



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN)
CỤC SỞ HỮU TRÍ TUỆ

(11)



1-0052249

(51)^{2021.01} H04N 19/60; H04N 19/124; H04N 19/137; H04N 19/18; H04N 9/67; H04N 19/61; H04N 19/625; H04N 19/70; H04N 19/119; H04N 19/186 (13) B

(21) 1-2022-03971

(22) 26/11/2020

(86) PCT/KR2020/016968 26/11/2020

(87) WO 2021/107641 03/06/2021

(30) 10-2019-0153835 26/11/2019 KR; 10-2019-0160096 04/12/2019 KR; 10-2019-0177150 27/12/2019 KR

(45) 27/10/2025 451

(43) 25/08/2022 413A

(73) WILUS INSTITUTE OF STANDARDS AND TECHNOLOGY INC. (KR)
5F 216 Hwangsaekul-ro Bundang-gu Seongnam-si Gyeonggi-do 13595, Republic of Korea

(72) JUNG, Jaehong (KR); SON, Juhyung (KR); KIM, Dongcheol (KR); KO, Geonjung (KR); KWAK, Jinsam (KR).

(74) Công ty TNHH Tầm nhìn và Liên danh (VISION & ASSOCIATES CO.LTD.)

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ GIẢI MÃ TÍN HIỆU VIDEO VÀ VẬT GHI ĐỌC ĐƯỢC BỞI MÁY TÍNH LÂU DÀI

(21) 1-2022-03971

(57) Sáng chế đề cập đến phương pháp và thiết bị xử lý tín hiệu video, phương pháp này bao gồm các bước: thu nhận, từ dòng bit, cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi có thể tồn tại trong cú pháp đơn vị biến đổi hay không, cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời; thu nhận, từ dòng bit, cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được áp dụng hay không; khi cờ kích hoạt bỏ qua phép biến đổi biểu thị sự tồn tại của cờ bỏ qua phép biến đổi trong cú pháp đơn vị biến đổi hoặc cờ kích hoạt bảng màu biểu thị việc áp dụng chế độ bảng màu, thu nhận, từ dòng bit, thông tin liên quan đến thông số lượng tử hóa nhỏ nhất mà được cho phép theo chế độ bỏ qua phép biến đổi; thu nhận thông số lượng tử hóa nhỏ nhất dựa trên thông tin liên quan đến thông số lượng tử hóa nhỏ nhất; hiệu chỉnh thông số lượng tử hóa dựa trên thông số lượng tử hóa nhỏ nhất được thu nhận; và khôi phục khối hiện thời dựa trên thông số lượng tử hóa được hiệu chỉnh.

(a)	
seq_parameter_set_rbsp()	Bộ mô tả
...	
chroma_format_idc	u(2)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
...	
bit_depth_minus8	ue(v)
min_qp_prime_ts_minus4	ue(v)
...	
sps_transform_skip_enabled_flag	u(1)
...	
if(chroma_format_idc == 3) {	
sps_act_enabled_flag	u(1)
sps_palette_enabled_flag	u(1)
}	
...	
}	

(b)	
seq_parameter_set_rbsp()	Bộ mô tả
...	
chroma_format_idc	u(2)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
...	
bit_depth_minus8	ue(v)
...	
sps_transform_skip_enabled_flag	u(1)
...	
if(chroma_format_idc == 3) {	
sps_act_enabled_flag	u(1)
sps_palette_enabled_flag	u(1)
}	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
...	
}	

Fig. 21

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp và thiết bị xử lý tín hiệu video, và cụ thể hơn là, phương pháp và thiết bị xử lý tín hiệu video để mã hóa hoặc giải mã tín hiệu video.

Tình trạng kỹ thuật của sáng chế

Sự mã hóa nén đề cập đến một loạt các kỹ thuật xử lý tín hiệu để truyền thông tin được số hóa qua đường truyền thông hoặc lưu trữ thông tin dưới dạng thích hợp với vật ghi lưu trữ. Đối tượng của sự mã hóa nén bao gồm các đối tượng chẳng hạn như âm thanh, video và văn bản, và cụ thể là, kỹ thuật thực hiện việc mã hóa nén đối với hình ảnh được gọi là sự nén video. Sự mã hóa nén đối với tín hiệu video được thực hiện bằng cách loại bỏ thông tin dư thừa xét theo độ tương quan về không gian, độ tương quan về thời gian và độ tương quan ngẫu nhiên. Tuy nhiên, với sự phát triển gần đây của các phương tiện và phương tiện truyền dữ liệu khác nhau, cần có phương pháp và thiết bị xử lý tín hiệu video hiệu quả hơn.

Bản chất kỹ thuật của sáng chế

Vấn đề kỹ thuật

Một khía cạnh của sáng chế là cải thiện hiệu quả mã hóa tín hiệu video.

Giải pháp cho vấn đề

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu, từ dòng bit, cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không, thu, từ dòng bit, cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được áp dụng hay không, khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị việc áp dụng chế

độ bảng màu, thu, từ dòng bit, thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, thu thông số lượng tử hóa nhỏ nhất, dựa trên thông tin liên quan đến thông số lượng tử hóa nhỏ nhất, hiệu chỉnh thông số lượng tử hóa, dựa trên thông số lượng tử hóa nhỏ nhất thu được, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh, trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi của khối hiện thời, thì phép biến đổi không được áp dụng cho khối biến đổi.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu cờ bỏ qua phép biến đổi từ dòng bit, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi, thu thông số lượng tử hóa được hiệu chỉnh dựa trên độ lệch thông số lượng tử hóa, khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, cắt thông số lượng tử hóa được hiệu chỉnh thành 0, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu cờ bỏ qua phép biến đổi từ dòng bit, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi, thu thông số lượng tử hóa được hiệu chỉnh dựa trên độ lệch thông số lượng tử hóa, khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn thông số lượng tử hóa nhỏ nhất, cắt thông số lượng tử hóa được hiệu chỉnh thành thông số lượng tử hóa nhỏ nhất, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu, từ dòng bit, cờ kích hoạt phép biến đổi không gian màu sắc biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không, và xác định độ lệch thông số lượng tử hóa, dựa trên cờ kích hoạt phép biến đổi không gian màu sắc và thành phần màu sắc của khối biến đổi.

Theo phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước xác định độ lệch thông số lượng tử hóa bao gồm các bước: khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng

cho khối hiện thời, xác định độ lệch thông số lượng tử hóa là 0, và khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, xác định độ lệch thông số lượng tử hóa là độ lệch định trước cho mỗi thành phần màu sắc của khối biến đổi.

Theo phương pháp giải mã tín hiệu video theo một phương án của sáng chế, cờ kích hoạt bỏ qua phép biến đổi, cờ kích hoạt bảng màu và thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được thu từ ít nhất một dòng bit trong số bộ thông số chuỗi (sequence parameter set, SPS), bộ thông số ảnh (picture parameter set, PPS), phần đầu ảnh hoặc phần đầu phân chia.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu, từ dòng bit, cờ hệ số biến đổi thứ hai biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ hai được bao gồm trong khối kênh thứ hai tương ứng với khối hiện thời hay không, thu, từ dòng bit, cờ hệ số biến đổi thứ ba biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ ba được bao gồm trong khối kênh thứ ba tương ứng với khối hiện thời hay không, và khi thông tin biểu thị kiểu cây đang được xử lý là DUAL_TREE_LUMA hoặc SINGLE_TREE, thì phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh, và chế độ phân chia con nội ảnh không được áp dụng cho khối hiện thời, thu cờ hệ số biến đổi thứ nhất từ dòng bit, dựa trên việc liệu cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng hay có được áp dụng cho khối hiện thời, cờ hệ số biến đổi thứ nhất biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất được bao gồm trong khối hiện thời hay không, trong đó SINGLE_TREE biểu thị rằng cây đơn được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời, và DUAL_TREE_LUMA biểu thị rằng cây đôi được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời, và thành phần được bao gồm trong khối hiện thời là thành phần kênh thứ nhất.

Theo phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước thu cờ hệ số biến đổi thứ nhất từ dòng bit bao gồm bước: khi cờ kích hoạt phép biến đổi

không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thu cờ hệ số biến đổi thứ nhất từ dòng bit.

Theo phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước thu cờ hệ số biến đổi thứ nhất từ dòng bit bao gồm bước: khi một trong số nhiều điều kiện được thỏa mãn và cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thu cờ hệ số biến đổi thứ nhất từ dòng bit, và các điều kiện này bao gồm điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ hai biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ hai, điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ ba biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ ba, điều kiện rằng chiều rộng của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng, và điều kiện rằng chiều cao của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế còn bao gồm bước: khi cờ hệ số biến đổi thứ nhất không được thu từ dòng bit, tạo cấu hình cờ hệ số biến đổi thứ nhất để biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ nhất.

Thiết bị giải mã tín hiệu video theo một phương án của sáng chế bao gồm bộ xử lý và bộ nhớ, trong đó, dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý thu, từ dòng bit, cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không, thu, từ dòng bit, cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được áp dụng hay không, khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị việc áp dụng chế độ bảng màu, thu, từ dòng bit, thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, thu thông số lượng tử hóa nhỏ nhất, dựa trên thông tin liên quan đến

thông số lượng tử hóa nhỏ nhất, hiệu chỉnh thông số lượng tử hóa, dựa trên thông số lượng tử hóa nhỏ nhất thu được, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh, và khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi của khối hiện thời, thì phép biến đổi không được áp dụng cho khối biến đổi.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế thu cờ bỏ qua phép biến đổi từ dòng bit, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi, thu thông số lượng tử hóa được hiệu chỉnh dựa trên độ lệch thông số lượng tử hóa, khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, cắt thông số lượng tử hóa được hiệu chỉnh thành 0, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế thu cờ bỏ qua phép biến đổi từ dòng bit, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi, thu thông số lượng tử hóa được hiệu chỉnh dựa trên độ lệch thông số lượng tử hóa, khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn thông số lượng tử hóa nhỏ nhất, cắt thông số lượng tử hóa được hiệu chỉnh thành thông số lượng tử hóa nhỏ nhất, và khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế thu, từ dòng bit, cờ kích hoạt phép biến đổi không gian màu sắc biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không, và xác định độ lệch thông số lượng tử hóa, dựa trên cờ kích hoạt phép biến đổi không gian màu sắc và thành phần màu sắc của khối biến đổi.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế xác định độ lệch thông số lượng tử hóa là 0 khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, và xác định độ lệch thông số lượng tử hóa là độ lệch định trước cho mỗi thành phần màu sắc của khối biến đổi khi cờ kích hoạt phép biến

đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời.

Cờ kích hoạt bỏ qua phép biến đổi, cờ kích hoạt bảng màu và thông tin liên quan đến thông số lượng tử hóa nhỏ nhất của thiết bị giải mã tín hiệu video theo một phương án của sáng chế được thu từ ít nhất một dòng bit trong số bộ thông số chuỗi (SPS), bộ thông số ảnh (PPS), phần đầu ảnh hoặc phần đầu phân chia.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế thu, từ dòng bit, cờ hệ số biến đổi thứ hai biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ hai được bao gồm trong khối kênh thứ hai tương ứng với khối hiện thời hay không, thu, từ dòng bit, cờ hệ số biến đổi thứ ba biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ ba được bao gồm trong khối kênh thứ ba tương ứng với khối hiện thời hay không, và khi thông tin biểu thị kiểu cây đang được xử lý là DUAL_TREE_LUMA hoặc SINGLE_TREE, thì phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh, và chế độ phân chia con nội ảnh không được áp dụng cho khối hiện thời, thu cờ hệ số biến đổi thứ nhất từ dòng bit, dựa trên việc liệu cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng hay có được áp dụng cho khối hiện thời, cờ hệ số biến đổi thứ nhất biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất được bao gồm trong khối hiện thời hay không, và SINGLE_TREE biểu thị rằng cây đơn được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời, và DUAL_TREE_LUMA biểu thị rằng cây đôi được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời, và thành phần được bao gồm trong khối hiện thời là thành phần kênh thứ nhất.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế thu cờ hệ số biến đổi thứ nhất từ dòng bit khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video

theo một phương án của sáng chế thu cờ hệ số biến đổi thứ nhất từ dòng bit khi một trong số nhiều điều kiện được thỏa mãn và cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, và các điều kiện này bao gồm điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ hai biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ hai, điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ ba biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ ba, điều kiện rằng chiều rộng của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng, và điều kiện rằng chiều cao của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng.

Dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý của thiết bị giải mã tín hiệu video theo một phương án của sáng chế tạo cấu hình, khi cờ hệ số biến đổi thứ nhất không được thu từ dòng bit, thì cờ hệ số biến đổi thứ nhất để biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ nhất.

Phương pháp mã hóa tín hiệu video theo một phương án của sáng chế bao gồm các bước: tạo ra cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không, tạo ra cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được áp dụng hay không, khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị việc áp dụng chế độ bảng màu, tạo ra thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, dựa trên thông số lượng tử hóa nhỏ nhất, và tạo ra dòng bit, dựa trên cờ kích hoạt bỏ qua phép biến đổi, cờ kích hoạt bảng màu và thông tin liên quan đến thông số lượng tử hóa nhỏ nhất, trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi của khối hiện thời, thì phép biến đổi không được áp dụng cho khối biến đổi.

Thiết bị mã hóa tín hiệu video theo một phương án của sáng chế bao gồm bộ xử lý và bộ nhớ, trong đó, dựa trên lệnh được lưu trữ trong bộ nhớ, bộ xử lý tạo ra cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời tồn tại trong cú pháp đơn vị biến đổi, tạo ra cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được áp dụng hay không, khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị việc áp dụng chế độ bảng màu, tạo ra thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, dựa trên thông số lượng tử hóa nhỏ nhất, và tạo ra dòng bit, dựa trên cờ kích hoạt bỏ qua phép biến đổi, cờ kích hoạt bảng màu và thông tin liên quan đến thông số lượng tử hóa nhỏ nhất, và khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi của khối hiện thời, thì phép biến đổi không được áp dụng cho khối biến đổi.

Hiệu quả đạt được của sáng chế

Theo một phương án của sáng chế, hiệu quả mã hóa tín hiệu video có thể được cải thiện.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối giản lược của thiết bị mã hóa tín hiệu video theo một phương án của sáng chế.

Fig.2 là sơ đồ khối giản lược của thiết bị giải mã tín hiệu video theo một phương án của sáng chế.

Fig.3 là hình vẽ thể hiện một phương án trong đó đơn vị cây mã hóa được phân chia thành các đơn vị mã hóa trong ảnh.

Fig.4 là hình vẽ thể hiện một phương án của phương pháp báo hiệu sự phân chia của cây tứ phân và cây đa loại.

Fig.5 và Fig.6 là các hình vẽ minh họa cụ thể hơn phương pháp dự đoán nội ảnh theo một phương án của sáng chế.

Fig.7 là sơ đồ minh họa cụ thể phương pháp biến đổi tín hiệu dư bởi bộ mã hóa.

Fig.8 là sơ đồ minh họa cụ thể phương pháp để thu nhận tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi bằng bộ mã hóa và bộ giải mã.

Fig.9 là hình vẽ minh họa hàm cơ sở tương ứng với mỗi kiểu phép biến đổi.

Fig.10 là hình vẽ minh họa cường độ tín hiệu đối với mỗi chỉ số của các kiểu phép biến đổi trong số DCT-II, DCT-V, DCT-VIII, DST-I và DST-VII.

Fig.11 là hình vẽ minh họa cường độ tín hiệu đối với mỗi chỉ số của các kiểu phép biến đổi trong số DST-IV, DCT-IV, DST-VII và DCT-VIII.

Fig.12 là hình vẽ minh họa phương pháp tạo cấu hình bộ ứng viên biến đổi theo chế độ dự đoán của khối hiện thời.

Fig.13 là hình vẽ minh họa nhân biến đổi ngang và nhân biến đổi dọc được xác định theo chỉ số bộ.

Fig.14 là sơ đồ minh họa chi tiết phương pháp biến đổi tín hiệu dư thành hệ số biến đổi bởi bộ mã hóa thực hiện phép biến đổi màu sắc thích ứng.

Fig.15 là sơ đồ minh họa chi tiết phương pháp biến đổi ngược hệ số biến đổi để thu tín hiệu dư bởi bộ giải mã thực hiện phép biến đổi màu sắc thích ứng.

Fig.16 là hình vẽ minh họa phương pháp hiệu chỉnh giá trị thông số lượng tử hóa, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên thông số lượng tử hóa được hiệu chỉnh theo phương pháp thứ nhất của sáng chế.

Fig.17 là hình vẽ minh họa phương pháp hiệu chỉnh giá trị thông số lượng tử hóa, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên thông số lượng tử hóa được hiệu chỉnh theo phương pháp thứ hai của sáng chế.

Fig.18 là hình vẽ minh họa phương pháp hiệu chỉnh giá trị thông số lượng tử hóa, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên thông số lượng tử hóa được hiệu chỉnh theo phương pháp thứ ba của sáng chế.

Fig.19 là hình vẽ minh họa phương pháp hiệu chỉnh giá trị thông số lượng tử hóa,

và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên thông số lượng tử hóa được hiệu chỉnh theo phương pháp thứ tư của sáng chế.

Fig.20 là hình vẽ minh họa phương pháp hiệu chỉnh giá trị thông số lượng tử hóa, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên thông số lượng tử hóa được hiệu chỉnh theo phương pháp thứ năm của sáng chế.

Fig.21 là hình vẽ minh họa phương pháp báo hiệu thông số lượng tử hóa nhỏ nhất được áp dụng cho chế độ bỏ qua phép biến đổi.

Fig.22 là hình vẽ minh họa một phương án phân chia khối theo chế độ phân chia con nội ảnh (intra sub-partition, ISP).

Fig.23 là hình vẽ minh họa phương pháp biến đổi khối con (sub-block transform, SBT).

Fig.24 là hình vẽ minh họa cấu trúc cú pháp transform_tree() mà là cấu trúc cú pháp liên quan đến cây biến đổi.

Fig.25 là hình vẽ minh họa cấu trúc cú pháp transform_unit() theo một phương án của sáng chế.

Fig.26 là hình vẽ minh họa cấu trúc cú pháp transform_unit() theo phương án khác của sáng chế.

Mô tả chi tiết sáng chế

Các thuật ngữ được sử dụng trong bản mô tả này có thể là các thuật ngữ chung hiện được sử dụng rộng rãi xét theo các chức năng theo sáng chế nhưng có thể thay đổi theo các mục đích của những người có hiểu biết trung bình trong lĩnh vực kỹ thuật, các tùy chỉnh, hoặc sự xuất hiện của công nghệ mới. Ngoài ra, trong các trường hợp nhất định, có thể có các thuật ngữ mà chủ đơn lựa chọn một cách tùy ý và trong trường hợp này, nghĩa của các thuật ngữ này được mô tả trong phần mô tả tương ứng của sáng chế. Theo đó, các thuật ngữ được sử dụng trong bản mô tả này nên được hiểu dựa trên các ý nghĩa quan trọng của các thuật ngữ và các nội dung trong toàn bộ bản mô tả.

Trong bản mô tả này, một số thuật ngữ có thể được hiểu như sau. Sự mã hóa có thể

được hiểu là sự mã hóa hoặc sự giải mã trong một số trường hợp. Theo sáng chế, thiết bị tạo ra dòng bit tín hiệu video bằng cách thực hiện việc mã hóa (sự mã hóa) tín hiệu video được gọi là thiết bị mã hóa hoặc bộ mã hóa, và thiết bị thực hiện việc giải mã (sự giải mã) dòng bit tín hiệu video để khôi phục tín hiệu video được gọi là thiết bị giải mã hoặc bộ giải mã. Ngoài ra, trong bản mô tả này, thiết bị xử lý tín hiệu video được sử dụng làm thuật ngữ có khái niệm bao gồm cả bộ mã hóa và bộ giải mã. Thông tin là thuật ngữ bao gồm tất cả các giá trị, thông số, hệ số, phần tử, v.v.. Trong một số trường hợp, nghĩa này được hiểu một cách khác, nên sáng chế không bị giới hạn ở đó. Thuật ngữ “đơn vị” được sử dụng với nghĩa dùng để chỉ đơn vị cơ bản để xử lý hình ảnh hoặc vị trí cụ thể của ảnh, và đề cập đến vùng hình ảnh bao gồm ít nhất một thành phần độ sáng và thành phần sắc độ. Ngoài ra, thuật ngữ “khối” đề cập đến vùng hình ảnh bao gồm thành phần cụ thể trong số các thành phần độ sáng và các thành phần sắc độ (nghĩa là, Cb và Cr). Tuy nhiên, tùy thuộc vào phương án, các thuật ngữ chẳng hạn như “đơn vị”, “khối”, “phần chia” và “vùng” có thể được sử dụng thay thế lẫn nhau. Ngoài ra, trong bản mô tả này, bộ phận có thể được sử dụng làm khái niệm bao gồm tất cả bộ phận mã hóa, bộ phận dự đoán và bộ phận biến đổi. Ảnh biểu thị trường hoặc khung, và theo một phương án, các thuật ngữ này có thể được sử dụng thay thế lẫn nhau.

Fig.1 là sơ đồ khối giản lược của thiết bị mã hóa tín hiệu video 100 theo một phương án của sáng chế. Dựa vào Fig.1, thiết bị mã hóa 100 theo sáng chế bao gồm bộ phận biến đổi 110, bộ phận lượng tử hóa 115, bộ phận lượng tử hóa ngược 120, bộ phận biến đổi ngược 125, bộ phận lọc 130, bộ phận dự đoán 150 và bộ phận mã hóa entropy 160.

Bộ phận biến đổi 110 thu giá trị của hệ số biến đổi bằng cách biến đổi tín hiệu dư, mà là chênh lệch giữa tín hiệu video được nhập vào và tín hiệu được dự đoán được tạo ra bởi bộ phận dự đoán 150. Ví dụ, phép biến đổi cosin rời rạc (Discrete Cosine Transform, DCT), phép biến đổi sin rời rạc (Discrete Sine Transform, DST) hoặc phép biến đổi wavelet có thể được sử dụng. DCT và DST thực hiện phép biến đổi bằng cách phân chia tín hiệu ảnh đầu vào thành các khối. Trong phép biến đổi này, hiệu suất mã hóa có thể thay đổi theo sự phân bố và các đặc điểm của các giá trị ở vùng biến đổi. Bộ phận lượng tử hóa 115 lượng tử hóa giá trị của giá trị hệ số biến đổi được kết xuất từ bộ phận biến đổi 110.

Để cải thiện hiệu suất mã hóa, thay vì mã hóa tín hiệu ảnh, phương pháp dự đoán ảnh sử dụng vùng đã được mã hóa thông qua bộ phận dự đoán 150 và thu ảnh được khôi phục bằng cách thêm giá trị dư giữa ảnh gốc và ảnh được dự đoán vào ảnh được dự đoán được sử dụng. Để ngăn ngừa sự không khớp trong bộ mã hóa và bộ giải mã, thông tin có thể được sử dụng trong bộ giải mã nên được sử dụng khi thực hiện phép dự đoán trong bộ mã hóa. Vì vậy, bộ mã hóa thực hiện lại quy trình khôi phục khối hiện thời được mã hóa. Bộ phận lượng tử hóa ngược 120 lượng tử hóa ngược (thay đổi tỷ lệ) giá trị của hệ số biến đổi, và bộ phận biến đổi ngược 125 khôi phục giá trị dư nhờ sử dụng giá trị hệ số biến đổi được lượng tử hóa ngược (thay đổi tỷ lệ). Trong khi đó, bộ phận lọc 130 thực hiện các hoạt động lọc để cải thiện chất lượng của ảnh được khôi phục và để cải thiện hiệu suất mã hóa. Ví dụ, bộ lọc giải khối, độ lệch thích ứng mẫu (sample adaptive offset, SAO) và bộ lọc vòng lặp thích ứng có thể được bao gồm. Ảnh được lọc được kết xuất hoặc lưu trữ trong bộ đệm ảnh được giải mã (decoded picture buffer, DPB) 156 để sử dụng làm ảnh tham chiếu.

Để tăng hiệu suất mã hóa, thay vì mã hóa tín hiệu ảnh, phương pháp để thu nhận ảnh được khôi phục được sử dụng trong đó ảnh được dự đoán nhờ sử dụng vùng đã được mã hóa thông qua bộ phận dự đoán 150, và giá trị dư giữa ảnh gốc và ảnh được dự đoán được thêm vào ảnh được dự đoán. Bộ phận dự đoán nội ảnh 152 thực hiện phép dự đoán nội ảnh trong ảnh hiện thời, và bộ phận dự đoán liên ảnh 154 dự đoán ảnh hiện thời bằng cách sử dụng ảnh tham chiếu được lưu trữ trong bộ đệm ảnh được giải mã 156. Bộ phận dự đoán nội ảnh 152 thực hiện phép dự đoán nội ảnh từ các vùng được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa nội ảnh đến bộ phận mã hóa entropy 160. Hơn nữa, bộ phận dự đoán liên ảnh 154 có thể bao gồm bộ phận đánh giá chuyển động 154a và bộ phận bù chuyển động 154b. Bộ phận đánh giá chuyển động 154a thu giá trị vector chuyển động của vùng hiện thời bằng cách tham chiếu đến vùng cụ thể được khôi phục. Bộ phận đánh giá chuyển động 154a có thể truyền thông tin vị trí (khung tham chiếu, vector chuyển động hoặc thông tin tương tự) của vùng tham chiếu đến bộ phận mã hóa entropy 160 để được bao gồm trong dòng bit. Bộ phận bù chuyển động 154b thực hiện việc bù chuyển động liên ảnh nhờ sử dụng giá trị vector chuyển động được truyền từ bộ phận đánh giá

chuyển động 154a.

Bộ phận dự đoán 150 bao gồm bộ phận dự đoán nội ảnh 152 và bộ phận dự đoán liên ảnh 154. Bộ phận dự đoán nội ảnh 152 thực hiện phép dự đoán nội ảnh trong ảnh hiện thời, và bộ phận dự đoán liên ảnh 154 thực hiện phép dự đoán liên ảnh để dự đoán ảnh hiện thời bằng cách sử dụng ảnh tham chiếu được lưu trữ trong DPB 156. Bộ phận dự đoán nội ảnh 152 thực hiện phép dự đoán nội ảnh từ các mẫu được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa nội ảnh đến bộ phận mã hóa entropy 160. Thông tin mã hóa nội ảnh có thể bao gồm ít nhất một trong số chế độ dự đoán nội ảnh, cờ chế độ có xác suất lớn nhất (Most Probable Mode, MPM) và chỉ số MPM. Thông tin mã hóa nội ảnh có thể bao gồm thông tin về mẫu tham chiếu. Bộ phận dự đoán liên ảnh 154 có thể bao gồm bộ phận đánh giá chuyển động 154a và bộ phận bù chuyển động 154b. Bộ phận đánh giá chuyển động 154a tham chiếu đến vùng cụ thể của ảnh tham chiếu được khôi phục để thu giá trị vector chuyển động của vùng hiện thời. Bộ phận đánh giá chuyển động 154a truyền tập thông tin chuyển động (chỉ số ảnh tham chiếu, thông tin vector chuyển động, v.v.) về vùng tham chiếu đến bộ phận mã hóa entropy 160. Bộ phận bù chuyển động 154b thực hiện việc bù chuyển động nhờ sử dụng giá trị vector chuyển động được truyền từ bộ phận đánh giá chuyển động 154a. Bộ phận dự đoán liên ảnh 154 truyền thông tin mã hóa liên ảnh bao gồm thông tin chuyển động về vùng tham chiếu đến bộ phận mã hóa entropy 160.

Theo một phương án khác, bộ phận dự đoán 150 có thể bao gồm bộ phận dự đoán sao chép khối (block copy, BC) nội ảnh (không được minh họa trên hình vẽ). Bộ phận dự đoán BC nội ảnh này thực hiện phép dự đoán BC nội ảnh từ các mẫu được khôi phục trong ảnh hiện thời, và truyền thông tin mã hóa BC nội ảnh đến bộ phận mã hóa entropy 160. Bộ phận dự đoán BC nội ảnh tham chiếu đến vùng cụ thể trong ảnh hiện thời và thu giá trị vector khối biểu thị vùng tham chiếu cần được sử dụng để dự đoán vùng hiện thời. Bộ phận dự đoán BC nội ảnh có thể thực hiện phép dự đoán BC nội ảnh nhờ sử dụng giá trị vector khối thu được. Bộ phận dự đoán BC nội ảnh truyền thông tin mã hóa BC nội ảnh đến bộ phận mã hóa entropy 160. Thông tin mã hóa BC nội ảnh có thể bao gồm thông tin vector khối.

Khi phép dự đoán ảnh được mô tả ở trên được thực hiện, thì bộ phận biến đổi 110 biến đổi giá trị dư giữa ảnh gốc và ảnh được dự đoán để thu giá trị hệ số biến đổi. Trong trường hợp này, phép biến đổi có thể được thực hiện trong đơn vị khối cụ thể nằm trong ảnh, và kích thước của khối cụ thể có thể được thay đổi nằm trong khoảng được thiết lập trước. Bộ phận lượng tử hóa 115 lượng tử hóa giá trị hệ số biến đổi được tạo ra trong bộ phận biến đổi 110 và truyền giá trị được lượng tử hóa này đến bộ phận mã hóa entropy 160.

Bộ phận mã hóa entropy 160 mã hóa entropy thông tin về các hệ số biến đổi được lượng tử hóa, thông tin mã hóa nội ảnh và thông tin mã hóa liên ảnh để tạo ra dòng bit tín hiệu video. Trong bộ phận mã hóa entropy 160, phương pháp mã hóa có độ dài khả biến (variable length coding, VLC), phương pháp mã hóa số học hoặc phương pháp tương tự có thể được sử dụng. Phương pháp VLC biến đổi các ký hiệu được nhập vào thành các từ mã liên tiếp, và độ dài của các từ mã này có thể thay đổi. Ví dụ, các ký hiệu xuất hiện thường xuyên được thể hiện dưới dạng các từ mã ngắn, và các ký hiệu xuất hiện ít thường xuyên hơn được thể hiện dưới dạng các từ mã dài. Là phương pháp VLC, phương pháp mã hóa có độ dài khả biến thích ứng dựa trên ngữ cảnh (context-based adaptive variable length coding, CAVLC) có thể được sử dụng. Sự mã hóa số học biến đổi các ký hiệu dữ liệu liên tiếp thành dấu phẩy thập phân duy nhất, và sự mã hóa số học có thể thu các bit thập phân tối ưu cần thiết để biểu diễn mỗi ký hiệu. Là sự mã hóa số học, sự mã hóa số học thích ứng dựa trên ngữ cảnh (context-based adaptive arithmetic coding, CABAC) có thể được sử dụng. Ví dụ, bộ phận mã hóa entropy 160 có thể nhị phân hóa thông tin biểu diễn hệ số biến đổi được lượng tử hóa. Ngoài ra, bộ phận mã hóa entropy 160 có thể tạo ra dòng bit bằng cách mã hóa số học thông tin nhị phân.

Dòng bit được tạo ra được đóng gói nhờ sử dụng đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) làm đơn vị cơ bản. Đơn vị NAL bao gồm số lượng nguyên các đơn vị cây mã hóa được mã hóa. Để giải mã dòng bit trong bộ giải mã video, đầu tiên, dòng bit phải được tách riêng theo các đơn vị NAL, và sau đó mỗi đơn vị NAL được tách riêng phải được giải mã. Trong khi đó, thông tin cần thiết để giải mã dòng bit tín hiệu video có thể được truyền qua tập hợp tải tin chuỗi byte thô (Raw Byte Sequence Payload, RBSP) mức cao hơn chẳng hạn như bộ thông số ảnh (Picture Parameter Set, PPS),

bộ thông số chuỗi (Sequence Parameter Set, SPS), bộ thông số video (Video Parameter Set, VPS) và bộ thông số tương tự.

Trong khi đó, sơ đồ khối theo Fig.1 thể hiện thiết bị mã hóa 100 theo một phương án của sáng chế, và các khối được thể hiện riêng biệt phân biệt về mặt logic và thể hiện các thành phần của thiết bị mã hóa 100. Theo đó, các thành phần của thiết bị mã hóa 100 được mô tả ở trên có thể được lắp dưới dạng một chip hoặc dưới dạng nhiều chip tùy thuộc vào thiết kế của thiết bị. Theo một phương án, hoạt động của mỗi thành phần của thiết bị mã hóa 100 được mô tả ở trên có thể được thực hiện bởi bộ xử lý (không được thể hiện trên hình vẽ).

Fig.2 là sơ đồ khối giản lược của thiết bị giải mã tín hiệu video 200 theo một phương án của sáng chế. Dựa vào Fig.2, thiết bị giải mã 200 theo sáng chế bao gồm bộ phận giải mã entropy 210, bộ phận giải lượng tử hóa 220, bộ phận biến đổi ngược 225, bộ phận lọc 230 và bộ phận dự đoán 250.

Bộ phận giải mã entropy 210 giải mã entropy dòng bit tín hiệu video để trích xuất thông tin hệ số biến đổi, thông tin mã hóa nội ảnh, thông tin mã hóa liên ảnh và thông tin tương tự đối với mỗi vùng. Ví dụ, bộ phận giải mã entropy 210 có thể thu mã nhị phân cho thông tin hệ số biến đổi của vùng cụ thể từ dòng bit tín hiệu video. Hơn nữa, bộ phận giải mã entropy 210 thu hệ số biến đổi được lượng tử hóa bằng cách nhị phân hóa ngược mã nhị phân. Bộ phận giải lượng tử hóa 220 giải lượng tử hóa hệ số biến đổi được lượng tử hóa. Bộ phận biến đổi ngược 225 khôi phục giá trị dư bằng cách sử dụng hệ số biến đổi được giải lượng tử hóa. Thiết bị xử lý tín hiệu video 200 khôi phục giá trị điểm ảnh gốc bằng cách cộng giá trị dư được thu bởi bộ phận biến đổi ngược 225 với giá trị dự đoán được thu bởi bộ phận dự đoán 250.

Trong khi đó, bộ phận lọc 230 thực hiện việc lọc đối với ảnh để cải thiện chất lượng hình ảnh. Bộ phận lọc này có thể bao gồm bộ lọc giải khối để làm giảm độ biến dạng khối và/hoặc bộ lọc vòng lặp thích ứng để loại bỏ độ biến dạng của toàn bộ ảnh. Ảnh được lọc được kết xuất hoặc lưu trữ trong DPB 256 để sử dụng làm ảnh tham chiếu cho ảnh tiếp theo.

Bộ phận dự đoán 250 bao gồm bộ phận dự đoán nội ảnh 252 và bộ phận dự đoán liên ảnh 254. Bộ phận dự đoán 250 tạo ra ảnh dự đoán bằng cách sử dụng kiểu mã hóa được giải mã thông qua bộ phận giải mã entropy 210 được mô tả ở trên, các hệ số biến đổi đối với mỗi vùng, và thông tin mã hóa nội ảnh/liên ảnh. Để khôi phục khối hiện thời trong đó việc giải mã được thực hiện, vùng được giải mã của ảnh hiện thời hoặc các ảnh khác bao gồm khối hiện thời có thể được sử dụng. Ảnh (hoặc tấm/phần chia) chỉ sử dụng ảnh hiện thời để khôi phục, nghĩa là, thực hiện phép dự đoán nội ảnh hoặc dự đoán BC nội ảnh được gọi là ảnh nội ảnh hoặc ảnh (hoặc tấm/phần chia) I, và ảnh (hoặc tấm/phần chia) thực hiện tất cả trong số dự đoán nội ảnh, dự đoán liên ảnh và dự đoán BC nội ảnh được gọi là ảnh (hoặc tấm/phần chia) liên ảnh. Để dự đoán các giá trị mẫu của mỗi khối trong số các ảnh (hoặc tấm/phần chia) liên ảnh, ảnh (hoặc tấm/phần chia) sử dụng lên đến một vector chuyển động và chỉ số ảnh tham chiếu được gọi là ảnh dự đoán hoặc ảnh (hoặc tấm/phần chia) P, và ảnh (hoặc tấm/phần chia) sử dụng lên đến hai vector chuyển động và chỉ số ảnh tham chiếu được gọi là ảnh dự đoán theo hai hướng hoặc ảnh (hoặc tấm/phần chia) B. Nói cách khác, ảnh (hoặc tấm/phần chia) P sử dụng lên đến một tập thông tin chuyển động để dự đoán mỗi khối, và ảnh (hoặc tấm/phần chia) B sử dụng lên đến hai tập thông tin chuyển động để dự đoán mỗi khối. Ở đây, tập thông tin chuyển động bao gồm một hoặc nhiều vector chuyển động và một chỉ số ảnh tham chiếu.

Bộ phận dự đoán nội ảnh 252 tạo ra khối dự đoán nhờ sử dụng thông tin mã hóa nội ảnh và các mẫu được khôi phục trong ảnh hiện thời. Như được mô tả ở trên, thông tin mã hóa nội ảnh có thể bao gồm ít nhất một trong số chế độ dự đoán nội ảnh, cờ chế độ có xác suất lớn nhất (MPM) và chỉ số MPM. Bộ phận dự đoán nội ảnh 252 dự đoán các giá trị mẫu của khối hiện thời bằng cách sử dụng các mẫu được khôi phục nằm ở phía bên trái và/hoặc bên trên của khối hiện thời làm các mẫu tham chiếu. Theo sáng chế này, các mẫu được khôi phục, các mẫu tham chiếu và các mẫu của khối hiện thời có thể biểu diễn các điểm ảnh. Ngoài ra, các giá trị mẫu có thể biểu diễn các giá trị điểm ảnh.

Theo một phương án, các mẫu tham chiếu có thể là các mẫu được bao gồm trong khối lân cận của khối hiện thời. Ví dụ, các mẫu tham chiếu có thể là các mẫu liền kề với biên bên trái của khối hiện thời và/hoặc các mẫu liền kề với biên phía trên. Ngoài ra, các

mẫu tham chiếu có thể là các mẫu nằm trên đường nằm trong khoảng cách định trước từ biên bên trái của khối hiện thời và/hoặc các mẫu nằm trên đường nằm trong khoảng cách định trước từ biên phía trên của khối hiện thời trong số các mẫu của các khối lân cận của khối hiện thời. Trong trường hợp này, khối lân cận của khối hiện thời có thể bao gồm khối bên trái (left, L), khối phía trên (upper, A), khối phía dưới bên trái (below left, BL), khối phía trên bên phải (above right, AR) hoặc khối phía trên bên trái (above left, AL).

Bộ phận dự đoán liên ảnh 254 tạo ra khối dự đoán nhờ sử dụng các ảnh tham chiếu và thông tin mã hóa liên ảnh được lưu trữ trong DPB 256. Thông tin mã hóa liên ảnh có thể bao gồm tập thông tin chuyển động (chỉ số ảnh tham chiếu, thông tin vector chuyển động, v.v.) của khối hiện thời đối với khối tham chiếu. Phép dự đoán liên ảnh có thể bao gồm phép dự đoán L0, phép dự đoán L1 và phép dự đoán theo hai hướng. Phép dự đoán L0 nghĩa là phép dự đoán sử dụng một ảnh tham chiếu được bao gồm trong danh sách ảnh L0, và phép dự đoán L1 nghĩa là phép dự đoán sử dụng một ảnh tham chiếu được bao gồm trong danh sách ảnh L1. Vì vậy, một tập hợp thông tin chuyển động (ví dụ, vector chuyển động và chỉ số ảnh tham chiếu) có thể cần thiết. Trong phương pháp dự đoán theo hai hướng, lên đến hai vùng tham chiếu có thể được sử dụng, và hai vùng tham chiếu này có thể tồn tại trong cùng ảnh tham chiếu hoặc có thể tồn tại trong các ảnh khác nhau. Nghĩa là, trong phương pháp dự đoán theo hai hướng, lên đến hai tập hợp thông tin chuyển động (ví dụ, vector chuyển động và chỉ số ảnh tham chiếu) có thể được sử dụng và hai vector chuyển động có thể tương ứng với cùng chỉ số ảnh tham chiếu hoặc các chỉ số ảnh tham chiếu khác nhau. Trong trường hợp này, các ảnh tham chiếu có thể được hiển thị (hoặc được kết xuất) cả trước và sau ảnh hiện thời về mặt thời gian. Theo một phương án, hai vùng tham chiếu được sử dụng trong sơ đồ dự đoán theo hai hướng có thể là các vùng được chọn từ mỗi trong số danh sách ảnh L0 và danh sách ảnh L1.

Bộ phận dự đoán liên ảnh 254 có thể thu khối tham chiếu của khối hiện thời nhờ sử dụng vector chuyển động và chỉ số ảnh tham chiếu. Khối tham chiếu ở trong ảnh tham chiếu tương ứng với chỉ số ảnh tham chiếu. Ngoài ra, giá trị mẫu của khối được quy định bởi vector chuyển động hoặc giá trị được nội suy của giá trị mẫu có thể được sử dụng làm bộ dự đoán của khối hiện thời. Đối với phép dự đoán chuyển động với độ chính xác điểm ảnh

ở mức đơn vị điểm ảnh con, ví dụ, bộ lọc nội suy 8 tap đối với tín hiệu độ sáng và bộ lọc nội suy 4 tap đối với tín hiệu sắc độ có thể được sử dụng. Tuy nhiên, bộ lọc nội suy để dự đoán chuyển động theo các đơn vị điểm ảnh con không bị giới hạn ở đó. Theo cách này, bộ phận dự đoán liên ảnh 254 thực hiện việc bù chuyển động để dự đoán bộ cục của đơn vị hiện thời từ các ảnh chuyển động được khôi phục trước đó. Trong trường hợp này, bộ phận dự đoán liên ảnh có thể sử dụng tập thông tin chuyển động.

Theo một phương án khác, bộ phận dự đoán 250 có thể bao gồm bộ phận dự đoán BC nội ảnh (không được minh họa trên hình vẽ). Bộ phận dự đoán BC nội ảnh có thể khôi phục vùng hiện thời bằng cách tham chiếu đến vùng cụ thể bao gồm các mẫu được khôi phục trong ảnh hiện thời. Bộ phận dự đoán BC nội ảnh thu thông tin mã hóa BC nội ảnh về vùng hiện thời từ bộ phận giải mã entropy 210. Bộ phận dự đoán BC nội ảnh thu giá trị vector khối của vùng hiện thời biểu thị vùng cụ thể trong ảnh hiện thời. Bộ phận dự đoán BC nội ảnh có thể thực hiện phép dự đoán BC nội ảnh nhờ sử dụng giá trị vector khối thu được. Thông tin mã hóa BC nội ảnh có thể bao gồm thông tin vector khối.

Ảnh video được khôi phục được tạo ra bằng cách thêm giá trị dự đoán được kết xuất từ bộ phận dự đoán nội ảnh 252 hoặc bộ phận dự đoán liên ảnh 254 và giá trị dư được kết xuất từ bộ phận biến đổi ngược 225. Nghĩa là, thiết bị giải mã tín hiệu video 200 khôi phục khối hiện thời nhờ sử dụng khối dự đoán được tạo ra bởi bộ phận dự đoán 250 và phần dư được thu từ bộ phận biến đổi ngược 225.

Trong khi đó, sơ đồ khối trên Fig.2 thể hiện thiết bị giải mã 200 theo một phương án của sáng chế, và các khối được thể hiện riêng biệt phân biệt về mặt logic và thể hiện các thành phần của thiết bị giải mã 200. Theo đó, các thành phần của thiết bị giải mã 200 được mô tả ở trên có thể được lắp dưới dạng một chip hoặc dưới dạng nhiều chip tùy thuộc vào thiết kế của thiết bị. Theo một phương án, hoạt động của mỗi thành phần của thiết bị giải mã 200 được mô tả ở trên có thể được thực hiện bởi bộ xử lý (không được thể hiện trên hình vẽ).

Fig.3 minh họa một phương án trong đó đơn vị cây mã hóa (coding tree unit, CTU) được phân chia thành các đơn vị mã hóa (coding unit, CU) trong ảnh. Trong quy trình mã

hóa tín hiệu video, ảnh có thể được phân chia thành chuỗi đơn vị cây mã hóa (CTU). Đơn vị cây mã hóa bao gồm NXN khối của các mẫu độ sáng và hai khối của các mẫu sắc độ tương ứng. Đơn vị cây mã hóa có thể được phân chia thành các đơn vị mã hóa. Đơn vị cây mã hóa không được phân chia và có thể là nút lá. Trong trường hợp này, đơn vị cây mã hóa này có thể là đơn vị mã hóa. Đơn vị mã hóa đề cập đến đơn vị cơ bản để xử lý ảnh trong quy trình xử lý tín hiệu video được mô tả ở trên, nghĩa là, phép dự đoán nội ảnh/liên ảnh, phép biến đổi, phép lượng tử hóa và/hoặc mã hóa entropy. Kích thước và hình dạng của đơn vị mã hóa trong một ảnh có thể không cố định. Đơn vị mã hóa có thể có hình dạng vuông hoặc hình dạng chữ nhật. Đơn vị mã hóa hình chữ nhật (hoặc khối hình chữ nhật) bao gồm đơn vị mã hóa dọc (hoặc khối dọc) và đơn vị mã hóa ngang (hoặc khối ngang). Theo sáng chế, khối dọc là khối có chiều cao lớn hơn chiều rộng, và khối ngang là khối có chiều rộng lớn hơn chiều cao. Ngoài ra, trong bản mô tả này, khối không vuông có thể đề cập đến khối hình chữ nhật, nhưng sáng chế không bị giới hạn ở đó.

Dựa vào Fig.3, đầu tiên, đơn vị cây mã hóa được phân chia thành cấu trúc cây tứ phân (quad tree, QT). Nghĩa là, một nút có kích thước $2N \times 2N$ trong cấu trúc cây tứ phân có thể được phân chia thành bốn nút có kích thước NXN. Theo sáng chế, cây tứ phân cũng có thể được gọi là cây có bốn phần. Việc phân chia cây tứ phân có thể được thực hiện theo cách đệ quy, và không phải tất cả các nút đều cần được phân chia với cùng độ sâu.

Trong khi đó, nút lá của cây tứ phân được mô tả ở trên có thể còn được phân chia thành cấu trúc cây đa loại (multi-type tree, MTT). Theo một phương án của sáng chế, trong cấu trúc cây đa loại, một nút có thể được phân chia thành cấu trúc cây nhị phân hoặc tam phân theo sự phân chia theo hướng ngang hoặc theo hướng dọc. Nghĩa là, trong cấu trúc cây đa loại, có bốn cấu trúc phân chia chẳng hạn như phân chia nhị phân theo hướng dọc, phân chia nhị phân theo hướng ngang, phân chia tam phân theo hướng dọc và phân chia tam phân theo hướng ngang. Theo một phương án của sáng chế, trong mỗi trong số các cấu trúc cây, tất cả chiều rộng và chiều cao của các nút có thể có lũy thừa cơ số 2. Ví dụ, trong cấu trúc cây nhị phân (binary tree, BT), nút có kích thước là $2N \times 2N$ có thể được phân chia thành hai nút $N \times 2N$ bằng cách phân chia nhị phân theo hướng dọc, và phân chia thành hai nút $2N \times N$ bằng cách phân chia nhị phân theo hướng ngang. Ngoài ra, trong cấu trúc cây

tam phân (ternary tree, TT), nút có kích thước $2N \times 2N$ được phân chia thành các nút $(N/2) \times 2N$, $N \times 2N$, và $(N/2) \times 2N$ bằng cách phân chia tam phân theo hướng dọc, và phân chia thành các nút $2N \times (N/2)$, $2N \times N$ và $2N \times (N/2)$ bằng cách phân chia tam phân theo hướng ngang. Việc phân chia cây đa loại này có thể được thực hiện theo cách đệ quy.

Nút lá của cây đa loại có thể là đơn vị mã hóa. Nếu sự phân chia đối với đơn vị mã hóa không được biểu thị hoặc đơn vị mã hóa không lớn so với độ dài biến đổi tối đa, thì đơn vị mã hóa được sử dụng làm đơn vị dự đoán và biến đổi mà không được phân chia hơn nữa. Mặt khác, ít nhất một thông số trong số các thông số sau về cây tứ phân và cây đa loại được mô tả ở trên có thể được xác định trước hoặc được truyền qua tập hợp RBSP có mức cao hơn chẳng hạn như PPS, SPS, VPS và dạng tương tự. 1) kích thước CTU: kích thước nút gốc của cây tứ phân, 2) kích thước QT tối thiểu MinQtSize: kích thước nút lá QT được cho phép tối thiểu, 3) kích thước BT tối đa MaxBtSize: kích thước nút gốc BT được cho phép tối đa, 4) kích thước TT tối đa MaxTtSize: kích thước nút gốc TT được cho phép tối đa, 5) độ sâu MTT tối đa MaxMttDepth: độ sâu được cho phép tối đa của MTT được phân chia từ nút lá của QT, 6) kích thước BT tối thiểu MinBtSize: kích thước nút lá BT được cho phép tối thiểu, 7) kích thước TT tối thiểu MinTtSize: kích thước nút lá TT được cho phép tối thiểu.

Fig.4 minh họa một phương án của phương pháp báo hiệu sự phân chia cây tứ phân và cây đa loại. Các cờ được thiết lập trước có thể được sử dụng để báo hiệu sự phân chia cây tứ phân và cây đa loại được mô tả ở trên. Dựa vào Fig.4, ít nhất một trong số cờ 'split_cu_flag' biểu thị xem nút có được phân chia hay không, cờ 'split_qt_flag' biểu thị xem nút cây tứ phân có được phân chia hay không, cờ 'mtt_split_cu_vertical_flag' biểu thị hướng phân chia của nút cây đa loại hoặc cờ 'mtt_split_cu_binary_flag' biểu thị hình dạng phân chia của nút cây đa loại có thể được sử dụng.

Theo một phương án của sáng chế, 'split_cu_flag', mà là cờ biểu thị xem nút hiện thời có được phân chia hay không, có thể được báo hiệu đầu tiên. Khi giá trị của 'split_cu_flag' là 0, thì biểu thị rằng nút hiện thời không được phân chia, và nút hiện thời trở thành đơn vị mã hóa. Khi nút hiện thời là đơn vị cây mã hóa, thì đơn vị cây mã hóa này

bao gồm một đơn vị mã hóa không được phân chia. Khi nút hiện thời là nút cây tứ phân 'QT node', thì nút hiện thời là nút lá 'QT leaf node' của cây tứ phân và trở thành đơn vị mã hóa. Khi nút hiện thời là nút cây đa loại 'MTT node', thì nút hiện thời là nút lá 'MTT nút lá' của cây đa loại và trở thành đơn vị mã hóa.

Khi giá trị của 'split_cu_flag' là 1, thì nút hiện thời có thể được phân chia thành các nút của cây tứ phân hoặc cây đa loại theo giá trị của 'split_qt_flag'. Đơn vị cây mã hóa là nút gốc của cây tứ phân, và có thể được phân chia đầu tiên thành cấu trúc cây tứ phân. Trong cấu trúc cây tứ phân, 'split_qt_flag' được báo hiệu đối với mỗi nút 'QT node'. Khi giá trị của 'split_qt_flag' là 1, thì nút được phân chia thành 4 nút vuông, và khi giá trị của 'split_qt_flag' là 0, thì nút trở thành nút lá của cây tứ phân 'QT nút lá', và nút được phân chia thành các nút của cây đa loại. Theo một phương án của sáng chế, sự phân chia cây tứ phân có thể bị giới hạn theo loại nút hiện thời. Khi nút hiện thời là đơn vị cây mã hóa (nút gốc của cây tứ phân) hoặc nút cây tứ phân, thì việc phân chia cây tứ phân có thể được cho phép, và khi nút hiện thời là nút cây đa loại, thì việc phân chia cây tứ phân có thể không được cho phép. Mỗi nút lá của cây tứ phân 'QT nút lá' có thể còn được phân chia thành các cấu trúc cây đa loại. Như được mô tả ở trên, khi 'split_qt_flag' là 0, thì nút hiện thời có thể được phân chia thành các nút đa loại. Để biểu thị hướng phân chia và hình dạng phân chia, 'mtt_split_cu_vertical_flag' và 'mtt_split_cu_binary_flag' có thể được báo hiệu. Khi giá trị của 'mtt_split_cu_vertical_flag' là 1, thì sự phân chia theo hướng dọc của nút 'MTT node' được biểu thị, và khi giá trị của 'mtt_split_cu_vertical_flag' là 0, thì sự phân chia theo hướng ngang của nút 'MTT node' được biểu thị. Ngoài ra, khi giá trị của 'mtt_split_cu_binary_flag' là 1, thì nút 'MTT node' được phân chia thành hai nút hình chữ nhật, và khi giá trị của 'mtt_split_cu_binary_flag' là 0, thì nút 'MTT node' được phân chia thành ba nút hình chữ nhật.

Phép dự đoán ảnh (bù chuyển động) để mã hóa được thực hiện đối với các đơn vị mã hóa không còn được phân chia (nghĩa là, các nút lá của cây đơn vị mã hóa). Đơn vị cơ bản mà thực hiện phép dự đoán này sau đây được gọi là đơn vị dự đoán hoặc khối dự đoán. Sau đây, thuật ngữ đơn vị được sử dụng trong bản mô tả có thể được sử dụng làm thuật ngữ để thay thế cho đơn vị dự đoán, mà là đơn vị cơ bản để thực hiện phép dự đoán. Tuy

nhiên, sáng chế không bị giới hạn ở đó, và có thể được hiểu một cách rộng hơn là khái niệm này bao gồm đơn vị mã hóa.

Fig.5 và Fig.6 là các hình vẽ minh họa cụ thể hơn phương pháp dự đoán nội ảnh theo một phương án của sáng chế. Như được mô tả ở trên, bộ phận dự đoán nội ảnh dự đoán các giá trị mẫu của khối hiện thời bằng cách sử dụng các mẫu được khôi phục nằm ở phía bên trái và/hoặc phía trên của khối hiện thời làm các mẫu tham chiếu.

Đầu tiên, Fig.5 thể hiện một phương án của các mẫu tham chiếu được sử dụng để dự đoán khối hiện thời theo chế độ dự đoán nội ảnh. Theo một phương án, các mẫu tham chiếu có thể là các mẫu liền kề với biên bên trái của khối hiện thời và/hoặc các mẫu liền kề với biên phía trên. Như được thể hiện trên Fig.5, khi kích thước của khối hiện thời là WXH và các mẫu của đường tham chiếu duy nhất liền kề với khối hiện thời được sử dụng để dự đoán nội ảnh, thì các mẫu tham chiếu có thể được tạo cấu hình nhờ sử dụng tối đa $2W+2H+1$ mẫu lân cận nằm ở phía bên trái và/hoặc phía trên của khối hiện thời.

Ngoài ra, nếu ít nhất một số mẫu trong số các mẫu cần được sử dụng làm các mẫu tham chiếu chưa được khôi phục, thì bộ phận dự đoán nội ảnh có thể thu mẫu tham chiếu bằng cách thực hiện quy trình đệm mẫu tham chiếu. Ngoài ra, bộ phận dự đoán nội ảnh có thể thực hiện quy trình lọc mẫu tham chiếu để làm giảm các lỗi trong phép dự đoán nội ảnh. Nghĩa là, việc thực hiện việc lọc đối với các mẫu bao quanh và/hoặc các mẫu tham chiếu được thu bằng quy trình đệm mẫu tham chiếu có thể được thực hiện để thu các mẫu tham chiếu được lọc. Do đó, bộ phận dự đoán nội ảnh dự đoán các mẫu của khối hiện thời nhờ sử dụng các mẫu tham chiếu được thu. Bộ phận dự đoán nội ảnh dự đoán các mẫu của khối hiện thời bằng cách sử dụng các mẫu tham chiếu chưa được lọc hoặc các mẫu tham chiếu đã được lọc. Theo sáng chế, các mẫu bao quanh có thể bao gồm các mẫu trên ít nhất một đường tham chiếu. Ví dụ, các mẫu bao quanh có thể bao gồm các mẫu liền kề nằm trên đường liền kề với biên của khối hiện thời.

Tiếp theo, Fig.6 thể hiện một phương án của các chế độ dự đoán được sử dụng để dự đoán nội ảnh. Để dự đoán nội ảnh, thông tin chế độ dự đoán nội ảnh biểu thị hướng dự đoán nội ảnh có thể được báo hiệu. Thông tin chế độ dự đoán nội ảnh biểu thị một chế độ

trong số các chế độ dự đoán nội ảnh được bao gồm trong bộ chế độ dự đoán nội ảnh. Khi khối hiện thời là khối dự đoán nội ảnh, thì bộ giải mã nhận thông tin chế độ dự đoán nội ảnh của khối hiện thời từ dòng bit. Bộ phận dự đoán nội ảnh của bộ giải mã thực hiện phép dự đoán nội ảnh đối với khối hiện thời dựa vào thông tin chế độ dự đoán nội ảnh được trích xuất.

Theo một phương án của sáng chế, bộ chế độ dự đoán nội ảnh có thể bao gồm tất cả các chế độ dự đoán nội ảnh được sử dụng trong phép dự đoán nội ảnh (ví dụ, tổng cộng 67 chế độ dự đoán nội ảnh). Cụ thể hơn, bộ chế độ dự đoán nội ảnh có thể bao gồm chế độ mặt phẳng, chế độ DC, và nhiều (ví dụ, 65) chế độ góc (nghĩa là, các chế độ định hướng). Mỗi chế độ dự đoán nội ảnh có thể được biểu thị thông qua chỉ số được thiết lập trước (nghĩa là, chỉ số chế độ dự đoán nội ảnh). Ví dụ, như được thể hiện trên Fig.6, chỉ số chế độ dự đoán nội ảnh 0 biểu thị chế độ mặt phẳng, và chỉ số chế độ dự đoán nội ảnh 1 biểu thị chế độ DC. Ngoài ra, các chỉ số chế độ dự đoán nội ảnh từ 2 đến 66 có thể lần lượt biểu thị các chế độ góc khác nhau. Các chế độ góc lần lượt biểu thị các góc khác nhau nằm trong phạm vi góc được thiết lập trước. Ví dụ, chế độ góc có thể biểu thị góc nằm trong phạm vi góc (nghĩa là, phạm vi góc thứ nhất) nằm trong khoảng từ 45 độ đến -135 độ theo chiều kim đồng hồ. Chế độ góc có thể được xác định dựa vào hướng 12 giờ. Trong trường hợp này, chỉ số chế độ dự đoán nội ảnh 2 biểu thị chế độ chéo ngang (horizontal diagonal, HDIA), chỉ số chế độ dự đoán nội ảnh 18 biểu thị chế độ ngang (horizontal, HOR), chỉ số chế độ dự đoán nội ảnh 34 biểu thị chế độ chéo (diagonal, DIA), chỉ số chế độ dự đoán nội ảnh 50 biểu thị chế độ dọc (vertical, VER) và chỉ số chế độ dự đoán nội ảnh 66 biểu thị chế độ chéo dọc (vertical diagonal, VDIA).

Mặt khác, để tăng hiệu suất mã hóa, thay vì mã hóa tín hiệu dư được mô tả ở trên, phương pháp trong đó giá trị hệ số biến đổi được thu bằng cách biến đổi tín hiệu dư được lượng tử hóa và hệ số biến đổi được lượng tử hóa được mã hóa có thể được sử dụng. Như được mô tả ở trên, bộ phận biến đổi có thể thu giá trị hệ số biến đổi bằng cách biến đổi tín hiệu dư. Trong trường hợp này, tín hiệu dư của khối cụ thể có thể được phân bố trên toàn bộ vùng của khối hiện thời. Theo đó, có thể cải thiện hiệu suất mã hóa bằng cách tập trung năng lượng trong miền tần số thấp thông qua phép biến đổi miền tần số trên tín hiệu dư.

Sau đây, phương pháp để biến đổi hoặc biến đổi ngược tín hiệu dư sẽ được mô tả một cách chi tiết.

Fig.7 là sơ đồ minh họa cụ thể phương pháp biến đổi tín hiệu dư bởi bộ mã hóa. Như được mô tả ở trên, tín hiệu dư trong miền không gian có thể được biến đổi sang miền tần số. Bộ mã hóa có thể thu hệ số biến đổi bằng cách biến đổi tín hiệu dư thu được. Đầu tiên, bộ mã hóa có thể thu nhận ít nhất một khối dư bao gồm tín hiệu dư cho khối hiện thời. Khối dư có thể là khối bất kỳ trong số khối hiện thời hoặc các khối mà khối hiện thời được chia thành. Theo sáng chế, khối dư có thể được gọi là mảng dư hoặc ma trận dư bao gồm các mẫu dư của khối hiện thời. Ngoài ra, theo sáng chế, khối dư có thể biểu diễn đơn vị biến đổi hoặc khối có kích thước giống với kích thước của khối biến đổi.

Tiếp theo, bộ mã hóa có thể biến đổi khối dư nhờ sử dụng nhân biến đổi. Nhân biến đổi được sử dụng để biến đổi khối dư có thể là nhân biến đổi có các đặc điểm có thể tách riêng thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Trong trường hợp này, phép biến đổi đối với khối dư có thể tách được thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Ví dụ, bộ mã hóa có thể thực hiện phép biến đổi theo hướng dọc bằng cách áp dụng nhân biến đổi theo hướng dọc của khối dư. Ngoài ra, bộ mã hóa có thể thực hiện phép biến đổi theo hướng ngang bằng cách áp dụng nhân biến đổi theo hướng ngang của khối dư. Theo sáng chế, nhân biến đổi có thể được sử dụng làm thuật ngữ để đề cập đến bộ thông số được sử dụng để biến đổi tín hiệu dư chẳng hạn như ma trận biến đổi, mảng biến đổi và hàm biến đổi. Theo một phương án, nhân biến đổi có thể là nhân bất kỳ trong số các nhân khả dụng. Hơn nữa, nhân biến đổi dựa vào các loại phép biến đổi khác nhau có thể được sử dụng cho mỗi phép biến đổi trong số phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Phương pháp chọn một trong số nhiều nhân biến đổi khả dụng sẽ được mô tả sau đây dựa vào các hình vẽ từ Fig.12 đến Fig.13.

Bộ mã hóa có thể truyền khối biến đổi được biến đổi từ khối dư đến bộ phận lượng tử hóa để lượng tử hóa khối biến đổi này. Trong trường hợp này, khối biến đổi có thể bao gồm các hệ số biến đổi. Cụ thể là, khối biến đổi có thể bao gồm các hệ số biến đổi được

sắp xếp theo hai chiều. Giống như khối dư, kích thước của khối biến đổi có thể giống với kích thước của khối bất kỳ trong số khối hiện thời hoặc khối mà khối hiện thời được chia. Các hệ số biến đổi được truyền đến bộ phận lượng tử hóa có thể được biểu diễn dưới dạng các giá trị được lượng tử hóa.

Ngoài ra, bộ mã hóa có thể thực hiện phép biến đổi khác trước khi hệ số biến đổi được lượng tử hóa. Như được minh họa trên Fig.7, phương pháp biến đổi được mô tả ở trên có thể được gọi là phép biến đổi sơ cấp, và phép biến đổi khác có thể được gọi là phép biến đổi thứ cấp. Phép biến đổi thứ cấp có thể chọn lọc đối với mỗi khối dư. Theo một phương án, bộ mã hóa có thể cải thiện hiệu suất mã hóa bằng cách thực hiện phép biến đổi thứ cấp đối với vùng trong đó khó tập trung năng lượng ở miền tần số thấp chỉ bằng phép biến đổi sơ cấp. Ví dụ, phép biến đổi thứ cấp có thể được thêm vào khối trong đó các giá trị dư xuất hiện tương đối lớn theo hướng khác với hướng ngang hoặc hướng dọc của khối dư. Các giá trị dư của khối được dự đoán nội ảnh có thể có xác suất thay đổi tương đối cao theo hướng khác với hướng ngang hoặc hướng dọc so với các giá trị dư của khối được dự đoán liên ảnh. Theo đó, bộ mã hóa có thể thực hiện thêm phép biến đổi thứ cấp đối với tín hiệu dư của khối được dự đoán nội ảnh. Ngoài ra, bộ mã hóa có thể lược bỏ phép biến đổi thứ cấp đối với tín hiệu dư của khối được dự đoán liên ảnh.

Ví dụ khác, việc xem có thực hiện phép biến đổi thứ cấp hay không có thể được xác định theo kích thước của khối hiện thời hoặc khối dư. Ngoài ra, các nhân biến đổi có các kích thước khác nhau có thể được sử dụng theo kích thước của khối hiện thời hoặc khối dư. Ví dụ, phép biến đổi thứ cấp 8×8 có thể được áp dụng cho khối trong đó cạnh ngắn hơn trong số chiều rộng hoặc chiều cao ngắn hơn chiều dài được thiết lập trước thứ nhất. Ngoài ra, phép biến đổi thứ cấp 4×4 có thể được áp dụng cho khối trong đó cạnh ngắn hơn trong số chiều rộng hoặc chiều cao lớn hơn chiều dài được thiết lập trước thứ hai. Trong trường hợp này, độ dài được thiết lập trước thứ nhất có thể là giá trị lớn hơn độ dài được thiết lập trước thứ hai; tuy nhiên, sáng chế không bị giới hạn ở đó. Hơn nữa, không giống như phép biến đổi sơ cấp, phép biến đổi thứ cấp có thể không thể tách riêng được thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Phép biến đổi thứ cấp này có thể được gọi là phép biến đổi không thể tách riêng được tần số thấp (low

frequency non-separable transform, LFNST).

Ngoài ra, trong trường hợp tín hiệu video ở vùng cụ thể, năng lượng ở dải tần số cao có thể không được giảm ngay cả khi phép biến đổi tần số được thực hiện do sự thay đổi đột ngột về độ sáng. Theo đó, hiệu suất của hiệu suất nén do lượng tử hóa có thể bị suy giảm. Ngoài ra, khi phép biến đổi được thực hiện đối với vùng trong đó giá trị dư gần như không tồn tại, thì thời gian mã hóa và thời gian giải mã có thể tăng lên một cách không cần thiết. Theo đó, phép biến đổi đối với tín hiệu dư của vùng cụ thể có thể được lược bỏ. Việc xem có thực hiện phép biến đổi đối với tín hiệu dư của vùng cụ thể hay không có thể được xác định bằng phần tử cú pháp liên quan đến phép biến đổi đối với vùng cụ thể. Ví dụ, phần tử cú pháp có thể bao gồm thông tin bỏ qua phép biến đổi. Thông tin bỏ qua phép biến đổi có thể là cờ bỏ qua phép biến đổi. Khi thông tin bỏ qua phép biến đổi về khối dư biểu thị bỏ qua phép biến đổi, thì phép biến đổi đối với khối dư không được thực hiện. Trong trường hợp này, bộ mã hóa có thể ngay lập tức lượng tử hóa tín hiệu dư mà phép biến đổi của vùng này chưa được thực hiện. Các hoạt động của bộ mã hóa được mô tả dựa vào Fig.7 có thể được thực hiện thông qua bộ phận biến đổi trên Fig.1.

Các phần tử cú pháp được mô tả ở trên liên quan đến phép biến đổi có thể là thông tin được phân tích từ dòng bit tín hiệu video. Bộ giải mã có thể giải mã entropy dòng bit tín hiệu video để thu các phần tử cú pháp liên quan đến phép biến đổi. Ngoài ra, bộ mã hóa có thể tạo ra dòng bit tín hiệu video bằng cách mã hóa entropy các phần tử cú pháp liên quan đến phép biến đổi.

Fig.8 là sơ đồ minh họa cụ thể phương pháp để thu nhận tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi bằng bộ mã hóa và bộ giải mã. Sau đây, nhằm thuận tiện cho phần mô tả, hoạt động biến đổi ngược được thực hiện thông qua bộ phận biến đổi ngược của mỗi trong số bộ mã hóa và bộ giải mã sẽ được mô tả. Bộ phận biến đổi ngược có thể thu tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi được lượng tử hóa ngược. Đầu tiên, bộ phận biến đổi ngược có thể phát hiện xem phép biến đổi ngược đối với vùng cụ thể có được thực hiện hay không từ phần tử cú pháp liên quan đến phép biến đổi của vùng cụ thể. Theo một phương án, khi phần tử cú pháp liên quan đến phép biến đổi đối với khối biến

đổi cụ thể biểu thị bỏ qua phép biến đổi, thì phép biến đổi đối với khối biến đổi có thể được lược bỏ. Trong trường hợp này, cả hai phép biến đổi sơ cấp ngược và phép biến đổi thứ cấp ngược có thể được lược bỏ đối với khối biến đổi. Ngoài ra, hệ số biến đổi được lượng tử hóa ngược có thể được sử dụng làm tín hiệu dư. Ví dụ, bộ giải mã có thể khôi phục khối hiện thời bằng cách sử dụng hệ số biến đổi được lượng tử hóa ngược làm tín hiệu dư. Phép biến đổi sơ cấp ngược được mô tả ở trên biểu diễn phép biến đổi ngược đối với phép biến đổi sơ cấp, và có thể được gọi là phép biến đổi ngược sơ cấp.

Theo một phương án khác, phần tử cú pháp liên quan đến phép biến đổi đối với khối biến đổi cụ thể có thể không biểu thị bỏ qua phép biến đổi. Ở đây, bộ phận biến đổi ngược có thể xác định xem có thực hiện phép biến đổi thứ cấp ngược đối với phép biến đổi thứ cấp hay không. Ví dụ, nếu khối biến đổi là khối biến đổi của khối được dự đoán nội ảnh, thì phép biến đổi thứ cấp ngược có thể được thực hiện đối với khối biến đổi này. Ngoài ra, nhân biến đổi thứ cấp được sử dụng cho khối biến đổi tương ứng có thể được xác định dựa trên chế độ dự đoán nội ảnh tương ứng với khối biến đổi. Là ví dụ khác, xem có thực hiện phép biến đổi thứ cấp ngược hay không có thể được xác định dựa trên kích thước của khối biến đổi.

Bộ phận biến đổi ngược có thể thực hiện phép biến đổi sơ cấp ngược của hệ số biến đổi được giải lượng tử hóa hoặc hệ số biến đổi được biến đổi thứ cấp ngược. Giống như phép biến đổi sơ cấp, phép biến đổi sơ cấp ngược có thể tách được thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Ví dụ, bộ phận biến đổi ngược có thể thực hiện phép biến đổi ngược theo hướng dọc và phép biến đổi ngược theo hướng ngang đối với khối biến đổi để thu khối dư. Bộ phận biến đổi ngược có thể biến đổi ngược khối biến đổi dựa vào nhân biến đổi được sử dụng để biến đổi khối biến đổi. Ví dụ, bộ mã hóa có thể báo hiệu một cách rõ ràng hoặc ngầm thông tin biểu thị nhân biến đổi được áp dụng cho khối biến đổi hiện thời trong số các nhân biến đổi khả dụng. Bộ giải mã có thể chọn nhân biến đổi cần được sử dụng cho phép biến đổi ngược của khối biến đổi trong số các nhân biến đổi khả dụng bằng cách sử dụng thông tin biểu thị nhân biến đổi được báo hiệu. Bộ phận biến đổi ngược có thể khôi phục khối hiện thời bằng cách sử dụng tín hiệu dư được thu thông qua phép biến đổi ngược đối với hệ số biến đổi.

Mặt khác, sự phân bố tín hiệu dư của ảnh có thể khác nhau đối với mỗi vùng. Ví dụ, sự phân bố các giá trị đối với tín hiệu dư ở vùng cụ thể có thể thay đổi theo phương pháp dự đoán. Khi biến đổi các vùng biến đổi khác nhau nhờ sử dụng cùng nhân biến đổi, thì hiệu suất mã hóa có thể thay đổi đối với mỗi vùng biến đổi tùy thuộc vào các sự phân bố và đặc điểm của các giá trị ở vùng biến đổi. Theo đó, khi nhân biến đổi được sử dụng để biến đổi khối biến đổi cụ thể được chọn một cách thích ứng trong số các nhân biến đổi khả dụng, thì hiệu suất mã hóa có thể được cải thiện hơn nữa. Nghĩa là, bộ mã hóa và bộ giải mã có thể được tạo cấu hình để sử dụng thêm nhân biến đổi khác với nhân biến đổi cơ bản khi biến đổi tín hiệu video. Phương pháp chọn một cách thích ứng nhân biến đổi có thể được gọi là phép biến đổi đa lõi thích ứng (adaptive multiple core transform, AMT) hoặc lựa chọn đa biến đổi (multiple transform selection, MTS). Phương pháp chọn một cách thích ứng nhân biến đổi sẽ được mô tả sau đây dựa vào các hình vẽ liên quan. Theo sáng chế, nhằm thuận tiện cho phần giải thích, phép biến đổi và phép biến đổi ngược có thể được gọi chung là phép biến đổi. Hơn nữa, nhân biến đổi và nhân biến đổi ngược được gọi chung là nhân biến đổi.

Sau đây, nhiều nhân biến đổi mà có thể được sử dụng để biến đổi tín hiệu video sẽ được mô tả dựa vào Fig.9. Theo một phương án của sáng chế, nhân biến đổi có thể là nhân được suy ra dựa trên hàm cơ sở cụ thể. Mỗi trong số nhiều nhân biến đổi khác nhau có thể được thu dựa trên hàm cơ sở khác nhau. Nhiều nhân biến đổi có thể được thu dựa trên hàm cơ sở tương ứng với mỗi trong số các kiểu phép biến đổi khác nhau.

Fig.9 là hình vẽ minh họa hàm cơ sở tương ứng với mỗi kiểu phép biến đổi. Theo một phương án, nhân biến đổi khả dụng để biến đổi tín hiệu dư video có thể bao gồm ít nhất một trong số nhân biến đổi dựa trên phép biến đổi cosin rời rạc kiểu 2 (DCT-II), nhân biến đổi dựa trên phép biến đổi cosin rời rạc kiểu 5 (DCT-V), nhân biến đổi dựa trên phép biến đổi cosin rời rạc kiểu 8 (DCT-VIII), nhân biến đổi dựa trên phép biến đổi sin rời rạc kiểu 1 (DST-I) và nhân biến đổi dựa trên phép biến đổi sin rời rạc kiểu 7 (DST-VII).

Dựa vào Fig.9, các hàm cơ sở tương ứng với các kiểu phép biến đổi tương ứng trong số DCT-II, DCT-V, DCT-VIII, DST-I và DST-VII được mô tả ở trên có thể được biểu diễn

dưới dạng các hàm cosin hoặc sin. Ví dụ, các hàm cơ sở tương ứng với DCT-II, DCT-V và DCT-VIII tương ứng có thể là hàm kiểu cosin, và các hàm cơ sở tương ứng với DST-I và DST-VII tương ứng có thể là hàm kiểu sin. Hơn nữa, hàm cơ sở tương ứng với kiểu phép biến đổi cụ thể có thể được biểu diễn ở dạng hàm cơ sở đối với mỗi bin tần số. Ví dụ, hàm cơ sở $T_i(j)$ có thể là hàm cơ sở tương ứng với bin tần số thứ i . Nghĩa là, do giá trị được biểu thị bởi i là nhỏ hơn, nên hàm cơ sở tương ứng với tần số thấp hơn được biểu thị. Hơn nữa, do giá trị được biểu thị bởi i là lớn hơn, nên hàm cơ sở tương ứng với tần số cao hơn được biểu thị. Trên Fig.9, j có thể biểu thị số nguyên nằm trong khoảng từ 0 đến $N-1$.

Hàm cơ sở $T_i(j)$ có thể được biểu diễn là ma trận hai chiều thể hiện phần tử ở hàng thứ i và cột thứ j . Ở đây, phép biến đổi sử dụng các nhân biến đổi dựa trên kiểu phép biến đổi của Fig.9 có đặc điểm tách riêng được khi thực hiện phép biến đổi. Nghĩa là, phép biến đổi đối với tín hiệu dư có thể được biến đổi theo cách tách riêng theo mỗi trong số hướng ngang và hướng dọc. Ví dụ, phép biến đổi nhờ sử dụng ma trận biến đổi T liên quan đến khối dư X có thể được biểu diễn là phép toán ma trận TXT' . Ở đây, T' đề cập đến ma trận chuyển vị của ma trận biến đổi T . Ngoài ra, phép biến đổi ngược sử dụng ma trận biến đổi T liên quan đến khối biến đổi Y có thể được biểu diễn là $T'YT$.

Các giá trị của ma trận biến đổi được xác định bởi các hàm cơ sở được minh họa trên Fig.9 có thể là ở dạng số thập phân thay vì dạng số nguyên. Có thể khó thực hiện các giá trị thập phân trong phần cứng trong thiết bị mã hóa video và thiết bị giải mã. Do đó, nhân biến đổi được tính xấp xỉ số nguyên từ nhân biến đổi gốc bao gồm các giá trị ở dạng các số thập phân có thể được sử dụng để mã hóa và giải mã tín hiệu video. Nhân biến đổi được tính xấp xỉ bao gồm các giá trị nguyên có thể được tạo ra bằng cách thay đổi tỷ lệ và làm tròn nhân biến đổi gốc. Giá trị nguyên được bao gồm trong nhân biến đổi được tính xấp xỉ có thể là giá trị nằm trong khoảng mà có thể biểu diễn bằng số lượng bit được tạo cấu hình trước. Số lượng bit được tạo cấu hình trước có thể là 8 hoặc 10. Theo phép tính xấp xỉ, các tính chất trực chuẩn của DCT và DST có thể không được duy trì. Tuy nhiên, sự mất mát hiệu quả mã hóa do đó là không lớn, và nhờ đó có thể có lợi, xét theo phương án thực hiện phần cứng, để tính xấp xỉ nhân biến đổi ở dạng số nguyên.

Fig.10 là hình vẽ minh họa cường độ tín hiệu đối với mỗi chỉ số của các kiểu phép biến đổi trong số DCT-II, DCT-V, DCT-VIII, DST-I và DST-VII. Fig.10 minh họa dạng của hàm cơ sở tương ứng với bin tần số thấp nhất trong số các hàm cơ sở đối với mỗi bin tần số. Fig.10 minh họa hàm cơ sở tương ứng với bin tần số thứ 0 trong số các hàm cơ sở đối với mỗi bin tần số. Trên Fig.10, trục ngang thể hiện chỉ số j ($j=0, 1, \dots, N-1$) trong hàm cơ sở, và trục dọc thể hiện giá trị cường độ của tín hiệu. N thể hiện số lượng mẫu ở vùng cụ thể cần được áp dụng phép biến đổi.

Như được minh họa trên Fig.10, trong DST-VII, khi chỉ số j tăng, thì cường độ tín hiệu có xu hướng tăng lên. Do đó, giống như khối dư của khối được dự đoán nội ảnh, DST-VII có thể có hiệu quả để biến đổi khối dư trong đó cường độ của tín hiệu dư tăng lên khi khoảng cách theo các hướng ngang và dọc tăng theo phía trên bên trái của khối dư.

Mặt khác, trong DCT-VIII, khi chỉ số j tăng, thì cường độ tín hiệu có xu hướng giảm. Nghĩa là, DCT-VIII thỏa mãn các đặc điểm đối ngẫu với DST-VII. Do đó, DCT-VIII có thể có hiệu quả để biến đổi khối dư trong đó cường độ của tín hiệu dư giảm khi khoảng cách theo các hướng ngang và dọc tăng theo phía trên bên trái của khối dư.

Trong DST-I, khi chỉ số j trong hàm cơ sở tăng, thì cường độ tín hiệu tăng, và sau đó cường độ tín hiệu giảm bắt đầu từ chỉ số cụ thể. Do đó, DST-I có thể có hiệu quả để biến đổi khối dư có cường độ tín hiệu dư lớn ở tâm của khối dư này.

Hàm cơ sở thứ 0 của DCT-II thể hiện DC. Do đó, có thể có hiệu quả để biến đổi khối dư có sự phân bố cường độ tín hiệu đồng đều bên trong khối dư này.

DCT-V tương tự với DCT-II. Tuy nhiên, trong DCT-V, cường độ tín hiệu khi chỉ số j là 0 có giá trị nhỏ hơn giá trị của cường độ tín hiệu khi chỉ số j không phải là 0. Nghĩa là, khi chỉ số j là 1, thì DCT-V có mô hình tín hiệu trong đó đường thẳng bị uốn cong.

Như được mô tả ở trên, nếu nhân biến đổi có đặc điểm tách riêng được được sử dụng, thì phép biến đổi có thể được thực hiện theo mỗi hướng trong số các hướng ngang và dọc của khối dư. Cụ thể là, phép biến đổi của khối dư có thể được thực hiện bằng cách tiến hành phép toán nhân ma trận 2D hai lần. Phép toán nhân ma trận có thể bao gồm số lượng phép toán, mà tương ứng với mức cao hơn hoặc bằng mức được tạo cấu hình trước.

Theo đó, khi khôi dư được biến đổi nhờ sử dụng nhân biến đổi dựa trên DCT-II, thì lượng phép tính toán có thể được làm giảm bằng cách sử dụng cấu trúc hình cánh bướm. Tuy nhiên, phương án thực hiện của DST-VII và DCT-VIII có thể là khó về mặt số lượng phép toán, do độ phức tạp của phương án thực hiện tương đối cao. Theo đó, các kiểu phép biến đổi, mà có các đặc điểm lần lượt tương tự với các đặc điểm của DST-VII và DCT-VIII, và có độ phức tạp thực hiện tương đối thấp, có thể được sử dụng.

Theo một phương án, phép biến đổi sin rời rạc kiểu IV (discrete sine transform type-IV, DST-IV) và phép biến đổi cosin rời rạc kiểu IV (discrete cosine transform type-IV, DCT-IV) có thể lần lượt thay thế DST-VII và DCT-VIII. Fig.11 minh họa cường độ tín hiệu đối với mỗi chỉ số của kiểu phép biến đổi, chẳng hạn như DST-IV, DCT-IV, DST-VII và DCT-VIII. Fig.11(a) minh họa hàm cơ sở tương ứng với DST-IV và hàm cơ sở tương ứng với DCT-IV. DST-IV và DCT-IV đối với số lượng mẫu, N , có thể được suy ra từ DCT-II đối với số lượng mẫu, $2N$. Nghĩa là, cấu trúc hình cánh bướm một phần DCT-II đối với số lượng mẫu, $2N$, bao gồm DCT-IV đối với số lượng mẫu, N . DST-IV đối với số lượng mẫu, N , có thể được thực hiện bằng cách sắp xếp phép toán đảo ngược ký hiệu và hàm cơ sở tương ứng theo thứ tự ngược từ DCT-IV đối với số lượng mẫu, N .

Như được minh họa trên Fig.11(b), DST-IV biểu thị mô hình tín hiệu tương tự với mô hình tín hiệu của DST-VII. Theo đó, giống như khối dư của khối được dự đoán nội ảnh, DST-IV có thể có hiệu quả để biến đổi khối dư trong đó cường độ tín hiệu dư tăng lên khi khoảng cách theo các hướng ngang và dọc tăng theo phía trên bên trái của khối dư. DCT-IV thể hiện mô hình báo hiệu tương tự với mô hình báo hiệu của DCT-VIII. Theo đó, DCT-IV có thể có hiệu quả để biến đổi khối dư có cường độ tín hiệu dư, mà là lớn nhất ở biên cụ thể và giảm theo hướng đến biên khác xa hơn so với biên cụ thể.

Nếu chỉ có nhân biến đổi dựa trên một trong số các kiểu phép biến đổi được mô tả ở trên được sử dụng để biến đổi tín hiệu video, thì khó thực hiện phép biến đổi thích ứng theo đặc điểm của tín hiệu gốc và mẫu hình của tín hiệu dư, mà thay đổi theo chế độ dự đoán. Do đó, khi biến đổi tín hiệu dư, bộ mã hóa và bộ giải mã theo một phương án của sáng chế có thể cải thiện hiệu quả mã hóa bằng cách sử dụng nhân biến đổi được chọn cho

mỗi vùng trong số các nhân biến đổi khả dụng.

Theo chuẩn mã hóa video hiệu quả cao (high efficiency video coding, HEVC) hiện có, tín hiệu dư được biến đổi nhờ sử dụng nhân biến đổi dựa trên DCT-II trong số các kiểu phép biến đổi được mô tả ở trên, và phép biến đổi tín hiệu dư được thực hiện bằng cách sử dụng một cách giới hạn nhân biến đổi dựa trên DST-VII chỉ cho khối được dự đoán nội ảnh có kích thước 4X4. Như được mô tả ở trên, DCT-II có thể thích hợp để biến đổi tín hiệu dư của khối được dự đoán liên ảnh, nhưng có thể không thích hợp để biến đổi tín hiệu dư của khối được dự đoán nội ảnh. Nghĩa là, theo phương pháp dự đoán khối hiện thời, mẫu hình của tín hiệu dư trong khối dư có thể khác đi.

Theo đó, ít nhất một nhân biến đổi được chọn trong số các nhân biến đổi khả dụng có thể được sử dụng để biến đổi khối biến đổi cụ thể. Theo một phương án, bộ mã hóa và bộ giải mã có thể chọn nhân biến đổi cho ít nhất một vùng biến đổi của khối hiện thời, dựa trên phương pháp dự đoán khối hiện thời. Phương pháp này có thể được gọi là phép biến đổi đa lõi thích ứng (AMT) được mô tả ở trên. Nếu nhân biến đổi được xác định theo phương pháp dự đoán cho vùng cụ thể được sử dụng để biến đổi vùng tương ứng, hiệu quả mã hóa có thể được cải thiện so với trường hợp trong đó phép biến đổi được thực hiện cho tất cả các vùng bằng cách sử dụng nhân biến đổi dựa trên DCT-II.

Fig.12 là hình vẽ minh họa phương pháp tạo cấu hình bộ ứng viên biến đổi theo chế độ dự đoán của khối hiện thời. Theo một phương án của sáng chế, bộ mã hóa và bộ giải mã có thể chọn nhân biến đổi được sử dụng cho ít nhất một vùng biến đổi của khối hiện thời, dựa trên bộ ứng viên biến đổi tương ứng với khối hiện thời. Bộ ứng viên biến đổi có thể bao gồm các nhân biến đổi ứng viên. Bộ ứng viên biến đổi có thể bao gồm các nhân biến đổi ứng viên khác nhau theo chế độ dự đoán của khối hiện thời.

Theo một phương án, khối hiện thời có thể là khối được dự đoán dựa trên một trong số các chế độ dự đoán nội ảnh. Ở đây, mẫu hình của tín hiệu dư của khối hiện thời có thể khác đi theo chế độ dự đoán nội ảnh được sử dụng để dự đoán khối hiện thời. Như được mô tả ở trên, thông tin chế độ dự đoán nội ảnh có thể biểu thị hướng dự đoán nội ảnh. Theo đó, mẫu hình của tín hiệu dư của khối hiện thời có thể khác đi theo hướng dự đoán được

biểu thị bởi thông tin chế độ dự đoán nội ảnh của khối hiện thời. Bộ mã hóa và bộ giải mã có thể cải thiện hiệu quả mã hóa bằng cách sử dụng nhiều bộ ứng viên biến đổi khác nhau theo hướng dự đoán.

Theo đó, bộ mã hóa và bộ giải mã có thể xác định nhân biến đổi cần được sử dụng ở vùng biến đổi của khối hiện thời từ bộ ứng viên biến đổi tương ứng với chế độ dự đoán nội ảnh của khối hiện thời. Ví dụ, nhân biến đổi cần được sử dụng ở vùng biến đổi có thể được báo hiệu bằng chỉ số ứng viên biểu thị một trong số các nhân biến đổi ứng viên được bao gồm trong bộ ứng viên biến đổi. Khi bộ ứng viên biến đổi tương ứng với chế độ dự đoán nội ảnh của khối hiện thời bao gồm hai nhân biến đổi ứng viên, thì chỉ số ứng viên biểu thị nhân biến đổi được sử dụng ở vùng biến đổi có thể được biểu diễn dưới dạng 1 bit.

Hơn nữa, các bộ ứng viên biến đổi khác nhau có thể được sử dụng cho mỗi trong số các hướng dọc và ngang của vùng biến đổi. Do các đặc điểm của phương pháp dự đoán nội ảnh được thực hiện dựa trên tính định hướng, nên mẫu hình hướng dọc và mẫu hình hướng ngang của tín hiệu dư của vùng được dự đoán có thể khác nhau. Do đó, nhân biến đổi dọc và nhân biến đổi ngang dùng cho khối được dự đoán sử dụng chế độ dự đoán nội ảnh cụ thể có thể được thu từ bộ ứng viên biến đổi riêng lẻ. Theo sáng chế, nhân biến đổi được sử dụng để biến đổi theo hướng dọc vùng biến đổi cụ thể có thể được gọi là nhân biến đổi dọc. Nhân biến đổi được sử dụng để biến đổi theo hướng ngang vùng biến đổi cụ thể có thể được gọi là nhân biến đổi ngang. Fig.12(a) minh họa các bộ ứng viên biến đổi lần lượt tương ứng với 67 chỉ số chế độ dự đoán nội ảnh. Trên Fig.12(a), dọc (V) thể hiện bộ ứng viên biến đổi dùng cho nhân biến đổi dọc. Ngang (H) thể hiện bộ ứng viên biến đổi dùng cho nhân biến đổi ngang.

Fig.12(b) minh họa nhân biến đổi ứng viên được xác định theo bộ ứng viên biến đổi. Theo một phương án, bộ ứng viên biến đổi thứ nhất (ví dụ, bộ phép biến đổi 0 trên Fig.12B(b)) có thể bao gồm nhân biến đổi ứng viên dựa trên DST-VII và nhân biến đổi ứng viên dựa trên DCT-VIII. Bộ ứng viên biến đổi thứ hai (ví dụ, bộ phép biến đổi 1 trên Fig.12(b)) có thể bao gồm nhân biến đổi ứng viên dựa trên DST-VII và nhân biến đổi ứng viên dựa trên DST-I. Bộ ứng viên biến đổi thứ ba (ví dụ, bộ phép biến đổi 2 trên Fig.12(b))

có thể bao gồm nhân biến đổi ứng viên dựa trên DST-VII và nhân biến đổi ứng viên dựa trên DCT-V. Fig.12 minh họa rằng bộ ứng viên biến đổi bao gồm hai nhân biến đổi ứng viên, nhưng sáng chế không bị giới hạn ở đó.

Theo một phương án khác, khối hiện thời có thể là khối được dự đoán liên ảnh. Ở đây, nhân biến đổi được sử dụng ở vùng biến đổi của khối hiện thời có thể được thu từ bộ ứng viên biến đổi được tạo cấu hình trước. Ví dụ, bộ ứng viên biến đổi được tạo cấu hình trước có thể là một trong số các bộ ứng viên biến đổi được mô tả ở trên. Fig.12(c) minh họa bộ ứng viên biến đổi tương ứng với khối hiện thời nếu khối hiện thời là khối dự đoán liên ảnh. Ví dụ, bộ ứng viên biến đổi được tạo cấu hình trước tương ứng với khối dự đoán liên ảnh có thể bao gồm nhân biến đổi ứng viên dựa trên DCT-VIII và nhân biến đổi ứng viên dựa trên DST-VII. Nghĩa là, nhân biến đổi được sử dụng cho vùng biến đổi của khối dự đoán liên ảnh có thể là một trong số nhân biến đổi ứng viên dựa trên DCT-VIII và nhân biến đổi ứng viên dựa trên DST-VII.

Theo một phương án khác, các phương án được mô tả ở trên có thể được áp dụng một cách giới hạn chỉ cho khối biến đổi của thành phần độ sáng. Ở đây, nhân biến đổi mặc định được tạo cấu hình trước có thể được sử dụng cho khối biến đổi của thành phần sắc độ. Ví dụ, nhân biến đổi mặc định có thể là nhân biến đổi dựa trên DCT-II.

Theo một phương án khác, bộ mã hóa có thể không báo hiệu chỉ số ứng viên theo số lượng hệ số biến đổi khác không ở vùng biến đổi. Ở đây, bộ giải mã có thể xác định nhân biến đổi bằng cách so sánh số lượng hệ số biến đổi khác không ở vùng biến đổi với số lượng được tạo cấu hình trước. Ví dụ, nếu số lượng hệ số biến đổi khác không là một hoặc hai, thì chỉ số ứng viên có thể không được báo hiệu. Ở đây, bộ phận biến đổi ngược có thể thực hiện phép biến đổi của vùng tương ứng bằng cách sử dụng nhân biến đổi được tạo cấu hình trước. Nhân biến đổi được tạo cấu hình trước có thể là nhân biến đổi dựa trên DST-VII.

Fig.13 là hình vẽ minh họa nhân biến đổi ngang và nhân biến đổi dọc được xác định theo chỉ số bộ. Theo một phương án của sáng chế, thông tin biểu thị nhân biến đổi dùng cho tín hiệu dư của vùng cụ thể trong số các nhân biến đổi khả dụng có thể được báo hiệu.

Ở đây, thông tin biểu thị một trong số nhiều nhân biến đổi khả dụng có thể được báo hiệu bất kể chế độ dự đoán tương ứng với vùng tương ứng. Ví dụ, bộ mã hóa có thể báo hiệu thông tin nhân biến đổi biểu thị nhân biến đổi dùng cho tín hiệu dư của vùng cụ thể. Bộ giải mã có thể thu nhân biến đổi đối với vùng tương ứng bằng cách sử dụng thông tin nhân biến đổi được báo hiệu. Thông tin nhân biến đổi có thể bao gồm ít nhất một trong số thông tin biểu thị nhân biến đổi ngang của vùng tương ứng và thông tin biểu thị nhân biến đổi dọc của vùng tương ứng. Phương pháp này có thể được gọi là lựa chọn đa biến đổi (MTS) được mô tả ở trên.

Nhân biến đổi có thể được biểu thị bởi một trong số các chỉ số nhân lần lượt biểu thị các nhân biến đổi khả dụng. Theo một phương án, nhân biến đổi tương ứng với chỉ số nhân “0” có thể là nhân biến đổi dựa trên DCT-II, nhân biến đổi tương ứng với chỉ số nhân “1” có thể là nhân biến đổi dựa trên DST-VII, và nhân biến đổi tương ứng với chỉ số nhân “2” có thể là nhân biến đổi dựa trên DCT-VIII.

Theo một phương án, thông tin nhân biến đổi có thể là chỉ số bộ biểu thị bộ nhân biến đổi. Bộ nhân biến đổi có thể biểu diễn tổ hợp của nhân biến đổi cần được sử dụng làm nhân biến đổi dọc và nhân biến đổi cần được sử dụng làm nhân biến đổi ngang. Chỉ số bộ có thể biểu thị tổ hợp của chỉ số nhân biểu thị nhân biến đổi ngang và chỉ số nhân biểu thị nhân biến đổi dọc. Dựa vào Fig.13(a), khi thông tin nhân biến đổi của vùng cụ thể biểu thị chỉ số bộ biểu thị bộ nhân biến đổi thứ nhất (ví dụ, trong trường hợp của Fig.13(a) trong đó $tu_mts_idx[x0][y0]$ là 0), thì nhân biến đổi dọc của vùng tương ứng có thể là nhân biến đổi tương ứng với chỉ số nhân ‘0’. Nhân biến đổi ngang của vùng có thể là nhân biến đổi tương ứng với chỉ số nhân “0”.

Hơn nữa, Fig.13(b) minh họa nhân biến đổi tương ứng với chỉ số nhân. Dựa vào Fig.13(b), nhân biến đổi tương ứng với chỉ số nhân thứ nhất “0” có thể biểu thị nhân biến đổi dựa trên DCT-II. Dựa vào Fig.13(b), nhân biến đổi tương ứng với chỉ số nhân thứ hai “1” có thể biểu thị nhân biến đổi dựa trên DST-VII. Dựa vào Fig.13(b), nhân biến đổi tương ứng với chỉ số nhân thứ ba “2” có thể biểu thị nhân biến đổi dựa trên DCT-VIII.

Theo một phương án khác, các phương án được mô tả ở trên có thể được áp dụng

một cách giới hạn chỉ cho khối độ sáng. Ở đây, nhân biến đổi mặc định được tạo cấu hình trước có thể được sử dụng cho thành phần sắc độ. Ví dụ, nhân biến đổi mặc định có thể là nhân biến đổi dựa trên DCT-II. Chỉ số bộ được mô tả ở trên có thể được báo hiệu theo các đơn vị của các khối biến đổi.

Hơn nữa, thông tin biểu thị xem chỉ số bộ có được báo hiệu hay không có thể được báo hiệu bằng phần đầu của mức cao bao gồm khối hiện thời. Ở đây, mức cao có thể biểu thị phần chia/tám, ảnh hoặc chuỗi, mà bao gồm khối hiện thời. Thông tin biểu thị xem chỉ số bộ có được báo hiệu hay không có thể được báo hiệu bằng cờ riêng lẻ đối với mỗi phương pháp dự đoán. Ví dụ, khi xác định nhân biến đổi của khối, cờ biểu thị xem chỉ số bộ có được sử dụng hay không có thể được tạo cấu hình độc lập đối với mỗi trong số khối được dự đoán nội ảnh và khối được dự đoán liên ảnh.

Theo một phương án, nếu thông tin, mà biểu thị xem chỉ số bộ của mức cao bao gồm khối biến đổi được báo hiệu, biểu thị rằng chỉ số bộ không được báo hiệu một cách rõ ràng, nhân biến đổi liên quan đến khối biến đổi tương ứng có thể được xác định nhờ sử dụng phương pháp khác. Ví dụ, nhân biến đổi cần được áp dụng cho phép biến đổi ngược của khối biến đổi hiện thời có thể được xác định dựa trên thông tin được suy ra từ thông tin khác với chỉ số bộ. Cụ thể là, bộ giải mã có thể suy ra thông tin biểu thị nhân biến đổi dùng cho khối biến đổi hiện thời, từ thông tin được báo hiệu liên quan đến khối biến đổi hiện thời. Nghĩa là, thông tin biểu thị nhân biến đổi dọc và thông tin biểu thị nhân biến đổi ngang, mà cần được sử dụng trong khối biến đổi có thể được báo hiệu ngầm. Theo một phương án khác, nếu thông tin, mà biểu thị xem chỉ số bộ của mức cao bao gồm khối biến đổi có được báo hiệu hay không, biểu thị rằng chỉ số bộ được báo hiệu một cách rõ ràng, thì bộ giải mã có thể thu nhân biến đổi cho khối biến đổi dựa trên chỉ số bộ được báo hiệu.

Theo một phương án khác, việc xem bộ mã hóa có báo hiệu chỉ số bộ hay không có thể được xác định theo số lượng hệ số biến đổi khác không ở vùng biến đổi. Ở đây, bộ giải mã có thể xác định nhân biến đổi bằng cách so sánh số lượng hệ số biến đổi khác không ở vùng biến đổi với số lượng được tạo cấu hình trước. Ví dụ, nếu số lượng hệ số biến đổi khác không là hai hoặc nhỏ hơn hai, thì chỉ số bộ có thể không được báo hiệu. Ở đây, bộ

phận biến đổi ngược có thể thực hiện phép biến đổi của vùng tương ứng bằng cách sử dụng nhân biến đổi được tạo cấu hình trước. Nhân biến đổi được tạo cấu hình trước có thể là nhân biến đổi dựa trên DST-VII.

Điểm ảnh của hình ảnh đầu vào trong hệ thống nén hình ảnh có thể có ba giá trị bao gồm đỏ (R), xanh lá (G) và xanh dương (B) (không gian màu sắc RGB), và mỗi trong số các kênh R, G và B có thể được mã hóa độc lập. Tuy nhiên, sự mã hóa này không hiệu quả đối với hình ảnh tự nhiên được thu bởi camera kỹ thuật số. Đó là vì mỗi kênh trong không gian màu sắc RGB có độ tương quan cao giữa chúng, và, nói chung, để nén một cách hiệu quả hình ảnh RGB, hình ảnh RGB có thể được biến đổi thành không gian màu sắc khác để mã hóa. Phép biến đổi giữa các không gian màu sắc như được mô tả ở trên có thể được gọi là phép biến đổi không gian màu sắc hoặc phép biến đổi màu sắc, và phép biến đổi không gian màu sắc thường được sử dụng nhất để nén hình ảnh RGB có thể là phép biến đổi từ không gian màu sắc RGB sang không gian màu sắc YCbCr. Trong trường hợp không gian màu sắc RGB, tất cả các kênh R, G và B biểu diễn các thành phần màu sắc, nhưng, trong không gian màu sắc YCbCr, Y biểu diễn thành phần độ sáng, và Cb và Cr biểu diễn các thành phần sắc độ. Hệ thống thị giác của con người nhạy cảm hơn với thành phần độ sáng so với thành phần sắc độ, và do đó, trong trường hợp lấy mẫu con thành phần sắc độ, và lượng tử hóa các thành phần độ sáng và sắc độ bằng cách áp dụng kích thước bậc lượng tử hóa lớn hơn cho thành phần sắc độ so với thành phần độ sáng, hình ảnh màu sắc có thể được nén hiệu quả. Theo mức độ lấy mẫu con của thành phần sắc độ, các hình ảnh YCbCr 4:4:4, 4:2:2, và 4:2:0 có thể được phân biệt. Phép biến đổi từ không gian màu sắc RGB sang không gian màu sắc YCbCr được xem là hữu dụng xét theo sự nén hình ảnh, nhưng gây ra sự mất mát thông tin trong quy trình biến đổi màu sắc, và do đó khó được sử dụng trong lĩnh vực ứng dụng sự nén hình ảnh không mất mát hoặc sự nén hình ảnh gần như không mất mát. Xét theo sự mất mát thông tin, cách tốt nhất để mã hóa hình ảnh RGB là mã hóa trực tiếp hình ảnh gốc RGB mà không biến đổi không gian màu sắc. Tuy nhiên, như được mô tả ở trên, sự mã hóa này có thể không đạt được hiệu quả nén cao do sự dư thừa tồn tại trong không gian màu sắc RGB.

Phép biến đổi màu sắc thích ứng (ACT, phép biến đổi không gian màu sắc thích

ứng, phép biến đổi màu sắc thích ứng) thực hiện một cách thích ứng phép biến đổi không gian màu sắc theo các đặc điểm của tín hiệu hình ảnh, sao cho sự dư thừa tồn tại giữa các kênh có thể được loại bỏ, và hiệu quả mã hóa có thể được cải thiện. ACT có thể được áp dụng cho tín hiệu dư sau khi dự đoán bởi bộ mã hóa, và có thể biến đổi một cách có lựa chọn không gian màu sắc của tín hiệu dư. Thông tin không gian màu sắc được chọn có thể được bao gồm trong dòng bit và sau đó được truyền, và bộ giải mã có thể biến đổi ngược không gian màu sắc của tín hiệu dư được thu thông qua phép biến đổi ngược, dựa trên thông tin không gian màu sắc nhận được. Sau đây, các quy trình ACT và ACT ngược được mô tả chi tiết dựa vào Fig.14 và Fig.15.

Fig.14 là sơ đồ minh họa chi tiết phương pháp biến đổi tín hiệu dư thành hệ số biến đổi bởi bộ mã hóa thực hiện ACT. Tín hiệu dư ở vùng không gian có thể được biến đổi sang vùng tần số, và bộ mã hóa có thể biến đổi tín hiệu dư ở vùng không gian, mà được thu sau khi dự đoán, để thu hệ số biến đổi. Đầu tiên, bộ mã hóa có thể thu ít nhất một khối dư bao gồm tín hiệu dư của khối hiện thời. Khối dư có thể là một trong số khối hiện thời hoặc các khối được thu bằng cách phân chia khối hiện thời. Theo sáng chế, khối dư có thể được gọi là mảng dư hoặc ma trận dư bao gồm các mẫu dư của khối hiện thời. Ngoài ra, theo sáng chế, khối dư có thể biểu thị khối có cùng kích thước với khối biến đổi hoặc đơn vị biến đổi.

Tiếp theo, bộ mã hóa có thể thực hiện phép biến đổi màu sắc thích ứng. Phép biến đổi màu sắc thích ứng có thể được áp dụng cho tín hiệu dư, và tín hiệu dư trong không gian màu sắc thứ nhất có thể được biến đổi thành tín hiệu dư trong không gian màu sắc thứ hai. Không gian màu sắc thứ nhất có thể được tạo cấu hình bằng nhiều kênh (hoặc thành phần) màu sắc bao gồm kênh thứ nhất, kênh thứ hai, kênh thứ ba và dạng tương tự, và có thể là không gian màu sắc chẳng hạn như GBR hoặc YCbCr. Trong không gian màu sắc GBR, kênh thứ nhất biểu thị màu xanh lá (G), kênh thứ hai biểu thị màu xanh dương (B) và kênh thứ ba biểu thị màu đỏ (R). Trong không gian màu sắc YCbCr, kênh thứ nhất biểu thị Y (độ sáng), kênh thứ hai biểu thị Cb (sắc độ) và kênh thứ ba biểu thị Cr (sắc độ). Không gian màu sắc thứ hai có thể được tạo cấu hình bằng nhiều kênh (hoặc thành phần) màu sắc bao gồm kênh thứ nhất, kênh thứ hai, kênh thứ ba và dạng tương tự, và có thể là không

gian màu sắc chẳng hạn như YCgCo hoặc YCbCr. Trong không gian màu sắc YCgCo, kênh thứ nhất biểu thị Y (độ sáng), kênh thứ hai biểu thị Cg (sắc độ) và kênh thứ ba biểu thị Co (sắc độ), và phép biến đổi (phép biến đổi xuôi) từ không gian màu sắc thứ nhất sang không gian màu sắc YCgCo mà là không gian màu sắc thứ hai có thể được xác định theo phương trình sau.

Phương trình 1

$$C'_0 = (2 \cdot C_0 + C_1 + C_2) / 4$$

$$C'_1 = (2 \cdot C_0 - C_1 - C_2) / 4$$

$$C'_2 = (-2 \cdot C_1 + 2 \cdot C_2) / 4$$

Trong phương trình ở trên, C_0 , C_1 và C_2 lần lượt biểu thị kênh thứ nhất, kênh thứ hai và kênh thứ ba của không gian màu sắc thứ nhất, và ví dụ, có thể là các kênh G, B và R trong không gian màu sắc GBR hoặc các kênh Y, Cb và Cr trong không gian màu sắc YCbCr. C'_0 , C'_1 và C'_2 lần lượt biểu thị Y, là kênh thứ nhất trong không gian màu sắc YCgCo mà là không gian màu sắc thứ hai, Cg, là kênh thứ hai và Co, là kênh thứ ba. Phép biến đổi ngược từ không gian màu sắc YCgCo mà là không gian màu sắc thứ hai sang không gian màu sắc thứ nhất có thể được xác định theo phương trình sau.

Phương trình 2

$$C_0 = (C'_0 + C'_1)$$

$$C_1 = (C'_0 - C'_1 - C'_2)$$

$$C_2 = (C'_0 - C'_1 + C'_2)$$

Không gian màu sắc tốt nhất trong sự nén hình ảnh là không gian màu sắc có các kênh có các đặc tính không tương quan giữa chúng. Do đó, ma trận tương quan giữa các thành phần màu sắc của tập dữ liệu hình ảnh màu sắc đủ được tính toán, và phép biến đổi Karhunen-Loeve (Karhunen-Loeve transform, KLT) được áp dụng cho nó, sao cho sự tương quan cực tiểu hóa phương trình biến đổi có thể được thu. Phép biến đổi sang không gian màu sắc YCgCo có thể làm giảm một cách hiệu quả sự tương quan tồn tại giữa các kênh của không gian màu sắc RGB (hoặc GBR), và phép biến đổi ngược được mô tả ở trên

được thực hiện chỉ bằng cách thêm các hoạt động.

Bộ mã hóa có thể lựa chọn một cách thích ứng không gian màu sắc xét theo chi phí tỷ lệ biến dạng, và thông tin về không gian màu sắc có thể được bao gồm trong dòng bit và sau đó được truyền. Ví dụ, ACT có thể được áp dụng ở mức đơn vị mã hóa, và cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) có thể được báo hiệu. Khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) là 1, thì phần dư của đơn vị mã hóa hiện thời đã được mã hóa sang không gian màu sắc YCgCo, và khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) là 0, thì phần dư của đơn vị mã hóa hiện thời đã được mã hóa sang không gian màu sắc gốc, và điều này có thể biểu thị rằng ACT không được áp dụng.

Tiếp theo, bộ mã hóa có thể biến đổi khối dư bằng cách sử dụng nhân biến đổi. Nhân biến đổi được sử dụng trong phép biến đổi khối dư có thể là nhân biến đổi mà có thể tách riêng thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Trong trường hợp này, phép biến đổi đối với khối dư có thể được tách thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang, và sau đó được thực hiện. Ví dụ, bộ mã hóa có thể áp dụng nhân biến đổi theo hướng dọc đối với khối dư để thực hiện phép biến đổi theo hướng dọc. Ngoài ra, bộ mã hóa có thể áp dụng nhân biến đổi theo hướng ngang đối với khối dư để thực hiện phép biến đổi theo hướng ngang. Theo sáng chế, nhân biến đổi có thể được sử dụng làm thuật ngữ biểu thị bộ thông số được sử dụng để biến đổi tín hiệu dư, chẳng hạn như ma trận biến đổi, mảng biến đổi, hàm biến đổi và phép biến đổi. Theo một phương án, nhân biến đổi có thể là một trong số các nhân khả dụng. Ngoài ra, các nhân biến đổi dựa trên các loại phép biến đổi khác nhau có thể lần lượt được sử dụng cho phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Theo phương pháp lựa chọn nhân biến đổi được mô tả ở trên dựa vào Fig.12 và Fig.13, một trong số các nhân biến đổi khả dụng có thể được chọn.

Bộ mã hóa có thể truyền khối biến đổi được thu bằng cách biến đổi khối dư thành đơn vị lượng tử hóa, và lượng tử hóa đơn vị này. Khối biến đổi có thể bao gồm nhiều hệ số biến đổi. Cụ thể là, khối biến đổi có thể bao gồm nhiều hệ số biến đổi được bố trí theo

hai chiều. Tương tự với khối dư, khối biến đổi có thể có cùng kích thước với một trong số khối hiện thời hoặc khối được thu bằng cách phân chia khối hiện thời. Các hệ số biến đổi được truyền đến bộ phận lượng tử hóa có thể được biểu diễn bởi giá trị được lượng tử hóa.

Ngoài ra, trước khi các hệ số biến đổi được lượng tử hóa, bộ mã hóa có thể thực hiện phép biến đổi khác. Như được minh họa trên Fig.14, phương pháp biến đổi được mô tả ở trên có thể được gọi là phép biến đổi sơ cấp, và phép biến đổi khác có thể được gọi là phép biến đổi thứ cấp. Phép biến đổi thứ cấp có thể chọn lọc đối với mỗi khối dư. Theo một phương án, bộ mã hóa có thể thực hiện phép biến đổi thứ cấp đến vùng mà khó tập trung năng lượng ở vùng tần số thấp chỉ với phép biến đổi sơ cấp, để cải thiện hiệu quả mã hóa. Ví dụ, phép biến đổi thứ cấp có thể được thêm vào khối có các giá trị dư lớn theo các hướng khác với hướng ngang hoặc hướng dọc của khối dư. Các giá trị dư của khối được dự đoán nội ảnh có thể có xác suất thay đổi cao hơn theo các hướng khác với hướng ngang hoặc hướng dọc, so với các giá trị dư của khối được dự đoán liên ảnh. Theo đó, bộ mã hóa có thể còn thực hiện phép biến đổi thứ cấp đối với tín hiệu dư của khối được dự đoán nội ảnh. Ngoài ra, bộ mã hóa có thể lược bỏ phép biến đổi thứ cấp đối với tín hiệu dư của khối được dự đoán liên ảnh.

Là một ví dụ khác, việc có thực hiện phép biến đổi thứ cấp hay không có thể được xác định theo kích thước của khối hiện thời hoặc khối dư. Ngoài ra, các nhân biến đổi với các kích thước khác nhau có thể được sử dụng theo kích thước của khối hiện thời hoặc khối dư. Ví dụ, phép biến đổi thứ cấp 8X8 có thể được áp dụng cho khối trong đó chiều dài của cạnh ngắn hơn trong số chiều rộng hoặc chiều cao nhỏ hơn chiều dài được tạo cấu hình trước thứ nhất. Ngoài ra, phép biến đổi thứ cấp 4X4 có thể được áp dụng cho khối trong đó chiều dài của cạnh ngắn hơn trong số chiều rộng hoặc chiều cao lớn hơn chiều dài được tạo cấu hình trước thứ hai. Chiều dài được tạo cấu hình trước thứ nhất có thể lớn hơn chiều dài được tạo cấu hình trước thứ hai, nhưng sáng chế không bị giới hạn ở đó. Ngoài ra, không giống như phép biến đổi sơ cấp, phép biến đổi thứ cấp có thể không được tách thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang. Phép biến đổi thứ cấp này có thể được gọi là phép biến đổi không thể tách riêng tần số thấp (LFNST).

Ngoài ra, trong trường hợp tín hiệu video ở vùng cụ thể, do sự thay đổi độ sáng nhanh chóng, năng lượng dải tần số cao có thể không được làm giảm thậm chí khi phép biến đổi tần số được thực hiện. Theo đó, hiệu suất nén do lượng tử hóa có thể bị suy giảm. Ngoài ra, khi phép biến đổi được thực hiện đối với vùng trong đó các giá trị dư gần như không tồn tại, thời gian mã hóa và thời gian giải mã có thể bị tăng lên một cách không cần thiết. Theo đó, phép biến đổi đối với tín hiệu dư ở vùng cụ thể có thể được lược bỏ. Việc liệu có thực hiện phép biến đổi đối với tín hiệu dư ở vùng cụ thể hay không có thể được xác định bởi phần tử cú pháp liên quan đến phép biến đổi của vùng cụ thể. Ví dụ, phần tử cú pháp có thể bao gồm thông tin bỏ qua phép biến đổi. Thông tin bỏ qua phép biến đổi có thể là cờ bỏ qua phép biến đổi. Khi thông tin bỏ qua phép biến đổi liên quan đến khối dư biểu thị sự bỏ qua phép biến đổi, thì phép biến đổi đối với khối dư không được thực hiện. Trong trường hợp này, bộ mã hóa có thể ngay lập tức lượng tử hóa tín hiệu dư mà không được biến đổi ở vùng tương ứng. Các hoạt động của bộ mã hóa được mô tả dựa vào Fig.14 có thể được thực hiện bởi bộ phận biến đổi của Fig.1.

Các phần tử cú pháp liên quan đến phép biến đổi được mô tả ở trên có thể là thông tin được phân tích từ dòng bit tín hiệu video. Bộ giải mã có thể thu các phần tử cú pháp liên quan đến phép biến đổi bằng cách giải mã entropy dòng bit tín hiệu video. Ngoài ra, bộ mã hóa có thể mã hóa entropy các phần tử cú pháp liên quan đến phép biến đổi để tạo ra dòng bit tín hiệu video.

Fig.15 là sơ đồ minh họa chi tiết phương pháp biến đổi ngược hệ số biến đổi để thu tín hiệu dư bởi bộ giải mã thực hiện phép biến đổi màu sắc thích ứng. Sau đây, nhằm thuận tiện để giải thích, hoạt động biến đổi ngược sẽ được mô tả là được thực hiện bởi bộ phận biến đổi ngược của mỗi trong số bộ mã hóa và bộ giải mã. Bộ phận biến đổi ngược có thể thu tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi được lượng tử hóa ngược. Đầu tiên, bộ phận biến đổi ngược có thể phát hiện, từ phần tử cú pháp liên quan đến phép biến đổi của vùng cụ thể, xem phép biến đổi ngược đối với vùng tương ứng có được thực hiện hay không. Theo một phương án, khi phần tử cú pháp liên quan đến phép biến đổi liên quan đến khối biến đổi cụ thể biểu thị sự bỏ qua phép biến đổi, thì phép biến đổi của khối biến đổi tương ứng có thể được lược bỏ. Trong trường hợp này, phép biến đổi ngược sơ

cấp và phép biến đổi ngược thứ cấp như được mô tả ở trên đối với khối biến đổi có thể đều được lược bỏ. Trong trường hợp này, hệ số biến đổi được lượng tử hóa ngược có thể được sử dụng làm tín hiệu dư, hoặc tín hiệu dư có thể được thu bằng cách thực hiện phép biến đổi màu sắc thích ứng ngược (ACT ngược) đối với hệ số biến đổi được lượng tử hóa ngược. Bộ phận biến đổi ngược có thể khôi phục khối hiện thời bằng cách sử dụng tín hiệu dư thu được.

Theo phương án khác, phần tử cú pháp liên quan đến phép biến đổi liên quan đến khối biến đổi cụ thể có thể không biểu thị sự bỏ qua phép biến đổi. Trong trường hợp này, bộ phận biến đổi ngược có thể xác định xem có thực hiện phép biến đổi thứ cấp ngược liên quan đến phép biến đổi thứ cấp hay không. Ví dụ, khi khối biến đổi là khối biến đổi của khối được dự đoán nội ảnh, phép biến đổi thứ cấp ngược của khối biến đổi có thể được thực hiện. Ngoài ra, nhân biến đổi thứ cấp được sử dụng cho khối biến đổi có thể được xác định dựa trên chế độ dự đoán nội ảnh tương ứng với khối biến đổi. Là ví dụ khác, việc có thực hiện phép biến đổi thứ cấp ngược hay không có thể được xác định theo kích thước của khối biến đổi. Phép biến đổi thứ cấp ngược có thể được thực hiện trước phép biến đổi sơ cấp ngược được thực hiện sau quy trình lượng tử hóa ngược.

Bộ phận biến đổi ngược có thể thực hiện phép biến đổi sơ cấp ngược đối với hệ số biến đổi được lượng tử hóa ngược hoặc hệ số biến đổi mà phép biến đổi thứ cấp ngược đã được thực hiện. Tương tự với phép biến đổi sơ cấp, phép biến đổi sơ cấp ngược có thể được tách thành phép biến đổi theo hướng dọc và phép biến đổi theo hướng ngang để được thực hiện. Ví dụ, bộ phận biến đổi ngược có thể thu khối dư bằng cách thực hiện phép biến đổi ngược theo hướng dọc và phép biến đổi ngược theo hướng ngang đối với khối biến đổi. Bộ phận biến đổi ngược có thể biến đổi ngược khối biến đổi, dựa trên nhân biến đổi mà đã được sử dụng để biến đổi khối biến đổi. Ví dụ, bộ mã hóa có thể báo hiệu một cách rõ ràng hoặc ngầm thông tin biểu thị nhân biến đổi được áp dụng cho khối biến đổi hiện thời trong số các nhân biến đổi khả dụng. Bộ giải mã có thể chọn nhân biến đổi cần được sử dụng cho phép biến đổi ngược đối với khối biến đổi trong số các nhân biến đổi khả dụng bằng cách sử dụng thông tin được báo hiệu biểu thị nhân biến đổi. Bộ phận biến đổi ngược có thể thực hiện phép biến đổi ngược đối với hệ số biến đổi, hoặc thực hiện phép biến đổi màu

sắc thích ứng ngược đối với tín hiệu được thu bằng cách thực hiện phép biến đổi ngược, để thu tín hiệu dư. Bộ phận biến đổi ngược có thể khôi phục khối hiện thời bằng cách sử dụng tín hiệu dư.

Bộ phận biến đổi ngược có thể thực hiện phép biến đổi màu sắc thích ứng ngược, dựa trên phần tử cú pháp liên quan đến phép biến đổi. Phép biến đổi màu sắc thích ứng ngược có thể được áp dụng cho tín hiệu mà phép biến đổi sơ cấp ngược đã được thực hiện, hoặc tín hiệu mà sự bỏ qua phép biến đổi đã được áp dụng, và có thể thực hiện phép biến đổi từ không gian màu sắc thứ hai sang không gian màu sắc thứ nhất. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ dòng bit, cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không. Cụ thể là, khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) là 1, thì điều này biểu thị rằng phần dư của khối hiện thời đã được mã hóa sang không gian màu sắc YCgCo, và do đó như được thể hiện trong phương trình 2, phép biến đổi ngược từ không gian màu sắc YCgCo, không gian màu sắc thứ hai, sang không gian màu sắc thứ nhất có thể được thực hiện. Khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) là 0, thì điều này biểu thị rằng phần dư của khối hiện thời đã được mã hóa sang không gian màu sắc gốc, và điều này có thể biểu thị rằng phép biến đổi màu sắc thích ứng ngược không được thực hiện. Cụ thể là, phép biến đổi màu sắc thích ứng ngược được thể hiện trong phương trình 2 có thể bao gồm quy trình như sau.

Quy trình biến đổi không gian màu sắc

- Đầu vào xử lý:
- $nTbW$: Chiều rộng của khối biến đổi
- $nTbH$: Chiều cao của khối biến đổi
- rY : rY có thể là mảng mẫu dư của kênh thứ nhất của không gian màu sắc thứ hai, và có thể là mảng có kích thước $(nTbW) \times (nTbH)$. Không gian màu sắc thứ hai có thể là không gian màu sắc YCgCo, và kênh thứ nhất có thể biểu thị độ sáng Y. Phần tử của mảng rY có thể được biểu diễn bởi $rY[x][y]$.

- rCb: rCb có thể là mảng mẫu dư của kênh thứ hai của không gian màu sắc thứ hai, và có thể là mảng có kích thước $(nTbW) \times (nTbH)$. Không gian màu sắc thứ hai có thể là không gian màu sắc YCgCo, và kênh thứ hai có thể biểu thị sắc độ Cg. Phần tử của mảng rCb có thể được biểu diễn bởi $rCb[x][y]$.

- rCr: rCr có thể là mảng mẫu dư của kênh thứ ba của không gian màu sắc thứ hai, và có thể là mảng có kích thước $(nTbW) \times (nTbH)$. Không gian màu sắc thứ hai có thể là không gian màu sắc YCgCo, và kênh thứ ba có thể biểu thị sắc độ Co. Phần tử của mảng rCr có thể được biểu diễn bởi $rCr[x][y]$.

- mảng rY, rCb và rCr có thể biểu thị một trong số tín hiệu sau khi bỏ qua phép biến đổi được áp dụng, hoặc tín hiệu sau khi phép biến đổi sơ cấp ngược được áp dụng.

- Xử lý đầu ra:

- Mảng mẫu dư rY có kích thước $(nTbW) \times (nTbH)$ được cải biến: theo quy trình của sáng chế, không gian màu sắc thứ hai có thể được biến đổi sang không gian màu sắc thứ nhất, và mảng mẫu dư của kênh thứ nhất của không gian màu sắc thứ nhất có thể được biểu thị. Phần tử của mảng rY có thể được biểu diễn bởi $rY[x][y]$.

- Mảng mẫu dư rCb có kích thước $(nTbW) \times (nTbH)$ được cải biến: theo quy trình của sáng chế, không gian màu sắc thứ hai có thể được biến đổi sang không gian màu sắc thứ nhất, và mảng mẫu dư của kênh thứ hai của không gian màu sắc thứ nhất có thể được biểu thị. Phần tử của mảng rCb có thể được biểu diễn bởi $rCb[x][y]$.

- Mảng mẫu dư rCr có kích thước $(nTbW) \times (nTbH)$ được cải biến: theo quy trình của sáng chế, không gian màu sắc thứ hai có thể được biến đổi sang không gian màu sắc thứ nhất, và mảng mẫu dư của kênh thứ ba của không gian màu sắc thứ nhất có thể được biểu thị. Phần tử của mảng rCr có thể được biểu diễn bởi $rCr[x][y]$.

- rY, rCb và rCr mà là các mảng mẫu dư, mỗi mảng này có kích thước $(nTbW) \times (nTbH)$ có thể được cải biến theo phương trình 3 đến phương trình 6 dưới đây:

Phương trình 3

$$tmp = rY[x][y] - rCb[x][y]$$

Phương trình 4

$$rY[x][y] = rY[x][y] + rCb[x][y]$$

Phương trình 5

$$rCb[x][y] = tmp - rCr[x][y]$$

Phương trình 6

$$rCr[x][y] = tmp + rCr[x][y]$$

Theo quy trình không gian màu sắc, rY, rCb và rCr mà là các mảng mẫu dư của không gian màu sắc thứ hai có thể được cải biến thành rY, rCb, rCr mà là các mảng mẫu dư của không gian màu sắc thứ nhất. Bộ phận biến đổi ngược có thể khôi phục khối hiện thời bằng cách sử dụng rY, rCb và rCr mà là các mảng mẫu dư được cải biến.

Trường hợp trong đó không gian màu sắc thứ hai là không gian màu sắc YCgCo đã được mô tả liên quan đến ACT và ACT ngược được mô tả dựa vào Fig.14 và Fig.15, nhưng sáng chế không bị giới hạn ở đó. Là ví dụ khác của không gian màu sắc thứ hai, không gian màu sắc YCgCo-R có thể được sử dụng. không gian màu sắc YCgCo-R có thể được suy ra từ không gian màu sắc YCgCo bằng cách sử dụng thuật toán nâng để cho phép khôi phục hoàn hảo chỉ với các phép toán số nguyên. Phép biến đổi xuôi từ không gian màu sắc thứ nhất sang không gian màu sắc YCgCo-R có thể được xác định theo phương trình sau.

Phương trình 7

$$C'_2 = C_2 - C_1$$

$$t = C_1 + (C'_2 \gg 1)$$

$$C'_1 = C_0 - t$$

$$C'_0 = t + (C'_1 \gg 1)$$

Trong phương trình ở trên, C₀, C₁ và C₂ lần lượt biểu thị kênh thứ nhất, kênh thứ hai và kênh thứ ba của không gian màu sắc thứ nhất, và ví dụ, có thể là các kênh G, B và R trong không gian màu sắc GBR hoặc các kênh Y, Cb và Cr trong không gian màu sắc YCbCr. C'₀, C'₁ và C'₂ lần lượt biểu thị Y, là kênh thứ nhất trong không gian màu sắc

YCgCo-R mà là không gian màu sắc thứ hai, Cg, là kênh thứ hai và Co, là kênh thứ ba. Phép biến đổi ngược từ không gian màu sắc YCgCo-R mà là không gian màu sắc thứ hai sang không gian màu sắc thứ nhất có thể được xác định theo phương trình sau.

Phương trình 8

$$t = C'_0 - (C'_1 \gg 1)$$

$$C_0 = C'_1 + t$$

$$C_1 = t - (C'_2 \gg 1)$$

$$C_2 = C_1 + C'_2$$

Là ví dụ khác của không gian màu sắc thứ hai, một trong số các không gian màu sắc YCgCo và YCgCo-R có thể được chọn một cách thích ứng theo chế độ mã hóa hiện thời. Ví dụ, khi chế độ mã hóa của khối hiện thời là chế độ nén không mất mát hoặc gần như không mất mát, thì không gian màu sắc YCgCo-R có thể được sử dụng làm không gian màu sắc thứ hai. Trong khi đó, khi chế độ mã hóa của khối hiện thời là chế độ nén mất mát, thì không gian màu sắc YCgCo có thể được sử dụng làm không gian màu sắc thứ hai. Chế độ nén mất mát hoặc gần như không mất mát, và chế độ nén mất mát có thể được báo hiệu ở mức đơn vị mã hóa bởi cờ, hoặc có thể được biểu thị bằng cách sử dụng thông số lượng tử hóa (quantization parameter, QP) và cờ bỏ qua phép biến đổi.

Trong khi đó, phép biến đổi sang không gian màu sắc YCgCo được thể hiện trong phương trình 1 không được chuẩn hóa. Trong $Y = [2/4, 1/4, 1/4] * [C_0, C_1, C_2]'$, l2-norm của vector $[2/4, 1/4, 1/4]$ bằng $\sqrt{6}/4$. Trong $C_g = [2/4, -1/4, -1/4] * [C_0, C_1, C_2]'$, l2-norm của vector $[2/4, -1/4, -1/4]$ bằng $\sqrt{6}/4$. Trong $C_o = [0, -2, 2] * [C_0, C_1, C_2]$, l2-norm của vector $[0, -2, 2]$ bằng $\sqrt{8}/4$. Nghĩa là, sự thay đổi về phạm vi động xảy ra trước và sau phép biến đổi không gian màu sắc, và do đó giá trị QP có thể được hiệu chỉnh để bù cho sự thay đổi này. Sự hiệu chỉnh có thể là dựa trên độ lệch thông số lượng tử hóa (qpActOffset). Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset), dựa trên thành phần màu sắc của khối biến đổi được bao gồm trong khối hiện thời và cờ kích hoạt phép biến đổi không gian màu sắc

(cu_act_enabled_flag). Thành phần màu sắc của khối hiện thời có thể bao gồm ít nhất một trong số kênh thứ nhất, kênh thứ hai và kênh thứ ba. Độ lệch thông số lượng tử hóa (qpActOffset) có thể bao gồm ít nhất một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai và độ lệch QP thứ ba.

Thiết bị xử lý tín hiệu video có thể xác định độ lệch thông số lượng tử hóa (qpActOffset), dựa trên thành phần màu sắc của khối hiện thời. Khi thông số lượng tử hóa (QP) được áp dụng cho kênh thứ nhất của không gian màu sắc thứ nhất là QP_0, thì QP'_0 mà là thông số lượng tử hóa (QP) được áp dụng cho kênh thứ nhất của không gian màu sắc thứ hai có thể được xác định bởi $QP_0 + \text{Offset}_0$, và Offset_0 biểu thị độ lệch QP thứ nhất. Độ lệch QP thứ nhất có thể được bao gồm trong cú pháp mức cao hơn, chẳng hạn như SPS, PPS, phần đầu ảnh hoặc phần đầu phần chia, và sau đó được truyền, và giá trị cố định được xác định trước bởi bộ mã hóa và bộ giải mã có thể được sử dụng làm độ lệch QP thứ nhất. Độ lệch QP thứ nhất có thể là -5.

Khi QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ nhất là QP_1, QP'_1 mà là QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ hai có thể được xác định bởi $QP_1 + \text{Offset}_1$, và Offset_1 biểu thị độ lệch QP thứ hai. Độ lệch QP thứ hai có thể được bao gồm trong cú pháp mức cao hơn, chẳng hạn như SPS, PPS, phần đầu ảnh hoặc phần đầu phần chia, và sau đó được truyền, và giá trị cố định được xác định trước bởi bộ mã hóa và bộ giải mã có thể được sử dụng làm độ lệch QP thứ hai. Độ lệch QP thứ hai có thể là -5.

Khi QP được áp dụng cho kênh thứ ba của không gian màu sắc thứ nhất là QP_2, QP'_2 mà là QP được áp dụng cho kênh thứ ba của không gian màu sắc thứ hai có thể được xác định bởi $QP_2 + \text{Offset}_2$, và Offset_2 biểu thị độ lệch QP thứ ba. Độ lệch QP thứ ba có thể được bao gồm trong cú pháp mức cao hơn, chẳng hạn như SPS, PPS, phần đầu ảnh hoặc phần đầu phần chia, và sau đó được truyền, và giá trị cố định được xác định trước bởi bộ mã hóa và bộ giải mã có thể được sử dụng làm độ lệch QP thứ ba. Độ lệch QP thứ ba có thể là -3.

Ngoài ra, thiết bị xử lý tín hiệu video có thể xác định độ lệch thông số lượng tử hóa

(qpActOffset), dựa trên cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag). Khi cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset) là 0.

Ngoài ra, khi cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa là độ lệch định trước, dựa trên thành phần màu sắc của khối biến đổi. Sau đây, phần mô tả về điều này sẽ được đưa ra một cách chi tiết hơn.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ nhất, và cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset) là độ lệch QP thứ nhất (Offset_0) định trước. Ngoài ra, khi thông số lượng tử hóa (QP) được áp dụng cho kênh thứ nhất của không gian màu sắc thứ nhất là QP_0, QP'_0 mà là thông số lượng tử hóa (QP) được áp dụng cho kênh thứ nhất của không gian màu sắc thứ hai có thể được xác định bởi $QP_0 + Offset_0$.

Khi thành phần màu sắc của khối hiện thời được biểu thị không phải là thành phần kênh thứ nhất, TuCResMode biểu thị chế độ mã hóa nối đối với các phần dư sắc độ (joint coding của chroma residuals, JCCR) biểu thị 2, và cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset) là độ lệch QP thứ hai (Offset_1) định trước. Ngoài ra, khi QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ nhất là QP_1, thì QP'_1 mà là QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ hai có thể được xác định bởi $QP_1 + Offset_1$. Chế độ JCCR có thể tương ứng với phương pháp mã hóa mà nhờ đó các tín hiệu dư của kênh thứ hai và kênh thứ ba được biểu diễn bởi một tín

hiệu.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ hai, và cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (`qpActOffset`) là độ lệch QP thứ hai (`Offset_1`) định trước. Ngoài ra, khi QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ nhất là QP_1 , thì QP'_1 mà là QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ hai có thể được xác định bởi $QP_1 + Offset_1$.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ ba, và cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (`qpActOffset`) là độ lệch QP thứ ba (`Offset_2`) định trước. Ngoài ra, khi QP được áp dụng cho kênh thứ ba của không gian màu sắc thứ nhất là QP_2 , thì QP'_2 mà là QP được áp dụng cho kênh thứ ba của không gian màu sắc thứ hai có thể được xác định bởi $QP_2 + Offset_2$.

Trong khi đó, QP được sử dụng cho phép lượng tử hóa và giải lượng tử hóa có thể được xác định trong phạm vi số nguyên được tạo cấu hình trước, và kích thước bậc lượng tử hóa được sử dụng trong phép lượng tử hóa có thể được xác định theo giá trị QP. Ví dụ, QP có thể được xác định trong phạm vi số nguyên trong đó số nguyên này không âm, và giá trị QP tăng có thể biểu thị kích thước bậc lượng tử hóa tăng. Kết quả, điều này có nghĩa là sai số lượng tử hóa có thể tăng. Khi ACT được áp dụng cho khối hiện thời, giá trị QP có thể được hiệu chỉnh bởi độ lệch QP thứ nhất (`Offset_0`), độ lệch QP thứ hai (`Offset_1`), và độ lệch QP thứ ba (`Offset_2`), và QP'_0 , QP'_1 và QP'_2 mà là các QP được hiệu chỉnh được sử dụng trong phép lượng tử hóa và giải lượng tử hóa thực tế có thể lệch khỏi phạm vi số nguyên được tạo cấu hình trước trong đó QP được xác định. Cụ thể là, `Offset_0`, `Offset_1` và `Offset_2` có thể là các số âm, và do đó QP'_0 , QP'_1 và QP'_2 có thể là các số âm. Các giá trị QP lệch khỏi phạm vi được xác định, và do đó sau khi hiệu chỉnh giá trị QP bằng độ lệch, phép toán cắt có thể được thực hiện để giá trị QP nằm trong phạm vi

được tạo cấu hình trước trong đó giá trị QP được xác định. Ví dụ, khi QP được xác định trong phạm vi $[0, 63]$, và Offset_0, Offset_1 và Offset_2 lần lượt là -5, -5 và -3, thì quy trình cắt có thể được thực hiện để ngăn không cho giá trị QP được hiệu chỉnh là số âm. QP'_0 mà là QP được áp dụng cho kênh thứ nhất của không gian màu sắc thứ hai có thể được xác định bởi $\text{Max}(0, \text{QP}_0 + \text{Offset}_0)$. Phép toán $\text{Max}(x, y)$ là hàm trả về x khi $x \geq y$, và trả về y khi $x < y$, và do đó khi giá trị của $\text{QP}_0 + \text{Offset}_0$ là số âm, QP'_0 có thể được tạo cấu hình là 0 và sau đó được sử dụng trong phép lượng tử hóa và giải lượng tử hóa. QP'_1 mà là QP được áp dụng cho kênh thứ hai của không gian màu sắc thứ hai có thể được xác định bởi $\text{Max}(0, \text{QP}_1 + \text{Offset}_1)$. Theo phép toán max, khi giá trị của $\text{QP}_1 + \text{Offset}_1$ là số âm, thì QP'_1 có thể được tạo cấu hình là 0 và sau đó được sử dụng trong phép lượng tử hóa và giải lượng tử hóa. QP'_2 mà là QP được áp dụng cho kênh thứ ba của không gian màu sắc thứ hai có thể được xác định bởi $\text{Max}(0, \text{QP}_2 + \text{Offset}_2)$. Theo phép toán max, khi giá trị của $\text{QP}_2 + \text{Offset}_2$ là số âm, thì QP'_2 có thể được tạo cấu hình là 0 và sau đó được sử dụng trong phép lượng tử hóa và giải lượng tử hóa.

Dựa vào các hình vẽ từ Fig.16 đến Fig.19 của sáng chế được mô tả dưới đây, phương pháp hiệu chỉnh giá trị QP được mô tả.

Fig.16 minh họa phương pháp hiệu chỉnh giá trị QP, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên QP được hiệu chỉnh theo phương pháp thứ nhất của sáng chế. Quy trình thay đổi tỷ lệ có thể nhận đầu vào của (xTbY, yTbY), nTbW, nTbH, predMode và cIdx, và phân mô tả cho mỗi biến là như sau.

- (xTbY, yTbY): Biến này là vị trí của độ sáng, có thể biểu thị các tọa độ trên cùng bên trái của khối biến đổi độ sáng hiện thời, và có thể là các tọa độ dựa trên mẫu độ sáng trên cùng bên trái của ảnh hiện thời.

- nTbW: Chiều rộng của khối biến đổi

- nTbH: Chiều cao của khối biến đổi

- predMode: biến này biểu thị phương pháp dự đoán đơn vị mã hóa hiện thời, và có thể biểu thị phép dự đoán nội ảnh khi biến là MODE_INTRA, biểu thị phép dự đoán liên ảnh khi biến là MODE_INTER, và biểu thị phép dự đoán IBC (phương pháp dự đoán để

tạo ra khối dự đoán từ khối tham chiếu được biểu thị bởi khối vector trong ảnh hiện thời) khi biến là `MODE_IBC`.

- `cIdx`: biến này biểu thị thành phần màu sắc của khối biến đổi hiện thời, và có thể biểu thị thành phần kênh thứ nhất khi `cIdx` là 0, biểu thị thành phần kênh thứ hai khi `cIdx` là 1 và biểu thị thành phần kênh thứ ba khi `cIdx` là 2. Khi ACT đã được áp dụng cho khối hiện thời, hệ số biến đổi hiện thời có thể biểu thị sự biểu diễn trong không gian màu sắc thứ hai, và không gian màu sắc thứ hai có thể là không gian màu sắc YCgCo. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời, thì hệ số biến đổi hiện thời biểu thị sự biểu diễn trong không gian màu sắc thứ nhất, mà là, không gian màu sắc gốc.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động khôi phục khối hiện thời, dựa trên thông số lượng tử hóa. Thông số lượng tử hóa có thể được hiệu chỉnh. Cụ thể là, trong quy trình thay đổi tỷ lệ, việc thay đổi tỷ lệ (giải lượng tử hóa) có thể được thực hiện đối với hệ số biến đổi được lượng tử hóa để thu hệ số biến đổi được thay đổi tỷ lệ (được giải lượng tử hóa). Phép biến đổi ngược có thể được thực hiện đối với hệ số biến đổi được thay đổi tỷ lệ để thu tín hiệu dư. Bộ phận biến đổi ngược có thể khôi phục khối hiện thời bằng cách sử dụng tín hiệu dư thu được. Mảng hệ số biến đổi được thay đổi tỷ lệ $d[x][y]$, mà là đầu ra của quy trình thay đổi tỷ lệ, có thể là mảng có kích thước $(nTbW) \times (nTbH)$.

Trong quy trình thay đổi tỷ lệ, thông số lượng tử hóa (biến `qP`) và biến độ lệch thông số lượng tử hóa (`qpActOffset`) cần thiết để thay đổi tỷ lệ có thể được suy ra đầu tiên như sau.

- Trường hợp trong đó biến `cIdx` là 0:

- $qP = Qp'Y$: khi `cIdx` là 0, thì điều này biểu thị thành phần kênh thứ nhất, và do đó biến `qP` mà là thông số lượng tử hóa được sử dụng để thay đổi tỷ lệ thực có thể được tạo cấu hình là $Qp'Y$ mà là thông số lượng tử hóa được áp dụng cho thành phần kênh thứ nhất. $Qp'Y$ có thể được xác định ở mức đơn vị mã hóa, và có thể được xác định dựa trên phần tử cú pháp liên quan đến phép biến đổi. Ví dụ, nhóm lượng tử hóa hình chữ nhật được xác định trong khối cây mã hóa, và các đơn vị mã hóa tồn tại trong nhóm lượng tử hóa có thể chia sẻ giá trị dự đoán của cùng các thông số lượng tử hóa. Giá trị dự đoán của thông số

lượng tử hóa có thể được suy ra từ thông số lượng tử hóa của khối lân cận với nhóm lượng tử hóa. Ví dụ, khi các tọa độ trên cùng bên trái của nhóm lượng tử hóa hình chữ nhật là (x_{Qg}, y_{Qg}) , thì giá trị trung bình giữa thông số lượng tử hóa của vị trí $(x_{Qg}-1, y_{Qg})$ và thông số lượng tử hóa của vị trí $(x_{Qg}, y_{Qg}-1)$ có thể được sử dụng làm giá trị dự đoán của thông số lượng tử hóa. Để làm giảm lượng bit cần thiết để báo hiệu thông số lượng tử hóa, giá trị khác biệt giữa giá trị dự đoán của thông số lượng tử hóa và thông số lượng tử hóa thực tế có thể được bao gồm trong dòng bit ở mức đơn vị mã hóa để truyền dẫn, và bộ giải mã có thể xác định $Qp'Y$, dựa trên giá trị dự đoán của thông số lượng tử hóa và giá trị khác biệt được phân tích từ dòng bit.

- $qpActOffset = -5$: khi ACT được áp dụng, phạm vi động thay đổi, và do đó để bù cho sự thay đổi, độ lệch có thể được thêm vào thông số lượng tử hóa để hiệu chỉnh thông số lượng tử hóa. Việc áp dụng ACT có thể biểu thị trường hợp trong đó cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời. Cụ thể là, khi thành phần màu sắc của khối hiện thời biểu thị thành phần kênh thứ nhất, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là độ lệch QP thứ nhất ($Offset_0$) định trước. Độ lệch QP thứ nhất được áp dụng cho thành phần kênh thứ nhất có thể là -5.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ nhất, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là 0.

- Trường hợp trong đó biến $cIdx$ không phải là 0, và biến $TuCResMode[xTbY][yTbY]$ là 2: theo chế độ mã hóa nối đối với các phần dư sắc độ (JCCR), hai tín hiệu dư sắc độ có thể được biểu diễn bởi một mảng dư bằng cách sử dụng sự tương quan tồn tại giữa các thành phần sắc độ. Khi $TuCResMode[xTbY][yTbY]$ là 2,

thì điều này có thể biểu thị rằng một trong số các chế độ JCCR được áp dụng, và khi $cIdx$ không phải là 0, thì điều này có thể biểu thị khối biến đổi hiện thời là khối hệ số biến đổi biểu diễn kênh thứ hai và kênh thứ ba là một tín hiệu dư. Khi mảng dư được thu bởi bộ phận biến đổi ngược của bộ giải mã là $C[x][y]$, mảng dư của kênh thứ hai là $Cb[x][y]$ và mảng dư của kênh thứ ba là $Cr[x][y]$, thì $Cb[x][y] = C[x][y]$ và $Cr[x][y] = CSign * C[x][y]$ có thể được xác định. Biến $CSign$ là thông tin ký hiệu được áp dụng cho kênh thứ ba, và có thể được bao gồm trong phần đầu phần chia để truyền dẫn.

- $qP = Qp'CbCr$: khi $cIdx$ không phải là 0, và $TuCResMode[xTbY][yTbY]$ là 2, qP mà là thông số lượng tử hóa được áp dụng trong quy trình thay đổi tỷ lệ có thể được tạo cấu hình là $Qp'CbCr$ mà là thông số lượng tử hóa được áp dụng khi $TuCResMode[xTbY][yTbY]$ là 2. Tương tự với $Qp'Y$, $Qp'CbCr$ có thể được xác định ở mức đơn vị mã hóa, và có thể được xác định dựa trên phân tử cú pháp liên quan đến phép biến đổi.

- $qpActOffset = -5$: khi ACT được áp dụng, phạm vi động thay đổi, và do đó để bù cho sự thay đổi, độ lệch có thể được thêm vào thông số lượng tử hóa để hiệu chỉnh thông số lượng tử hóa. Việc áp dụng ACT có thể biểu thị trường hợp trong đó cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời. Cụ thể hơn là, khi thành phần màu sắc của khối hiện thời được biểu thị không phải là thành phần kênh thứ nhất, $TuCResMode$ biểu thị chế độ mã hóa nối đối với các phần dư sắc độ (JCCR) là 2, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là độ lệch QP thứ hai ($Offset_1$) định trước. Độ lệch QP được áp dụng khi $TuCResMode[xTbY][yTbY]$ trong số các chế độ JCCR là 2 có thể là -5.

Khi thành phần màu sắc của khối hiện thời được biểu thị không phải là thành phần kênh thứ nhất, $TuCResMode$ biểu thị chế độ mã hóa nối đối với các phần dư sắc độ (JCCR) biểu thị 2, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu

thì rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là 0.

- Trường hợp trong đó biến $TuCResMode[xTbY][yTbY]$ không phải là 2, và biến $cIdx$ là 1: trường hợp này có thể bao gồm trường hợp trong đó biến $TuCResMode[xTbY][yTbY]$ là 0, và biến $cIdx$ là 1, và trường hợp trong đó biến $TuCResMode[xTbY][yTbY]$ là 1, và biến $cIdx$ là 1. Trường hợp trước có thể biểu thị rằng các mảng dư của kênh thứ hai và kênh thứ ba được mã hóa độc lập, và $cIdx$ là 1, và do đó điều này biểu thị rằng khối biến đổi hiện thời là mảng dư của kênh thứ hai. Trường hợp sau có thể biểu thị rằng khối biến đổi hiện thời là khối hệ số biến đổi biểu diễn kênh thứ hai và kênh thứ ba là một tín hiệu dư bằng một trong số các chế độ JCCR. Khi mảng dư được thu bởi bộ phận biến đổi ngược của bộ giải mã là $C[x][y]$, mảng dư của kênh thứ hai là $Cb[x][y]$ và mảng dư của kênh thứ ba là $Cr[x][y]$, thì $Cb[x][y] = C[x][y]$, và $Cr[x][y] = (CSign * C[x][y]) \gg 1$ có thể được xác định. Biến $CSign$ là thông tin ký hiệu được áp dụng cho kênh thứ ba, và có thể được bao gồm trong phần đầu phần chia để truyền dẫn.

- $qP = Qp'Cb$: trong cả hai trường hợp, qP mà là thông số lượng tử hóa được áp dụng cho việc thay đổi tỷ lệ có thể được tạo cấu hình là $Qp'Cb$ mà là thông số lượng tử hóa được áp dụng cho kênh thứ hai. Tương tự với $Qp'Y$, $Qp'Cb$ có thể được xác định ở mức đơn vị mã hóa, và có thể được xác định dựa trên phần tử cú pháp liên quan đến phép biến đổi.

- $qpActOffset = -5$: khi ACT được áp dụng, phạm vi động thay đổi, và do đó để bù cho sự thay đổi, độ lệch có thể được thêm vào thông số lượng tử hóa để hiệu chỉnh thông số lượng tử hóa. Việc áp dụng ACT có thể biểu thị trường hợp trong đó cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời. Cụ thể hơn là, khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ hai, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác

định độ lệch thông số lượng tử hóa ($qpActOffset$) là độ lệch QP thứ hai ($Offset_1$) định trước. Độ lệch QP thứ hai được áp dụng cho kênh thứ hai có thể là -5.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ hai, và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là 0.

- Trường hợp trong đó biến $cIdx$ là 2: trường hợp này có thể bao gồm trường hợp trong đó biến $TuCResMode[xTbY][yTbY]$ là 0, và biến $cIdx$ là 2, và trường hợp trong đó biến $TuCResMode[xTbY][yTbY]$ là 3, và biến $cIdx$ là 2. Trường hợp trước có thể biểu thị rằng các mảng dư của kênh thứ hai và kênh thứ ba được mã hóa độc lập, và $cIdx$ là 2, và do đó điều này biểu thị rằng khối biến đổi hiện thời là mảng dư của kênh thứ ba. Trường hợp sau có thể biểu thị rằng khối biến đổi hiện thời là khối hệ số biến đổi biểu diễn kênh thứ hai và kênh thứ ba là một tín hiệu dư bằng một trong số các chế độ JCCR. Khi mảng dư được thu bởi bộ phận biến đổi ngược của bộ giải mã là $C[x][y]$, mảng dư của kênh thứ hai là $Cb[x][y]$ và mảng dư của kênh thứ ba là $Cr[x][y]$, thì $Cb[x][y] = (CSign * C[x][y]) >> 1$, và $Cr[x][y] = C[x][y]$ có thể được xác định. Biến $CSign$ là thông tin ký hiệu được áp dụng cho kênh thứ hai, và có thể được bao gồm trong phần đầu phần chia để truyền dẫn.

- $qP = Qp'Cr$: trong cả hai trường hợp, qP mà là thông số lượng tử hóa được áp dụng cho việc thay đổi tỷ lệ có thể được tạo cấu hình là $Qp'Cr$ mà là thông số lượng tử hóa được áp dụng cho kênh thứ ba. Tương tự với $Qp'Y$, $Qp'Cr$ có thể được xác định ở mức đơn vị mã hóa, và có thể được xác định dựa trên phần tử cú pháp liên quan đến phép biến đổi.

- $qpActOffset = -3$: khi ACT được áp dụng, phạm vi động thay đổi, và do đó để bù cho sự thay đổi, độ lệch có thể được thêm vào thông số lượng tử hóa để hiệu chỉnh thông số lượng tử hóa. Việc áp dụng ACT có thể biểu thị trường hợp trong đó cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời. Khi thành phần màu sắc của khối hiện thời biểu thị thành phần kênh thứ ba, và cờ kích hoạt phép biến đổi không gian màu sắc

(cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset) là độ lệch QP thứ ba (Offset_2) định trước. Độ lệch QP thứ ba được áp dụng cho kênh thứ ba có thể là -3.

Khi thành phần màu sắc của khối hiện thời được biểu thị là thành phần kênh thứ ba, và cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa (qpActOffset) là 0.

Thông số lượng tử hóa có thể được cải biến dựa trên thông số lượng tử hóa (qP) và độ lệch thông số lượng tử hóa (qpActOffset) được mô tả ở trên, và biến rectNonTsFlag và biến bdShift cần thiết cho quy trình thay đổi tỷ lệ có thể được tạo cấu hình như sau.

- Trường hợp trong đó sự bỏ qua phép biến đổi không được áp dụng (transform_skip_flag[xTbY][yTbY][cIdx] là 0): phần tử cú pháp transform_skip_flag[xTbY][yTbY][cIdx] mà là cờ bỏ qua phép biến đổi có thể biểu thị xem sự bỏ qua phép biến đổi có được áp dụng cho khối biến đổi hiện thời hay không. Khối biến đổi hiện thời có thể là khối biến đổi được bao gồm trong khối hiện thời. Khi transform_skip_flag[xTbY][yTbY][cIdx] là 0, thì điều này có thể biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi hiện thời. Điều này có thể ngụ ý rằng mẫu dư được thu bằng cách thực hiện phép biến đổi ngược đối với hệ số biến đổi được thay đổi tỷ lệ (được giải lượng tử hóa).

- $qP = \text{Max}(0, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? qpActOffset : 0))$; Theo phương pháp thứ nhất của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để bù cho sự thay đổi về phạm vi động, và giá trị qP được hiệu chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ nhất, dựa trên việc thông số lượng tử hóa được xác định trong vùng nguyên trong đó số nguyên này không âm, giá trị nhỏ hơn 0 có thể được cắt thành 0 để cho phép giá trị qP được hiệu chỉnh lớn hơn hoặc bằng 0.

Trong công thức trên, phép toán bậc ba $x^y:z$ có nghĩa là phép toán trả về y khi x là đúng hoặc không phải là 0, và trả về z khi x là sai hoặc là 0. Do đó, khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ nhất của sáng chế, việc cắt dựa trên toán tử $\max$ có thể được thực hiện để ngăn không cho giá trị qP được hiệu chỉnh là số âm. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh.

- $rectNonTsFlag = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1) ? 1 : 0$; nhân biến đổi dựa trên sin hoặc cosin có thể không có kiểu số nguyên như được mô tả dựa vào Fig.9, và nhân biến đổi có thể được làm xấp xỉ thành kiểu số nguyên để làm giảm độ phức tạp của phương án thực hiện. Để biểu diễn nhân biến đổi theo kiểu số nguyên, lũy thừa cơ số 2 (số mũ có thể không phải là số nguyên) có thể được nhân với nhân biến đổi và sau đó được làm tròn để thu nhân biến đổi số nguyên được làm xấp xỉ. Theo sai số xấp xỉ, các đặc tính trực chuẩn của nhân biến đổi có thể bị suy giảm. Tuy nhiên, sự suy giảm hiệu quả mã hóa do đó là nhỏ, và độ phức tạp của phương án thực hiện của sự mã hóa biến đổi dựa trên nhân biến đổi số nguyên trong bộ mã hóa và bộ giải mã phần cứng là thấp đáng kể, và do đó nhân biến đổi số nguyên bao gồm sai số xấp xỉ có thể được sử dụng. Nhân biến đổi được số nguyên hóa có tỷ lệ được phóng đại theo lũy thừa cơ số 2, và do đó sau phép biến đổi, thì cần có quy trình làm giảm tỷ lệ, và sự giảm tỷ lệ này có thể được thực hiện trong quy trình thay đổi tỷ lệ và quy trình biến đổi. Biến $rectNonTsFlag$ và $bdShift$ được mô tả dưới đây có thể được diễn dịch là thông số để việc giảm tỷ lệ được thực hiện trong quy trình thay đổi tỷ lệ. Khi kết quả của phép toán $(\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1$ trong $rectNonTsFlag$ là 1, thì hệ số biến đổi sau phép biến đổi ngược theo hướng ngang và phép biến đổi ngược theo hướng dọc có thể có tỷ lệ được tăng thêm $2^{(N+0,5)}$. Khi kết quả của phép toán $(\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1$ là 0, thì hệ số biến đổi sau phép biến đổi ngược theo

hướng ngang và phép biến đổi ngược theo hướng dọc có thể có tỷ lệ được tăng thêm 2^M , và N và M là các số nguyên dương. Nghĩa là, khi phép biến đổi ngược được thực hiện, để xem xét thừa số thay đổi tỷ lệ $\sqrt{2}$ theo chiều rộng và chiều cao của khối biến đổi, thì biến `rectNonTsFlag` có thể được tạo cấu hình là 1 trong trường hợp trong đó kết quả của phép toán $(\text{Log2}(\text{nTbW}) + \text{Log2}(\text{nTbH})) \& 1$ là 1 khi phép biến đổi ngược được thực hiện đối với khối biến đổi hiện thời (`transform_skip_flag[xTbY][yTbY][cIdx]` là 0).

- $\text{bdShift} = \text{BitDepth} + \text{rectNonTsFlag} + ((\text{Log2}(\text{nTbW}) + \text{Log2}(\text{nTbH})) / 2) - 5 + \text{pic_dep_quant_enabled_flag}$: biến `BitDepth` biểu thị độ sâu bit của các mẫu mảng độ sáng và sắc độ. Ví dụ, khi `BitDepth` là N (số tự nhiên), thì các mẫu mảng kênh thứ nhất, kênh thứ hai và kênh thứ ba có thể có các giá trị nằm trong phạm vi $[0, 2^N - 1]$. Biến `BitDepth` có thể được tạo cấu hình dựa trên phần tử cú pháp được bao gồm trong SPS, PPS, phần đầu ảnh, phần đầu phần chia, v.v.. Ví dụ, phần tử cú pháp `bit_depth_minus8` có thể được báo hiệu để báo hiệu thông tin độ sâu bit trong SPS. Bộ giải mã có thể tạo cấu hình biến `BitDepth` là $8 + \text{bit_depth_minus8}$, dựa trên phần tử cú pháp `bit_depth_minus8` được phân tích. `pic_dep_quant_enabled_flag` là phần tử cú pháp biểu thị xem phép lượng tử hóa phụ thuộc được kích hoạt hoặc hủy kích hoạt trong phần chia hiện thời, và khi `pic_dep_quant_enabled_flag` là 1, thì điều này có thể biểu thị rằng phép lượng tử hóa phụ thuộc được kích hoạt trong phần chia hiện thời. Biến `bdShift` là thông số để giảm tỷ lệ, và có thể được xác định theo độ sâu bit, việc có xem xét thừa số thay đổi tỷ lệ $\sqrt{2}$ hay không, kích thước của khối biến đổi, và việc phép lượng tử hóa phụ thuộc được kích hoạt hoặc hủy kích hoạt.

- $\text{bdOffset} = (1 \ll \text{bdShift}) >> 1$: trong quy trình thay đổi tỷ lệ được mô tả dưới đây, phép toán dịch bit có thể được thực hiện dựa trên thông số `bdShift`, và biến `bdOffset` có thể biểu thị độ lệch làm tròn của phép toán dịch bit.

- Trường hợp trong đó sự bỏ qua phép biến đổi được áp dụng (`transform_skip_flag[xTbY][yTbY][cIdx]` là 1): khi `transform_skip_flag[xTbY][yTbY][cIdx]` mà là cờ bỏ qua phép biến đổi là 1, thì điều này biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi hiện thời, và điều

này có thể có nghĩa là hệ số biến đổi được thay đổi tỷ lệ (được giải lượng tử hóa) được sử dụng làm phần dư.

- $qP = \text{Max}(0, \text{Max}(QpPrimeTsMin, qP) + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? qpActOffset : 0))$; Theo phương pháp thứ nhất của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để bù cho sự thay đổi về phạm vi động, và giá trị qP được hiệu chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ nhất, đầu tiên, thông qua phép so sánh giữa biến $QpPrimeTsMin$ và giá trị qP , việc cắt có thể được thực hiện để cho phép giá trị qP lớn hơn hoặc bằng $QpPrimeTsMin$. Biến $QpPrimeTsMin$ xác định thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, và có thể được bao gồm và được biểu thị trong cấu trúc cú pháp mức cao hơn chẳng hạn như SPS, PPS, phần đầu ảnh hoặc phần đầu phân chia. Thông tin ($\text{min_qp_prime_ts_minus4}$) liên quan đến thông số lượng tử hóa nhỏ nhất có thể tương ứng với phần tử cú pháp. Ví dụ, bộ giải mã có thể thu thông tin ($\text{min_qp_prime_ts_minus4}$) liên quan đến thông số lượng tử hóa nhỏ nhất trong SPS từ dòng bit. Tuy nhiên, sáng chế không bị giới hạn ở đó, và bộ giải mã có thể thu thông tin ($\text{min_qp_prime_ts_minus4}$) liên quan đến thông số lượng tử hóa nhỏ nhất từ một dòng bit trong số SPS, PPS, phần đầu ảnh hoặc phần đầu phân chia. Ngoài ra, bộ giải mã có thể thực hiện hoạt động thu thông số lượng tử hóa nhỏ nhất ($QpPrimeTsMin$), dựa trên thông tin ($\text{min_qp_prime_ts_minus4}$) liên quan đến thông số lượng tử hóa nhỏ nhất. Ví dụ, bộ giải mã có thể tạo cấu hình biến $QpPrimeTsMin$ thành $4 + \text{min_qp_prime_ts_minus4}$, dựa trên phần tử cú pháp $\text{min_qp_prime_ts_minus4}$. $\text{min_qp_prime_ts_minus4}$ có thể là thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động hiệu chỉnh thông số lượng tử hóa, dựa trên thông số lượng tử hóa nhỏ nhất. Ngoài ra, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh.

Theo phương pháp thứ nhất của sáng chế, thông số lượng tử hóa trước tiên có thể được cắt thành giá trị lớn hơn hoặc bằng $QpPrimeTsMin$ bởi biến $QpPrimeTsMin$, và sau

đó độ lệch thông số lượng tử hóa ($qpActOffset$) để bù cho sự thay đổi về phạm vi động do ACT có thể được thêm vào đó.

Như đã được mô tả ở trên, $qpActOffset$ có thể được xác định là 0 hoặc một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba, dựa trên $cu_act_enabled_flag[xTbY][yTbY]$. Tuy nhiên, sáng chế không bị giới hạn ở đó, và theo các phương án khác nhau của sáng chế, giá trị có thể được thêm vào thông số lượng tử hóa (qP) bởi quy trình như sau. Khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ nhất của sáng chế, việc cắt dựa trên toán tử max có thể được thực hiện để ngăn không cho giá trị qP được hiệu chỉnh là số âm. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Theo sự hiệu chỉnh qP được mô tả ở trên, khi ACT được áp dụng cho khối hiện thời, thì giá trị nhỏ nhất của qP có thể được tạo cấu hình là 0, và khi ACT không được áp dụng cho khối hiện thời, thì giá trị nhỏ nhất của qP có thể được tạo cấu hình là thông số lượng tử hóa nhỏ nhất ($QpPrimeTsMin$).

- $rectNonTsFlag=0$: $rectNonTsFlag$ là biến xem xét thừa số tỷ lệ tăng do nhân biến đổi, và do đó biến $rectNonTsFlag$ có thể được tạo cấu hình là 0 theo chế độ bỏ qua phép biến đổi trong đó phép biến đổi không được áp dụng.

- $bdShift=10$: theo chế độ bỏ qua phép biến đổi trong đó phép biến đổi và phép lượng tử hóa phụ thuộc không được áp dụng, biến $bdShift$ có thể được tạo cấu hình là 10 mà là giá trị cố định.

- $bdOffset=(1<<bdShift)>>1$: trong quy trình thay đổi tỷ lệ được mô tả dưới đây, phép toán dịch bit có thể được thực hiện dựa trên thông số $bdShift$, và biến $bdOffset$ có thể biểu thị độ lệch làm tròn của phép toán dịch bit.

Danh sách $\text{levelScale}[j][k]$ có thể biểu thị kích thước bậc lượng tử hóa được thay đổi tỷ lệ. Kích thước bậc lượng tử hóa về cơ bản thuộc kiểu số thực, và có thể được làm xấp xỉ thành kiểu số nguyên. Ngoài ra, khi qP được tăng thêm 6, thì kích thước bậc lượng tử hóa tăng gấp đôi, và do đó kích thước bậc lượng tử hóa được xác định cho qP từ 0 đến 5, và trong đoạn qP còn lại, kích thước bậc lượng tử hóa có thể được suy ra thông qua phép toán $qP\%6$ và phép toán $qP/6$.

Mảng dz có kích thước $(nTbW) \times (nTbH)$ có thể được tạo cấu hình giống với $\text{TransCoeffLevel}[xTbY][yTbY][cIdx]$ mà là mảng có kích thước $(nTbW) \times (nTbH)$, và $\text{TransCoeffLevel}[xTbY][yTbY][cIdx]$ biểu thị hệ số biến đổi được lượng tử hóa được thu khi mã hóa hệ số (mã hóa phần dư).

$d[x][y]$ mà là hệ số biến đổi được thay đổi tỷ lệ (x nằm trong khoảng từ 0 đến $nTbW-1$, và y nằm trong khoảng từ 0 đến $nTbH-1$) có thể được thu thông qua quy trình như sau.

Đầu tiên, quy trình tạo cấu hình $m[x][y]$ mà là mảng phần tử thay đổi tỷ lệ trung gian có thể được thực hiện. Các đặc điểm phản hồi tần số của hệ thống thị giác của con người không nhạy với các dải tần số cao, và tương đối nhạy với các dải tần số thấp, và do đó xét theo các đặc điểm này, các thừa số thay đổi tỷ lệ khác nhau có thể được tạo cấu hình theo x và y mà là các chỉ số bin tần số. Điều này biểu thị rằng giá trị của mảng phần tử $m[x][y]$ thay đổi theo x và y . Ví dụ, khi giá trị thay đổi tỷ lệ lớn được sử dụng cho vùng tần số cao và giá trị thay đổi tỷ lệ nhỏ được sử dụng cho vùng tần số thấp, phép lượng tử hóa có thể được thực hiện hiệu quả. Đó là vì cá nhân có khả năng không nhận ra được sai số lượng tử hóa tốt, sai số xuất hiện do sử dụng giá trị thay đổi tỷ lệ lớn cho vùng tần số cao. Trong khi đó, khi sự bỏ qua phép biến đổi được áp dụng, thì hệ số biến đổi được lượng tử hóa nhận được là tín hiệu ở vùng không gian thay vì là tín hiệu ở vùng tần số. Do đó, khi giá trị của mảng $m[x][y]$ được tạo cấu hình để thay đổi theo x và y , sự biến dạng trở nên lớn trong vùng không gian một phần, và do đó chất lượng hình ảnh có thể bị suy giảm. Do đó, giá trị của mảng phần tử $m[x][y]$ có thể được tạo cấu hình là hằng số không đổi bất kể x và y . Hằng số không đổi có thể là 16. Là ví dụ khác, khi phép biến đổi thứ cấp được áp dụng, thì giá trị của mảng phần tử $m[x][y]$ có thể được tạo cấu hình là hằng số không

đổi bất kể x và y . Hằng số không đổi có thể là 16. Phép biến đổi thứ cấp có thể được diễn dịch để ánh xạ hệ số biến đổi được biến đổi sơ cấp đến không gian khác, và các đặc điểm của hệ số được biến đổi thứ cấp khác hoàn toàn với các đặc điểm của hệ số được biến đổi sơ cấp. Ví dụ, khi phép biến đổi thứ cấp không được áp dụng, và chỉ phép biến đổi sơ cấp dựa trên nhân DCT-II được thực hiện, $d[0][0]$ có thể biểu thị hệ số DC. Trong khi đó, khi phép biến đổi thứ cấp được thực hiện đối với hệ số được biến đổi sơ cấp dựa trên nhân DCT-II, $d[0][0]$ không biểu thị hệ số DC. Để thu hệ số DC của hệ số được biến đổi sơ cấp, tích số chấm giữa vector hệ số được biến đổi thứ cấp và vector cơ sở thứ nhất của nhân được biến đổi thứ cấp ngược cần được thực hiện, và điều này có nghĩa là tất cả các hệ số được biến đổi thứ cấp ảnh hưởng đến vùng tần số thấp của hệ số được biến đổi sơ cấp. Do đó, xem xét các phần tử thay đổi tỷ lệ khác nhau so với hệ số biến đổi thứ cấp theo các chỉ số x và y có thể không chỉ làm suy giảm hiệu suất của bộ lượng tử hóa, mà còn gây ra sự suy giảm của chất lượng hình ảnh. Do đó, khi phép biến đổi thứ cấp được áp dụng cho khối hiện thời, thì $m[x][y]$ có thể được tạo cấu hình để có cùng giá trị đối với tất cả các chỉ số x và y . Ví dụ, $m[x][y]$ có thể là 16 bất kể các chỉ số x và y .

$ls[x][y]$ mà là mảng phần tử thay đổi tỷ lệ cuối cùng có thể được tạo cấu hình dựa trên $m[x][y]$ và $levelScale[j][k]$. Cụ thể là, khi phép lượng tử hóa phụ thuộc được áp dụng ($pic_dep_quant_enabled_flag$ là 1, và $transform_skip_flag[xTbY[yTbY][cIdx]$ là 0), thì $ls[x][y]$ có thể được tạo cấu hình bằng $(m[x][y]*levelScale[rectNonTsFlag][(qP+1)\%6])<<((qP+1)/6)$. $levelScale$ có tỷ lệ được tăng thêm $\sqrt{2}$ có thể được áp dụng khi $rectNonTsFlag$ là 1, so với trường hợp khi $rectNonTsFlag$ là 0. Khi phép dịch bit sang trái được thực hiện đối với kích thước bậc lượng tử hóa cơ sở ($levelScale[rectNonTsFlag][(qP+1)\%6]$) bằng $(qP+1)/6$ bằng cách sử dụng $(qP+1)\%6$ làm chỉ số, kích thước bậc lượng tử hóa cuối cùng có thể được xác định. Điều này phản ánh đặc điểm trong đó, trong trường hợp trong đó phép lượng tử hóa phụ thuộc được áp dụng, khi $(qP+1)$ được tăng thêm 6, thì kích thước bậc lượng tử hóa tăng gấp đôi. Trong khi đó, khi phép lượng tử hóa phụ thuộc không được áp dụng ($pic_dep_quant_enabled_flag$ là 0, hoặc $transform_skip_flag[xTbY[yTbY][cIdx]$ là 1), thì $ls[x][y]$ có thể được tạo cấu hình bằng

$(m[x][y] * levelScale[rectNonTsFlag][qP \% 6]) << (qP / 6)$. Khi phép dịch bit sang trái được thực hiện đối với kích thước bậc lượng tử hóa cơ sở ($levelScale[rectNonTsFlag][qP \% 6]$) bằng $qP / 6$ bằng cách sử dụng $qP \% 6$ làm chỉ số, thì kích thước bậc lượng tử hóa cuối cùng có thể được xác định. Điều này phản ánh đặc điểm trong đó, trong trường hợp trong đó phép lượng tử hóa phụ thuộc không được áp dụng, khi qP được tăng thêm 6, thì kích thước bậc lượng tử hóa tăng gấp đôi.

Trong khi đó, trong trường hợp phép dự đoán điều biến mã hóa xung delta dựa trên khối (block-based delta pulse code modulation, BDPCM), phép dự đoán nội ảnh được thực hiện theo chế độ được tạo cấu hình trước, và hệ số biến đổi được lượng tử hóa có thể được biểu diễn bởi tổng của mẫu ở vị trí trước đó và mẫu ở vị trí hiện thời theo chế độ được tạo cấu hình trước. Nghĩa là, theo chế độ BDPCM, do hệ số biến đổi được lượng tử hóa, nên sự khác biệt giữa mẫu ở vị trí trước đó và mẫu ở vị trí hiện thời theo chế độ được tạo cấu hình trước được truyền. Chế độ được tạo cấu hình trước có thể là chế độ ngang hoặc chế độ dọc. Khi BDPCM được áp dụng cho khối mã hóa hiện thời, thì $BdpcmFlag[xTbY][yTbY]$ có thể là 1, và $BdpcmDir[xTbY][yTbY]$ có thể biểu thị chế độ được tạo cấu hình trước.

Khi $BdpcmFlag[xTbY][yTbY]$ là 1, và $BdpcmDir[xTbY][yTbY]$ là 0, thì điều này có thể biểu thị rằng chế độ ngang BDPCM được áp dụng, và $dz[x][y]$ có thể được tạo cấu hình bằng $Clip3(CoeffMin, CoeffMax, dz[x-1][y] + dz[x][y])$. Toán tử $Clip3(x, y, z)$ là toán tử để cắt z để nằm giữa x và y , và khi $z < x$, thì $Clip3(x, y, z) = x$, khi $z > y$, thì $Clip3(x, y, z) = y$, và ngược lại, ($x \leq z \leq y$), thì $Clip3(x, y, z) = z$ có thể được biểu diễn. Hệ số biến đổi có thể được biểu diễn với độ chính xác bit được tạo cấu hình trước, và độ chính xác này có thể là 16 bit. Trong trường hợp này, $CoeffMin$ có thể được tạo cấu hình là $-2^{(16)}$, và $CoeffMax$ có thể được tạo cấu hình là $2^{(16)} - 1$. Khi $BdpcmFlag[xTbY][yTbY]$ là 1, và $BdpcmDir[xTbY][yTbY]$ là 1, thì điều này có thể biểu thị rằng chế độ dọc BDPCM được áp dụng, và $dz[x][y]$ có thể được tạo cấu hình bằng $Clip3(CoeffMin, CoeffMax, dz[x][y-1] + dz[x][y])$.

$dnc[x][y]$ có thể được tính toán bằng $(dz[x][y] * ls[x][y] + bdOffset) >> bdShift$, và

$d[x][y]$ mà là hệ số biến đổi được thay đổi tỷ lệ có thể được thu thông qua $Clip3(CoeffMin, CoeffMax, dnc[x][y])$.

Fig.17 minh họa phương pháp hiệu chỉnh giá trị QP, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên QP được hiệu chỉnh theo phương pháp thứ hai của sáng chế. Phần mô tả mà không được đưa ra dựa vào Fig.17 có thể tuân theo phần mô tả được đưa ra dựa vào Fig.16, và sau đây, phương pháp hiệu chỉnh thông số lượng tử hóa khi sự bỏ qua phép biến đổi không được áp dụng và khi sự bỏ qua phép biến đổi được áp dụng theo phương pháp thứ hai của sáng chế được mô tả.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ bỏ qua phép biến đổi ($transform_skip_flag$) biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời. Thiết bị xử lý tín hiệu video có thể thu cờ bỏ qua phép biến đổi từ dòng bit. Ngoài ra, khi cờ bỏ qua phép biến đổi không được thu từ dòng bit, thì thiết bị xử lý tín hiệu video có thể tạo cấu hình cờ bỏ qua phép biến đổi thành giá trị định trước.

- Trường hợp trong đó sự bỏ qua phép biến đổi không được áp dụng ($transform_skip_flag[xTbY][yTbY][cIdx]$ là 0):

- $qP = \text{Max}(0, qP + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$; Theo phương pháp thứ hai của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để bù cho sự thay đổi về phạm vi động. Nghĩa là, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu thông số lượng tử hóa được hiệu chỉnh, dựa trên độ lệch thông số lượng tử hóa ($qpActOffset$).

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa, dựa trên thành phần màu sắc của khối biến đổi và cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$). Thông số lượng tử hóa được hiệu chỉnh có thể là $qP + qpActOffset$.

Như được mô tả ở trên, khi cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng

cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là 0. Trong trường hợp này, độ lệch thông số lượng tử hóa ($qpActOffset$) có thể là 0, và thông số lượng tử hóa (qP) gốc có thể được duy trì. Khi cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba. Quy trình xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba đã được mô tả dựa vào các hình vẽ từ Fig.14 đến Fig.16, và do đó phần giải thích lặp lại được lược bỏ. Ngoài ra, thiết bị xử lý tín hiệu video có thể áp dụng độ lệch thông số lượng tử hóa ($qpActOffset$) cho thông số lượng tử hóa (qP) để thu thông số lượng tử hóa được hiệu chỉnh.

Khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động cắt thông số lượng tử hóa được hiệu chỉnh thành 0. Nghĩa là, thiết bị xử lý tín hiệu video có thể ngăn không cho thông số lượng tử hóa (qP) nhỏ hơn 0. Cụ thể hơn là, giá trị qP được hiệu chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ hai, dựa trên việc thông số lượng tử hóa được xác định trong vùng nguyên trong đó số nguyên này không âm, giá trị nhỏ hơn 0 có thể được cắt thành 0 để cho phép giá trị qP được hiệu chỉnh lớn hơn hoặc bằng 0.

Như đã được mô tả ở trên, $qpActOffset$ có thể được xác định là 0 hoặc một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba, dựa trên $cu_act_enabled_flag[xTbY][yTbY]$. Tuy nhiên, sáng chế không bị giới hạn ở đó, và theo các phương án khác nhau của sáng chế, giá trị có thể được thêm vào thông số lượng tử hóa (qP) bởi quy trình như sau. Khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối

hiện thời ($\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ hai của sáng chế, việc cắt dựa trên toán tử max có thể được thực hiện để ngăn không cho giá trị qP được hiệu chỉnh là số âm.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh. Thông số lượng tử hóa được hiệu chỉnh có thể là thông số lượng tử hóa được hiệu chỉnh được cắt. Cụ thể là, khi ACT được áp dụng, thì việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh qP và phương pháp khôi phục khối hiện thời có thể tuân theo các phương pháp được mô tả dựa vào Fig.16.

- Trường hợp trong đó sự bỏ qua phép biến đổi được áp dụng ($\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ là 1):

-
$$qP = \text{Max}(QpPrimeTsMin, qP + (\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? qpActOffset : 0));$$
 Theo phương pháp thứ hai của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để bù cho sự thay đổi về phạm vi động. Khi bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu thông số lượng tử hóa được hiệu chỉnh, dựa trên độ lệch thông số lượng tử hóa ($qpActOffset$). Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa, dựa trên thành phần màu sắc của khối biến đổi và cờ kích hoạt phép biến đổi không gian màu sắc ($\text{cu_act_enabled_flag}$). Thông số lượng tử hóa được hiệu chỉnh có thể là $qP + qpActOffset$. Tuy nhiên, sáng chế không bị giới hạn ở đó, và khi cờ kích hoạt phép biến đổi không gian màu sắc ($\text{cu_act_enabled_flag}$) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì giá trị qP có thể không được hiệu chỉnh.

Như được mô tả ở trên, khi cờ kích hoạt phép biến đổi không gian màu sắc ($\text{cu_act_enabled_flag}$) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ

lệch thông số lượng tử hóa ($qpActOffset$) là 0. Trong trường hợp này, độ lệch thông số lượng tử hóa ($qpActOffset$) có thể là 0, và thông số lượng tử hóa (qP) gốc có thể được duy trì. Khi cờ kích hoạt phép biến đổi không gian màu sắc ($cu_act_enabled_flag$) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba. Quy trình xác định độ lệch thông số lượng tử hóa ($qpActOffset$) là một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba đã được mô tả dựa vào các hình vẽ từ Fig.14 đến Fig.16, và do đó phần giải thích lặp lại được lược bỏ. Ngoài ra, thiết bị xử lý tín hiệu video có thể áp dụng độ lệch thông số lượng tử hóa ($qpActOffset$) cho thông số lượng tử hóa (qP) để thu thông số lượng tử hóa được hiệu chỉnh.

Khi thông số lượng tử hóa được hiệu chỉnh nhỏ hơn thông số lượng tử hóa nhỏ nhất ($QpPrimeTsMin$), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động cắt thông số lượng tử hóa được hiệu chỉnh thành thông số lượng tử hóa nhỏ nhất ($QpPrimeTsMin$). Nghĩa là, thiết bị xử lý tín hiệu video có thể ngăn không cho thông số lượng tử hóa (qP) nhỏ hơn thông số lượng tử hóa nhỏ nhất ($QpPrimeTsMin$). Cụ thể hơn là, giá trị qP được hiệu chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ hai, dựa trên thông tin về thông số lượng tử hóa được cho phép khi sự bỏ qua phép biến đổi được áp dụng, việc cắt có thể được thực hiện sao cho giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng thông số lượng tử hóa được cho phép theo chế độ bỏ qua phép biến đổi.

Như đã được mô tả ở trên, $qpActOffset$ có thể được xác định là 0 hoặc một trong số độ lệch QP thứ nhất, độ lệch QP thứ hai hoặc độ lệch QP thứ ba, dựa trên $cu_act_enabled_flag[xTbY][yTbY]$. Tuy nhiên, sáng chế không bị giới hạn ở đó, và theo các phương án khác nhau của sáng chế, giá trị có thể được thêm vào thông số lượng tử hóa (qP) bởi quy trình như sau. Khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0$, và thông số lượng tử hóa qP có thể

được hiệu chỉnh bởi `qpActOffset`. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời (`cu_act_enabled_flag[xTbY][yTbY]` là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa `qP` có thể không được hiệu chỉnh. Theo phương pháp thứ hai của sáng chế, việc cắt có thể được thực hiện sao cho giá trị thông số lượng tử hóa được hiệu chỉnh nhờ sử dụng toán tử `max` trở nên lớn hơn hoặc bằng thông số lượng tử hóa nhỏ nhất (`QpPrimeTsMin`). Biến `QpPrimeTsMin` xác định thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, và điều kiện rằng thông số lượng tử hóa cần lớn hơn hoặc bằng `QpPrimeTsMin` thậm chí đối với khối mà ACT được áp dụng có thể cần được thỏa mãn. Do đó, khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn `QpPrimeTsMin`, thì giá trị này có thể được cắt thành `QpPrimeTsMin`. Ví dụ, `QpPrimeTsMin` là số tự nhiên, và do đó khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì giá trị này có thể được cắt thành `QpPrimeTsMin`. Là ví dụ khác, thậm chí khi giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng 0, nếu giá trị này nhỏ hơn `QpPrimeTsMin`, thì giá trị này có thể được cắt thành `QpPrimeTsMin`.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được hiệu chỉnh. Thông số lượng tử hóa được hiệu chỉnh có thể là thông số lượng tử hóa được hiệu chỉnh được cắt. Cụ thể là, khi ACT được áp dụng, thì việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh `qP` và phương pháp khôi phục khối hiện thời có thể tuân theo các phương pháp được mô tả dựa vào Fig.16.

Fig.18 minh họa phương pháp hiệu chỉnh giá trị `QP`, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên `QP` được hiệu chỉnh theo phương pháp thứ ba của sáng chế. Phần được lược bỏ trong phần mô tả liên quan đến Fig.18 có thể tuân theo phần mô tả được đưa ra dựa vào Fig.16, và sau đây, phương pháp điều chỉnh thông số lượng tử hóa khi sự bỏ qua phép biến đổi không được áp dụng và khi sự bỏ qua phép biến đổi được áp dụng theo phương pháp thứ ba của sáng chế được mô tả.

- Trường hợp trong đó sự bỏ qua phép biến đổi không được áp dụng (`transform_skip_flag[xTbY][yTbY][cIdx]` là 0):

$$- qP = \text{Max}(QpBdOffset, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? qpActOffset : 0));$$

Theo phương pháp thứ ba của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ ba, giá trị nhỏ hơn $QpBdOffset$ có thể được cắt thành $QpBdOffset$ để làm cho thông số lượng tử hóa được điều chỉnh có giá trị lớn hơn hoặc bằng biến $QpBdOffset$, dựa trên độ lệch thông số lượng tử hóa phản ánh độ sâu bit của hình ảnh. $QpBdOffset$ là độ lệch thông số lượng tử hóa được thêm vào theo độ sâu bit của hình ảnh, và có thể được bao gồm và được biểu thị trong cấu trúc cú pháp mức cao hơn chẳng hạn như SPS, PPS, phần đầu ảnh hoặc phần đầu phân chia. Ví dụ, phần tử cú pháp bit_depth_minus8 có thể được báo hiệu để báo hiệu thông tin độ sâu bit trong SPS. Bộ giải mã có thể tạo cấu hình biến $QpBdOffset$ là $6 * \text{bit_depth_minus8}$, dựa trên phần tử cú pháp bit_depth_minus8 được phân tích. Ví dụ, nếu độ sâu bit (BitDepth) là 8, thì $QpBdOffset$ có thể được tạo cấu hình là 0. Là ví dụ khác, nếu độ sâu bit là 10, thì $QpBdOffset$ có thể được tạo cấu hình là 12. $Qp'Y$, $Qp'CbCr$, $Qp'Cb$ và $Qp'Cr$ được sử dụng trong quy trình thay đổi tỷ lệ được thu bằng cách thêm $QpBdOffset$, và biểu diễn hệ số biến đổi với độ chính xác được tạo cấu hình trước (ví dụ, 16 bit) theo độ sâu bit. Khi ACT được áp dụng cho khối hiện thời ($\text{cu_act_enabled_flag}[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $\text{cu_act_enabled_flag}[xTbY][yTbY] ? qpActOffset : 0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời ($\text{cu_act_enabled_flag}[xTbY][yTbY]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ ba của sáng chế, để xem xét độ lệch thông số lượng tử hóa theo độ sâu bit, việc cắt dựa trên toán tử max có thể được thực hiện sao cho giá trị qP mà là thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng $QpBdOffset$. Ví dụ, $QpBdOffset$ là số nguyên không âm, và do đó khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì giá trị này có thể được cắt thành $QpBdOffset$. Là ví dụ khác, thậm chí khi giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng 0, nếu giá trị này nhỏ hơn $QpBdOffset$, thì giá trị này có thể

được cắt thành QpBdOffset. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh qP có thể tuân theo phương pháp được mô tả dựa vào Fig.16.

- Trường hợp trong đó sự bỏ qua phép biến đổi được áp dụng (transform_skip_flag[xTbY][yTbY][cIdx] là 1):

- $qP = \text{Max}(\text{QpBdOffset}, \text{Max}(\text{QpPrimeTsMin}, qP) + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{qpActOffset} : 0))$; Theo phương pháp thứ ba của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ ba, đầu tiên, thông qua phép so sánh giữa biến QpPrimeTsMin và giá trị qP, việc cắt có thể được thực hiện để cho phép giá trị qP lớn hơn hoặc bằng QpPrimeTsMin. Biến QpPrimeTsMin xác định thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, thông số lượng tử hóa trước tiên có thể được cắt thành giá trị lớn hơn hoặc bằng QpPrimeTsMin bởi biến QpPrimeTsMin, và sau đó độ lệch thông số lượng tử hóa (qpActOffset) để bù cho sự thay đổi về phạm vi động do ACT có thể được thêm vào đó. Khi ACT được áp dụng cho khối hiện thời (cu_act_enabled_flag[xTbY][yTbY] là 1), thì qpActOffset có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{qpActOffset} : 0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi qpActOffset. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời (cu_act_enabled_flag[xTbY][yTbY] là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ ba của sáng chế, để xem xét độ lệch thông số lượng tử hóa theo độ sâu bit, việc cắt dựa trên toán tử max có thể được thực hiện sao cho giá trị qP mà là thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng QpBdOffset. Ví dụ, QpBdOffset là số nguyên không âm, và do đó khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì giá trị này có thể được cắt thành QpBdOffset. Là ví dụ khác, thậm chí khi giá trị thông số lượng tử hóa được

hiệu chỉnh lớn hơn hoặc bằng 0, nếu giá trị này nhỏ hơn QpBdOffset, thì giá trị này có thể được cắt thành QpBdOffset. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh qP có thể tuân theo phương pháp được mô tả dựa vào Fig.16.

Fig.19 minh họa phương pháp điều chỉnh giá trị QP, và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên QP được điều chỉnh theo phương pháp thứ tư của sáng chế. Phần được lược bỏ từ phần mô tả liên quan đến Fig.19 có thể tuân theo phần mô tả được đưa ra dựa vào Fig.16, và sau đây, phương pháp điều chỉnh thông số lượng tử hóa khi sự bỏ qua phép biến đổi không được áp dụng và khi sự bỏ qua phép biến đổi được áp dụng theo phương pháp thứ tư của sáng chế được mô tả.

- Trường hợp trong đó sự bỏ qua phép biến đổi không được áp dụng (transform_skip_flag[xTbY][yTbY][cIdx] là 0):

- $qP = \text{Max}(\text{QpBdOffset}, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{qpActOffset} : 0))$;

Theo phương pháp thứ tư của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ tư, giá trị nhỏ hơn QpBdOffset có thể được cắt thành QpBdOffset để làm cho thông số lượng tử hóa được điều chỉnh có giá trị lớn hơn hoặc bằng biến QpBdOffset, dựa trên độ lệch thông số lượng tử hóa phản ánh độ sâu bit của hình ảnh. QpBdOffset là độ lệch thông số lượng tử hóa được thêm vào theo độ sâu bit của hình ảnh, và biểu diễn hệ số biến đổi với độ chính xác được tạo cấu hình trước (ví dụ, 16 bit) theo độ sâu bit. Khi ACT được áp dụng cho khối hiện thời (cu_act_enabled_flag[xTbY][yTbY] là 1), thì qpActOffset có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{qpActOffset} : 0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi qpActOffset. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời (cu_act_enabled_flag[xTbY][yTbY] là 0), thì độ lệch thông số lượng tử hóa là 0,

và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ tư của sáng chế, để xem xét độ lệch thông số lượng tử hóa theo độ sâu bit, việc cắt dựa trên toán tử max có thể được thực hiện sao cho giá trị qP mà là thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng $QpBdOffset$. Ví dụ, $QpBdOffset$ là số nguyên không âm, và do đó khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì giá trị này có thể được cắt thành $QpBdOffset$. Là ví dụ khác, thậm chí khi giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng 0, nếu giá trị này nhỏ hơn $QpBdOffset$, thì giá trị này có thể được cắt thành $QpBdOffset$. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh qP có thể tuân theo phương pháp được mô tả dựa vào Fig.16.

- Trường hợp trong đó sự bỏ qua phép biến đổi được áp dụng ($transform_skip_flag[xTbY][yTbY][cIdx]$ là 1):

- $qP = \text{Max}(QpPrimeTsMin, qP + (cu_act_enabled_flag[xTbY][yTbY][cIdx] ? qpActOffset : 0))$; Theo phương pháp thứ tư của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được hiệu chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ tư, dựa trên thông tin về thông số lượng tử hóa được cho phép khi sự bỏ qua phép biến đổi được áp dụng, việc cắt có thể được thực hiện sao cho giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng thông số lượng tử hóa được cho phép theo chế độ bỏ qua phép biến đổi. Khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY] ? qpActOffset : 0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ tư của sáng chế, việc cắt có thể được thực hiện sao cho giá trị thông số lượng tử hóa được

hiệu chỉnh nhờ sử dụng toán tử max trở nên lớn hơn hoặc bằng $QpPrimeTsMin$. Biến $QpPrimeTsMin$ xác định thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, và điều kiện rằng thông số lượng tử hóa cần lớn hơn hoặc bằng $QpPrimeTsMin$ thậm chí đối với khối mà ACT được áp dụng có thể cần được thỏa mãn. Do đó, khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn $QpPrimeTsMin$, thì giá trị này có thể được cắt thành $QpPrimeTsMin$. Ví dụ, $QpPrimeTsMin$ là số tự nhiên, và do đó khi giá trị thông số lượng tử hóa được hiệu chỉnh nhỏ hơn 0, thì giá trị này có thể được cắt thành $QpPrimeTsMin$. Là ví dụ khác, thậm chí khi giá trị thông số lượng tử hóa được hiệu chỉnh lớn hơn hoặc bằng 0, nếu giá trị này nhỏ hơn $QpPrimeTsMin$, thì giá trị này có thể được cắt thành $QpPrimeTsMin$. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh. Phương pháp thay đổi tỷ lệ dựa trên thông số lượng tử hóa được hiệu chỉnh qP có thể tuân theo phương pháp được mô tả dựa vào Fig.16.

Fig.20 minh họa phương pháp điều chỉnh giá trị QP , và thực hiện việc thay đổi tỷ lệ (giải lượng tử hóa), dựa trên QP được điều chỉnh theo phương pháp thứ năm của sáng chế. Phần được lược bỏ từ phần mô tả liên quan đến Fig.20 có thể tuân theo phần mô tả được đưa ra dựa vào Fig.16, và sau đây, phương pháp điều chỉnh thông số lượng tử hóa khi sự bỏ qua phép biến đổi không được áp dụng và khi sự bỏ qua phép biến đổi được áp dụng theo phương pháp thứ năm của sáng chế được mô tả.

- Trường hợp trong đó sự bỏ qua phép biến đổi không được áp dụng ($transform_skip_flag[xTbY][yTbY]$ là 0):

- $qP = \text{Max}(0, qP + (cu_act_enabled_flag[xTbY][yTbY] ? qpActOffset : 0))$; Theo phương pháp thứ năm của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện. Theo phương pháp thứ năm, dựa trên việc thông số lượng tử hóa được xác định trong vùng nguyên trong đó số nguyên này không âm, giá trị nhỏ hơn 0 có thể được cắt thành 0 để cho phép giá trị qP được điều chỉnh lớn hơn hoặc

bằng 0. Khi ACT được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 1), thì $qpActOffset$ có thể được trả về dưới dạng độ lệch thông số lượng tử hóa bằng phép toán $cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0$, và thông số lượng tử hóa qP có thể được hiệu chỉnh bởi $qpActOffset$. Trong khi đó, khi ACT không được áp dụng cho khối hiện thời ($cu_act_enabled_flag[xTbY][yTbY]$ là 0), thì độ lệch thông số lượng tử hóa là 0, và do đó thông số lượng tử hóa qP có thể không được hiệu chỉnh. Theo phương pháp thứ năm của sáng chế, việc cắt dựa trên toán tử max có thể được thực hiện để ngăn không cho giá trị qP được hiệu chỉnh là số âm. Khi ACT được áp dụng, việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh.

- Trường hợp trong đó sự bỏ qua phép biến đổi được áp dụng ($transform_skip_flag[xTbY][yTbY]$ là 1): Theo phương pháp thứ năm của sáng chế, trong trường hợp trong đó ACT được áp dụng cho khối hiện thời, giá trị qP có thể được hiệu chỉnh để điều chỉnh sự thay đổi về phạm vi động, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó phép toán cắt có thể được thực hiện.

- Trường hợp trong đó ACT được áp dụng ($cu_act_enabled_flag[xTbY][yTbY]$ là 1):

- $qP = \text{Max}(0, qP + qpActOffset)$; Khi ACT được áp dụng, thì giá trị qP có thể được điều chỉnh bởi $qpActOffset$ mà là độ lệch thông số lượng tử hóa, và giá trị qP được điều chỉnh có thể lệch khỏi phạm vi được tạo cấu hình trước, và do đó việc cắt dựa trên toán tử max có thể được thực hiện. Khi giá trị qP được điều chỉnh là số âm, thì qP có thể được tạo cấu hình là 0 bằng toán tử max.

- Trường hợp trong đó ACT không được áp dụng ($cu_act_enabled_flag[xTbY][yTbY]$ là 0):

- $qP = \text{Max}(QpPrimeTsMin, qP)$; Khi ACT không được áp dụng, thì $qpActOffset$ mà là độ lệch thông số lượng tử hóa không được áp dụng. Do đó, việc cắt dựa trên toán tử max có thể được thực hiện sao cho giá trị qP trở nên lớn hơn hoặc bằng giá trị thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi. Khi giá trị qP hiện thời nhỏ hơn $QpPrimeTsMin$, thì giá trị qP có thể được tạo cấu hình là $QpPrimeTsMin$ bởi

toán tử max.

Theo phương pháp thứ năm, khi sự bỏ qua phép biến đổi được áp dụng, thì thông số lượng tử hóa nhỏ nhất được cho phép có thể được tạo cấu hình để thay đổi theo việc ACT có được áp dụng hay không. Khi ACT được áp dụng, thì giá trị thông số lượng tử hóa nhỏ nhất có thể được tạo cấu hình là 0, và khi ACT không được áp dụng, thì giá trị thông số lượng tử hóa nhỏ nhất có thể được tạo cấu hình là $QpPrimeTsMin$.

ACT được biểu thị ở mức đơn vị mã hóa theo các phương pháp được mô tả ở trên dựa vào các hình vẽ từ Fig.16 đến Fig.20, nhưng sáng chế không bị giới hạn ở đó. Ví dụ, thậm chí nếu ACT được biểu thị ở mức đơn vị biến đổi, trong trường hợp trong đó ACT được áp dụng cho đơn vị (khối) biến đổi mà khối biến đổi hiện thời được xử lý trong quy trình thay đổi tỷ lệ thuộc về, thông số lượng tử hóa có thể được hiệu chỉnh theo các phương pháp từ thứ nhất đến thứ năm được mô tả ở trên, và việc thay đổi tỷ lệ có thể được thực hiện dựa trên thông số lượng tử hóa được hiệu chỉnh.

Fig.21 minh họa phương pháp báo hiệu thông số lượng tử hóa nhỏ nhất được áp dụng cho chế độ bỏ qua phép biến đổi. Như được mô tả dựa vào các hình vẽ từ Fig.16 đến Fig.20, giá trị thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi có thể được bao gồm và được biểu thị trong ít nhất một trong số các cấu trúc cú pháp mức cao hơn bao gồm SPS, PPS, phần đầu ảnh, phần đầu phần chia hoặc dạng tương tự.

Cấu trúc (a) của Fig.21 thể hiện phương pháp biểu thị thông số lượng tử hóa nhỏ nhất được áp dụng cho chế độ bỏ qua phép biến đổi trong cấu trúc cú pháp SPS theo một phương án của sáng chế.

`chroma_format_idc` có thể được báo hiệu trong cấu trúc cú pháp SPS. Phần tử cú pháp `chroma_format_idc` biểu thị sự lấy mẫu sắc độ so với sự lấy mẫu độ sáng, và định dạng sắc độ của hình ảnh đang được xử lý có thể được xác định theo `chroma_format_idc`. Khi `chroma_format_idc` là 0, thì điều này có thể biểu thị định dạng đơn sắc, và chỉ một mảng mẫu độ sáng (kênh thứ nhất) có thể tồn tại ở định dạng đơn sắc. Khi `chroma_format_idc` là 1, thì điều này có thể biểu thị định dạng 4:2:0, chiều cao và chiều rộng của mảng mẫu sắc độ (kênh thứ hai và kênh thứ ba) ở định dạng 4:2:0 có thể bằng

một nửa chiều cao và chiều rộng của mảng mẫu độ sáng (kênh thứ nhất), một cách tương ứng. Khi `chroma_format_idc` là 2, điều này có thể biểu thị định dạng 4:2:2, chiều cao của mảng mẫu sắc độ (kênh thứ hai và kênh thứ ba) ở định dạng 4:2:2 có thể giống như chiều cao của mảng mẫu độ sáng (kênh thứ nhất), và chiều rộng của mảng mẫu sắc độ có thể bằng một nửa chiều rộng của mảng mẫu độ sáng. Khi `chroma_format_idc` là 3, điều này có thể biểu thị định dạng 4:4:4, và ngoài ra, `separate_colour_plane_flag` có thể được báo hiệu. Khi `separate_colour_plane_flag` là 1, thì điều này có thể biểu thị rằng ba thành phần của định dạng 4:4:4 được tách riêng mà được mã hóa độc lập, và khi `separate_colour_plane_flag` là 0, thì điều này có thể biểu thị rằng các thành phần tạo cấu hình không gian màu sắc không được mã hóa riêng. `separate_colour_plane_flag` có thể được báo hiệu chỉ trong trường hợp trong đó `chroma_format_idc` là 3, trường hợp biểu thị định dạng 4:4:4, và khi `separate_colour_plane_flag` không được báo hiệu (cũng như không được bao gồm trong dòng bit), thì `separate_colour_plane_flag` có thể được suy ra là 0. Khi `separate_colour_plane_flag` là 0 ở định dạng 4:4:4, thì chiều cao và chiều rộng của mảng mẫu thành phần sắc độ (kênh thứ hai và kênh thứ ba) có thể giống như chiều cao và chiều rộng của mảng mẫu thành phần độ sáng (kênh thứ nhất). Khi `separate_colour_plane_flag` là 1 ở định dạng 4:4:4, thì các thành phần tạo cấu hình không gian màu sắc có thể được mã hóa riêng, và mỗi thành phần có thể được xử lý bởi ảnh đơn sắc. Khi `separate_colour_plane_flag` là 1, thì `colour_plane_id` có thể được báo hiệu bổ sung trong phần đầu ảnh, và `colour_plane_id` có thể biểu thị thành phần mà, trong số thành phần kênh thứ nhất, thành phần kênh thứ hai và thành phần kênh thứ ba tạo cấu hình không gian màu sắc, ảnh đang được xử lý tương ứng với.

Trong cấu trúc cú pháp SPS, phần tử cú pháp `bit_depth_minus8` biểu thị thông tin độ sâu bit của các mẫu mảng độ sáng và sắc độ có thể được báo hiệu. Bộ giải mã có thể tạo cấu hình `BitDepth`, mà là biến biểu thị độ sâu bit của các mẫu mảng độ sáng và sắc độ, là $8 + \text{bit_depth_minus8}$, dựa trên cú pháp `bit_depth_minus8`. Khi `BitDepth` là N (số tự nhiên), thì các mẫu mảng kênh thứ nhất, kênh thứ hai và kênh thứ ba có thể có các giá trị nằm trong phạm vi $[0, 2^N - 1]$. Ví dụ, khi `BitDepth` là 8, thì các mẫu mảng kênh thứ nhất, kênh thứ hai và kênh thứ ba có thể có các giá trị nằm trong phạm vi $[0, 255]$.

Trong cấu trúc cú pháp SPS, phần tử cú pháp `min_qp_prime_ts_minus4` biểu thị thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi có thể được báo hiệu. Bộ giải mã có thể tạo cấu hình `QpPrimeTsMin`, mà là biến biểu thị giá trị thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, thành `4+min_qp_prime_ts_minus4`, dựa trên phần tử cú pháp `min_qp_prime_ts_minus4`.

Trong cấu trúc cú pháp SPS, phần tử cú pháp `sps_transform_skip_enabled_flag` có thể được báo hiệu. Khi `sps_transform_skip_enabled_flag` là 1, thì điều này biểu thị rằng `transform_skip_flag[][][]`, mà là phần tử cú pháp biểu thị chế độ bỏ qua phép biến đổi, có thể tồn tại trong cú pháp đơn vị biến đổi, và điều này biểu thị rằng chế độ bỏ qua phép biến đổi có thể được áp dụng ở mức đơn vị biến đổi trình tự thấp hơn. Trong khi đó, khi `sps_transform_skip_enabled_flag` là 0, thì điều này biểu thị rằng `transform_skip_flag[][][]` không tồn tại trong cú pháp đơn vị biến đổi (cũng như không được bao gồm trong dòng bit), và điều này biểu thị rằng chế độ bỏ qua phép biến đổi có thể không được áp dụng ở mức đơn vị biến đổi trình tự thấp hơn.

Trong cấu trúc cú pháp SPS, khi `chroma_format_idc` là 3 (định dạng là định dạng 4:4:4), thì thông tin biểu thị xem phương pháp mã hóa đối với định dạng 4:4:4 là bật hay tắt có thể được báo hiệu bổ sung.

Khi `sps_act_enabled_flag` là 1, thì điều này biểu thị rằng ACT có thể được sử dụng và `cu_act_enabled_flag` biểu thị xem ACT có được áp dụng hay không ở mức đơn vị mã hóa có thể tồn tại trong cú pháp đơn vị mã hóa. Khi `sps_act_enabled_flag` là 0, thì điều này biểu thị rằng ACT không được sử dụng và `cu_act_enabled_flag` không tồn tại trong cú pháp đơn vị mã hóa (cũng như không được bao gồm trong dòng bit). Khi `sps_act_enabled_flag` không tồn tại, thì nó có thể được suy ra là 0.

Khi `sps_palette_enabled_flag` là 1, thì điều này biểu thị rằng `pred_mode_plt_flag` biểu thị xem chế độ bảng màu có được áp dụng hay không ở mức đơn vị mã hóa có thể tồn tại trong cú pháp đơn vị mã hóa, và điều này có thể biểu thị rằng chế độ bảng màu mà là một trong số các phương pháp dự đoán có thể được áp dụng ở mức đơn vị mã hóa trình tự

thấp hơn. Khi `sps_palette_enabled_flag` là 0, thì điều này biểu thị rằng `pred_mode_plt_flag` không tồn tại trong cú pháp đơn vị mã hóa (cũng như không được bao gồm trong dòng bit), và điều này có thể biểu thị rằng chế độ bảng màu không được áp dụng. Khi `sps_palette_enabled_flag` không tồn tại, thì nó có thể được suy ra là 0.

Như được mô tả ở trên, `QpPrimeTsMin`, mà là biến biểu thị thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, có thể được tạo cấu hình dựa trên phần tử cú pháp `min_qp_prime_ts_minus4`, và phần tử cú pháp `min_qp_prime_ts_minus4` có thể luôn luôn được báo hiệu không phụ thuộc vào phần tử cú pháp khác trong cấu trúc cú pháp SPS.

Trong khi đó, thông số `QpPrimeTsMin` biểu thị thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi, và khi giá trị thoát được mã hóa trong trường hợp trong đó chế độ bảng màu được áp dụng cho khôi hiện thời, hoặc chế độ bỏ qua phép biến đổi được áp dụng, thông số có thể được sử dụng một cách giới hạn. Nghĩa là, trong trường hợp trong đó biểu thị rằng cả chế độ bỏ qua phép biến đổi và chế độ bảng màu không được áp dụng ở mức SPS (`sps_transform_skip_enabled_flag` là 0 và `sps_palette_enabled_flag` là 0), có thể không cần thiết bao gồm phần tử cú pháp `min_qp_prime_ts_minus4` trong dòng bit cho `QpPrimeTsMin`. Do đó, chỉ trong trường hợp trong đó biểu thị rằng ít nhất một trong số chế độ bỏ qua phép biến đổi và chế độ bảng màu có thể được sử dụng ở mức SPS (`sps_transform_skip_enabled_flag` là 1, hoặc `sps_palette_enabled_flag` là 1), khi phần tử cú pháp `min_qp_prime_ts_minus4` được báo hiệu, thì lượng dòng bit có thể được làm giảm.

Sơ đồ (b) của Fig.21 thể hiện phương pháp biểu thị thông số lượng tử hóa nhỏ nhất được áp dụng cho chế độ bỏ qua phép biến đổi trong cấu trúc cú pháp SPS theo phương án khác của sáng chế.

Trong sơ đồ (b) của Fig.21, `min_qp_prime_ts_minus4` có thể biểu thị thông tin liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi. Dựa vào sơ đồ (b) của Fig.21, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ kích hoạt bỏ qua phép biến đổi (`sps_transform_skip_enabled_flag`) từ dòng bit. Cờ kích

hoạt bỏ qua phép biến đổi (`sps_transform_skip_enabled_flag`) có thể biểu thị xem phần tử cú pháp (`transform_skip_flag`) biểu thị chế độ bỏ qua phép biến đổi có thể tồn tại trong cú pháp đơn vị biến đổi hay không. Phần tử cú pháp (`transform_skip_flag`) biểu thị chế độ bỏ qua phép biến đổi có thể là cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời. Khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi được bao gồm trong khối hiện thời, phép biến đổi có thể không được áp dụng. Cờ kích hoạt bỏ qua phép biến đổi có thể được thu từ dòng bit trong số SPS, PPS, phần đầu ảnh hoặc phần đầu phần chia.

Ngoài ra, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ kích hoạt bảng màu (`sps_palette_enabled_flag`) từ dòng bit. Cờ kích hoạt bảng màu (`sps_palette_enabled_flag`) có thể biểu thị xem chế độ bảng màu có được áp dụng hay không. Cờ kích hoạt bảng màu (`sps_palette_enabled_flag`) có thể được thu từ dòng bit trong số SPS, PPS, phần đầu ảnh hoặc phần đầu phần chia.

Ngoài ra, trên sơ đồ (b) của Fig.21, phần tử cú pháp `min_qp_prime_ts_minus4` có thể được báo hiệu một cách giới hạn như sau, dựa trên `sps_transform_skip_enabled_flag` và `sps_palette_enabled_flag`.

```
if(sps_transform_skip_enabled_flag || sps_palette_enabled_flag)
    min_qp_prime_ts_minus4
```

Nghĩa là, khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng phần tử cú pháp (`transform_skip_flag`) biểu thị chế độ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc cờ kích hoạt bảng màu (`sps_palette_enabled_flag`) biểu thị việc áp dụng chế độ bảng màu, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu thông tin (`min_qp_prime_ts_minus4`) liên quan đến thông số lượng tử hóa nhỏ nhất được cho phép theo chế độ bỏ qua phép biến đổi từ dòng bit. Cụ thể hơn là, khi biểu thị rằng ít nhất một trong số chế độ bỏ qua phép biến đổi và chế độ bảng màu có thể được sử dụng ở mức SPS (`sps_transform_skip_enabled_flag` là 1, hoặc `sps_palette_enabled_flag` là 1), thì phần tử cú pháp `min_qp_prime_ts_minus4` có thể được bao gồm trong dòng bit, và ngược lại

(sps_transform_skip_enabled_flag và sps_palette_enabled_flag both 0), thì phần tử cú pháp min_qp_prime_ts_minus4 có thể không được bao gồm trong dòng bit. Khi phần tử cú pháp min_qp_prime_ts_minus4 không tồn tại trong dòng bit, thì phần tử cú pháp có thể được suy ra là 0. Như được mô tả ở trên, khi cả chế độ bỏ qua phép biến đổi và chế độ bảng màu không được áp dụng ở mức SPS, thì phần tử cú pháp min_qp_prime_ts_minus4 không được báo hiệu, và do đó lượng dòng bit có thể được làm giảm.

Theo phương án trên, biến QpPrimeTsMin được mô tả là được sử dụng chỉ theo chế độ bỏ qua phép biến đổi và chế độ bảng màu, nhưng sáng chế không bị giới hạn ở đó. Ví dụ, khi thông số QpPrimeTsMin được sử dụng trong các phương pháp mã hóa A, B, C và D, và các phần tử cú pháp bật/tắt của các phương pháp mã hóa ở mức SPS lần lượt là sps_tool_A_enabled_flag, sps_tool_B_enabled_flag, sps_tool_C_enabled_flag và sps_tool_D_enabled_flag, phần tử cú pháp min_qp_prime_ts_minus4 có thể được báo hiệu một cách giới hạn như sau.

```
if(sps_tool_A_enabled_flag || sps_tool_B_enabled_flag || sps_tool_C_enabled_flag
|| sps_tool_D_enabled_flag)
```

```
min_qp_prime_ts_minus4
```

Điều này biểu thị rằng, chỉ trong trường hợp trong đó biểu thị rằng ít nhất một trong số các phương pháp mã hóa A, B, C và D có thể được áp dụng ở mức SPS, phần tử cú pháp min_qp_prime_ts_minus4 được bao gồm trong dòng bit.

Fig.22 là sơ đồ minh họa một phương án phân chia khối theo chế độ phân chia con nội ảnh (ISP). Chế độ ISP tương ứng với phương pháp mà, khi phép dự đoán nội ảnh được áp dụng cho đơn vị mã hóa hiện thời, phân chia khối hiện thời thành nhiều đơn vị (khối) biến đổi và thực hiện việc dự đoán và việc khôi phục. Theo chế độ ISP, khối hiện thời có thể được phân chia thành nhiều khối con bằng phương pháp được tạo cấu hình trước, và phương pháp được tạo cấu hình trước có thể bao gồm phân chia ngang và phân chia dọc. Số lượng khối con được thu bằng cách phân chia khối hiện thời có thể được xác định theo kích thước của khối hiện thời. Ví dụ, khi kích thước của đơn vị mã hóa là 4x4, thì chế độ ISP có thể không được áp dụng, và điều này có thể có nghĩa là đơn vị mã hóa không được

phân chia. Khi kích thước của đơn vị mã hóa là 4×8 hoặc 8×4 , thì đơn vị mã hóa có thể được phân chia thành hai đơn vị (khối) biến đổi, và trong trường hợp các kích thước khác, đơn vị mã hóa có thể được phân chia thành bốn đơn vị biến đổi. Kích thước đơn vị mã hóa mà chế độ ISP có thể được áp dụng có thể được xác định theo thông tin liên quan đến phép biến đổi. Ví dụ, chế độ ISP có thể được áp dụng cho trường hợp trong đó cả chiều rộng và chiều cao của đơn vị mã hóa đều nhỏ hơn hoặc bằng kích thước lớn nhất của khối biến đổi độ sáng, và tích số của chiều rộng và chiều cao của đơn vị mã hóa lớn hơn bình phương của kích thước nhỏ nhất của khối biến đổi độ sáng. Kích thước nhỏ nhất và kích thước lớn nhất của khối biến đổi độ sáng có thể lần lượt là 4 và 64.

Sơ đồ (a) của Fig.22 minh họa phương pháp phân chia đơn vị biến đổi mà có thể được áp dụng cho đơn vị mã hóa 4×8 hoặc 8×4 , và khi sự phân chia ngang được biểu thị cho đơn vị mã hóa có kích thước $W \times H$, thì đơn vị mã hóa có thể được phân chia thành hai đơn vị biến đổi có kích thước $W \times (H/2)$, và khi sự phân chia dọc được biểu thị, thì đơn vị mã hóa có thể được phân chia thành hai đơn vị biến đổi có kích thước $(W/2) \times H$. Việc sự phân chia là sự phân chia ngang hay sự phân chia dọc có thể được biểu thị bởi cờ 1 bit.

Sơ đồ (b) của Fig.22 minh họa phương pháp phân chia đơn vị biến đổi mà có thể được áp dụng cho đơn vị mã hóa không có kích thước 4×8 hoặc 8×4 , và khi sự phân chia ngang được biểu thị cho đơn vị mã hóa có kích thước $W \times H$, thì đơn vị mã hóa có thể được phân chia thành bốn đơn vị biến đổi có kích thước $W \times (H/4)$, và khi sự phân chia dọc được biểu thị, thì đơn vị mã hóa có thể được phân chia thành bốn đơn vị biến đổi có kích thước $(W/4) \times H$. Việc sự phân chia là sự phân chia ngang hay sự phân chia dọc có thể được biểu thị bởi cờ 1 bit.

Khi chế độ ISP được áp dụng cho đơn vị mã hóa hiện thời, khối sắc độ có thể không được phân chia. Số lượng được tạo cấu hình trước của các đơn vị biến đổi được phân chia bởi chế độ ISP có thể bao gồm các khối biến đổi độ sáng, nhưng khối biến đổi sắc độ có thể chỉ được bao gồm trong một số đơn vị biến đổi. Cụ thể là, khối biến đổi sắc độ có thể chỉ được bao gồm trong đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia. Ví dụ, khi chế độ ISP được áp dụng cho đơn vị mã hóa 4×8 hoặc 8×4 , thì đơn vị mã

hóa có thể bao gồm hai đơn vị biến đổi được phân chia, và mỗi trong số các đơn vị biến đổi có thể bao gồm khối biến đổi độ sáng. Trong khi đó, khối biến đổi sắc độ có thể chỉ được bao gồm trong đơn vị biến đổi được phân chia thứ hai, và kích thước của khối biến đổi sắc độ có thể giống như kích thước của khối mã hóa sắc độ. Là ví dụ khác, khi chế độ ISP được áp dụng cho đơn vị mã hóa không có kích thước 4×8 hoặc 8×4 , thì đơn vị mã hóa có thể bao gồm bốn đơn vị biến đổi được phân chia, và mỗi trong số các đơn vị biến đổi có thể bao gồm khối biến đổi độ sáng. Trong khi đó, khối biến đổi sắc độ có thể chỉ được bao gồm trong đơn vị biến đổi được phân chia thứ tư, và kích thước của khối biến đổi sắc độ có thể giống như kích thước của khối mã hóa sắc độ.

Đối với tín hiệu dư của khối được dự đoán liên ảnh, tín hiệu dư có thể tồn tại đồng tâm ở vị trí cụ thể của khối (năng lượng của tín hiệu dư là lớn ở cùng vị trí). Ví dụ, mẫu hình trong đó năng lượng của tín hiệu dư là lớn ở vùng biên của khối tín hiệu dư, và năng lượng của tín hiệu dư là tương đối nhỏ ở phần tâm của khối và vùng biên phía khác có thể tồn tại. Trong trường hợp sử dụng mẫu hình này của tín hiệu dư của khối được dự đoán liên ảnh để phân chia khối hiện thời thành các khối con và thực hiện phép biến đổi và phép lượng tử hóa, hiệu quả mã hóa của khối được dự đoán liên ảnh có thể được cải thiện.

Fig.23 minh họa phương pháp biến đổi khối con (SBT). SBT có thể phân chia đơn vị mã hóa hiện thời với tỷ lệ được tạo cấu hình trước, và thực hiện phép biến đổi và phép lượng tử hóa đối với chỉ một số đơn vị biến đổi trong số các đơn vị biến đổi được phân chia. Theo sáng chế, sự phân chia thành hai đơn vị biến đổi có tỷ lệ 1:1 hoặc 1:3 được mô tả, nhưng sáng chế không bị giới hạn ở đó.

SBT có thể giả sử rằng tín hiệu dư được tập trung vào một đơn vị biến đổi trong số các đơn vị biến đổi được phân chia, thực hiện phép biến đổi và phép lượng tử hóa đối với chỉ một đơn vị biến đổi, và giả sử rằng tín hiệu dư không tồn tại trong các đơn vị biến đổi còn lại. Trên các sơ đồ (a), (b), (c) và (d) của Fig.23, đơn vị biến đổi trong vùng được biểu thị bởi “A” tương ứng với vùng trong đó phép biến đổi và phép lượng tử hóa được thực hiện, và việc hệ số biến đổi (hệ số biến đổi được lượng tử hóa) khác 0 tồn tại trong khối biến đổi sau khi lượng tử hóa có thể được biểu thị nhờ sử dụng phần tử cú pháp liên quan

đến phép biến đổi. Cụ thể là, cờ hệ số biến đổi có thể bao gồm cờ hệ số biến đổi thứ nhất (tu_cbf_luma), cờ hệ số biến đổi thứ hai (tu_cbf_cb) hoặc cờ hệ số biến đổi thứ ba (tu_cbf_cr). Theo sáng chế, cờ hệ số biến đổi thứ nhất, cờ hệ số biến đổi thứ hai và cờ hệ số biến đổi thứ ba cũng có thể được gọi lần lượt là $tu_y_coded_flag$, $tu_cb_coded_flag$ và $tu_cr_coded_flag$. Bộ mã hóa có thể báo hiệu phần tử cú pháp tu_cbf_cb , và bộ giải mã có thể xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ hai hay không, dựa trên phần tử cú pháp tu_cbf_cb được phân tích từ dòng bit. Khi cờ hệ số biến đổi thứ hai (tu_cbf_cb) là 1, thì điều này có thể biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ hai, và khi cờ hệ số biến đổi thứ hai (tu_cbf_cb) là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ hai là 0. Khi ACT được áp dụng cho khối hiện thời, thì khối biến đổi kênh thứ hai có thể là khối biến đổi Cg. Khi ACT không được áp dụng cho khối hiện thời, thì khối có thể biểu thị khối biến đổi kênh thứ hai của không gian màu sắc gốc, và có thể là khối biến đổi Cb của YCbCr.

Tiếp theo, bộ mã hóa có thể báo hiệu phần tử cú pháp tu_cbf_cr , và bộ giải mã có thể xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ ba hay không, dựa trên phần tử cú pháp tu_cbf_cr được phân tích từ dòng bit. Khi cờ hệ số biến đổi thứ ba (tu_cbf_c) là 1, thì điều này có thể biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ ba, và khi cờ hệ số biến đổi thứ ba (tu_cbf_cr) là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ ba là 0. Khi ACT được áp dụng cho khối hiện thời, thì khối biến đổi kênh thứ ba có thể là khối biến đổi Co. Khi ACT không được áp dụng cho khối hiện thời, thì khối có thể biểu thị khối biến đổi kênh thứ ba của không gian màu sắc gốc, và có thể là khối biến đổi Cr của YCbCr.

Tiếp theo, bộ mã hóa có thể báo hiệu phần tử cú pháp tu_cbf_luma mà là cờ hệ số biến đổi thứ nhất và bộ giải mã có thể xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không, dựa trên phần tử cú pháp tu_cbf_luma được phân tích từ dòng bit. Khi cờ hệ số biến đổi thứ nhất (tu_cbf_luma) là 1, thì điều này có thể biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối

biến đổi kênh thứ nhất, và khi cờ hệ số biến đổi thứ nhất (tu_cbf_luma) là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ nhất là 0. Khối biến đổi kênh thứ nhất có thể là khối biến đổi độ sáng.

Trên các sơ đồ (a), (b), (c) và (d) của Fig.23, vùng được biểu thị bởi “A” tương ứng với vùng trong đó tín hiệu dư tồn tại, và bộ mã hóa có thể thực hiện phép biến đổi và phép lượng tử hóa đối với đơn vị biến đổi được biểu thị bởi “A”, và sau đó báo hiệu các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr , dựa trên thông tin hệ số biến đổi được lượng tử hóa. Bộ giải mã có thể xác định xem có thực hiện việc mã hóa hệ số (mã hóa phần dư), thay đổi tỷ lệ (giải lượng tử hóa), và phép biến đổi ngược đối với khối biến đổi tương ứng hay không, dựa trên các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr được phân tích từ dòng bit. Khi khối biến đổi tương ứng bao gồm một hoặc nhiều hệ số khác 0, dựa trên các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr , bộ giải mã có thể thu hệ số biến đổi được lượng tử hóa (mảng TransCoeffLevel), dựa trên các phần tử cú pháp được phân tích từ dòng bit. Bộ giải mã có thể giải lượng tử hóa hệ số biến đổi được lượng tử hóa để thu hệ số biến đổi, và biến đổi ngược hệ số biến đổi để thu tín hiệu dư. Trong khi đó, khi tất cả các hệ số của khối biến đổi là 0, thì bộ giải mã có thể không thực hiện việc mã hóa hệ số, thay đổi tỷ lệ và phép biến đổi ngược, và tạo cấu hình tất cả các mảng mẫu dư là 0. Phương pháp báo hiệu tu_cbf_luma , tu_cbf_cb và tu_cbf_cr có thể được mô tả dưới đây dựa vào Fig.25 và Fig.26.

Trong khi đó, vùng không được biểu thị bởi “A” trên các sơ đồ (a), (b), (c) và (d) của Fig.23 là vùng đơn vị biến đổi giả sử rằng tín hiệu dư không tồn tại trong đó, các quy trình mã hóa hệ số, giải lượng tử hóa và biến đổi ngược có thể không được thực hiện, và bộ mã hóa có thể không báo hiệu các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr . Bộ giải mã có thể tạo cấu hình các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr là 0 mà không phân tích các phần tử cú pháp này từ dòng bit.

Sơ đồ (a) của Fig.23 minh họa phương pháp mà, khi kích thước của đơn vị mã hóa là $W \times H$, phân chia dọc đơn vị mã hóa thành hai đơn vị biến đổi có kích thước $(W/2) \times H$. Theo cách khác, khi cờ 1 bit biểu thị sự phân chia $\{1/4, 3/4\}$ là 1, thì đơn vị mã hóa có thể

được phân chia thành hai đơn vị biến đổi có kích thước $(W/4) \times H$ và kích thước $(3W/4) \times H$. Trong cả hai trường hợp trên, phép biến đổi có thể được thực hiện đối với đơn vị biến đổi bên trái, DCT-VIII có thể được áp dụng theo hướng ngang, và DST-VII có thể được áp dụng theo hướng dọc. Nếu chiều rộng hoặc chiều cao của đơn vị biến đổi tương ứng lớn hơn giá trị được tạo cấu hình trước, thì DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc. Giá trị được tạo cấu hình trước có thể là 32. Trong trường hợp khối sắc độ, DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc.

Sơ đồ (b) của Fig.23 minh họa phương pháp mà, khi kích thước của đơn vị mã hóa là $W \times H$, phân chia dọc đơn vị mã hóa thành hai đơn vị biến đổi có kích thước $(W/2) \times H$. Theo cách khác, khi cờ 1 bit biểu thị sự phân chia $\{1/4, 3/4\}$ là 1, thì đơn vị mã hóa có thể được phân chia thành hai đơn vị biến đổi có kích thước $(3W/4) \times H$ và kích thước $(W/4) \times H$. Trong cả hai trường hợp trên, phép biến đổi có thể được thực hiện đối với đơn vị biến đổi bên phải, và DST-VII có thể được áp dụng cả theo hướng ngang và hướng dọc. Nếu chiều rộng hoặc chiều cao của đơn vị biến đổi tương ứng lớn hơn giá trị được tạo cấu hình trước, thì DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc. Giá trị được tạo cấu hình trước có thể là 32. Trong trường hợp khối sắc độ, DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc.

Sơ đồ (c) của Fig.23 minh họa phương pháp mà, khi kích thước của đơn vị mã hóa là $W \times H$, phân chia ngang đơn vị mã hóa thành hai đơn vị biến đổi có kích thước $W \times (H/2)$. Theo cách khác, khi cờ 1 bit biểu thị sự phân chia $\{1/4, 3/4\}$ là 1, thì đơn vị mã hóa có thể được phân chia thành hai đơn vị biến đổi có kích thước $W \times (H/4)$ và kích thước $W \times (3H/4)$. Trong cả hai trường hợp trên, phép biến đổi có thể được thực hiện đối với đơn vị biến đổi phía trên, DST-VII có thể được áp dụng theo hướng ngang, và DCT-VIII có thể được áp dụng theo hướng dọc. Nếu chiều rộng hoặc chiều cao của đơn vị biến đổi tương ứng lớn hơn giá trị được tạo cấu hình trước, thì DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc. Giá trị được tạo cấu hình trước có thể là 32. Trong trường hợp khối sắc độ, DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc.

Sơ đồ (d) của Fig.23 minh họa phương pháp mà, khi kích thước của đơn vị mã hóa

là $W \times H$, phân chia ngang đơn vị mã hóa thành hai đơn vị biến đổi có kích thước $W \times (H/2)$. Theo cách khác, khi cờ 1 bit biểu thị sự phân chia $\{1/4, 3/4\}$ là 1, thì đơn vị mã hóa có thể được phân chia thành hai đơn vị biến đổi có kích thước $W \times (3H/4)$ và kích thước $W \times (H/4)$. Trong cả hai trường hợp trên, phép biến đổi có thể được thực hiện đối với đơn vị biến đổi phía dưới, và DST-VII có thể được áp dụng cả theo hướng ngang và hướng dọc. Nếu chiều rộng hoặc chiều cao của đơn vị biến đổi tương ứng lớn hơn giá trị được tạo cấu hình trước, thì DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc. Giá trị được tạo cấu hình trước có thể là 32. Trong trường hợp khối sắc độ, DCT-II có thể được áp dụng theo cả hai hướng ngang và dọc.

Việc có sử dụng SBT hay không có thể được xác định theo `cu_cbf` mà là phần tử cú pháp biểu thị sự tồn tại/vắng mặt của cấu trúc cú pháp `transform_tree()` mà là cấu trúc cú pháp liên quan đến phép biến đổi, và việc có áp dụng SBT hay không có thể được báo hiệu ở mức đơn vị mã hóa. Theo sáng chế, phần tử cú pháp `cu_cbf` cũng có thể được gọi là `cu_coded_flag`. Ví dụ, khi cờ SBT (`cu_sbt_flag`) là 1, thì điều này có thể biểu thị rằng SBT được sử dụng cho đơn vị mã hóa hiện thời, và khi cờ SBT (`cu_sbt_flag`) là 0, thì điều này có thể biểu thị rằng SBT không được sử dụng cho đơn vị mã hóa hiện thời. Khi `cu_cbf` là 1, thì điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` tồn tại cho đơn vị mã hóa hiện thời, và điều này có thể có nghĩa là đơn vị mã hóa hiện thời bao gồm một hoặc nhiều hệ số mà ít nhất không bằng 0. Trong khi đó, khi `cu_cbf` là 0, thì điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` không tồn tại, và điều này có thể có nghĩa là tín hiệu dư không tồn tại. Bộ mã hóa và bộ giải mã có thể tạo cấu hình nút gốc có cùng kích thước với đơn vị mã hóa, và phân chia cây biến đổi, và nút lá của cây biến đổi được phân chia có thể là đơn vị biến đổi. Cây biến đổi có thể được thực hiện bởi cấu trúc cú pháp `transform_tree()`, và bộ giải mã có thể xác định xem cấu trúc cú pháp `transform_tree()` có tồn tại hay không, dựa trên phần tử cú pháp `cu_cbf` được phân tích từ dòng bit. Khi không có `transform_tree()`, thì sự phân chia đơn vị biến đổi không thể được thực hiện bởi `transform_tree()`, và do đó SBT có thể không được áp dụng. Trong trường hợp này, bộ giải mã có thể không phân tích cờ SBT (`cu_sbt_flag`) từ dòng bit và tạo cấu hình cờ SBT là 0.

Trong phương án khác, việc có sử dụng SBT hay không có thể được xác định theo

phương pháp dự đoán. SBT có hiệu quả khi tín hiệu dư được phân bố để được tập trung vào một vùng, và do đó, khi phương pháp dự đoán là phép dự đoán liên ảnh, thì SBT có thể được áp dụng. Trong trường hợp phép dự đoán nội ảnh, SBT có thể không được áp dụng, và bộ giải mã có thể không phân tích cờ SBT (`cu_sbt_flag`) từ dòng bit và tạo cấu hình cờ SBT là 0. Ngoài ra, việc sử dụng SBT có thể bị giới hạn ở phương pháp dự đoán cụ thể của phép dự đoán liên ảnh. Ví dụ, là phương pháp của phép dự đoán hợp nhất trong số các phép dự đoán liên ảnh, khối tham chiếu được biểu thị bởi ứng viên hợp nhất và khối dự đoán nội ảnh được tạo ra nhờ sử dụng chế độ dự đoán nội ảnh được tạo cấu hình trước có thể được hợp nhất để tạo ra khối dự đoán của khối hiện thời. Trong trường hợp này, đặc điểm của tín hiệu dư có thể thể hiện đặc điểm của khối được dự đoán nội ảnh, và do đó SBT có thể không được áp dụng.

Khi cờ SBT (`cu_sbt_flag`) là 1, thì điều này có thể biểu thị rằng SBT được áp dụng cho khối hiện thời, và phần tử cú pháp khác có thể được sử dụng để biểu thị phương pháp phân chia cây biến đổi. Đầu tiên, phần tử cú pháp biểu thị xem sự phân chia này là phân chia $\{1/4, 3/4\}$ hay là phân chia $\{1/2, 1/2\}$ có thể được báo hiệu. Ví dụ, khi `cu_sbt_quad_flag` là 1, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời bao gồm đơn vị biến đổi mà được thu thông qua SBT và bằng $1/4$ kích thước của đơn vị mã hóa hiện thời, và do đó có thể có nghĩa là đơn vị mã hóa hiện thời bao gồm hai đơn vị biến đổi được phân chia thành các kích thước $\{1/4, 3/4\}$. Trong khi đó, khi `cu_sbt_quad_flag` là 0, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời bao gồm đơn vị biến đổi mà được thu thông qua SBT và bằng $1/2$ kích thước của đơn vị mã hóa hiện thời, và do đó có thể có nghĩa là đơn vị mã hóa hiện thời bao gồm hai đơn vị biến đổi được phân chia thành các kích thước $\{1/2, 1/2\}$.

Tiếp theo, thông tin về hướng phân chia có thể được báo hiệu. Ví dụ, khi `cu_sbt_horizontal_flag` là 1, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời được phân chia ngang thành hai đơn vị biến đổi bởi SBT. Trong khi đó, khi `cu_sbt_horizontal_flag` là 0, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời được phân chia dọc thành hai đơn vị biến đổi bởi SBT.

Tiếp theo, thông tin biểu thị đơn vị biến đổi trong số các đơn vị biến đổi được phân chia bởi SBT, trong đó tín hiệu dư tồn tại, có thể được báo hiệu. Ví dụ, khi cờ vị trí SBT (`cu_sbt_pos_flag`) là 1, thì điều này có thể biểu thị rằng các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` không tồn tại trong dòng bit đối với đơn vị biến đổi thứ nhất trong số hai đơn vị biến đổi được phân chia bởi SBT. Điều này có thể ngụ ý rằng tín hiệu dư không tồn tại trong đơn vị biến đổi thứ nhất. Bộ mã hóa có thể không báo hiệu các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` đối với đơn vị biến đổi thứ nhất. Bộ giải mã có thể tạo cấu hình các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` là 0 mà không phân tích các phần tử cú pháp này từ dòng bit đối với đơn vị biến đổi thứ nhất. Trong khi đó, khi cờ vị trí SBT (`cu_sbt_pos_flag`) là 0, thì điều này có thể biểu thị rằng các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` không tồn tại trong dòng bit đối với đơn vị biến đổi thứ hai trong số hai đơn vị biến đổi được phân chia bởi SBT. Điều này có thể ngụ ý rằng tín hiệu dư không tồn tại trong đơn vị biến đổi thứ hai. Bộ mã hóa có thể không báo hiệu các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` đối với đơn vị biến đổi thứ hai. Bộ giải mã có thể tạo cấu hình các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` là 0 mà không phân tích các phần tử cú pháp này từ dòng bit đối với đơn vị biến đổi thứ hai.

Khi SBT được áp dụng, thì nhân biến đổi được áp dụng theo hướng ngang và hướng dọc theo chỉ số (vị trí) của đơn vị biến đổi và kích thước của đơn vị biến đổi có thể được chọn, và do đó phần tử cú pháp biểu thị nhân biến đổi có thể không được báo hiệu rõ ràng.

Chế độ ISP và SBT được mô tả ở trên dựa vào Fig.22 và Fig.23 có thể phân chia cây biến đổi, và Fig.24 minh họa cấu trúc cú pháp `transform_tree()` mà là cấu trúc cú pháp liên quan đến cây biến đổi. Cấu trúc cú pháp `transform_tree()` có thể xử lý sự phân chia của cây biến đổi, và các nút lá của cây biến đổi có thể là các đơn vị biến đổi (TU), và việc phân tích và xử lý đối với phần tử cú pháp liên quan đến tín hiệu dư có thể được thực hiện trong đơn vị biến đổi.

`transform_tree()` có thể nhận đầu vào của (`x0`, `y0`) mà là các tọa độ trên cùng bên trái của nút đang được xử lý, `tbWidth` là chiều rộng của nút đang được xử lý, `tbHeight` là

chiều cao của nút đang được xử lý, `treeType` là biến biểu thị kiểu của cây mã hóa đang được xử lý, và biến `chType` được tạo cấu hình theo biến `treeType`. Khi thông tin (`treeType`) biểu thị kiểu cây đang được xử lý là `SINGLE_TREE`, thì điều này có thể biểu thị rằng độ sáng và sắc độ đã được mã hóa theo cùng cấu trúc cây mã hóa. Nghĩa là, `SINGLE_TREE` có thể biểu thị rằng cây đơn được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời. Khi thông tin (`treeType`) biểu thị kiểu cây đang được xử lý là `DUAL_TREE_LUMA`, thì điều này có thể biểu thị rằng cây đang được xử lý là cây cho thành phần độ sáng khi độ sáng và sắc độ đã được mã hóa theo các cấu trúc cây mã hóa khác nhau. Nghĩa là, `DUAL_TREE_LUMA` có thể biểu thị rằng cây đôi được sử dụng để phân chia vùng phía trên bao gồm khối hiện thời, và thành phần được bao gồm trong khối hiện thời là thành phần kênh thứ nhất (LUMA). Khi `treeType` là `DUAL_TREE_CHROMA`, thì điều này có thể có nghĩa là cây đang được xử lý là cây cho thành phần sắc độ khi độ sáng và sắc độ được mã hóa theo các cấu trúc cây mã hóa khác nhau. Biến `chType` có thể được tạo cấu hình dựa trên `treeType`, và khi `treeType` là `DUAL_TREE_CHROMA`, thì `chType` có thể được tạo cấu hình là 1, và ngược lại, `chType` có thể được tạo cấu hình là 0. Khi `transform_tree()` được gọi ban đầu, (`x0`, `y0`) có thể giống như tọa độ trên cùng bên trái của đơn vị mã hóa, và `tbWidth`, `tbHeight`, `treeType` và `chType` có thể lần lượt giống như chiều rộng, chiều cao, `treeType` và `chType` của đơn vị mã hóa. Việc phân chia nút gốc của cây biến đổi có cùng kích thước với đơn vị mã hóa có thể được chia thành ba trường hợp như sau.

i) Khi chiều rộng của nút hiện thời lớn hơn số lượng tối đa (`MaxTbSizeY`) của các mẫu biến đổi độ sáng, hoặc khi chiều cao của nút hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng (`tbWidth > MaxTbSizeY || tbHeight > MaxTbSizeY`): `MaxTbSizeY` là biến biểu thị kích thước (số lượng các mẫu) của khối biến đổi lớn nhất độ sáng, và có thể được xác định theo kích thước lớn nhất của nhân biến đổi được xác định trong bộ mã hóa-giải mã. Kích thước lớn nhất tăng của nhân biến đổi có thể nói chung tăng khả năng nén năng lượng, và do đó có thể tăng hiệu quả mã hóa, nhưng kích thước của bộ đệm liên hợp của bộ mã hóa và bộ giải mã phần cứng có thể được xác định theo kích thước lớn nhất của nhân biến đổi. Do đó, kích thước lớn nhất của nhân biến đổi có thể được xác định xét theo

cả độ phức tạp và hiệu quả mã hóa. MaxTbSizeY có thể được tạo cấu hình là hằng số không đổi tương tự cho cả bộ mã hóa và bộ giải mã, và có thể được tạo cấu hình theo phần tử cú pháp được báo hiệu ở mức cao hơn. Ví dụ, thông tin về kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng có thể được bao gồm và được biểu thị trong một trong số SPS và PPS. Ví dụ, sps_max_luma_transform_size_64_flag có thể được bao gồm trong SPS và sau đó được báo hiệu. Khi sps_max_luma_transform_size_64_flag là 1, thì điều này có thể biểu thị kích thước biến đổi lớn nhất (số lượng các mẫu) của mẫu độ sáng là 64, và khi cờ này là 0, thì điều này có thể biểu thị kích thước biến đổi lớn nhất (số lượng các mẫu) của mẫu độ sáng là 32. Các biến MaxTbLog2SizeY và MaxTbSizeY có thể được tạo cấu hình như dưới đây, dựa trên sps_max_luma_transform_size_64_flag.

$$\text{MaxTbLog2SizeY} = \text{sps_max_luma_transform_size_64_flag} ? 6 : 5$$

$$\text{MaxTbSizeY} = 1 \ll \text{MaxTbLog2SizeY}$$

Biến MaxTbSizeY biểu thị kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng, và MaxTbLog2SizeY có thể giống như giá trị mà là logarit của MaxTbSizeY với cơ số 2. Theo phương án trên, được mô tả rằng thông tin về kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng được báo hiệu theo kiểu cờ ở mức SPS, và kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng được xác định là một trong số 64 hoặc 32. Tuy nhiên, sáng chế không bị giới hạn ở đó. Ví dụ, giá trị của logarit, với cơ số 2, của kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng có thể được báo hiệu trực tiếp. Theo phương án trên, MaxTbLo2SizeY và MaxTbSizeY có thể được xác định ở mức SPS, và các thông số có thể được áp dụng cho mức ảnh, tấm và phân chia mà là các mức thấp hơn của SPS.

Khi phân chia cây biến đổi dựa trên kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng, phương pháp phân chia cây biến đổi có thể không được báo hiệu rõ ràng. Trong khi đó, trong trường hợp ISP hoặc SBT, phương pháp phân chia cây biến đổi có thể được báo hiệu rõ ràng, và việc phân chia cây biến đổi dựa trên kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng có thể được thực hiện trong trường hợp trong đó ISP và SBT không được áp dụng. Do đó, việc ISP hoặc SBT có được áp dụng cho đơn

vị mã hóa hiện thời hay không có thể được xác định trong câu lệnh if đầu tiên của cấu trúc cú pháp `transform_tree` của Fig.24. Khi `IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`, thì điều này có thể biểu thị rằng ISP không được áp dụng cho đơn vị mã hóa hiện thời, và khi `IntraSubPartitionsSplitType` không phải là `ISP_NO_SPLIT`, thì điều này có thể biểu thị rằng ISP được áp dụng cho đơn vị mã hóa hiện thời. Khi cờ SBT (`cu_sbt_flag`) là 1, thì điều này có thể biểu thị rằng SBT được áp dụng cho đơn vị mã hóa hiện thời, và khi cờ SBT (`cu_sbt_flag`) là 0, thì điều này có thể biểu thị rằng SBT không được áp dụng cho đơn vị mã hóa hiện thời. Việc phân chia cây biến đổi dựa trên kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng có thể được thực hiện trong trường hợp trong đó ISP và SBT không được áp dụng. Do đó, khi, trong câu lệnh if đầu tiên, `IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`, và cờ SBT (`cu_sbt_flag`) là 0, thì việc phân chia cây biến đổi dựa trên kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng có thể được thực hiện.

Khi chiều rộng (`tbWidth`) hoặc chiều cao (`tbHeight`) của nút gốc và nút giữa được phân chia của cây biến đổi lớn hơn `MaxTbSizeY`, thì phương pháp phân chia không được báo hiệu rõ ràng và phân tích, và cây biến đổi có thể được phân chia cho đến khi cả chiều rộng và chiều cao của nút được phân chia trở nên nhỏ hơn hoặc bằng `MaxTbSizeY`. Khi cả `tbWidth` và `tbHeight` nhỏ hơn hoặc bằng `MaxTbSizeY`, thì nút tương ứng có thể trở thành đơn vị biến đổi, mà là nút lá của cây biến đổi.

Ví dụ, khi `MaxTbSizeY` là 32 đối với nút gốc 64x64 (cùng kích thước với đơn vị mã hóa), thì cây biến đổi có thể được phân chia thành bốn nút có kích thước 32x32, và nút lá có kích thước 32x32 có thể trở thành đơn vị biến đổi mà không phân chia hơn nữa. Là ví dụ khác, khi `MaxTbSizeY` là 32 đối với nút gốc 64x32, cây biến đổi có thể được phân chia thành hai nút có kích thước 32x32, và nút lá có kích thước 32x32 có thể trở thành đơn vị biến đổi mà không phân chia hơn nữa. Khi chiều rộng hoặc chiều cao của nút gốc và nút giữa được phân chia của cây biến đổi lớn hơn `MaxTbSizeY`, thì `transform_tree()` có thể được gọi đệ quy để phân chia cây biến đổi. Khi cả chiều rộng và chiều cao của nút gốc và nút giữa được phân chia của cây biến đổi nhỏ hơn hoặc bằng `MaxTbSizeY`, thì nút tương ứng có thể trở thành đơn vị biến đổi mà không phân chia hơn nữa, và việc phân tích và xử

lý đối với phần tử cú pháp liên quan đến tín hiệu dư có thể được thực hiện theo cấu trúc cú pháp `transform_unit()`.

ii) Khi SBT được áp dụng cho đơn vị mã hóa hiện thời (`cu_sbt_flag==1`): khác với trường hợp i) được mô tả ở trên, khi SBT được áp dụng, phương pháp phân chia cây biến đổi có thể được xác định thông qua báo hiệu rõ ràng của phần tử cú pháp ở mức đơn vị mã hóa.

iii) Khi ISP được áp dụng cho đơn vị mã hóa hiện thời (`IntraSubPartitionsSplitType != ISP_NO_SPLIT`): khác với trường hợp được mô tả trong chỉ số i), khi ISP được áp dụng, phương pháp phân chia cây biến đổi có thể được xác định thông qua báo hiệu rõ ràng của phần tử cú pháp ở mức đơn vị mã hóa. Khi `IntraSubPartitionsSplitType` là `ISP_HOR_SPLIT`, thì điều này có thể biểu thị sự phân chia ngang bởi ISP, và khi `IntraSubPartitionsSplitType` là `ISP_VER_SPLIT`, thì điều này có thể biểu thị sự phân chia dọc bởi ISP.

Là các trường hợp i), ii) và iii) được mô tả ở trên, sự phân chia của cây biến đổi có thể được phân loại thành trường hợp i) phân chia mà không có báo hiệu rõ ràng của phần tử cú pháp, và các trường hợp ii) và iii) biểu thị phương pháp phân chia bằng cách sử dụng báo hiệu rõ ràng của phần tử cú pháp. Trong trường hợp cấu trúc cú pháp `transform_tree()` được minh họa trên Fig.24, việc phân chia cây biến đổi được mô tả trong tất cả các trường hợp i), ii) và iii) có thể được thực hiện, và cấu trúc cú pháp `transform_tree()` của Fig.24 tóm tắt là như sau.

```
transform_tree()
{
    if(IntraSubPartitionsSplitType==ISP_NO_SPLIT && !cu_sbt_flag)
    {
        Thực hiện phân chia ngầm dựa trên kích thước khối
    }
    else if(cu_sbt_flag)
```

```

{
    Thực hiện phân chia rõ ràng dựa trên chế độ SBT
}

else if(IntraSubPartitionsSplitType==ISP_HOR_SPLIT)
{
    Thực hiện phân chia rõ ràng dựa trên chế độ ISP
}

else if(IntraSubPartitionsSplitType==ISP_VER_SPLIT)
{
    Thực hiện phân chia rõ ràng dựa trên chế độ ISP
}
}

```

Trong cấu trúc cú pháp `transform_tree()` được minh họa trên Fig.24, đầu tiên, khi `IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`, và cờ SBT (`cu_sbt_flag`) là 0, việc phân chia ngang dựa trên thông tin kích thước của nút và kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu) có thể được thực hiện. Tiếp theo, khi `IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`, và cờ SBT (`cu_sbt_flag`) là 1, thì cây biến đổi có thể được phân chia dựa trên phần tử cú pháp liên quan đến SBT được báo hiệu và được phân tích theo đơn vị mã hóa. Tiếp theo, khi `IntraSubPartitionsSplitType` không phải là `ISP_NO_SPLIT`, và cờ SBT (`cu_sbt_flag`) là 0, thì việc phân chia ngang hoặc phân chia dọc bởi ISP có thể được thực hiện dựa trên `IntraSubPartitionsSplitType`.

Đơn vị biến đổi mà là nút lá của cây biến đổi có thể bao gồm phần tử cú pháp liên quan đến phép biến đổi và hệ số biến đổi, và có thể được xử lý theo cấu trúc cú pháp `transform_unit()`. Fig.25 minh họa cấu trúc cú pháp `transform_unit()` theo một phương án của sáng chế. Cấu trúc cú pháp `transform_unit()` biểu thị cấu trúc cú pháp liên quan đến đơn vị biến đổi, và có thể nhận đầu vào là các biến `tbWidth`, `tbHeight`, `treeType`,

subTuIndex và chType. (x0, y0) có thể biểu thị vị trí của mẫu độ sáng trên cùng bên trái của khối biến đổi hiện thời, và có thể là vị trí tương đối với vị trí mẫu độ sáng trên cùng bên trái của ảnh. Các biến tbWidth và tbHeight có thể lần lượt biểu thị chiều rộng của khối biến đổi hiện thời và chiều cao của khối biến đổi hiện thời. Cấu trúc cây mã hóa cho thành phần độ sáng và cấu trúc cây mã hóa cho thành phần sắc độ có thể được mã hóa giống hoặc khác nhau, và do đó kiểu cây mã hóa đang được xử lý có thể được biểu thị bởi biến treeType. Khi treeType là SINGLE_TREE, thì điều này có nghĩa là thành phần độ sáng và thành phần sắc độ đã được mã hóa theo cùng cấu trúc cây mã hóa, và trong trường hợp này, đơn vị biến đổi có thể bao gồm khối biến đổi sắc độ theo định dạng màu sắc và khối biến đổi độ sáng. Khi treeType là DUAL_TREE_LUMA, thì điều này có thể có nghĩa là cây mã hóa đang được xử lý là cây mã hóa cho thành phần độ sáng khi thành phần độ sáng và thành phần sắc độ được mã hóa theo các cấu trúc cây mã hóa khác nhau. Trong trường hợp này, đơn vị biến đổi có thể bao gồm khối biến đổi độ sáng. Khi treeType là DUAL_TREE_CHROMA, thì điều này có thể có nghĩa là cây mã hóa đang được xử lý là cây mã hóa cho thành phần sắc độ khi thành phần độ sáng và thành phần sắc độ được mã hóa theo các cấu trúc cây mã hóa khác nhau. Trong trường hợp này, đơn vị biến đổi có thể bao gồm khối biến đổi sắc độ theo định dạng màu sắc. subTuIndex biểu thị chỉ số của đơn vị biến đổi đang được xử lý. chType có thể được tạo cấu hình dựa trên treeType, và khi treeType là DUAL_TREE_CHROMA, thì chType có thể được tạo cấu hình là 1, và ngược lại, chType có thể được tạo cấu hình là 0.

Trong cấu trúc cú pháp transform_unit(), đầu tiên, (xC, yC) mà là chỉ số cho thành phần kênh thứ hai và thành phần kênh thứ ba, và các chiều rộng và các chiều cao của khối biến đổi kênh thứ hai và khối biến đổi kênh thứ ba có thể được tạo cấu hình. (x0, y0), tbWidth, và tbHeight nhập vào transform_unit() có thể là các giá trị được tạo cấu hình dựa vào khối độ sáng. Ví dụ, tbWidth và tbHeight có thể là chiều rộng và chiều cao của khối biến đổi độ sáng, và chiều rộng (biến wC) và chiều cao (biến hC) của khối biến đổi sắc độ có thể khác với chiều rộng và chiều cao của khối biến đổi độ sáng theo định dạng màu sắc (hoặc định dạng sắc độ).

Định dạng sắc độ có thể được chia thành các kiểu đơn sắc, 4:2:0, 4:2:2 và 4:4:4 theo

tỷ lệ lấy mẫu con của mảng mẫu sắc độ. Khi lấy mẫu đơn sắc, chỉ một mảng mẫu có thể tồn tại, và có thể là mảng mẫu độ sáng. Khi lấy mẫu 4:2:0, các chiều rộng và các chiều cao của hai mảng mẫu sắc độ (kênh thứ hai và kênh thứ ba) có thể bằng một nửa chiều rộng và chiều cao của mảng mẫu độ sáng (kênh thứ nhất), một cách tương ứng. Khi lấy mẫu 4:2:2, các chiều rộng của hai mảng mẫu sắc độ (kênh thứ hai và kênh thứ ba) có thể bằng một nửa chiều rộng của mảng mẫu độ sáng (kênh thứ nhất), và các chiều cao của chúng có thể giống như chiều cao của mảng mẫu độ sáng (kênh thứ nhất). Khi lấy mẫu 4:4:4, các chiều rộng và các chiều cao của hai mảng mẫu sắc độ (kênh thứ hai và kênh thứ ba) có thể giống như chiều rộng và chiều cao của mảng mẫu độ sáng (kênh thứ nhất), một cách tương ứng. Khi lấy mẫu 4:4:4, thành phần kênh thứ nhất, thành phần kênh thứ hai và thành phần kênh thứ ba có thể được tách riêng và được xử lý độc lập, và trong trường hợp này, mỗi thành phần kênh có thể được xử lý bởi ảnh được lấy mẫu đơn sắc. Theo định dạng sắc độ, biến (SubWidthC) liên quan đến tỷ lệ lấy mẫu ngang, biến (SubHeightC) liên quan đến tỷ lệ lấy mẫu dọc, và bộ chỉ báo (ChromaArrayType) biểu thị loại mảng sắc độ được xác định theo định dạng sắc độ có thể được tạo cấu hình như sau.

- Monochrome
- ChromaArrayType=0, SubWidthC=1, SubHeightC=1
- 4:2:0
- ChromaArrayType=1, SubWidthC=2, SubHeightC=2
- 4:2:2
- ChromaArrayType=2, SubWidthC=2, SubHeightC=1
- 4:4:4
- ChromaArrayType=3, SubWidthC=1, SubHeightC=1

Trường hợp cụ thể là lấy mẫu 4:4:4, khi thành phần thứ nhất, thành phần thứ hai, và thành phần thứ ba được tách riêng và được xử lý độc lập, thì ChromaArrayType có thể được tạo cấu hình là 0 như trong trường hợp đơn sắc. Các biến SubWidthC và SubHeightC biểu thị các tỷ lệ lấy mẫu con của hướng ngang và hướng dọc, một cách tương ứng, và kích

thước của khối sắc độ có thể được suy ra dựa trên kích thước của khối độ sáng, SubWidthC và SubHeightC. Ví dụ, chiều rộng và chiều cao của khối mã hóa sắc độ lần lượt giống với giá trị được thu bằng cách chia chiều rộng (CbWidth[chType][x0][y0]) của khối mã hóa độ sáng cho SubWidthC, và giá trị được thu bằng cách chia chiều cao (CbHeight[chType][x0][y0]) của khối mã hóa độ sáng cho SubHeightC. Là ví dụ khác, chiều rộng và chiều cao của khối biến đổi sắc độ lần lượt giống với giá trị được thu bằng cách chia chiều rộng (tbWidth) của khối biến đổi độ sáng cho SubWidthC, và giá trị được thu bằng cách chia chiều cao (tbHeight) của khối biến đổi độ sáng cho SubHeightC.

Khi chế độ ISP được áp dụng cho đơn vị mã hóa hiện thời, thì đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi bởi cây biến đổi, và NumIntraSubPartitions mà là biến biểu thị số lượng của các đơn vị biến đổi được phân chia có thể được xác định dựa trên thông tin chế độ ISP và kích thước của khối mã hóa. Thậm chí khi chế độ ISP được áp dụng cho đơn vị mã hóa hiện thời, thì khối sắc độ có thể không được phân chia. Ví dụ, khi chế độ phân chia ngang ISP (ISP_HOR_SPLIT) được áp dụng cho đơn vị mã hóa có kích thước 64x64, thì đơn vị mã hóa có thể được phân chia thành bốn đơn vị biến đổi 64x16. Trong số các đơn vị biến đổi, đơn vị biến đổi thứ nhất (subTuIndex là 0), đơn vị biến đổi thứ hai (subTuIndex là 1), và đơn vị biến đổi thứ ba (subTuIndex là 2) có thể chỉ bao gồm khối biến đổi độ sáng có kích thước 64x16. Đơn vị biến đổi thứ tư (subTuIndex là 3) có thể bao gồm khối biến đổi sắc độ có kích thước không được phân chia (kích thước giống với kích thước của khối mã hóa sắc độ) và khối biến đổi độ sáng có kích thước 64x16.

Trong cấu trúc cây mà có thể bao gồm khối sắc độ (điều kiện thứ nhất, khi treeType là SINGLE_TREE), khi chế độ ISP được áp dụng (điều kiện thứ hai, IntraSubPartitionsSplitType không phải là ISP_NO_SPLIT), cho đơn vị biến đổi được phân chia cuối cùng (điều kiện thứ ba, khi subTuIndex là NumIntraSubPartitions-1), các biến xC và yC liên quan đến chỉ số của mảng sắc độ, và chiều rộng wC và chiều cao hC của mảng sắc độ có thể được tạo cấu hình như sau.

$$- \text{xC} = \text{CbPosX}[\text{chType}][\text{x0}][\text{y0}]$$

$$- \text{yC} = \text{CbPosY}[\text{chType}][\text{x0}][\text{y0}]$$

$$- wC = CbWidth[chType][x0][y0] / SubWidthC$$

$$- hC = CbHeight[chType][x0][y0] / SubHeightC$$

Điều này có thể biểu thị rằng, khi chế độ ISP được áp dụng, thì khối sắc độ không được phân chia, và khối biến đổi sắc độ có kích thước giống với kích thước của khối mã hóa sắc độ mà không được phân chia được bao gồm trong đơn vị biến đổi được phân chia cuối cùng (đơn vị biến đổi, mà có subTuIndex là NumIntraSubPartitions-1).

Khi điều kiện bất kỳ trong số điều kiện thứ nhất, điều kiện thứ hai và điều kiện thứ ba là sai, thì xC, yC, wC và hC có thể được tạo cấu hình như sau.

$$- xC = x0$$

$$- yC = y0$$

$$- wC = tbWidth / SubWidthC$$

$$- hC = tbHeight / SubHeightC$$

chromaAvailable mà là biến biểu thị sự tồn tại/vắng mặt của mảng mẫu sắc độ (thành phần thứ hai và thành phần thứ ba) có thể được tạo cấu hình là đúng khi tất cả các điều kiện thứ nhất đến điều kiện thứ ba sau là đúng.

$$- [\text{Điều kiện thứ nhất}] \text{ treeType} \neq \text{DUAL_TREE_LUMA}$$

Khi cấu trúc của cây mã hóa đang được xử lý tương ứng với cây chỉ bao gồm độ sáng (thành phần thứ nhất), thì mảng mẫu sắc độ có thể không được bao gồm.

$$- [\text{Điều kiện thứ hai}] \text{ ChromaArrayType} \neq 0$$

Khi định dạng màu sắc của hình ảnh được xử lý là đơn sắc, hoặc thành phần thứ nhất, thành phần thứ hai và thành phần thứ ba được xử lý độc lập và riêng trong hình ảnh 4:4:4, ChromaArrayType có thể được tạo cấu hình là 0, và điều này có thể biểu thị rằng sắc độ (thành phần thứ hai và thành phần thứ ba) không tồn tại.

$$- [\text{Điều kiện thứ ba}] \text{ IntraSubPartitionsSplitType} == \text{ISP_NO_SPLIT} \quad || \\ (\text{IntraSubPartitionsSplitType} \neq \text{ISP_NO_SPLIT} \quad \&\& \\ \text{subTuIndex} == \text{NumIntraSubPartitions}-1)$$

Khi chế độ ISP không được áp dụng cho đơn vị mã hóa hiện thời (IntraSubPartitionsSplitType là ISP_NO_SPLIT), hoặc khi chế độ ISP được áp dụng (IntraSubPartitionsSplitType không phải là ISP_NO_SPLIT), trong trường hợp trong đó đơn vị biến đổi đang được xử lý là đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia (subTuIndex là NumIntraSubPartitions-1), mảng mẫu sắc độ có thể được bao gồm.

Trong cấu trúc cú pháp transform_unit(), đầu tiên, phần tử cú pháp biểu thị xem khối biến đổi có bao gồm một hoặc nhiều hệ số biến đổi khác 0 hay không có thể được xử lý. Phần tử cú pháp tu_cbf_cb[xC][yC] là phần tử cú pháp biểu thị xem khối biến đổi kênh thứ hai có bao gồm một hoặc nhiều hệ số biến đổi khác 0 hay không. Khi tu_cbf_cb[xC][yC] là 1, thì điều này có thể có nghĩa là khối biến đổi kênh thứ hai bao gồm một hoặc nhiều hệ số biến đổi khác 0. Trong khi đó, khi tu_cbf_cb[xC][yC] là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ hai là 0. Các chỉ số xC và yC có thể biểu thị vị trí trên cùng bên trái (xC, yC) của khối biến đổi kênh thứ hai, và kênh thứ hai có thể là kênh Cb của không gian màu sắc YCbCr, kênh Cg của không gian màu sắc YCgCo, và kênh B của không gian màu sắc GBR (cũng có thể được gọi là không gian màu sắc RGB).

Phần tử cú pháp tu_cbf_cr[xC][yC] là phần tử cú pháp biểu thị xem khối biến đổi kênh thứ ba có bao gồm một hoặc nhiều hệ số biến đổi khác 0 hay không. Khi tu_cbf_cr[xC][yC] là 1, thì điều này có thể có nghĩa là khối biến đổi kênh thứ ba bao gồm một hoặc nhiều hệ số biến đổi khác 0. Trong khi đó, khi tu_cbf_cr[xC][yC] là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ ba là 0. Các chỉ số xC và yC có thể biểu thị vị trí trên cùng bên trái (xC, yC) của khối biến đổi kênh thứ ba, và kênh thứ ba có thể là kênh Cr của không gian màu sắc YCbCr, kênh Co của không gian màu sắc YCgCo, và kênh R của không gian màu sắc GBR.

Phần tử cú pháp tu_cbf_luma[x0][y0] là phần tử cú pháp biểu thị xem khối biến đổi kênh thứ nhất có bao gồm một hoặc nhiều hệ số biến đổi khác 0 hay không. Khi tu_cbf_luma[x0][y0] là 1, thì điều này có thể có nghĩa là khối biến đổi kênh thứ nhất bao

gồm một hoặc nhiều hệ số biến đổi khác 0. Trong khi đó, khi `tu_cbf_luma[x0][y0]` là 0, thì điều này có thể biểu thị rằng tất cả các hệ số biến đổi của khối biến đổi kênh thứ nhất là 0. Các chỉ số `x0` và `y0` có thể biểu thị vị trí mẫu kênh thứ nhất trên cùng bên trái (`x0`, `y0`) của khối biến đổi kênh thứ nhất, và có thể là vị trí tương đối với vị trí mẫu kênh thứ nhất trên cùng bên trái của ảnh. Kênh thứ nhất có thể là kênh Y độ sáng của không gian màu sắc YCbCr, kênh Y độ sáng của không gian màu sắc YCgCo và kênh G của không gian màu sắc GBR.

Theo các điều kiện từ thứ nhất đến thứ ba sau, các phần tử cú pháp `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` có thể được báo hiệu trước tiên.

- [Điều kiện thứ nhất] `(treeType==SINGLE_TREE || treeType==DUAL_TREE_CHROMA) && ChromaArrayType!=0`

Khi `treeType` là `SINGLE_TREE` hoặc `DUAL_TREE_CHROMA`, thì đơn vị hiện thời có thể bao gồm khối đối với thành phần kênh thứ hai và thành phần kênh thứ ba, và khi `ChromaArrayType` là 0, thì điều này có thể biểu thị rằng mẫu mảng kênh thứ hai và mẫu mảng kênh thứ ba không tồn tại. Do đó, trong trường hợp trong đó điều kiện thứ nhất được thỏa mãn, khi điều kiện thứ hai là đúng, hoặc điều kiện thứ ba là đúng, thì `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` có thể được báo hiệu. Khi điều kiện thứ nhất là sai, thì bộ mã hóa có thể không báo hiệu `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]`. Bộ giải mã có thể tạo cấu hình cả `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` là 0 mà không phân tích các phần tử cú pháp này từ dòng bit.

- [Điều kiện thứ hai] `IntraSubPartitionsSplitType==ISP_NO_SPLIT &&`

`!(cu_sbt_flag && ((subTuIndex==0 && cu_sbt_pos_flag) ||`

`(subTuIndex==1 && !cu_sbt_pos_flag) ))`

Điều kiện thứ hai có thể biểu thị điều kiện rằng, khi chế độ ISP không được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`), thì `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` được báo hiệu. Khi chế độ ISP không được áp dụng, thì SBT có thể được áp dụng, và khi SBT được áp dụng, thì đơn vị mã hóa có thể được phân chia thành

số lượng được tạo cấu hình trước của các đơn vị biến đổi bởi cây biến đổi, và số lượng được tạo cấu hình trước của các đơn vị biến đổi có thể là 2. Khi SBT được áp dụng (`cu_sbt_flag` là 1), như được mô tả dựa vào Fig.23, cờ vị trí SBT (`cu_sbt_pos_flag`) mà là phần tử cú pháp biểu thị đơn vị biến đổi trong đó tín hiệu dư tồn tại, trong số các đơn vị biến đổi được phân chia có thể được báo hiệu. Khi cờ vị trí SBT (`cu_sbt_pos_flag`) là 1, thì điều này biểu thị rằng các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` không tồn tại trong dòng bit đối với đơn vị biến đổi thứ nhất (đơn vị biến đổi, có `subTuIndex` là 0) trong số hai đơn vị biến đổi được phân chia, và điều này có nghĩa là tín hiệu dư có thể tồn tại trong đơn vị biến đổi thứ hai (`subTuIndex` là 1). Trong trường hợp này, bộ mã hóa có thể không báo hiệu `tu_cbf_luma[x0][y0]`, `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` đối với đơn vị biến đổi thứ nhất. Bộ giải mã có thể tạo cấu hình tất cả `tu_cbf_luma[x0][y0]`, `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` là 0 mà không phân tích các phần tử cú pháp này từ dòng bit.

Trong khi đó, khi cờ vị trí SBT (`cu_sbt_pos_flag`) là 0, thì điều này biểu thị rằng các phần tử cú pháp `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` không tồn tại trong dòng bit đối với đơn vị biến đổi thứ hai (`subTuIndex` là 1) trong số hai đơn vị biến đổi được phân chia bởi SBT, và điều này có nghĩa là tín hiệu dư có thể tồn tại trong đơn vị biến đổi thứ nhất (`subTuIndex` là 0). Trong trường hợp này, bộ mã hóa có thể không báo hiệu `tu_cbf_luma[x0][y0]`, `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` đối với đơn vị biến đổi thứ hai. Bộ giải mã có thể tạo cấu hình tất cả `tu_cbf_luma[x0][y0]`, `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` là 0 mà không phân tích các phần tử cú pháp này từ dòng bit. Do đó, điều kiện thứ hai có thể có nghĩa là, trong trường hợp trong đó chế độ ISP và SBT không được áp dụng, hoặc tín hiệu dư có thể tồn tại trong đơn vị biến đổi đang được xử lý khi SBT được áp dụng, `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` có thể được báo hiệu.

- [Điều kiện thứ ba] `IntraSubPartitionsSplitType!=ISP_NO_SPLIT &&`

`(subTuIndex==NumIntraSubPartitions-1)`

Điều kiện thứ ba có thể biểu thị điều kiện rằng, khi chế độ ISP được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` không phải là `ISP_NO_SPLIT`), thì

$tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$ được báo hiệu. Khi chế độ ISP được áp dụng cho khối hiện thời, đơn vị mã hóa có thể được phân chia thành số lượng $NumIntraSubPartitions$ của các đơn vị biến đổi bởi cây biến đổi, và khối sắc độ (kênh thứ hai và kênh thứ ba) có thể chỉ được bao gồm trong đơn vị biến đổi cuối cùng ($subTuIndex$ là $NumIntraSubPartitions-1$) trong số các đơn vị biến đổi được phân chia. Các khối sắc độ có thể không được bao gồm trong các đơn vị biến đổi khác ($subTuIndex$ lớn hơn hoặc bằng 0, và nhỏ hơn $NumIntraSubPartitions-1$), và $tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$ có thể không được báo hiệu. Trong trường hợp này, bộ mã hóa có thể không báo hiệu $tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$. Bộ giải mã có thể tạo cấu hình cả $tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$ là 0 mà không phân tích các phân tử cú pháp này từ dòng bit.

Khi điều kiện thứ nhất là đúng, và điều kiện thứ hai là đúng hoặc điều kiện thứ ba là đúng, thì $tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$ có thể được báo hiệu. Khi $tu_cbf_cb[xC][yC]$ và $tu_cbf_cr[xC][yC]$ không tồn tại, thì bộ giải mã có thể tạo cấu hình cả hai phân tử cú pháp này là 0.

Trong cấu trúc cú pháp `transform_unit()`, tiếp theo, $tu_cbf_luma[x0][y0]$ có thể được báo hiệu. Theo các điều kiện từ thứ nhất đến thứ ba sau, phân tử cú pháp $tu_cbf_luma[x0][y0]$ có thể được báo hiệu.

[Điều kiện thứ nhất] `treeType==SINGLE_TREE` ||
`treeType==DUAL_TREE_LUMA`

Khi `treeType` là `SINGLE_TREE` hoặc `DUAL_TREE_LUMA`, thì đơn vị hiện thời có thể bao gồm khối đối với thành phần kênh thứ nhất, và trong trường hợp trong đó điều kiện thứ nhất được thỏa mãn, khi điều kiện thứ hai là đúng, hoặc điều kiện thứ ba là đúng, thì $tu_cbf_luma[x0][y0]$ có thể được báo hiệu. Khi `treeType` là `DUAL_TREE_CHROMA`, thì đơn vị hiện thời có thể không bao gồm khối đối với thành phần kênh thứ nhất. Trong trường hợp này, bộ mã hóa có thể không báo hiệu $tu_cbf_luma[x0][y0]$. Bộ giải mã có thể tạo cấu hình $tu_cbf_luma[x0][y0]$ là 0 mà không phân tích phân tử cú pháp này từ dòng bit.

```

- [Điều kiện thứ hai] IntraSubPartitionsSplitType==ISP_NO_SPLIT &&
!(cu_sbt_flag && ( (subTuIndex==0 && cu_sbt_pos_flag) ||
(subTuIndex==1 && !cu_sbt_pos_flag) ) ) &&
(CuPredMode[chType][x0][y0]==MODE_INTRA ||
(chromaAvailable && (tu_cbf_cb[xC][yC] || tu_cbf_cr[xC][yC])) ||
CbWidth[chType][x0][y0]>MaxTbSizeY ||
CbHeight[chType][x0][y0]>MaxTbSizeY)

```

Điều kiện thứ hai có thể được tạo cấu hình là đúng khi điều kiện thứ (2-1) là đúng, và ít nhất một trong số điều kiện thứ (2-2) đến điều kiện thứ (2-5) là đúng.

```

- [Điều kiện thứ (2-1)] IntraSubPartitionsSplitType==ISP_NO_SPLIT &&
!(cu_sbt_flag && ( (subTuIndex==0 && cu_sbt_pos_flag) ||
(subTuIndex==1 && !cu_sbt_pos_flag) ) )

```

Điều kiện thứ (2-1) có thể có nghĩa là, trong trường hợp trong đó chế độ ISP và SBT không được áp dụng, hoặc tín hiệu dư có thể tồn tại trong đơn vị biến đổi đang được xử lý khi SBT được áp dụng, thì `tu_cbf_luma[x0][y0]` có thể được báo hiệu theo điều kiện thứ (2-2) đến điều kiện thứ (2-5).

```

- [Điều kiện thứ (2-2)] CuPredMode[chType][x0][y0]==MODE_INTRA

```

Phương pháp dự đoán có thể được xác định ở mức đơn vị mã hóa, và biến `CuPredMode[chType][x0][y0]` có thể được tạo cấu hình dựa trên thông tin được nhận ở mức đơn vị mã hóa. Phương pháp dự đoán có thể bao gồm phép dự đoán nội ảnh, phép dự đoán liên ảnh, sao chép khối nội ảnh (intra block copy, IBC), chế độ bảng màu, v.v.. Phép dự đoán IBC có thể tương ứng với phương pháp tạo ra khối dự đoán từ vùng được khôi phục của ảnh hiện thời mà đang được khôi phục. Chế độ bảng màu có thể tương ứng với phương pháp ánh xạ, với chỉ số cụ thể, giá trị điểm ảnh thường được tạo ra trong khối, báo hiệu thông tin chỉ số bởi bộ mã hóa, và khôi phục, bởi bộ giải mã, khối hiện thời, dựa trên thông tin chỉ số được phân tích từ dòng bit. Khi `CuPredMode[chType][x0][y0]` là

MODE_INTRA, thì điều này biểu thị rằng phép dự đoán nội ảnh được áp dụng cho khối hiện thời, và khi $\text{CuPredMode}[\text{chType}][x0][y0]$ là MODE_INTER, thì điều này có thể biểu thị rằng phép dự đoán liên ảnh được áp dụng cho khối hiện thời. Ngoài ra, khi $\text{CuPredMode}[\text{chType}][x0][y0]$ là MODE_IBC, thì điều này có thể biểu thị rằng phép dự đoán IBC được áp dụng cho khối hiện thời, và khi $\text{CuPredMode}[\text{chType}][x0][y0]$ là MODE_PLT, thì điều này có thể biểu thị rằng chế độ bảng màu được áp dụng cho khối hiện thời. Khi điều kiện thứ (2-1) là đúng, và phép dự đoán nội ảnh được áp dụng cho khối hiện thời ($\text{CuPredMode}[\text{chType}][x0][y0]$ là MODE_INTRA), thì $\text{tu_cbf_luma}[x0][y0]$ có thể được báo hiệu. Khi phép dự đoán liên ảnh hoặc phép dự đoán IBC được áp dụng cho khối hiện thời, thì $\text{tu_cbf_luma}[x0][y0]$ có thể được báo hiệu theo điều kiện thứ (2-3) đến điều kiện thứ (2-5).

- [Điều kiện thứ (2-3)] $\text{chromaAvailable} \&\&$

$(\text{tu_cbf_cb}[xC][yC] \parallel \text{tu_cbf_cr}[xC][yC])$

cu_cbf biểu thị xem cấu trúc cú pháp $\text{transform_tree}()$ có tồn tại hay không có thể được báo hiệu theo phương pháp dự đoán.

Ví dụ, trong trường hợp chế độ bảng màu, khối có thể được khôi phục mà không sử dụng tín hiệu dư, và do đó khi chế độ bảng màu được áp dụng cho khối hiện thời, thì cu_cbf có thể không được báo hiệu. Nghĩa là, bộ mã hóa không báo hiệu rõ ràng cu_cbf , và bộ giải mã có thể tạo cấu hình cu_cbf là 0 mà không phân tích phần tử cú pháp này từ dòng bit.

Trong trường hợp phép dự đoán nội ảnh, mẫu dự đoán được tạo ra theo chế độ dự đoán được tạo cấu hình trước, dựa trên mẫu được khôi phục xung quanh khối hiện thời, và do đó độ chính xác dự đoán có thể giảm so với phép dự đoán liên ảnh. Điều này có thể ngụ ý rằng năng lượng của tín hiệu dư tồn tại trong khối là tương đối lớn, và khả năng rằng tất cả các hệ số biến đổi được lượng tử hóa tồn tại trong khối trở thành 0 là cũng thấp. Ngoài ra, khi phép dự đoán nội ảnh được áp dụng, thì việc dự đoán và việc khôi phục có thể được thực hiện ở mức đơn vị biến đổi, và thậm chí khi tất cả các hệ số biến đổi tồn tại trong khối là 0, thì cấu trúc phân chia của cây biến đổi có thể cần được xác định. Do đó, dựa trên đặc

điểm này, khi phép dự đoán nội ảnh được áp dụng cho khối hiện thời, thì `cu_cbf` có thể không được báo hiệu. Bộ mã hóa không báo hiệu rõ ràng `cu_cbf`, và bộ giải mã có thể tạo cấu hình `cu_cbf` là 1 mà không phân tích phần tử cú pháp này từ dòng bit.

Trong trường hợp phép dự đoán liên ảnh và phép dự đoán IBC, `cu_cbf` có thể được báo hiệu theo chế độ hợp nhất và chế độ bỏ qua. Chế độ hợp nhất có thể tạo ra danh sách ứng viên hợp nhất bao gồm nhiều tập thông tin chuyển động theo phương pháp được tạo cấu hình trước, và báo hiệu chỉ số tập thông tin chuyển động được sử dụng trong danh sách ứng viên hợp nhất, và tập thông tin chuyển động có thể bao gồm vector chuyển động, chỉ số ảnh tham chiếu, v.v.. Bộ giải mã có thể phân tích cờ hợp nhất từ dòng bit, và xác định xem chế độ hợp nhất có được áp dụng cho khối hiện thời hay không. Khi chế độ hợp nhất được áp dụng cho khối hiện thời, thì bộ giải mã có thể còn phân tích chỉ số hợp nhất, và chọn tập thông tin chuyển động cụ thể được biểu thị bởi chỉ số hợp nhất trong danh sách ứng viên hợp nhất bao gồm nhiều tập thông tin chuyển động. Bộ giải mã có thể tạo ra khối dự đoán, dựa trên tập thông tin chuyển động được chọn. Chế độ bỏ qua là chế độ cụ thể của chế độ hợp nhất, và có thể là chế độ hợp nhất trong đó tín hiệu dư không tồn tại, và điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` không tồn tại. Việc chế độ bỏ qua có được áp dụng cho khối hiện thời hay không có thể được biểu thị bởi cờ bỏ qua. Chế độ bỏ qua (trường hợp trong đó cờ bỏ qua là 1 và cờ hợp nhất là 1) biểu thị rằng tín hiệu dư không tồn tại, và chế độ hợp nhất (trường hợp trong đó cờ bỏ qua là 0, và cờ hợp nhất là 1) thay vì chế độ bỏ qua biểu thị rằng tín hiệu dư tồn tại, và điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` tồn tại. Do đó, bộ mã hóa có thể không báo hiệu `cu_cbf` khi cờ hợp nhất là 1. Bộ giải mã có thể không phân tích `cu_cbf` từ dòng bit, và xác định `cu_cbf` dựa trên cờ bỏ qua. Khi cờ bỏ qua là 1, thì `cu_cbf` có thể được tạo cấu hình là 0, và khi cờ bỏ qua là 0, thì `cu_cbf` có thể được tạo cấu hình là 1. Khi chế độ hợp nhất không được áp dụng cho khối hiện thời (cờ hợp nhất là 0), thì bộ mã hóa có thể báo hiệu rõ ràng `cu_cbf`. Bộ giải mã có thể phân tích `cu_cbf` từ dòng bit, và xác định xem cấu trúc cú pháp `transform_tree()` có tồn tại hay không.

Khi `cu_cbf` là 1 đối với khối được dự đoán liên ảnh và được dự đoán IBC, thì điều này có thể có nghĩa là một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong đơn vị mã hóa

hiện thời, và có thể biểu thị rằng ít nhất một trong số `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` tồn tại theo cấu trúc cú pháp `transform_unit()` không phải là 0. Các phần tử cú pháp `tu_cbf_cb`, và `tu_cbf_cr` được báo hiệu trước `tu_cbf_luma`, và do đó khi mảng mẫu thành phần sắc độ (kênh thứ hai và kênh thứ ba) không tồn tại (`chromaAvailable` là 0), hoặc cả `tu_cbf_cb` và `tu_cbf_cr` là 0, thì `tu_cbf_luma` có thể được báo hiệu theo điều kiện thứ (2-4) và điều kiện thứ (2-5). Điều kiện thứ (2-4) và điều kiện thứ (2-5) biểu thị điều kiện rằng đơn vị mã hóa được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước lớn nhất của khối biến đổi độ sáng, và khi chiều rộng hoặc chiều cao của đơn vị mã hóa lớn hơn kích thước lớn nhất (số lượng các mẫu) của khối biến đổi độ sáng, thì đơn vị mã hóa hiện thời có thể được phân chia thành nhiều đơn vị biến đổi. Khi điều kiện thứ (2-4) và điều kiện thứ (2-5) là sai, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời bao gồm một đơn vị biến đổi, và khi `chromaAvailable` là 0, hoặc cả `tu_cbf_cb` và `tu_cbf_cr` là 0, thì `tu_cbf_luma[x0][y0]` có thể không được phân tích từ dòng bit, và có thể được tạo cấu hình là 1. Trong khi đó, khi điều kiện thứ (2-3) là đúng, (mảng mẫu sắc độ tồn tại, và ít nhất một trong số `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` là 1), thì không thể giới hạn giá trị của `tu_cbf_luma[x0][y0]`, và do đó `tu_cbf_luma[x0][y0]` có thể được báo hiệu. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

- [Điều kiện thứ (2-4)] `CbWidth[chType][x0][y0]>MaxTbSizeY`

Như được mô tả dựa vào Fig.24, khi chiều rộng của khối mã hóa độ sáng lớn hơn kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), thì đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi bởi cây biến đổi. Khi đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), `tu_cbf_luma[x0][y0]` có thể được báo hiệu. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

- [Điều kiện thứ (2-5)] `CbHeight[chType][x0][y0]>MaxTbSizeY`

Như được mô tả dựa vào Fig.24, khi chiều cao của khối mã hóa độ sáng lớn hơn

kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), thì đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi bởi cây biến đổi. Khi đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), `tu_cbf_luma[x0][y0]` có thể được báo hiệu. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

- [Điều kiện thứ ba] `IntraSubPartitionsSplitType!=ISP_NO_SPLIT &&`

`(subTuIndex<NumIntraSubPartitions-1 || !InferTuCbfLuma)`

Khi chế độ ISP được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` không phải là `ISP_NO_SPLIT`), thì phần tử cú pháp `tu_cbf_luma[x0][y0]` có thể cần là 1 đối với ít nhất một đơn vị biến đổi trong số các đơn vị biến đổi được phân chia bởi chế độ ISP. Như vậy, trong trường hợp trong đó, khi chế độ ISP được áp dụng cho đơn vị mã hóa hiện thời, các giá trị `tu_cbf_luma` của tất cả các đơn vị biến đổi được phân chia là 0, bộ mã hóa có thể không chọn chế độ ISP. Khi đơn vị biến đổi đang được xử lý không phải là đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia bởi chế độ ISP (`subTuIndex` nhỏ hơn `NumIntraSubPartitions-1`), thì bộ mã hóa có thể báo hiệu rõ ràng `tu_cbf_luma[x0][y0]`. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không. Khi đơn vị biến đổi đang được xử lý là đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia bởi chế độ ISP (`subTuIndex` là `NumIntraSubPartitions-1`), thì `tu_cbf_luma[x0][y0]` có thể được báo hiệu theo biến `InferTuCbfLuma`. Khi `InferTuCbfLuma` là 1, thì điều này có thể biểu thị rằng, đối với các đơn vị biến đổi được phân chia bởi chế độ ISP, các giá trị của `tu_cbf_luma[x0][y0]` của tất cả các đơn vị biến đổi được xử lý trước đơn vị biến đổi đang được xử lý là 0. Do đó, khi `InferTuCbfLuma` là 1 đối với đơn vị biến đổi cuối cùng, thì bộ mã hóa có thể không báo hiệu `tu_cbf_luma[x0][y0]`. Bộ giải mã có thể tạo cấu hình `tu_cbf_luma[x0][y0]` là 1 mà không phân tích phần tử cú pháp này từ dòng bit. Trong khi đó, khi `InferTuCbfLuma` là 0, thì điều này có thể biểu thị rằng, đối với các đơn vị biến đổi được phân chia bởi chế độ ISP, giá trị

của $tu_cbf_luma[x0][y0]$ của ít nhất một đơn vị biến đổi trong số các đơn vị biến đổi được xử lý trước đơn vị biến đổi đang được xử lý là 1. Do đó, không thể giới hạn giá trị của $tu_cbf_luma[x0][y0]$ của đơn vị biến đổi cuối cùng, và do đó khi $InferTuCbfLuma$ là 0 đối với đơn vị biến đổi cuối cùng, thì bộ mã hóa có thể báo hiệu $tu_cbf_luma[x0][y0]$. Bộ giải mã có thể phân tích $tu_cbf_luma[x0][y0]$ từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

Theo phương pháp báo hiệu tu_cbf_cb , tu_cbf_cr và tu_cbf_luma được mô tả ở trên, bộ giải mã có thể xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi tương ứng hay không. Khi một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi tương ứng, dựa trên các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr , bộ giải mã có thể phân tích các phần tử cú pháp liên quan đến hệ số biến đổi từ dòng bit và thu mảng hệ số biến đổi.

Fig.26 minh họa cấu trúc cú pháp $transform_unit()$ theo phương án khác của sáng chế. Phần được lược bỏ từ phần mô tả liên quan đến Fig.26 có thể tuân theo phần mô tả được đưa ra ở trên dựa vào Fig.25, và Fig.26 minh họa phương pháp khác để báo hiệu phần tử cú pháp $tu_cbf_luma[x0][y0]$.

ACT được mô tả dựa vào Fig.14 đến Fig.19 có thể biến đổi tín hiệu dư trong không gian màu sắc thứ nhất sang không gian màu sắc thứ hai, và khi tất cả cờ hệ số biến đổi thứ hai (tu_cbf_cb), cờ hệ số biến đổi thứ ba (tu_cbf_cr) và cờ hệ số biến đổi thứ nhất (tu_cbf_luma) của mỗi trong số tất cả các đơn vị biến đổi được bao gồm trong đơn vị mã hóa là 0, có thể không có tác động nào bị gây ra bởi phép biến đổi không gian màu sắc. Khi tất cả tu_cbf_cb , tu_cbf_cr và tu_cbf_luma là 0, thì điều này biểu thị rằng tất cả các hệ số biến đổi tồn tại trong khối biến đổi kênh thứ hai, khối biến đổi kênh thứ ba, và khối biến đổi kênh thứ nhất là 0, và điều này có thể biểu thị rằng tất cả các tín hiệu dư là 0. Do đó, khi ACT được áp dụng cho đơn vị mã hóa hiện thời, thì có thể thỏa mãn được rằng cu_cbf là 1, và có thể thỏa mãn được rằng ít nhất một trong số các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr không phải là 0. Dựa trên trường hợp trên, theo các điều kiện từ thứ nhất đến thứ ba sau, phần tử cú pháp $tu_cbf_luma[x0][y0]$ có thể được báo hiệu.

```
[Điều kiện thứ nhất] treeType==SINGLE_TREE ||
treeType==DUAL_TREE_LUMA
```

Khi treeType là SINGLE_TREE hoặc DUAL_TREE_LUMA, thì đơn vị hiện thời có thể bao gồm khối có thành phần kênh thứ nhất, và trong trường hợp trong đó điều kiện thứ nhất được thỏa mãn, khi điều kiện thứ hai là đúng hoặc điều kiện thứ ba là đúng, thì tu_cbf_luma[x0][y0] có thể được báo hiệu.

Khi thông tin (treeType) biểu thị kiểu cây đang được xử lý không biểu thị cây đơn (SINGLE_TREE), và thông tin (treeType) biểu thị kiểu cây đang được xử lý không biểu thị độ sáng cây đôi (DUAL_TREE_LUMA), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động không thu, từ dòng bit, cờ hệ số biến đổi thứ nhất (tu_cbf_luma[x0][y0]) biểu thị xem hệ số biến đổi khác 0 có tồn tại trong khối biến đổi được bao gồm trong khối hiện thời hay không. Thiết bị xử lý tín hiệu video có thể tạo cấu hình tu_cbf_luma[x0][y0] là giá trị định trước. Ví dụ, khi treeType là DUAL_TREE_CHROMA, thì đơn vị hiện thời có thể không bao gồm khối có thành phần kênh thứ nhất. Trong trường hợp này, bộ mã hóa có thể không báo hiệu tu_cbf_luma[x0][y0]. Bộ giải mã có thể tạo cấu hình tu_cbf_luma[x0][y0] là 0 mà không phân tích phân tử cú pháp này từ dòng bit.

```
- [Điều kiện thứ hai] IntraSubPartitionsSplitType==ISP_NO_SPLIT &&
!(cu_sbt_flag && ( (subTuIndex==0 && cu_sbt_pos_flag) ||
(subTuIndex==1 && !cu_sbt_pos_flag) ) ) &&
( (CuPredMode[chType][x0][y0]==MODE_INTRA &&
!cu_act_enabled_flag[x0][y0]) ||
(chromaAvailable && (tu_cbf_cb[xC][yC] || tu_cbf_cr[xC][yC])) ||
CbWidth[chType][x0][y0]>MaxTbSizeY ||
CbHeight[chType][x0][y0]>MaxTbSizeY)
```

Điều kiện thứ hai có thể được tạo cấu hình là đúng khi điều kiện thứ (2-1) là đúng, và ít nhất một trong số điều kiện thứ (2-2) đến điều kiện thứ (2-5) là đúng.

- [Điều kiện thứ (2-1)] IntraSubPartitionsSplitType==ISP_NO_SPLIT &&

!(cu_sbt_flag && ((subTuIndex==0 && cu_sbt_pos_flag) ||

(subTuIndex==1 && !cu_sbt_pos_flag)))

Điều kiện thứ (2-1) có thể có nghĩa là, trong trường hợp trong đó chế độ ISP và phép biến đổi khối con (SBT) không được áp dụng, hoặc tín hiệu dư có thể tồn tại trong đơn vị biến đổi đang được xử lý khi SBT được áp dụng, tu_cbf_luma[x0][y0] có thể được báo hiệu theo điều kiện thứ (2-2) đến điều kiện thứ (2-5).

- [Điều kiện thứ (2-2)] CuPredMode[chType][x0][y0]==MODE_INTRA &&

!cu_act_enabled_flag[x0][y0]

Phương pháp dự đoán có thể được xác định ở mức đơn vị mã hóa, và biến CuPredMode[chType][x0][y0] có thể được tạo cấu hình dựa trên thông tin được nhận ở mức đơn vị mã hóa. Phương pháp dự đoán có thể bao gồm phép dự đoán nội ảnh, phép dự đoán liên ảnh, sao chép khối nội ảnh (intra block copy, IBC), chế độ bảng màu, v.v.. Phép dự đoán IBC có thể tương ứng với phương pháp tạo ra khối dự đoán từ vùng được khôi phục của ảnh hiện thời mà đang được khôi phục. Chế độ bảng màu có thể tương ứng với phương pháp ánh xạ, với chỉ số cụ thể, giá trị điểm ảnh thường được tạo ra trong khối, báo hiệu thông tin chỉ số bởi bộ mã hóa, và khôi phục, bởi bộ giải mã, khối hiện thời, dựa trên thông tin chỉ số được phân tích từ dòng bit. Khi CuPredMode[chType][x0][y0] là MODE_INTRA, thì điều này biểu thị rằng phép dự đoán nội ảnh được áp dụng cho khối hiện thời, và khi CuPredMode[chType][x0][y0] là MODE_INTER, thì điều này có thể biểu thị rằng phép dự đoán liên ảnh được áp dụng cho khối hiện thời. Ngoài ra, khi CuPredMode[chType][x0][y0] là MODE_IBC, thì điều này có thể biểu thị rằng phép dự đoán IBC được áp dụng cho khối hiện thời, và khi CuPredMode[chType][x0][y0] là MODE_PLT, thì điều này có thể biểu thị rằng chế độ bảng màu được áp dụng cho khối hiện thời. Thậm chí khi phép dự đoán nội ảnh được áp dụng cho khối hiện thời, thì giá trị của tu_cbf_luma[x0][y0] có thể được giới hạn theo việc ACT có được áp dụng cho khối hiện thời hay không. Do đó, tu_cbf_luma[x0][y0] có thể được báo hiệu dựa trên chế độ dự đoán (CuPredMode[chType][x0][y0]) được áp dụng cho khối hiện thời và cò

(cu_act_enabled_flag[x0][y0]) biểu thị xem ACT có được áp dụng cho khối hiện thời hay không. Khi điều kiện thứ (2-1) là đúng, phép dự đoán nội ảnh được áp dụng cho khối hiện thời (CuPredMode[chType][x0][y0] là MODE_INTRA), và ACT không được áp dụng cho khối hiện thời (cu_act_enabled_flag[x0][y0] là 0), thì bộ mã hóa có thể báo hiệu tu_cbf_luma[x0][y0]. Bộ giải mã có thể phân tích tu_cbf_luma[x0][y0] từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

Khi phép dự đoán nội ảnh và ACT được áp dụng cho khối hiện thời, hoặc phép dự đoán liên ảnh hoặc phép dự đoán IBC được áp dụng cho khối hiện thời, thì tu_cbf_luma[x0][y0] có thể được báo hiệu theo điều kiện thứ (2-3) đến điều kiện thứ (2-5).

- [Điều kiện thứ (2-3)] chromaAvailable &&

(tu_cbf_cb[xC][yC] || tu_cbf_cr[xC][yC])

cu_cbf biểu thị xem cấu trúc cú pháp transform_tree() có tồn tại hay không có thể được báo hiệu theo phương pháp dự đoán.

Ví dụ, trong trường hợp chế độ bảng màu, khối có thể được khôi phục mà không sử dụng tín hiệu dư, và do đó khi chế độ bảng màu được áp dụng cho khối hiện thời, thì cu_cbf có thể không được báo hiệu. Nghĩa là, bộ mã hóa không báo hiệu rõ ràng cu_cbf, và bộ giải mã có thể tạo cấu hình cu_cbf là 0 mà không phân tích phần tử cú pháp này từ dòng bit.

Trong trường hợp phép dự đoán nội ảnh, mẫu dự đoán được tạo ra theo chế độ dự đoán được tạo cấu hình trước, dựa trên mẫu được khôi phục xung quanh khối hiện thời, và do đó độ chính xác dự đoán có thể giảm so với phép dự đoán liên ảnh. Điều này có thể ngụ ý rằng năng lượng của tín hiệu dư tồn tại trong khối là tương đối lớn, và khả năng rằng tất cả các hệ số biến đổi được lượng tử hóa tồn tại trong khối trở thành 0 là cũng thấp. Ngoài ra, khi phép dự đoán nội ảnh được áp dụng, thì việc dự đoán và việc khôi phục có thể được thực hiện ở mức đơn vị biến đổi, và thậm chí khi tất cả các hệ số biến đổi tồn tại trong khối là 0, thì cấu trúc phân chia của cây biến đổi có thể cần được xác định. Do đó, dựa trên đặc

điểm này, khi phép dự đoán nội ảnh được áp dụng cho khối hiện thời, thì `cu_cbf` có thể không được báo hiệu. Bộ mã hóa không báo hiệu rõ ràng `cu_cbf`, và bộ giải mã có thể tạo cấu hình `cu_cbf` là 1 mà không phân tích phần tử cú pháp này từ dòng bit.

Trong trường hợp phép dự đoán liên ảnh và phép dự đoán IBC, `cu_cbf` có thể được báo hiệu theo chế độ hợp nhất và chế độ bỏ qua. Chế độ hợp nhất có thể tạo ra danh sách ứng viên hợp nhất bao gồm nhiều tập thông tin chuyển động theo phương pháp được tạo cấu hình trước, và báo hiệu chỉ số tập thông tin chuyển động được sử dụng trong danh sách ứng viên hợp nhất, và tập thông tin chuyển động có thể bao gồm vector chuyển động, chỉ số ảnh tham chiếu, v.v.. Bộ giải mã có thể phân tích cờ hợp nhất từ dòng bit, và xác định xem chế độ hợp nhất có được áp dụng cho khối hiện thời hay không. Khi chế độ hợp nhất được áp dụng cho khối hiện thời, thì bộ giải mã có thể còn phân tích chỉ số hợp nhất, và chọn tập thông tin chuyển động cụ thể được biểu thị bởi chỉ số hợp nhất trong danh sách ứng viên hợp nhất bao gồm nhiều tập thông tin chuyển động. Bộ giải mã có thể tạo ra khối dự đoán, dựa trên tập thông tin chuyển động được chọn. Chế độ bỏ qua là chế độ cụ thể của chế độ hợp nhất, và có thể là chế độ hợp nhất trong đó tín hiệu dư không tồn tại, và điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` không tồn tại. Việc chế độ bỏ qua có được áp dụng cho khối hiện thời hay không có thể được biểu thị bởi cờ bỏ qua. Chế độ bỏ qua (trường hợp trong đó cờ bỏ qua là 1 và cờ hợp nhất là 1) biểu thị rằng tín hiệu dư không tồn tại, và chế độ hợp nhất (trường hợp trong đó cờ bỏ qua là 0, và cờ hợp nhất là 1) thay vì chế độ bỏ qua biểu thị rằng tín hiệu dư tồn tại, và điều này có thể biểu thị rằng cấu trúc cú pháp `transform_tree()` tồn tại. Do đó, bộ mã hóa có thể không báo hiệu `cu_cbf` khi cờ hợp nhất là 1. Bộ giải mã có thể không phân tích `cu_cbf` từ dòng bit, và xác định `cu_cbf`, dựa trên cờ bỏ qua. Khi cờ bỏ qua là 1, thì `cu_cbf` có thể được tạo cấu hình là 0, và khi cờ bỏ qua là 0, thì `cu_cbf` có thể được tạo cấu hình là 1. Khi chế độ hợp nhất không được áp dụng cho khối hiện thời (cờ hợp nhất là 0), thì bộ mã hóa có thể báo hiệu rõ ràng `cu_cbf`. Bộ giải mã có thể phân tích `cu_cbf` từ dòng bit, và xác định xem cấu trúc cú pháp `transform_tree()` có tồn tại hay không.

Khi `cu_cbf` là 1 đối với khối được dự đoán liên ảnh và được dự đoán IBC, thì điều này có thể có nghĩa là một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong đơn vị mã hóa

hiện thời, và có thể biểu thị rằng ít nhất một trong số `tu_cbf_luma`, `tu_cbf_cb` và `tu_cbf_cr` tồn tại theo cấu trúc cú pháp `transform_unit()` không phải là 0. Các phần tử cú pháp `tu_cbf_cb`, và `tu_cbf_cr` được báo hiệu trước `tu_cbf_luma`, và do đó khi mảng mẫu thành phần sắc độ (kênh thứ hai và kênh thứ ba) không tồn tại (`chromaAvailable` là 0), hoặc cả `tu_cbf_cb` và `tu_cbf_cr` là 0, thì `tu_cbf_luma` có thể được báo hiệu theo điều kiện thứ (2-4) và điều kiện thứ (2-5). Điều kiện thứ (2-4) và điều kiện thứ (2-5) biểu thị điều kiện rằng đơn vị mã hóa được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước lớn nhất của khối biến đổi độ sáng, và khi chiều rộng hoặc chiều cao của đơn vị mã hóa lớn hơn kích thước lớn nhất của khối biến đổi độ sáng, thì đơn vị mã hóa hiện thời có thể được phân chia thành nhiều đơn vị biến đổi. Khi điều kiện thứ (2-4) và điều kiện thứ (2-5) là sai, thì điều này có thể biểu thị rằng đơn vị mã hóa hiện thời bao gồm một đơn vị biến đổi, và khi `chromaAvailable` là 0, hoặc cả `tu_cbf_cb` và `tu_cbf_cr` là 0, thì `tu_cbf_luma[x0][y0]` có thể không được phân tích từ dòng bit, và có thể được tạo cấu hình là 1. Trong khi đó, khi điều kiện thứ (2-3) là đúng, (mảng mẫu sắc độ tồn tại, và ít nhất một trong số `tu_cbf_cb[xC][yC]` và `tu_cbf_cr[xC][yC]` là 1), thì không thể giới hạn giá trị của `tu_cbf_luma[x0][y0]`, và do đó `tu_cbf_luma[x0][y0]` có thể được báo hiệu. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

Khi ACT được áp dụng cho khối được dự đoán nội ảnh (`CuPredMode[chType][x0][y0]` là `MODE_INTRA`, và `cu_act_enabled_flag[x0][y0]` là 1), thì điều kiện thứ (2-2) là sai, và do đó `tu_cbf_luma[x0][y0]` có thể được báo hiệu theo điều kiện thứ (2-3) đến điều kiện thứ (2-5). Khi ACT được áp dụng cho khối được dự đoán nội ảnh, và điều kiện thứ (2-3) là đúng, thì không thể giới hạn giá trị của `tu_cbf_luma[x0][y0]`, và do đó bộ mã hóa có thể báo hiệu `tu_cbf_luma[x0][y0]`, và bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit. Trong khi đó, khi ACT được áp dụng cho khối được dự đoán nội ảnh, và điều kiện thứ (2-3) là sai, thì `tu_cbf_luma[x0][y0]` có thể được báo hiệu theo điều kiện thứ (2-4) và điều kiện thứ (2-5). Khi ACT được áp dụng cho khối được dự đoán nội ảnh, và tất cả điều kiện thứ (2-3) đến điều kiện thứ (2-5) là sai, thì giá trị của `tu_cbf_luma[x0][y0]` có thể bị giới hạn ở 1, và do đó bộ mã hóa có thể không báo hiệu

$tu_cbf_luma[x0][y0]$. Bộ giải mã có thể tạo cấu hình $tu_cbf_luma[x0][y0]$ là 1 mà không phân tích phần tử cú pháp này từ dòng bit. Trong trường hợp này, số lượng bit cần thiết để báo hiệu $tu_cbf_luma[x0][y0]$ có thể được tiết kiệm, và hiệu quả mã hóa do ACT có thể được cải thiện hơn nữa.

- [Điều kiện thứ (2-4)] $CbWidth[chType][x0][y0] > MaxTbSizeY$

Như được mô tả dựa vào Fig.24, khi chiều rộng của khối mã hóa độ sáng lớn hơn kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), thì đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi bởi cây biến đổi. Khi đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), $tu_cbf_luma[x0][y0]$ có thể được báo hiệu. Bộ giải mã có thể phân tích $tu_cbf_luma[x0][y0]$ từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

- [Điều kiện thứ (2-5)] $CbHeight[chType][x0][y0] > MaxTbSizeY$

Như được mô tả dựa vào Fig.24, khi chiều cao của khối mã hóa độ sáng lớn hơn kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), thì đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi bởi cây biến đổi. Khi đơn vị mã hóa có thể được phân chia thành nhiều đơn vị biến đổi, dựa trên kích thước khối biến đổi lớn nhất độ sáng (số lượng các mẫu), $tu_cbf_luma[x0][y0]$ có thể được báo hiệu. Bộ giải mã có thể phân tích $tu_cbf_luma[x0][y0]$ từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

Điều kiện thứ nhất, và điều kiện thứ (2-1) đến điều kiện thứ (2-5) đã được mô tả chi tiết. Như được mô tả ở trên, khi điều kiện thứ nhất là đúng, điều kiện thứ (2-1) là đúng, và ít nhất một trong số điều kiện thứ (2-2) đến điều kiện thứ (2-5) là đúng, thì cờ hệ số biến đổi thứ nhất ($tu_cbf_luma[x0][y0]$) có thể được thu từ dòng bit.

Tổ hợp của các điều kiện cho phép cờ hệ số biến đổi thứ nhất ($tu_cbf_luma[x0][y0]$) được thu từ dòng bit có thể thay đổi. Ví dụ, khi điều kiện thứ nhất là đúng, điều kiện thứ (2-1) là đúng, và điều kiện thứ (2-2) là đúng, thì thiết bị xử lý tín hiệu video có thể thu cờ hệ số biến đổi thứ nhất ($tu_cbf_luma[x0][y0]$) từ dòng bit.

Cụ thể hơn là, theo điều kiện thứ nhất, thiết bị xử lý tín hiệu video có thể xác định xem thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị cây đơn (SINGLE_TREE), hoặc thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị độ sáng cây đôi (DUAL_TREE_LUMA).

Khi thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị cây đơn (SINGLE_TREE), hoặc thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị độ sáng cây đôi (DUAL_TREE_LUMA), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem chế độ ISP có được áp dụng cho khối hiện thời hay không, theo điều kiện thứ (2-1).

Ngoài ra, khi chế độ ISP không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem cờ SBT (cu_sbt_flag) biểu thị xem phép biến đổi khối con (SBT) có được áp dụng cho khối hiện thời hay không biểu thị việc không áp dụng SBT. Trên Fig.26, IntraSubPartitionsSplitType giống với ISP_NO_SPLIT có thể biểu thị rằng chế độ ISP không được áp dụng cho khối hiện thời.

Ngoài ra, khi cờ SBT (cu_sbt_flag) biểu thị việc không áp dụng SBT, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem phương pháp dự đoán khối hiện thời có phải là phép dự đoán nội ảnh hay không, theo điều kiện thứ (2-2). Cờ SBT (cu_sbt_flag) biểu thị việc không áp dụng SBT có thể có nghĩa là cờ SBT (cu_sbt_flag) là 0. Trong điều kiện thứ (2-1), trong trường hợp trong đó cờ SBT (cu_sbt_flag) biểu thị việc không áp dụng SBT, thậm chí khi ((subTuIndex==0 && cu_sbt_pos_flag) || (subTuIndex==1 && !cu_sbt_pos_flag)) không được kiểm tra, !(cu_sbt_flag && ((subTuIndex==0 && cu_sbt_pos_flag) || (subTuIndex==1 && !cu_sbt_pos_flag))) có thể trở thành 1.

Ngoài ra, khi phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) có biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời hay không, theo điều kiện thứ (2-2). Trên Fig.26, CuPredMode[chType][x0][y0] giống với MODE_INTRA có thể biểu thị rằng phương

pháp dự đoán khối hiện thời là phép dự đoán nội ảnh.

Khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ dòng bit, cờ hệ số biến đổi thứ nhất (`tu_cbf_luma[x0][y0]`) biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi độ sáng được bao gồm trong khối hiện thời hay không.

Theo các phương án khác nhau của sáng chế, khi điều kiện thứ nhất là đúng, điều kiện thứ (2-1) là đúng và điều kiện thứ (2-2) là đúng, thì thiết bị xử lý tín hiệu video có thể thu cờ hệ số biến đổi thứ nhất (`tu_cbf_luma[x0][y0]`) từ dòng bit.

Ví dụ, khi thông tin (`treeType`) biểu thị kiểu cây đang được xử lý là `DUAL_TREE_LUMA` hoặc `SINGLE_TREE`, thì phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh (`CuPredMode[chType][x0][y0]==MODE_INTRA`), và chế độ phân chia con nội ảnh không được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`), thiết bị xử lý tín hiệu video có thể thu cờ hệ số biến đổi thứ nhất (`tu_cbf_luma`) từ dòng bit, dựa trên việc cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) có biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời hay không. Theo một phương án của sáng chế, việc có áp dụng SBT hay không có thể được giới hạn theo phương pháp dự đoán khối hiện thời. Ví dụ, khi phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh (`CuPredMode[chType][x0][y0]` là `MODE_INTRA`), thì SBT có thể không được áp dụng cho khối hiện thời. Nghĩa là, khi thông tin (`treeType`) biểu thị kiểu cây đang được xử lý là `DUAL_TREE_LUMA` hoặc `SINGLE_TREE`, phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh (`CuPredMode[chType][x0][y0]` là `MODE_INTRA`), và chế độ phân chia con nội ảnh không được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` là `ISP_NO_SPLIT`), thì điều này có thể biểu thị rằng SBT không được áp dụng cho khối hiện thời.

Dựa vào điều kiện thứ (2-2), khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng

cho khối hiện thời, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ hệ số biến đổi thứ nhất (`tu_cbf_luma`) từ dòng bit.

Dựa vào điều kiện thứ (2-2), khi một trong số nhiều điều kiện được thỏa mãn và cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ hệ số biến đổi thứ nhất (`tu_cbf_luma`) từ dòng bit. Các điều kiện có thể bao gồm ít nhất một trong số điều kiện thứ (2-3) đến điều kiện thứ (2-5). Nghĩa là, các điều kiện này có thể bao gồm điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ hai biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ hai (điều kiện thứ (2-3)), điều kiện rằng khối kênh thứ hai và khối kênh thứ ba tương ứng với khối hiện thời tồn tại, và cờ hệ số biến đổi thứ ba biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ ba (điều kiện thứ (2-3)), điều kiện rằng chiều rộng của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng (điều kiện thứ (2-4)) và điều kiện rằng chiều cao của khối hiện thời lớn hơn số lượng tối đa của các mẫu biến đổi độ sáng (điều kiện thứ (2-5)).

Nếu cờ hệ số biến đổi thứ nhất không được thu từ dòng bit, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động tạo cấu hình cờ hệ số biến đổi thứ nhất (`tu_cbf_luma`) để biểu thị rằng một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi kênh thứ nhất.

Sau đây, khi điều kiện thứ nhất là đúng, điều kiện thứ (2-1) là đúng, và điều kiện thứ (2-2) là đúng, thì quy trình trong đó cờ hệ số biến đổi thứ nhất (`tu_cbf_luma[x0][y0]`) được thu sẽ được mô tả chi tiết hơn.

Theo các phương án khác nhau của sáng chế, khi điều kiện thứ nhất là đúng, điều kiện thứ (2-1) là đúng và điều kiện thứ (2-2) là đúng, thì thiết bị xử lý tín hiệu video có thể thu cờ hệ số biến đổi thứ nhất (`tu_cbf_luma[x0][y0]`) từ dòng bit. Cụ thể hơn là, theo điều kiện thứ nhất, thiết bị xử lý tín hiệu video có thể xác định xem thông tin (`treeType`) biểu thị kiểu cây đang được xử lý biểu thị cây đơn (`SINGLE_TREE`), hoặc thông tin (`treeType`) biểu thị kiểu cây đang được xử lý biểu thị độ sáng cây đôi (`DUAL_TREE_LUMA`).

Khi thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị cây đơn (SINGLE_TREE), hoặc thông tin (treeType) biểu thị kiểu cây đang được xử lý biểu thị độ sáng cây đôi (DUAL_TREE_LUMA), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem chế độ ISP có được áp dụng cho khối hiện thời hay không, theo điều kiện thứ (2-1).

Ngoài ra, khi chế độ ISP không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện, theo điều kiện thứ (2-1), hoạt động xác định xem chỉ số (subTuIndex) biểu thị khối biến đổi đang được xử lý trong khối hiện thời có biểu thị đơn vị biến đổi thứ nhất hay không, và cờ vị trí SBT (cu_sbt_pos_flag) biểu thị khối biến đổi có cờ hệ số biến đổi trong số các khối biến đổi được bao gồm trong khối hiện thời có biểu thị sự vắng mặt của cờ hệ số biến đổi đối với đơn vị biến đổi thứ nhất trong số các khối biến đổi hay không. Trên Fig.26, IntraSubPartitionsSplitType giống với ISP_NO_SPLIT có thể biểu thị rằng chế độ ISP không được áp dụng cho khối hiện thời. Ngoài ra, cờ hệ số biến đổi có thể bao gồm tu_cbf_luma, tu_cbf_cb và tu_cbf_cr.

Ngoài ra, khi không phải là trường hợp trong đó chỉ số (subTuIndex) biểu thị đơn vị biến đổi đang được xử lý biểu thị đơn vị biến đổi thứ nhất, và cờ vị trí SBT (cu_sbt_pos_flag) biểu thị sự vắng mặt của cờ hệ số biến đổi đối với đơn vị biến đổi thứ nhất trong số các khối biến đổi, và không phải là trường hợp trong đó chỉ số (subTuIndex) biểu thị đơn vị biến đổi đang được xử lý biểu thị đơn vị biến đổi thứ hai, và cờ vị trí SBT (cu_sbt_pos_flag) biểu thị sự vắng mặt của cờ hệ số biến đổi đối với đơn vị biến đổi thứ hai trong số các khối biến đổi, thiết bị xử lý tín hiệu video có thể thực hiện, theo điều kiện thứ (2-2), hoạt động xác định xem phương pháp dự đoán khối hiện thời có phải là phép dự đoán nội ảnh hay không.

Ngoài ra, khi phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh, thì thiết bị xử lý tín hiệu video có thể thực hiện, theo điều kiện thứ (2-2), hoạt động xác định xem cờ kích hoạt phép biến đổi không gian màu sắc (cu_act_enabled_flag) có biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời hay không. Trên Fig.26, CuPredMode[chType][x0][y0] giống với MODE_INTRA có thể biểu thị rằng

phương pháp dự đoán khối hiện thời là phép dự đoán nội ảnh.

Khi cờ kích hoạt phép biến đổi không gian màu sắc (`cu_act_enabled_flag`) biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ dòng bit, cờ hệ số biến đổi thứ nhất (`tu_cbf_luma[x0][y0]`) biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi độ sáng được bao gồm trong khối hiện thời hay không.

- [Điều kiện thứ ba] `IntraSubPartitionsSplitType!=ISP_NO_SPLIT &&`

`(subTuIndex<NumIntraSubPartitions-1 || !InferTuCbfLuma)`

Khi thông tin (`treeType`) biểu thị kiểu cây đang được xử lý theo điều kiện thứ nhất biểu thị cây đơn (`SINGLE_TREE`), hoặc thông tin (`treeType`) biểu thị kiểu cây đang được xử lý biểu thị độ sáng cây đôi (`DUAL_TREE_LUMA`), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem chế độ ISP có được áp dụng cho khối hiện thời hay không, theo điều kiện thứ ba.

Khi chế độ ISP được áp dụng cho khối hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định xem khối biến đổi đang được xử lý được bao gồm trong khối hiện thời có phải là khối biến đổi cuối cùng trong số các khối biến đổi được phân chia theo chế độ ISP hay không, hoặc hệ số biến đổi của ít nhất một khối biến đổi trong số các khối biến đổi được xử lý trước khối biến đổi đang được xử lý có phải là 0 hay không. `IntraSubPartitionsSplitType` khác với `ISP_NO_SPLIT` có thể biểu thị rằng chế độ ISP được áp dụng cho khối hiện thời.

Khi khối biến đổi đang được xử lý được bao gồm trong khối hiện thời không phải là khối biến đổi cuối cùng trong số các khối biến đổi được phân chia theo chế độ ISP, hoặc hệ số biến đổi của ít nhất một khối biến đổi trong số các khối biến đổi được xử lý trước khối biến đổi đang được xử lý không phải là 0, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ dòng bit, cờ hệ số biến đổi thứ nhất biểu thị xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi độ sáng được bao gồm trong khối hiện thời hay không. `SubTuIndex` nhỏ hơn `NumIntraSubPartitions-1` có thể có nghĩa là khối biến đổi đang được xử lý không phải là khối biến đổi cuối cùng trong số các khối biến đổi được

phân chia theo chế độ ISP. Ngoài ra, `InferTuCbfLuma` là 0 có thể biểu thị rằng hệ số biến đổi của ít nhất một khối biến đổi trong số các khối biến đổi được xử lý trước khối biến đổi đang được xử lý không phải là 0. Liên quan đến trường hợp này, phần mô tả sẽ được đưa ra một cách chi tiết hơn dưới đây.

Khi chế độ ISP được áp dụng cho khối hiện thời (`IntraSubPartitionsSplitType` không phải là `ISP_NO_SPLIT`), thì phần tử cú pháp `tu_cbf_luma[x0][y0]` có thể cần là 1 đối với ít nhất một đơn vị biến đổi trong số các đơn vị biến đổi được phân chia bởi chế độ ISP. Như vậy, trong trường hợp trong đó, khi chế độ ISP được áp dụng cho đơn vị mã hóa hiện thời, các giá trị `tu_cbf_luma` của tất cả các đơn vị biến đổi được phân chia là 0, bộ mã hóa có thể không chọn chế độ ISP. Khi đơn vị biến đổi đang được xử lý không phải là đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia bởi chế độ ISP (`subTuIndex` nhỏ hơn `NumIntraSubPartitions-1`), thì bộ mã hóa có thể báo hiệu rõ ràng `tu_cbf_luma[x0][y0]`. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không. Khi đơn vị biến đổi đang được xử lý là đơn vị biến đổi cuối cùng trong số các đơn vị biến đổi được phân chia bởi chế độ ISP (`subTuIndex` là `NumIntraSubPartitions-1`), thì `tu_cbf_luma[x0][y0]` có thể được báo hiệu theo biến `InferTuCbfLuma`. Khi `InferTuCbfLuma` là 1, thì điều này có thể biểu thị rằng, đối với các đơn vị biến đổi được phân chia bởi chế độ ISP, các giá trị của `tu_cbf_luma[x0][y0]` của tất cả các đơn vị biến đổi được xử lý trước đơn vị biến đổi đang được xử lý là 0. Do đó, khi `InferTuCbfLuma` là 1 đối với đơn vị biến đổi cuối cùng, thì bộ mã hóa có thể không báo hiệu `tu_cbf_luma[x0][y0]`. Bộ giải mã có thể tạo cấu hình `tu_cbf_luma[x0][y0]` là 1 mà không phân tích phần tử cú pháp này từ dòng bit. Trong khi đó, khi `InferTuCbfLuma` là 0, thì điều này có thể biểu thị rằng, đối với các đơn vị biến đổi được phân chia bởi chế độ ISP, giá trị của `tu_cbf_luma[x0][y0]` của ít nhất một đơn vị biến đổi trong số các đơn vị biến đổi được xử lý trước đơn vị biến đổi đang được xử lý là 1. Do đó, không thể giới hạn giá trị của `tu_cbf_luma[x0][y0]` của đơn vị biến đổi cuối cùng, và do đó khi `InferTuCbfLuma` là 0 đối với đơn vị biến đổi cuối cùng, thì bộ mã hóa có thể báo hiệu `tu_cbf_luma[x0][y0]`. Bộ giải mã có thể phân tích `tu_cbf_luma[x0][y0]` từ dòng bit, và xác định xem một hoặc nhiều hệ

số biến đổi khác 0 có tồn tại trong khối biến đổi kênh thứ nhất hay không.

Theo phương pháp báo hiệu tu_cbf_cb , tu_cbf_cr và tu_cbf_luma được mô tả ở trên, bộ giải mã có thể xác định xem một hoặc nhiều hệ số biến đổi khác 0 có tồn tại trong khối biến đổi tương ứng hay không. Khi một hoặc nhiều hệ số biến đổi khác 0 tồn tại trong khối biến đổi tương ứng, dựa trên các phần tử cú pháp tu_cbf_luma , tu_cbf_cb và tu_cbf_cr , bộ giải mã có thể phân tích các phần tử cú pháp liên quan đến hệ số biến đổi từ dòng bit và thu mảng hệ số biến đổi.

Các phương án được mô tả ở trên của sáng chế có thể được thực hiện bằng các phương tiện khác nhau. Ví dụ, các phương án của sáng chế có thể được thực hiện bởi phần cứng, phần sụn, phần mềm hoặc sự kết hợp của chúng.

Đối với phương án thực hiện bởi phần cứng, phương pháp theo các phương án của sáng chế có thể được thực hiện bởi một hoặc nhiều bộ phận trong số các mạch tích hợp cho ứng dụng cụ thể (Application Specific Integrated Circuit, ASIC), bộ xử lý tín hiệu kỹ thuật số (Digital Signal Processor, DSP), thiết bị xử lý tín hiệu kỹ thuật số (Digital Signal Processing Device, DSPD), thiết bị logic lập trình được (Programmable Logic Device, PLD), mảng cổng lập trình được dạng trường (Field Programmable Gate Array, FPGA), bộ xử lý, bộ điều khiển, bộ vi điều khiển, bộ vi xử lý và bộ phận tương tự.

Trong trường hợp phương án thực hiện bởi phần sụn hoặc phần mềm, phương pháp theo các phương án của sáng chế có thể được thực hiện dưới dạng mô đun, thủ tục hoặc chức năng dùng để thực hiện các chức năng hoặc các hoạt động được mô tả ở trên. Mã phần mềm có thể được lưu trữ trong bộ nhớ và được điều khiển bởi bộ xử lý. Bộ nhớ có thể được định vị bên trong hoặc bên ngoài bộ xử lý, và có thể trao đổi dữ liệu với bộ xử lý bằng các phương tiện khác nhau đã biết.

Một số phương án có thể còn được thực hiện ở dạng vật ghi bao gồm các lệnh thực thi được bởi máy tính, chẳng hạn như mô đun chương trình được thực thi bởi máy tính. Vật ghi đọc được bởi máy tính có thể là vật ghi khả dụng bất kỳ có thể truy cập được bởi máy tính, và bao gồm cả vật ghi khả biến và không khả biến, vật ghi tháo lắp được và không tháo lắp được. Hơn nữa, vật ghi đọc được bởi máy tính có thể bao gồm cả vật ghi lưu trữ

máy tính và vật ghi truyền thông. Vật ghi lưu trữ máy tính bao gồm cả hai vật ghi khả biến và không khả biến, tháo lắp được và không tháo lắp được được thực hiện theo phương pháp hoặc công nghệ bất kỳ để lưu trữ thông tin chẳng hạn như các lệnh đọc được bởi máy tính, cấu trúc dữ liệu, môđun chương trình hoặc dữ liệu khác. Vật ghi truyền thông thường bao gồm các lệnh đọc được bởi máy tính, cấu trúc dữ liệu, dữ liệu khác trong tín hiệu dữ liệu được điều biến chẳng hạn như các môđun chương trình, hoặc các cơ cấu truyền dẫn khác, và bao gồm vật ghi phân phối thông tin bất kỳ.

Phần mô tả nêu trên của sáng chế chỉ nhằm mục đích minh họa, và sẽ được hiểu rằng người có hiểu biết trung bình trong lĩnh vực kỹ thuật mà sáng chế thuộc về có thể thực hiện các thay đổi đối với sáng chế mà không thay đổi các ý tưởng kỹ thuật hoặc các đặc điểm thiết yếu của sáng chế và sáng chế có thể được cải biến một cách dễ dàng ở các dạng cụ thể khác. Do đó, các phương án được mô tả ở trên chỉ là minh họa và không bị giới hạn ở tất cả các khía cạnh. Ví dụ, mỗi thành phần được mô tả dưới dạng thực thể duy nhất có thể được phân tán và thực hiện, và tương tự, các thành phần được mô tả dưới dạng được phân tán có thể còn được thực hiện dưới dạng kết hợp.

Phạm vi của sáng chế được xác định bởi các điểm yêu cầu bảo hộ kèm theo thay vì phần mô tả chi tiết ở trên, và tất cả các sự thay đổi hoặc cải biến được suy ra từ ý nghĩa và phạm vi của các điểm yêu cầu bảo hộ kèm theo và các dấu hiệu tương đương của các điểm yêu cầu bảo hộ này sẽ được hiểu là nằm trong phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã tín hiệu video, phương pháp này bao gồm các bước:

thu cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không;

thu cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được kích hoạt hay không;

phương pháp này khác biệt ở chỗ

khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị rằng chế độ bảng màu được kích hoạt, thu thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất cho chế độ bỏ qua phép biến đổi;

thu thông số lượng tử hóa được cho phép nhỏ nhất, dựa trên thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất;

thu cờ kích hoạt phép biến đổi không gian màu sắc biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không;

thu thông số lượng tử hóa được cải biến thứ nhất bằng cách cộng thêm thông số lượng tử hóa và độ lệch thông số lượng tử hóa;

thu thông số lượng tử hóa được cải biến thứ hai bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất, và

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là 0,

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là giá trị khác không,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn 0, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành 0,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn thông số lượng tử hóa được cho phép nhỏ nhất, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành thông số lượng tử hóa được cho phép nhỏ nhất,

khôi phục khối hiện thời dựa trên thông số lượng tử hóa được cải biến thứ hai.

2. Phương pháp theo điểm 1, trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là giá trị định trước cho mỗi thành phần màu sắc của khối biến đổi.

3. Phương pháp theo điểm 1, trong đó cờ kích hoạt bỏ qua phép biến đổi, cờ kích hoạt bảng màu, và thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất được thu từ ít nhất một trong số bộ thông số chuỗi (sequence parameter set, SPS), bộ thông số ảnh (picture parameter set, PPS), phần đầu ảnh và/hoặc phần đầu phân chia.

4. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước:

khi chế độ dự đoán của khối hiện thời là chế độ dự đoán nội ảnh khác với chế độ phân chia con nội ảnh, thu cờ hệ số biến đổi thứ nhất biểu thị xem một hoặc nhiều hệ số biến đổi khác không có tồn tại trong khối biến đổi độ sáng của khối hiện thời hay không dựa trên cờ kích hoạt phép biến đổi không gian màu sắc.

5. Phương pháp theo điểm 4, trong đó, khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, cờ hệ số biến đổi thứ nhất được thu.

6. Phương pháp theo điểm 4, trong đó, khi cờ kích hoạt phép biến đổi không gian màu sắc

biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời hoặc một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi sắc độ của thành phần màu sắc Cb của khối hiện thời hoặc một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi sắc độ của thành phần màu sắc Cr của khối hiện thời hoặc chiều rộng của khối hiện thời lớn hơn kích thước biến đổi độ sáng tối đa hoặc chiều cao của khối hiện thời lớn hơn kích thước biến đổi độ sáng tối đa, cờ hệ số biến đổi thứ nhất được thu.

7. Phương pháp theo điểm 4, trong đó khi cờ hệ số biến đổi thứ nhất không được thu, cờ hệ số biến đổi thứ nhất được suy ra để biểu thị rằng một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi độ sáng của khối hiện thời.

8. Thiết bị giải mã tín hiệu video, thiết bị này bao gồm bộ xử lý,

trong đó bộ xử lý được tạo cấu hình để:

thu cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng cho khối biến đổi hay không được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không,

thu cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được kích hoạt hay không,

phương pháp này khác biệt ở chỗ

khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị rằng chế độ bảng màu được kích hoạt, thu thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất cho chế độ bỏ qua phép biến đổi,

thu thông số lượng tử hóa được cho phép nhỏ nhất, dựa trên thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất,

thu cờ kích hoạt phép biến đổi không gian màu sắc biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không,

thu thông số lượng tử hóa được cải biến thứ nhất bằng cách cộng thêm thông số lượng tử hóa và độ lệch thông số lượng tử hóa,

thu thông số lượng tử hóa được cải biến thứ hai bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất, và

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là 0,

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là giá trị khác không,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn 0, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành 0,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn thông số lượng tử hóa được cho phép nhỏ nhất, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành thông số lượng tử hóa được cho phép nhỏ nhất,

khôi phục khối hiện thời dựa trên thông số lượng tử hóa được cải biến thứ hai.

9. Thiết bị theo điểm 8, trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là giá trị định trước cho mỗi thành phần màu sắc của khối biến đổi.

10. Thiết bị theo điểm 8, trong đó bộ xử lý được tạo cấu hình để:

khi chế độ dự đoán của khối hiện thời là chế độ dự đoán nội ảnh khác với chế độ phân chia con nội ảnh, thu cờ hệ số biến đổi thứ nhất biểu thị xem một hoặc nhiều hệ số biến đổi khác không có tồn tại trong khối biến đổi độ sáng của khối hiện thời hay không dựa trên cờ kích hoạt phép biến đổi không gian màu sắc.

11. Thiết bị theo điểm 10, trong đó, khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, cờ hệ số biến đổi thứ nhất được thu.

12. Thiết bị theo điểm 10, trong đó, khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời hoặc một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi sắc độ của thành phần màu sắc Cb của khối hiện thời hoặc một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi sắc độ của thành phần màu sắc Cr của khối hiện thời hoặc chiều rộng của khối hiện thời lớn hơn kích thước biến đổi độ sáng tối đa hoặc chiều cao của khối hiện thời lớn hơn kích thước biến đổi độ sáng tối đa, cờ hệ số biến đổi thứ nhất được thu.

13. Thiết bị theo điểm 10, trong đó khi cờ hệ số biến đổi thứ nhất không được thu, cờ hệ số biến đổi thứ nhất được suy ra để biểu thị rằng một hoặc nhiều hệ số biến đổi khác không tồn tại trong khối biến đổi độ sáng của khối hiện thời.

14. Vật ghi đọc được bởi máy tính lâu dài lưu trữ dòng bit, dòng bit này được giải mã bằng phương pháp giải mã,

trong đó phương pháp giải mã bao gồm các bước:

thu cờ kích hoạt bỏ qua phép biến đổi biểu thị xem cờ bỏ qua phép biến đổi biểu thị xem sự bỏ qua phép biến đổi có được áp dụng hay không cho khối biến đổi được bao gồm trong khối hiện thời có tồn tại trong cú pháp đơn vị biến đổi hay không;

thu cờ kích hoạt bảng màu biểu thị xem chế độ bảng màu có được kích hoạt hay không;

phương pháp này khác biệt ở chỗ

khi cờ kích hoạt bỏ qua phép biến đổi biểu thị rằng cờ bỏ qua phép biến đổi tồn tại trong cú pháp đơn vị biến đổi, hoặc khi cờ kích hoạt bảng màu biểu thị rằng chế độ bảng màu được kích hoạt, thu thông tin liên quan đến thông số lượng tử hóa được cho phép nhỏ nhất cho chế độ bỏ qua phép biến đổi;

thu thông số lượng tử hóa được cho phép nhỏ nhất dựa trên thông tin liên quan đến

thông số lượng tử hóa được cho phép nhỏ nhất;

thu cờ kích hoạt phép biến đổi không gian màu sắc biểu thị xem phép biến đổi không gian màu sắc có được áp dụng cho khối hiện thời hay không;

thu thông số lượng tử hóa được cải biến thứ nhất bằng cách cộng thêm thông số lượng tử hóa và độ lệch thông số lượng tử hóa;

thu thông số lượng tử hóa được cải biến thứ hai bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất, và

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc không được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là 0,

trong đó khi cờ kích hoạt phép biến đổi không gian màu sắc biểu thị rằng phép biến đổi không gian màu sắc được áp dụng cho khối hiện thời, độ lệch thông số lượng tử hóa được xác định là giá trị khác không,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi không được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn 0, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành 0,

trong đó, khi cờ bỏ qua phép biến đổi biểu thị rằng sự bỏ qua phép biến đổi được áp dụng cho khối biến đổi và thông số lượng tử hóa được cải biến thứ nhất nhỏ hơn thông số lượng tử hóa được cho phép nhỏ nhất, thông số lượng tử hóa được cải biến thứ hai được thu bằng cách cắt thông số lượng tử hóa được cải biến thứ nhất thành thông số lượng tử hóa được cho phép nhỏ nhất,

khôi phục khối hiện thời, dựa trên thông số lượng tử hóa được cải biến thứ hai.

1/23

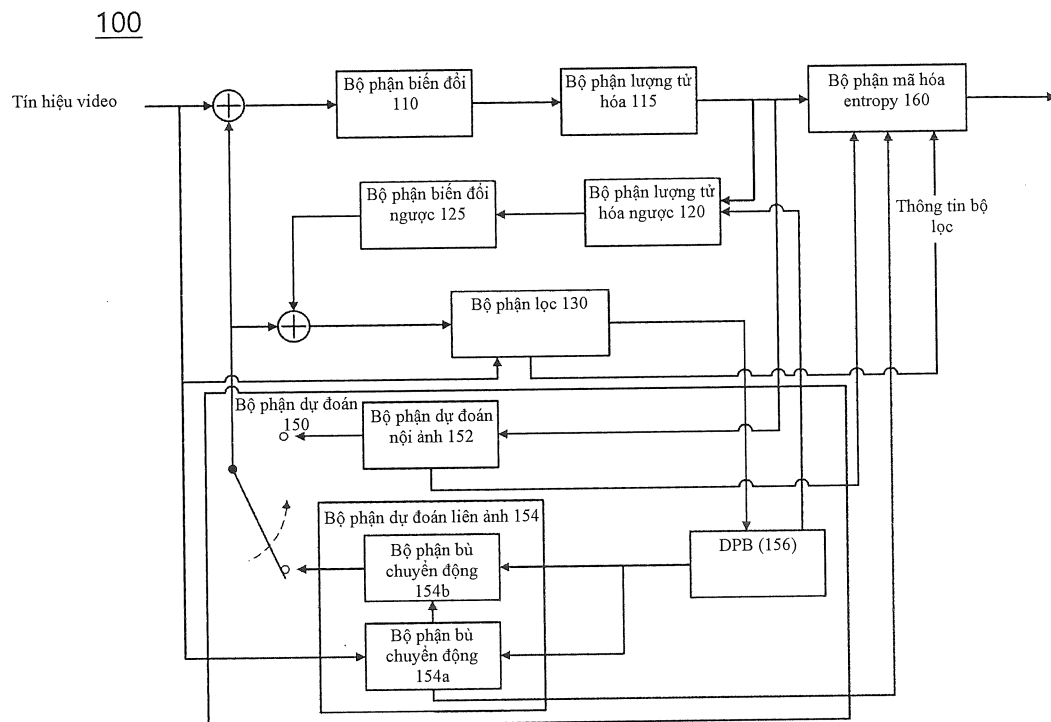


Fig. 1

2/23

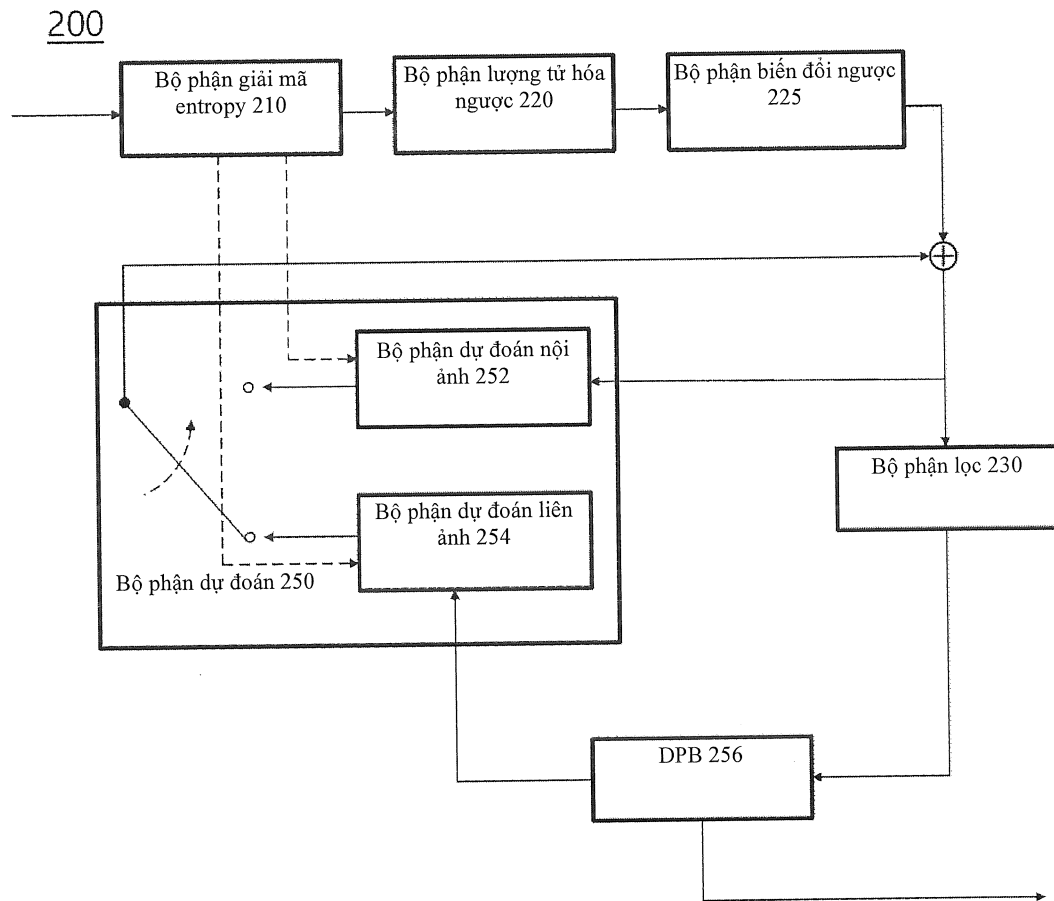
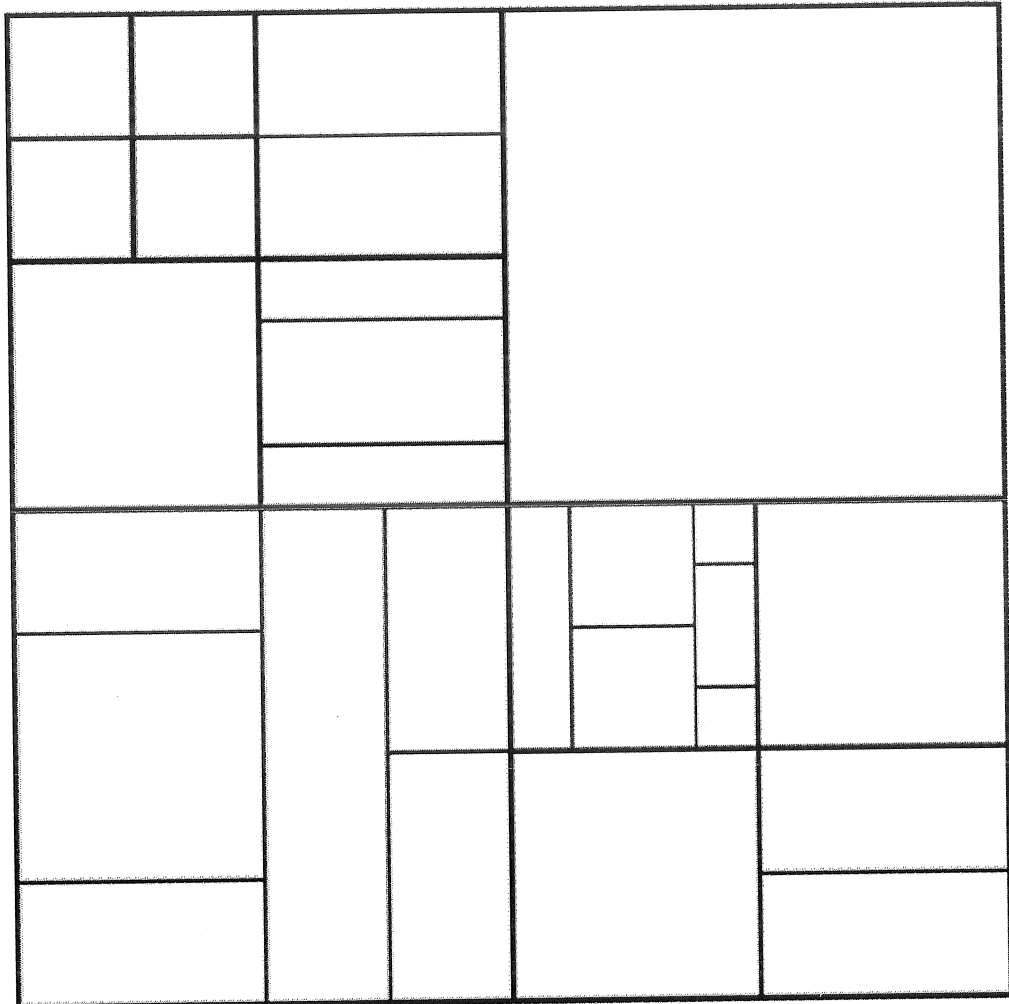
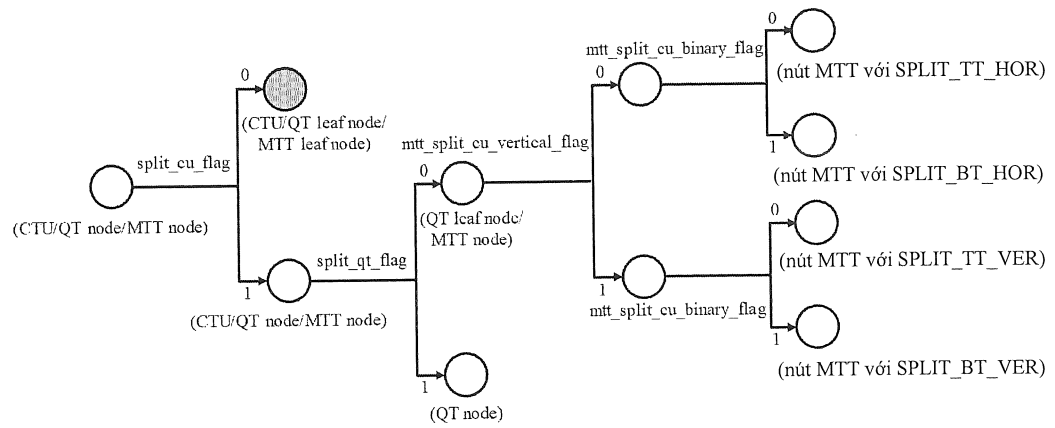


Fig. 2

3/23

*Fig. 3*

4/23

*Fig. 4*

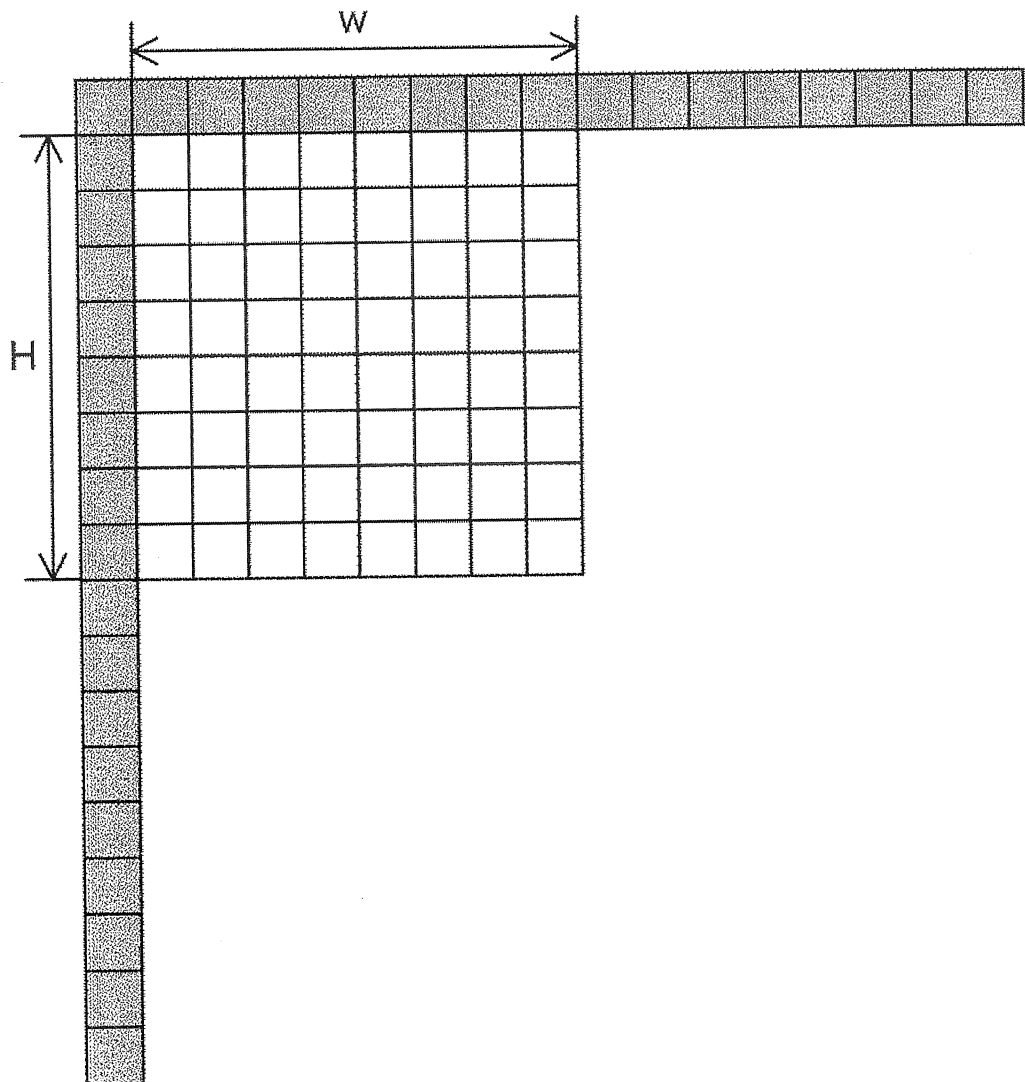
5/23



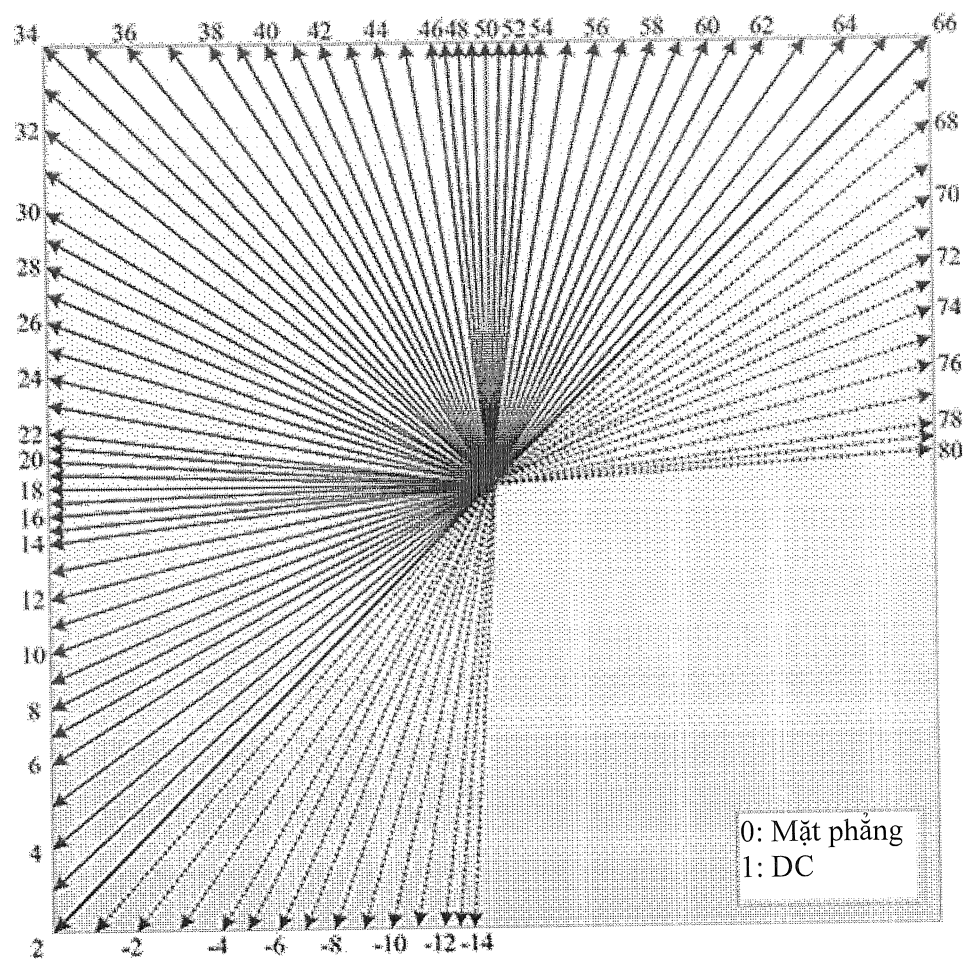
MẪU THAM CHIỀU



MẪU CỦA ĐƠN VỊ HIỆN THỜI

*Fig. 5*

6/23

*Fig. 6*

7/23

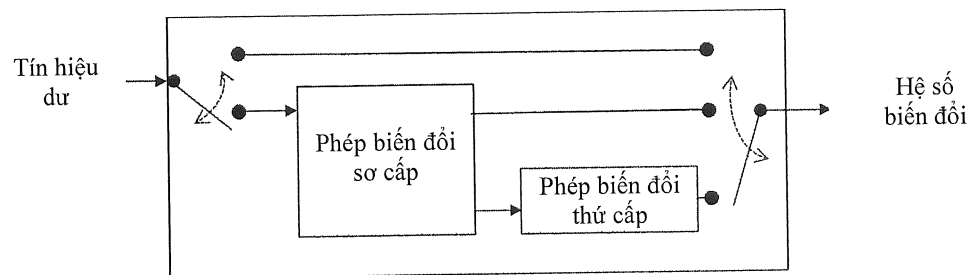


Fig. 7

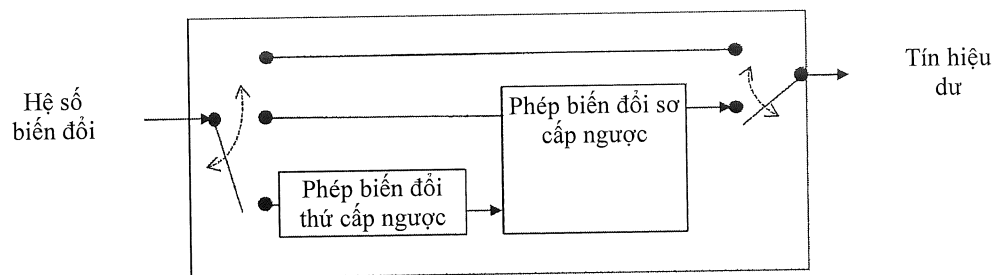


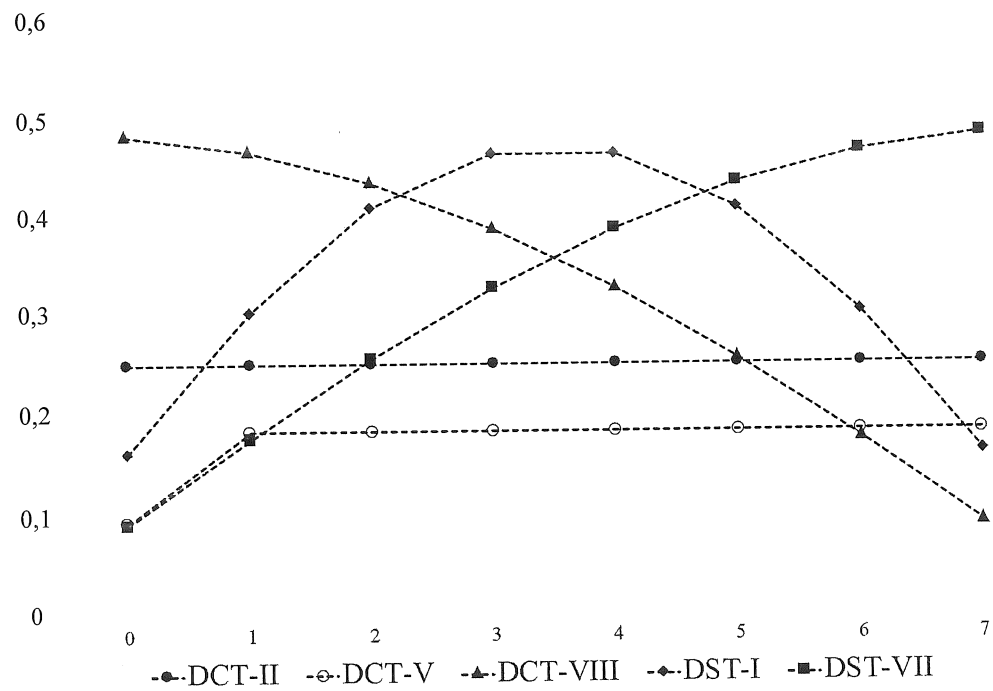
Fig. 8

8/23

Kiểu phép biến đổi	Hàm cơ sở $T_i(j), i, j=0, 1, \dots, N-1$
DCT-II	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot i \cdot (2j+1)}{2N}\right)$ trong đó $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}$
DCT-V	$T_i(j) = \omega_0 \cdot \omega_1 \cdot \sqrt{\frac{2}{2N-1}} \cdot \cos\left(\frac{2\pi \cdot i \cdot j}{2N-1}\right),$ trong đó $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}, \omega_1 = \begin{cases} \sqrt{\frac{2}{N}} & j = 0 \\ 1 & j \neq 0 \end{cases}$
DCT-VIII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \cos\left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N+2}\right)$
DST-I	$T_i(j) = \sqrt{\frac{2}{N+1}} \cdot \sin\left(\frac{\pi \cdot (i+1) \cdot (j+1)}{N+1}\right)$
DST-VII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \sin\left(\frac{\pi \cdot (2i+1) \cdot (j+1)}{2N+1}\right)$

Fig. 9

9/23

*Fig. 10*

10/23

(a)

Kiểu phép biến đổi	Hàm cơ sở $T_i(j)$, $i, j=0, 1, \dots, N-1$
DCT-IV	$T_i(j) = \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N}\right)$
DST-IV	$T_i(j) = \sqrt{\frac{2}{N}} \cdot \sin\left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N}\right)$

(b)

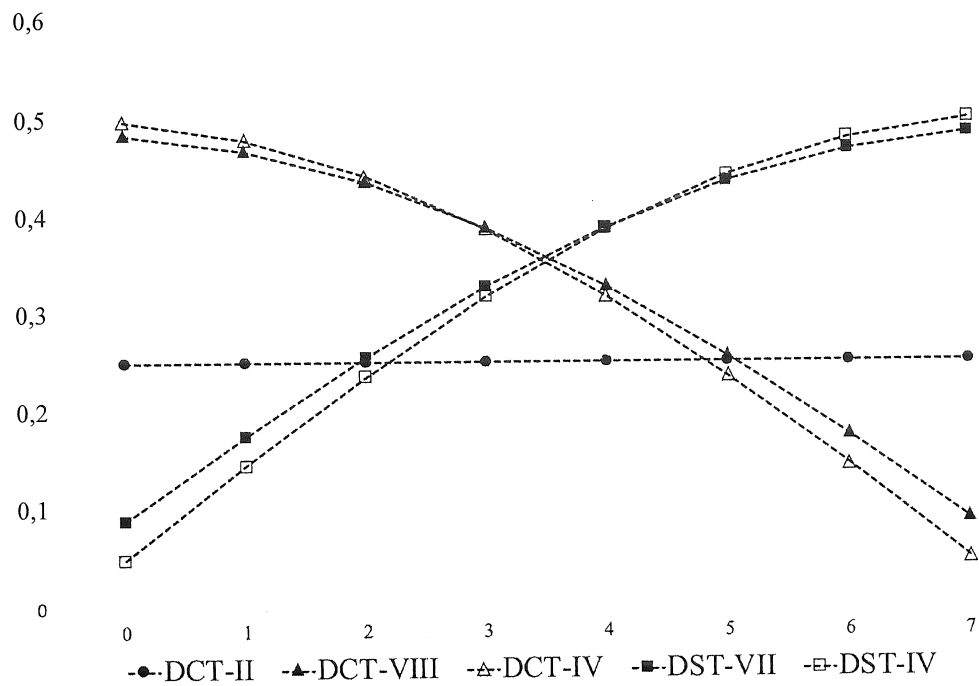


Fig. 11

11/23

(a)

Chế độ nội ảnh	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
V	2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0
H	2	1	0	1	0	1	0	1	0	1	0	1	0	1	2	2	2	2
Chế độ nội ảnh	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	
V	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	
H	2	2	2	2	2	1	0	1	0	1	0	1	0	1	0	1	0	
Chế độ nội ảnh	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52
V	1	0	1	0	1	0	1	0	1	0	1	2	2	2	2	2	2	2
H	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
Chế độ nội ảnh	53	54	55	56	57	58	59	60	61	62	63	64	65	66				
V	2	2	1	0	1	0	1	0	1	0	1	0	1	0				
H	0	0	1	0	1	0	1	0	1	0	1	0	1	0				

(b)

Bộ phép biến đổi	Các ứng viên biến đổi
0	DST-VII, DCT-VIII
1	DST-VII, DST-I
2	DST-VII, DCT-V

(c)

Bộ phép biến đổi	Các ứng viên biến đổi
0	DCT-VIII, DST-VII

Fig. 12

12/23

(a)

Chế độ nội ảnh	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
V	2	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0
H	2	1	0	1	0	1	0	1	0	1	0	1	0	1	2	2	2	2
Chế độ nội ảnh	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	
V	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	
H	2	2	2	2	2	1	0	1	0	1	0	1	0	1	0	1	0	
Chế độ nội ảnh	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52
V	1	0	1	0	1	0	1	0	1	0	1	2	2	2	2	2	2	2
H	1	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
Chế độ nội ảnh	53	54	55	56	57	58	59	60	61	62	63	64	65	66				
V	2	2	1	0	1	0	1	0	1	0	1	0	1	0				
H	0	0	1	0	1	0	1	0	1	0	1	0	1	0				

(b)

Bộ phép biến đổi	Các ứng viên biến đổi
0	DST-VII, DCT-VIII
1	DST-VII, DST-I
2	DST-VII, DCT-V

(c)

Bộ phép biến đổi	Các ứng viên biến đổi
0	DCT-VIII, DST-VII

Fig. 13

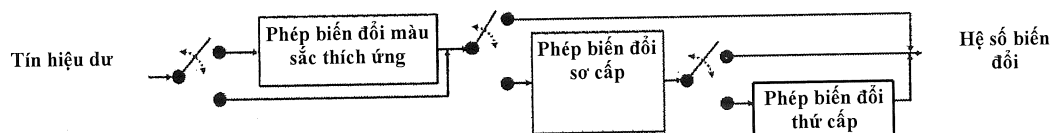


Fig. 14

13/23

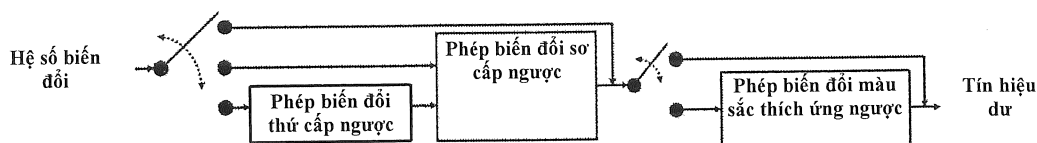


Fig. 15

Quy trình thay đổi tỷ lệ đối với các hệ số biến đổi

Các đầu vào của quy trình này là:

- vị trí độ sáng ($xTbY$, $yTbY$) xác định mẫu trên cùng bên trái của khối biến đổi độ sáng hiện thời so với mẫu độ sáng trên cùng bên trái của ảnh hiện thời,
- biến $nTbW$ xác định chiều rộng của khối biến đổi,
- biến $nTbH$ xác định chiều cao của khối biến đổi,
- biến $predMode$ xác định chế độ dự đoán của đơn vị mã hóa,
- biến $cIdx$ xác định thành phần màu sắc của khối hiện thời.

Đầu ra của quy trình này là mảng $(nTbW) \times (nTbH)$ d của các hệ số biến đổi được thay đổi tỷ lệ với các phần tử $d[x][y]$.

Thông số lượng tử hóa qP và biến $qpActOffset$ được suy ra như sau:

- Nếu $cIdx$ bằng 0, các giá trị sau áp dụng:
 $qP = Qp'Y$
 $qpActOffset = -5$
- Nếu không thì, nếu $TuCRMode[xTbY][yTbY]$ bằng 2, các giá trị sau áp dụng:
 $qP = Qp'CbCr$
 $qpActOffset = -5$
- Nếu không thì, nếu $cIdx$ bằng 1, các giá trị sau áp dụng:
 $qP = Qp'Cb$
 $qpActOffset = -5$
- Nếu không thì ($cIdx$ bằng 2), các giá trị sau áp dụng:
 $qP = Qp'Cr$
 $qpActOffset = -3$

Thông số lượng tử hóa qP được sửa đổi và các biến $rectNonTsFlag$ và $bdShift$ được suy ra như sau:

- Nếu $transform_skip_flag[xTbY][yTbY][cIdx]$ bằng 0, các giá trị sau áp dụng:
 $qP = \text{Max}(0, qP + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$
 $rectNonTsFlag = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1) ? 1:0)$
 $bdShift = \text{BitDepth} + rectNonTsFlag + ((\text{Log2}(nTbW) + \text{Log2}(nTbH))/2) - 5 + pic_dep_quant_enabled_flag$
- Nếu không thì, nếu $transform_skip_flag[xTbY][yTbY][cIdx]$ bằng 1, các giá trị sau áp dụng:
 $qP = \text{Max}(0, \text{Max}(QpPrimeTsMin, qP) + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$
 $rectNonTsFlag = 0$
 $bdShift = 10$

Biến $bdOffset$ được suy ra như sau:

$$bdOffset = (1 \ll bdShift) >> 1$$

Danh sách $levelScale[][]$ được xác định là $levelScale[j][k] = \{ \{40, 45, 51, 57, 64, 72\}, \{57, 64, 72, 80, 90, 102\} \}$ với $j = 0..1$, $k = 0..5$

Mảng $(nTbW) \times (nTbH)$ dz được thiết lập bằng mảng $(nTbW) \times (nTbH)$ $TransCoeffLevel[xTbY][yTbY][cIdx]$.

Fig. 16

14/23

Quy trình thay đổi tỷ lệ đổi với các hệ số biến đổi

Các đầu vào của quy trình này là:

- vị trí độ sáng (xTbY, yTbY) xác định mẫu trên cùng bên trái của khối biến đổi độ sáng hiện thời so với mẫu độ sáng trên cùng bên trái của ảnh hiện thời,
- biến nTbW xác định chiều rộng của khối biến đổi,
- biến nTbH xác định chiều cao của khối biến đổi,
- biến predMode xác định chế độ dự đoán của đơn vị mã hóa,
- biến cIdx xác định thành phần màu sắc của khối hiện thời.

Đầu ra của quy trình này là mảng (nTbW)x(nTbH) d của các hệ số biến đổi được thay đổi tỷ lệ với các phần tử d[x][y].

Thông số lượng tử hóa qP và biến qpActOffset được suy ra như sau:

- Nếu cIdx bằng 0, các giá trị sau áp dụng:
 $qP = Qp'Y$
 $qpActOffset = -5$
- Nếu không thì, nếu TuCResMode[xTbY][yTbY] bằng 2, các giá trị sau áp dụng:
 $qP = Qp'CbCr$
 $qpActOffset = -5$
- Nếu không thì, nếu cIdx bằng 1, các giá trị sau áp dụng:
 $qP = Qp'Cb$
 $qpActOffset = -5$
- Nếu không thì (cIdx bằng 2), các giá trị sau áp dụng:
 $qP = Qp'Cr$
 $qpActOffset = -3$

Thông số lượng tử hóa qP được sửa đổi và các biến rectNonTsFlag và bdShift được suy ra như sau:

- Nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 0, các giá trị sau áp dụng:
 $qP = \text{Max}(0, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY]?qpActOffset:0))$
 $\text{rectNonTsFlag} = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1)?1:0$
 $\text{bdShift} = \text{BitDepth} + \text{rectNonTsFlag} + ((\text{Log2}(nTbW) + \text{Log2}(nTbH))/2) - 5 + \text{pic_dep_quant_enabled_flag}$
- Nếu không thì, nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 1, các giá trị sau áp dụng:
 $qP = \text{Max}(QpPrimeTsMin, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY]?qpActOffset:0))$
 $\text{rectNonTsFlag} = 0$
 $\text{bdShift} = 10$

Biến bdOffset được suy ra như sau:

$$\text{bdOffset} = (1 \ll \text{bdShift}) >> 1$$

Danh sách levelScale[][] được xác định là levelScale[j][k] = {{40, 45, 51, 57, 64, 72}, {57, 64, 72, 80, 90, 102}} với j = 0..1, k = 0..5

Mảng (nTbW)x(nTbH) dz được thiết lập bằng mảng (nTbW)x(nTbH) TransCoeffLevel[xTbY][yTbY][cIdx].

Fig. 17

15/23

Quy trình thay đổi tỷ lệ đối với các hệ số biến đổi

Các đầu vào của quy trình này là:

- vị trí độ sáng (xTbY, yTbY) xác định mẫu trên cùng bên trái của khối biến đổi độ sáng hiện thời so với mẫu độ sáng trên cùng bên trái của ảnh hiện thời,
- biến nTbW xác định chiều rộng của khối biến đổi,
- biến nTbH xác định chiều cao của khối biến đổi,
- biến predMode xác định chế độ dự đoán của đơn vị mã hóa,
- biến cIdx xác định thành phần màu sắc của khối hiện thời.

Đầu ra của quy trình này là mảng (nTbW)x(nTbH) d của các hệ số biến đổi được thay đổi tỷ lệ với các phần tử d[x][y].

Thông số lượng tử hóa qP và biến qpActOffset được suy ra như sau:

- Nếu cIdx bằng 0, các giá trị sau áp dụng:
 $qP = Qp'Y$
 $qpActOffset = -5$
- Nếu không thì, nếu TuCResMode[xTbY][yTbY] bằng 2, các giá trị sau áp dụng:
 $qP = Qp'CbCr$
 $qpActOffset = -5$
- Nếu không thì, nếu cIdx bằng 1, các giá trị sau áp dụng:
 $qP = Qp'Cb$
 $qpActOffset = -5$
- Nếu không thì (cIdx bằng 2), các giá trị sau áp dụng:
 $qP = Qp'Cr$
 $qpActOffset = -3$

Thông số lượng tử hóa qP được sửa đổi và các biến rectNonTsFlag và bdShift được suy ra như sau:

- Nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 0, các giá trị sau áp dụng:
 $qP = \text{Max}(QpBdOffset, qP + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$
 $rectNonTsFlag = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1) ? 1 : 0)$
 $bdShift = \text{BitDepth} + rectNonTsFlag + ((\text{Log2}(nTbW) + \text{Log2}(nTbH))/2) - 5 + pic_dep_quant_enabled_flag$
- Nếu không thì, nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 1, các giá trị sau áp dụng:
 $qP = \text{Max}(QpBdOffset, \text{Max}(QpPrimeTsMin, qP) + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$
 $rectNonTsFlag = 0$
 $bdShift = 10$

Biến bdOffset được suy ra như sau:

$$bdOffset = (1 \ll bdShift) \gg 1$$

Danh sách levelScale[][] được xác định là levelScale[j][k] = {{40, 45, 51, 57, 64, 72}, {57, 64, 72, 80, 90, 102}} với j = 0..1, k = 0..5

Mảng (nTbW)x(nTbH) dz được thiết lập bằng mảng (nTbW)x(nTbH) TransCoeffLevel[xTbY][yTbY][cIdx].

Fig. 18

16/23

Quy trình thay đổi tỷ lệ đối với các hệ số biến đổi

Các đầu vào của quy trình này là:

- vị trí độ sáng (xTbY, yTbY) xác định mẫu trên cùng bên trái của khối biến đổi độ sáng hiện thời so với mẫu độ sáng trên cùng bên trái của ảnh hiện thời,
- biến nTbW xác định chiều rộng của khối biến đổi,
- biến nTbH xác định chiều cao của khối biến đổi,
- biến predMode xác định chế độ dự đoán của đơn vị mã hóa,
- biến cIdx xác định thành phần màu sắc của khối hiện thời.

Đầu ra của quy trình này là mảng (nTbW)x(nTbH) d của các hệ số biến đổi được thay đổi tỷ lệ với các phần tử d[x][y].

Thông số lượng tử hóa qP và biến qpActOffset được suy ra như sau:

- Nếu cIdx bằng 0, các giá trị sau áp dụng:
 $qP = Qp'Y$
 $qpActOffset = -5$
- Nếu không thì, nếu TuCResMode[xTbY][yTbY] bằng 2, các giá trị sau áp dụng:
 $qP = Qp'CbCr$
 $qpActOffset = -5$
- Nếu không thì, nếu cIdx bằng 1, các giá trị sau áp dụng:
 $qP = Qp'Cb$
 $qpActOffset = -5$
- Nếu không thì (cIdx bằng 2), các giá trị sau áp dụng:
 $qP = Qp'Cr$
 $qpActOffset = -3$

Thông số lượng tử hóa qP được sửa đổi và các biến rectNonTsFlag và bdShift được suy ra như sau:

- Nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 0, các giá trị sau áp dụng:
 $qP = \text{Max}(QpBdOffset, qP + (cu_act_enabled_flag[xTbY][yTbY]?qpActOffset:0))$
 $rectNonTsFlag = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1)?1:0)$
 $bdShift = \text{BitDepth} + rectNonTsFlag + ((\text{Log2}(nTbW) + \text{Log2}(nTbH))/2) - 5 + pic_dep_quant_enabled_flag$
- Nếu không thì, nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 1, các giá trị sau áp dụng:
 $qP = \text{Max}(QpPrimeTsMin, qP + (cu_act_enabled_flag[xTbY][yTbY][cIdx]?qpActOffset:0))$
 $rectNonTsFlag = 0$
 $bdShift = 10$

Biến bdOffset được suy ra như sau:

$$bdOffset = (1 \ll bdShift) \gg 1$$

Danh sách levelScale[][] được xác định là levelScale[j][k] = {{40, 45, 51, 57, 64, 72}, {57, 64, 72, 80, 90, 102}}

với j = 0..1, k = 0..5

Mảng (nTbW)x(nTbH) dz được thiết lập bằng mảng (nTbW)x(nTbH) TransCoeffLevel[xTbY][yTbY][cIdx].

Fig. 19

17/23

Quy trình thay đổi tỷ lệ đối với các hệ số biến đổi

Các đầu vào của quy trình này là:

- vị trí độ sáng (xTbY, yTbY) xác định mẫu trên cùng bên trái của khối biến đổi độ sáng hiện thời so với mẫu độ sáng trên cùng bên trái của ảnh hiện thời,
- biến nTbW xác định chiều rộng của khối biến đổi,
- biến nTbH xác định chiều cao của khối biến đổi,
- biến predMode xác định chế độ dự đoán của đơn vị mã hóa,
- biến cIdx xác định thành phần màu sắc của khối hiện thời.

Đầu ra của quy trình này là mảng (nTbW)x(nTbH) d của các hệ số biến đổi được thay đổi tỷ lệ với các phần tử d[x][y].

Thông số lượng tử hóa qP và biến qpActOffset được suy ra như sau:

- Nếu cIdx bằng 0, các giá trị sau áp dụng:
 $qP = Qp'Y$
 $qpActOffset = -5$
- Nếu không thì, nếu TuCResMode[xTbY][yTbY] bằng 2, các giá trị sau áp dụng:
 $qP = Qp'CbCr$
 $qpActOffset = -5$
- Nếu không thì, nếu cIdx bằng 1, các giá trị sau áp dụng:
 $qP = Qp'Cb$
 $qpActOffset = -5$
- Nếu không thì (cIdx bằng 2), các giá trị sau áp dụng:
 $qP = Qp'Cr$
 $qpActOffset = -3$

Thông số lượng tử hóa qP được sửa đổi và các biến rectNonTsFlag và bdShift được suy ra như sau:

- Nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 0, các giá trị sau áp dụng:
 $qP = \text{Max}(0, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? qpActOffset : 0))$
 $\text{rectNonTsFlag} = (((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1) ? 1 : 0)$
 $\text{bdShift} = \text{BitDepth} + \text{rectNonTsFlag} + ((\text{Log2}(nTbW) + \text{Log2}(nTbH)) / 2) - 5 + \text{pic_dep_quant_enabled_flag}$
- Nếu không thì, nếu transform_skip_flag[xTbY][yTbY][cIdx] bằng 1, các giá trị sau áp dụng:
 - Nếu cu_act_enabled_flag[xTbY][yTbY] bằng 1, giá trị sau áp dụng:
 $qP = \text{Max}(0, qP + qpActOffset)$
 - Nếu không thì (cu_act_enabled_flag[xTbY][yTbY] bằng 0), giá trị sau áp dụng:
 $qP = \text{Max}(QpPrimeTsMin, qP)$
 $\text{rectNonTsFlag} = 0$
 $\text{bdShift} = 10$

Biến bdOffset được suy ra như sau:

$$\text{bdOffset} = (1 \ll \text{bdShift}) \gg 1$$

Danh sách levelScale[][] được xác định là levelScale[j][k] = { {40, 45, 51, 57, 64, 72}, {57, 64, 72, 80, 90, 102} }
 với j = 0..1, k = 0..5

Mảng (nTbW)x(nTbH) dz được thiết lập bằng mảng (nTbW)x(nTbH) TransCoeffLevel[xTbY][yTbY][cIdx].

Fig. 20

18/23

(a)

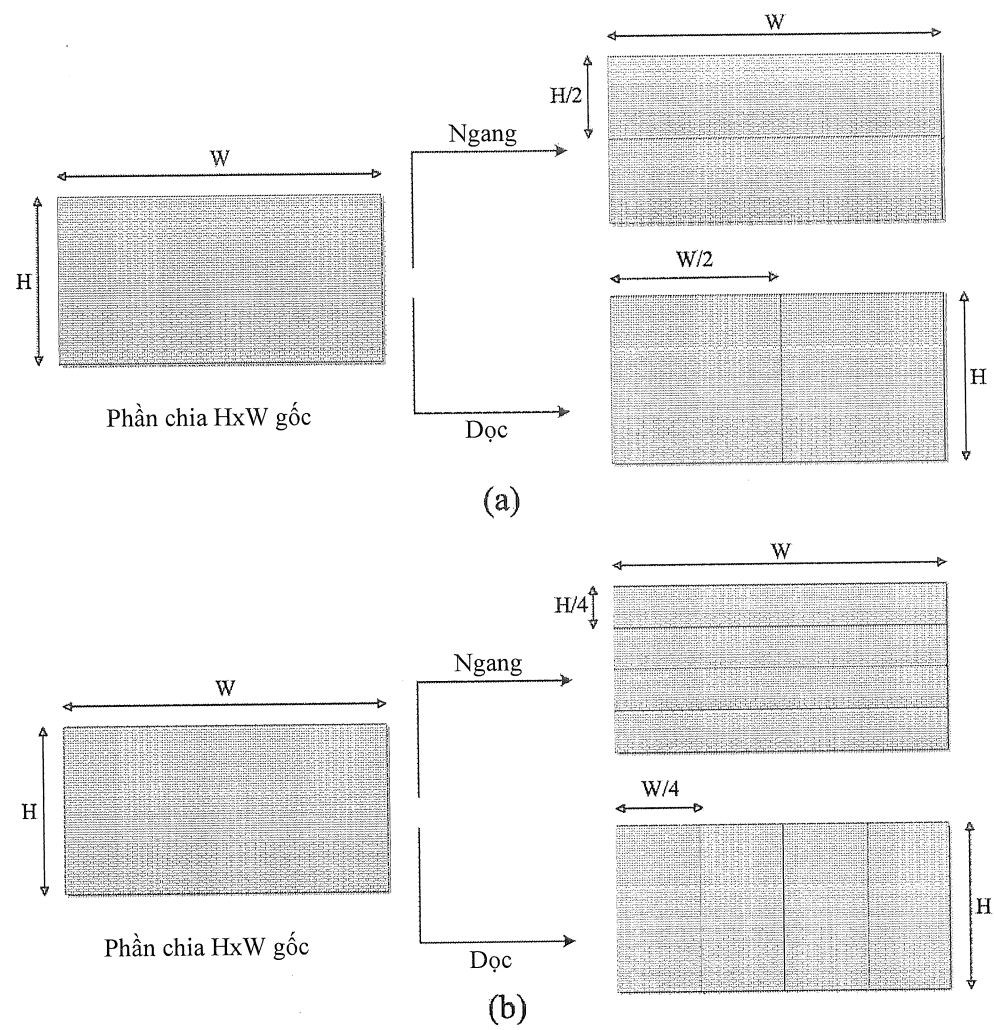
seq_parameter_set_rbsp() {	Bộ mô tả
...	
chroma_format_idc	u(2)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
...	
bit_depth_minus8	ue(v)
min_qp_prime_ts_minus4	ue(v)
...	
sps_transform_skip_enabled_flag	u(1)
...	
if(chroma_format_idc == 3) {	
sps_act_enabled_flag	u(1)
sps_palette_enabled_flag	u(1)
}	
...	
}	

(b)

seq_parameter_set_rbsp() {	Bộ mô tả
...	
chroma_format_idc	u(2)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
...	
bit_depth_minus8	ue(v)
...	
sps_transform_skip_enabled_flag	u(1)
...	
if(chroma_format_idc == 3) {	
sps_act_enabled_flag	u(1)
sps_palette_enabled_flag	u(1)
}	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
...	
}	

Fig. 21

19/23

*Fig. 22*

20/23

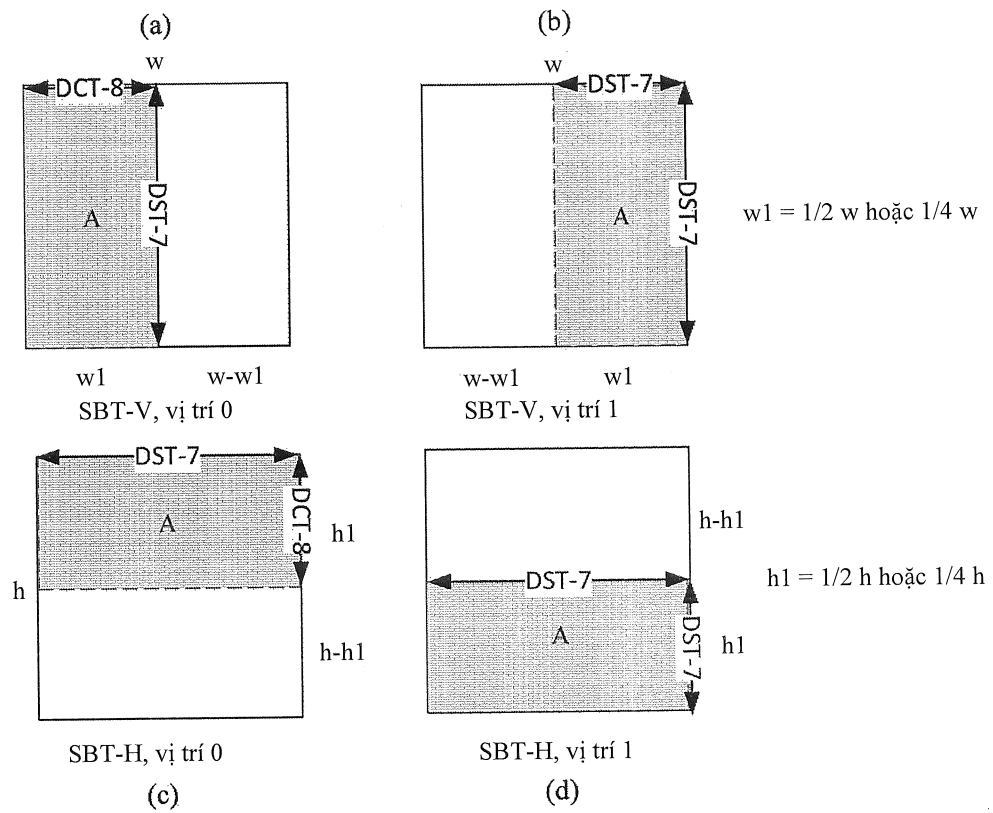


Fig. 23

21/23

	Bộ mô tả
<code>transform_tree(x0, y0, tbWidth, tbHeight, treeType, chType) {</code>	
<code>InferTuCbFluma = 1</code>	
<code>if(IntraSubPartitionsSplitType == ISP_NO_SPLIT && !cu_sbt_flag) {</code>	
<code>if(tbWidth > MaxTbSizeY tbHeight > MaxTbSizeY) {</code>	
<code>verSplitFirst = (tbWidth > MaxTbSizeY && tbWidth > tbHeight) ? 1 : 0</code>	
<code>trafoWidth = verSplitFirst ? (tbWidth / 2) : tbWidth</code>	
<code>trafoHeight = !verSplitFirst ? (tbHeight / 2) : tbHeight</code>	
<code>transform_tree(x0, y0, trafoWidth, trafoHeight, chType)</code>	
<code>if(verSplitFirst)</code>	
<code>transform_tree(x0 + trafoWidth, y0, trafoWidth, trafoHeight, treeType, chType)</code>	
<code>else</code>	
<code>transform_tree(x0, y0 + trafoHeight, trafoWidth, trafoHeight, treeType, chType)</code>	
<code>} else {</code>	
<code>transform_unit(x0, y0, tbWidth, tbHeight, treeType, 0, chType)</code>	
<code>}</code>	
<code>} else if(cu_sbt_flag) {</code>	
<code>if(!cu_sbt_horizontal_flag) {</code>	
<code>trafoWidth = tbWidth * SbtNumFourthsTb0 / 4</code>	
<code>transform_unit(x0, y0, trafoWidth, tbHeight, treeType, 0, 0)</code>	
<code>transform_unit(x0 + trafoWidth, y0, tbWidth - trafoWidth, tbHeight, treeType, 1, 0)</code>	
<code>} else {</code>	
<code>trafoHeight = tbHeight * SbtNumFourthsTb0 / 4</code>	
<code>transform_unit(x0, y0, tbWidth, trafoHeight, treeType, 0, 0)</code>	
<code>transform_unit(x0, y0 + trafoHeight, tbWidth, tbHeight - trafoHeight, treeType, 1, 0)</code>	
<code>}</code>	
<code>} else if(IntraSubPartitionsSplitType == ISP_HOR_SPLIT) {</code>	
<code>trafoHeight = tbHeight / NumIntraSubPartitions</code>	
<code>for(partIdx = 0; partIdx < NumIntraSubPartitions; partIdx++)</code>	
<code>transform_unit(x0, y0 + trafoHeight * partIdx, tbWidth, trafoHeight, treeType, partIdx, 0)</code>	
<code>} else if(IntraSubPartitionsSplitType == ISP_VER_SPLIT) {</code>	
<code>trafoWidth = tbWidth / NumIntraSubPartitions</code>	
<code>for(partIdx = 0; partIdx < NumIntraSubPartitions; partIdx++)</code>	
<code>transform_unit(x0 + trafoWidth * partIdx, y0, trafoWidth, tbHeight, treeType, partIdx, 0)</code>	
<code>}</code>	
<code>}</code>	

Fig. 24

22/23

transform_unit(x0, y0, tbWidth, tbHeight, treeType, subTuIndex, chType) {	Bộ mô tả
if(IntraSubPartitionsSplitType != ISP_NO_SPLIT && treeType == SINGLE_TREE && subTuIndex == NumIntraSubPartitions - 1) {	
xC = CbPosX[chType][x0][y0]	
yC = CbPosY[chType][x0][y0]	
wC = CbWidth[chType][x0][y0] / SubWidthC	
hC = CbHeight[chType][x0][y0] / SubHeightC	
} else {	
xC = x0	
yC = y0	
wC = tbWidth / SubWidthC	
hC = tbHeight / SubHeightC	
}	
chromaAvailable = treeType != DUAL_TREE_LUMA && ChromaArrayType != 0 && (IntraSubPartitionsSplitType == ISP_NO_SPLIT (IntraSubPartitionsSplitType != ISP_NO_SPLIT && subTuIndex == NumIntraSubPartitions - 1))	
if((treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) && ChromaArrayType != 0) {	
if((IntraSubPartitionsSplitType == ISP_NO_SPLIT && !(cu_sbt_flag && ((subTuIndex == 0 && cu_sbt_pos_flag) (subTuIndex == 1 && !cu_sbt_pos_flag)))) (IntraSubPartitionsSplitType != ISP_NO_SPLIT && (subTuIndex == NumIntraSubPartitions - 1))) {	
tu_cbf_cb[xC][yC]	ae(v)
tu_cbf_cr[xC][yC]	ae(v)
}	
}	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if((IntraSubPartitionsSplitType == ISP_NO_SPLIT && !(cu_sbt_flag && ((subTuIndex == 0 && cu_sbt_pos_flag) (subTuIndex == 1 && !cu_sbt_pos_flag))) && (CuPredMode[chType][x0][y0] == MODE_INTRA (chromaAvailable && (tu_cbf_cb[xC][yC] tu_cbf_cr[xC][yC])) CbWidth[chType][x0][y0] > MaxTbSizeY CbHeight[chType][x0][y0] > MaxTbSizeY)) (IntraSubPartitionsSplitType != ISP_NO_SPLIT && (subTuIndex < NumIntraSubPartitions - 1 !InferTuCbfLuma)))	
tu_cbf_luma[x0][y0]	ae(v)
if(IntraSubPartitionsSplitType != ISP_NO_SPLIT)	
InferTuCbfLuma = InferTuCbfLuma && !tu_cbf_luma[x0][y0]	
}	
...	
}	

Fig. 25

23/23

transform_unit(x0, y0, tbWidth, tbHeight, treeType, subTuIndex, chType) {	Bộ mô tả
if(IntraSubPartitionsSplitType != ISP_NO_SPLIT && treeType == SINGLE_TREE && subTuIndex == NumIntraSubPartitions - 1) {	
xC = CbPosX[chType][x0][y0]	
yC = CbPosY[chType][x0][y0]	
wC = CbWidth[chType][x0][y0] / SubWidthC	
hC = CbHeight[chType][x0][y0] / SubHeightC	
} else {	
xC = x0	
yC = y0	
wC = tbWidth / SubWidthC	
hC = tbHeight / SubHeightC	
}	
chromaAvailable = treeType != DUAL_TREE_LUMA && ChromaArrayType != 0 && (IntraSubPartitionsSplitType == ISP_NO_SPLIT (IntraSubPartitionsSplitType != ISP_NO_SPLIT && subTuIndex == NumIntraSubPartitions - 1))	
if((treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) && ChromaArrayType != 0) {	
if((IntraSubPartitionsSplitType == ISP_NO_SPLIT && !(cu_sbt_flag && ((subTuIndex == 0 && cu_sbt_pos_flag) (subTuIndex == 1 && !cu_sbt_pos_flag))) (IntraSubPartitionsSplitType != ISP_NO_SPLIT && (subTuIndex == NumIntraSubPartitions - 1))) {	
tu_cbf_cb[xC][yC]	ae(v)
tu_cbf_cr[xC][yC]	ae(v)
}	
}	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if((IntraSubPartitionsSplitType == ISP_NO_SPLIT && !(cu_sbt_flag && ((subTuIndex == 0 && cu_sbt_pos_flag) (subTuIndex == 1 && !cu_sbt_pos_flag))) && ((CuPredMode[chType][x0][y0] == MODE_INTRA && !cu_act_enabled_flag[x0][y0]) (chromaAvailable && (tu_cbf_cb[xC][yC] tu_cbf_cr[xC][yC])) (CbWidth[chType][x0][y0] > MaxTbSizeY CbHeight[chType][x0][y0] > MaxTbSizeY)) (IntraSubPartitionsSplitType != ISP_NO_SPLIT && (subTuIndex < NumIntraSubPartitions - 1 !InferTuCbfLuma)))	
tu_cbf_luma[x0][y0]	ae(v)
if(IntraSubPartitionsSplitType != ISP_NO_SPLIT)	
InferTuCbfLuma = InferTuCbfLuma && !tu_cbf_luma[x0][y0]	
}	
...	
}	

Fig. 26