



(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)



1-0052166

(51)^{2021.01} H04N 19/70; H04N 19/174; H04N
19/186; H04N 19/96; H04N 19/593;
H04N 19/119; H04N 19/30

(13) B

(21) 1-2022-07006

(22) 22/02/2021

(86) PCT/KR2021/002174 22/02/2021

(87) WO 2021/194100 30/09/2021

(30) 10-2020-0037836 27/03/2020 KR

(45) 27/10/2025 451

(43) 26/12/2022 417A

(73) HUMAX CO., LTD. (KR)

2 Yeongmun-ro Cheoin-gu Yongin-si Gyeonggi-do 17040, Republic of Korea

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

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

(54) VẬT GHI VÀ THIẾT BỊ XỬ LÝ TÍN HIỆU VIDEO BẰNG CÁCH SỬ DỤNG
PHẦN ĐẦU ẢNH

(21) 1-2022-07006

(57) Sáng chế đề cập đến phương pháp xử lý tín hiệu video, phương pháp này bao gồm các bước: thu, từ bộ thông số trình tự (sequence parameter set, SPS) được áp dụng cho trình tự video lớp được mã hóa (coded layer video sequence, CLVS) hiện thời, cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho CLVS hiện thời hay không; nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không; thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không; và nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép, thu, từ phần đầu phần chia, cờ cho phép công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không.

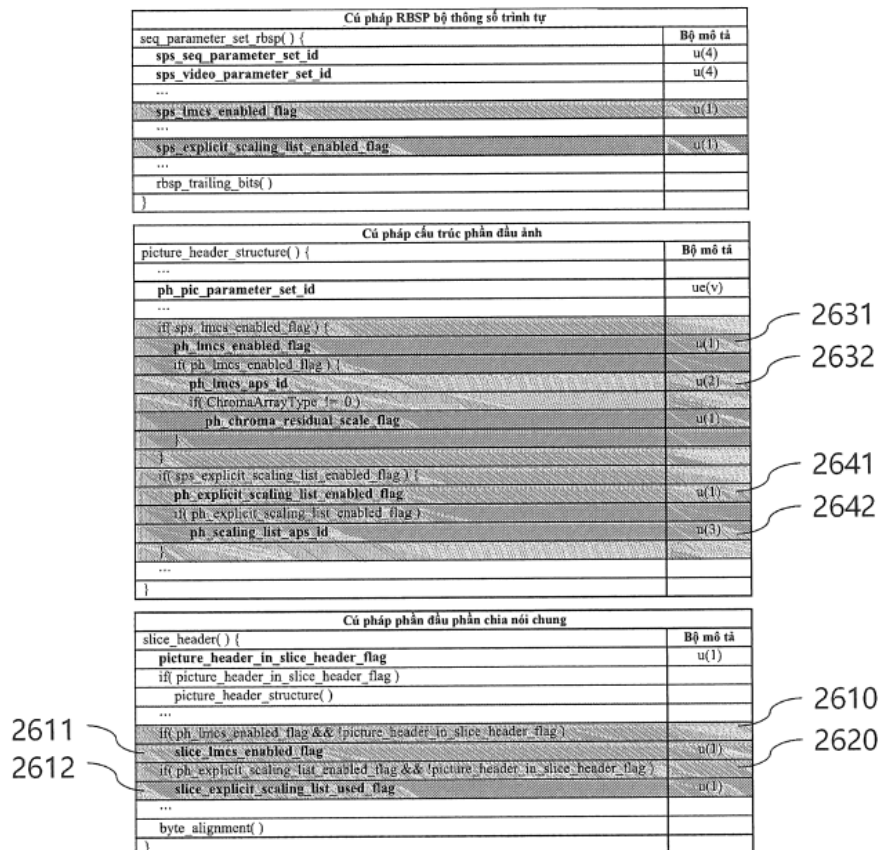


Fig.26

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 xử lý tín hiệu video và thiết bị 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

Sáng chế nhằm cải thiện hiệu quả mã hóa của tín hiệu video.

Giải pháp cho vấn đề

Để đạt được tác vụ ở trên, phương pháp giải mã tín hiệu video theo một phương án của sáng chế bao gồm các bước: thu cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (coded layer video sequence, CLVS) hiện thời hay không từ bộ thông số trình tự (sequence parameter set, SPS) được áp dụng cho CLVS hiện thời; nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao

gồm trong CLVS hiện thời hay không; thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không; nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa, thu, từ phần đầu phần chia, cờ cho phép công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không; và giải mã phần chia hiện thời, dựa trên cờ cho phép công cụ mã hóa cấp phần chia.

Phương pháp giải mã tín hiệu video theo một phương án của sáng chế còn bao gồm, nếu cờ cho phép công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia, bước xác định cờ cho phép công cụ mã hóa cấp phần chia, dựa trên cờ cho phép công cụ mã hóa cấp ảnh.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước xác định cờ cho phép công cụ mã hóa cấp phần chia, dựa trên cờ cho phép công cụ mã hóa cấp ảnh, bao gồm các bước: nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, xác định giá trị của cờ cho phép công cụ mã hóa cấp phần chia là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, xác định cờ cho phép công cụ mã hóa cấp phần chia để biểu thị việc không sử dụng công cụ mã hóa.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước xác định các giá trị giống nhau bao gồm các bước: nếu cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa, xác định cờ cho phép công cụ mã hóa cấp phần chia để biểu thị việc sử dụng công cụ mã hóa; và nếu cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự không cho phép công cụ mã hóa, xác định cờ cho phép công cụ mã hóa cấp phần chia để biểu thị việc không sử dụng công cụ mã hóa.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, các giá trị của các cờ phần đầu ảnh trong phần đầu phần chia của tất cả các phần chia được mã

hóa của trình tự video lớp được mã hóa (CLVS) hiện thời là giống nhau.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, nếu cờ phần đầu ảnh trong phần đầu phân chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phân chia, đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) của phần đầu ảnh bao gồm cấu trúc cú pháp phần đầu ảnh không tồn tại trong trình tự video lớp được mã hóa (CLVS) hiện thời, và bước thu cờ cho phép công cụ mã hóa cấp ảnh bao gồm bước thu cờ cho phép công cụ mã hóa cấp ảnh từ cấu trúc cú pháp phần đầu ảnh được bao gồm trong phần đầu phân chia.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, nếu cờ phần đầu ảnh trong phần đầu phân chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phân chia, đơn vị ảnh (picture unit, PU) hiện thời bao gồm đơn vị NAL phần đầu ảnh, và bước thu cờ cho phép công cụ mã hóa cấp ảnh bao gồm bước thu cờ cho phép công cụ mã hóa cấp ảnh từ cấu trúc cú pháp phần đầu ảnh được bao gồm trong đơn vị NAL phần đầu ảnh.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, đơn vị NAL phần đầu ảnh tồn tại trước đơn vị NAL lớp mã hóa video (video coding layer, VCL) thứ nhất của đơn vị ảnh (PU) hiện thời.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước thu cờ cho phép công cụ mã hóa cấp trình tự bao gồm bước thu cờ cho phép tỷ lệ độ sáng ánh xạ với sắc độ (LMCS) cấp trình tự biểu thị xem LMCS cho CLVS hiện thời có được cho phép hay không, bước thu cờ cho phép công cụ mã hóa cấp ảnh bao gồm các bước: nếu cờ cho phép LMCS cấp trình tự biểu thị sự cho phép LMCS, thu, từ cấu trúc cú pháp phần đầu ảnh, cờ cho phép LMCS cấp ảnh biểu thị xem LMCS có được cho phép cho ảnh hiện thời hay không; và nếu cờ cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, thu bộ định danh cho bộ thông số thích ứng (adaptation parameter set, APS) LMCS bao gồm thông số LMCS từ cấu trúc cú pháp phần đầu ảnh, bước thu cờ cho phép công cụ mã hóa cấp phần chia từ phần đầu phân chia bao gồm, nếu cờ phần đầu ảnh trong phần đầu phân chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phân chia,

và cờ cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, bước thu, từ phần đầu phần chia, cờ cho phép LMCS cấp phần chia biểu thị xem LMCS có được sử dụng cho phần chia hiện thời hay không, và bước giải mã phần chia hiện thời bao gồm bước giải mã phần chia hiện thời, dựa trên cờ cho phép LMCS cấp phần chia và bộ định danh cho APS LMCS.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước thu cờ cho phép công cụ mã hóa cấp ảnh bao gồm, nếu cờ cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, và định dạng màu của ảnh hiện thời là định dạng bao gồm thành phần sắc độ, bước thu, từ cấu trúc cú pháp phần đầu ảnh, cờ tỷ lệ phần dư sắc độ cấp ảnh biểu thị xem sự thay đổi tỷ lệ tín hiệu dư sắc độ có được cho phép cho ảnh hiện thời hay không.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước giải mã phần chia hiện thời bao gồm bước thực hiện quy trình ánh xạ độ sáng nếu cờ cho phép LMCS cấp phần chia biểu thị việc sử dụng LMCS, và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối độ sáng.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước giải mã phần chia hiện thời bao gồm bước thực hiện quy trình thay đổi tỷ lệ phần dư sắc độ nếu cờ cho phép LMCS cấp phần chia biểu thị việc sử dụng LMCS, khối được giải mã hiện thời được bao gồm trong phần chia hiện thời không phải là khối độ sáng, và cờ tỷ lệ phần dư sắc độ cấp ảnh biểu thị sự cho phép sự thay đổi tỷ lệ tín hiệu dư sắc độ cho ảnh hiện thời.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước thu cờ cho phép công cụ mã hóa cấp trình tự bao gồm bước thu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho CLVS hiện thời có được cho phép hay không, bước thu cờ cho phép công cụ mã hóa cấp ảnh bao gồm: nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu, từ cấu trúc cú pháp phần đầu ảnh, cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho ảnh hiện thời có được cho phép hay không; và nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp

ảnh biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ từ cấu trúc cú pháp phần đầu ảnh, bước thu cờ cho phép công cụ mã hóa cấp phần chia từ phần đầu phần chia bao gồm, nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu, từ phần đầu phần chia, cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng cho phần chia hiện thời hay không, và bước giải mã phần chia hiện thời bao gồm bước giải mã phần chia hiện thời, dựa trên cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia và bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ.

Trong phương pháp giải mã tín hiệu video theo một phương án của sáng chế, bước giải mã phần chia hiện thời bao gồm các bước: nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được bao gồm trong APS dựa trên bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ; và nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được tạo cấu hình trước, trong đó tất cả các phần tử của danh sách thay đổi tỷ lệ được tạo cấu hình trước là 16.

Thiết bị giải mã tín hiệu video theo một phương án của sáng chế bao gồm bộ xử lý và bộ nhớ, trong đó bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: thu cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (CLVS) hiện thời hay không từ bộ thông số trình tự (SPS) được áp dụng cho CLVS hiện thời; nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không; thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia

biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không; nếu có phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và có cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa, thu, từ phần đầu phần chia, có cho phép công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không; và giải mã phần chia hiện thời, dựa trên có cho phép công cụ mã hóa cấp phần chia.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, nếu có cho phép công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia, xác định có cho phép công cụ mã hóa cấp phần chia, dựa trên có cho phép công cụ mã hóa cấp ảnh.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: nếu có phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, xác định giá trị của có cho phép công cụ mã hóa cấp phần chia là giống như giá trị của có cho phép công cụ mã hóa cấp ảnh; và nếu có phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, xác định có cho phép công cụ mã hóa cấp phần chia để biểu thị việc không sử dụng công cụ mã hóa.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: nếu có cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa, xác định có cho phép công cụ mã hóa cấp phần chia để biểu thị việc sử dụng công cụ mã hóa; và nếu có cho phép công cụ mã hóa cấp ảnh biểu thị sự không cho phép công cụ mã hóa, xác định có cho phép công cụ mã hóa cấp phần chia để biểu thị việc không sử dụng công cụ mã hóa.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, các giá trị của các có phần đầu ảnh trong phần đầu phần chia của tất cả các phần chia được mã hóa của trình tự video lớp được mã hóa (CLVS) hiện thời là giống nhau.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, nếu có phần

đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, đơn vị lớp trừu tượng mạng (NAL) của phần đầu ảnh bao gồm cấu trúc cú pháp phần đầu ảnh không tồn tại trong trình tự video lớp được mã hóa (CLVS) hiện thời, và bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, thu cò cho phép công cụ mã hóa cấp ảnh từ cấu trúc cú pháp phần đầu ảnh được bao gồm trong phần đầu phần chia.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, nếu cò phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, đơn vị ảnh (PU) hiện thời bao gồm đơn vị NAL phần đầu ảnh, và bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, thu cò cho phép công cụ mã hóa cấp ảnh từ cấu trúc cú pháp phần đầu ảnh được bao gồm trong đơn vị NAL phần đầu ảnh.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, đơn vị NAL phần đầu ảnh tồn tại trước đơn vị NAL lớp mã hóa video (VCL) thứ nhất của đơn vị ảnh (PU) hiện thời.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: thu cò cho phép tỷ lệ độ sáng ánh xạ với sắc độ (LMCS) cấp trình tự biểu thị xem LMCS cho CLVS hiện thời có được cho phép hay không; nếu cò cho phép LMCS cấp trình tự biểu thị sự cho phép LMCS, thu, từ cấu trúc cú pháp phần đầu ảnh, cò cho phép LMCS cấp ảnh biểu thị xem LMCS có được cho phép cho ảnh hiện thời hay không; nếu cò cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, thu bộ định danh cho bộ thông số thích ứng (APS) LMCS bao gồm thông số LMCS từ cấu trúc cú pháp phần đầu ảnh; nếu cò phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và cò cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, thu, từ phần đầu phần chia, cò cho phép LMCS cấp phần chia biểu thị xem LMCS có được sử dụng cho phần chia hiện thời hay không; và giải mã phần chia hiện thời, dựa trên cò cho phép LMCS cấp phần chia và bộ định danh cho APS LMCS.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, nếu có cho phép LMCS cấp ảnh biểu thị sự cho phép LMCS, và định dạng màu của ảnh hiện thời là định dạng bao gồm thành phần sắc độ, thu, từ cấu trúc cú pháp phần đầu ảnh, có tỷ lệ phần dư sắc độ cấp ảnh biểu thị xem sự thay đổi tỷ lệ tín hiệu dư sắc độ có được cho phép cho ảnh hiện thời hay không.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, thực hiện quy trình ánh xạ độ sáng nếu có cho phép LMCS cấp phần chia biểu thị việc sử dụng LMCS, và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối độ sáng.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ, thực hiện quy trình thay đổi tỷ lệ phần dư sắc độ nếu có cho phép LMCS cấp phần chia biểu thị việc sử dụng LMCS, khối được giải mã hiện thời được bao gồm trong phần chia hiện thời không phải là khối độ sáng, và có tỷ lệ phần dư sắc độ cấp ảnh biểu thị sự cho phép sự thay đổi tỷ lệ tín hiệu dư sắc độ cho ảnh hiện thời.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: thu có cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho CLVS hiện thời có được cho phép hay không; nếu có cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu, từ cấu trúc cú pháp phần đầu ảnh, có cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho ảnh hiện thời có được cho phép hay không; nếu có cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ từ cấu trúc cú pháp phần đầu ảnh; nếu có phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và có cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thu, từ

phần đầu phần chia, cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng cho phần chia hiện thời hay không; và giải mã phần chia hiện thời, dựa trên cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia và bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ.

Trong thiết bị giải mã tín hiệu video theo một phương án của sáng chế, bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được bao gồm trong APS dựa trên bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ; và nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được tạo cấu hình trước, trong đó tất cả các phần tử của danh sách thay đổi tỷ lệ được tạo cấu hình trước là 16.

Phương pháp mã hóa tín hiệu video theo một phương án của sáng chế bao gồm các bước: tạo ra cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (CLVS) hiện thời hay không từ bộ thông số trình tự (SPS) được áp dụng cho CLVS hiện thời; nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, tạo ra cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không; tạo ra cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không, và thêm cờ phần đầu ảnh trong phần đầu phần chia này vào phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời; nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa, tạo ra cờ cho phép công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không, và thêm cờ cho phép công cụ mã hóa cấp phần chia này vào phần đầu phần chia; và mã hóa phần chia hiện thời, dựa trên cờ cho phép

công cụ mã hóa cấp phần chia, trong đó cờ cho phép công cụ mã hóa cấp trình tự bao gồm cờ cho phép LMCS cấp trình tự hoặc cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự, cờ cho phép công cụ mã hóa cấp ảnh bao gồm cờ cho phép LMCS cấp ảnh hoặc cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh, và cờ cho phép công cụ mã hóa cấp phần chia bao gồm cờ cho phép LMCS cấp phần chia hoặc cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia.

Thiết bị mã hóa tín hiệu video theo một phương án của sáng chế bao gồm bộ xử lý và bộ nhớ, trong đó bộ xử lý được tạo cấu hình để, dựa trên các lệnh được lưu trữ trong bộ nhớ: tạo ra cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (CLVS) hiện thời hay không từ bộ thông số trình tự (SPS) được áp dụng cho CLVS hiện thời; nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, tạo ra cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không; thêm cờ phần đầu ảnh trong phần đầu phần chia, mà biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không, vào phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời; nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự cho phép công cụ mã hóa thêm, vào phần đầu phần chia, cờ cho phép công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không; và mã hóa phần chia hiện thời, dựa trên cờ cho phép công cụ mã hóa cấp phần chia, trong đó cờ cho phép công cụ mã hóa cấp trình tự bao gồm cờ cho phép LMCS cấp trình tự hoặc cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự, cờ cho phép công cụ mã hóa cấp ảnh bao gồm cờ cho phép LMCS cấp ảnh hoặc cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh, và cờ cho phép công cụ mã hóa cấp phần chia bao gồm cờ cho phép LMCS cấp phần chia hoặc cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia.

Hiệu quả đạt được của sáng chế

Theo một phương án của sáng chế, hiệu quả mã hóa của 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 minh họa phương pháp dự đoán liên ảnh theo một phương án của sáng chế.

Fig.8 là sơ đồ minh họa cụ thể phương pháp biến đổi tín hiệu dư bởi bộ mã hóa.

Fig.9 là sơ đồ minh họa cụ thể phương pháp thu tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi bởi bộ mã hóa và bộ giải mã.

Fig.10 minh họa sơ đồ khối của bộ mã hóa hỗ trợ sơ đồ mã hóa video có thể thay đổi tỷ lệ.

Fig.11 minh họa sơ đồ khối của bộ giải mã hỗ trợ sơ đồ mã hóa video có thể thay đổi tỷ lệ.

Fig.12 minh họa cấu trúc của dòng bit bao gồm nhiều lớp.

Fig.13 là sơ đồ minh họa mối tương quan tham chiếu giữa các ảnh trong dòng bit bao gồm nhiều lớp.

Fig.14 minh họa các phần của cấu trúc cú pháp cấp cao (high-level syntax, HLS) và các mối tương quan tham chiếu giữa chúng.

Fig.15 minh họa cấu trúc cú pháp bộ thông số video (VPS).

Fig.16 minh họa các mã giả suy ra các biến liên quan đến các môi trường tham chiếu giữa các lớp.

Fig.17 minh họa các mã giả suy ra các biến liên quan đến bộ lớp đầu ra.

Fig.18 minh họa các mã giả suy ra số lượng các lớp được bao gồm trong bộ lớp đầu ra và các bộ định danh lớp của các lớp được bao gồm trong bộ lớp đầu ra.

Fig.19 minh họa các mã giả suy ra các biến liên quan đến bộ lớp đầu ra theo phương án khác.

Fig.20 minh họa cấu trúc cú pháp bộ thông số video (VPS) theo phương án khác.

Fig.21 là sơ đồ minh họa việc sử dụng cờ liên quan đến số lượng các lớp con trong lớp, trong các nội dung của cấu trúc cú pháp VPS.

Fig.22 là sơ đồ minh họa các phần tử ngữ nghĩa và cú pháp mà là các phần của các nội dung của cấu trúc cú pháp SPS và biểu thị các lớp con.

Fig.23 là sơ đồ minh họa ví dụ của cấu hình đơn vị ảnh (PU).

Fig.24 là sơ đồ minh họa bộ thông số.

Fig.25 là sơ đồ minh họa bộ thông số.

Fig.26 là sơ đồ minh họa phương pháp báo hiệu xem liệu có sử dụng tỷ lệ độ sáng ánh xạ với sắc độ (LMCS) và danh sách thay đổi tỷ lệ rõ ràng, ở cấp trình tự, ảnh và phần chia hay không.

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. Hoặc sự mã hóa có thể được hiểu là bao gồm cả sự mã hóa và sự giải mã. 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 sự khác biệt 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 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. Trong khi đó, bộ phận lọc 130 thực hiện các hoạt động lọc để cải thiện chất lượng của ảnh được khôi phục và 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 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

khởi 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 trình tự byte thô (Raw Byte Sequence Payload, RBSP) cấp cao hơn chẳng hạn như bộ thông số ảnh (Picture Parameter Set, PPS), bộ thông số trình tự (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 tính tổng 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 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 vectơ 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 vectơ 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 bố 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 vectơ 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 vectơ khối thu được. Thông tin mã hóa BC nội ảnh có thể bao gồm thông tin vectơ 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 trình tự đơn vị cây mã hóa (CTU). Đơn vị cây mã hóa bao gồm khối NXN 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 $N \times N$. 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. Khi sự phân chia đối với đơn vị tạo mã không được biểu thị hoặc đơn vị tạo mã không lớn hơn so với chiều dài chuyển đổi lớn nhất, thì đơn vị tạo mã có thể được sử dụng làm đơn vị để dự đoán và/hoặc chuyển đổi mà không được phân chia hơn nữa. Theo một phương án, khi chiều rộng hoặc chiều cao của đơn vị mã hóa hiện thời lớn hơn độ dài chuyển đổi tối đa, thì đơn vị mã hóa hiện thời có thể được phân chia thành các đơn vị chuyển đổi mà không báo hiệu rõ ràng về sự phân chia. 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 cấp 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, còn 'split_qt_flag' biểu thị xem nút cây tứ phân có được phân chia hay không, còn 'mtt_split_cu_vertical_flag' biểu thị hướng phân chia của nút cây đa loại hoặc còn '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, 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, “qt_split_flag” được báo hiệu cho mỗi nút “Qt_node”. Nếu giá trị của “qt_split_flag” là 1, thì nút tương ứng được phân chia thành 4 nút vuông, và nếu giá trị của “qt_split_flag” là 0, thì nút là nút lá “QT_leaf_node” của cây tứ phân.

Mỗi nút lá của cây tứ phân 'QT_leaf_node' có thể còn được phân chia thành cấu trúc cây đa loại. Trong cấu trúc cây đa loại, 'mtt_split_flag' được báo hiệu cho mỗi nút 'MTT_node'. Nếu giá trị của “mtt_split_flag” là 1, thì nút tương ứng được phân chia thành nhiều nút hình chữ nhật, và nếu giá trị của “mtt_split_flag” là 0, thì nút là nút lá “MTT_leaf_node” của cây đa loại. Nếu nút của cây đa loại “MTT_node” được phân chia thành nhiều nút hình chữ nhật (nghĩa là, nếu giá trị của “mtt_split_flag” là 1), “mtt_split_vertical_flag” và “mtt_split_binary_flag” cho nút “MTT_node” có thể được báo hiệu bổ sung. Nếu giá trị của “mtt_split_vertical_flag” là 1, sự phân chia theo chiều dọc của nút “MTT_node” được biểu thị, và nếu giá trị của “mtt_split_vertical_flag” là 0, sự phân chia theo chiều ngang của nút “MTT_node” được biểu thị. Ngoài ra, nếu giá trị của “mtt_split_binary_flag” là 1, thì nút “MTT_node” được phân chia thành 2 nút hình chữ nhật, và nếu giá trị của “mtt_split_binary_flag” là 0, thì nút “MTT_node” được phân chia thành 3 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). Sau đây, đơn vị cơ bản để thực hiện phép dự đoán này được gọi là đơn vị dự đoán hoặc khối dự đoán.

Sau đây, thuật ngữ “đơn vị” được sử dụng trong sáng chế có thể được sử dụng làm thuật ngữ để thay thế 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 tập 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, tập 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, tập 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).

Trong khi đó, phạm vi góc được thiết lập trước có thể được thiết lập khác đi theo hình dạng của khối hiện thời. Ví dụ, khi khối hiện thời là khối hình chữ nhật, chế độ góc rộng biểu thị góc mà lớn hơn 45 độ hoặc nhỏ hơn -135 độ theo hướng chiều kim đồng hồ có thể còn được sử dụng. Khi khối hiện thời là khối ngang, thì chế độ góc có thể biểu thị góc nằm trong phạm vi góc (nghĩa là, phạm vi góc thứ hai) giữa $(45 + \text{độ lệch}1)$ độ và $(-135 + \text{độ lệch}1)$ độ theo hướng chiều kim đồng hồ. Ở đây, các chế độ góc từ 67 đến 76 nằm ngoài phạm vi góc thứ nhất có thể còn được sử dụng. Hơn nữa, khi khối hiện thời là khối

dọc, thì chế độ góc có thể biểu thị góc nằm trong phạm vi góc (nghĩa là, phạm vi góc thứ ba) giữa $(45 - \text{độ lệch}_2)$ độ và $(-135 - \text{độ lệch}_2)$ độ theo hướng chiều kim đồng hồ. Ở đây, các chế độ góc từ -10 đến -1 nằm ngoài phạm vi góc thứ nhất có thể còn được sử dụng. Theo một phương án của sáng chế, các giá trị của độ lệch₁ và độ lệch₂ có thể được xác định khác đi theo tỷ lệ giữa chiều rộng và chiều cao của khối hình chữ nhật. Hơn nữa, độ lệch₁ và độ lệch₂ có thể là các số dương.

Theo một phương án khác của sáng chế, các chế độ góc cấu thành tập chế độ dự đoán nội ảnh có thể bao gồm chế độ góc cơ sở và chế độ góc mở rộng. Ở đây, chế độ góc mở rộng có thể được xác định dựa trên chế độ góc cơ sở.

Theo một phương án, chế độ góc cơ sở có thể là chế độ tương ứng với góc được sử dụng trong phép dự đoán nội ảnh của tiêu chuẩn mã hóa video hiệu quả cao (High Efficiency Video Coding, HEVC) hiện có, và chế độ góc mở rộng có thể là chế độ tương ứng với góc mới được thêm vào trong phép dự đoán nội ảnh của tiêu chuẩn bộ mã hóa-giải mã video thế hệ tiếp theo. Một cách chi tiết hơn, chế độ góc cơ sở có thể là chế độ góc tương ứng với chế độ bất kỳ trong số các chế độ dự đoán nội ảnh $\{2, 4, 6, \dots, 66\}$, và chế độ góc mở rộng có thể là chế độ góc tương ứng với chế độ bất kỳ trong số các chế độ dự đoán nội ảnh $\{3, 5, 7, \dots, 65\}$. Cụ thể là, chế độ góc mở rộng có thể là chế độ góc giữa các chế độ góc cơ sở nằm trong phạm vi góc thứ nhất. Do đó, góc được biểu thị bởi chế độ góc mở rộng có thể được xác định dựa trên góc được biểu thị bởi chế độ góc cơ sở.

Theo phương án khác, chế độ góc cơ sở có thể là chế độ tương ứng với góc nằm trong phạm vi góc thứ nhất được thiết lập trước, và chế độ góc mở rộng có thể là chế độ góc rộng nằm ngoài phạm vi góc thứ nhất. Cụ thể là, chế độ góc cơ sở có thể là chế độ góc tương ứng với chế độ bất kỳ trong số các chế độ dự đoán nội ảnh $\{2, 3, 4, \dots, 66\}$, và chế độ góc mở rộng có thể là chế độ góc tương ứng với chế độ bất kỳ trong số các chế độ dự đoán nội ảnh $\{-10, -9, \dots, -1\}$ và $\{67, 68, \dots, 76\}$. Góc được biểu thị bởi chế độ góc mở rộng có thể được xác định là góc đối diện với góc được biểu thị bởi chế độ góc cơ sở tương ứng. Do đó, góc được biểu thị bởi chế độ góc mở rộng có thể được xác định dựa trên góc được biểu thị bởi chế độ góc cơ sở. Trong khi đó, số lượng các chế độ góc mở rộng không

bị giới hạn ở đó, và các góc mở rộng khác có thể được xác định theo kích thước và/hoặc hình dạng của khối hiện thời. Ví dụ, chế độ góc được mở rộng có thể được xác định là chế độ góc tương ứng với chế độ bất kỳ trong số các chế độ dự đoán nội ảnh $\{-14, -13, \dots, -1\}$ và $\{67, 68, \dots, 80\}$. Trong khi đó, tổng số lượng của các chế độ dự đoán nội ảnh được bao gồm trong tập chế độ dự đoán nội ảnh có thể được thay đổi theo các cấu hình nêu trên của chế độ góc cơ sở và chế độ góc mở rộng.

Theo các phương án trên, khoảng giữa các chế độ góc mở rộng có thể được thiết lập dựa trên khoảng giữa các chế độ góc cơ sở tương ứng. Ví dụ, khoảng giữa các chế độ góc mở rộng $\{3, 5, 7, \dots, 65\}$ có thể được xác định dựa trên khoảng giữa các chế độ góc cơ sở tương ứng $\{2, 4, 6, \dots, 66\}$. Hơn nữa, khoảng giữa các chế độ góc mở rộng $\{-10, -9, \dots, -1\}$ có thể được xác định dựa trên khoảng giữa các chế độ góc cơ sở đối diện tương ứng $\{56, 57, \dots, 65\}$, và khoảng giữa các chế độ góc mở rộng $\{67, 68, \dots, 76\}$ có thể được xác định dựa trên khoảng giữa các chế độ góc cơ sở đối diện tương ứng $\{3, 4, \dots, 12\}$. Khoảng giữa các chế độ góc mở rộng có thể được thiết lập bằng khoảng góc giữa các chế độ góc cơ sở tương ứng. Hơn nữa, số lượng các chế độ góc mở rộng trong tập chế độ dự đoán nội ảnh có thể được thiết lập bằng hoặc nhỏ hơn số lượng các chế độ góc cơ sở.

Theo một phương án của sáng chế, chế độ góc được mở rộng có thể được báo hiệu trên cơ sở chế độ góc cơ sở. Ví dụ, chế độ góc rộng (nghĩa là, chế độ góc mở rộng) có thể thay thế ít nhất một chế độ góc (nghĩa là, chế độ góc cơ sở) nằm trong phạm vi góc thứ nhất. Chế độ góc cơ sở được thay thế có thể là chế độ góc tương ứng với phía đối diện của chế độ góc rộng. Cụ thể là, chế độ góc cơ sở được thay thế là chế độ góc tương ứng với góc đối diện với góc được biểu thị bởi chế độ góc rộng hoặc tương ứng với góc mà khác nhiều nhất bằng chỉ số độ lệch được thiết lập trước từ góc đối diện. Theo một phương án của sáng chế, chỉ số độ lệch được thiết lập trước là 1. Chỉ số chế độ dự đoán nội ảnh tương ứng với chế độ góc cơ sở được thay thế có thể được ánh xạ lại đến chế độ góc rộng để báo hiệu chế độ góc rộng. Ví dụ, các chế độ góc rộng $\{-10, -9, \dots, -1\}$ có thể lần lượt được báo hiệu bởi các chỉ số chế độ dự đoán nội ảnh $\{57, 58, \dots, 66\}$ và các chế độ góc rộng $\{67, 68, \dots, 76\}$ có thể lần lượt được báo hiệu bởi các chỉ số chế độ dự đoán nội ảnh $\{2, 3, \dots, 11\}$. Như được mô tả ở trên, vì chỉ số chế độ dự đoán nội ảnh cho chế độ góc cơ sở báo

hiệu chế độ góc mở rộng, các chỉ số chế độ dự đoán nội ảnh của cùng tập hợp có thể được sử dụng để báo hiệu chế độ dự đoán nội ảnh thậm chí nếu các chế độ góc được sử dụng trong phép dự đoán nội ảnh của mỗi khối có các cấu hình khác nhau. Do đó, tổng phí báo hiệu do sự thay đổi về cấu hình chế độ dự đoán nội ảnh có thể được giảm thiểu.

Trong khi đó, việc có sử dụng chế độ góc mở rộng hay không có thể được xác định dựa trên ít nhất một trong số hình dạng và kích thước của khối hiện thời. Theo một phương án, chế độ góc mở rộng có thể được sử dụng cho phép dự đoán nội ảnh của khối hiện thời khi kích thước của khối hiện thời lớn hơn kích thước được thiết lập trước; nếu không thì, chỉ chế độ góc cơ sở có thể được sử dụng cho phép dự đoán nội ảnh của khối hiện thời. Theo phương án khác, chế độ góc mở rộng có thể được sử dụng cho phép dự đoán nội ảnh của khối hiện thời khi khối hiện thời là khối không vuông, và chỉ chế độ góc cơ sở có thể được sử dụng cho phép dự đoán nội ảnh của khối hiện thời khi khối hiện thời là khối vuông.

Sau đây, phương pháp dự đoán liên ảnh theo một phương án của sáng chế sẽ được mô tả dựa vào Fig.7. Theo sáng chế, phương pháp dự đoán liên ảnh có thể bao gồm phương pháp dự đoán liên ảnh chung được tối ưu hóa cho chuyển động tịnh tiến và phương pháp dự đoán liên ảnh dựa trên mô hình afin. Vector chuyển động có thể bao gồm ít nhất một trong số vector chuyển động chung để bù chuyển động theo phương pháp dự đoán liên ảnh chung và vector chuyển động điểm điều khiển để bù chuyển động afin.

Fig.7 minh họa phương pháp dự đoán liên ảnh theo một phương án của sáng chế. Như được mô tả ở trên, bộ giải mã có thể dự đoán khối hiện thời bằng cách tham chiếu đến các mẫu được khôi phục của ảnh được giải mã khác. Dựa vào Fig.7, bộ giải mã thu khối tham chiếu 702 trong ảnh tham chiếu 720, dựa trên tập thông tin chuyển động của khối hiện thời 701. Tập thông tin chuyển động có thể bao gồm chỉ số ảnh tham chiếu và vector chuyển động. Chỉ số ảnh tham chiếu biểu thị ảnh tham chiếu 720 bao gồm khối tham chiếu dùng cho phép dự đoán liên ảnh của khối hiện thời trong danh sách ảnh tham chiếu. Theo một phương án, danh sách ảnh tham chiếu có thể bao gồm ít nhất một trong số danh sách ảnh L0 và danh sách ảnh L1 nêu trên. Vector chuyển động biểu diễn độ lệch giữa giá trị tọa độ của khối hiện thời 701 trong ảnh hiện thời 710 và giá trị tọa độ của khối tham chiếu 702

trong ảnh tham chiếu 720. Bộ giải mã thu nhận bộ dự đoán của khối hiện thời 701 trên cơ sở các giá trị mẫu của khối tham chiếu 702 và khôi phục khối hiện thời 701 bằng cách sử dụng bộ dự đoán.

Cụ thể là, bộ mã hóa có thể thu khối tham chiếu nêu trên bằng cách tìm kiếm khối tương tự với khối hiện thời từ các ảnh có thứ tự khôi phục sớm hơn. Ví dụ, bộ mã hóa có thể tìm kiếm khối tham chiếu trong đó tổng của các chênh lệch giữa khối hiện thời và các giá trị mẫu là nhỏ nhất trong khu vực tìm kiếm được tạo cấu hình trước. Để đo độ tương tự giữa các mẫu của khối hiện thời và khối tham chiếu, ít nhất một trong số tổng của chênh lệch tuyệt đối (sum of absolute difference, SAD) hoặc tổng của chênh lệch được biến đổi Hadamard (sum of Hadamard transformed difference, SATD) có thể được sử dụng. SAD có thể là giá trị được thu bằng cách tính tổng tất cả các giá trị tuyệt đối của các chênh lệch tương đối giữa các giá trị mẫu được bao gồm trong hai khối. SATD có thể là giá trị được thu bằng cách cộng tất cả các giá trị tuyệt đối của các hệ số biến đổi Hadamard được thu bằng cách thực hiện phép biến đổi Hadamard của chênh lệch giữa các giá trị mẫu được bao gồm trong hai khối.

Khối hiện thời có thể được dự đoán nhờ sử dụng một hoặc nhiều khu vực tham chiếu. Như được mô tả ở trên, khối hiện thời có thể được dự đoán liên ảnh qua sơ đồ dự đoán theo hai hướng nhờ sử dụng hai hoặc nhiều hơn hai khu vực tham chiếu. Theo một phương án, bộ giải mã có thể thu hai khối tham chiếu, dựa trên hai tập thông tin chuyển động của khối hiện thời. Bộ giải mã có thể thu bộ dự đoán thứ nhất và bộ dự đoán thứ hai của khối hiện thời, dựa trên các giá trị mẫu của hai khối tham chiếu được thu tương ứng. Bộ giải mã có thể khôi phục khối hiện thời bằng cách sử dụng bộ dự đoán thứ nhất và bộ dự đoán thứ hai. Ví dụ, bộ giải mã có thể khôi phục khối hiện thời, dựa trên trung bình đối với mỗi mẫu đối với bộ dự đoán thứ nhất và bộ dự đoán thứ hai.

Như được mô tả ở trên, để bù chuyển động của khối hiện thời, một hoặc nhiều tập thông tin chuyển động có thể được báo hiệu. Độ tương đồng giữa các tập thông tin chuyển động để bù chuyển động cho mỗi trong số các khối có thể được sử dụng. Ví dụ, tập thông tin chuyển động được sử dụng để dự đoán khối hiện thời có thể được suy ra từ tập thông

tin chuyển động được sử dụng để dự đoán một trong số các mẫu khác được khôi phục trước đó. Theo đó, bộ mã hóa và bộ giải mã có thể làm giảm tổng phí báo hiệu.

Ví dụ, có thể có nhiều khối ứng viên mà có khả năng đã được dự đoán dựa trên tập thông tin chuyển động giống hoặc tương tự với tập thông tin chuyển động của khối hiện thời. Bộ giải mã có thể tạo ra danh sách ứng viên hợp nhất trên cơ sở của các khối ứng viên. Ở đây, danh sách ứng viên hợp nhất có thể bao gồm các ứng viên tương ứng với các mẫu mà có khả năng đã được dự đoán dựa trên tập thông tin chuyển động liên quan đến tập thông tin chuyển động của khối hiện thời, từ trong số các mẫu được khôi phục trước khối hiện thời. Bộ mã hóa và bộ giải mã có thể tạo cấu hình các danh sách ứng viên hợp nhất của khối hiện thời theo quy tắc được xác định trước. Các danh sách ứng viên hợp nhất lần lượt được tạo cấu hình bởi bộ mã hóa và bộ giải mã có thể giống nhau. Ví dụ, bộ mã hóa và bộ giải mã có thể tạo cấu hình các danh sách ứng viên hợp nhất của khối hiện thời, dựa trên vị trí của khối hiện thời nằm trong ảnh hiện thời. Theo sáng chế, vị trí của khối cụ thể biểu thị vị trí tương đối của mẫu phía trên bên trái của khối cụ thể trong ảnh bao gồm khối cụ thể. Danh sách ứng viên hợp nhất có thể bao gồm ứng viên không gian và ứng viên thời gian. Phương pháp dự đoán để thực hiện phép bù chuyển động nhờ sử dụng tập thông tin chuyển động được thu dựa trên danh sách ứng viên hợp nhất có thể được gọi là chế độ hợp nhất. Trong trường hợp cờ (cờ hợp nhất) biểu thị xem chế độ hợp nhất có được áp dụng cho khối hiện thời hay không, và khi chế độ hợp nhất được áp dụng cho khối hiện thời, bộ mã hóa có thể báo hiệu chỉ số (chỉ số ứng viên hợp nhất) biểu thị tập thông tin chuyển động được sử dụng trong danh sách ứng viên hợp nhất. Bộ giải mã có thể xác định, dựa trên cờ hợp nhất, xem chế độ hợp nhất có được áp dụng cho khối hiện thời hay không, và có thể thu tập thông tin chuyển động, dựa trên chỉ số ứng viên hợp nhất.

Phương pháp khác để báo hiệu tập thông tin chuyển động là, vector chuyển động (MV) được sử dụng cho phép bù chuyển động có thể được phân chia thành thông tin bộ dự đoán vector chuyển động (MVP) và thông tin chênh lệch vector chuyển động (MVD) để được báo hiệu, trong trường hợp đó là MV có thể được biểu diễn dưới dạng tổng của MVP và MVD. Vì vector chuyển động của khối hiện thời có sự tương quan với vector chuyển động của vị trí liền kề theo không gian hoặc thời gian với khối hiện thời, MVP được suy

ra dựa trên đó, và bằng cách báo hiệu MVD tương ứng với giá trị chênh lệch giữa thông tin MVP và MVP của vector chuyển động thực tế, tổng phí của báo hiệu tập thông tin chuyển động có thể được làm giảm. Trong bộ giải mã, MVP có thể được thu dựa trên danh sách ứng viên bộ dự đoán vector chuyển động (MVP) bao gồm nhiều ứng viên bộ dự đoán vector chuyển động và thông tin chỉ số biểu thị ứng viên được sử dụng trong danh sách ứng viên. Bộ mã hóa và bộ giải mã có thể tạo ra cùng danh sách ứng viên bộ dự đoán vector chuyển động, dựa trên nhiều vector chuyển động tương ứng với các vị trí liên kề theo không gian hoặc thời gian.

Để cải thiện hiệu quả mã hóa, phương pháp, thay vì mã hóa phần dư nêu trên, lượng tử hóa giá trị hệ số biến đổi được thu bằng cách biến đổi tín hiệu dư và mã hóa hệ số biến đổi được lượng tử 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 khu vực của khối hiện thời. Theo đó, hiệu quả mã hóa có thể được cải thiện bằng cách tập trung năng lượng trong miền tần số thấp qua phép biến đổi miền tần số của tín hiệu dư. Sau đây, phương pháp biến đổi hoặc biến đổi ngược tín hiệu dư sẽ được mô tả một cách chi tiết.

Fig.8 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 vùng 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 ít nhất một khối dư bao gồm tín hiệu dư cho khối hiện thời. Khối dư có thể hoặc là khối hiện thời hoặc là một trong số các khối được phân chia từ khối hiện thời. Theo sáng chế, khối dư có thể được gọi là mảng dư hoặc ma trận dư, mà bao gồm các mẫu dư của khối hiện thời. Theo sáng chế, khối dư có thể thể hiện bộ phận biến đổi hoặc khối có kích thước giống như kích thước của khối biến đổi.

Tiếp theo, bộ mã hóa có thể biến đổi khối dư bằng cách 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 tách riêng được của phép biến đổi theo chiều dọc và phép biến đổi theo chiều ngang. Trong trường hợp này, phép biến đổi dùng cho khối dư có thể được thực hiện một cách tách riêng

thành phép biến đổi theo chiều dọc và phép biến đổi theo chiều ngang. Ví dụ, bộ mã hóa có thể thực hiện phép biến đổi theo chiều dọc bằng cách áp dụng nhân biến đổi theo hướng dọc của khối dư. Bộ mã hóa có thể thực hiện phép biến đổi theo chiều 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, hàm biến đổi, và phép biến đổi. Theo một phương án, nhân biến đổi có thể là nhân bất kỳ trong số các nhân khả dụng. Nhân biến đổi dựa trên các loại phép biến đổi khác nhau có thể được sử dụng cho mỗi trong số phép biến đổi theo chiều dọc và phép biến đổi theo chiều ngang.

Bộ mã hóa có thể chuyển khối biến đổi được biến đổi từ khối dư đến bộ phận lượng tử hóa và lượng tử hóa khối biến đổi. 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 bố trí theo hai chiều. Như trong trường hợp khối dư, kích thước của khối biến đổi có thể giống với kích thước của hoặc khối hiện thời hoặc khối được phân chia từ khối hiện thời. 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.

Bộ mã hóa có thể thực hiện phép biến đổi bổ sung trước khi các hệ số biến đổi được lượng tử hóa. Như được minh họa trên Fig.8, phương pháp biến đổi được mô tả ở trên có thể được gọi là phép biến đổi sơ cấp, và phép biến đổi bổ sung có thể được gọi là phép biến đổi thứ cấp. Phép biến đổi thứ cấp có thể là tùy chọn đối với mỗi khối dư. Theo một phương án, bộ mã hóa có thể cải thiện hiệu quả mã hóa bằng cách thực hiện phép biến đổi thứ cấp đối với khu vực trong đó khó tập trung năng lượng ở miền tần số thấp chỉ bằng cách 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 lớn hơn 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 khả năng cao được thay đổi 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 bổ sung phép biến đổi thứ cấp đối với tín hiệu dư của khối được dự đoán nội ảnh. 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 xem có thực hiện phép biến đổi thứ cấp hay không có thể được xác định tùy thuộc vào kích thước của khối hiện thời hoặc kích thước của khối dư. Các nhân biến đổi có các kích thước khác nhau có thể được sử dụng tùy thuộc vào kích thước của khối hiện thời hoặc kích thước của khối dư. Ví dụ, phép biến đổi thứ cấp 8×8 có thể được áp dụng cho khối trong đó độ dài của mặt ngắn hơn trong số chiều rộng hoặc chiều cao lớn hơn hoặc bằng độ dài được tạo cấu hình trước thứ nhất. Hơn nữa, phép biến đổi thứ cấp 4×4 có thể được áp dụng cho khối trong đó độ dài của mặt ngắn hơn trong số chiều rộng hoặc chiều cao lớn hơn hoặc bằng độ dài được tạo cấu hình trước thứ hai và nhỏ hơn độ dài được tạo cấu hình trước thứ nhất. Trong trường hợp này, độ dài được tạo cấu hình trước thứ nhất có thể là giá trị lớn hơn độ dài được tạo cấu hình trước thứ hai, nhưng sáng chế không bị giới hạn ở đó. Không giống như phép biến đổi sơ cấp, phép biến đổi thứ cấp có thể không được thực hiện một cách tách riêng thành phép biến đổi theo chiều dọc và phép biến đổi theo chiều 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).

Trong trường hợp tín hiệu video trong khu vực cụ thể, năng lượng ở dải tần số cao có thể không được làm giảm thậm chí nếu phép biến đổi tần số được thực hiện do sự thay đổi đột ngột về độ sáng. Theo đó, hiệu suất nén do lượng tử hóa có thể bị suy giảm. Khi phép biến đổi được thực hiện đối với khu vực trong đó giá trị dư hiếm khi 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 của khu vực cụ thể có thể được lược bỏ. Việc liệu có thực hiện phép biến đổi đối với tín hiệu dư của khu vực cụ thể hay không có thể được xác định bởi phần tử cú pháp liên quan đến phép biến đổi của khu vực 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. Nếu thông tin bỏ qua phép biến đổi đối với 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 phép biến đổi của khu vực tương ứng không được thực hiện. Các hoạt động của bộ mã hóa được mô tả dựa vào Fig.8 có thể được thực hiện bằng bộ phận biến đổi của Fig.1.

Các phần tử cú pháp liên quan đến phép biến đổi nêu trên 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. 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.9 là sơ đồ minh họa cụ thể phương pháp thu tín hiệu dư bằng cách biến đổi ngược hệ số biến đổi bởi bộ mã hóa và bộ giải mã. Nhằm thuận tiện cho phần mô tả, hoạt động biến đổi ngược được thực hiện bằng 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 khu vực cụ thể có được thực hiện hay không, từ phần tử cú pháp liên quan đến phép biến đổi của khu vực. Theo một phương án, nếu phần tử cú pháp liên quan đến phép biến đổi đối với khối biến đổi cụ thể biểu thị sự bỏ qua biến đổi, phép biến đổi đối với khối biến đổi có thể được lược bỏ. Trong trường hợp này, cả phép biến đổi ngược sơ cấp và phép biến đổi ngược thứ cấp được mô tả ở trên liên quan đến khối biến đổi có thể được lược bỏ. 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 ngược sơ cấp nêu trên biểu diễn phép biến đổi ngược đối với phép biến đổi sơ cấp, và có thể được gọi là phép biến đổi sơ cấp ngược. Phép biến đổi ngược thứ cấp biểu diễn phép biến đổi ngược đối với phép biến đổi thứ cấp (LFNST), và có thể được gọi là phép biến đổi ngược thứ cấp hoặc LFNST ngược. Theo sáng chế, phép biến đổi (ngược) sơ cấp có thể được gọi là phép biến đổi (ngược) sơ cấp, và phép biến đổi (ngược) thứ cấp có thể được gọi là phép biến đổi (ngược) thứ 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 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 ngược sơ cấp đối với phép biến đổi thứ cấp hay không. Ví dụ, khi khối biến đổi là khối biến đổi của khối được dự đoán nội ảnh, thì phép biến đổi ngược thứ cấp có thể được thực hiện đối với khối biến đổi. 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 một ví dụ khác, việc xem có thực hiện phép biến đổi ngược thứ cấp hay không có thể được xác định dựa trên kích thước của khối biến đổi. Phép biến đổi ngược thứ cấp có thể được thực hiện sau khi lượng tử hóa ngược và trước khi biến đổi ngược sơ cấp.

Bộ phận biến đổi ngược có thể thực hiện phép biến đổi ngược sơ cấp đối với hệ số biến đổi được lượng tử hóa ngược hoặc hệ số biến đổi được biến đổi ngược thứ cấp. Trong trường hợp phép biến đổi ngược sơ cấp, phép biến đổi theo chiều dọc và phép biến đổi theo chiều ngang có thể được thực hiện một cách tách riêng như trong trường hợp biến đổi sơ cấp. Ví dụ, bộ phận biến đổi ngược có thể thu khối dư bằng cách thực hiện phép biến đổi ngược theo chiều dọc và phép biến đổi ngược theo chiều ngang đối với khối biến đổi. Bộ phận biến đổi ngược có thể biến đổi ngược khối biến đổi, dựa trên nhân biến đổi được sử dụng để biến đổi khối biến đổi. Ví dụ, bộ mã hóa có thể báo hiệu 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 để biến đổi ngược 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 qua phép biến đổi ngược đối với hệ số biến đổi.

Nói chung các thiết bị đầu cuối di động, chẳng hạn như các điện thoại thông minh và các PC dạng bảng, đã thay đổi mạnh mẽ các mô hình xem các nội dung video. Thay vì ngồi trước TV và xem buổi phát sóng của chương trình vào thời gian cố định, xu hướng xem các nội dung video mong muốn mọi lúc và mọi nơi thông qua dịch vụ video theo yêu cầu bằng cách sử dụng điện thoại thông minh đang tăng lên. Theo đó, có nhu cầu đối với sơ đồ mã hóa video mà qua đó các nội dung video có thể được cung cấp mượt cho các thiết bị đầu cuối khác nhau và trong các môi trường mạng khác nhau. Sơ đồ mã hóa video có thể thay đổi tỷ lệ là công nghệ để giải quyết các thiết bị đầu cuối và các môi trường mạng đa dạng, và bộ mã hóa tạo ra dòng bit bằng cách mã hóa hình ảnh theo cách phân cấp để hỗ trợ các độ phân giải, các tốc độ khung hình (số khung hình trên mỗi giây), các chất lượng hình ảnh khác nhau, v.v. đối với một hình ảnh. Bộ giải mã có thể trích xuất tất cả hoặc một phần của dòng bit và có thể giải mã dòng bit để thu hình ảnh thích hợp cho thiết

bị đầu cuối và môi trường tương ứng.

Fig.10 minh họa sơ đồ khối của bộ mã hóa hỗ trợ sơ đồ mã hóa video có thể thay đổi tỷ lệ. Fig.10 minh họa ví dụ của việc tạo ra dòng bit hỗ trợ tất cả các hình ảnh có độ phân giải 8K (7680 x 4320) độ nét cực cao (ultra-high definition, UHD), độ phân giải 4K (3840 x 2160) UHD và độ phân giải (1920 x 1080) độ nét cao (FHD). Đối với hình ảnh đầu vào có độ phân giải 8K, sau khi xử lý sơ bộ, hình ảnh đầu vào có độ phân giải 4K và độ phân giải FHD có thể được thu, và việc xử lý sơ bộ có thể bao gồm việc lấy mẫu xuống. Các trình tự hình ảnh đầu vào có các độ phân giải khác nhau có thể được mã hóa trong các lớp khác nhau. Hình ảnh đầu vào FHD có độ phân giải thấp nhất có thể được mã hóa dưới dạng lớp 0 (lớp thấp nhất), và các hình ảnh đầu vào có độ phân giải 4K và độ phân giải 8K có thể lần lượt được mã hóa dưới dạng lớp 1 và lớp 2. Để làm giảm phần dư tồn tại giữa các lớp, phép dự đoán liên lớp (inter-layer prediction, ILP) có thể được thực hiện. Vì có sự tương quan giữa lớp thấp hơn và lớp cao hơn, khi ảnh hiện thời của lớp cao hơn được mã hóa, mẫu hoặc phần tử cú pháp của ảnh đã được mã hóa (được khôi phục) trong lớp thấp hơn có thể được sử dụng để mã hóa ảnh hiện thời của lớp cao hơn. Ví dụ, ảnh đã được khôi phục của lớp thấp hơn, mà tương ứng với thứ tự đầu ra giống như thứ tự đầu ra của ảnh hiện thời, có thể được sử dụng làm ảnh tham chiếu của ảnh hiện thời. Điều này biểu thị rằng ảnh được khôi phục có thể được sử dụng làm khối tham chiếu để dự đoán bố cục (giá trị mẫu điểm ảnh) ảnh hiện thời hoặc có thể được sử dụng để mã hóa tập thông tin chuyển động. Ảnh được sử dụng làm ảnh tham chiếu bằng phép dự đoán liên lớp có thể được gọi là ảnh tham chiếu liên lớp (inter-layer reference picture, ILRP) và có thể được lưu trữ trong bộ đệm ảnh được giải mã (DPB) của lớp hiện thời sau khi xử lý liên lớp. Xử lý liên lớp là thủ tục xử lý để sử dụng ảnh đã được khôi phục của lớp thấp hơn để mã hóa ảnh hiện thời, và, ví dụ, nếu khả năng thay đổi tỷ lệ không gian được hỗ trợ, sự lấy mẫu lên có thể được bao gồm. Bằng cách lấy mẫu lên, độ phân giải của lớp thấp hơn có thể được điều chỉnh thành độ phân giải của lớp hiện thời, và bộ lọc nội suy được tạo cấu hình trước có thể được sử dụng. Ví dụ, bộ lọc nội suy dựa trên phép biến đổi cosin rời rạc (discrete cosine transform, DCT) (DCT-IF) có thể được sử dụng. Để vector chuyển động của lớp thấp hơn được sử dụng để mã hóa vector chuyển động lớp cao hơn, sự ánh xạ của vector chuyển động

của lớp thấp hơn để khớp với độ phân giải của lớp cao hơn có thể được bao gồm.

Như được mô tả ở trên, sự mã hóa lớp cao hơn có thể được thực hiện với tham chiếu đến lớp thấp hơn, và khối bộ mã hóa của Fig.10 có thể bao gồm các phần tử của bộ mã hóa được mô tả ở trên dựa vào Fig.1. Dữ liệu được mã hóa của video có thể được đóng gói theo các đơn vị của các lớp trừu tượng mạng (NAL), và phần đầu của đơn vị NAL có thể bao gồm bộ định danh lớp (nuh_layer_id) để biểu thị lớp bao gồm dữ liệu được bao gồm trong đơn vị NAL tương ứng. Nghĩa là, các đơn vị NAL được biểu thị rằng được bao gồm trong các lớp tương ứng có thể được phân loại đối với mỗi lớp bởi các bộ định danh lớp được bao gồm trong các phần đầu của các đơn vị NAL. Dòng bit được tạo ra đối với mỗi lớp có thể được tạo cấu hình bởi một dòng bit bởi bộ ghép kênh, và dòng bit tương ứng có thể được lưu trữ trong thiết bị lưu trữ hoặc được truyền thông qua mạng.

Fig.11 minh họa sơ đồ khối của bộ giải mã hỗ trợ sơ đồ mã hóa video có thể thay đổi tỷ lệ. Dòng bit đầu vào mà là đầu vào bộ giải mã có thể bao gồm nhiều lớp, và bộ giải mã có thể thu hình ảnh thích hợp cho thiết bị đầu cuối bằng cách giải mã tất cả hoặc một phần của dòng bit được phân chia cho mỗi lớp bởi bộ giải ghép kênh. Dòng bit đầu vào của bộ giải mã có thể bao gồm nhiều phần của thông tin bộ lớp đầu ra (OLS) và thông tin tham chiếu giữa các lớp. Bộ lớp đầu ra là tập hợp các lớp và có thể bao gồm các lớp đầu ra và lớp tham chiếu, và bộ giải mã có thể chọn bộ lớp đầu ra được giải mã trong số nhiều bộ lớp đầu ra. Bộ giải mã có thể trích xuất và giải mã một cách chọn lọc lớp được bao gồm trong bộ lớp đầu ra tương ứng từ dòng bit đầu vào, và lớp không được bao gồm trong bộ lớp đầu ra có thể được loại bỏ khỏi dòng bit đầu vào. Ví dụ, để bộ giải mã của thiết bị đầu cuối thu hình ảnh có độ phân giải 4K, dòng bit tương ứng với lớp 1 mà tương ứng với lớp đầu ra và lớp 0 mà là lớp tham chiếu của lớp 1 có thể được giải mã. Vì lớp 2 không được bao gồm trong bộ lớp đầu ra, dòng bit tương ứng với lớp 2 có thể không được giải mã. Ví dụ khác là, để bộ giải mã của thiết bị đầu cuối thu hình ảnh có độ phân giải 8K, dòng bit tương ứng với lớp 2 mà tương ứng với lớp đầu ra, lớp 1 mà là lớp tham chiếu của lớp 2, và lớp 0 mà là lớp tham chiếu của lớp 1 có thể được giải mã. Khối bộ giải mã của Fig.11 có thể bao gồm các phần tử của bộ giải mã được mô tả trên Fig.2. Giống như bộ mã hóa, phép dự đoán liên lớp có thể được thực hiện để làm giảm phần dư giữa các lớp. Nghĩa là, ảnh

của lớp cao hơn có thể được giải mã dựa trên mẫu hoặc phần tử cú pháp của ảnh đã được giải mã của lớp thấp hơn. Ví dụ, ảnh đã được khôi phục của lớp thấp hơn, mà tương ứng với thứ tự đầu ra giống như thứ tự đầu ra của ảnh hiện thời của lớp cao hơn, có thể được sử dụng làm ảnh tham chiếu của ảnh hiện thời. Điều này biểu thị rằng ảnh được khôi phục có thể được sử dụng làm khối tham chiếu để dự đoán bố cục ảnh hiện thời hoặc có thể được sử dụng để mã hóa tập thông tin chuyển động. Giống như bộ mã hóa, ảnh được khôi phục của lớp thấp hơn được sử dụng làm ảnh tham chiếu liên lớp của lớp cao hơn có thể được xử lý thành dạng mà có thể được tham chiếu bởi lớp cao hơn bằng cách xử lý liên lớp, và sau đó được lưu trữ trong DPB. Ví dụ, khi khả năng thay đổi tỷ lệ không gian được hỗ trợ, sự lấy mẫu lên đối với ảnh của lớp thấp hơn có thể được thực hiện, và sự ánh xạ của vector chuyển động để khớp với lớp cao hơn có thể được thực hiện.

Fig.12 minh họa cấu trúc của dòng bit bao gồm nhiều lớp. Dữ liệu bao gồm bộ thông số và ảnh được mã hóa có thể được bao gồm trong dòng bit ở dạng đơn vị NAL, và dòng bit có thể bao gồm một hoặc nhiều lớp. Đơn vị NAL có thể được phân loại thành đơn vị NAL lớp mã hóa video (VCL) và đơn vị NAL phi VCL theo việc liệu dữ liệu được bao gồm trong đơn vị NAL là dữ liệu (nghĩa là, dữ liệu liên quan đến chính video) cho phần chia được mã hóa cấu thành ảnh hay là dữ liệu khác chẳng hạn như bộ thông số. Phần đầu đơn vị NAL có thể bao gồm thông tin loại đơn vị NAL (`nal_unit_type`) để biểu thị loại dữ liệu được bao gồm trong đơn vị NAL. Phần đầu đơn vị NAL có thể bao gồm bộ định danh lớp (`nuh_layer_id`) để biểu thị lớp, mà dữ liệu được bao gồm trong đơn vị NAL thuộc về, và `nuh_layer_id` trong tất cả các đơn vị NAL được bao gồm trong lớp cụ thể có cùng giá trị. Lớp có thể bao gồm các lớp con thời gian, và phần đầu của đơn vị NAL có thể bao gồm bộ định danh thời gian (`TemporalId`) để biểu thị lớp con mà dữ liệu được bao gồm trong đơn vị NAL thuộc về. `TemporalId` trong tất cả các đơn vị NAL được bao gồm trong lớp con cụ thể của lớp có thể có cùng giá trị. Fig.12 minh họa ví dụ trong đó dòng bit bao gồm 3 lớp (lớp 0, lớp 1 và lớp 2), và mỗi lớp bao gồm 3 lớp con (lớp con 0, lớp con 1 và lớp con 2). Trên Fig.12, các hình chữ nhật được thể hiện bởi các đường nét liền biểu thị các ảnh và các đơn vị ảnh (PU) được mã hóa, và các cột của các đơn vị ảnh được biểu thị bởi các đường nét liền biểu thị các đơn vị truy cập (AU). Đơn vị truy cập là tập hợp các đơn vị

ảnh thuộc về các lớp khác nhau, và các sự định thời đầu ra của các ảnh được mã hóa được bao gồm trong mỗi đơn vị ảnh được bao gồm trong đơn vị truy cập có thể là giống nhau. Đơn vị ảnh là tập hợp các đơn vị NAL bao gồm một ảnh được mã hóa, và ảnh được mã hóa là ảnh mà đã được mã hóa và có thể bao gồm các đơn vị NAL VCL bao gồm tất cả các đơn vị cây mã hóa của ảnh.

Trình tự của các đơn vị truy cập trên Fig.12 có thể cấu thành trình tự video được mã hóa (coded video sequence, CVS). CVS là trình tự của các đơn vị truy cập, và có thể bao gồm AU bắt đầu trình tự video được mã hóa (coded video sequence start, CVSS) và các AU tiếp theo đó. PU được bao gồm trong AU CVSS là PU bắt đầu trình tự video lớp được mã hóa (coded layer video sequence start, CLVSS), và ảnh được mã hóa được bao gồm trong PU CLVSS là ảnh bắt đầu trình tự video lớp được mã hóa (CLVSS). Ảnh CLVSS có thể là ảnh làm mới giải mã dần dần (gradual decoding refresh, GDR) hoặc ảnh điểm truy cập ngẫu nhiên nội ảnh (intra truy cập ngẫu nhiên point, IRAP) trong đó kiểu ảnh được mã hóa là ảnh I để hỗ trợ truy cập ngẫu nhiên. Trình tự video lớp được mã hóa (CLVS) là trình tự của các đơn vị ảnh có cùng `nuh_layer_id` (nghĩa là, trình tự của các đơn vị ảnh trong cùng lớp), và có thể bao gồm PU CLVSS và các PU tiếp theo đó. Nếu đơn vị truy cập thứ nhất của Fig.12 là AU CVSS, trình tự của các AU bao gồm 5 AU có thể là CVS. CVS có thể đề cập đến bộ thông số video (VPS), và các thông số được xác định trong VPS có thể được tham chiếu đến trong CVS. Nếu tất cả các đơn vị ảnh của đơn vị truy cập thứ nhất của Fig.12 là các PU CLVSS, trình tự của các PU được bao gồm trong mỗi lớp có thể là CLVS, và 3 CLVS có thể tồn tại. CLVS có thể tham chiếu đến bộ thông số trình tự (SPS), và các thông số được xác định trong SPS có thể được tham chiếu đến trong CLVS. Ảnh được mã hóa được bao gồm trong đơn vị ảnh có thể tham chiếu đến bộ thông số ảnh (PPS), và các thông số được xác định trong PPS có thể được tham chiếu trong ảnh.

Fig.13 là sơ đồ minh họa mối tương quan tham chiếu giữa các ảnh trong dòng bit bao gồm nhiều lớp. Ví dụ được đưa ra bao gồm 2 lớp, và mỗi lớp bao gồm 4 lớp con. Mỗi hình chữ nhật là ảnh, và số trong hình chữ nhật biểu thị thứ tự mã hóa (các thứ tự mã hóa và giải mã) của ảnh. Các ảnh của Fig.13 được liệt kê theo thứ tự đầu ra, và ảnh ở bên trái biểu thị rằng ảnh được kết xuất trước ảnh ở bên phải.

Khi mã hóa ảnh được bao gồm trong lớp cụ thể, ảnh được mã hóa trước đó trong cùng lớp có thể được sử dụng làm ảnh tham chiếu. Ví dụ, ảnh được mã hóa thứ nhất của lớp 1 có thể sử dụng ảnh được mã hóa thứ 0 làm ảnh tham chiếu, và ảnh được mã hóa thứ ba có thể sử dụng, làm các ảnh tham chiếu, ảnh được mã hóa thứ 0 và ảnh được mã hóa thứ hai. Trong mối tương quan tham chiếu giữa các ảnh, vì ảnh có TemporalId nhỏ không thể tham chiếu đến ảnh có TemporalId lớn, thậm chí nếu lớp con có TemporalId lớn hơn giá trị cụ thể không được giải mã, ảnh được bao gồm trong lớp con có TemporalId nhỏ hơn hoặc bằng giá trị cụ thể có thể được giải mã. Ví dụ, thậm chí nếu lớp con 3 của lớp 1 không được giải mã, các ảnh được bao gồm trong lớp 0, lớp 1 và lớp 2 có thể được khôi phục hoàn toàn. Bộ giải mã có thể điều chỉnh tốc độ khung hình bằng cách sử dụng mối tương quan tham chiếu này.

Không chỉ có ảnh của cùng lớp mà ảnh của lớp thấp hơn cũng có thể được sử dụng làm ảnh tham chiếu bằng phép dự đoán liên lớp. Như được mô tả ở trên, phép dự đoán liên lớp có thể được sử dụng để làm giảm phần dư tồn tại giữa các lớp. Khi mã hóa ảnh hiện thời của lớp cao hơn, ảnh được khôi phục của lớp thấp hơn có thể được sử dụng làm ảnh tham chiếu, và ảnh được khôi phục của lớp thấp hơn được bao gồm trong đơn vị truy cập giống như ảnh hiện thời có thể được sử dụng làm ảnh tham chiếu liên lớp. Giống như ảnh tham chiếu chung, ảnh tham chiếu liên lớp có thể được lưu trữ trong DPB, và có thể được sử dụng để mã hóa tập thông tin chuyển động hoặc phép dự đoán bố cục của khối trong ảnh hiện thời. Trong trường hợp này, các ảnh trong đó TemporalId của lớp tham chiếu lớn hơn giá trị cụ thể có thể không được sử dụng làm các ảnh tham chiếu liên lớp. Ảnh trong đó TemporalId (Tid) của lớp 0 là 0 có thể được sử dụng làm ảnh tham chiếu liên lớp của ảnh được bao gồm trong cùng AU của lớp 1. Tương tự, ảnh trong đó TemporalId (Tid) của lớp 0 là 1 có thể được sử dụng làm ảnh tham chiếu liên lớp của ảnh được bao gồm trong cùng AU của lớp 1. Mặt khác, ảnh trong đó TemporalId (Tid) của lớp 0 là 2 hoặc 3 không được sử dụng làm ảnh tham chiếu liên lớp của lớp 1. Trong trường hợp này, nếu bộ lớp đầu ra bao gồm lớp 0 và lớp 1, và chỉ lớp 1 được sử dụng làm đầu ra, lớp con trong đó TemporalId của lớp 0 là 2 hoặc 3 không được sử dụng cho lớp 1 mà là lớp đầu ra, đơn vị NAL tương ứng có thể không được giải mã. Theo cách này, theo cấu hình của bộ lớp đầu

ra, hình ảnh thích hợp cho thiết bị đầu cuối có thể được khôi phục hiệu quả bằng cách trích xuất, từ dòng bit, chỉ lớp được sử dụng làm đầu ra hoặc được sử dụng làm tham chiếu hoặc một số lớp con của lớp, và giải mã lớp này.

Fig.14 minh họa các phần của cấu trúc cú pháp cấp cao (high-level syntax, HLS) và các mối tương quan tham chiếu giữa chúng. Đối với dòng bit được tạo ra bởi bộ mã hóa, bộ giải mã có thể giải mã dòng bit theo chuỗi các quy tắc định trước và có thể thu hình ảnh được khôi phục. Mỗi đoạn thông tin cấu thành dòng bit được tạo ra bởi chuỗi các quy tắc định trước được gọi là phần tử cú pháp, và cấu hình của các phần tử cú pháp này có thể được gọi là cấu trúc cú pháp. Các cú pháp được căn chỉnh theo các đơn vị của byte và được đóng gói theo các đơn vị của đơn vị NAL, mà có thể được gọi là tải tin trình tự byte thô (raw byte sequence payload, RBSP). Theo loại dữ liệu được bao gồm trong đơn vị NAL, đơn vị NAL có thể được phân loại thành đơn vị NAL VCL và đơn vị NAL phi VCL, và đơn vị NAL tương ứng với phi VCL có thể bao gồm cú pháp RBSP thông tin năng lực giải mã (DCI), cú pháp RBSP bộ thông số video (VPS), cú pháp RBSP bộ thông số trình tự (SPS), cú pháp RBSP bộ thông số ảnh (PPS), cú pháp RBSP phần đầu ảnh (PH) và dạng tương tự. Trong đơn vị NAL tương ứng với VCL, có thể có cú pháp RBSP lớp phân chia bao gồm dữ liệu phân chia, và dạng tương tự. Dòng bit được mã hóa có thể được tạo cấu hình trong trình tự của đơn vị NAL VPS, đơn vị NAL SPS, đơn vị NAL PPS, đơn vị NAL PH, đơn vị NAL lớp phân chia, và dạng tương tự, và đơn vị NAL cụ thể có thể không tồn tại. Trình tự và số lượng có thể thay đổi tùy thuộc vào môi trường nén hình ảnh. Đơn vị NAL bộ phân cách đơn vị truy cập (access unit delimiter, AUD) và đơn vị NAL DCI có thể ưu tiên cú pháp RBSP VPS. Trên Fig.14, mối tương quan tham chiếu giữa cấu trúc cú pháp cao hơn và cấu trúc cú pháp thấp hơn có thể được xác định gián tiếp. Các phần tử cú pháp được sử dụng chung trong trình tự video, đơn vị ảnh (ảnh con), hoặc đơn vị phân chia có thể có mối quan hệ như được thể hiện trên Fig.14. Ví dụ, các thông số được xác định trong cú pháp RBSP VPS được tham chiếu dựa trên `sps_vps_id` được xác định trong cú pháp RBSP SPS có thể được áp dụng chung cho trình tự video tương ứng. Một số thông số có thể được xác định lại/được cập nhật trong cú pháp thấp hơn. Trong phần mô tả sau, khi thuật ngữ đối với cấu trúc cú pháp được sử dụng, thậm chí nếu chỉ tên gọi một phần được

sử dụng thay vì tên gọi đầy đủ, điều này có thể biểu thị rằng tên gọi một phần được bao gồm trong cấu trúc cú pháp. Việc báo hiệu/phân tích đối với cấu trúc cú pháp và phần tử cú pháp của sáng chế có thể được áp dụng tương đương tại thời điểm mã hóa/giải mã, trong đó nó có thể được xem xét rằng việc báo hiệu được thực hiện trong khi mã hóa, và việc phân tích được thực hiện trong khi giải mã do việc đọc và diễn dịch từ dòng bit, nhưng các ý nghĩa của chúng có thể là tương tự.

Fig.15 minh họa cấu trúc cú pháp bộ thông số video (VPS). Bộ thông số video (VPS) có thể bao gồm thông tin lớp, thông tin bộ lớp đầu ra, cấu hình, hạng, thông tin cấp, thông tin bộ đệm ảnh được giải mã (DPB), và thông tin bộ giải mã tham chiếu giả định (HRD), và thông tin tương tự, và thông tin được xác định trong VPS có thể được áp dụng cho CVS mà tham chiếu VPS.

Cột thứ hai của Fig.15 biểu thị bộ mô tả cho phần tử cú pháp và có thể biểu thị cách mà phần tử cú pháp được phân tích. Trường hợp trong đó bộ mô tả là $u(n)$ có thể biểu thị rằng phần tử cú pháp là số nguyên không dấu và được biểu diễn bởi độ dài cố định là n bit. Trong trường hợp này, n bit được đọc từ dòng bit, và dấu biểu thị vị trí bit cụ thể trong dòng bit có thể được di chuyển bởi vị trí n -bit. Chuỗi bit nhị phân có độ dài n được thu từ dòng bit có thể được diễn dịch dưới dạng phép biểu diễn nhị phân của số nguyên không dấu trong đó bit có ý nghĩa nhiều nhất (most significant bit, MSB) được ghi đầu tiên (hoặc ngoài cùng bên trái). Ví dụ, nếu bộ mô tả cho phần tử cú pháp là $u(2)$ và chuỗi bit được đọc từ dòng bit là “10”, điều này có thể biểu thị rằng giá trị của phần tử cú pháp là 2.

Đầu tiên, phần tử cú pháp `vps_video_parameter_set_id` có thể được bao gồm trong RBSP bộ thông số video. `vps_video_parameter_set_id` có thể biểu thị bộ định danh để tham chiếu bởi phần tử cú pháp khác hoặc cấu trúc cú pháp khác. Ví dụ, RBSP bộ thông số trình tự (SPS) có thể bao gồm bộ định danh của VPS được tham chiếu, và các thông số của VPS được biểu thị bởi bộ định danh có thể được tham chiếu. Giá trị của `vps_video_parameter_set_id` có thể lớn hơn giá trị được tạo cấu hình trước. Giá trị được tạo cấu hình trước có thể là 0. Trong trường hợp mã hóa video đơn lớp, thông tin, chẳng hạn như đa lớp và cấu hình bộ lớp đầu ra được thu từ VPS, có thể là không cần thiết. Theo

đó, nếu bộ định danh của VPS được tham chiếu bởi RBSP SPS là 0, thì VPS có thể không được tham chiếu. Do đó, đơn vị NAL bao gồm dữ liệu VPS có thể không tồn tại trong dòng bit được mã hóa trong lớp duy nhất. Mặt khác, nếu bộ định danh của VPS được tham chiếu lớn hơn 0, các thông số của VPS được biểu thị bởi bộ định danh có thể được tham chiếu đến.

Sau đó, phần tử cú pháp của `vps_max_layers_minus1` có thể được bao gồm trong RBSP bộ thông số video. `vps_max_layers_minus1+1` xác định số lượng lớp tối đa được cho phép trong CVS tham chiếu VPS tương ứng.

Sau đó, phần tử cú pháp của `vps_max_sublayers_minus1` có thể được bao gồm trong RBSP bộ thông số video. `vps_max_sublayers_minus1+1` xác định số lượng lớp con thời gian tối đa mà có thể tồn tại trong lớp riêng lẻ trong CVS tham chiếu VPS tương ứng. Giá trị của `vps_max_sublayers_minus1` có thể là giá trị nằm trong khoảng từ 0 đến 6. Điều này có thể biểu thị rằng số lượng lớp con thời gian tối đa mà có thể tồn tại trong lớp riêng lẻ trong CVS tham chiếu VPS là bảy.

Sau đó, nếu CVS đề cập đến VPS có thể bao gồm nhiều lớp và mỗi lớp riêng lẻ có thể bao gồm nhiều lớp con thời gian (nếu `vps_max_layers_minus1 > 0` và `vps_max_sublayers_minus1 > 0`), phần tử cú pháp của `vps_all_layers_same_num_sublayers_flag` có thể được bao gồm trong RBSP bộ thông số video. `vps_all_layers_same_num_sublayers_flag` có thể là thông tin biểu thị xem tất cả các lớp cấu thành CVS có bao gồm số lượng lớp con thời gian giống nhau hay không. Nếu `vps_all_layers_same_num_sublayers_flag` là 1, điều này xác định rằng số lượng lớp con thời gian của tất cả các lớp trong CVS tham chiếu VPS là giống nhau. Mặt khác, nếu `vps_all_layers_same_num_sublayers_flag` là 0, điều này xác định rằng số lượng lớp con thời gian của các lớp trong CVS tham chiếu VPS có thể là giống nhau hoặc khác nhau. Nếu CVS tham chiếu VPS bao gồm lớp duy nhất (nếu `vps_max_layers_minus1` là 0), hoặc nhiều nhất một lớp con thời gian có thể được bao gồm trong mỗi lớp (nếu `vps_max_sublayers_minus1` là 0), `vps_all_layers_same_num_sublayers_flag` có thể không được bao gồm trong dòng bit, và phần tử cú pháp có thể được suy ra là 1.

Sau đó, nếu CVS tham chiếu VPS có thể bao gồm nhiều lớp (nếu $vps_max_layers_minus1 > 0$), phần tử cú pháp $vps_all_independent_layers_flag$ biểu thị sự phụ thuộc giữa các lớp có thể được bao gồm trong RBSP bộ thông số video. Như được mô tả dựa vào Fig.13, trong ảnh trong lớp cụ thể, bằng phép dự đoán liên lớp (ILP), bố cục có thể được dự đoán (giá trị điểm ảnh được dự đoán) từ ảnh được mã hóa trong lớp thấp hơn khác, hoặc thông tin chuyển động có thể được mã hóa nhờ sử dụng tập thông tin chuyển động tương ứng với ảnh được mã hóa trong lớp khác. Điều này có thể biểu thị rằng ảnh được mã hóa của lớp thấp hơn có thể được sử dụng làm ảnh tham chiếu của ảnh hiện thời. Theo sáng chế, phép dự đoán liên lớp chủ yếu được mô tả sao cho ảnh được mã hóa trong lớp thấp hơn khác được sử dụng làm ảnh tham chiếu, nhưng sáng chế không bị giới hạn ở đó. Phép dự đoán liên lớp có thể biểu thị rằng ảnh được mã hóa hoặc các phần tử cú pháp của lớp thấp hơn được sử dụng cho ảnh mã hóa trong lớp hiện thời. Trường hợp trong đó lớp hiện thời sử dụng phép dự đoán liên lớp biểu thị rằng có sự phụ thuộc vào lớp duy nhất hoặc nhiều lớp khác với lớp hiện thời, mà có thể biểu thị rằng, để ảnh của lớp hiện thời được mã hóa, thì ảnh của lớp duy nhất hoặc ảnh của nhiều lớp có sự phụ thuộc nên được mã hóa trước tiên. Mặt khác, khi lớp hiện thời không sử dụng phép dự đoán liên lớp, thì lớp có thể được mã hóa một cách độc lập bất kể lớp khác. Nếu $vps_all_independent_layers_flag$ là 1, điều này xác định rằng tất cả các lớp trong CVS tham chiếu VPS được mã hóa một cách độc lập mà không sử dụng phép dự đoán liên lớp. Mặt khác, nếu $vps_all_independent_layers_flag$ là 0, thì điều này xác định rằng một hoặc nhiều lớp trong CVS tham chiếu VPS có thể sử dụng phép dự đoán liên lớp. Điều này biểu thị rằng có thể có sự phụ thuộc giữa các lớp nhất định. Khi CVS tham chiếu VPS bao gồm lớp duy nhất ($vps_max_layers_minus1$ là 0), thì $vps_all_independent_layers_flag$ có thể không được bao gồm trong dòng bit, và phần tử cú pháp có thể được suy ra là 1.

Sau đó, trong câu lệnh vòng lặp (câu lệnh “for”), trong khi chỉ số lớp i được tăng thêm 1 trong khoảng từ 0 đến $vps_max_layers_minus1$, thông tin về các lớp cấu thành CVS có thể được bao gồm trong RBSP bộ thông số video.

Cụ thể là, thông tin bộ định danh lớp được bao gồm trong phần đầu đơn vị NAL có thể được bao gồm. RBSP phần đầu đơn vị NAL có thể bao gồm phần tử cú pháp

nuh_layer_id, và điều này có thể biểu thị bộ định danh để xác định lớp mà dữ liệu VCL hoặc dữ liệu phi VCL được bao gồm trong đơn vị NAL tương ứng thuộc về. vps_layer_id[i] biểu thị nuh_layer_id được bao gồm trong phần đầu đơn vị NAL của đơn vị NAL tương ứng với lớp thứ i. Đối với hai số nguyên không âm là m và n, nếu m nhỏ hơn n, thì giá trị của vps_layer_id[m] có thể nhỏ hơn giá trị của vps_layer_id[n].

Nếu lớp không phải là lớp thứ 0 (lớp thấp nhất) (chỉ số lớp i lớn hơn 0), và một hoặc nhiều lớp có thể sử dụng phép dự đoán liên lớp (vps_all_independent_layers_flag là 0), thông tin biểu thị sự phụ thuộc và mối tương quan tham chiếu giữa các lớp có thể được bao gồm trong RBSP bộ thông số video.

Phần tử cú pháp vps_independent_layer_flag[i] có thể biểu thị sự phụ thuộc của lớp, mà có chỉ số lớp là i, vào lớp khác. Trường hợp trong đó vps_independent_layer_flag[i] là 1 xác định rằng lớp có chỉ số i không sử dụng phép dự đoán liên lớp, và điều này có thể biểu thị rằng lớp này có thể được mã hóa một cách độc lập bất kể lớp khác. Mặt khác, trường hợp trong đó vps_independent_layer_flag[i] là 0 xác định rằng lớp có chỉ số i có thể sử dụng phép dự đoán liên lớp, và điều này có thể biểu thị rằng lớp này có thể có sự phụ thuộc vào lớp khác. Nếu vps_independent_layer_flag[i] là 0, thì phần tử cú pháp vps_direct_ref_layer_flag[i][j] để biểu thị sự phụ thuộc vào lớp khác có thể tồn tại trong RBSP VPS, và chỉ số j có thể được xác định nằm trong khoảng từ 0 đến i-1. Trong trường hợp lớp thứ 0 (chỉ số i là 0), hoặc nếu tất cả các lớp không sử dụng phép dự đoán liên lớp (nếu vps_all_independent_layers_flag là 1), thì vps_independent_layer_flag[i] có thể không tồn tại trong VPS và có thể được suy ra là 1. Theo đó, ảnh của lớp thấp nhất có chỉ số i là 0 có thể được mã hóa một cách độc lập bất kể ảnh của lớp khác.

Nếu lớp thứ i có thể sử dụng phép dự đoán liên lớp (nếu vps_independent_layer_flag[i] là 0), thông tin về lớp tham chiếu có thể được bao gồm trong RBSP bộ thông số video.

Như được mô tả dựa vào Fig.13, lớp cao hơn có thể thực hiện phép dự đoán liên lớp với tham chiếu đến lớp thấp hơn. Ví dụ, nếu lớp hiện thời là lớp thứ i, lớp có chỉ số lớp nằm trong khoảng từ 0 đến i-1 có thể được sử dụng làm lớp tham chiếu. Điều này biểu thị

rằng khi ảnh hiện thời của lớp có chỉ số lớp là i được mã hóa, các ảnh được bao gồm trong AU giống như ảnh hiện thời trong số các ảnh của các lớp có chỉ số lớp nằm trong khoảng từ 0 đến $i-1$ có thể được sử dụng làm các ảnh tham chiếu. Để thực hiện điều này, trong cú pháp vòng lặp “for”, phần tử cú pháp `vps_direct_ref_layer_flag[i][j]` có thể được phân tích từ dòng bit trong khi j trong chỉ số j được tăng thêm 1 trong khoảng từ 0 đến $i-1$. Trường hợp trong đó `vps_direct_ref_layer_flag[i][j]` là 0 có thể biểu thị rằng lớp có chỉ số lớp j không phải là lớp tham chiếu trực tiếp của lớp có chỉ số lớp i . Mặt khác, trường hợp trong đó `vps_direct_ref_layer_flag[i][j]` là 1 có thể biểu thị rằng lớp có chỉ số lớp j là lớp tham chiếu trực tiếp của lớp có chỉ số lớp i . Điều này biểu thị rằng phép dự đoán liên lớp có thể được thực hiện nhờ sử dụng lớp thứ j làm lớp tham chiếu đối với lớp thứ i . Trường hợp trong đó `vps_direct_ref_layer_flag[i][j]` không tồn tại trong RBSP VPS có thể được suy ra là 0, và phạm vi của chỉ số j có thể nằm trong khoảng từ 0 đến $i-1$. Trường hợp trong đó `vps_independent_layer_flag[i]` là 0 biểu thị rằng lớp tương ứng có thể sử dụng lớp khác làm lớp tham chiếu, và do đó có thể yêu cầu ít nhất một `vps_direct_ref_layer_flag[i][j]` là 1. Chỉ số j có thể nằm trong khoảng từ 0 đến $i-1$.

Theo phương án nêu trên, thậm chí nếu lớp thứ j không phải là lớp tham chiếu trực tiếp của lớp thứ i , thì có thể cần có lớp thứ j để mã hóa và giải mã lớp thứ i . Ví dụ, nếu lớp có chỉ số lớp là j là lớp tham chiếu trực tiếp đối với lớp có chỉ số lớp là k (nếu `vps_direct_ref_layer_flag[k][j]` là 1), và nếu lớp có chỉ số lớp là k là lớp tham chiếu trực tiếp đối với lớp có chỉ số lớp là i (`vps_direct_ref_layer_flag[i][k]` là 1), thì lớp thứ j không được sử dụng làm lớp tham chiếu trực tiếp của lớp thứ i , nhưng có thể cần thiết để giải mã lớp thứ i . Điều này sẽ được mô tả sau với tham chiếu đến các hình vẽ liên quan.

Ngoài ra, khi lớp thứ i có thể sử dụng phép dự đoán liên lớp (nếu `vps_independent_layer_flag[i]` là 0), thì thông tin về bộ định danh thời gian mà có thể được sử dụng làm ảnh tham chiếu liên lớp (ILRP) có thể được bao gồm trong RBSP bộ thông số video.

Ảnh tham chiếu liên lớp là ảnh có `nuh_layer_id` nhỏ hơn `nuh_layer_id` của ảnh hiện thời, trong số các ảnh được bao gồm trong AU giống như ảnh hiện thời và có thể được sử

dùng làm ảnh tham chiếu đối với ảnh hiện thời bằng phép dự đoán liên lớp. Nghĩa là, điều này biểu thị rằng, trong số các ảnh được bao gồm trong AU giống như ảnh hiện thời, các ảnh tương ứng với lớp thấp hơn so với lớp hiện thời có thể được sử dụng làm các ảnh tham chiếu.

RBSP phần đầu đơn vị NAL có thể bao gồm phần tử cú pháp `nuh_temporal_id_plus1` mà là bộ định danh thời gian cho đơn vị NAL. Biến của `TemporalId` (hoặc `Tid`) thể hiện bộ định danh thời gian dựa trên phần tử cú pháp `nuh_temporal_id_plus1` có thể được suy ra dựa trên `nuh_temporal_id_plus1-1`. Giá trị của `nuh_temporal_id_plus1` có thể lớn hơn 0, và `TemporalId` của tất cả các đơn vị NAL VCL được bao gồm trong AU có thể có cùng giá trị.

Nếu `vps_direct_ref_layer_flag[i][j]` là 1, thì ảnh của lớp có chỉ số lớp là j có thể được sử dụng làm ảnh tham chiếu liên lớp đối với ảnh của lớp có chỉ số lớp là i trong cùng AU. Trong trường hợp này, các ảnh trong đó `TemporalId` của lớp tham chiếu lớn hơn giá trị cụ thể có thể không được sử dụng làm các ảnh tham chiếu liên lớp. Ví dụ, trên Fig.13, ảnh trong đó `TemporalId` (`Tid`) của lớp 0 là 0 có thể được sử dụng làm ảnh tham chiếu liên lớp của ảnh được bao gồm trong cùng AU của lớp 1. Tương tự, ảnh trong đó `TemporalId` (`Tid`) của lớp 0 là 1 có thể được sử dụng làm ảnh tham chiếu liên lớp của ảnh được bao gồm trong cùng AU của lớp 1. Mặt khác, ảnh trong đó `TemporalId` (`Tid`) của lớp 0 là 2 hoặc 3 không được sử dụng làm ảnh tham chiếu liên lớp của lớp 1. Trong trường hợp này, nếu bộ lớp đầu ra (OLS) bao gồm lớp 0 và lớp 1, và chỉ lớp 1 được kết xuất, các lớp con có `TemporalId` là 2 và 3 của lớp 0 không được sử dụng cho lớp 1, sao cho đơn vị NAL tương ứng có thể không được giải mã. Theo cách này, theo cấu hình của bộ lớp đầu ra, hình ảnh thích hợp cho thiết bị đầu cuối có thể được khôi phục hiệu quả bằng cách trích xuất, từ dòng bit, chỉ lớp được sử dụng để kết xuất hoặc được sử dụng làm tham chiếu hoặc một số lớp con của lớp, và giải mã lớp này.

Phần tử cú pháp `max_tid_ref_present_flag[i]` có thể được bao gồm trong RBSP bộ thông số video. Trường hợp trong đó `max_tid_ref_present_flag[i]` là 1 có thể xác định rằng phần tử cú pháp `max_tid_il_ref_pics_plus1[i]` tồn tại. Mặt khác, trường hợp trong đó

$\text{max_tid_ref_present_flag}[i]$ là 0 có thể xác định rằng phần tử cú pháp $\text{max_tid_il_ref_pics_plus1}[i]$ không tồn tại.

Nếu $\text{max_tid_ref_present_flag}[i]$ là 1, thì thông tin (phần tử cú pháp $\text{max_tid_il_ref_pics_plus1}[i]$) làm giới hạn TemporalId của ảnh mà có thể được sử dụng làm ảnh tham chiếu liên lớp có thể được bao gồm trong RBSP bộ thông số video. Trường hợp trong đó $\text{max_tid_il_ref_pics_plus1}[i]$ là 0 có thể xác định rằng các ảnh (các ảnh phi IRAP) khác với ảnh IRAP không được sử dụng làm các ảnh tham chiếu liên lớp để mã hóa ảnh của lớp thứ i . Nghĩa là, điều này biểu thị rằng chỉ ảnh IRAP của lớp tham chiếu có thể được sử dụng làm ảnh tham chiếu liên lớp đối với ảnh của lớp thứ i . Trường hợp trong đó $\text{max_tid_il_ref_pics_plus1}[i]$ lớn hơn 0 có thể xác định rằng ảnh có TemporalId lớn hơn $\text{max_tid_il_ref_pics_plus1}[i]-1$ không được sử dụng làm ảnh tham chiếu liên lớp để mã hóa ảnh của lớp thứ i . Khi $\text{max_tid_il_ref_pics_plus1}[i]$ không tồn tại, thì giá trị của $\text{max_tid_il_ref_pics_plus1}[i]$ có thể được suy ra là 7.

Như được mô tả ở trên, phần tử cú pháp $\text{max_tid_il_ref_pics_plus1}[i]$ làm giới hạn TemporalId cho ảnh tham chiếu liên lớp của lớp tham chiếu mà có thể được sử dụng để mã hóa ảnh của lớp thứ i , sao cho, theo cấu hình của bộ lớp đầu ra, lớp con trong đó TemporalId của lớp tham chiếu lớn hơn hoặc bằng $\text{max_tid_il_ref_pics_plus1}[i]$ có thể không được giải mã. Phương án được mô tả trên Fig.13 đề cập đến trường hợp trong đó giá trị của $\text{max_tid_il_ref_pics_plus1}[1]$ cho lớp 1 là 2, và điều này biểu thị rằng các ảnh có TemporalId là 2 và 3 của lớp 0 không được sử dụng làm các ảnh tham chiếu liên lớp đối với ảnh của lớp 1. Nếu giá trị của $\text{max_tid_il_ref_pics_plus1}[1]$ là 0, thì các ảnh phi IRAP của lớp 0 không được sử dụng làm các ảnh tham chiếu liên lớp, và do đó chỉ ảnh IRAP của lớp 0 có thể được sử dụng làm ảnh tham chiếu liên lớp của các ảnh được bao gồm trong lớp 1.

Nếu nhiều lớp có thể tồn tại (nếu $\text{vps_max_layers_minus1}$ lớn hơn 0), thông tin về lớp được sử dụng làm đầu ra có thể được bao gồm trong RBSP bộ thông số video. Bộ lớp đầu ra (OLS) có thể biểu thị bộ lớp bao gồm một hoặc nhiều lớp được sử dụng làm các đầu ra đối với lớp duy nhất hoặc nhiều lớp. Trong VPS, một hoặc nhiều bộ lớp đầu ra có thể

được xác định theo phương pháp tạo cấu hình bộ lớp đầu ra, và thông tin lớp đầu ra về mỗi bộ lớp đầu ra có thể được bao gồm trong RBSP bộ thông số video.

Nếu tất cả các lớp trong CVS tham chiếu VPS được mã hóa một cách độc lập mà không sử dụng phép dự đoán liên lớp (`vps_all_independent_layers_flag` là 1), thì phần tử cú pháp `each_layer_is_an_ols_flag` có thể được bao gồm trong bộ thông số video RBPS. Nếu `each_layer_is_an_ols_flag` là 1, thì mỗi trong số các bộ lớp đầu ra có thể chỉ bao gồm một lớp, và điều này có thể xác định rằng mỗi lớp trong CVS tham chiếu VPS là bộ lớp đầu ra và là lớp đầu ra duy nhất. Trong trường hợp này, tổng số lượng các bộ lớp đầu ra được xác định trong VPS có thể bằng `vps_max_layers_minus1+1`. Ví dụ, khi tổng ba lớp của lớp 0, lớp 1 và lớp 2 tồn tại trong CVS tham chiếu VPS, ba bộ lớp đầu ra có thể được xác định bởi VPS. Bộ lớp đầu ra thứ 0 có thể bao gồm lớp 0, bộ lớp đầu ra thứ nhất có thể bao gồm lớp 1, và bộ lớp đầu ra thứ hai có thể bao gồm lớp 2. Mặt khác, trường hợp trong đó `each_layer_is_an_ols_flag` là 0 có thể xác định rằng nhiều lớp có thể được bao gồm trong bộ lớp đầu ra. Nếu CVS bao gồm lớp duy nhất (nếu `vps_max_layers_minus1` là 0), thì giá trị của `each_layer_is_an_ols_flag` có thể được suy ra là 1. Nếu CVS tham chiếu VPS có thể bao gồm nhiều lớp (nếu `vps_max_layers_minus1` lớn hơn 0), và một hoặc nhiều lớp trong CVS có thể sử dụng phép dự đoán liên lớp (nếu `vps_all_independent_layers_flag` là 0), thì giá trị của `each_layer_is_an_ols_flag` có thể được suy ra là 0.

Nếu nhiều lớp có thể được bao gồm trong bộ lớp đầu ra (nếu `each_layer_is_an_ols_flag` là 0), thì thông tin về phương pháp tạo cấu hình bộ lớp đầu ra có thể được bao gồm thêm trong RBSP bộ thông số video.

Nếu một hoặc nhiều lớp có thể sử dụng phép dự đoán liên lớp (nếu `vps_all_independent_layers_flag` là 0), thì phần tử cú pháp của `ols_mode_idc` biểu thị chế độ bộ lớp đầu ra có thể được bao gồm trong RBSP bộ thông số video.

Trường hợp trong đó `ols_mode_idc` là 0 có thể xác định rằng số lượng bộ lớp đầu ra được xác định bởi VPS bằng `vps_max_layers_minus1+1`. Ngoài ra, bộ lớp đầu ra thứ i có thể bao gồm các lớp có giá trị chỉ số lớp nằm trong khoảng từ 0 đến i (nghĩa là, các lớp $i+1$), và có thể xác định rằng lớp cao nhất của chúng là lớp đầu ra. Ví dụ, nếu CVS tham

chiều VPS bao gồm tổng ba lớp (lớp 0, lớp 1 và lớp 2), thì bộ lớp đầu ra thứ 0 bao gồm lớp 0, và lớp 0 có thể được sử dụng làm đầu ra. Bộ lớp đầu ra thứ nhất bao gồm lớp 0 và lớp 1, và lớp 1 có thể được sử dụng làm đầu ra. Bộ lớp đầu ra thứ hai bao gồm lớp 0, lớp 1 và lớp 2, và lớp 2 có thể được sử dụng làm đầu ra.

Trường hợp trong đó `ols_mode_idc` là 1 có thể xác định rằng số lượng bộ lớp đầu ra được xác định bởi VPS bằng `vps_max_layers_minus1+1`. Ngoài ra, bộ lớp đầu ra thứ i có thể bao gồm các lớp có giá trị chỉ số lớp nằm trong khoảng từ 0 đến i (nghĩa là, các lớp $i+1$), và có thể xác định rằng tất cả các lớp được bao gồm trong bộ lớp đầu ra là các đầu ra. Ví dụ, nếu CVS tham chiếu VPS bao gồm tổng ba lớp (lớp 0, lớp 1 và lớp 2), thì bộ lớp đầu ra thứ 0 bao gồm lớp 0, và lớp 0 có thể được sử dụng làm đầu ra. Bộ lớp đầu ra thứ nhất bao gồm các lớp 0 và 1, và cả hai lớp 0 và 1 có thể được sử dụng làm các đầu ra. Bộ lớp đầu ra thứ hai bao gồm lớp 0, lớp 1 và lớp 2, và tất cả các lớp trong số lớp 0, lớp 1 và lớp 2 có thể được sử dụng làm các đầu ra.

Trường hợp trong đó `ols_mode_idc` là 2 có thể xác định số lượng bộ lớp đầu ra được xác định bởi VPS và báo hiệu rõ ràng của lớp đầu ra cho mỗi bộ lớp đầu ra, và có thể biểu thị rằng, trong mỗi bộ đầu ra, các lớp còn lại khác với lớp đầu ra là các lớp tham chiếu trực tiếp hoặc gián tiếp của lớp đầu ra được bao gồm trong bộ lớp đầu ra.

Nếu tất cả các lớp của CVS tham chiếu VPS được mã hóa một cách độc lập mà không có phép dự đoán liên lớp (nếu `vps_all_independent_layers_flag` là 1), và `each_layer_is_an_ols_flag` là 0, `ols_mode_idc` có thể không tồn tại, và giá trị của `ols_mode_idc` có thể được suy ra là 2.

Nếu `ols_mode_idc` là 2, thì thông tin biểu thị số lượng bộ lớp đầu ra và thông tin biểu thị các lớp đầu ra được sử dụng trong bộ lớp đầu ra tương ứng có thể được bao gồm trong RBSP bộ thông số video.

Phần tử cú pháp `num_output_layer_sets_minus1` là thông tin biểu thị số lượng bộ lớp đầu ra, và `num_output_layer_sets_minus1+1` có thể xác định số lượng bộ lớp đầu ra được xác định bởi VPS khi `ols_mode_idc` là 2.

Khi biến của `TotalNumOlss` biểu thị số lượng bộ lớp đầu ra được xác định bởi VPS,

thì TotalNumOlss có thể được suy ra theo mã giả sau.

```

if(vps_max_layers_minus==0)

TotalNumOlss=1

else if(each_layer_is_an_ols_flag || ols_mode_idc==0 || ols_mode_idc==1)

TotalNumOlss=vps_max_layers_minus1+1

else if(ols_mode_idc==2)

TotalNumOlss=num_output_layer_set_minus1+1

```

Trong mã giả, nếu CVS tham chiếu VPS đầu tiên là lớp duy nhất, thì số lượng bộ lớp đầu ra có thể là 1. Nếu CVS tham chiếu VPS bao gồm nhiều lớp, và each_layer_is_an_ols_flag là 1 hoặc ols_mode_idc là 0 hoặc 1, thì số lượng bộ lớp đầu ra có thể bằng số lượng tối đa (vps_max_layers_minus1+1) của các lớp được cho phép trong CVS tương ứng. Theo cách khác, nếu ols_mode_idc là 2, thì số lượng bộ lớp đầu ra có thể bằng num_output_layer_set_minus1+1.

Nếu ols_mode_idc là 2, thì thông tin biểu thị lớp đầu ra được sử dụng cho mỗi bộ lớp đầu ra có thể được bao gồm thêm trong RBSP bộ thông số video.

Trong câu lệnh vòng lặp (câu lệnh “for” đối với chỉ số i), câu lệnh vòng lặp đối với chỉ số j có thể được thực hiện trong khi chỉ số i được tăng thêm 1 trong khoảng từ 1 đến num_output_layer_sets_minus1. Chỉ số j có thể được tăng thêm 1 trong khoảng từ 0 đến vps_max_layers_minus1, và các phần tử cú pháp ols_output_layer_flag[i][j] cho các chỉ số i và j có thể được bao gồm trong RBSP bộ thông số video.

Nếu ols_output_layer_flag[i][j] là 1, khi ols_mode_idc là 2, thì điều này có thể xác định rằng lớp trong đó bộ định danh lớp (nuh_layer_id) của phần đầu đơn vị NAL bằng vps_layer_id[j] là lớp đầu ra của bộ lớp đầu ra thứ i. Mặt khác, nếu ols_output_layer_flag[i][j] là 0, khi ols_mode_idc là 2, thì điều này có thể xác định rằng lớp trong đó nuh_layer_id bằng vps_layer_id[j] không phải là lớp đầu ra của bộ lớp đầu ra thứ i. Trong trường hợp bộ lớp đầu ra thứ 0 (bộ lớp đầu ra có chỉ số i bằng 0), chỉ lớp thấp nhất (nghĩa là, lớp trong đó nuh_layer_id bằng vps_layer_id[0]) được bao gồm, và lớp thấp

nhất là lớp đầu ra, sao cho, khi i là 0, thì `ols_output_layer_flag[i][j]` có thể không tồn tại.

Fig.16 minh họa các mã giả suy ra các biến liên quan đến các mối tương quan tham chiếu giữa các lớp. Biến của `NumDirectRefLayers[i]` biểu thị số lượng các lớp tham chiếu trực tiếp của lớp thứ i , và biến của `DirectRefLayerIdx[i][d]` biểu thị chỉ số lớp của lớp tham chiếu trực tiếp thứ d đối với lớp thứ i . Biến của `NumRefLayers[i]` biểu thị số lượng các lớp tham chiếu (các lớp tham chiếu trực tiếp và các lớp tham chiếu gián tiếp) của lớp thứ i , và `RefLayerIdx[i][r]` biểu thị chỉ số lớp của lớp tham chiếu thứ r của lớp thứ i . Biến của `LayerUsedAsRefLayerFlag[j]` biểu thị xem lớp thứ j có được sử dụng làm lớp tham chiếu trực tiếp hay không. Như được mô tả ở trên, lớp cao hơn có thể thực hiện phép dự đoán liên lớp với tham chiếu đến lớp thấp hơn. Ví dụ, nếu lớp hiện thời là lớp thứ i , lớp thứ 0 đến lớp thứ $i-1$ có thể được sử dụng làm các lớp tham chiếu. Điều này biểu thị rằng, khi ảnh hiện thời của lớp có chỉ số lớp là i được mã hóa, thì ảnh được bao gồm trong AU giống như ảnh hiện thời trong số các ảnh của các lớp có chỉ số lớp nằm trong khoảng từ 0 đến $i-1$ có thể được sử dụng làm ảnh tham chiếu liên lớp. Nếu `vps_direct_ref_layer_flag[i][j]` là 1, thì lớp có chỉ số lớp j có thể được sử dụng làm lớp tham chiếu trực tiếp của lớp có chỉ số lớp i , và bằng phép dự đoán liên lớp, ảnh của lớp có chỉ số lớp j có thể được sử dụng làm ảnh tham chiếu liên lớp đối với ảnh của lớp có chỉ số lớp là i . Mặt khác, trường hợp trong đó `vps_direct_ref_layer_flag[i][j]` là 0 có thể biểu thị rằng lớp có chỉ số lớp j không phải là lớp tham chiếu trực tiếp của lớp có chỉ số lớp i . Trong trường hợp này, lớp thứ j không được sử dụng trực tiếp để mã hóa lớp thứ i , nhưng có thể cần thiết để mã hóa lớp thứ i . Ví dụ, nếu lớp thứ k là lớp tham chiếu trực tiếp của lớp thứ i (`vps_direct_ref_layer_flag[i][k]` là 1), và lớp thứ j là lớp tham chiếu trực tiếp của lớp thứ k (`vps_direct_ref_layer_flag[k][j]` là 1), thì lớp thứ j không phải là lớp tham chiếu trực tiếp của lớp thứ i , nhưng có thể cần thiết để mã hóa lớp thứ i . Lớp này có thể được gọi là lớp tham chiếu gián tiếp.

Trong cú pháp for kép thứ nhất, giá trị của `dependencyFlag[i][j]` có thể được tạo cấu hình trong khi tăng chỉ số i thêm 1 trong khoảng từ 0 đến `vps_max_layers_minus1`, và tăng chỉ số j thêm 1 trong khoảng từ 0 đến `vps_max_layer_minus1`. Trường hợp trong đó `dependencyFlag[i][j]` là 1 biểu thị rằng lớp thứ i phụ thuộc vào lớp thứ j , và biểu thị rằng

lớp thứ j có thể được sử dụng làm lớp tham chiếu của lớp thứ i . Mặt khác, trường hợp trong đó $\text{dependencyFlag}[i][j]$ là 0 biểu thị rằng lớp thứ i không phụ thuộc vào lớp thứ j , và biểu thị rằng lớp thứ j có thể không được sử dụng làm lớp tham chiếu của lớp thứ i . Đầu tiên, nếu lớp thứ j là lớp tham chiếu trực tiếp của lớp thứ i (nếu $\text{vps_direct_ref_layer_flag}[i][j]$ là 1), $\text{dependencyFlag}[i][j]$ có thể được tạo cấu hình là 1, và nếu lớp thứ j không phải là lớp tham chiếu trực tiếp của lớp thứ i (nếu $\text{vps_direct_ref_layer_flag}[i][j]$ là 0), $\text{dependencyFlag}[i][j]$ có thể được tạo cấu hình là 0. Ngoài ra, trong các câu lệnh “for” (chỉ số k tăng thêm 1 trong khoảng từ 0 đến $i-1$) đối với chỉ số k , nếu lớp thứ k là lớp tham chiếu trực tiếp của lớp thứ i ($\text{vps_direct_ref_layer_flag}[i][k]$ là 1), và lớp thứ j là lớp tham chiếu của lớp thứ k (nếu $\text{dependencyFlag}[k][j]$ là 1), $\text{dependencyFlag}[i][j]$ có thể được tạo cấu hình là 1. Ở cuối câu lệnh “for” đối với chỉ số i , biến của $\text{LayerUsedAsRefLayerFlag}[i]$, mà biểu thị xem lớp thứ i có được sử dụng làm lớp tham chiếu trực tiếp hay không, có thể được khởi tạo thành 0. Kết quả là, dựa trên $\text{vps_direct_ref_layer_flag}[i][j]$, nếu lớp thứ j là lớp tham chiếu của lớp thứ i , $\text{dependencyFlag}[i][j]$ có thể bằng 1, và nếu lớp thứ j không phải là lớp tham chiếu của lớp thứ i , $\text{dependencyFlag}[i][j]$ có thể bằng 0.

Sau đó, trong cú pháp for kép đối với các chỉ số i và j , thông tin liên quan đến chỉ số lớp tham chiếu có thể được tạo cấu hình trong khi tăng chỉ số i thêm 1 trong khoảng từ 0 đến $\text{vps_max_layers_minus1}$ và tăng chỉ số j thêm 1 trong khoảng từ 0 đến $\text{vps_max_layers_minus1}$. Trong câu lệnh “for” đối với chỉ số j , chỉ số d liên quan đến lớp tham chiếu trực tiếp và chỉ số r liên quan đến lớp tham chiếu có thể được khởi tạo thành 0. Trường hợp trong đó $\text{vps_direct_ref_layer_flag}[i][j]$ là 1 biểu thị rằng lớp thứ j là lớp tham chiếu trực tiếp của lớp thứ i , và do đó biến của $\text{DirectRefLayerIdx}[i][d]$ được tạo cấu hình thành chỉ số j , và chỉ số d có thể được tăng thêm 1. Biến của $\text{LayerUsedAsRefLayerFlag}[j]$ biểu thị xem lớp thứ j có được sử dụng làm lớp tham chiếu trực tiếp hay không có thể được tạo cấu hình là 1. Ngoài ra, nếu lớp thứ j là lớp tham chiếu trực tiếp hoặc gián tiếp của lớp thứ i (nếu $\text{dependencyFlag}[i][j]$ là 1), thì $\text{RefLayerIdx}[i][r]$ được tạo cấu hình thành chỉ số j , và chỉ số r có thể được tăng thêm 1. Sau khi câu lệnh “for” đối với chỉ số j kết thúc, các giá trị của các chỉ số d và r có thể biểu thị số lượng các lớp tham chiếu trực tiếp và số lượng các lớp tham chiếu của lớp thứ i , một cách tương ứng. Theo đó, $\text{NumDirectRefLayers}[i]$

và NumRefLayers[i] có thể được tạo cấu hình thành các giá trị d và r, một cách tương ứng.

Fig.17 minh họa các mã giả suy ra các biến liên quan đến bộ lớp đầu ra. Biến của NumOutputLayersInOls[i] biểu thị số lượng các lớp đầu ra của bộ lớp đầu ra thứ i, và biến của NumSubLayersInLayerInOLS[i][j] biểu thị số lượng các lớp con (thời gian) cho lớp thứ j của bộ lớp đầu ra thứ i. Đơn vị NAL tương ứng với lớp con cụ thể có thể được loại bỏ khỏi dòng bit, dựa trên biến của NumSubLayersInLayerInOLS[i][j]. Biến của OutputLayerIdInOls[i][j] biểu thị bộ định danh lớp (nuh_layer_id) của phần đầu đơn vị NAL cho lớp thứ j của bộ lớp đầu ra thứ i, và biến của LayerUsedAsOutputLayerFlag[k] biểu thị xem lớp thứ k có được sử dụng làm lớp đầu ra trong ít nhất một bộ lớp đầu ra hay không.

Trong trường hợp bộ lớp đầu ra thứ 0 (bộ lớp đầu ra có chỉ số i bằng 0), vì chỉ lớp thấp nhất (nghĩa là, lớp có nuh_layer_id bằng vps_layer_id[0]) được bao gồm và lớp thấp nhất là lớp đầu ra, các biến NumOutputLayersInOls[0], OutputLayerIdInOls[0], NumSubLayersInLayerInOLS[0][0] và LayerUsedAsOutputLayerFlag[0] có thể được tạo cấu hình là 1, vps_layer_id[0], vps_max_sub_layers_minus1+1 và 1, một cách tương ứng.

Trong cú pháp “for” đối với chỉ số i, biến của LayerUsedAsOutputLayerFlag[i] biểu thị xem lớp thứ i có được sử dụng làm lớp đầu ra hay không có thể được suy ra trong khi i được tăng thêm 1 trong khoảng từ 1 đến vps_max_layers_minus1. Nếu each_layer_is_an_ols_flag là 1 hoặc ols_mode_idc là 0 hoặc 1 (nếu ols_mode_idc nhỏ hơn 2), vì tất cả các lớp của CVS có thể được sử dụng làm các lớp đầu ra, nên LayerUsedAsOutputLayerFlag[i] có thể được tạo cấu hình là 1. Mặt khác, nếu each_layer_is_an_ols_flag là 0 và ols_mode_idc là 2, vì lớp đầu ra được biểu thị bởi ols_output_layer_flag[[]], nên LayerUsedAsOutputLayerFlag[i] có thể được khởi tạo trước tiên thành 0.

Trong cú pháp "cho" đối với chỉ số i sau, trong khi tăng i thêm 1 trong khoảng từ 1 đến TotalNumOlss(tổng số lượng các bộ lớp đầu ra)-1, các biến có liên quan có thể được suy ra theo phương pháp tạo cấu hình bộ lớp đầu ra. Nghĩa là, NumOutputLayersInOls[i], NumSubLayersInLayerInOLS[i][j], OutputLayerIdInOls[i][j] và

LayerUsedAsOutputLayerFlag[k] có thể được suy ra cho tất cả các bộ lớp đầu ra ngoại trừ bộ lớp đầu ra thứ 0.

Đầu tiên, nếu each_layer_is_an_ols_flag là 1 hoặc ols_mode_idc là 0, số lượng các lớp đầu ra của bộ lớp đầu ra thứ i có thể là một, và lớp đầu ra có thể là lớp đầu ra thứ i. Theo đó, NumOutputLayersInOls[i] có thể được tạo cấu hình là 1, và OutputLayerIdInOls[i][0] có thể được tạo cấu hình thành vps_layer_id[i] mà là bộ định danh lớp của lớp thứ i (lớp đầu ra). Nếu ols_mode_idc là 0, thì bộ lớp đầu ra thứ i có thể bao gồm các lớp i+1, và lớp thứ i (lớp trong đó nuh_layer_id là vps_layer_id[i]) có thể được sử dụng làm lớp đầu ra. Mặt khác, nếu each_layer_is_an_ols_flag là 1, bộ lớp đầu ra thứ i chỉ bao gồm lớp thứ i, và lớp thứ i có thể được sử dụng làm lớp đầu ra. Trong câu lệnh “for” đối với chỉ số j, nếu ols_mode_idc là 0, số lượng lớp con thời gian cho lớp tham chiếu loại trừ lớp đầu ra có thể được tạo cấu hình trong khi tăng j thêm 1 trong khoảng từ 0 đến i-1. Như được mô tả ở trên, bộ định danh thời gian (TemporalId) của ảnh tham chiếu liên lớp mà có thể được sử dụng cho ảnh của lớp cụ thể có thể được giới hạn theo max_tid_il_ref_pics_plus1[i]. Nếu giá trị của max_tid_il_ref_pics_plus1[i] lớn hơn 0, ảnh có TemporalId lớn hơn max_tid_il_ref_pics_plus1[i]-1 có thể không được sử dụng làm ảnh tham chiếu liên lớp khi mã hóa ảnh trong lớp thứ i. Nghĩa là, theo giá trị của max_tid_il_ref_pics_plus1[i], không phải tất cả các lớp con cấu thành lớp tham chiếu cần thiết để mã hóa lớp thứ i, nhưng chỉ các lớp con có TemporalId nhỏ hơn hoặc bằng max_tid_il_ref_pics_plus1[i]-1 có thể cần thiết. Do đó, đối với lớp có chỉ số của lớp đầu ra là i, đơn vị NAL trong đó TemporalId của lớp tham chiếu lớn hơn hoặc bằng max_tid_il_ref_pics_plus1[i] có thể được loại bỏ khỏi dòng bit và có thể không được giải mã. Sau khi kết thúc câu lệnh “for” đối với chỉ số j, lớp thứ i của bộ lớp đầu ra thứ i là lớp đầu ra, và do đó NumSubLayersInLayerInOLS[i][i] có thể được tạo cấu hình thành vps_max_sub_layers_minus1+1.

Sau đó, nếu each_layer_is_an_ols_flag là 0 và ols_mode_idc là 1, thì số lượng các lớp đầu ra của bộ lớp đầu ra thứ i có thể bằng i+1. Ví dụ, nếu CVS đề cập đến VPS bao gồm lớp 0, lớp 1 và lớp 2, bộ lớp đầu ra thứ 0 có thể bao gồm lớp 0, và lớp 0 có thể được sử dụng làm lớp đầu ra. Bộ lớp đầu ra thứ nhất bao gồm lớp 0 và lớp 1, và cả hai lớp 0 và

lớp 1 có thể được sử dụng làm các lớp đầu ra. Bộ lớp đầu ra thứ hai bao gồm lớp 0, lớp 1 và lớp 2, và tất cả các lớp trong số lớp 0, lớp 1 và lớp 2 có thể được sử dụng làm các lớp đầu ra. Theo đó, biến của NumOutputLayersInOls[i] biểu thị số lượng các lớp đầu ra của bộ lớp đầu ra thứ i có thể được tạo cấu hình thành i+1. Trong câu lệnh “for” đối với chỉ số j, trong khi tăng j thêm 1 trong khoảng từ 0 đến NumOutputLayersInOls[i]-1, bộ định danh lớp của lớp đầu ra cấu thành bộ lớp đầu ra và số lượng các lớp con được bao gồm trong lớp này có thể được suy ra. OutputLayerIdInOls[i][j], mà là bộ định danh lớp đầu ra thứ j của bộ lớp đầu ra thứ i có thể được tạo cấu hình thành vps_layer_id[j] mà là bộ định danh lớp của lớp thứ j, tất cả các lớp cấu thành bộ lớp đầu ra là các lớp đầu ra, và do đó NumSubLayersInLayerInOLS[i][j] có thể được tạo cấu hình thành vps_max_sub_layers_minus1+1.

Sau đó, nếu each_layer_is_an_ols_flag là 0 và ols_mode_idc là 2, trong câu lệnh “for” đối với chỉ số j, layerIncludedInOlsFlag[i][j] và NumSubLayersInLayerInOLS[i][j] có thể được khởi tạo trước tiên thành 0 trong khi tăng j thêm 1 trong khoảng từ 0 đến vps_max_layers_minus1. Trong câu lệnh “for” đối với chỉ số k, chỉ số k có thể được tăng thêm 1 trong khoảng từ 0 đến vps_max_layers_minus1, và chỉ số j có thể được khởi tạo thành 0. Theo đó, nếu lớp thứ k là lớp đầu ra của bộ lớp đầu ra thứ i (nếu ols_output_layer_flag[i][k] là 1), layerIncludedInOlsFlag[i][k] và LayerUsedAsOutputLayerFlag[k] có thể được tạo cấu hình là 1. Điều này có thể biểu thị rằng lớp thứ k được bao gồm trong bộ lớp đầu ra thứ i, và lớp thứ k được sử dụng làm lớp đầu ra. OutputLayerIdx[i][j] mà là biến biểu thị chỉ số lớp đầu ra thứ j của bộ lớp đầu ra thứ i có thể được tạo cấu hình thành chỉ số lớp k, và OutputLayerIdInOls[i][j] mà là bộ định danh lớp của lớp đầu ra thứ j của bộ lớp đầu ra thứ i có thể được tạo cấu hình thành vps_layer_id[k]. Sau đó, chỉ số j có thể được tăng thêm 1. Lớp thứ k của bộ lớp đầu ra thứ i là lớp đầu ra, và NumSubLayersInLayerInOLS[i][k] do đó có thể được tạo cấu hình thành vps_max_sub_layers_minus1+1. Sau khi kết thúc câu lệnh “for” đối với chỉ số j và câu lệnh “for” đối với chỉ số k, vì j bằng số lượng các lớp đầu ra được bao gồm trong bộ lớp đầu ra thứ i, nên NumOutputLayersInOls[i] có thể được tạo cấu hình thành j. NumSubLayersInLayerInOLS[i][j] bằng vps_max_sub_layers_minus1+1 nếu lớp thứ j là

lớp đầu ra, và bằng 0 nếu lớp thứ j không phải là lớp đầu ra.

Trong câu lệnh “for” đối với chỉ số j , các giá trị biến đổi với lớp tham chiếu của lớp đầu ra có thể được suy ra trong khi tăng j thêm 1 trong khoảng từ 0 đến $\text{NumOutputLayersInOls}[i]-1$. Biến của idx biểu thị chỉ số lớp của lớp đầu ra thứ j cấu thành bộ lớp đầu ra thứ i , và biến của $\text{NumRefLayers}[\text{idx}]$ biểu thị số lượng các lớp tham chiếu của lớp đầu ra thứ j cấu thành bộ lớp đầu ra thứ i . Biến của $\text{RefLayerIdx}[\text{idx}][k]$ biểu thị chỉ số lớp của lớp tham chiếu thứ k của lớp đầu ra thứ j cấu thành bộ lớp đầu ra thứ i . Trong câu lệnh “for” đối với chỉ số k , có thể được tạo cấu hình rằng, trong khi tăng k thêm 1 trong khoảng từ 0 đến $\text{NumRefLayers}[\text{idx}]-1$, các lớp tham chiếu của lớp đầu ra thứ j cấu thành bộ lớp đầu ra thứ i được bao gồm trong bộ lớp đầu ra thứ i ($\text{layerIncludedInOlsFlag}[i][\text{RefLayerIdx}[\text{idx}][k]]=1$). Nếu số lượng ($\text{NumSubLayersInLayersInOLS}[i][\text{RefLayerIdx}[\text{idx}][k]]$) các lớp con cho lớp tham chiếu nhỏ hơn $\text{max_tid_il_ref_pics_plus1}[\text{idx}]$, thì $\text{NumSubLayersInLayerInOLS}[i][\text{RefLayerIdx}[\text{idx}][k]]$ có thể được tạo cấu hình thành $\text{max_tid_il_ref_pics_plus1}[\text{idx}]$. Nghĩa là, lớp con bao gồm ảnh mà không được sử dụng làm đầu ra cho lớp tham chiếu và không được sử dụng làm ảnh tham chiếu liên lớp theo $\text{max_tid_il_ref_pics_plus1}[\text{idx}]$ có thể được loại bỏ khỏi dòng bit. Trong trường hợp này, phần tử cú pháp $\text{max_tid_il_ref_pics_plus1}[]$ được xác định theo chỉ số lớp mà không phải là bộ định danh lớp, và khi $\text{max_tid_il_ref_pics_plus1}[\text{OutputLayerIdInOls}[i][j]]$ được sử dụng thay vì $\text{max_tid_il_ref_pics_plus1}[\text{idx}]$, lớp con của lớp tham chiếu cần thiết để giải mã lớp đầu ra có thể được loại bỏ khỏi dòng bit. Do đó, thay vì $\text{OutputLayerIdInOls}[i][j]$ biểu thị bộ định danh lớp đầu ra, $\text{max_tid_il_ref_pics_plus1}$ tương ứng với $\text{GeneralLayerIdx}[\text{OutputLayerIdInOls}[i][j]]$ hoặc idx mà là chỉ số lớp đầu ra có thể được sử dụng. $\text{GeneralLayerIdx}[]$ có thể biểu thị chỉ số lớp cho bộ định danh lớp. $\text{GeneralLayerIdx}[]$ có thể được suy ra bởi mã giả sau.

```
for(i=0; i<=vps_max_layers_minus1; i++)
```

```
GeneralLayerIdx[vps_layer_id[i]]=i
```

Nghĩa là, $\text{GeneralLayerIdx}[\text{OutputLayerIdInOls}[i][j]]$ có thể giống như chỉ số lớp

đầu ra (`OutputLayerIdx[i][j]`) tương ứng với bộ định danh lớp đầu ra (`OutputLayerIdInOls[i][j]`).

Fig.18 minh họa các mã giả suy ra số lượng các lớp được bao gồm trong bộ lớp đầu ra và các bộ định danh lớp của các lớp được bao gồm trong bộ lớp đầu ra.

Biến của `NumLayersInOls[i]` biểu thị số lượng các lớp được bao gồm trong bộ lớp đầu ra thứ *i*, và biến của `LayerIdInOls[i][j]` biểu thị bộ định danh lớp (`nuh_layer_id`) của lớp thứ *j* được bao gồm trong bộ lớp đầu ra thứ *i*.

Trong trường hợp bộ lớp đầu ra thứ 0, vì lớp thứ 0 chỉ là lớp được bao gồm trong bộ lớp đầu ra và là lớp đầu ra, `NumLayersInOls[0]` và `LayerIdInOls[0][0]` có thể được tạo cấu hình là 1 và `vps_layer_id[0]`, một cách tương ứng.

`NumLayersInOls[i]` và `LayerIdInOls[i][j]` có thể được suy ra theo phương pháp tạo cấu hình bộ lớp đầu ra, trong khi tăng *i* thêm 1 trong khoảng từ 1 đến `TotalNumOls-1` trong câu lệnh “for” đối với chỉ số *i*.

Đầu tiên, nếu `each_layer_is_an_ols_flag` là 1, bộ lớp đầu ra thứ *i* chỉ bao gồm lớp thứ *i* có bộ định danh lớp bằng `vps_layer_id[i]`, và `NumLayersInOls[i]` và `LayerIdInOls[i][0]` do đó có thể được tạo cấu hình là 1 và `vps_layer_id[i]`, một cách tương ứng.

Sau đó, nếu `each_layer_is_an_ols_flag` là 0 và `ols_mode_idc` là 0 hoặc 1, thì bộ lớp đầu ra thứ *i* bao gồm tổng các lớp *i+1* trong khoảng từ 0 đến lớp thứ *i*, và `NumLayersInOls[i]` do đó có thể được tạo cấu hình thành *i+1*. Từ 0 đến *i* trong chỉ số *j*, `LayerIdInOls[i][j]` có thể được tạo cấu hình thành `vps_layer_id[j]`.

Sau đó, nếu `each_layer_is_an_ols_flag` là 0 và `ols_mode_idc` là 2, trong câu lệnh “for” đối với chỉ số *k*, *k* có thể được tăng thêm 1 trong khoảng từ 0 đến `vps_max_layers_minus1`, và chỉ số *j* có thể được khởi tạo thành 0. Nếu `layerIncludedInOlsFlag[i][k]` được mô tả ở trên trên Fig.17 là 1 (nghĩa là, nếu lớp thứ *k* được bao gồm trong bộ lớp đầu ra thứ *i*), `LayerIdInOls[i][j]` có thể được tạo cấu hình thành `vps_layer_id[k]` và chỉ số *j* có thể được tăng thêm 1. Sau khi kết thúc câu lệnh “for” đối

với chỉ số k, vì j biểu thị số lượng các lớp được bao gồm trong bộ lớp đầu ra thứ i, nên NumLayersInOls[i] có thể bằng j.

Dòng bit mà được nhập vào bộ giải mã có thể bao gồm các đơn vị NAL cho nhiều lớp, và bộ giải mã của thiết bị đầu cuối có thể trích xuất, từ dòng bit, chỉ lớp được bao gồm trong bộ lớp đầu ra cần được giải mã trong số các bộ lớp đầu ra được xác định bởi VPS để giải mã lớp được trích xuất. Quy trình trích xuất dòng bit con có thể nhận các đầu vào gồm inBitstream mà là dòng bit đầu vào, targetOlsIdx mà là chỉ số của bộ lớp đầu ra cần được giải mã và tIdTarget mà là biến biểu thị TemporalId cao nhất của dòng bit cần được trích xuất. OutBitstream mà là dòng bit con của dòng bit đầu vào có thể được kết xuất bởi quy trình trích xuất dòng bit con.

outBitstream mà là dòng bit đầu ra có thể được suy ra theo các thủ tục sau gồm i), ii), iii) và iv).

i) outBitstream có thể được tạo cấu hình là giống như inBitstream mà là dòng bit đầu vào. Nghĩa là, đối với dòng bit đầu ra mà giống như dòng bit đầu vào, các đơn vị NAL không cần thiết có thể được loại bỏ khỏi dòng bit đầu ra theo các thủ tục ii), iii) và iv).

ii) Đối với outBistream, tất cả các đơn vị NAL có TemporalId lớn hơn tIdTarget có thể được loại bỏ.

iii) Đối với outBitsream, tất cả các đơn vị NAL, trong đó kiểu đơn vị NAL (nal_unit_type) không phải là một trong số VPS_NUT, DCI_NUT và EOB_NUT, và bộ định danh lớp (nuh_layer_id) của phần đầu đơn vị NAL không được bao gồm trong danh sách LayerIdInOls [targetOlsIdx], có thể được loại bỏ. Đối với tất cả các j trong đó chỉ số j nằm trong khoảng từ 0 đến NumLayersInOls[targetOlsIdx]-1, nếu nuh_layer_id không bằng LayerIdInOls[targetOlsIdx][j], thì điều này biểu thị rằng đơn vị NAL tương ứng không được bao gồm trong bộ lớp đầu ra hiện thời và do đó có thể được loại bỏ khỏi dòng bit đầu ra. Phần đầu đơn vị NAL có thể bao gồm thông tin về kiểu đơn vị NAL, và nếu kiểu của đơn vị NAL là VPS_NUT, thì đơn vị NAL có thể bao gồm RBSP bộ thông số video. Nếu kiểu của đơn vị NAL là DCI_NUT, thì đơn vị NAL có thể bao gồm RBSP thông tin năng lực bộ giải mã. Nếu kiểu của đơn vị NAL là EOB_NUT, thì đơn vị NAL có

thể bao gồm thông tin liên quan đến đầu cuối của dòng bit. Nghĩa là, nếu kiểu của đơn vị NAL là một trong số VPS_NUT, DCI_NUT và EOB_NUT, thậm chí nếu bộ định danh lớp không được bao gồm trong danh sách LayerIdInOls[targetOlsIdx], thì bộ định danh lớp có thể là thông tin cần thiết để giải mã và do đó có thể được bao gồm trong dòng bit đầu ra.

iv) Đối với outBitstream, tất cả các đơn vị NAL mà tất cả các điều kiện a), b) và c) sau là đúng có thể được loại bỏ.

a) Trường hợp trong đó đơn vị NAL không bao gồm lớp phần chia cho ảnh IRAP (nếu kiểu của đơn vị NAL không phải là IDR_W_RADL, IDR_N_LP cũng như không phải là CRA_NUT)

Lớp phần chia có thể bao gồm phần đầu phần chia và dữ liệu phần chia, và trường hợp trong đó kiểu của đơn vị NAL là một trong số IDR_W_RADL, IDR_N_LP và CRA_NUT, đơn vị NAL có thể bao gồm lớp phần chia cho ảnh IRAP. TemporalID của ảnh IRAP có thể là 0.

b) Trường hợp trong đó nuh_layer_id bằng LayerIdInOls[targetOlsIdx][j] đối với j có giá trị nằm trong khoảng từ 0 đến NumLayersInOls[targetOlsIdx]-1

c) Trường hợp trong đó TemporalId lớn hơn hoặc bằng NumSubLayersInLayerInOLS[targetOlsIdx][GeneralLayerIdx[LayerIdInOls[targetOlsIdx][j]]] đối với j có giá trị nằm trong khoảng từ 0 đến NumLayersInOls[targetOlsIdx]-1

Danh sách phương pháp suy ra NumSubLayersInLayerInOLS[targetOlsIdx][j] được mô tả trên Fig.17 có thể bao gồm số lượng các lớp con cho các lớp được bao gồm trong bộ lớp đầu ra được biểu thị bởi targetOlsIdx và số lượng các lớp con cho các lớp mà không được bao gồm trong bộ lớp đầu ra. Trong NumSubLayersInLayerInOLS[targetOlsIdx][j], chỉ số j có thể thể hiện các chỉ số lớp cho tất cả các lớp cấu thành CVS, thay vì các chỉ số lớp cho các lớp được bao gồm trong bộ lớp đầu ra được biểu thị bởi targetOlsIdx. Do đó, để giới hạn ở các lớp được bao gồm trong bộ lớp đầu ra được biểu thị bởi targetOlsIdx, thì GeneralLayerIdx[LayerIdInOls[targetOlsIdx][j]] có thể được sử dụng thay vì chỉ số j. GeneralLayerIdx[LayerIdInOls[targetOlsIdx][j]] có thể biểu thị chỉ số lớp của lớp thứ j được bao gồm trong bộ lớp đầu ra thứ targetOlsIdx, và kết quả là,

$\text{NumSubLayersInLayerInOLS}[\text{targetOlsIdx}][\text{GeneralLayerIdx}[\text{LayerIdInOls}[\text{targetOlsIdx}][j]]]$ có thể biểu thị số lượng các lớp con cho lớp thứ j được bao gồm trong bộ lớp đầu ra thứ targetOlsIdx .

Điều kiện a) dành cho thời điểm khi $\text{max_tid_il_ref_pics_plus1}[i]$ là 0, và trường hợp trong đó $\text{max_tid_il_ref_pics_plus1}[i]$ là 0 có thể biểu thị rằng chỉ ảnh IRAP được sử dụng làm ảnh tham chiếu liên lớp của lớp thứ i . Thậm chí nếu TemporalId của ảnh IRAP là 0, và các điều kiện b) và c) là đúng, thì đơn vị NAL cho ảnh IRAP có thể được bao gồm trong dòng bit đầu ra và có thể được sử dụng làm ảnh tham chiếu liên lớp.

Kết quả là, trường hợp trong đó các điều kiện a), b) và c) được thỏa mãn biểu thị đơn vị NAL cho lớp con không được sử dụng (không phải đầu ra cũng như không phải được tham chiếu) của lớp thứ j được bao gồm trong bộ lớp đầu ra được biểu thị bởi targetOlsIdx , và đơn vị NAL do đó có thể được loại bỏ khỏi dòng bit đầu ra.

Fig.19 minh họa các mã giả suy ra các biến liên quan đến bộ lớp đầu ra theo phương án khác. Trong phương pháp suy ra $\text{NumSubLayersInLayerInOLS}[i][j]$ được mô tả trên Fig.17, chỉ số j biểu diễn các chỉ số lớp cho tất cả các lớp cấu thành CVS, và cũng có thể bao gồm số lượng các lớp con cho lớp không được bao gồm trong bộ lớp đầu ra thứ i . Do đó, khi tham chiếu $\text{NumSubLayersInLayerInOLS}[i][j]$ trong quy trình trích xuất dòng bit con, thủ tục làm giới hạn ở lớp thứ j được bao gồm trong bộ lớp đầu ra thứ i bằng cách sử dụng $\text{GeneralLayerIdx}[\text{LayerIdInOls}[i][j]]$ là cần thiết. Theo cách khác, $\text{NumSubLayersInLayerInOLS}[i][j]$ có thể chỉ được suy ra cho lớp được bao gồm trong bộ lớp đầu ra. Trên Fig.19, $\text{NumSubLayersInLayerInOLS}[i][j]$ có thể xác định số lượng các lớp con cho lớp thứ j được bao gồm trong bộ lớp đầu ra thứ i . Trên Fig.19, các nội dung khác với các nội dung của Fig.17 sẽ chủ yếu được mô tả, và đối với các nội dung mà không được mô tả trên Fig.19, các nội dung nêu trên trên Fig.17 có thể được tham chiếu.

Nếu $\text{each_layer_is_an_ols_flag}$ là 1, thì bộ lớp đầu ra thứ i chỉ bao gồm lớp thứ i mà có thể là lớp đầu ra duy nhất. Trong trường hợp này, $\text{NumSubLayersInLayerInOLS}[i][0]$ có thể bằng $\text{vps_max_sub_layers_minus1}+1$.

Đối với trường hợp trong đó $\text{each_layer_is_an_ols_flag}$ là 0 và ols_mode_idc là 0

hoặc 1, phương pháp giống như phương pháp suy ra NumSubLayersInLayerInOLS[i][j] được mô tả dựa vào Fig.17 có thể được áp dụng.

Nếu each_layer_is_an_ols_flag là 0 và ols_mode_idc là 2, biến tạm thời của temp[i][j] có thể được sử dụng thay vì NumSubLayersInLayerInOLS[i][j]. Nếu lớp thứ j là lớp đầu ra của bộ lớp đầu ra thứ i, temp[i][j] có thể bằng vps_max_sub_layers_minus1+1, và nếu lớp thứ j không phải là lớp đầu ra của bộ lớp đầu ra thứ i, thì temp[i][j] có thể bằng 0. Đối với tất cả các lớp đầu ra được bao gồm trong bộ lớp đầu ra thứ i, nếu số lượng (temp[i][RefLayerIdx[idx][k]]) các lớp con cho lớp tham chiếu thứ k của lớp đầu ra nhỏ hơn max_tid_il_ref_pics_plus1[idx], thì temp[i][RefLayerIdx[idx][k]] có thể được tạo cấu hình thành max_tid_il_ref_pics_plus1[idx].

Trong câu lệnh “for” đối với chỉ số j, j có thể được tăng thêm 1 trong khoảng từ 0 đến vps_max_layers_minus1, và k có thể được khởi tạo thành 0. Nếu lớp thứ j được bao gồm trong bộ lớp đầu ra thứ i (nếu layerIncludedInOlsFlag[i][j] là 1), thì NumSubLayersInLayerInOLS[i][k] có thể được tạo cấu hình thành temp[i][j] và chỉ số k có thể được tăng thêm 1.

Không giống như phương pháp suy ra được mô tả trên Fig.17, trong NumSubLayersInLayerInOLS[i][j], số lượng các lớp con chỉ được suy ra cho các lớp được bao gồm trong bộ lớp đầu ra thứ i, và chỉ số j biểu thị lớp thứ j được bao gồm trong bộ lớp đầu ra thứ i. Trong trường hợp này, NumSubLayersInLayerInOLS[targetOlsIdx][j] có thể được sử dụng thay vì NumSubLayersInLayerInOLS[targetOlsIdx][GeneralLayerIdx[LayerIdInOls[targetOlsIdx][j]]] trong điều kiện c) của quy trình trích xuất dòng bit con.

Fig.20 minh họa cấu trúc cú pháp bộ thông số video (VPS) theo phương án khác. Như được mô tả ở trên, max_tid_il_ref_pics_plus1[i] mà là phần tử cú pháp làm giới hạn TemporalID của ảnh tham chiếu liên lớp (ILRP) mà có thể được sử dụng cho lớp có chỉ số lớp là i có thể được bao gồm trong RBSP bộ thông số video. Dựa trên phần tử cú pháp, phương pháp tạo cấu hình bộ lớp đầu ra, và mối tương quan tham chiếu giữa các lớp, lớp con không cần thiết (không được sử dụng làm đầu ra và tham chiếu) trong số các lớp được

bao gồm trong bộ lớp đầu ra có thể được loại bỏ khỏi dòng bit.

Trong cấu trúc cú pháp RBSP VPS của Fig.15, đối với lớp có chỉ số lớp i lớn hơn 0, nếu lớp có chỉ số lớp i có thể sử dụng phép dự đoán liên lớp (nếu $\text{vps_independent_layer_flag}[i]$ là 0), $\text{max_tid_ref_present_flag}[i]$ và $\text{max_tid_il_ref_pics_plus1}[i]$ có thể tồn tại trong RBSP bộ thông số video. Nghĩa là, trên Fig.15, $\text{max_tid_ref_present_flag}[i]$ và $\text{max_tid_il_ref_pics_plus1}[i]$ có thể được báo hiệu bất kể phương pháp tạo cấu hình bộ lớp đầu ra.

Trong cấu trúc cú pháp RBSP VPS của Fig.20, các phần tử cú pháp $\text{max_tid_ref_present_flag}[i]$ và $\text{max_tid_il_ref_pics_plus1}[i]$ có thể được báo hiệu khác với phương pháp của Fig.15.

$\text{NumSubLayersInLayerInOLS}[i][j]$ được sử dụng trong quy trình trích xuất dòng bit con có thể được tạo cấu hình dựa trên phương pháp tạo cấu hình bộ lớp đầu ra và mối tương quan tham chiếu giữa các lớp. Nếu lớp thứ j của bộ lớp đầu ra thứ i được sử dụng làm lớp đầu ra, thì $\text{NumSubLayersInLayerInOLS}[i][j]$ có thể được tạo cấu hình thành giá trị lớn hơn $\text{vps_max_sublayers_minus1}+1$ hoặc $\text{vps_max_sublayers_minus1}+1$. Điều này có thể biểu thị rằng, vì lớp tương ứng là lớp đầu ra, tất cả các lớp con trong lớp này là cần thiết. Mặt khác, nếu lớp thứ j của bộ lớp đầu ra thứ i không được sử dụng làm lớp đầu ra và được sử dụng làm lớp tham chiếu, thì $\text{NumSubLayersInLayerInOLS}[i][j]$ có thể được tạo cấu hình dựa trên $\text{max_tid_il_ref_pics_plus1}[]$, mà có thể biểu thị rằng lớp con cụ thể của lớp tương ứng có thể được loại bỏ khỏi dòng bit. Nếu ols_mode_idc là 1, thì bộ lớp đầu ra thứ i bao gồm các lớp $i+1$, và tất cả các lớp cấu thành bộ lớp đầu ra có thể được sử dụng làm các lớp đầu ra. Nghĩa là, nếu ols_mode_idc là 1, thì $\text{NumSubLayersInLayerInOLS}[i][j]$ có thể không được suy ra dựa trên $\text{max_tid_il_ref_pics_plus1}[i]$ nhưng có thể luôn luôn được suy ra dựa trên $\text{vps_max_sublayers_minus1}+1$.

Phần tử cú pháp $\text{max_tid_il_ref_pics_plus1}[i]$ không được sử dụng trong phương pháp tạo cấu hình bộ lớp đầu ra cụ thể, và do đó có thể được báo hiệu theo phương pháp tạo cấu hình bộ lớp đầu ra. Không giống như cấu trúc cú pháp RBSP VPS của Fig.15, $\text{max_tid_ref_present_flag}[i]$ và $\text{max_tid_il_ref_pics_plus1}[i]$ có thể không được bao gồm

ngay tiếp theo phần tử cú pháp `vps_direct_ref_layer_flag[i][j]`. Như được mô tả ở trên, `max_tid_il_ref_pics_plus1[i]` không được sử dụng khi `ols_mode_idc` là 1, sao cho `max_tid_il_ref_pics_plus1[i]` có thể được bao gồm trong RBSP bộ thông số video sau phần tử cú pháp `ols_mode_idc`. Ví dụ, như được minh họa trên Fig.20, `max_tid_il_ref_pics_plus1[i]` có thể được bao gồm trong RBSP bộ thông số video tiếp theo thông tin liên quan đến bộ lớp đầu ra (`ols_output_layer_flag[i][j]`). Trong câu lệnh “for” đối với chỉ số `i`, `i` có thể được tăng thêm 1 trong khoảng từ 1 đến `vps_max_layers_minus1`, và nếu lớp thứ `i` có thể sử dụng phép dự đoán liên lớp (nếu `vps_independent_layer_flag[i]` là 0) và `ols_mode_idc` là 0 hoặc 2 (nếu `ols_mode_idc` không 1), thì `max_tid_ref_present_flag[i]` có thể được bao gồm trong RBSP bộ thông số video. Nếu `max_tid_ref_present_flag[i]` biểu thị rằng `max_tid_il_ref_pics_plus1[i]` tồn tại (nếu `max_tid_ref_present_flag[i]` là 1), thì `max_tid_il_ref_pics_plus1[i]` có thể được bao gồm trong RBSP bộ thông số video.

Fig.21 là sơ đồ minh họa việc sử dụng cờ liên quan đến số lượng các lớp con trong lớp, trong các nội dung của cấu trúc cú pháp VPS.

Dòng bit có thể bao gồm nhiều lớp, và lớp riêng lẻ có thể bao gồm nhiều lớp con. Số lượng các lớp con có thể là giống nhau hoặc có thể là khác nhau đối với mỗi lớp. Nếu số lượng các lớp con là khác nhau đối với mỗi lớp, thì `ptl_max_temporal_id[i]`, `dpb_max_temporal_id[i]` và `hrd_max_tid[i]` trong số các phần tử cú pháp có thể bị ảnh hưởng. `ptl_max_temporal_id[i]` có thể là phần tử cú pháp liên quan đến cấu hình, hạng, và cấp. `dpb_max_temporal_id[i]` có thể là phần tử cú pháp liên quan đến bộ đệm ảnh được giải mã. `hrd_max_tid[i]` có thể là phần tử cú pháp liên quan đến bộ giải mã tham chiếu giả định.

Trên Fig.21, `vps_all_layers_same_num_sublayers_flag` được sử dụng làm phần tử cú pháp biểu thị rằng số lượng các lớp con là giống nhau đối với mỗi lớp. Nếu giá trị của cờ này bằng 1, thì số lượng các lớp con là giống nhau đối với mỗi lớp, và nếu giá trị cờ bằng 0, thì số lượng các lớp con đối với mỗi lớp có thể là giống nhau hoặc khác nhau. Nếu giá trị cờ bằng 1, thì số lượng các lớp con đối với mỗi lớp có thể bằng

`vps_max_sublayers_minus1+1`.

Fig.22 là sơ đồ minh họa các phần tử ngữ nghĩa và cú pháp mà là các phần của các nội dung của cấu trúc cú pháp SPS và biểu thị các lớp con.

Phần tử cú pháp biểu thị số lượng các lớp con của lớp tương ứng có thể là `sps_max_sublayers_minus1`. Giá trị được thu bằng cách cộng 1 với `sps_max_sublayers_minus1` có thể xác định số lượng lớp con thời gian được cho phép tối đa. Giá trị của `sps_max_sublayers_minus1` có thể nằm trong khoảng từ 0 đến `vps_max_sublayers_minus1`.

Nếu cấu trúc cú pháp VPS tồn tại và giá trị của `vps_all_layers_same_num_sublayers_flag` bằng 1, thì giá trị của phần tử cú pháp `sps_max_sublayers_minus1` trong cấu trúc cú pháp SPS tham chiếu VPS nên bằng giá trị của `vps_max_sublayers_minus1`. Điều này có thể là giới hạn rằng, khi được biểu thị rằng tất cả các lớp có số lượng các lớp con giống nhau trong cấu trúc cú pháp VPS, thì các phần tử cú pháp trong cấu trúc cú pháp SPS tham chiếu chỉ báo cũng nên được tuân thủ. Ngược với điều này, nếu số lượng các lớp con được tạo cấu hình/được biểu thị khác nhau đối với mỗi lớp, sự không phù hợp với cấu hình/hạng/cấp, thông số DPB, và thông số HRD có thể xảy ra, làm khó khăn việc giải mã dòng bit.

Phương án khác là, phần tử cú pháp `sps_max_sublayer_minus1` có thể được báo hiệu trong cấu trúc sau.

```
sps_max_sublayers_minus1_present_flag
if(sps_max_sublayers_minus1_present_flag)
    sps_max_sublayer_minus1
```

`sps_max_sublayer_minus1` có thể được báo hiệu dựa trên phần tử cú pháp `sps_max_sublayers_minus1_present_flag`. `sps_max_sublayers_minus1_present_flag` biểu thị xem phần tử cú pháp `sps_max_sublayer_minus1` có tồn tại hay không, và trường hợp trong đó `sps_max_sublayers_minus1_present_flag` là 1 xác định rằng `sps_max_sublayer_minus1` có tồn tại. Mặt khác, trường hợp trong đó

`sps_max_sublayers_minus1_present_flag` là 0 xác định rằng `sps_max_sublayer_minus1` không tồn tại. Nếu VPS được tham chiếu bởi SPS tồn tại (nghĩa là, `sps_video_parameter_set_id` lớn hơn 0) và `vps_all_layers_same_num_sublayers_flag` là 1, thì giới hạn rằng giá trị của `sps_max_sublayers_minus1_present_flag` nên là 0 có thể tồn tại. Mặt khác, nếu VPS được tham chiếu bởi SPS không tồn tại (nghĩa là, nếu `sps_video_parameter_set_id` là 0) hoặc `vps_all_layers_same_num_sublayers_flag` là 0, thì giới hạn rằng giá trị của `sps_max_sublayers_minus1_present_flag` nên là 1 có thể tồn tại. Nếu `sps_max_sublayers_minus1_present_flag` là 0, thì `sps_max_sublayers_minus1` có thể không tồn tại và có thể được suy ra là `vps_max_sublayers_minus1`.

Fig.23 minh họa ví dụ của cấu hình đơn vị ảnh (PU). Đơn vị ảnh là tập hợp các đơn vị NAL và có thể bao gồm một ảnh được mã hóa. Nghĩa là, PU có thể bao gồm đơn vị NAL phần đầu ảnh (PH), đơn vị NAL phi VCL khác với đơn vị NAL phần đầu ảnh, và một ảnh được mã hóa. Phần đầu ảnh là cấu trúc cú pháp bao gồm các thông số có thể áp dụng cho tất cả các phần chia được bao gồm trong một ảnh được mã hóa, và trong một số trường hợp, đơn vị NAL PH có thể không tồn tại trong PU. Trên Fig.23, đơn vị NAL phi VCL, chẳng hạn như đơn vị NAL DCI, đơn vị NAL VPS, đơn vị NAL SPS và đơn vị NAL PPS, không được minh họa. Tuy nhiên, nếu đơn vị NAL phi VCL này tồn tại trong PU, thì đơn vị NAL phi VCL này có thể tồn tại trước đơn vị NAL PH và đơn vị NAL VCL thứ nhất của PU.

Fig.23 (a) minh họa cấu hình của đơn vị ảnh cho trường hợp trong đó ảnh được mã hóa bao gồm nhiều phần chia được mã hóa. Ảnh có thể được phân chia thành nhiều phần chia, và mỗi phần chia được mã hóa có thể được bao gồm trong PU ở dạng đơn vị NAL VCL. Một phần chia được mã hóa được đóng gói trong một đơn vị NAL VCL, và RBSP được bao gồm trong đơn vị NAL VCL tương ứng là `slice_layer_rbsp()` mà là dữ liệu cho lớp phần chia. Dựa trên thông tin (`nal_unit_type`) biểu thị kiểu đơn vị NAL được bao gồm trong phần đầu của đơn vị NAL VCL, kiểu ảnh bao gồm phần chia được mã hóa của đơn vị NAL VCL có thể được xác định. Ví dụ, nếu `nal_unit_type` là `CRA_NUT`, thì điều này có thể biểu thị rằng đơn vị NAL VCL bao gồm phần chia được mã hóa của ảnh truy cập ngẫu nhiên sạch (clean random access, CRA) mà là một trong số các ảnh IRAP. Nếu PU

bao gồm hai hoặc nhiều hơn hai đơn vị NAL VCL, đơn vị NAL PH có thể tồn tại trước đơn vị NAL VCL thứ nhất của PU. Các thông số được áp dụng cho một ảnh có thể được báo hiệu bằng cách được bao gồm trong phần đầu ảnh, thay vì được bao gồm trong phần đầu phần chia và được báo hiệu một cách lặp lại. Điều này có thể làm giảm tổng phí báo hiệu thông số. Đơn vị NAL PH là đơn vị NAL bao gồm phần đầu ảnh, và `nal_unit_type` mà là thông tin biểu thị kiểu đơn vị NAL của phần đầu đơn vị NAL là giống như `PH_NUT`. RBSP được bao gồm trong đơn vị NAL PH là `picture_header_rbsp()` mà là RBSP cho phần đầu ảnh.

Fig.23 (b) minh họa cấu hình của đơn vị ảnh cho trường hợp trong đó ảnh được mã hóa bao gồm một phần chia được mã hóa.

Như được mô tả ở trên, vì phần đầu ảnh bao gồm các thông số có thể áp dụng cho tất cả các phần chia nằm trong một ảnh, phần đầu ảnh duy nhất cần thiết đối với mỗi ảnh được mã hóa. Đơn vị NAL yêu cầu tổng phí báo hiệu bổ sung theo định dạng tập tin phương tiện và giao thức truyền dẫn, mà biểu thị rằng, khi số lượng các đơn vị NAL cấu thành dòng bit tăng, thì tổng phí báo hiệu cho các đơn vị NAL tăng. Cụ thể là, tổng phí báo hiệu liên quan đến các đơn vị NAL này có thể nặng nề trong các ứng dụng tốc độ bit thấp. Để thực hiện điều này, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, đơn vị NAL PH bao gồm phần đầu ảnh có thể không tồn tại trong PU. Thay vì sử dụng đơn vị NAL duy nhất bao gồm phần đầu ảnh, phần đầu ảnh có thể được bao gồm trong phần đầu phần chia. Nghĩa là, phần đầu ảnh có thể được bao gồm trong RBSP lớp phần chia (`slice_layer_rbsp()`) của đơn vị NAL VCL.

Như được minh họa trên Fig. 23(a) và Fig.23(b), khi đơn vị NAL PH tồn tại trong PU, ảnh được mã hóa bao gồm hai hoặc nhiều hơn hai phần chia được mã hóa (nghĩa là, bao gồm hai hoặc nhiều hơn hai đơn vị NAL VCL), và đơn vị NAL phần đầu ảnh (PH) có thể tồn tại trước đơn vị NAL VCL thứ nhất của đơn vị ảnh (PU). Đơn vị ảnh PU có thể bao gồm đơn vị ảnh hiện thời. Đơn vị NAL PH bao gồm phần đầu ảnh RBSP, và nhiều phần chia cấu thành ảnh có thể tham chiếu phần đầu ảnh. Mặt khác, nếu không có đơn vị NAL PH nào tồn tại trong PU, ảnh được mã hóa chỉ bao gồm một phần chia được mã hóa (nghĩa

là, bao gồm một đơn vị NAL VCL). Trong trường hợp này, phần đầu ảnh có thể được bao gồm trong phần đầu phân chia thay vì đơn vị NAL PH, và một phần chia cấu thành ảnh có thể tham chiếu phần đầu ảnh được bao gồm trong phần đầu phân chia.

Fig.24 minh họa bộ thông số. Như được mô tả trên Fig.14, dòng bit video có thể bao gồm các bộ thông số được sử dụng để mã hóa và giải mã. Đơn vị NAL bao gồm bộ thông số này là đơn vị NAL phi VCL, và bộ thông số được bao gồm trong đơn vị NAL có thể được biểu thị dựa trên `nal_unit_type` của phần đầu đơn vị NAL. Nếu `nal_unit_type` là `DCI_NUT`, thì đơn vị NAL thông tin năng lực giải mã (DCI) được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp `decoding_capability_information_rbsp()`. Nếu `nal_unit_type` là `VPS_NUT`, thì đơn vị NAL bộ thông số video (VPS) được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp của `video_parameter_set_rbsp()`. Nếu `nal_unit_type` là `SPS_NUT`, thì đơn vị NAL bộ thông số trình tự (SPS) được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp `seq_parameter_set_rbsp()`. Nếu `nal_unit_type` là `PPS_NUT`, thì đơn vị NAL bộ thông số ảnh (PPS) được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp `pic_parameter_set_rbsp()`. Nếu `nal_unit_type` là `PREFIX_APS_NUT` hoặc `SUFFIX_APS_NUT`, thì đơn vị NAL bộ thông số thích ứng (APS) được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp `adaptation_parameter_set_rbsp()`. Nếu `nal_unit_type` là `PH_NUT`, thì đơn vị NAL PH được biểu thị, và đơn vị NAL bao gồm cấu trúc cú pháp `picture_header_rbsp()`.

Đầu tiên, đơn vị NAL SPS bao gồm `seq_parameter_set_rbsp()`, và RBSP SPS có thể bao gồm các thông số có thể áp dụng cho CLVS. Nghĩa là, CLVS có thể tham chiếu SPS cụ thể, và RBSP SPS có thể bao gồm bộ định danh để tham chiếu. `sps_seq_parameter_set_id` biểu thị bộ định danh SPS được sử dụng để tham chiếu bởi phần tử cú pháp khác. SPS có thể tham chiếu VPS, và nếu `sps_video_parameter_set_id` lớn hơn 0, thì `vps_video_parameter_set_id` mà là bộ định danh của VPS được tham chiếu bởi SPS được xác định. Mặt khác, trường hợp trong đó `sps_video_parameter_set_id` bằng 0 xác định rằng SPS không tham chiếu VPS, và điều này biểu thị rằng VPS không được tham chiếu khi CLVS mà tham chiếu SPS được giải mã. Tất cả `sps_video_parameter_set_id` của các SPS được tham chiếu bởi các CLVS tương ứng được bao gồm trong CVS nên giống

nhau. Mặc dù không được minh họa trên Fig.24, đơn vị NAL VPS bao gồm video_parameter_set_rbsp(), và RBSP VPS bao gồm vps_video_parameter_set_id mà là bộ định danh để tham chiếu. VPS được biểu thị bởi sps_video_parameter_set_id có thể được tham chiếu bởi SPS.

RBSP SPS có thể bao gồm thông tin điều khiển cho phương pháp mã hóa (hoặc công cụ mã hóa) cụ thể. Thông tin điều khiển này có thể là cờ xác định xem phương pháp mã hóa cụ thể có thể được sử dụng (bật hoặc được kích hoạt) hay không được sử dụng (tắt hoặc không được kích hoạt) trong CLVS. Ví dụ, cờ cho phép công cụ mã hóa (sps_coding_tool_enabled_flag) của cấp trình tự có thể được bao gồm trong RBSP SPS. Cờ cho phép công cụ mã hóa (sps_coding_tool_enabled_flag) của cấp trình tự có thể được thu từ SPS. Cờ cho phép công cụ mã hóa (sps_coding_tool_enabled_flag) của cấp trình tự có thể biểu thị xem công cụ mã hóa có được cho phép cho CLVS hiện thời hay không từ SPS được áp dụng cho CLVS hiện thời. Nếu cờ cho phép công cụ mã hóa (sps_coding_tool_enabled_flag) của cấp trình tự là 1, thì điều này xác định rằng phương pháp mã hóa cụ thể có thể được sử dụng cho CLVS tham chiếu SPS. Mặt khác, nếu cờ cho phép công cụ mã hóa (sps_coding_tool_enabled_flag) của cấp trình tự là 0, thì điều này xác định rằng phương pháp mã hóa cụ thể có thể không được sử dụng trong CLVS tham chiếu SPS.

RBSP có cấu trúc theo các đơn vị của các byte, và do đó có thể bao gồm bit biểu thị đầu cuối của dữ liệu (thông tin) được bao gồm trong RBSP và bit bổ sung cho cấu hình theo các đơn vị của các byte. rbsp_trailing_bits() có thể bao gồm rbsp_stop_one_bit (bit “1”) và rbsp_alignment_zero_bit (bit “0”). rbsp_stop_one_bit luôn luôn là 1, và có thể được thêm vào tiếp theo dữ liệu cuối cùng được bao gồm trong RBSP. Ngoài ra, bit 0 (rbsp_alignment_zero_bit) có thể được thêm vào cho đến khi dòng bit trở thành đơn vị byte. Nghĩa là, bộ giải mã có thể quét dòng bit theo các đơn vị của các byte theo cách ngược lại để tìm ra vị trí tương ứng với giá trị bit là 1 mà đến trước tiên, và điều này có thể biểu thị vị trí tiếp theo đầu cuối của dữ liệu được bao gồm thực tế trong RBSP. Ví dụ, nếu dòng bit là 10011010, RBSP dữ liệu thực tế tương ứng với 100110.

Tiếp theo, đơn vị NAL PPS bao gồm `pic_parameter_set_rbsp()`, và các thông số có thể áp dụng cho ảnh được mã hóa có thể được bao gồm trong RBSP PPS. Nghĩa là, ảnh được mã hóa có thể tham chiếu PPS cụ thể, và RBSP PPS có thể bao gồm bộ định danh để tham chiếu. `pps_pic_parameter_set_id` biểu thị bộ định danh PPS được sử dụng để tham chiếu bởi phần tử cú pháp khác. PPS có thể tham chiếu SPS, và `pps_seq_parameter_set_id` xác định `sps_seq_parameter_set_id` mà là bộ định danh của SPS được tham chiếu bởi PPS. Tất cả `pps_seq_parameter_set_id` của các PPS được tham chiếu bởi các ảnh được mã hóa tương ứng được bao gồm trong CLVS nên giống nhau. Tiếp theo dữ liệu cuối cùng được bao gồm trong RBSP PPS, `rbsp_trailing_bits()` có thể được thêm vào.

Tiếp theo, đơn vị NAL PH bao gồm `picture_header_rbsp()`, và RBSP PH có thể bao gồm các thông số có thể áp dụng cho tất cả các phần chia của một ảnh được mã hóa. `picture_header_rbsp()` có thể bao gồm `rbsp_trailing_bits()` và `picture_header_structure()` (tham chiếu đến dòng 2430 của Fig.24) bao gồm dữ liệu phần đầu ảnh thực tế. Như được mô tả ở trên, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, thì đơn vị NAL PH có thể không tồn tại trong PU, và như được thể hiện ở dòng 2420 của Fig.24, `picture_header_structure()` bao gồm thông số phần đầu ảnh thực tế có thể được bao gồm trong phần đầu phần chia.

`picture_header_structure()` có thể bao gồm thông tin (hoặc các thông số) chung có thể áp dụng cho tất cả các phần chia của ảnh được mã hóa. Vì phần đầu ảnh có thể tham chiếu PPS, `ph_pic_parameter_set_id` xác định `pps_pic_parameter_set_id` của PPS được tham chiếu bởi phần đầu ảnh có thể được bao gồm trong phần đầu ảnh.

Nếu phương pháp mã hóa cụ thể có thể được sử dụng trong CLVS (nếu `sps_coding_tool_enabled_flag` là 1), thông tin để điều khiển của phương pháp mã hóa cụ thể ở cấp ảnh có thể được bao gồm trong phần đầu ảnh. Thông tin điều khiển này có thể là cờ xác định xem phương pháp mã hóa cụ thể có thể được sử dụng (bật hoặc được kích hoạt) hay không được sử dụng (tắt hoặc không được kích hoạt) trong phần chia được mã hóa của ảnh được mã hóa.

Ví dụ, tham chiếu đến dòng 2440 của Fig.24, cờ cho phép công cụ mã hóa cấp ảnh

(`ph_coding_tool_enabled_flag`) có thể được bao gồm trong phần đầu ảnh. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) có thể biểu thị xem công cụ mã hóa có được cho phép cho ảnh hay không. Cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) có thể bao gồm cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) hoặc cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`).

Nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) là 1, thì điều này xác định rằng phương pháp mã hóa cụ thể có thể được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, phương pháp mã hóa cụ thể được xác định rằng có thể được sử dụng cho ảnh được mã hóa. Mặt khác, nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) là 0, thì điều này xác định rằng phương pháp mã hóa cụ thể có thể không được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, phương pháp mã hóa cụ thể được xác định rằng có thể không được sử dụng cho ảnh được mã hóa. Nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) không tồn tại, thì 0 có thể được suy ra. Theo đó, nếu phương pháp mã hóa cụ thể có thể được sử dụng trong CLVSL (nếu `sps_coding_tool_enabled_flag` là 1), thì việc liệu phương pháp mã hóa có thể được sử dụng ở cấp ảnh hay không có thể được điều khiển theo cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`).

Phương pháp mã hóa (hoặc công cụ mã hóa) cụ thể nêu trên có thể bao gồm việc sử dụng ít nhất một trong số ánh xạ độ sáng với sự thay đổi tỷ lệ sắc độ (LMCS) hoặc danh sách thay đổi tỷ lệ rõ ràng. Việc sử dụng LMCS và việc sử dụng danh sách thay đổi tỷ lệ rõ ràng sẽ được mô tả chi tiết hơn dựa vào Fig.26.

Tiếp theo, đơn vị NAL lớp phân chia tương ứng với đơn vị NAL VCL bao gồm `slice_layer_rbsp()` mà là dữ liệu cho phần chia được mã hóa. `slice_layer_rbsp()` có thể bao gồm `slice_header()`, `slice_data()` và `rbp_slice_trailing_bits()`. `slice_header()` bao gồm thông số được áp dụng cho phần chia, và `slice_data()` bao gồm tất cả các đơn vị cây mã

hóa được bao gồm trong phần chia. `rbp_slice_trailing_bits()` là chuỗi bit bổ sung cho RBSP lớp phần chia và có thể bao gồm `rbp_trailing_bits()` và byte bổ sung.

Như được mô tả ở trên, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, thì phần đầu ảnh có thể được bao gồm trong phần đầu phần chia. `slice_header()` có thể bao gồm cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) mà là thông tin biểu thị xem phần đầu ảnh có tồn tại trong phần đầu phần chia hay không. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ dòng bit cấp phần chia, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không. Cụ thể hơn là, thiết bị xử lý tín hiệu video có thể thu cờ phần đầu ảnh trong phần đầu phần chia từ phần đầu phần chia (`slice_header()`) của phần chia hiện thời được bao gồm trong ảnh hiện thời.

Trường hợp trong đó cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 1 xác định rằng cấu trúc cú pháp PH `picture_header_structure()` tồn tại trong phần đầu phần chia. Mặt khác, trường hợp trong đó cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0 xác định rằng cấu trúc cú pháp PH không tồn tại trong phần đầu phần chia. Các giá trị của các cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) của tất cả các phần chia được mã hóa của CLVS nên giống nhau. Ví dụ, các giá trị của các cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) của tất cả các phần chia được mã hóa của CLVS hiện thời có thể là giống nhau.

Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 1, đơn vị NAL PH không nên tồn tại trong CLVS. Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, thì đơn vị lớp trừ tượng mạng (NAL), trong đó thông tin loại đơn vị NAL là PH_NUT, có thể không tồn tại trong trình tự video lớp được mã hóa (CLVS), mà có thể là giống như được thể hiện trên Fig.23B. Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 1 như được thể hiện ở dòng 2420 của Fig.24 hoặc

dòng 2520 của Fig.25, thì `picture_header_structure()` có thể được bao gồm trong phần đầu phần chia. Ví dụ, nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, đơn vị lớp trừu tượng mạng (NAL) của phần đầu ảnh bao gồm cấu trúc cú pháp phần đầu ảnh có thể không tồn tại trong trình tự video lớp được mã hóa (CLVS) hiện thời.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) từ cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`) được bao gồm trong phần đầu phần chia (`slice_header()`) để thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit cấp ảnh.

Mặt khác, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0, thì PU hiện thời cần bao gồm đơn vị NAL PH, mà biểu thị rằng ảnh được mã hóa bao gồm nhiều phần chia được mã hóa. Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, thì đơn vị ảnh (PU) hiện thời có thể bao gồm NAL phần đầu ảnh (PH), mà có thể là giống như được thể hiện trên Fig.23(a). Ví dụ, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh không tồn tại trong phần đầu phần chia, thì đơn vị ảnh (PU) hiện thời có thể bao gồm đơn vị NAL phần đầu ảnh.

Như được thể hiện ở dòng 2430 của Fig.24 hoặc dòng 2530 của Fig.25, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép công cụ mã hóa cấp ảnh từ cấu trúc cú pháp phần đầu ảnh được bao gồm trong đơn vị NAL phần đầu ảnh, để thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit cấp ảnh.

Nếu phương pháp mã hóa cụ thể có thể được sử dụng trong phần chia được mã hóa của ảnh được mã hóa (nếu `ph_coding_tool_enabled_flag` là 1), thì thông tin để điều khiển của phương pháp mã hóa cụ thể ở cấp phần chia có thể được bao gồm trong phần đầu phần chia. Thông tin điều khiển này có thể là cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`). Cờ cho phép công cụ mã hóa cấp phần chia

(slice_coding_tool_enabled_flag) có thể bao gồm cờ cho phép LMCS cấp phần chia (slice_lmcs_enabled_flag) hoặc cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag).

Cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) có thể là cờ xác định xem phương pháp mã hóa cụ thể có thể được sử dụng (bật hoặc được kích hoạt) hay không được sử dụng (tắt hoặc không được kích hoạt) trong phần chia hiện thời. Ví dụ, nếu cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) là 1, thì điều này xác định rằng phương pháp mã hóa cụ thể có thể được sử dụng cho phần chia hiện thời. Mặt khác, nếu cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) là 0, thì điều này xác định rằng phương pháp mã hóa cụ thể không được sử dụng cho phần chia hiện thời. Nếu cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) không tồn tại, thì giá trị 0 có thể được suy ra. Theo đó, nếu phương pháp mã hóa cụ thể có thể được sử dụng cho phần chia được mã hóa của ảnh được mã hóa (nếu ph_coding_tool_enabled_flag là 1), thì việc liệu phương pháp mã hóa có thể được sử dụng ở cấp phần chia hay không có thể được điều khiển theo cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag). Thậm chí nếu cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) là 1, thì phương pháp mã hóa cụ thể có thể không được áp dụng cho tất cả các cấu trúc con được bao gồm trong phần chia tương ứng. Ví dụ, mặc dù không được minh họa trên Fig.24, thông tin để điều khiển của phương pháp mã hóa cụ thể ở cấp phần chia con có thể được sử dụng. Theo cách khác, dựa trên phần tử cú pháp khác, phương pháp mã hóa cụ thể có thể không được sử dụng ở cấp phần chia con.

Phần dư có thể tồn tại trong báo hiệu thông tin điều khiển của phương pháp mã hóa cụ thể ở mỗi cấp được mô tả trên Fig.24. Ví dụ, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa và phần đầu ảnh được bao gồm trong phần đầu phần chia (nếu picture_header_in_slice_header_flag là 1), khi cờ cho phép công cụ mã hóa cấp ảnh (ph_coding_tool_enabled_flag) là 1, thì cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) có thể tồn tại trong phần đầu phần chia. Khi được biểu thị rằng công cụ mã hóa cụ thể có thể được sử dụng cho phần chia được mã hóa của ảnh

được mã hóa (nếu `ph_coding_tool_enabled_flag` là 1), vì ảnh được mã hóa chỉ bao gồm một phần chia được mã hóa, nên có thể thấy rằng phương pháp mã hóa cụ thể có thể chỉ được sử dụng cho một phần chia được mã hóa của ảnh được mã hóa. Do đó, trong trường hợp này, không cần báo hiệu phần tử cú pháp (`slice_coding_tool_enabled_flag`) mà xác định xem phương pháp mã hóa cụ thể có được sử dụng ở cấp phần chia hay không, và việc liệu có sử dụng phương pháp mã hóa cụ thể ở cấp phần chia hay không có thể được xác định dựa trên cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`).

Fig.25 minh họa bộ thông số. Dòng 2510 trên Fig.25 khác với dòng 2410 trên Fig.24. Các phần còn lại được mô tả theo cách tương tự trên Fig.24 và Fig.25 đã được mô tả dựa vào Fig.24, và do đó các phần mô tả trùng lặp sẽ được lược bỏ.

Như được mô tả trên Fig.24, đối với phương pháp báo hiệu thông tin điều khiển (`slice_coding_tool_enabled_flag`) của phương pháp mã hóa cụ thể ở cấp phần chia, bất kể số lượng các phần chia được mã hóa được bao gồm trong ảnh được mã hóa, nếu phương pháp mã hóa cụ thể có thể được sử dụng trong các phần chia được mã hóa của ảnh được mã hóa (nếu `ph_coding_tool_enabled_flag` là 1), thì cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) mà xác định xem phương pháp mã hóa cụ thể có thể được sử dụng cho phần chia hiện thời hay không có thể được bao gồm trong RBSP phần đầu phần chia. Nghĩa là, nếu phương pháp mã hóa cụ thể có thể được sử dụng cho các phần chia của ảnh được mã hóa, việc liệu phương pháp mã hóa cụ thể có được sử dụng (bật/tắt hoặc được kích hoạt/không được kích hoạt) đối với mỗi phần chia được mã hóa cấu thành ảnh được mã hóa hay không có thể được xác định theo cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`).

Tuy nhiên, nếu ảnh được mã hóa chỉ bao gồm một phần chia được mã hóa, thì việc liệu phương pháp mã hóa cụ thể có được sử dụng cho phần chia hiện thời hay không có thể được xác định theo cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) là 1, thì điều này có thể biểu thị rằng phương pháp mã hóa cụ thể có thể được sử dụng cho một phần chia được mã hóa cấu thành ảnh được mã hóa. Nếu cờ cho phép công cụ mã hóa cấp ảnh

(`ph_coding_tool_enabled_flag`) là 0, thì điều này có thể biểu thị rằng phương pháp mã hóa cụ thể có thể không được sử dụng cho một phần chia được mã hóa cấu thành ảnh được mã hóa.

Theo đó, nếu phần đầu ảnh tồn tại trong phần đầu phần chia (nếu `picture_header_in_slice_header_flag` là 1), điều này biểu thị rằng không có đơn vị NAL PH nào tồn tại trong PU và ảnh được mã hóa bao gồm một phần chia được mã hóa, và cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) không được bao gồm trong RBSP phần đầu phần chia. Giá trị của cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) có thể được suy ra là giá trị của cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`).

Dựa vào dòng 2510 của Fig.25, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) biểu thị sự cho phép công cụ mã hóa, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`), mà biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không, từ phần đầu phần chia (`slice_header()`). Ví dụ, cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) của Fig.25, mà xác định xem phương pháp mã hóa cụ thể có được sử dụng (bật/tắt hoặc được kích hoạt/không được kích hoạt) hay không ở cấp phần chia, có thể được bao gồm trong RBSP phần đầu phần chia nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) là 1 và cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0. Nếu cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) không tồn tại trong RBSP phần đầu phần chia, thì giá trị của cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) có thể được suy ra là giá trị của cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Điều này biểu thị rằng, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, thì việc liệu phương pháp mã hóa cụ thể có thể được sử dụng cho phần chia hiện thời hay không được xác định theo cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Nếu cờ cho phép công cụ mã hóa cấp ảnh

(`ph_coding_tool_enabled_flag`) là 1 và ảnh được mã hóa bao gồm hai hoặc nhiều hơn hai phần chia được mã hóa, thì điều này biểu thị rằng cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) có thể được bao gồm trong RBSP phần đầu phần chia, và việc liệu phương pháp mã hóa cụ thể có được sử dụng hay không có thể được xác định cho mỗi phần chia được mã hóa của ảnh được mã hóa.

Nếu cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) không được thu từ phần đầu phần chia (`slice_header()`), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`), dựa trên cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Dựa vào dòng 2510 của Fig.25, có thể có hai trường hợp trong đó cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) không được thu từ dòng bit. Nghĩa là, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia (trường hợp thứ nhất), hoặc nếu cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) biểu thị sự không cho phép công cụ mã hóa (trường hợp thứ hai), thì cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) có thể không được thu từ dòng bit.

Thiết bị xử lý tín hiệu video có thể thực hiện các thủ tục sau để xác định cờ cho phép công cụ mã hóa cấp phần chia, dựa trên cờ cho phép công cụ mã hóa cấp ảnh. Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định giá trị của cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Cụ thể là, trong trường hợp thứ nhất, cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`) có thể biểu thị việc kích hoạt công cụ mã hóa hoặc không kích hoạt công cụ mã hóa. Trong trường hợp này, thiết bị xử lý tín hiệu video có thể xác định cờ cho phép công cụ mã hóa cấp phần chia (`slice_coding_tool_enabled_flag`) là giống như cờ cho phép công cụ mã hóa cấp ảnh (`ph_coding_tool_enabled_flag`). Nghĩa là, nếu cờ cho phép công cụ mã hóa cấp ảnh biểu

thị sự cho phép công cụ mã hóa, thì thiết bị xử lý tín hiệu video có thể xác định cờ cho phép công cụ mã hóa cấp phần chia để biểu thị việc sử dụng công cụ mã hóa. Nếu cờ cho phép công cụ mã hóa cấp ảnh biểu thị sự không cho phép công cụ mã hóa, thì thiết bị xử lý tín hiệu video có thể xác định cờ cho phép công cụ mã hóa cấp phần chia để biểu thị việc không sử dụng công cụ mã hóa.

Nếu cờ phần đầu ảnh trong phần đầu phần chia (picture_header_in_slice_header_flag) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) để biểu thị việc không sử dụng công cụ mã hóa. Dựa vào dòng 2510 của Fig.25, trường hợp trong đó cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) không được thu từ dòng bit mặc dù cờ phần đầu ảnh trong phần đầu phần chia không có cấu trúc cú pháp phần đầu ảnh trong phần đầu phần chia có thể là trường hợp trong đó, như trong trường hợp thứ hai, cờ cho phép công cụ mã hóa cấp ảnh (ph_coding_tool_enabled_flag) biểu thị sự không cho phép công cụ mã hóa. Thiết bị xử lý tín hiệu video có thể xác định giá trị của cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh (ph_coding_tool_enabled_flag). Nghĩa là, thiết bị xử lý tín hiệu video có thể xác định cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) để biểu thị việc không sử dụng công cụ mã hóa. Vì cờ cho phép công cụ mã hóa cấp ảnh (ph_coding_tool_enabled_flag) biểu thị việc không kích hoạt trường hợp thứ hai, việc xác định giá trị của cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh (ph_coding_tool_enabled_flag) có thể có ý nghĩa giống như việc xác định cờ cho phép công cụ mã hóa cấp phần chia (slice_coding_tool_enabled_flag) để biểu thị việc không sử dụng công cụ mã hóa.

Fig.26 minh họa phương pháp báo hiệu xem liệu có sử dụng tỷ lệ độ sáng ánh xạ với sắc độ (LMCS) và danh sách thay đổi tỷ lệ rõ ràng, ở cấp trình tự, ảnh và phần chia hay không.

Như đã được mô tả, phương pháp mã hóa (công cụ mã hóa) cụ thể của Fig.24 đến Fig.25 có thể bao gồm ít nhất một trong số bước sử dụng ánh xạ độ sáng với sự thay đổi tỷ lệ sắc độ (LMCS) hoặc danh sách thay đổi tỷ lệ rõ ràng.

LMCS có thể được áp dụng trước khi thực thi bởi bộ phận lọc, và có thể bao gồm thủ tục (ánh xạ trong vòng) ánh xạ thành phần độ sáng dựa trên mô hình tuyến tính theo mảnh thích ứng và thủ tục (thay đổi tỷ lệ sắc độ) thay đổi tỷ lệ tín hiệu dư của thành phần sắc độ.

Đầu tiên, ánh xạ trong vòng cho thành phần độ sáng có thể cải thiện hiệu quả mã hóa bằng cách điều chỉnh phạm vi động của tín hiệu đầu vào. Ánh xạ trong vòng cho thành phần độ sáng bao gồm ánh xạ xuôi và ánh xạ ngược, và bộ dự đoán được thu sau khi bù chuyển động có thể có phạm vi động được điều chỉnh bởi chức năng ánh xạ xuôi. Khối được khôi phục được thu bằng cách tính tổng của bộ dự đoán được ánh xạ và khối dư, và trước khi bộ phận lọc thực thi, thì ảnh được khôi phục có thể được ánh xạ đến phạm vi động gốc bởi chức năng ánh xạ ngược. Trong trường hợp phép dự đoán nội ảnh, vì các mẫu tham chiếu được sử dụng cho phép dự đoán nội ảnh là các mẫu được ánh xạ, phép ánh xạ xuôi có thể không được thực hiện.

Tiếp theo, trong thủ tục thay đổi tỷ lệ sắc độ, thông số thay đổi tỷ lệ được suy ra dựa trên mẫu độ sáng được khôi phục xung quanh khối hiện thời hoặc khối đơn vị được tạo cấu hình trước (ví dụ, khối đơn vị có kích thước là 64x64) bao gồm một phần của khối hiện thời, và tín hiệu dư của thành phần sắc độ có thể được thay đổi tỷ lệ dựa trên đó.

Thông số liên quan đến thay đổi tỷ lệ sắc độ và chức năng ánh xạ xuôi của LMCS có thể được bao gồm trong bộ thông số thích ứng (APS), và chức năng ánh xạ ngược có thể được suy ra dựa trên chức năng ánh xạ xuôi. Bộ thông số thích ứng có thể bao gồm các hệ số lọc được sử dụng trong ảnh hoặc phân chia, các phần tử của danh sách thay đổi tỷ lệ (thay đổi tỷ lệ đối với các hệ số biến đổi), các chức năng ánh xạ và dạng tương tự. Nghĩa là, bộ thông số thích ứng bao gồm các thông số được sử dụng trong ảnh hoặc phân chia và bộ định danh để tham chiếu, và phần đầu ảnh hoặc phần đầu phân chia có thể bao gồm bộ định danh của APS được tham chiếu. Trường hợp trong đó `nal_unit_type` là

PREFIX_APS_NUT hoặc SUFFIX_APS_NUT biểu thị rằng đơn vị NAL phi VCL tương ứng là đơn vị NAL APS, và `adaptation_parameter_set_rbsp()` có thể được bao gồm.

`Adaptation_parameter_set_rbsp()` mà là RBSP APS có thể bao gồm `adaptation_parameter_set_id` mà là bộ định danh APS được sử dụng để tham chiếu bởi cú pháp khác. Phần tử cú pháp `aps_params_type` để xác định loại của thông số được bao gồm trong APS có thể được bao gồm trong RBSP APS. Nếu `aps_params_type` là ALF_APS, thì APS tương ứng biểu thị rằng hệ số của bộ lọc vòng thích ứng (ALF) được bao gồm, và `adaptation_parameter_set_rbsp()` có thể bao gồm `alf_data()` mà là hệ số ALF. Nếu `aps_params_type` là LMCS_APS, thì APS biểu thị rằng thông số liên quan đến chức năng ánh xạ xuôi của LMCS và thông số liên quan đến thay đổi tỷ lệ sắc độ được bao gồm, và `adaptation_parameter_set_rbsp()` có thể bao gồm `lmcs_data()` mà là thông số LMCS. Nếu `aps_params_type` là SCALING_APS, thì APS biểu thị rằng danh sách thay đổi tỷ lệ được sử dụng trong quy trình thay đổi tỷ lệ (thủ tục lượng tử hóa ngược) được bao gồm, và `adaptation_parameter_set_rbsp()` bao gồm `scaling_list_data()` mà là danh sách thay đổi tỷ lệ.

Việc liệu có sử dụng LMCS hay không có thể được báo hiệu ở cấp trình tự, ảnh và phần chia. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép LMCS cấp trình tự biểu thị xem ánh xạ độ sáng với sự thay đổi tỷ lệ sắc độ (LMCS) cho CLVS hiện thời được cho phép. Để báo hiệu xem LMCS có được sử dụng ở cấp trình tự hay không, `seq_parameter_set_rbsp()` có thể bao gồm cờ cho phép LMCS cấp trình tự (`sps_lmcs_enabled_flag`). Trường hợp trong đó cờ cho phép LMCS cấp trình tự (`sps_lmcs_enabled_flag`) là 1 xác định rằng LMCS có thể được sử dụng trong CLVS tham chiếu SPS, và trường hợp trong đó cờ cho phép LMCS cấp trình tự (`sps_lmcs_enabled_flag`) là 0 xác định rằng LMCS không được sử dụng trong CLVS tham chiếu SPS.

Tiếp theo, tham chiếu đến dòng 2631 của Fig.26, nếu cờ cho phép LMCS cấp trình tự (`sps_lmcs_enabled_flag`) biểu thị sự cho phép LMCS, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`),

cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị xem LMCS cho ảnh hiện thời có được cho phép hay không. Hoạt động của dòng 2631 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2440 trên Fig.24, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Theo cách khác, hoạt động của dòng 2631 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2540 trên Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Cụ thể là, để báo hiệu xem LMCS có được sử dụng ở cấp ảnh hay không, cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`) có thể bao gồm cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) nếu cờ cho phép LMCS cấp trình tự (`sps_lmcs_enabled_flag`) là 1. Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 1, thì điều này xác định rằng LMCS có thể được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, LMCS được xác định rằng có thể được sử dụng cho ảnh được mã hóa. Mặt khác, nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 0, thì điều này xác định rằng LMCS không được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, LMCS được xác định rằng không được sử dụng cho ảnh được mã hóa. Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) không tồn tại, thì giá trị của cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) được suy ra là 0.

Dựa vào dòng 2632 của Fig.26, nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự cho phép LMCS, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu bộ định danh (`ph_lmcs_aps_id`) cho APS LMCS từ cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`). Ví dụ, nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 1, thì phần đầu ảnh có thể bao gồm bộ định danh cho APS LMCS được tham chiếu. Bộ định danh (`ph_lmcs_aps_id`) cho APS LMCS xác định `adaptation_parameter_set_id` mà là bộ định danh APS của APS LMCS được tham chiếu trong phần chia được mã hóa của ảnh được mã hóa. Hoạt động của dòng 2632 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2440 trên Fig.24, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Ngoài ra, hoạt động của dòng 2632 của Fig.26 có thể được bao gồm trong hoạt động của Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit.

Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự cho phép

LMCS, và định dạng màu của ảnh hiện thời là định dạng bao gồm thành phần sắc độ, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ cấu trúc cú pháp phần đầu ảnh, cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) biểu thị xem sự thay đổi tỷ lệ tín hiệu dư sắc độ có được cho phép cho ảnh hiện thời hay không. Ví dụ, nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 1, và định dạng màu của hình ảnh là định dạng bao gồm thành phần sắc độ (`ChromaArrayType` không phải là 0), thì phần đầu ảnh có thể bao gồm thông tin về việc liệu sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS có được sử dụng hay không. Nếu cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) là 1, thì điều này xác định rằng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS có thể được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS được xác định rằng có thể được sử dụng cho ảnh được mã hóa. Mặt khác, nếu cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) là 0, thì điều này xác định rằng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS có thể không được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS được xác định rằng không được sử dụng cho ảnh được mã hóa. Nếu cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) không tồn tại, thì giá trị của cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) có thể được suy ra là 0.

Tiếp theo, để báo hiệu xem LMCS có được sử dụng ở cấp phần chia hay không, nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 1, và cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0, `slice_header()` có thể bao gồm cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`). Nghĩa là, tham chiếu đến dòng 2611 của Fig.26, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, và cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự cho phép LMCS, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`), mà biểu thị xem LMCS có được sử dụng cho phần chia hiện thời hay không, từ phần đầu phần chia (`slice_header()`). Hoạt động của dòng 2611 của Fig.26 có thể được bao gồm trong hoạt

động của dòng 2511 trên Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp phần chia từ dòng bit.

Thiết bị xử lý tín hiệu video có thể thực hiện việc giải mã phần chia hiện thời, dựa trên cờ cho phép LMCS cấp phần chia và bộ định danh cho APS LMCS. Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 1, điều này xác định rằng LMCS có thể được sử dụng cho phần chia hiện thời. Nghĩa là, LMCS có thể được sử dụng cho phần chia hiện thời với tham chiếu đến thông số của APS LMCS được biểu thị bởi `ph_lmcs_aps_id`. Mặt khác, nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 0, thì điều này xác định rằng LMCS không được sử dụng cho phần chia hiện thời.

Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) không tồn tại, thì giá trị của cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) có thể được suy ra là giá trị của cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Nghĩa là, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, thì việc liệu LMCS có thể được sử dụng cho phần chia hiện thời hay không có thể được xác định theo cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 1, thì phần chia hiện thời (chỉ phần chia được mã hóa của ảnh được mã hóa) được bao gồm trong ảnh được mã hóa có thể sử dụng LMCS. Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) là 0, thì phần chia hiện thời được bao gồm trong ảnh được mã hóa không thể sử dụng LMCS.

Cụ thể hơn là, tham chiếu đến dòng 2610 của Fig.26, có thể có hai trường hợp trong đó cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) không được thu từ dòng bit. Nghĩa là, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia (trường hợp thứ nhất), hoặc nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự không cho phép LMCS của ảnh hiện thời (trường hợp thứ hai), thì cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) có thể không được thu từ dòng bit.

Thiết bị xử lý tín hiệu video có thể thực hiện các thủ tục sau để xác định cờ cho phép

LMCS cấp phần chia (`slice_lmcs_enabled_flag`), dựa trên cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là giống như cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Cụ thể là, trong trường hợp thứ nhất, cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) có thể biểu thị việc kích hoạt LMCS hoặc không kích hoạt LMCS của ảnh hiện thời. Thiết bị xử lý tín hiệu video có thể xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là giống như cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự cho phép LMCS, thì thiết bị xử lý tín hiệu video có thể xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) để biểu thị việc sử dụng LMCS. Nếu cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự không cho phép LMCS, thì thiết bị xử lý tín hiệu video có thể xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) để biểu thị việc không sử dụng LMCS.

Cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) có thể là công cụ mã hóa được áp dụng cho phần chia được bao gồm trong ảnh hiện thời.

Nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) để biểu thị việc không sử dụng LMCS. Dựa vào dòng 2510 của Fig.25, trường hợp trong đó cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) không được thu từ dòng bit mặc dù cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) không có cấu trúc cú pháp phần đầu ảnh trong phần đầu phần chia có thể là trường hợp trong đó, như trong trường hợp thứ hai, cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị sự không cho phép LMCS. Thiết bị xử lý tín hiệu video có thể xác định giá trị của cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là giống như giá trị của cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`). Nghĩa là, thiết bị xử lý tín hiệu video có thể

xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) để biểu thị việc không sử dụng LMCS cho phần chia hiện thời được bao gồm trong ảnh hiện thời. Vì cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) biểu thị việc không kích hoạt, việc xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là giống như cờ cho phép LMCS cấp ảnh (`ph_lmcs_enabled_flag`) có thể có ý nghĩa giống như xác định cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) để biểu thị việc không sử dụng LMCS.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động giải mã phần chia hiện thời, dựa trên cờ cho phép công cụ mã hóa cấp phần chia. Cờ cho phép công cụ mã hóa cấp phần chia có thể bao gồm cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`). Thiết bị xử lý tín hiệu video có thể giải mã ảnh hiện thời bằng cách giải mã phần chia hiện thời được bao gồm trong ảnh hiện thời, dựa trên cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`). Cụ thể là, LMCS có thể được áp dụng khi giải mã phần chia hiện thời theo cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`). Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 1, và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối độ sáng (nếu chỉ số màu sắc là 0), quy trình ánh xạ độ sáng có thể được thực hiện. Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 1, và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối sắc độ (nếu chỉ số màu sắc lớn hơn 0), quy trình thay đổi tỷ lệ tín hiệu dư sắc độ có thể được thực hiện.

Ngoài ra, mặc dù không được minh họa trên Fig.26, để báo hiệu xem có sử dụng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS ở cấp phần chia hay không, `slice_header()` có thể bao gồm `slice_chroma_residual_scale_flag`. Nghĩa là, `slice_header()` có thể bao gồm cấu trúc cú pháp sau.

```
if(ph_lmcs_enabled_flag && !picture_header_in_slice_header_flag) {

slice_lmcs_enabled_flag // Descriptor: u(1)

if(ph_chroma_residual_scale_flag && slice_lmcs_enabled_flag)

slice_chroma_residual_scale_flag // Descriptor: u(1)
```

}

Trong cấu trúc ở trên, `slice_chroma_residual_scale_flag` có thể được bao gồm trong `slice_header()` nếu `ph_chroma_residual_scale_flag` là 1 và cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 1. Nếu `slice_chroma_residual_scale_flag` là 1, thì điều này xác định rằng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS có thể được sử dụng cho phần chia hiện thời. Nghĩa là, sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS có thể được sử dụng cho phần chia hiện thời với tham chiếu đến thông số của APS LMCS được biểu thị bởi bộ định danh (`ph_lmcs_aps_id`) cho APS LMCS. Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) là 0, thì điều này xác định rằng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS không được sử dụng cho phần chia hiện thời. Nếu `slice_chroma_residual_scale_flag` không tồn tại, thì giá trị của cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) có thể được suy ra khi cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 1, và 0 có thể được suy ra khi cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0. Nghĩa là, nếu ảnh được mã hóa chỉ bao gồm một phần chia được mã hóa, việc có sử dụng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS trong phần chia hiện thời hay không được xác định theo việc có sử dụng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS trong ảnh hiện thời hay không. Nếu cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) là 1, thì phần chia hiện thời (chỉ phần chia được mã hóa của ảnh được mã hóa) được bao gồm trong ảnh được mã hóa có thể sử dụng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS. Nếu cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) là 0, thì phần chia hiện thời được bao gồm trong ảnh được mã hóa có thể không sử dụng sự thay đổi tỷ lệ tín hiệu dư sắc độ của LMCS. LMCS có thể được áp dụng khi phần chia hiện thời được giải mã, theo ít nhất một trong số cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`), bộ định danh (`ph_lmcs_aps_id`) cho APS LMCS, cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) và cờ tỷ lệ phần dư sắc độ cấp phần chia (`slice_chroma_residual_scale_flag`). Nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) biểu thị việc sử dụng LMCS (`slice_lmcs_enabled_flag==1`), và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối độ sáng (nếu

chỉ số màu sắc là 0), thì quy trình ánh xạ độ sáng có thể được thực hiện.

Ngoài ra, nếu cờ cho phép LMCS cấp phần chia (`slice_lmcs_enabled_flag`) biểu thị việc sử dụng LMCS, thì khối được giải mã hiện thời được bao gồm trong phần chia hiện thời không phải là khối độ sáng (nếu chỉ số màu sắc không phải là 0), và cờ tỷ lệ phần dư sắc độ cấp ảnh (`ph_chroma_residual_scale_flag`) biểu thị sự cho phép sự thay đổi tỷ lệ tín hiệu dư sắc độ cho ảnh hiện thời, quy trình thay đổi tỷ lệ phần dư sắc độ có thể được thực hiện.

Nếu cờ tỷ lệ phần dư sắc độ cấp phần chia (`slice_chroma_residual_scale_flag`) là 1 và khối được giải mã hiện thời được bao gồm trong phần chia hiện thời là khối sắc độ (nếu chỉ số màu sắc lớn hơn 0), thì quy trình thay đổi tỷ lệ tín hiệu dư sắc độ có thể được thực hiện.

Hệ thống trực quan của con người (human visual system, HVS) có các đặc điểm là tương đối nhạy với dải tần số thấp và tương đối không nhạy với dải tần số cao. Dựa trên các đặc điểm của hệ thống trực quan của con người, khi phép lượng tử hóa được thực hiện với các trọng số biến theo các chỉ số tần số (chỉ số tần số ngang x và chỉ số tần số dọc y), hệ số biến đổi có thể được nén một cách hiệu quả. Theo sáng chế, danh sách trọng số biến $m[x][y]$ được gọi là hệ số thay đổi tỷ lệ hoặc danh sách thay đổi tỷ lệ theo các chỉ số tần số. Bộ giải mã có thể thu khôi hệ số biến đổi bằng cách lượng tử hóa ngược (thay đổi tỷ lệ) khối hệ số biến đổi được lượng tử hóa được thu từ dòng bit. Bộ giải mã có thể suy ra danh sách thay đổi tỷ lệ cuối cùng dựa trên danh sách thay đổi tỷ lệ $m[x][y]$ và danh sách `levelScale[][]` mà là kích thước bậc lượng tử hóa ngược mặc định trong quy trình thay đổi tỷ lệ (thủ tục lượng tử hóa ngược) cho hệ số biến đổi. Danh sách `levelScale[][]` biểu thị các kích thước bậc lượng tử hóa ngược được làm xấp xỉ bởi các số nguyên, và một giá trị bất kỳ được bao gồm trong `levelScale[][]` có thể được lựa chọn theo giá trị thông số lượng tử hóa (QP).

Danh sách thay đổi tỷ lệ $m[x][y]$ nêu trên có thể được báo hiệu rõ ràng, hoặc có thể được suy ra từ giá trị được tạo cấu hình trước khi không được báo hiệu rõ ràng. Theo sáng chế, danh sách thay đổi tỷ lệ $m[x][y]$ được báo hiệu rõ ràng được gọi là danh sách thay đổi

tỷ lệ rõ ràng. Danh sách thay đổi tỷ lệ rõ ràng có thể được bao gồm trong đơn vị NAL APS. Đơn vị NAL APS trong đó `aps_params_type` là `SCALING_APS` bao gồm dữ liệu danh sách thay đổi tỷ lệ rõ ràng, và danh sách thay đổi tỷ lệ rõ ràng `m[x][y]` có thể được suy ra dựa trên đó.

Việc liệu có sử dụng danh sách thay đổi tỷ lệ rõ ràng hay không có thể được báo hiệu ở cấp trình tự, ảnh và phần chia. Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho CLVS hiện thời có được cho phép hay không. Để báo hiệu xem có sử dụng danh sách thay đổi tỷ lệ rõ ràng ở cấp trình tự hay không, `seq_parameter_set_rbsp()` có thể bao gồm cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự (`sps_explicit_scaling_list_enabled_flag`). Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự (`sps_explicit_scaling_list_enabled_flag`) là 1, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng có thể được sử dụng trong CLVS tham chiếu SPS. Nói cách khác, được xác định rằng, khi giải mã phần chia được bao gồm trong CLVS, danh sách thay đổi tỷ lệ rõ ràng được bao gồm trong đơn vị NAL APS để được báo hiệu có thể được sử dụng trong quy trình thay đổi tỷ lệ cho hệ số biến đổi. Mặt khác, nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự (`sps_explicit_scaling_list_enabled_flag`) là 0, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng trong CLVS tham chiếu SPS. Nói cách khác, được xác định rằng, khi giải mã phần chia được bao gồm trong CLVS, danh sách thay đổi tỷ lệ được tạo cấu hình trước khác với danh sách thay đổi tỷ lệ rõ ràng, mà được bao gồm trong đơn vị NAL APS để được báo hiệu, có thể được sử dụng trong quy trình thay đổi tỷ lệ cho hệ số biến đổi. Danh sách thay đổi tỷ lệ được tạo cấu hình trước `m[x][y]` có thể luôn luôn được suy ra là 16 bất kể chỉ số tần số ngang x và chỉ số tần số dọc y .

Tiếp theo, để báo hiệu xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng ở cấp ảnh hay không, `picture_header_structure()` có thể bao gồm cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự (`sps_explicit_scaling_list_enabled_flag`) là 1. Dựa vào dòng 2641 của Fig.26, nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu

thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng (`sps_explicit_scaling_list_enabled_flag==1`), thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`), cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho ảnh hiện thời có được cho phép hay không. Hoạt động của dòng 2641 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2440 trên Fig.24, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Hoạt động của dòng 2641 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2540 trên Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit.

Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 1, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng có thể được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, được xác định rằng, khi giải mã phần chia được mã hóa của ảnh được mã hóa, danh sách thay đổi tỷ lệ rõ ràng được bao gồm trong đơn vị NAL APS để được báo hiệu có thể được sử dụng trong quy trình thay đổi tỷ lệ cho hệ số biến đổi. Mặt khác, nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 0, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia được mã hóa của ảnh được mã hóa. Nói cách khác, được xác định rằng, khi giải mã phần chia được mã hóa của ảnh được mã hóa, thì danh sách thay đổi tỷ lệ được tạo cấu hình trước khác với danh sách thay đổi tỷ lệ rõ ràng, mà được bao gồm trong đơn vị NAL APS để được báo hiệu, có thể được sử dụng trong quy trình thay đổi tỷ lệ cho hệ số biến đổi. Danh sách thay đổi tỷ lệ được tạo cấu hình trước $m[x][y]$ có thể luôn luôn được suy ra là 16 bất kể chỉ số tần số ngang x và chỉ số tần số dọc y . Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) không tồn tại, thì giá trị của cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) có thể được suy ra là 0.

Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 1, thì phần đầu ảnh có thể bao gồm bộ định danh cho APS (`aps_params_type` của `SCALING_APS`) bao gồm phần tử của danh sách thay đổi

tỷ lệ được tham chiếu. Dựa vào dòng 2642 của Fig.26, nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu bộ định danh (`ph_scaling_list_aps_id`) cho APS LMCS bao gồm phần tử của danh sách thay đổi tỷ lệ từ cấu trúc cú pháp phần đầu ảnh (`picture_header_structure()`). Hoạt động của dòng 2642 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2440 trên Fig.24, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Hoạt động của dòng 2642 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2540 trên Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp ảnh từ dòng bit. Bộ định danh (`ph_scaling_list_aps_id`) cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ xác định `adaptation_parameter_set_id` mà là bộ định danh APS của APS danh sách thay đổi tỷ lệ được tham chiếu trong phần chia được mã hóa của ảnh được mã hóa.

Tiếp theo, để báo hiệu xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng ở cấp phần chia hay không, `slice_header()` có thể bao gồm cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 1, và cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) là 0. Dựa vào dòng 2612 của Fig.26, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, và cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thu, từ phần đầu phần chia `slice_header()`, cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) biểu thị xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng cho phần chia hiện thời hay không. Hoạt động của dòng 2612 của Fig.26 có thể được bao gồm trong hoạt động của dòng 2511 trên Fig.25, gồm việc thu cờ cho phép công cụ mã hóa cấp phần chia từ dòng bit.

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động giải mã ảnh hiện thời dựa trên cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia và bộ định danh

(ph_scaling_list_aps_id) cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ.

Nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) là 1, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng có thể được sử dụng cho phần chia hiện thời. Nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng được sử dụng cho phần chia hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được bao gồm trong APS dựa trên bộ định danh (ph_scaling_list_aps_id) cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ. Biểu thị, bởi cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag), rằng danh sách thay đổi tỷ lệ rõ ràng được sử dụng cho phần chia hiện thời xác định rằng, khi phần chia hiện thời được giải mã, thì adaptation_parameter_set_id được bao gồm trong APS danh sách thay đổi tỷ lệ giống như bộ định danh (ph_scaling_list_aps_id) sao cho danh sách thay đổi tỷ lệ rõ ràng được báo hiệu có thể được sử dụng trong quy trình thay đổi tỷ lệ cho hệ số biến đổi.

Mặt khác, nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) là 0, thì điều này xác định rằng danh sách thay đổi tỷ lệ rõ ràng có thể được sử dụng cho phần chia hiện thời. Nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia hiện thời, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được tạo cấu hình trước. Tất cả các phần tử của danh sách thay đổi tỷ lệ được tạo cấu hình trước có thể là 16. Biểu thị, bởi cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag), rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia hiện thời xác định rằng, khi phần chia hiện thời được giải mã, thì danh sách thay đổi tỷ lệ được tạo cấu hình trước khác với danh sách thay đổi tỷ lệ rõ ràng được bao gồm trong đơn vị NAL APS để được báo hiệu được sử dụng trong quy trình thay đổi tỷ lệ cho các hệ số biến đổi. Danh sách thay đổi tỷ lệ được tạo cấu hình trước $m[x][y]$ có thể luôn luôn được suy ra là 16 bất

kể chỉ số tần số ngang x và chỉ số tần số dọc y.

Nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) không tồn tại, thì giá trị của cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) có thể được suy ra là giá trị của cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`). Nghĩa là, nếu ảnh được mã hóa bao gồm một phần chia được mã hóa, thì việc liệu danh sách thay đổi tỷ lệ rõ ràng có thể được sử dụng cho phần chia hiện thời hay không có thể được xác định theo cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`). Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 1, thì phần chia hiện thời (chỉ phần chia được mã hóa của ảnh được mã hóa) được bao gồm trong ảnh được mã hóa có thể sử dụng danh sách thay đổi tỷ lệ rõ ràng. Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) là 0, thì phần chia hiện thời được bao gồm trong ảnh được mã hóa không thể sử dụng danh sách thay đổi tỷ lệ rõ ràng.

Cụ thể hơn là, tham chiếu đến dòng 2620 của Fig.26, có thể có hai trường hợp trong đó cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia `slice_explicit_scaling_list_used_flag` không được thu từ dòng bit. Nghĩa là, nếu cờ phần đầu ảnh trong phần đầu phần chia (`picture_header_in_slice_header_flag`) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia (trường hợp thứ nhất), hoặc nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng (trường hợp thứ hai), cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) có thể không được thu từ dòng bit.

Thiết bị xử lý tín hiệu video có thể thực hiện các thủ tục sau để xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) dựa trên cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`). Nếu cờ phần đầu ảnh trong phần đầu phần chia

(picture_header_in_slice_header_flag) biểu thị rằng cấu trúc cú pháp phần đầu ảnh tồn tại trong phần đầu phần chia, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định giá trị của cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) là giống như giá trị của cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (ph_explicit_scaling_list_enabled_flag). Cụ thể là, trong trường hợp thứ nhất, cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (ph_explicit_scaling_list_enabled_flag) có thể biểu thị việc sử dụng danh sách thay đổi tỷ lệ rõ ràng hoặc việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng. Trong trường hợp này, thiết bị xử lý tín hiệu video có thể xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) là giống như cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (ph_explicit_scaling_list_enabled_flag). Nghĩa là, nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (ph_explicit_scaling_list_enabled_flag) biểu thị sự cho phép danh sách thay đổi tỷ lệ rõ ràng, thì thiết bị xử lý tín hiệu video có thể xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) để biểu thị việc sử dụng danh sách thay đổi tỷ lệ rõ ràng. Nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (ph_explicit_scaling_list_enabled_flag) biểu thị sự không cho phép danh sách thay đổi tỷ lệ rõ ràng, thì thiết bị xử lý tín hiệu video có thể xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) để biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng.

Nếu cờ phần đầu ảnh trong phần đầu phần chia (picture_header_in_slice_header_flag) biểu thị rằng không có cấu trúc cú pháp phần đầu ảnh nào tồn tại trong phần đầu phần chia, thì thiết bị xử lý tín hiệu video có thể thực hiện hoạt động xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) để biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng cho phần chia được bao gồm trong ảnh hiện thời. Dựa vào dòng 2510 của Fig.25, trường hợp trong đó cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (slice_explicit_scaling_list_used_flag) không được thu từ dòng bit mặc dù cờ phần đầu ảnh trong phần đầu phần chia (picture_header_in_slice_header_flag) không tồn tại trong phần

đầu phần chia có thể là trường hợp trong đó, như trong trường hợp thứ hai, cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng. Thiết bị xử lý tín hiệu video có thể xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) là giống như cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`). Nghĩa là, thiết bị xử lý tín hiệu video có thể xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) để biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng. Vì cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) biểu thị việc không kích hoạt, việc xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) là giống như cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh (`ph_explicit_scaling_list_enabled_flag`) có thể có ý nghĩa giống như xác định cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) để biểu thị việc không sử dụng danh sách thay đổi tỷ lệ rõ ràng.

Thậm chí nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) là 1, thì danh sách thay đổi tỷ lệ rõ ràng có thể không được sử dụng cho tất cả các khối biến đổi được bao gồm trong phần chia. Ví dụ, thậm chí nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`) là 1, nếu sự bỏ qua biến đổi được áp dụng cho khối biến đổi hiện thời, thì danh sách thay đổi tỷ lệ được tạo cấu hình trước khác với danh sách thay đổi tỷ lệ rõ ràng có thể được áp dụng. Ví dụ khác là, nếu LFNST được áp dụng cho khối biến đổi hiện thời, và nếu được biểu thị rằng không áp dụng danh sách thay đổi tỷ lệ rõ ràng cho khối biến đổi mà LFNT đã được áp dụng, thì danh sách thay đổi tỷ lệ được tạo cấu hình trước khác với danh sách thay đổi tỷ lệ rõ ràng có thể được áp dụng. Danh sách thay đổi tỷ lệ được tạo cấu hình trước $m[x][y]$ có thể luôn luôn được suy ra là 16 bất kể chỉ số tần số ngang x và chỉ số tần số dọc y .

Thiết bị xử lý tín hiệu video có thể thực hiện hoạt động giải mã phần chia hiện thời, dựa trên cờ cho phép công cụ mã hóa cấp phần chia. Cờ cho phép công cụ mã hóa cấp phần

chia có thể bao gồm cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`). Thiết bị xử lý tín hiệu video có thể giải mã ảnh hiện thời bằng cách giải mã phần chia hiện thời được bao gồm trong ảnh hiện thời, dựa trên cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia (`slice_explicit_scaling_list_used_flag`).

Các phương án nêu trên của sáng chế có thể được thực hiện bằng các cách khác nhau. Ví dụ, các phương án của sáng chế có thể được thực hiện bằng phần cứng, phần sụn, phần mềm hoặc tổ hợp của chúng.

Trong trường hợp thực hiện bởi phần cứng, phương pháp theo các phương án của sáng chế có thể được thực hiện bởi một hoặc nhiều mạch tích hợp cho ứng dụng cụ thể (application specific integrated circuit, ASIC), các bộ xử lý tín hiệu kỹ thuật số (digital signal processor, DSP), các thiết bị xử lý tín hiệu kỹ thuật số (digital signal processing device, DSPD), các thiết bị logic lập trình được (programmable logic device, PLD), mảng cổng lập trình được dạng trường (field programmable gate array, FPGA), các bộ xử lý, các bộ điều khiển, các bộ vi điều khiển, các bộ vi xử lý và thiết bị tương tự.

Trong trường hợp phương án thực hiện bằng phần sụn hoặc phần mềm, phương pháp theo các phương án của sáng chế có thể được thực hiện ở dạng môđun, thủ tục, chức năng, hoặc dạng tương tự mà thực hiện các chức năng hoặc các hoạt động được mô tả ở trên. Các mã phần mềm có thể được lưu trữ trong bộ nhớ để 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 cũng có thể được thực hiện ở dạng vật ghi bao gồm các lệnh thực thi được bởi máy tính, chẳng hạn như môđun chương trình được thực thi bởi máy tính. Vật ghi đọc được bởi máy tính có thể là vật ghi khả dụng bất kỳ mà có thể được truy cập bởi máy tính và bao gồm cả vật ghi khả biến và vật ghi không khả biến và cả vật ghi tháo lắp được và vật ghi không tháo lắp được. Hơn nữa, vật ghi đọc được bởi máy tính có thể bao gồm cả vật ghi lưu trữ bởi máy tính và vật ghi truyền thông. Vật ghi lưu trữ bởi máy tính bao gồm tất cả trong số vật ghi khả biến, không khả biến, tháo lắp được và không tháo lắp

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

Các phần mô tả ở trên của sáng chế nhằm mục đích minh họa, và người có hiểu biết trung bình trong lĩnh vực kỹ thuật, mà sáng chế thuộc về, sẽ có thể hiểu rằng sự cải biến đối với các dạng cụ thể khác có thể dễ dàng đạt được mà không thay đổi nguyên lý kỹ thuật hoặc các dấu hiệu cơ bản của sáng chế. Do đó, cần hiểu rằng các phương án được mô tả ở trên nhằm minh họa và không giới hạn ở tất cả các khía cạnh. Ví dụ, mỗi thành phần được mô tả dưới dạng kiểu duy nhất có thể được thực hiện theo cách phân tán, và tương tự, các thành phần được mô tả dưới dạng phân tán cũng có thể được thực hiện ở dạng kết hợp.

Phạm vi của sáng chế được thể hiện bởi các điểm yêu cầu bảo hộ được mô tả sau đây thay vì phần mô tả chi tiết, và tất cả các sự thay đổi hoặc các sự cải biến được suy ra từ ý nghĩa và phạm vi của các điểm yêu cầu bảo hộ và các khái niệm tương đương của chúng nên được hiểu là được bao gồm trong phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Vật ghi đọc được bởi máy tính lâu dài lưu trữ dòng bit, dòng bit được giải mã bằng phương pháp giải mã, phương pháp giải mã này bao gồm các bước:

thu cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (coded layer video sequence, CLVS) hiện thời hay không từ bộ thông số trình tự (sequence parameter set, SPS) được áp dụng cho CLVS hiện thời;

nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không;

thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có xuất hiện trong phần đầu phần chia hay không;

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia và cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép cho ảnh hiện thời, thu, từ phần đầu phần chia, cờ sử dụng công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là một giá trị trong số các giá trị,

trong đó một giá trị này bao gồm trường hợp trong đó một giá trị là cùng giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và

giải mã phần chia hiện thời, dựa trên cờ sử dụng công cụ mã hóa cấp phần chia.

2. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1, trong đó nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh xuất hiện trong phần đầu

phần chia hoặc cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa không được cho phép cho ảnh hiện thời, cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia.

3. Vật ghi đọc được bởi máy tính lâu dài theo điểm 2,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia:

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh xuất hiện trong phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa không được sử dụng cho phần chia hiện thời.

4. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia:

nếu giá trị của cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép cho ảnh hiện thời, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa được sử dụng cho phần chia hiện thời,

nếu giá trị của cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa không được cho phép cho ảnh hiện thời, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa không được sử dụng cho phần chia hiện thời.

5. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1, trong đó các giá trị của các cờ phần đầu ảnh trong phần đầu phần chia của tất cả các phần chia được mã hóa của trình tự video lớp được mã hóa (CLVS) hiện thời là giống nhau.

6. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1,

trong đó nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh xuất hiện trong phần đầu phần chia:

đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) phần đầu ảnh bao gồm cấu trúc cú pháp phần đầu ảnh không xuất hiện trong trình tự video lớp được mã hóa (CLVS) hiện thời.

7. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1,

trong đó nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia:

đơn vị ảnh (picture unit, PU) hiện thời bao gồm đơn vị lớp trừu tượng mạng (NAL) phần đầu ảnh.

8. Vật ghi đọc được bởi máy tính lâu dài theo điểm 7, trong đó đơn vị NAL phần đầu ảnh đứng trước đơn vị NAL lớp mã hóa video (video coding layer, VCL) thứ nhất của đơn vị ảnh (PU) hiện thời.

9. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1,

trong đó cờ cho phép công cụ mã hóa cấp trình tự bao gồm cờ cho phép tỷ lệ độ sáng ánh xạ với sắc độ (luma-mapping-with-chroma-scaling, LMCS) cấp trình tự biểu thị xem LMCS cho CLVS hiện thời có được cho phép hay không,

trong đó cờ cho phép công cụ mã hóa cấp ảnh bao gồm cờ cho phép LMCS cấp ảnh biểu thị xem LMCS có được cho phép cho ảnh hiện thời hay không,

trong đó cờ sử dụng công cụ mã hóa cấp phần chia bao gồm cờ sử dụng LMCS cấp phần chia biểu thị xem LMCS có được sử dụng cho phần chia hiện thời hay không,

phương pháp giải mã còn bao gồm bước:

nếu cờ cho phép LMCS cấp ảnh biểu thị rằng LMCS được cho phép cho ảnh hiện thời, thu bộ định danh cho bộ thông số thích ứng (adaptation parameter set, APS) LMCS bao gồm thông số LMCS từ cấu trúc cú pháp phần đầu ảnh,

trong đó phần chia hiện thời được giải mã dựa trên cờ sử dụng LMCS cấp phần chia và bộ định danh cho APS LMCS.

10. Vật ghi đọc được bởi máy tính lâu dài theo điểm 9, trong đó phương pháp còn bao gồm bước:

nếu cờ cho phép LMCS cấp ảnh biểu thị rằng LMCS được cho phép cho ảnh hiện thời và định dạng màu của ảnh hiện thời là định dạng bao gồm thành phần sắc độ, thu, từ cấu trúc cú pháp phần đầu ảnh, cờ tỷ lệ phần dư sắc độ cấp ảnh biểu thị xem sự thay đổi tỷ lệ tín hiệu dư sắc độ có được cho phép cho ảnh hiện thời hay không.

11. Vật ghi đọc được bởi máy tính lâu dài theo điểm 10, trong đó nếu cờ sử dụng LMCS cấp phần chia biểu thị rằng LMCS được sử dụng cho phần chia hiện thời và khối hiện thời được bao gồm trong phần chia hiện thời là khối độ sáng, quy trình ánh xạ độ sáng được áp dụng cho khối hiện thời.

12. Vật ghi đọc được bởi máy tính lâu dài theo điểm 10, trong đó nếu cờ sử dụng LMCS cấp phần chia biểu thị rằng LMCS được sử dụng cho phần chia hiện thời và khối hiện thời được bao gồm trong phần chia hiện thời không phải là khối độ sáng, quy trình thay đổi tỷ lệ phần dư sắc độ được áp dụng cho khối hiện thời.

13. Vật ghi đọc được bởi máy tính lâu dài theo điểm 1,

trong đó cờ cho phép công cụ mã hóa cấp trình tự bao gồm cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp trình tự biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho CLVS hiện thời có được cho phép hay không,

trong đó cờ cho phép công cụ mã hóa cấp ảnh bao gồm cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị xem danh sách thay đổi tỷ lệ rõ ràng cho ảnh hiện thời có được cho phép hay không,

trong đó cờ sử dụng công cụ mã hóa cấp phần chia bao gồm cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị xem danh sách thay đổi tỷ lệ rõ ràng có được sử dụng cho phần chia hiện thời hay không,

phương pháp giải mã còn bao gồm bước:

nếu cờ cho phép danh sách thay đổi tỷ lệ rõ ràng cấp ảnh biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng được cho phép cho ảnh hiện thời, thu bộ định danh cho bộ thông số thích ứng (APS) bao gồm phần tử của danh sách thay đổi tỷ lệ từ cấu trúc cú pháp phần đầu ảnh,

trong đó phần chia hiện thời được giải mã dựa trên cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia và bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ.

14. Vật ghi đọc được bởi máy tính lâu dài theo điểm 13, trong đó bước giải mã phần chia hiện thời bao gồm các bước:

nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được bao gồm trong APS dựa trên bộ định danh cho APS bao gồm phần tử của danh sách thay đổi tỷ lệ; và

nếu cờ sử dụng danh sách thay đổi tỷ lệ rõ ràng cấp phần chia biểu thị rằng danh sách thay đổi tỷ lệ rõ ràng không được sử dụng cho phần chia hiện thời, thay đổi tỷ lệ hệ số biến đổi của phần chia hiện thời bằng cách sử dụng danh sách thay đổi tỷ lệ được tạo cấu hình trước,

trong đó tất cả các phần tử của danh sách thay đổi tỷ lệ được tạo cấu hình trước là 16.

15. Thiết bị giải mã tín hiệu video bao gồm bộ xử lý,

trong đó bộ xử lý được tạo cấu hình để:

thu cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (CLVS) hiện thời hay không từ bộ thông số trình tự (SPS) được áp dụng cho CLVS hiện thời;

nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ

mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không;

thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có tồn tại trong phần đầu phần chia hay không;

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép cho ảnh hiện thời, thu, từ phần đầu phần chia, cờ sử dụng công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là một giá trị trong số các giá trị,

trong đó một giá trị này bao gồm trường hợp trong đó một giá trị là cùng giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và

giải mã phần chia hiện thời, dựa trên cờ sử dụng công cụ mã hóa cấp phần chia.

16. Thiết bị giải mã tín hiệu video theo điểm 15, trong đó nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh xuất hiện trong phần đầu phần chia hoặc cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa không được cho phép cho ảnh hiện thời, cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia.

17. Thiết bị giải mã tín hiệu video theo điểm 16,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia:

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh xuất hiện trong phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giống như giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa không được sử dụng cho phần chia hiện thời.

18. Thiết bị giải mã tín hiệu video theo điểm 15,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia:

nếu giá trị của cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép cho ảnh hiện thời, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa được sử dụng cho phần chia hiện thời,

nếu giá trị của cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa không được cho phép cho ảnh hiện thời, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là giá trị biểu thị rằng công cụ mã hóa không được sử dụng cho phần chia hiện thời.

19. Thiết bị giải mã tín hiệu video theo điểm 15, trong đó các giá trị của các cờ phần đầu ảnh trong phần đầu phần chia của tất cả các phần chia được mã hóa của trình tự video lớp được mã hóa (CLVS) hiện thời là giống nhau.

20. Thiết bị mã hóa tín hiệu video bao gồm bộ xử lý,

trong đó bộ xử lý được tạo cấu hình để:

thu dòng bit cần được giải mã bởi thiết bị giải mã sử dụng phương pháp giải mã, phương pháp giải mã bao gồm các bước:

thu cờ cho phép công cụ mã hóa cấp trình tự biểu thị xem công cụ mã hóa có được cho phép cho trình tự video lớp được mã hóa (CLVS) hiện thời hay không từ bộ thông số trình tự (SPS) được áp dụng cho CLVS hiện thời;

nếu cờ cho phép công cụ mã hóa cấp trình tự biểu thị rằng công cụ mã hóa được cho phép cho CLVS hiện thời, thu cờ cho phép công cụ mã hóa cấp ảnh biểu thị xem công cụ

mã hóa có được cho phép cho ảnh hiện thời được bao gồm trong CLVS hiện thời hay không;

thu, từ phần đầu phần chia của phần chia hiện thời được bao gồm trong ảnh hiện thời, cờ phần đầu ảnh trong phần đầu phần chia biểu thị xem cấu trúc cú pháp phần đầu ảnh có xuất hiện trong phần đầu phần chia hay không;

nếu cờ phần đầu ảnh trong phần đầu phần chia biểu thị rằng cấu trúc cú pháp phần đầu ảnh không xuất hiện trong phần đầu phần chia, và cờ cho phép công cụ mã hóa cấp ảnh biểu thị rằng công cụ mã hóa được cho phép cho ảnh hiện thời, thu, từ phần đầu phần chia, cờ sử dụng công cụ mã hóa cấp phần chia biểu thị xem công cụ mã hóa có được sử dụng cho phần chia hiện thời hay không,

trong đó nếu cờ sử dụng công cụ mã hóa cấp phần chia không được thu từ phần đầu phần chia, giá trị của cờ sử dụng công cụ mã hóa cấp phần chia được suy ra là một giá trị trong số các giá trị,

trong đó một giá trị này bao gồm trường hợp trong đó một giá trị là cùng giá trị của cờ cho phép công cụ mã hóa cấp ảnh; và

giải mã phần chia hiện thời, dựa trên cờ sử dụng công cụ mã hóa cấp phần chia.

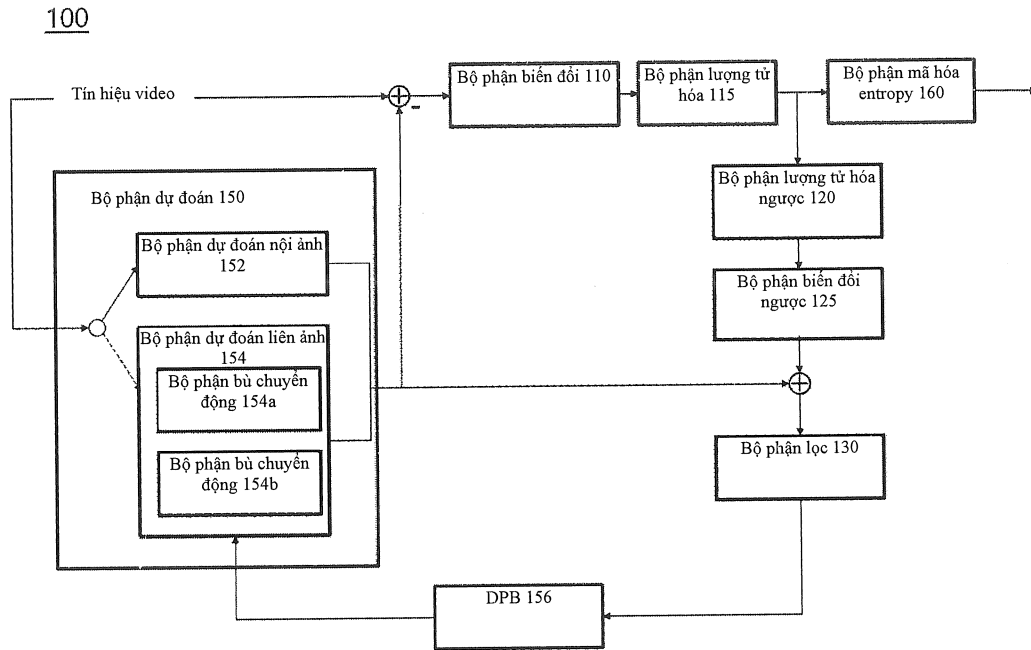


Fig.1

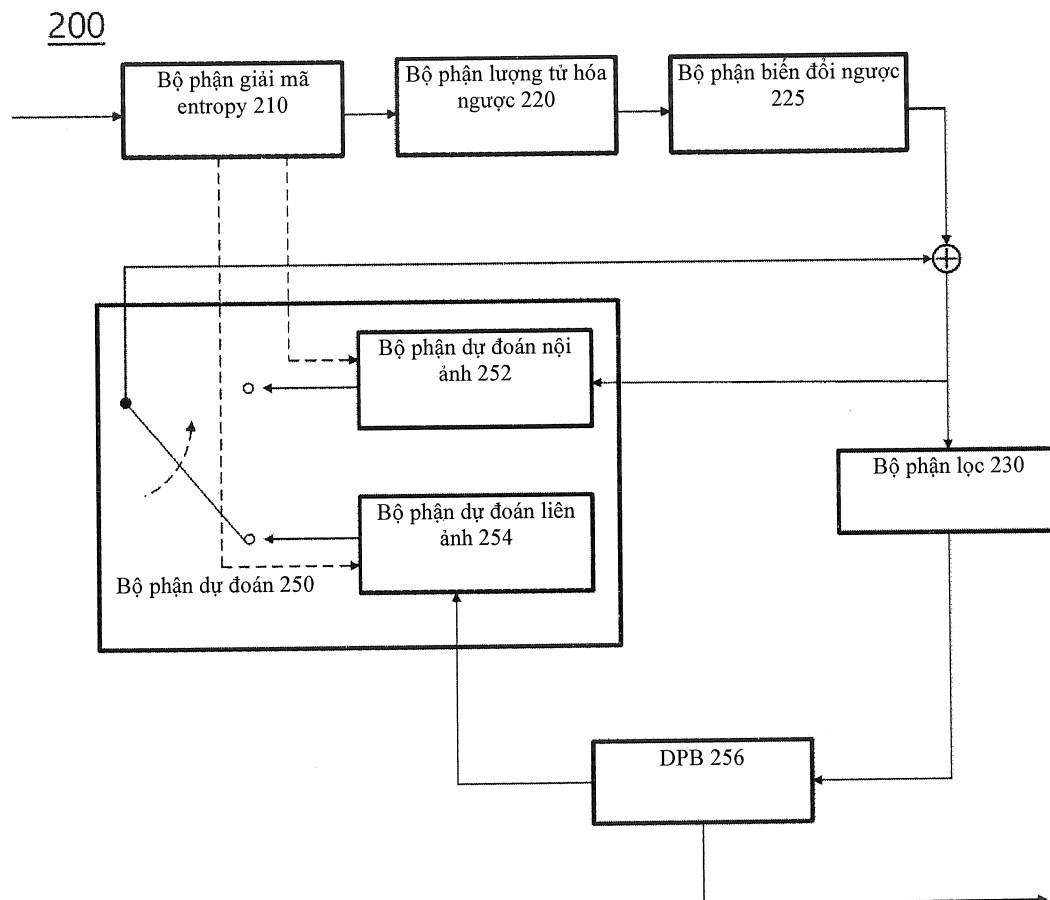


Fig.2

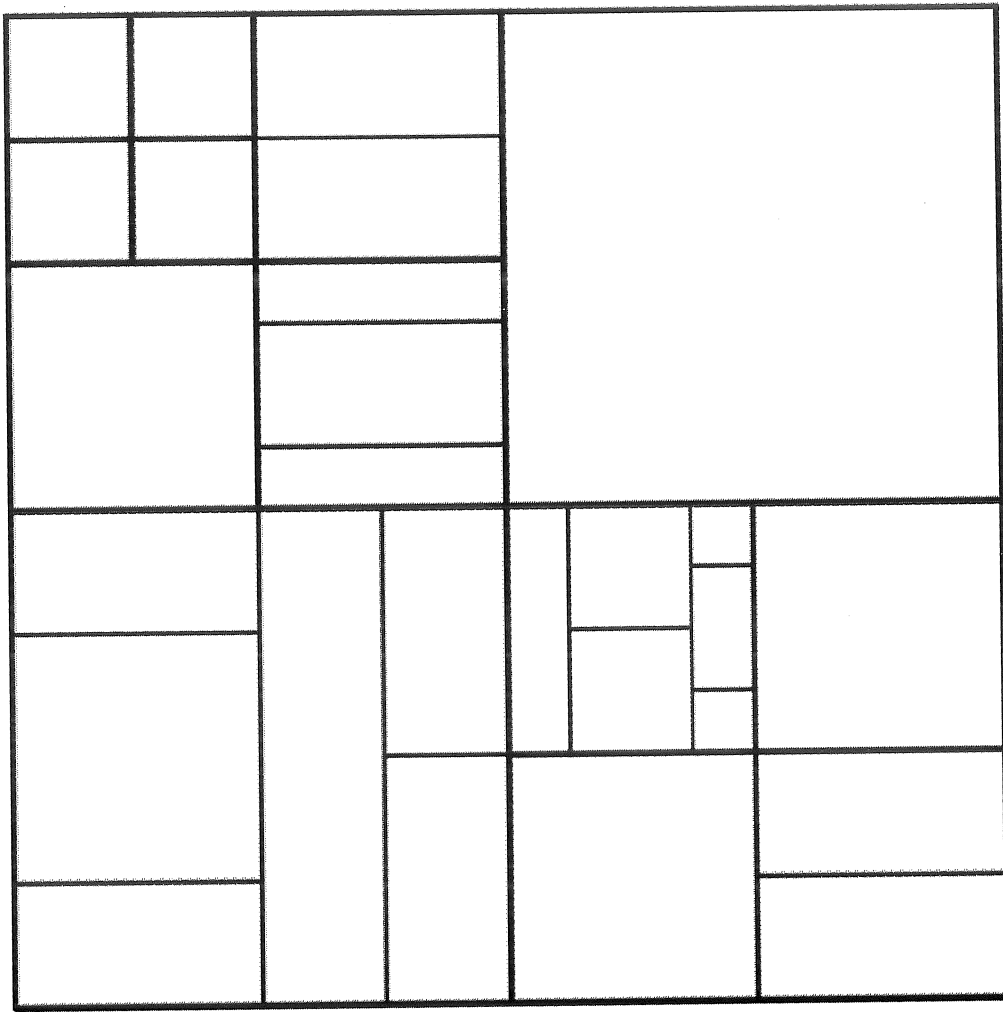


Fig.3

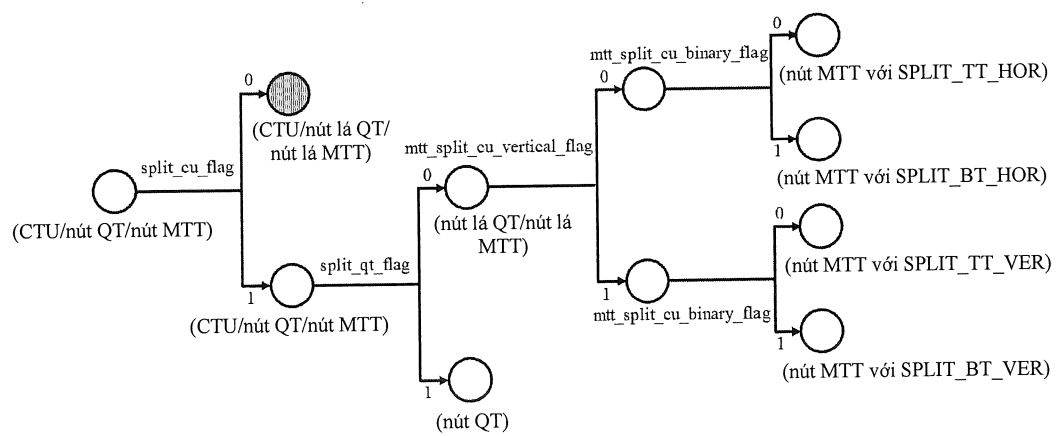
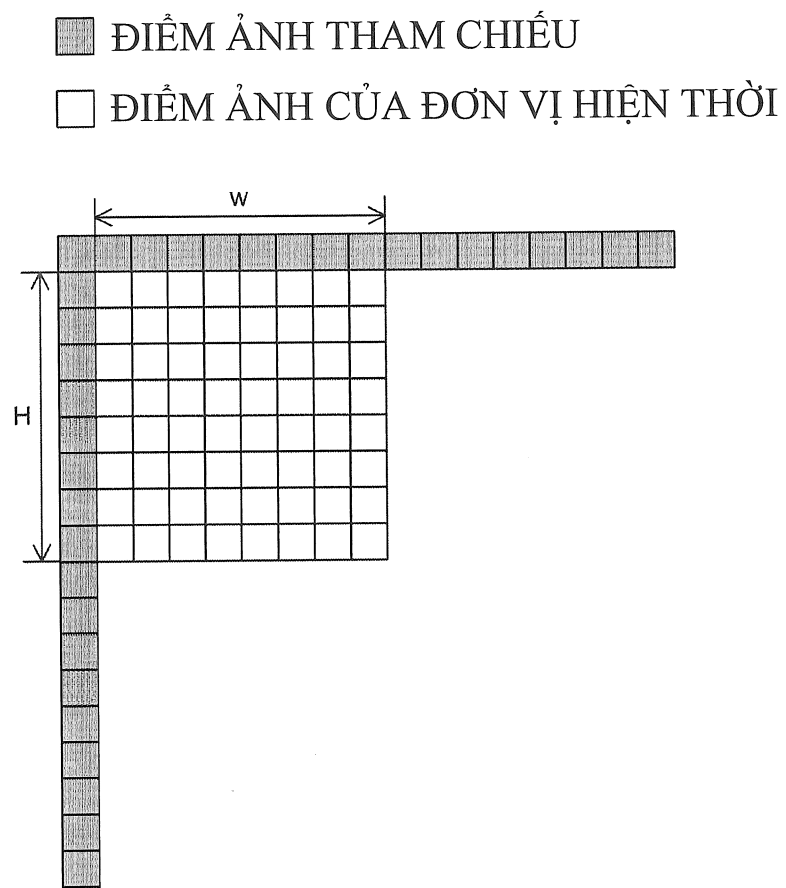


Fig.4

*Fig.5*

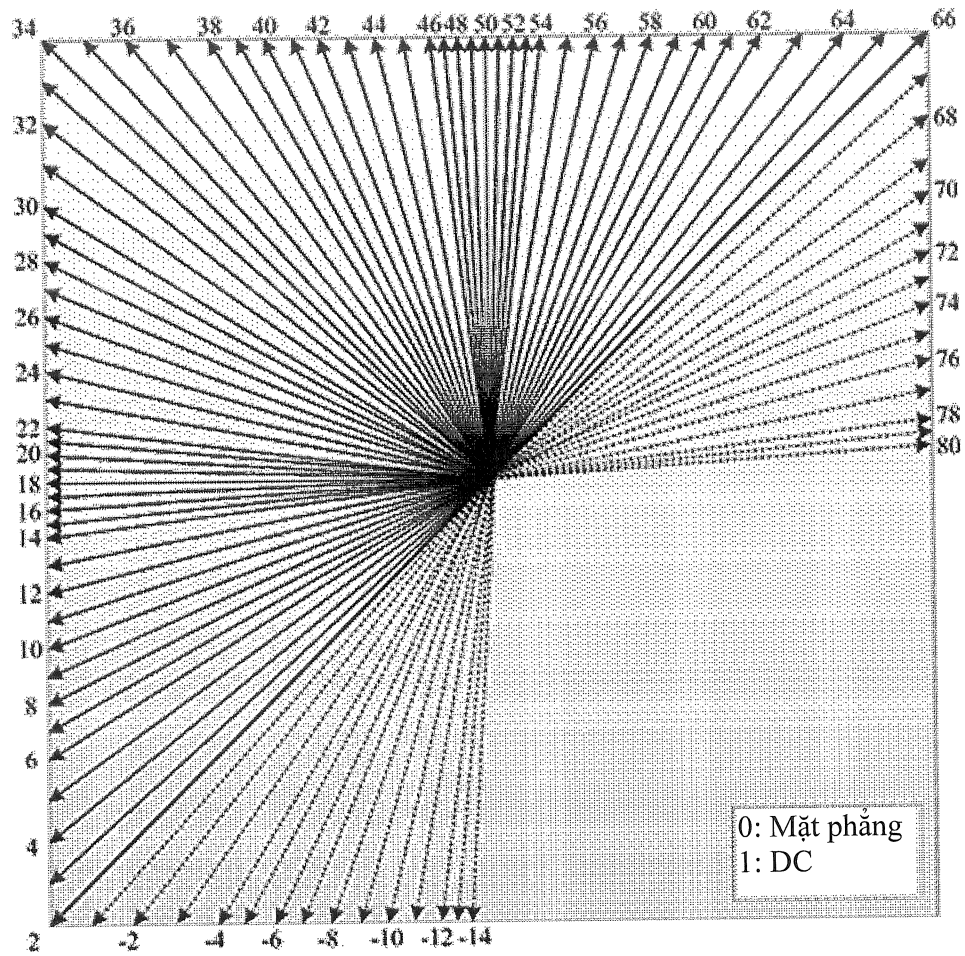


Fig. 6

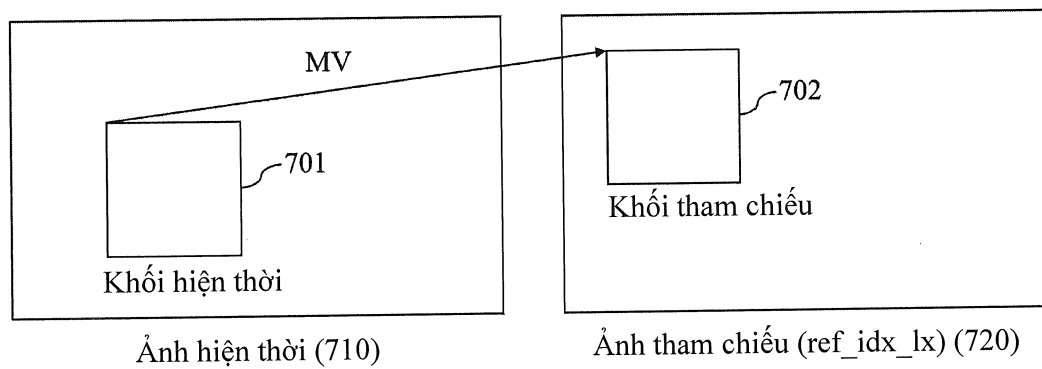


Fig. 7

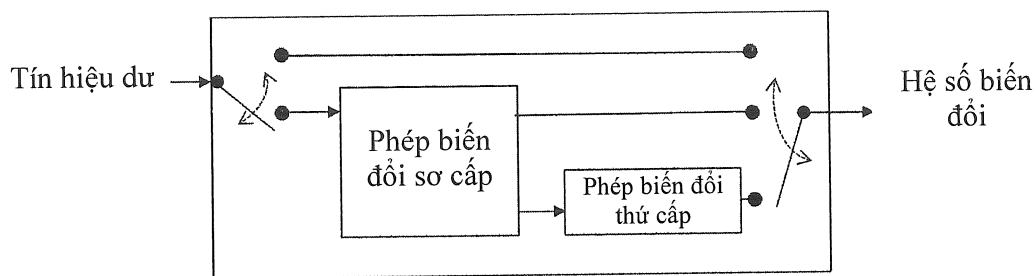


Fig. 8

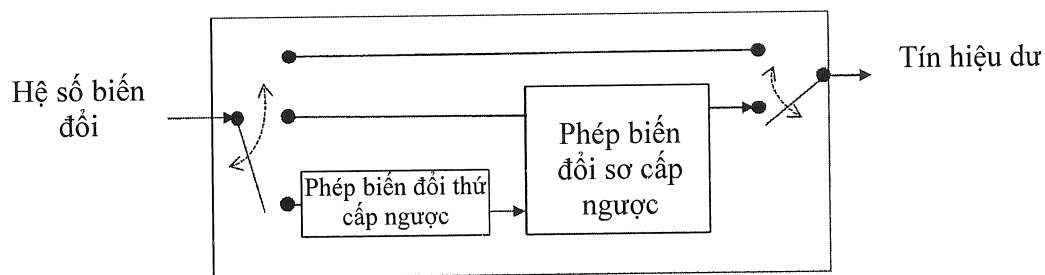


Fig. 9

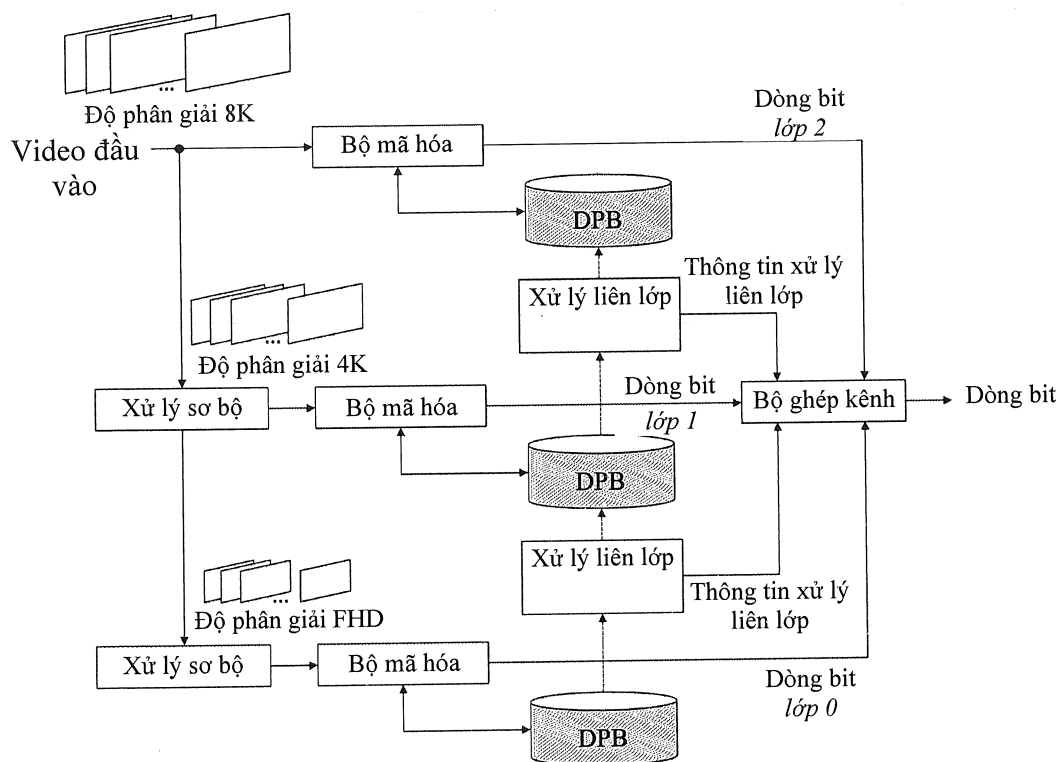


Fig. 10

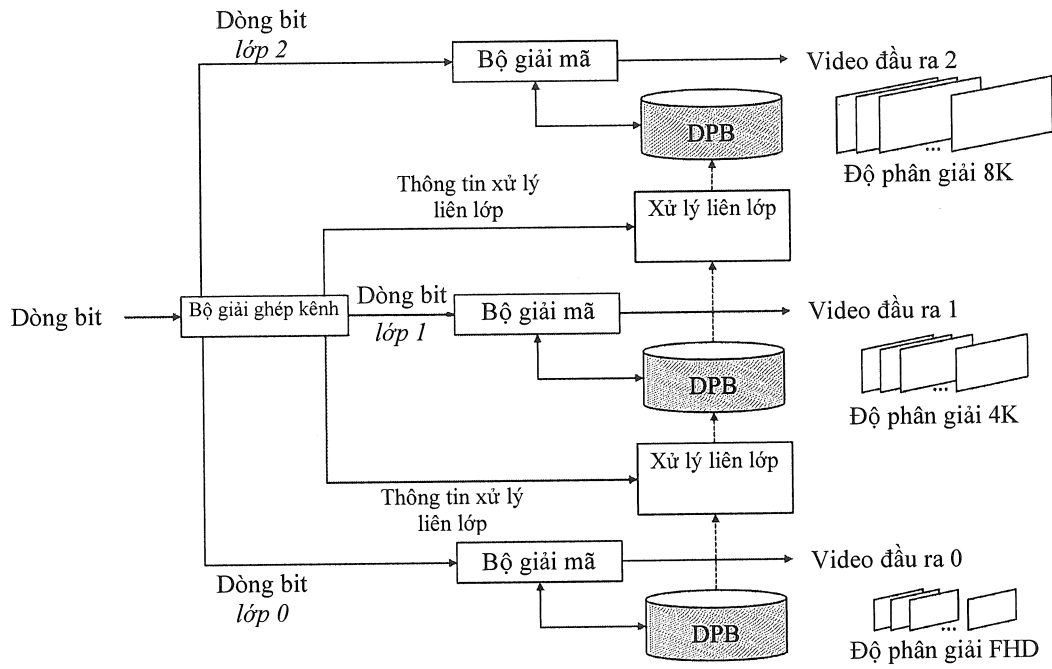


Fig.11

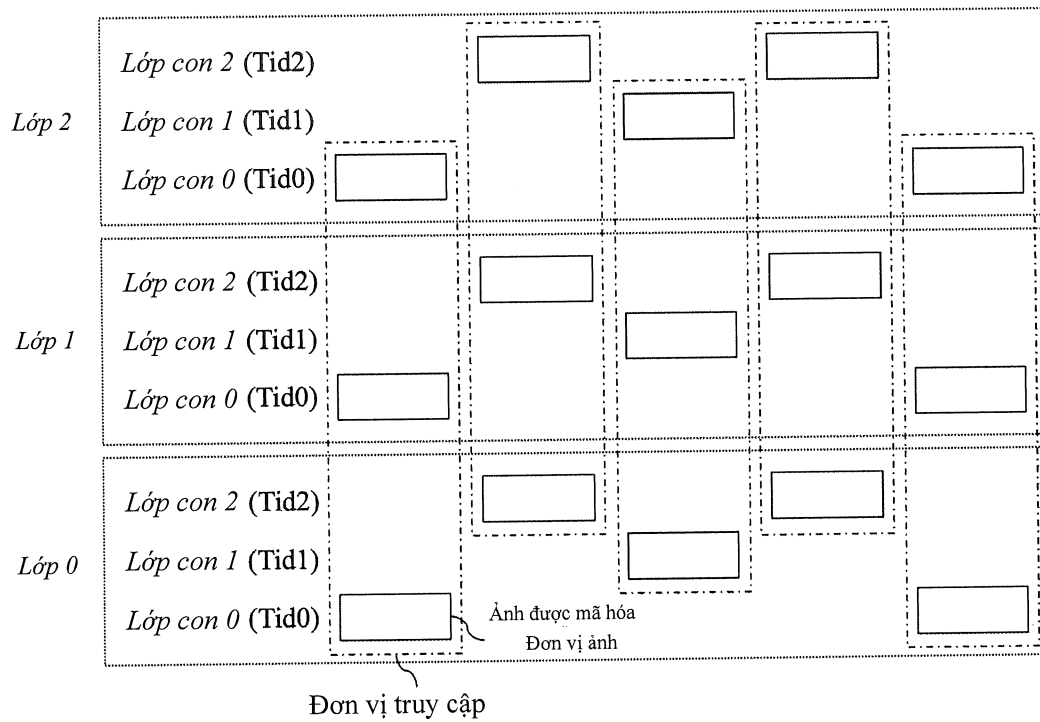


Fig.12

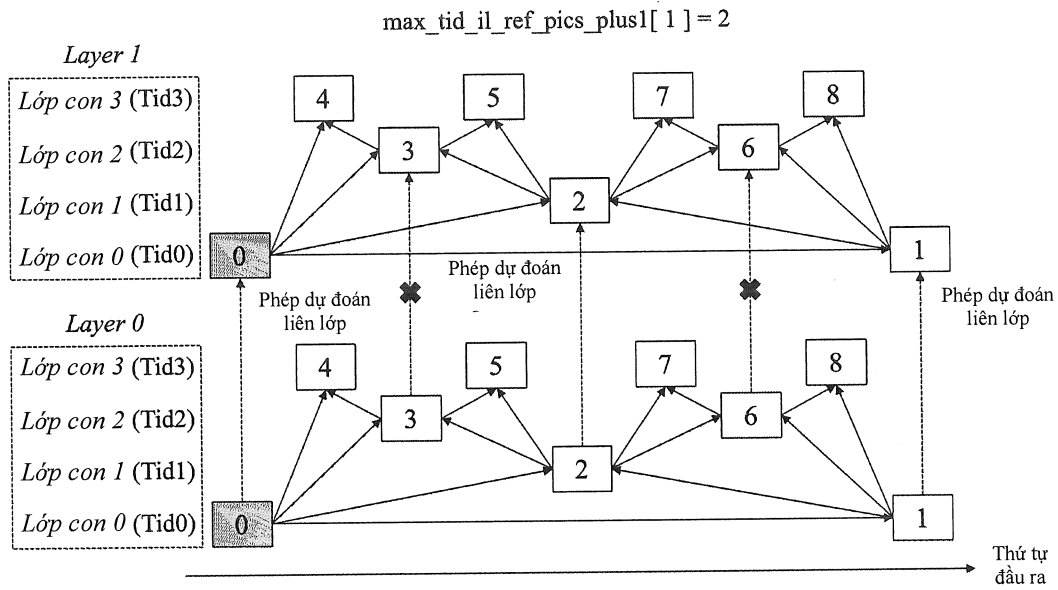


Fig.13

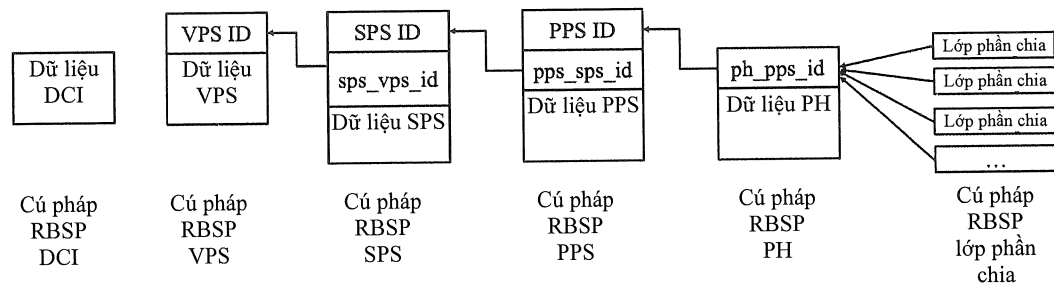


Fig.14

video_parameter_set_rbsp() {	Bộ mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(6)
vps_max_sublayers_minus1	u(3)
if(vps_max_layers_minus1 > 0 && vps_max_sublayers_minus1 > 0)	
vps_all_layers_same_num_sublayers_flag	u(1)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i]) {	
for(j = 0; j < i; j++)	
vps_direct_ref_layer_flag[i][j]	u(1)
max_tid_ref_present_flag[i]	u(1)
if(max_tid_ref_present_flag[i])	
max_tid_il_ref_pics_plus1[i]	u(3)
}	
}	
}	
}	
if(vps_max_layers_minus1 > 0) {	
if(vps_all_independent_layers_flag)	
each_layer_is_an_ols_flag	u(1)
if(!each_layer_is_an_ols_flag) {	
if(!vps_all_independent_layers_flag)	
ols_mode_idc	u(2)
if(ols_mode_idc == 2) {	
num_output_layer_sets_minus1	u(8)
for(i = 1; i <= num_output_layer_sets_minus1; i++)	
for(j = 0; j <= vps_max_layers_minus1; j++)	
ols_output_layer_flag[i][j]	u(1)
}	
}	
}	
}	
...	
}	

Fig.15

Các biến NumDirectRefLayers[i], DirectRefLayerIdx[i][d], NumRefLayers[i], RefLayerIdx[i][r] và LayerUsedAsRefLayerFlag[j] được suy ra như sau:

```

for( i = 0; i <= vps_max_layers_minus1; i++ ) {
    for( j = 0; j <= vps_max_layers_minus1; j++ ) {
        dependencyFlag[ i ][ j ] = vps_direct_ref_layer_flag[ i ][ j ]
        for( k = 0; k < i; k++ )
            if( vps_direct_ref_layer_flag[ i ][ k ] && dependencyFlag[ k ][ j ] )
                dependencyFlag[ i ][ j ] = 1
    }
    LayerUsedAsRefLayerFlag[ i ] = 0
}
for( i = 0; i <= vps_max_layers_minus1; i++ ) {
    for( j = 0, d = 0, r = 0; j <= vps_max_layers_minus1; j++ ) {
        if( vps_direct_ref_layer_flag[ i ][ j ] ) {
            DirectRefLayerIdx[ i ][ d++ ] = j
            LayerUsedAsRefLayerFlag[ j ] = 1
        }
        if( dependencyFlag[ i ][ j ] )
            RefLayerIdx[ i ][ r++ ] = j
    }
    NumDirectRefLayers[ i ] = d
    NumRefLayers[ i ] = r
}

```

Fig.16

```

NumOutputLayersInOls[ 0 ] = 1
OutputLayerIdInOls[ 0 ][ 0 ] = vps_layer_id[ 0 ]
NumSubLayersInLayerInOLS[ 0 ][ 0 ] = vps_max_sub_layers_minus1 + 1
LayerUsedAsOutputLayerFlag[ 0 ] = 1
for( i = 1; i <= vps_max_layers_minus1; i++ ) {
    if( each_layer_is_an_ols_flag || ols_mode_idc < 2 )
        LayerUsedAsOutputLayerFlag[ i ] = 1
    else /*( !each_layer_is_an_ols_flag && ols_mode_idc == 2 ) */
        LayerUsedAsOutputLayerFlag[ i ] = 0
}
for( i = 1; i < TotalNumOls; i++ ) {
    if( each_layer_is_an_ols_flag || ols_mode_idc == 0 ) {
        NumOutputLayersInOls[ i ] = 1
        OutputLayerIdInOls[ i ][ 0 ] = vps_layer_id[ i ]
        for( j = 0; j < i && ( ols_mode_idc == 0 ); j++ )
            NumSubLayersInLayerInOLS[ i ][ j ] = max_tid_il_ref_pics_plus1[ i ]
        NumSubLayersInLayerInOLS[ i ][ i ] = vps_max_sub_layers_minus1 + 1
    } else if( ols_mode_idc == 1 ) {
        NumOutputLayersInOls[ i ] = i + 1
        for( j = 0; j < NumOutputLayersInOls[ i ]; j++ ) {
            OutputLayerIdInOls[ i ][ j ] = vps_layer_id[ j ]
            NumSubLayersInLayerInOLS[ i ][ j ] = vps_max_sub_layers_minus1 + 1
        }
    } else if( ols_mode_idc == 2 ) {
        for( j = 0; j <= vps_max_layers_minus1; j++ ) {
            layerIncludedInOlsFlag[ i ][ j ] = 0
            NumSubLayersInLayerInOLS[ i ][ j ] = 0
        }
        for( k = 0; j = 0; k <= vps_max_layers_minus1; k++ )
            if( ols_output_layer_flag[ i ][ k ] ) {
                layerIncludedInOlsFlag[ i ][ k ] = 1
                LayerUsedAsOutputLayerFlag[ k ] = 1
                OutputLayerIdx[ i ][ j ] = k
                OutputLayerIdInOls[ i ][ j++ ] = vps_layer_id[ k ]
                NumSubLayersInLayerInOLS[ i ][ k ] = vps_max_sub_layers_minus1 + 1
            }
        NumOutputLayersInOls[ i ] = j
        for( j = 0; j < NumOutputLayersInOls[ i ]; j++ ) {
            idx = OutputLayerIdx[ i ][ j ]
            for( k = 0; k < NumRefLayers[ idx ]; k++ ) {
                layerIncludedInOlsFlag[ i ][ RefLayerIdx[ idx ][ k ] ] = 1
                if( NumSubLayersInLayerInOLS[ i ][ RefLayerIdx[ idx ][ k ] ] <
                    max_tid_il_ref_pics_plus1[ idx ] )
                    NumSubLayersInLayerInOLS[ i ][ RefLayerIdx[ idx ][ k ] ] =
                        max_tid_il_ref_pics_plus1[ idx ]
            }
        }
    }
}
}
}

```

Fig.17


```

NumLayersInOls[ 0 ] = 1
LayerIdInOls[ 0 ][ 0 ] = vps_layer_id[ 0 ]
for( i = 1; i < TotalNumOls; i++ ) {
    if( each_layer_is_an_ols_flag ) {
        NumLayersInOls[ i ] = 1
        LayerIdInOls[ i ][ 0 ] = vps_layer_id[ i ]
    } else if( ols_mode_idc == 0 || ols_mode_idc == 1 ) {
        NumLayersInOls[ i ] = i + 1
        for( j = 0; j < NumLayersInOls[ i ]; j++ )
            LayerIdInOls[ i ][ j ] = vps_layer_id[ j ]
    } else if( ols_mode_idc == 2 ) {
        for( k = 0, j = 0; k <= vps_max_layers_minus1; k++ )
            if( layerIncludedInOlsFlag[ i ][ k ] )
                LayerIdInOls[ i ][ j++ ] = vps_layer_id[ k ]
        NumLayersInOls[ i ] = j
    }
}

```

Fig.18

```

NumOutputLayersInOls[ 0 ] = 1
OutputLayerIdInOls[ 0 ][ 0 ] = vps_layer_id[ 0 ]
NumSubLayersInLayerInOLS[ 0 ][ 0 ] = vps_max_sub_layers_minus1 + 1
LayerUsedAsOutputLayerFlag[ 0 ] = 1
for( i = 1; i <= vps_max_layers_minus1; i++ ) {
    if( each_layer_is_an_ols_flag || ols_mode_idc < 2 )
        LayerUsedAsOutputLayerFlag[ i ] = 1
    else /*( !each_layer_is_an_ols_flag && ols_mode_idc == 2 ) */
        LayerUsedAsOutputLayerFlag[ i ] = 0
}
for( i = 1; i < TotalNumOls; i++ ) {
    if( each_layer_is_an_ols_flag || ols_mode_idc == 0 ) {
        NumOutputLayersInOls[ i ] = 1
        OutputLayerIdInOls[ i ][ 0 ] = vps_layer_id[ i ]
        if( each_layer_is_an_ols_flag )
            NumSubLayersInLayerInOLS[ i ][ 0 ] = vps_max_sub_layers_minus1 + 1
        else {
            for( j = 0; j < i && ( ols_mode_idc == 0 ); j++ )
                NumSubLayersInLayerInOLS[ i ][ j ] = max_tid_il_ref_pics_plus1[ i ]
            NumSubLayersInLayerInOLS[ i ][ i ] = vps_max_sub_layers_minus1 + 1
        }
    } else if( ols_mode_idc == 1 ) {
        NumOutputLayersInOls[ i ] = i + 1
        for( j = 0; j < NumOutputLayersInOls[ i ]; j++ ) {
            OutputLayerIdInOls[ i ][ j ] = vps_layer_id[ j ]
            NumSubLayersInLayerInOLS[ i ][ j ] = vps_max_sub_layers_minus1 + 1
        }
    } else if( ols_mode_idc == 2 ) {
        for( j = 0; j <= vps_max_layers_minus1; j++ ) {
            layerIncludedInOlsFlag[ i ][ j ] = 0
            temp[ i ][ j ] = 0
        }
        for( k = 0; j = 0; k <= vps_max_layers_minus1; k++ )
            if( ols_output_layer_flag[ i ][ k ] ) {
                layerIncludedInOlsFlag[ i ][ k ] = 1
                LayerUsedAsOutputLayerFlag[ k ] = 1
                OutputLayerIdx[ i ][ j ] = k
                OutputLayerIdInOls[ i ][ j++ ] = vps_layer_id[ k ]
                temp[ i ][ k ] = vps_max_sub_layers_minus1 + 1
            }
        NumOutputLayersInOls[ i ] = j
        for( j = 0; j < NumOutputLayersInOls[ i ]; j++ ) {
            idx = OutputLayerIdx[ i ][ j ]
            for( k = 0; k < NumRefLayers[ idx ]; k++ ) {
                layerIncludedInOlsFlag[ i ][ RefLayerIdx[ idx ][ k ] ] = 1
                if( temp[ i ][ RefLayerIdx[ idx ][ k ] ] < max_tid_il_ref_pics_plus1[ idx ] )
                    temp[ i ][ RefLayerIdx[ idx ][ k ] ] = max_tid_il_ref_pics_plus1[ idx ]
            }
        }
        for( j = 0; k = 0; j <= vps_max_layers_minus1; j++ ) {
            if( layerIncludedInOlsFlag[ i ][ j ] )
                NumSubLayersInLayerInOLS[ i ][ k++ ] = temp[ i ][ j ]
        }
    }
}
}

```

Fig.19

video_parameter_set_rbsp() {	Bộ mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(6)
vps_max_sublayers_minus1	u(3)
if(vps_max_layers_minus1 > 0 && vps_max_sublayers_minus1 > 0)	
vps_all_layers_same_num_sublayers_flag	u(1)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i]) {	
for(j = 0; j < i; j++)	
vps_direct_ref_layer_flag[i][j]	u(1)
max_tid_ref_present_flag[i]	u(1)
if(max_tid_ref_present_flag[i])	
max_tid_il_ref_pics_plus1[i]	u(3)
}	
}	
}	
if(vps_max_layers_minus1 > 0) {	
if(vps_all_independent_layers_flag)	
each_layer_is_an_ols_flag	u(1)
if(!each_layer_is_an_ols_flag) {	
if(!vps_all_independent_layers_flag)	
ols_mode_idc	u(2)
if(ols_mode_idc == 2) {	
num_output_layer_sets_minus1	u(8)
for(i = 1; i <= num_output_layer_sets_minus1; i++)	
for(j = 0; j <= vps_max_layers_minus1; j++)	
ols_output_layer_flag[i][j]	u(1)
}	
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(!vps_independent_layer_flag[i] && ols_mode_idc != 1) {	
max_tid_ref_present_flag[i]	u(1)
if(max_tid_ref_present_flag[i])	
max_tid_il_ref_pics_plus1[i]	u(3)
}	
}	
...	
}	

Fig.20

video parameter set rbsp() {	Bộ mô tả
vps video parameter set id	u(4)
vps max layers minus1	u(6)
vps max sublayers minus1	u(3)
if(vps max layers minus1 > 0 && vps max sublayers minus1 > 0)	
vps all layers same num sublayers flag	u(1)
...	
vps num ptls minus1	u(8)
for(i = 0; i <= vps num ptls minus1; i++) {	
if(i > 0)	
pt present flag[i]	u(1)
if(vps_max_sublayers_minus1 > 0 && !vps all layers same num sublayers flag)	
ptl max temporal id[i]	u(3)
}	
for(i = 0; i < vps num dpb params; i++) {	
if(vps_max_sublayers_minus1 > 0 && !vps all layers same num sublayers flag)	
dpb max temporal id[i]	u(3)
dpb_parameters(dpb_max_temporal_id[i], vps sublayer dpb params present flag)	
}	
...	
for(i = 0; i <= num ols hrd params minus1; i++) {	
if(vps_max_sublayers_minus1 > 0 && !vps all layers same num sublayers flag)	
hrd max tid[i]	u(3)
firstSubLayer = vps sublayer cpb params present flag ? 0 : hrd_max_tid[i]	
ols hrd_parameters(firstSubLayer, hrd_max_tid[i])	
}	
...	

Fig.21

seq parameter set rbsp() {	Bộ mô tả
sps seq parameter set id	u(4)
sps video parameter set id	u(4)
sps_max_sublayers_minus1	u(3)
...	

sps_max_sublayers_minus1 cộng 1 xác định số lượng các lớp con thời gian tối đa mà có thể có mặt trong mỗi CLVS tham chiếu đến SPS. Giá trị của **sps_max_sublayers_minus1** nên nằm trong khoảng từ 0 đến **vps_max_sublayers_minus1**.



sps_max_sublayers_minus1 cộng 1 xác định số lượng các lớp con thời gian tối đa mà có thể có mặt trong mỗi CLVS tham chiếu đến SPS. Giá trị của **sps_max_sublayers_minus1** nên nằm trong khoảng từ 0 đến **vps_max_sublayers_minus1**. Nếu **vps_all_layers_same_num_sublayers_flag**, khi có mặt, bằng 1, thì giá trị của **sps_max_sublayers_minus1** nên bằng **vps_max_sublayers_minus1**.

Fig.22

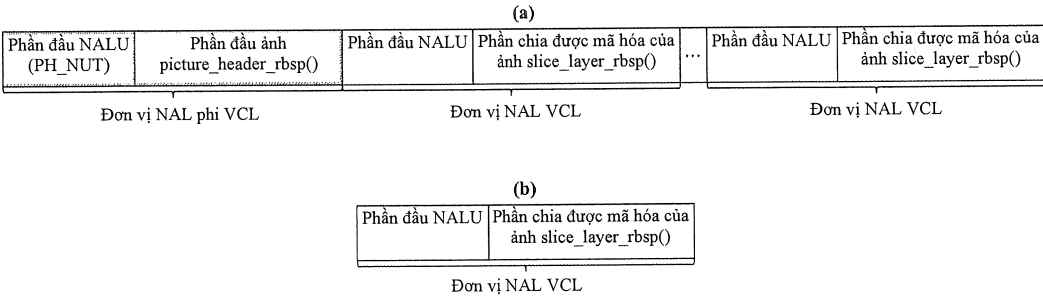


Fig.23

Cú pháp RBSP bộ thông số trình tự	
seq_parameter_set_rbsp() {	Bộ mô tả
sps_seq_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
...	
sps_coding_tool_enabled_flag	u(1)
...	
rbsp_trailing_bits()	
}	
Cú pháp RBSP bộ thông số ảnh	
pic_parameter_set_rbsp() {	Bộ mô tả
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	u(4)
...	
rbsp_trailing_bits()	
}	
Cú pháp RBSP phần đầu ảnh	
picture_header_rbsp() {	Bộ mô tả
picture_header_structure()	
rbsp_trailing_bits()	
}	
Cú pháp cấu trúc phần đầu ảnh	
picture_header_structure() {	Bộ mô tả
...	
ph_pic_parameter_set_id	ue(v)
...	
if(sps_coding_tool_enabled_flag)	
ph_coding_tool_enabled_flag	u(1)
...	
}	
Cú pháp RBSP lớp phần chia	
slice_layer_rbsp() {	Bộ mô tả
slice_header()	
slice_data()	
rbsp_slice_trailing_bits()	
}	
Cú pháp phân đầu phần chia nói chung	
slice_header() {	Bộ mô tả
picture_header_in_slice_header_flag	u(1)
if(picture_header_in_slice_header_flag)	
picture_header_structure()	
...	
if(ph_coding_tool_enabled_flag)	
slice_coding_tool_enabled_flag	u(1)
...	
byte_alignment()	
}	

2430

2440

2420

2410

Fig.24

Cú pháp RBSP bộ thông số trình tự	
seq_parameter_set_rbsp() {	Bộ mô tả
sps_seq_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
...	
sps_coding_tool_enabled_flag	u(1)
...	
rbsp_trailing_bits()	
}	
Cú pháp RBSP bộ thông số ảnh	
pic_parameter_set_rbsp() {	Bộ mô tả
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	u(4)
...	
rbsp_trailing_bits()	
}	
Cú pháp RBSP phần đầu ảnh	
picture_header_rbsp() {	Bộ mô tả
picture_header_structure()	
rbsp_trailing_bits()	
}	
Cú pháp cấu trúc phần đầu ảnh	
picture_header_structure() {	Bộ mô tả
...	
ph_pic_parameter_set_id	ue(v)
...	
if(sps_coding_tool_enabled_flag)	
ph_coding_tool_enabled_flag	u(1)
...	
}	
Cú pháp RBSP lớp phần chia	
slice_layer_rbsp() {	Bộ mô tả
slice_header()	
slice_data()	
rbsp_slice_trailing_bits()	
}	
Cú pháp phần đầu phần chia nói chung	
slice_header() {	Bộ mô tả
picture_header_in_slice_header_flag	u(1)
if(picture_header_in_slice_header_flag)	
picture_header_structure()	
...	
if(ph_coding_tool_enabled_flag && !picture_header_in_slice_header_flag)	
slice_coding_tool_enabled_flag	u(1)
...	
byte_alignment()	
}	

2530

2540

2520

2510

2511

Fig.25

Cú pháp Rbsp bộ thông số trình tự		
seq_parameter_set_rbsp() {		Bộ mô tả
sps_seq_parameter_set_id	u(4)	
sps_video_parameter_set_id	u(4)	
...		
sps_lmcs_enabled_flag	u(1)	
...		
sps_explicit_scaling_list_enabled_flag	u(1)	
...		
rbsp_trailing_bits()		
}		
Cú pháp cấu trúc phần đầu ảnh		
picture_header_structure() {		Bộ mô tả
...		
ph_pic_parameter_set_id	ue(v)	
...		
if(sps_lmcs_enabled_flag) {		2631
ph_lmcs_enabled_flag	u(1)	2632
if(ph_lmcs_enabled_flag) {		
ph_lmcs_aps_id	u(2)	
if(ChromaArrayType != 0)		
ph_chroma_residual_scale_flag	u(1)	
}		
}		
if(sps_explicit_scaling_list_enabled_flag) {		2641
ph_explicit_scaling_list_enabled_flag	u(1)	2642
if(ph_explicit_scaling_list_enabled_flag)		
ph_scaling_list_aps_id	u(3)	
}		
...		
}		
Cú pháp phần đầu phân chia nói chung		
slice_header() {		Bộ mô tả
picture_header_in_slice_header_flag	u(1)	
if(picture_header_in_slice_header_flag)		
picture_header_structure()		
...		
if(ph_lmcs_enabled_flag && !picture_header_in_slice_header_flag)		2610
slice_lmcs_enabled_flag	u(1)	2620
if(ph_explicit_scaling_list_enabled_flag && !picture_header_in_slice_header_flag)		
slice_explicit_scaling_list_used_flag	u(1)	
...		
byte_alignment()		
}		

Fig.26