	
…	
}	

40. The encoder according to any one of claims 33 to 39, wherein the encoding the at least one chrominance QP offset into the bitstream based on the JCCR control flag comprises:

encoding, based on the JCCR control flag, the at least one chrominance QP offset into a picture parameter set, PPS, level syntax of the bitstream.

41. The encoder according to claim 40, wherein the PPS level syntax comprises the following structure:

pic_parameter_set_rbsp( ) {	Descriptor
…	
pps_cb_qp_offset	**se(v)**
pps_cr_qp_offset	**se(v)**
if(sps_joint_cbcr_enabled_flag)	
pps_joint_cbcr_qp_offset	**se(v)**
cu_chroma_qp_offset_enabled_flag	**u(1)**
if(pps_cu_chroma_qp_offset_list_enabled_flag) {	
cu_chroma_qp_offset_subdiv	**ue(v)**
chroma_qp_offset_list_len_minus1	**ue(v)**
for(i = 0; i <= chroma_qp_offset_list_len_minus1; i++) {	
cb_qp_offset_list[i]	**se(v)**
cr_qp_offset_list[i]	**se(v)**
if(sps_joint_cbcr_enabled_flag)	
joint_cbcr_qp_offset_list[i]	**se(v)**
}	
}	
…	
}	

42. An encoder (20), comprising:

a first encoding unit configured to encode syntax elements from Slice Header and PPS syntax into the bitstream, wherein the syntax elements comprise chrominance quantization parameter, QP, offsets;

a second encoding unit configured to encode chrominance QP offset information from the Slice Header into the bitstream, wherein the QP offset information is obtained independently of any PPS syntax elements in the PPS syntax;

a providing unit configured to provide the bitstream.