



- (12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ
1-0045027
(51)^{2020.01} H04N 19/176; H04N 19/159; H04N 19/70; H04N 19/44; H04N 19/105 (13) B

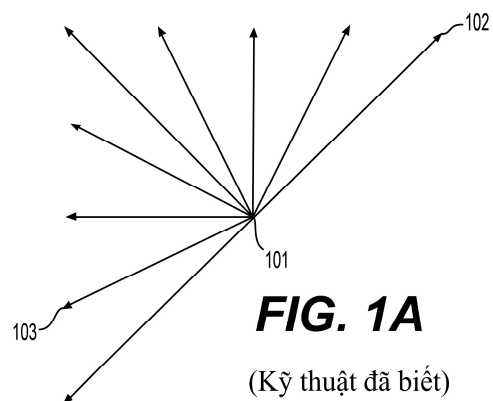
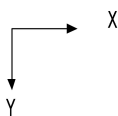
-
- (21) 1-2021-07347 (22) 02/04/2021
(86) PCT/US2021/025559 02/04/2021 (87) WO2021/207023 14/10/2021
(30) 63/005,345 05/04/2020 US; 63/010,272 15/04/2020 US; 63/024,140 13/05/2020 US;
63/027,848 20/05/2020 US; 63/028,222 21/05/2020 US; 17/220,672 01/04/2021 US
(45) 25/04/2025 445 (43) 27/06/2022 411A
(71) TENCENT AMERICA LLC (US)
2747 Park Boulevard, Palo Alto, California 94306, United States of America
(72) CHOI, Byeongdoo (KR); LIU, Shan (US); WENGER, Stephan (DE).
(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)
-

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ MÃ HÓA VIDEO

(21) 1-2021-07347

(57) Sáng chế đề cập đến các phương pháp và thiết bị mã hóa/giải mã video. Thiết bị bao gồm mạch xử lý mà thu nhận cờ cổng đối với mỗi trong số ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa. Mỗi cờ cổng chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không. Thông tin điều kiện ràng buộc dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa. Việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không được xác định dựa trên cờ cổng. Nhóm thông tin điều kiện ràng buộc bao gồm ít nhất một cờ điều kiện ràng buộc đối với tập con của các khối mã hóa. Thông tin dự đoán đối với tập con của các khối mã hóa được xác định dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không.

	R01	R02	R03	R04	R05	R06	R07	R08	R09
R10	S11	S12	S13	S14					
R20	S21	S22	S23	S24					
R30	S31	S32	S33	S34					
R40	S41	S42	S43	S44					
R50									
R60									
R70									



Lĩnh vực kỹ thuật được đề cập

Sáng chế mô tả các phương án liên quan đến việc mã hóa video.

Tình trạng kỹ thuật của sáng chế

Phần mô tả tình trạng kỹ thuật được đề xuất ở đây nhằm mục đích trình bày chung ngữ cảnh của sáng chế. Công việc của các tác giả sáng chế, trong phạm vi công việc được mô tả trong phần tình trạng kỹ thuật này, cũng như các khía cạnh của phần mô tả mà có thể không phải là kỹ thuật đã biết tại thời điểm nộp đơn, không được thừa nhận một cách rõ ràng hoặc ngụ ý như kỹ thuật đã biết so với sáng chế.

Việc mã hóa và giải mã video có thể được thực hiện nhờ sử dụng dự đoán liên ảnh với việc bù chuyển động. Video số không được nén có thể bao gồm một loạt ảnh, mỗi ảnh mà có kích thước không gian, ví dụ, là 1920 x 1080 mẫu độ chói và các mẫu sắc độ được kết hợp. Một loạt ảnh có thể có tốc độ ảnh cố định hoặc biến thiên (cũng thường được gọi là tốc độ khung), ví dụ, bằng 60 ảnh trên giây hoặc 60 Hz. Video không được nén có các yêu cầu tốc độ bit cụ thể. Ví dụ, video 1080p60 4:2:0 tại 8 bit trên mẫu (độ phân giải mẫu độ chói 1920x1080 tại tốc độ khung 60 Hz) yêu cầu gần 1,5 Gbit/s băng thông. Mỗi giờ của video này yêu cầu nhiều hơn 600 GByte của không gian lưu trữ.

Một mục đích của việc mã hóa và giải mã video có thể là làm giảm sự dư thừa trong tín hiệu video đầu vào, thông qua việc nén. Việc nén có thể giúp làm giảm các yêu cầu băng thông và/hoặc không gian lưu trữ nêu trên, trong một vài trường hợp xuống by hai bậc độ lớn hoặc nhiều hơn. Cả việc nén tổn hao và nén không tổn hao, cũng như kết hợp của chúng có thể được sử dụng. Việc nén không tổn hao liên quan đến các kỹ thuật trong đó bản sao chính xác của tín hiệu gốc có thể được khôi phục từ tín hiệu gốc được nén. Khi sử dụng việc nén tổn hao, tín hiệu được khôi phục có thể không đồng nhất với tín hiệu gốc, nhưng độ méo giữa các tín hiệu gốc và tín hiệu được khôi phục là đủ nhỏ để tạo ra tín hiệu được khôi phục hữu ích cho việc áp dụng dự định. Trong trường hợp của video,

việc nén tổn hao được sử dụng rộng rãi. Lượng méo được cho phép phụ thuộc vào việc ứng dụng; ví dụ, các người dùng của các ứng dụng tạo dòng người tiêu dùng có thể cho phép lượng méo cao hơn so với các người dùng của các ứng dụng phân phối ti-vi. Tỷ lệ nén có thể đạt được có thể chỉ báo rằng: độ méo cho phép/chấp nhận cao hơn có thể đạt được các tỷ lệ nén cao hơn.

Bộ mã hóa video và bộ giải mã có thể sử dụng các kỹ thuật từ một vài hạng mục chung, bao gồm, ví dụ, bù chuyển động, biến đổi, lượng tử hóa, và mã hóa entropy.

Kỹ thuật mã hóa-giải mã video có thể bao gồm các kỹ thuật được biết đến như là nội mã hóa. Trong việc nội mã hóa, các giá trị mẫu được biểu diễn mà không tham chiếu tới các mẫu hoặc dữ liệu khác từ các ảnh tham chiếu được khôi phục trước đó. Trong một vài việc mã hóa-giải mã video, ảnh được phân chia theo không gian thành các khối của các mẫu. Khi tất cả các khối của các mẫu được mã hóa trong chế độ nội mã hóa, ảnh này có thể ảnh nội mã hóa. Các ảnh nội dự đoán và các dẫn xuất của chúng như các ảnh làm mới bộ giải mã độc lập, có thể được sử dụng để thiết lập lại trạng thái bộ giải mã và do đó có thể được sử dụng như là ảnh thứ nhất trong dòng bit video được mã hóa và phiên video, hoặc như là ảnh tĩnh. Các mẫu của khối nội mã hóa có thể được đưa vào biến đổi, và các hệ số biến đổi có thể được lượng tử hóa trước khi mã hóa entropy. Việc nội dự đoán có thể là kỹ thuật mà tối thiểu hóa các giá trị mẫu trong miền biến đổi trước. Trong một vài trường hợp, giá trị DC càng nhỏ sau khi biến đổi, và các hệ số AC càng nhỏ, các ít bit hơn mà được yêu cầu tại kích bước lượng tử hóa định trước để biểu diễn khối sau khi mã hóa entropy.

Việc nội mã hóa truyền thông như đã biết từ, ví dụ các kỹ thuật mã hóa thế hệ MPEG-2, không sử dụng việc nội dự đoán. Tuy nhiên, một vài kỹ thuật nén video mới hơn bao gồm các kỹ thuật mà thử nghiệm, ví dụ, từ dữ liệu mẫu xung quanh và/hoặc siêu dữ liệu thu được trong việc mã hóa/giải mã các khối dữ liệu lân cận về mặt không gian, và đứng trước trong thứ tự mã hóa. Do đó, các kỹ thuật này được gọi là các kỹ thuật “nội dự đoán”. Lưu ý rằng trong ít nhất

một vài trường hợp, việc nội dự đoán là việc sử dụng dữ liệu tham chiếu chỉ từ ảnh hiện tại đang khôi phục và không phải từ các ảnh tham chiếu.

Có thể có nhiều dạng nội dự đoán khác nhau. Khi nhiều hơn một trong số các kỹ thuật này có thể được sử dụng trong kỹ thuật mã hóa video định trước, kỹ thuật được sử dụng có thể được mã hóa trong chế độ nội dự đoán. Trong các trường hợp nhất định, các chế độ có thể có các chế độ con và/hoặc các tham số, và chúng có thể được mã hóa riêng biệt hoặc được chứa trong từ mã chế độ. Từ mã nào để sử dụng cho kết hợp chế độ/chế độ con/tham số định trước có thể có tác động tới hiệu quả mã hóa thông qua việc nội dự đoán, và vì vậy kỹ thuật mã hóa entropy được sử dụng để dịch từ mã thành dòng bit.

Chế độ định trước của việc nội dự đoán được giới thiệu với H.264, được tinh chỉnh trong H.265, và được tinh chỉnh hơn nữa trong các kỹ thuật mã hóa mới hơn như joint exploration model (JEM), mã hóa video đa năng (VVC-versatile video coding), và tập hợp điểm chuẩn (BMS-benchmark set). Khối biến dự đoán có thể được tạo thành nhờ sử dụng các giá trị mẫu lân cận thuộc về các mẫu đã khả dụng. Các giá trị mẫu của các mẫu lân cận được sao chép vào khối biến dự đoán theo chiều. Việc tham chiếu tới chiều được sử dụng có thể được mã hóa trong dòng bit hoặc có thể được tự dự đoán.

Viện dẫn tới Fig.1A, được thể hiện trong phần phía dưới bên phải là tập con con của chín chiều biến dự đoán đã biết từ 33 chiều biến dự đoán có thể theo H.265 (tương ứng với 33 chế độ góc trong số 35 chế độ nội dự đoán). Điểm trong đó các mũi tên hội tụ 101 biểu diễn mẫu được dự đoán. Các mũi tên biểu diễn chiều mà từ đó mẫu đang được dự đoán. Ví dụ, mũi tên 102 chỉ báo rằng mẫu 101 được dự đoán từ mẫu hoặc các mẫu tới phía trên bên phải, tại góc 45 độ từ chiều ngang. Tương tự, mũi tên 103 chỉ báo rằng mẫu 101 được dự đoán từ mẫu hoặc các mẫu tới phía dưới bên trái của mẫu 101, trong góc 22.5 độ từ chiều ngang.

Viện dẫn tới Fig.1A, trên phía trên cùng bên trái có khối hình vuông 104 được thể hiện của các mẫu 4 x 4 (được chỉ báo bởi đường đậm, nét đứt). Khối hình vuông 104 bao gồm 16 mẫu, được gắn nhãn “S”, vị trí của nó trong

chiều Y (ví dụ, chỉ số hàng) và vị trí của nó trong chiều X (ví dụ, chỉ số cột). Ví dụ, mẫu S21 là mẫu thứ hai trong chiều Y (từ trên cùng) và mẫu thứ nhất (từ bên trái) trong chiều X. Tương tự, mẫu S44 là mẫu thứ tư trong khối 104 trong cả các chiều Y và X. Do khối là các mẫu có kích cỡ 4×4 , S44 nằm tại dưới cùng bên phải. Được thể hiện tiếp là các mẫu tham chiếu mà tuân theo phương pháp đánh số tương tự. Mẫu tham chiếu được gắn nhãn R, vị trí Y (ví dụ, chỉ số hàng) và vị trí X (chỉ số cột) của nó so với khối 104. Trong cả tiêu chuẩn H.264 và H.265, các mẫu dự đoán lân cận với khối đang khôi phục; do đó không có các giá trị âm cần cần được sử dụng.

Việc dự đoán nội ảnh có thể hoạt động bằng cách sao chép các giá trị mẫu tham chiếu từ các mẫu lân cận nếu cần bởi chiều dự đoán được báo hiệu. Ví dụ, giả định dòng bit được mã hóa bao gồm báo hiệu mà, đối với khối này, chỉ báo chiều dự đoán đồng nhất với mũi tên 102—tức là, các mẫu được dự đoán từ mẫu dự đoán hoặc các mẫu tới phía trên bên phải, tại góc 45 độ từ chiều ngang. Trong trường hợp này, các mẫu S41, S32, S23, và S14 được dự đoán từ cùng mẫu tham chiếu R05. Sau đó, mẫu S44 được dự đoán từ mẫu tham chiếu R08.

Trong các trường hợp định trước, các giá trị của nhiều mẫu tham chiếu có thể được kết hợp, ví dụ thông qua việc nội suy, để tính toán mẫu tham chiếu; đặc biệt khi các chiều không thể được phân chia đều bởi 45 độ.

Số lượng chiều có thể tăng lên khi kỹ thuật mã hóa video phát triển. Trong tiêu chuẩn H.264 (năm 2003), chín chiều khác nhau có thể được biểu diễn. Số lượng chiều được tăng lên 33 trong tiêu chuẩn H.265 (năm 2013), và JEM/VVC/BMS, tại thời điểm bộc lộ, có thể hỗ trợ tối đa 65 chiều. Các thử nghiệm đã được thực hiện để nhận dạng các chiều có khả năng nhất, và các kỹ thuật nhất định trong việc mã hóa entropy được sử dụng để biểu diễn các chiều có khả năng này trong số lượng bit nhỏ, chấp nhận sự bất lợi nhất định đối với các chiều có khả năng kém hơn. Ngoài ra, các chiều này có thể đôi lúc được dự đoán từ các chiều lân cận được sử dụng trong các khối lân cận đã được giải mã.

FIG.1B thể hiện là sơ đồ giản lược 180 mà thể hiện 65 chiều nội dự đoán theo JEM để minh họa số lượng tăng lên của các chiều dự đoán theo thời gian.

Việc ánh xạ của các bit chiều nội dự đoán trong dòng bit được mã hóa mà biểu diễn chiều có thể khác nhau giữa các kỹ thuật mã hóa video; và có thể nằm trong phạm vi, ví dụ, từ các ánh xạ trực tiếp đơn giản của chiều dự đoán tới chế độ nội dự đoán, tới các từ mã, tới các phương pháp thích nghi phức tạp bao gồm các chế độ có khả năng xảy ra nhất, và các kỹ thuật tương tự. Tuy nhiên, trong tất cả các trường hợp, có thể có các chiều nhất định mà ít có khả năng xảy ra trong nội dung video so với các chiều nhất định khác. Do mục đích của việc nén video làm giảm sự dư thừa, các chiều ít có khả năng này, trong kỹ thuật mã hóa video hoạt động tốt, sẽ được biểu diễn bởi số lượng bit lớn hơn so với các chiều có nhiều khả năng.

Việc bù chuyển động có thể là kỹ thuật nén tổn hao và có thể liên quan đến các kỹ thuật trong đó khối của dữ liệu mẫu từ ảnh được khôi phục trước đó hoặc một phần của ảnh này (ảnh tham chiếu), sau khi được dịch chuyển về mặt không gian trong chiều được chỉ báo bởi vector chuyển động (sau đây gọi tắt là MV-motion vector), được sử dụng cho việc dự đoán của ảnh được khôi phục mới hoặc một phần của ảnh này. Trong một vài trường hợp, ảnh tham chiếu có thể tương tự như ảnh đang được khôi phục. Các MV có thể có hai chiều X và Y, hoặc ba chiều, chiều thứ ba là chỉ báo của ảnh tham chiếu được sử dụng (chiều thứ ba, một cách gián tiếp, có thể là chiều thời gian).

Trong một vài kỹ thuật nén video, MV có thể được áp dụng tới vùng định trước của dữ liệu mẫu có thể được dự đoán từ các MV khác, ví dụ từ các MV liên quan đến vùng khác của dữ liệu mẫu liền kề về mặt không gian với vùng được khôi phục, và đứng trước MV này trong thứ tự giải mã. Nhờ đó có thể về mặt bản chất làm giảm lượng dữ liệu được yêu cầu để mã hóa MV, nhờ đó loại bỏ sự dư thừa và làm tăng hiệu quả nén. Việc dự đoán MV có thể hoạt động hiệu quả, ví dụ, do khi việc mã hóa tín hiệu video đầu vào thu được từ camera (được biết đến là video tự nhiên) có khả năng rằng các vùng lớn hơn

vùng mà MV đơn có thể được áp dụng tới di chuyển trong chiều tương tự và do đó, có thể, trong một vài trường hợp, được dự đoán nhờ sử dụng vector chuyển động tương tự thu được từ các MV của vùng lân cận. Điều này dẫn đến MV được tìm thấy đối với vùng định trước là tương tự hoặc giống như MV được dự đoán từ các MV xung quanh, và nói cách khác, có thể được biểu diễn, sau khi mã hóa entropy, trong số lượng bit nhỏ hơn so với những gì được sử dụng nếu mã hóa MV một cách trực tiếp. Trong một vài trường hợp, việc dự đoán MV có thể là ví dụ của việc nén không tổn hao của tín hiệu (tức là các MV) thu được từ tín hiệu gốc (tức là dòng mẫu). Trong các trường hợp khác, bản thân việc dự đoán MV có thể là có tổn hao, ví dụ do các sai số làm tròn khi tính toán biến dự đoán từ một vài MV xung quanh.

Các cơ chế dự đoán MV được mô tả trong H.265/HEVC (ITU-T Rec. H.265, "Mã hóa video hiệu quả cao", tháng 12 năm 2016). Trong số các cơ chế dự đoán MV mà H.265 khuyến nghị, được mô tả ở đây là kỹ thuật mà sau đây được gọi là "hộp nhất không gian".

Viện dẫn tới Fig.2, khối hiện tại 201 bao gồm các mẫu mà đã được tìm thấy bởi bộ mã hóa trong xử lý tìm kiếm chuyển động có thể được dự đoán từ khối trước đó có cùng kích cỡ mà đã được dịch chuyển về mặt không gian. Thay vì mã hóa MV này một cách trực tiếp, MV có thể thu được từ siêu dữ liệu được kết hợp với một hoặc nhiều ảnh tham chiếu, ví dụ từ ảnh tham chiếu gần nhất (trong thứ tự giải mã), nhờ sử dụng MV được kết hợp với một trong số năm mẫu xung quanh, được ký hiệu là A0, A1, và B0, B1, B2 (202 đến 206, một cách lần lượt). Trong tiêu chuẩn H.265, việc dự đoán MV có thể sử dụng các biến dự đoán từ cùng ảnh tham chiếu mà khối lân cận đang sử dụng.

Bản chất kỹ thuật của sáng chế

Các khía cạnh của sáng chế đề xuất các phương pháp và thiết bị mã hóa/giải mã video. Trong một vài ví dụ, thiết bị giải mã video bao gồm hệ mạch xử lý. Mạch xử lý có cấu trúc thu nhận cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa. Mỗi cờ cổng có thể chỉ báo rằng nhóm thông

tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ công tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không. Thông tin điều kiện ràng buộc có thể dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa. Mạch xử lý có cấu trúc để xác định, dựa trên cờ công của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc, rằng nhóm thông tin điều kiện ràng buộc của cờ công có được hiện diện trong thông tin điều kiện ràng buộc hay không, nhóm thông tin điều kiện ràng buộc của cờ công bao gồm ít nhất một cờ điều kiện ràng buộc dùng cho tập con của các khối mã hóa. Mạch xử lý có cấu trúc để xác định thông tin dự đoán dùng cho tập con của các khối mã hóa dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ công có được hiện diện trong thông tin điều kiện ràng buộc hay không. Mạch xử lý có cấu trúc để khôi phục tập con của các khối mã hóa dựa trên thông tin dự đoán.

Trong phương án của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ công được xác định là được hiện diện trong thông tin điều kiện ràng buộc. Mạch xử lý có cấu trúc để xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công.

Trong phương án của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ công được xác định là không được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, mỗi cờ công chỉ báo rằng nhóm thông tin điều kiện ràng buộc tương ứng với cờ công tương ứng không được hiện diện trong thông tin điều kiện ràng buộc, và không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, thông tin điều kiện ràng buộc được hiện diện trong cú pháp mức cao đối với ít nhất một ảnh, và tập con của các khối mã hóa bao gồm các khối mã hóa trong ít nhất một ảnh.

Trong phương án của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit được mã hóa.

Các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được báo hiệu theo thứ tự định trước. Mạch xử lý có cấu trúc để giải mã - chạy các cờ điều kiện ràng buộc, và xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên các cờ điều kiện ràng buộc.

Trong phương án của sáng chế, ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công bao gồm các cờ điều kiện ràng buộc được báo hiệu theo thứ tự định trước. Mạch xử lý có cấu trúc để giải mã-chạy các cờ điều kiện ràng buộc.

Trong phương án của sáng chế, độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit được mã hóa. Đối với mỗi một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc, độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong một trong số một hoặc nhiều của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng được hiện diện trong thông tin điều kiện ràng buộc. Số lượng độ dịch được báo hiệu trong dòng bit được mã hóa.

Trong phương án của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit được mã hóa. Một hoặc nhiều phần tử cú pháp trong thông tin điều kiện ràng buộc được căn thẳng theo byte. Một hoặc nhiều phần tử cú pháp có thể bao gồm cờ công đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc và các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc.

Các khía cạnh của sáng chế cũng đề xuất phương tiện bất biến đọc được bởi máy tính mà lưu trữ các lệnh, mà khi được thực thi bởi máy tính để giải mã video làm cho máy tính thực hiện các phương pháp để giải mã video.

Mô tả vắn tắt các hình vẽ

Các đặc điểm, bản chất, và các hiệu quả khác nữa của đối tượng được bộc lộ sẽ rõ ràng hơn từ phần mô tả chi tiết sau đây và các hình vẽ kèm theo:

Fig.1A là minh họa giản lược của tập hợp con ví dụ của các chế độ nội dự đoán.

Fig.1B là minh họa của các chiều nội dự đoán ví dụ.

Fig.2 là minh họa giản lược của khối hiện tại và các ứng viên hợp nhất không gian xung quanh của khối này trong một ví dụ.

Fig.3 là minh họa giản lược của sơ đồ khối đơn giản hóa của hệ thống truyền thông 300 theo phương án của sáng chế.

Fig.4 là minh họa giản lược của sơ đồ khối đơn giản hóa của hệ thống truyền thông 400 theo phương án của sáng chế.

Fig.5 là minh họa giản lược của sơ đồ khối đơn giản hóa của bộ giải mã theo phương án của sáng chế.

Fig.6 là minh họa giản lược của sơ đồ khối đơn giản hóa của bộ mã hóa theo phương án của sáng chế.

Fig.7 thể hiện sơ đồ khối của bộ mã hóa theo phương án khác.

Fig.8 thể hiện sơ đồ khối của bộ giải mã theo phương án khác.

FIG.9 thể hiện các ví dụ để báo hiệu các tham số thay đổi độ phân giải thích nghi (ARC-adaptive resolution change) theo các phương án của sáng chế.

FIG.10 thể hiện các ví dụ về báo hiệu các tham số ARC theo các phương án của sáng chế.

FIG.11 thể hiện ví dụ về tập hợp của các phần tử cú pháp lớp ô mẫu (PTL) theo phương án của sáng chế.

FIG.12 thể hiện ví dụ về thông tin điều kiện ràng buộc chung theo phương án của sáng chế.

FIG.13A-13B thể hiện ví dụ về tập hợp của các phần tử cú pháp PTL theo phương án của sáng chế.

Các Fig.14A-14B thể hiện cấu trúc cú pháp thông tin điều kiện ràng buộc chung ví dụ theo phương án của sáng chế.

Các Fig.15A-15D thể hiện cấu trúc cú pháp thông tin điều kiện ràng buộc chung ví dụ theo phương án của sáng chế.

FIG.11 thể hiện ví dụ về tập hợp của các phần tử cú pháp PTL trong cấu trúc cú pháp PTL theo phương án của sáng chế.

FIG.17 thể hiện ví dụ về bảng cú pháp để chỉ báo thông tin mã hóa trường trong tập hợp tham số chuỗi theo phương án của sáng chế.

FIG.18 thể hiện ví dụ về bảng cú pháp để chỉ báo thông tin mã hóa trường trong tập hợp tham số video theo phương án của sáng chế.

Fig.19 thể hiện lưu đồ mô tả xử lý theo phương án của sáng chế.

FIG.20 thể hiện các ví dụ của các bảng cú pháp theo các phương án của sáng chế.

FIG.21 thể hiện lưu đồ phân tích và giải mã chu kỳ số đếm thứ tự ảnh (POC-picture order count) theo đơn vị truy nhập và giá trị đếm đơn vị truy nhập theo phương án của sáng chế.

FIG.22 thể hiện ví dụ về cấu trúc dự đoán đối với khả năng biến đổi tỷ lệ với sự thay đổi độ phân giải thích nghi theo phương án của sáng chế.

FIG.23 thể hiện ví dụ về cấu trúc dòng bit video bao gồm các ảnh con đa lớp theo phương án của sáng chế.

FIG.24 thể hiện minh họa giản lược về hiển thị của ảnh con được lựa chọn với độ phân giải được nâng cao theo phương án của sáng chế.

FIG.25 thể hiện sơ đồ khối của xử lý giải mã và hiển thị đối với dòng bit video bao gồm các ảnh con đa lớp theo phương án của sáng chế.

FIG.26 thể hiện minh họa giản lược của hiển thị video 360 với lớp tăng cường của ảnh con theo phương án của sáng chế.

FIG.27 thể hiện ví dụ về thông tin bố trí của các ảnh con và lớp tương ứng và cấu trúc dự đoán ảnh theo phương án của sáng chế.

FIG.28 thể hiện ví dụ về thông tin bố trí của các ảnh con và lớp tương ứng của chúng và cấu trúc dự đoán ảnh với phương thức khả năng biến đổi tỷ lệ không gian theo phương án của sáng chế.

Các Fig.29A-29B thể hiện các ví dụ về các bảng cú pháp đối với thông tin bố trí ảnh con theo phương án của sáng chế.

FIG.30 thể hiện ví dụ về Bảng cú pháp của bản tin SEI đối với thông tin bố trí ảnh con theo phương án của sáng chế.

FIG.31 thể hiện ví dụ về bảng cú pháp để chỉ báo các lớp đầu ra và thông tin PTL đối với tập hợp lớp đầu ra theo phương án của sáng chế.

FIG.32 thể hiện ví dụ về bảng cú pháp để chỉ báo chế độ lớp đầu ra trên tập hợp lớp đầu ra theo phương án của sáng chế.

FIG.33 thể hiện ví dụ về Bảng cú pháp để chỉ báo ảnh con được biểu diễn của mỗi lớp đối với mỗi tập hợp lớp đầu ra theo phương án của sáng chế.

FIG.34 thể hiện ví dụ về bảng cú pháp của tập hợp tham số video theo phương án của sáng chế.

FIG.35 thể hiện ví dụ về bảng cú pháp để chỉ báo tập hợp lớp đầu ra với chế độ tập hợp lớp đầu ra theo phương án của sáng chế.

Fig.36 là minh họa giản lược của hệ thống máy tính theo phương án của sáng chế.

Mô tả chi tiết sáng chế

Fig.3 minh họa sơ đồ khối đơn giản hóa của hệ thống truyền thông 300 theo phương án của sáng chế. Hệ thống truyền thông 300 bao gồm nhiều thiết bị đầu cuối mà có thể truyền thông với nhau, ví dụ, thông qua mạng 350. Ví dụ, hệ thống truyền thông 300 bao gồm cặp thiết bị đầu cuối thứ nhất 310 và 320 được liên kết thông qua mạng 350. Trong ví dụ trên Fig.3, cặp thiết bị đầu cuối thứ nhất 310 và 320 thực hiện việc truyền dữ liệu đơn hướng. Ví dụ, thiết bị đầu cuối 310 có thể mã hóa dữ liệu video (ví dụ, dòng của các ảnh video mà được quay bởi thiết bị đầu cuối 310) cho việc truyền tới thiết bị đầu cuối 320 khác thông qua mạng 350. Dữ liệu video được mã hóa có thể được truyền trong dạng của một hoặc nhiều dòng bit video được mã hóa. Thiết bị đầu cuối 320 có thể

thu dữ liệu video được mã hóa từ mạng 350, giải mã dữ liệu video được mã hóa để khôi phục các ảnh video và hiển thị các ảnh video theo dữ liệu video được khôi phục. Việc truyền dữ liệu đơn hướng có thể dùng chung trong các ứng dụng phục vụ nội dung đa phương tiện và loại tương tự.

Trong ví dụ khác, hệ thống truyền thông 300 bao gồm cặp thiết bị đầu cuối thứ hai 330 và 340 mà thực hiện việc truyền song hướng của dữ liệu video được mã hóa mà có thể xảy ra, ví dụ, trong hội nghị video. Đối với việc truyền dữ liệu song hướng, trong ví dụ của sáng chế, mỗi thiết bị đầu cuối trong số thiết bị đầu cuối 330 và 340 có thể mã hóa dữ liệu video (ví dụ, dòng của các ảnh video mà được quay bởi thiết bị đầu cuối) cho việc truyền tới thiết bị đầu cuối khác của các thiết bị đầu cuối 330 và 340 thông qua mạng 350. Mỗi thiết bị đầu cuối trong số các thiết bị đầu cuối 330 và 340 cũng có thể thu dữ liệu video được mã hóa được truyền bởi thiết bị đầu cuối còn lại trong số các thiết bị đầu cuối 330 và 340, và có thể giải mã dữ liệu video được mã hóa để khôi phục các ảnh video và có thể hiển thị các ảnh video tại thiết bị hiển thị có thể truy nhập được theo dữ liệu video được khôi phục.

Trong ví dụ trên Fig.3, các thiết bị đầu cuối 310, 320, 330 và 340 có thể được minh họa như là các máy chủ, máy tính cá nhân và điện thoại thông minh tuy nhiên các nguyên tắc của sáng chế có thể không bị giới hạn ở đó. Các phương án của sáng chế tìm kiếm ứng dụng bằng các máy tính xách tay, máy tính bảng, các máy chơi nội dung đa phương tiện và/hoặc thiết bị hội nghị video chuyên dụng. Mạng 350 biểu diễn số lượng mạng bất kỳ mà truyền dữ liệu video được mã hóa giữa các thiết bị đầu cuối 310, 320, 330 và 340, bao gồm ví dụ các mạng truyền thông đường dây (có dây) và/hoặc không dây. Mạng truyền thông 350 có thể trao đổi dữ liệu trong các kênh chuyển mạch và/hoặc các kênh chuyển gói tin. Các mạng đại diện bao gồm các mạng viễn thông, các mạng vùng cục bộ, các mạng diện rộng và/hoặc Internet. Nhằm mục đích thảo luận hiện tại, cấu trúc và cấu trúc liên kết của mạng 350 có thể không được xem xét đối với hoạt động của sáng chế trừ khi được giải thích dưới đây.

Fig.4 minh họa, như là ví dụ đối với ứng dụng cho đối tượng được bộc lộ, việc bố trí của bộ mã hóa video và bộ giải mã video trong môi trường tạo dòng. Đối tượng được bộc lộ có thể được áp dụng đồng đều tới các ứng dụng cho phép video khác, bao gồm, ví dụ, hội nghị video, TV số, lưu trữ video được nén trên nội dung đa phương tiện số bao gồm CD, DVD, thẻ nhớ và loại tương tự, và v.v.

Hệ thống tạo dòng có thể bao gồm hệ thống thu phụ 413, mà có thể bao gồm nguồn video 401, ví dụ camera số, mà tạo ra, ví dụ, dòng của các ảnh video 402 mà không được nén. Trong ví dụ của sáng chế, dòng của các ảnh video 402 bao gồm các mẫu mà được lấy bởi camera số. Dòng của các ảnh video 402, được thể hiện như đường đậm để nhấn mạnh khối lượng dữ liệu cao khi được so sánh với dữ liệu video được mã hóa 404 (hoặc các dòng video được mã hóa), có thể được xử lý bởi thiết bị điện tử 420 mà bao gồm bộ mã hóa video 403 được ghép nối tới nguồn video 401. Bộ mã hóa video 403 có thể bao gồm phần cứng, phần mềm, hoặc kết hợp của chúng để cho phép hoặc thực hiện các khía cạnh của đối tượng được bộc lộ được mô tả chi tiết dưới đây. Dữ liệu video được mã hóa 404 (hoặc dòng bit video được mã hóa 404), được ký hiệu như là đường mỏng để nhấn mạnh khối lượng dữ liệu thấp hơn khi được so sánh với dòng của các ảnh video 402, có thể được lưu trữ trên máy chủ tạo dòng 405 cho việc sử dụng trong tương lai. Một hoặc nhiều hệ thống máy khách tạo dòng phụ, như các hệ thống máy khách phụ 406 và 408 trong Fig.4 có thể truy nhập máy chủ tạo dòng 405 để truy hồi các bản sao 407 và 409 của dữ liệu video được mã hóa 404. hệ thống máy khách phụ 406 có thể bao gồm bộ giải mã video 410, ví dụ, trong thiết bị điện tử 430. Bộ giải mã video 410 giải mã bản sao đi vào 407 của dữ liệu video được mã hóa và tạo ra dòng đi ra của các ảnh video 411 mà có thể được kết xuất đồ họa trên màn ảnh 412 (ví dụ, màn ảnh) hoặc thiết bị kết xuất đồ họa khác (không được thể hiện). Trong các hệ thống tạo dòng khác, dữ liệu video được mã hóa 404, 407, và 409 (ví dụ, các dòng video) có thể được mã hóa theo các tiêu chuẩn nén/mã hóa video định trước. Các ví dụ của các tiêu chuẩn này bao gồm khuyến nghị ITU-T H.265. Trong ví dụ của sáng chế, tiêu

chuẩn mã hóa video đang phát triển được biết đến không chính thức là mã hóa video đa năng (VVC-Versatile Video Coding). Đối tượng được bộc lộ có thể được sử dụng trong ngữ cảnh của VVC.

Lưu ý rằng các thiết bị điện tử 420 và 430 có thể bao gồm các bộ phận khác (không được thể hiện). Ví dụ, thiết bị điện tử 420 có thể bao gồm bộ giải mã video (không được thể hiện) và thiết bị điện tử 430 có thể cũng bao gồm bộ mã hóa video (không được thể hiện).

Fig.5 thể hiện sơ đồ khối của bộ giải mã video 510 theo phương án của sáng chế. Bộ giải mã video 510 có thể được chứa trong thiết bị điện tử 530. Thiết bị điện tử 530 có thể bao gồm bộ thu 531 (ví dụ, hệ mạch thu). Bộ giải mã video 510 có thể được sử dụng thay thế cho bộ giải mã video 410 trong ví dụ trên Fig.4.

Bộ thu 531 có thể thu một hoặc nhiều chuỗi video được mã hóa cần được giải mã bởi bộ giải mã video 510; trong cùng phương án hoặc phương án khác, một chuỗi video được mã hóa tại thời điểm, trong đó việc giải mã của mỗi chuỗi video được mã hóa độc lập với các chuỗi video được mã hóa khác. Chuỗi video được mã hóa có thể thu được từ kênh 501, mà có thể là liên kết phần cứng/phần mềm tới thiết bị lưu trữ mà lưu trữ dữ liệu video được mã hóa. Bộ thu 531 có thể thu dữ liệu video được mã hóa với dữ liệu khác, ví dụ, dữ liệu audio được mã hóa và/hoặc các dòng dữ liệu bổ trợ, mà có thể được chuyển tiếp tới các phần tương ứng sử dụng các thực thể (không được thể hiện). Bộ thu 531 có thể tách biệt chuỗi video được mã hóa từ dữ liệu khác. Để đối phó với sự biến động mạng, bộ nhớ đệm 515 có thể được ghép nối giữa bộ thu 531 và bộ giải mã entropy / bộ phân tích 520 (sau đây gọi là "bộ phân tích 520"). Trong các ứng dụng nhất định, bộ nhớ đệm 515 là một phần của bộ giải mã video 510. Trong các ứng dụng khác, bộ nhớ đệm này có thể nằm ngoài bộ giải mã video 510 (không được thể hiện). Trong các ứng dụng khác, có thể có bộ nhớ đệm (không được thể hiện) nằm ngoài bộ giải mã video 510, ví dụ để đối phó với sự biến động mạng, và bộ nhớ đệm 515 bổ sung khác bên trong bộ giải mã video 510, chẳng hạn để xử lý thời gian phát lại. Khi bộ thu 531 đang thu dữ liệu từ thiết bị

lưu trữ/chuyển tiếp có băng thông và khả năng điều khiển đủ, hoặc từ mạng đồng bộ, bộ nhớ đệm 515 có thể không cần thiết, hoặc có thể là nhỏ. Để sử dụng tốt nhất các mạng gói tin như Internet, bộ nhớ đệm 515 có thể được yêu cầu, có thể tương đối lớn và một cách thuận lợi, có thể có kích cỡ thích nghi, và có thể ít nhất được thực hiện một phần trong hệ điều hành hoặc các thành phần tương tự (không được thể hiện) phía ngoài của bộ giải mã video 510.

Bộ giải mã video 510 có thể bao gồm bộ phân tích 520 để khôi phục các ký tự 521 từ chuỗi video được mã hóa. Các hạng mục của các ký tự này bao gồm thông tin được sử dụng để quản lý hoạt động của bộ giải mã video 510, và thông tin khả năng để điều khiển thiết bị kết xuất đồ họa như thiết bị kết xuất đồ họa 512 (ví dụ, màn hiển thị) mà không phải là phần liên khối của thiết bị điện tử 530 mà có thể được ghép nối tới thiết bị điện tử 530, như được thể hiện trên Fig.5. Thông tin điều khiển đối với (các) thiết bị kết xuất đồ họa có thể nằm trong dạng của Thông tin tăng cường bổ sung (các bản tin SEI) hoặc các phân đoạn tập hợp tham số thông tin khả năng sử dụng video (VUI-Video Usability Information) (không được thể hiện). Bộ phân tích 520 có thể phân tích/giải mã entropy chuỗi video được mã hóa thu được. Việc mã hóa chuỗi video được mã hóa có thể tuân theo kỹ thuật hoặc tiêu chuẩn mã hóa video, và có thể tuân theo các nguyên tắc khác nhau, bao gồm mã hóa độ dài biến thiên, mã hóa Huffman, mã hóa số học có hoặc không có cảm nhận ngữ cảnh, và v.v. Bộ phân tích 520 có thể tách từ chuỗi video được mã hóa, tập hợp của các tham số nhóm con đối với ít nhất một trong số các nhóm con của các điểm ảnh trong bộ giải mã video, dựa trên ít nhất một tham số tương ứng với nhóm. Các nhóm con có thể bao gồm các Nhóm của các ảnh (GOPs), các ảnh, các ô, các lát, các khối macrô, các đơn vị mã hóa (CU-Coding Unit), các khối, các đơn vị biến đổi (TU-Transform Unit), các đơn vị dự đoán (PU-Prediction Unit) và v.v. Bộ phân tích 520 có thể cũng tách từ chuỗi video được mã hóa thông tin như các hệ số biến đổi, các giá trị tham số lượng tử hóa, các vectơ chuyển động, và v.v.

Bộ phân tích 520 có thể thực hiện hoạt động giải mã entropy/phân tích trên chuỗi video thu được từ bộ nhớ đệm 515, để tạo ra các ký tự 521.

Việc khôi phục của các ký tự 521 có thể liên quan đến nhiều đơn vị khác nhau phụ thuộc vào loại của ảnh video được mã hóa hoặc các thành phần của chúng (như: ảnh nội bộ và liên đới, khối nội bộ và liên đới), và các yếu tố khác. Các đơn vị nào được liên quan, và làm thế nào, có thể được điều khiển bởi thông tin điều khiển nhóm con mà đã được phân tích từ chuỗi video được mã hóa bởi bộ phân tích 520. Dòng của thông tin điều khiển nhóm con giữa bộ phân tích 520 và các bộ phận dưới đây không được thể hiện nhằm mục đích rõ ràng.

Ngoài các khối chức năng đã được đề cập, bộ giải mã video 510 có thể được phân chia về mặt khái niệm thành các bộ phận chức năng như được mô tả dưới đây. Trong vận hành thực hiện thực tế theo các điều kiện thương mại, nhiều bộ phận này tương tác khăng khít với nhau và có thể, ít nhất một phần, được tích hợp vào nhau. Tuy nhiên, nhằm mục đích mô tả đối tượng được bộc lộ, việc phân chia theo khái niệm thành các bộ phận chức năng dưới đây là thích hợp.

Bộ phận thứ nhất là bộ biến đổi tỷ lệ/bộ biến đổi ngược 551. Bộ biến đổi tỷ lệ/bộ biến đổi ngược 551 thu hệ số biến đổi được lượng tử hóa cũng như thông tin điều khiển, bao gồm biến đổi nào được sử dụng, kích cỡ khối, hệ số lượng tử hóa, các ma trận biến đổi tỷ lệ lượng tử hóa, v.v như các ký tự 521 từ bộ phân tích 520. Bộ biến đổi tỷ lệ/bộ biến đổi ngược 551 có thể xuất ra các khối bao gồm các giá trị mẫu mà có thể được đưa vào bộ kết hợp 555.

Trong một vài trường hợp, các mẫu đầu ra của bộ biến đổi tỷ lệ/bộ biến đổi ngược 551 có thể thuộc về khối được mã hóa nội bộ; tức là: khối mà đang không sử dụng thông tin dự đoán từ các ảnh được khôi phục trước đó, mà có thể sử dụng thông tin dự đoán từ các phần được khôi phục trước đó của ảnh hiện tại. Thông tin dự đoán này có thể được cấp bởi bộ dự đoán nội ảnh 552. Trong một vài trường hợp, bộ dự đoán nội ảnh 552 tạo ra khối có cùng dạng và kích cỡ của khối đang được khôi phục, nhờ sử dụng thông tin đã được khôi phục xung quanh được truy xuất từ bộ đệm ảnh hiện tại 558. Bộ đệm ảnh hiện tại 558 lưu trữ đệm, ví dụ, ảnh hiện tại được khôi phục một phần và/hoặc ảnh hiện tại được khôi phục toàn phần. Bộ kết hợp 555, trong một vài trường hợp, cộng, trên

cơ sở mẫu, thông tin dự đoán mà bộ nội dự đoán 552 đã tạo ra với thông tin mẫu đầu ra như được cấp bởi bộ biến đổi tỷ lệ/bộ biến đổi ngược 551.

Trong các trường hợp khác, các mẫu đầu ra của của bộ biến đổi tỷ lệ/bộ biến đổi ngược 551 có thể thuộc về khối có khả năng được bù chuyển chuyển động và được mã hóa liên đới. Trong trường hợp này, bộ dự đoán bù chuyển động 553 có thể truy nhập bộ nhớ ảnh tham chiếu 557 để truy xuất các mẫu được sử dụng cho việc dự đoán. Sau khi bù chuyển động các mẫu được truy xuất theo các ký tự 521 thuộc về khối, các mẫu này có thể được bổ sung bởi bộ kết hợp 555 tới đầu ra của bộ biến đổi tỷ lệ/bộ biến đổi ngược 551 (trong trường hợp này gọi là các mẫu dư hoặc tín hiệu dư) để tạo ra thông tin mẫu đầu ra. Các địa chỉ bộ nhớ ảnh tham chiếu 557 từ trong đó bộ dự đoán bù chuyển động 553 truy xuất các mẫu dự đoán có thể được điều khiển bởi các vector chuyển động, khả dụng đối với bộ dự đoán bù chuyển động 553 trong dạng của các ký tự 521 mà có thể có, ví dụ X, Y, và các thành phần ảnh tham chiếu. Việc bù chuyển động cũng có thể bao gồm việc nội suy các giá trị mẫu như được truy xuất từ bộ nhớ ảnh tham chiếu 557 khi các vector chuyển động chính xác mẫu con được sử dụng, các cơ chế dự đoán vector chuyển động, và v.v.

Các mẫu đầu ra của bộ kết hợp 555 có thể được đưa vào các kỹ thuật lọc vòng khác nhau trong bộ lọc vòng 556. Các kỹ thuật nén video có thể bao gồm các kỹ thuật lọc vòng trong mà được điều khiển bởi các tham số được chứa trong chuỗi video được mã hóa (cũng gọi là dòng bit được mã hóa) và khả dụng đối với bộ lọc vòng 556 như các ký tự 521 từ bộ phân tích 520, nhưng có thể cũng phản hồi lại siêu thông tin thu được trong khi giải mã các phần trước đó (trong thứ tự giải mã) của ảnh được mã hóa hoặc chuỗi video được mã hóa, cũng như phản hồi lại các giá trị mẫu được lọc vòng và được khôi phục trước đó.

Đầu ra của bộ lọc vòng 556 có thể là dòng mẫu mà có thể được xuất ra tới thiết bị kết xuất đồ họa 512 cũng như được lưu trữ trong bộ nhớ ảnh tham chiếu 557 để sử dụng trong việc dự đoán liên ảnh tương lai.

Các ảnh được mã hóa định trước, được khôi phục đầy đủ một lần, có thể được sử dụng như là các ảnh tham chiếu cho việc dự đoán tương lai. Ví dụ,

một khi ảnh được mã hóa tương ứng với ảnh hiện tại được khôi phục đầy đủ và ảnh được mã hóa đã được nhận dạng là ảnh tham chiếu (ví dụ, bởi bộ phân tích 520), bộ đệm ảnh hiện tại 558 có thể trở thành một phần của bộ nhớ ảnh tham chiếu 557, và bộ đệm ảnh hiện tại mới có thể được bố trí lại trước khi bắt đầu việc khôi phục của ảnh được mã hóa tiếp theo.

Bộ giải mã video 510 có thể thực hiện các hoạt động giải mã theo kỹ thuật nén video định trước trong tiêu chuẩn, như khuyến nghị ITU-T H.265. Chuỗi video được mã hóa có thể tuân theo cú pháp được chỉ rõ bởi tiêu chuẩn hoặc kỹ thuật nén video đang được sử dụng, trong trường hợp mà chuỗi video được mã hóa tuân theo cả cú pháp của tiêu chuẩn hoặc kỹ thuật nén video và các mẫu như được đề cập trong tiêu chuẩn hoặc kỹ thuật nén video. Cụ thể, mẫu có thể lựa chọn các công cụ định trước như là chỉ các công cụ khả dụng để sử dụng theo mẫu này từ tất cả các công cụ khả dụng trong kỹ thuật hoặc tiêu chuẩn nén video. Ngoài ra, sự cần thiết đối với tính tương thích có thể là độ phức tạp của chuỗi video được mã hóa nằm trong các giới hạn như được xác định bởi mức của kỹ thuật hoặc tiêu chuẩn nén video. Trong một vài trường hợp, các mức giới hạn kích cỡ ảnh lớn nhất, tốc độ khung lớn nhất, tốc độ mẫu khôi phục lớn nhất (được đo lường trong, ví dụ các siêu mẫu trên giây), kích cỡ ảnh tham chiếu lớn nhất, và v.v. Các giới hạn được thiết lập bởi các mức có thể, trong một vài trường hợp, còn được giới hạn thông qua các tiêu chuẩn kỹ thuật của bộ giải mã tham chiếu giả thiết (HRD-Hypothetical Reference Decoder) và siêu dữ liệu đối với việc quản lý bộ đệm HRD được báo hiệu trong chuỗi video được mã hóa.

Trong phương án của sáng chế, bộ thu 531 có thể thu dữ liệu bổ sung (dư thừa) với video được mã hóa. Dữ liệu bổ sung có thể được bao gồm như là một phần của (các) chuỗi video được mã hóa. Dữ liệu bổ sung có thể được sử dụng bởi bộ giải mã video 510 để giải mã một cách thích hợp dữ liệu và/hoặc khôi phục chính xác hơn dữ liệu video gốc. Dữ liệu bổ sung có thể nằm trong dạng của, ví dụ, các lớp nâng cao tỷ lệ tín hiệu trên tạp âm (SNR) hoặc theo thời gian, không gian, các lát dư thừa, các ảnh dư thừa, các mã sửa lỗi tiến, và loại tương tự.

Fig.6 thể hiện sơ đồ khối của bộ mã hóa video 603 theo phương án của sáng chế. Bộ mã hóa video 603 được chứa trong thiết bị điện tử 620. Thiết bị điện tử 620 bao gồm bộ truyền 640 (ví dụ, hệ mạch truyền). Bộ mã hóa video 603 có thể được sử dụng trong vị trí của bộ mã hóa video 403 trong ví dụ trên Fig.4.

Bộ mã hóa video 603 có thể thu các mẫu video từ nguồn video 601 (mà không phải một phần của thiết bị điện tử 620 trong ví dụ trên Fig.6) mà có thể chụp các ảnh video cần được mã hóa bởi bộ mã hóa video 603. Trong ví dụ khác, nguồn video 601 là một phần của thiết bị điện tử 620.

Nguồn video 601 có thể cấp chuỗi video nguồn cần được mã hóa bởi bộ mã hóa video 603 trong dạng của dòng mẫu video số mà có thể có độ sâu bit thích hợp bất kỳ (ví dụ: 8 bit, 10 bit, 12 bit, ...), bất kỳ không gian màu (ví dụ, BT.601 Y CrCb, RGB, ...), và bất kỳ cấu trúc lấy mẫu thích hợp (ví dụ Y CrCb 4:2:0, Y CrCb 4:4:4). Trong hệ thống phục vụ nội dung đa phương tiện, nguồn video 601 có thể là thiết bị lưu trữ mà lưu trữ video được chuẩn bị trước. Trong hệ thống hội nghị video, nguồn video 601 có thể là camera mà thu thông tin ảnh cục bộ như là chuỗi video. Dữ liệu video có thể được cấp như là các ảnh riêng biệt mà truyền chuyển động khi được quan sát trong chuỗi. Bản thân các ảnh có thể được sắp xếp như là mảng không gian của các điểm ảnh, trong đó mỗi điểm ảnh có thể bao gồm một hoặc nhiều mẫu phụ thuộc vào cấu trúc lấy mẫu, không gian màu, v.v được sử dụng. Người có hiểu biết trung bình trong lĩnh vực kỹ thuật có thể hiểu được quan hệ giữa các điểm ảnh và các mẫu. Phần mô tả dưới đây tập trung vào các mẫu.

Theo phương án của sáng chế, bộ mã hóa video 603 có thể mã hóa và nén các ảnh của chuỗi video nguồn thành chuỗi video được mã hóa 643 trong thời gian thực hoặc theo bất kỳ điều kiện thời gian khác như được yêu cầu bởi ứng dụng. Cường ép tốc độ mã hóa thích hợp là một chức năng của bộ điều khiển 650. Trong một vài phương án, bộ điều khiển 650 điều khiển các bộ phận chức năng khác như được mô tả dưới đây và được ghép nối chức năng tới các bộ phận chức năng khác. Việc ghép nối không được thể hiện nhằm mục đích rõ

ràng. Các tham số được thiết lập bởi bộ điều khiển 650 có thể bao gồm các tham số liên quan đến điều khiển tỷ lệ (bỏ qua ảnh, bộ lượng tử hóa, giá trị lam-da của các kỹ thuật tối ưu hóa méo tỷ lệ, ...), kích cỡ ảnh, cấu trúc nhóm của các ảnh (GOP), phạm vi tìm kiếm vectơ chuyển động lớn nhất, và v.v. Bộ điều khiển 650 có thể có cấu trúc để có các chức năng thích hợp khác mà thuộc về bộ mã hóa video 603 được tối ưu hóa cho thiết kế hệ thống nhất định.

Trong một vài phương án, bộ mã hóa video 603 có cấu trúc để hoạt động trong vòng mã hóa. Theo phân mô tả được đơn giản hóa, trong ví dụ của sáng chế, vòng mã hóa có thể bao gồm bộ mã hóa nguồn 630 (ví dụ, chịu trách nhiệm tạo ra các ký tự, như dòng ký tự, dựa trên ảnh đầu vào cần được mã hóa, và các ảnh tham chiếu), và bộ giải mã (cục bộ) 633 được gắn trong bộ mã hóa video 603. Bộ giải mã 633 khôi phục các ký tự để tạo ra dữ liệu mẫu trong cách thức tương tự như bộ giải mã (tù xa) tạo ra (do bất kỳ việc nén giữa các ký tự và dòng bit video được mã hóa là không tổn hao trong các kỹ thuật nén video được xem xét trong đối tượng được bộc lộ). Dòng mẫu được khôi phục (dữ liệu mẫu) được đưa vào bộ nhớ ảnh tham chiếu 634. Khi việc giải mã của dòng ký tự dẫn tới các kết quả chính xác bit độc lập với vị trí bộ giải mã (cục bộ hoặc từ xa), nội dung trong bộ nhớ ảnh tham chiếu 634 cũng là chính xác bit giữa bộ mã hóa cục bộ và bộ mã hóa từ xa. Nói cách khác, phần dự đoán của bộ mã hóa “thấy được” như là các mẫu ảnh tham chiếu một cách chính xác tương tự các giá trị mẫu khi bộ giải mã “thấy được” khi sử dụng việc dự đoán trong khi giải mã. Nguyên tắc cơ bản này của đồng bộ ảnh tham chiếu (và độ trượt hệ quả, nếu tính đồng bộ không thể được duy trì, ví dụ do các lỗi kênh) cũng được sử dụng trong một vài kỹ thuật liên quan.

Hoạt động của bộ giải mã “cục bộ” 633 có thể tương tự như của bộ giải mã “tù xa”, như bộ giải mã video 510, mà đã được mô tả chi tiết nêu trên kết hợp với Fig.5. Tuy nhiên, viện dẫn ngắn gọn tới Fig.5, khi các ký tự là khả dụng và việc mã hóa/giải mã của các ký tự đối với chuỗi video được mã hóa bởi bộ mã hóa entropy 645 và bộ phân tích 520 có thể là không tổn hao, các phần giải mã entropy decoding của bộ giải mã video 510, bao gồm bộ nhớ đệm 515,

và bộ phân tích 520 có thể không được thực hiện đầy đủ trong bộ giải mã cục bộ 633.

Việc quan sát có thể được thực hiện tại điểm này đó là bất kỳ kỹ thuật bộ giải mã ngoại trừ việc phân tích/giải mã entropy mà được hiện diện trong bộ giải mã cũng cần được hiện diện, về cơ bản trong dạng chức năng đồng nhất, trong bộ mã hóa tương ứng. Vì lý do này, đối tượng được bộc lộ tập trung vào hoạt động bộ giải mã. Phần mô tả của các kỹ thuật bộ mã hóa có thể được rút gọn do chúng ngược với các kỹ thuật bộ giải mã đã được mô tả toàn diện. Chỉ trong các vùng nhất định, phần mô tả chi tiết hơn được yêu cầu và được mô tả dưới đây.

Trong khi hoạt động, trong một vài ví dụ, bộ mã hóa nguồn 630 có thể thực hiện mã hóa dự đoán bù chuyển động, mã mã hóa ảnh đầu vào theo cách dự đoán có viện dẫn tới một hoặc nhiều ảnh được mã hóa trước đó từ chuỗi video mà được chỉ định là “các ảnh tham chiếu”. Theo cách thức này, cơ cấu mã hóa 632 mã hóa các độ chênh lệch giữa các khối điểm ảnh của ảnh đầu vào và các khối điểm ảnh của các ảnh tham chiếu mà có thể được lựa chọn như là các tham chiếu dự đoán đối với ảnh đầu vào.

Bộ giải mã video cục bộ 633 có thể giải mã dữ liệu video được mã hóa của các ảnh mà có thể được chỉ định là các ảnh tham chiếu, dựa trên các ký tự đã được tạo ra bởi bộ mã hóa nguồn 630. Các hoạt động của cơ cấu mã hóa 632 có thể có ưu điểm là các xử lý tổn hao. Khi dữ liệu video được mã hóa có thể được giải mã tại bộ giải mã video (không được thể hiện trong FIG.6), chuỗi video được khôi phục thường có thể là bản sao của chuỗi video nguồn với một vài lỗi. Bộ giải mã video cục bộ 633 sao chép các xử lý giải mã mà có thể được thực hiện bởi bộ giải mã video trên các ảnh tham chiếu và có thể làm cho các ảnh tham chiếu được khôi phục được lưu trữ trong bộ lưu trữ đệm ảnh tham chiếu 634. Theo cách thức này, bộ mã hóa video 603 có thể lưu trữ các bản sao của các ảnh tham chiếu được khôi phục cục bộ mà có nội dung chung như các ảnh tham chiếu được khôi phục mà sẽ thu được bởi bộ giải mã video phía xa (không có các lỗi truyền).

Bộ dự đoán 635 có thể thực hiện các tìm kiếm dự đoán đối với cơ cấu mã hóa 632. Tức là, đối với ảnh mới cần được mã hóa, bộ dự đoán 635 có thể tìm kiếm bộ nhớ ảnh tham chiếu 634 đối với dữ liệu mẫu (như các khối điểm ảnh tham chiếu ứng viên) hoặc siêu dữ liệu định trước như các vector chuyển động ảnh tham chiếu, các dạng khối, và v.v, mà có thể đóng vai trò là tham chiếu dự đoán thích hợp đối với các ảnh mới. Bộ dự đoán 635 có thể hoạt động trên cơ sở mẫu điểm ảnh đến mẫu khối để tìm kiếm các tham chiếu dự đoán thích hợp. Trong một vài trường hợp, như được xác định bởi các kết quả tìm kiếm thu được bởi bộ dự đoán 635, ảnh đầu vào có thể có các tham chiếu dự đoán được rút ra từ các ảnh tham chiếu được lưu trữ trong bộ nhớ ảnh tham chiếu 634.

Bộ điều khiển 650 có thể quản lý các hoạt động mã hóa của bộ mã hóa nguồn 630, bao gồm, ví dụ, thiết lập các tham số và các tham số nhóm con được sử dụng để mã hóa dữ liệu video.

Đầu ra của tất cả các bộ phận chức năng nêu trên có thể được mã hóa entropy trong bộ mã hóa entropy 645. Bộ mã hóa entropy 645 dịch các ký tự như được tạo ra bởi các bộ phận chức năng khác nhau thành chuỗi video được mã hóa, bằng cách nén không tổn hao các ký tự theo các kỹ thuật như mã hóa Huffman, mã hóa độ dài biến thiên, mã hóa số học, và v.v.

Bộ truyền 640 có thể lưu trữ đệm (các) chuỗi video được mã hóa như được tạo ra bởi bộ mã hóa entropy 645 để chuẩn bị cho việc truyền thông qua kênh truyền thông 660, mà có thể là liên kết phần cứng/phần mềm tới thiết bị lưu trữ mà sẽ lưu trữ dữ liệu video được mã hóa. Bộ truyền 640 có thể hợp nhất dữ liệu video được mã hóa từ bộ mã hóa video 603 với dữ liệu khác cần được truyền, ví dụ, dữ liệu audio được mã hóa và/hoặc các dòng dữ liệu bổ trợ (các nguồn không được thể hiện).

Bộ điều khiển 650 có thể quản lý hoạt động của bộ mã hóa video 603. Trong khi mã hóa, bộ điều khiển 650 có thể gán tới mỗi ảnh được mã hóa loại ảnh được mã hóa định trước, mà có thể ảnh hưởng tới các kỹ thuật mã hóa mà

có thể được áp dụng tới ảnh tương ứng. Ví dụ, các ảnh thường có thể được gán như là một trong số các loại ảnh sau đây:

Ảnh nội dự đoán (ảnh I) có thể là ảnh mà có thể được mã hóa và giải mã mà không sử dụng bất kỳ ảnh khác trong chuỗi như là nguồn dự đoán. Một vài bộ mã hóa-giải mã video cho phép các loại ảnh nội dự đoán khác nhau, bao gồm, ví dụ các ảnh làm mới bộ giải mã độc lập (“IDR-Independent Decoder Refresh”). Người có hiểu biết trung bình trong lĩnh vực kỹ thuật sẽ nhận ra các biến thể của các ảnh I và các ứng dụng và các đặc tính tương ứng của chúng.

Ảnh dự đoán (ảnh P) có thể là ảnh mà có thể được mã hóa và giải mã nhờ sử dụng việc nội dự đoán hoặc việc dự đoán liên đới nhờ sử dụng nhiều nhất một vectơ chuyển động và chỉ số tham chiếu để dự đoán các giá trị mẫu của mỗi khối.

Ảnh dự đoán hai chiều (ảnh B) có thể là ảnh mà có thể được mã hóa và được giải mã nhờ sử dụng việc nội dự đoán hoặc việc dự đoán liên đới nhờ sử dụng nhiều nhất hai vectơ chuyển động và các chỉ số tham chiếu để dự đoán các giá trị mẫu của mỗi khối. Tương tự, các ảnh đa dự đoán có thể sử dụng nhiều hơn hai ảnh tham chiếu và siêu dữ liệu được kết hợp cho việc khôi phục khối đơn.

Các ảnh nguồn thường có thể được phân chia về mặt không gian thành nhiều khối mẫu (ví dụ, các khối của các mẫu 4x4, 8x8, 4x8, hoặc 16x16) và được mã hóa trên cơ sở lần lượt từng khối. Các khối có thể được mã hóa dự đoán có viện dẫn tới các khối khác (đã được mã hóa) như được xác định bởi việc gán mã hóa được áp dụng tới các ảnh tương ứng của các khối. Ví dụ, các khối của các ảnh I có thể được mã hóa không dự đoán hoặc chúng có thể được mã hóa dự đoán có viện dẫn tới các khối đã được mã hóa của cùng ảnh (dự đoán không gian hoặc nội dự đoán). Các khối điểm ảnh của các ảnh P có thể được mã hóa dự đoán, thông qua dự đoán không gian hoặc thông qua dự đoán theo thời gian có viện dẫn tới một ảnh tham chiếu được mã hóa trước đó. Các khối của các ảnh B có thể được mã hóa dự đoán, thông qua dự đoán theo không gian hoặc

thông qua dự đoán theo thời gian có viện dẫn tới một hoặc hai ảnh tham chiếu được mã hóa trước đó.

Bộ mã hóa video 603 có thể thực hiện các hoạt động mã hóa theo kỹ thuật hoặc tiêu chuẩn mã hóa video định trước, như Khuyến nghị ITU-T H.265. Trong hoạt động của bộ mã hóa video 603, bộ mã hóa video 603 có thể thực hiện các hoạt động nén khác nhau, bao gồm các hoạt động mã hóa dự đoán mà khai thác các dư thừa về mặt không gian và thời gian trong chuỗi video đầu vào. Do đó, dữ liệu video được mã hóa, có thể tuân theo cú pháp được chỉ rõ bởi kỹ thuật hoặc tiêu chuẩn mã hóa video đang được sử dụng.

Trong phương án của sáng chế, bộ truyền 640 có thể truyền dữ liệu bổ sung với video được mã hóa. Bộ mã hóa nguồn 630 có thể bao gồm dữ liệu này như là một phần của chuỗi video được mã hóa. Dữ liệu bổ sung có thể bao gồm các lớp nâng cao SNR/theo thời gian/không gian, các dạng khác của dữ liệu dư thừa như các ảnh và các lát dư thừa, các bản tin SEI, các phân đoạn tập hợp tham số VUI, và v.v.

Video có thể được quay như là các ảnh nguồn (các ảnh video) trong chuỗi thời gian. Việc dự đoán nội ảnh (thường gọi tắt là việc nội dự đoán) tận dụng sự tương quan về không gian trong ảnh định trước, và dự đoán liên ảnh tận dụng sự tương quan (theo thời gian hoặc theo cách thức khác) giữa các ảnh. Trong ví dụ của sáng chế, ảnh cụ thể đang được mã hóa/giải mã, mà được gọi là ảnh hiện tại, được phân chia thành các khối. Khi khối trong ảnh hiện tại tương tự như khối tham chiếu trong ảnh tham chiếu vẫn đang được lưu trữ đệm và được mã hóa trước trong video, khối trong ảnh hiện tại có thể được mã hóa bởi vector mà được gọi là vector chuyển động. Vector chuyển động trỏ tới khối tham chiếu trong ảnh tham chiếu, và có thể có chiều thứ ba mà nhận dạng ảnh tham chiếu, trong trường hợp các ảnh tham chiếu đang được sử dụng.

Trong một vài phương án, kỹ thuật dự đoán hai chiều có thể được sử dụng trong dự đoán liên ảnh. Theo kỹ thuật dự đoán hai chiều, hai ảnh tham chiếu, như ảnh tham chiếu thứ nhất và ảnh tham chiếu thứ hai mà đều đứng trước trong thứ tự giải mã đối với ảnh hiện tại trong video (nhưng có thể nằm

trong quá khứ và tương lai, một cách lần lượt, trong thứ tự hiển thị) được sử dụng. Khối trong ảnh hiện tại có thể được mã hóa bởi vectơ chuyển động thứ nhất mà trở tới khối tham chiếu thứ nhất trong ảnh tham chiếu thứ nhất, và vectơ chuyển động thứ hai mà trở tới khối tham chiếu thứ hai trong ảnh tham chiếu thứ hai. Khối có thể được dự đoán bởi kết hợp của khối tham chiếu thứ nhất và khối tham chiếu thứ hai.

Ngoài ra, kỹ thuật chế độ hợp nhất có thể được sử dụng trong dự đoán liên ảnh để cải thiện hiệu quả mã hóa.

Theo một vài phương án của sáng chế, các dự đoán, như các dự đoán liên ảnh và các dự đoán nội ảnh được thực hiện trong đơn vị của các khối. Ví dụ, theo tiêu chuẩn HEVC, ảnh trong chuỗi của các ảnh video được phân chia thành các đơn vị cây mã hóa (CTU) cho việc nén, các CTU trong ảnh có cùng kích cỡ, như các điểm ảnh 64×64 , các điểm ảnh 32×32 , hoặc các điểm ảnh 16×16 . Nói chung, CTU bao gồm ba khối cây mã hóa (CTB-coding tree block), mà là một CTB độ chói và hai CTB sắc độ. Mỗi CTU có thể được phân chia cây tứ phân đệ quy thành một hoặc nhiều đơn vị mã hóa (CU-coding unit). Ví dụ, CTU của điểm ảnh 64×64 có thể được phân chia thành một CU của điểm ảnh 64×64 , hoặc 4 CU của điểm ảnh 32×32 , hoặc 16 CU của điểm ảnh 16×16 . Trong ví dụ của sáng chế, mỗi CU được phân tích để xác định loại dự đoán đối với CU, như loại dự đoán liên đới hoặc loại nội dự đoán. CU được phân chia thành một hoặc nhiều đơn vị dự đoán (PU-prediction unit) phụ thuộc vào khả năng dự đoán theo không gian và/hoặc thời gian. Nói chung, mỗi PU bao gồm khối dự đoán (PB-prediction block) độ chói, và hai PB sắc độ. Trong phương án của sáng chế, hoạt động dự đoán trong việc mã hóa (mã hóa/giải mã) được thực hiện trong đơn vị của khối dự đoán. Việc sử dụng khối dự đoán sắc độ như ví dụ về khối dự đoán, khối dự đoán bao gồm ma trận của các giá trị (ví dụ, các giá trị độ chói) đối với các điểm ảnh, như điểm ảnh 8×8 , điểm ảnh 16×16 , điểm ảnh 8×16 , điểm ảnh 16×8 , và loại tương tự.

Fig.7 thể hiện sơ đồ của bộ mã hóa video 703 theo phương án khác của sáng chế. Bộ mã hóa video 703 có cấu trúc để thu khối xử lý (ví dụ, khối dự

đoán) của các giá trị mẫu trong ảnh video hiện tại trong chuỗi của các ảnh video, và mã hóa khối xử lý thành ảnh được mã hóa mà là một phần của chuỗi video được mã hóa. Trong ví dụ của sáng chế, bộ mã hóa video 703 được sử dụng trong vị trí của bộ mã hóa video 403 trong ví dụ trên Fig.4.

Trong ví dụ HEVC, bộ mã hóa video 703 thu ma trận của các giá trị mẫu đối với khối xử lý, như khối dự đoán của các mẫu 8×8 , và loại tương tự. Bộ mã hóa video 703 xác định rằng khối xử lý có được mã hóa tốt nhất hay không sử dụng chế độ nội dự đoán, chế độ dự đoán liên đới, hoặc chế độ dự đoán hai chiều sử dụng, ví dụ, tối ưu hóa tỷ lệ méo. Khi khối xử lý cần được mã hóa trong chế độ nội dự đoán, bộ mã hóa video 703 có thể sử dụng kỹ thuật nội dự đoán để mã hóa khối xử lý thành ảnh được mã hóa; và khi khối xử lý cần được mã hóa trong chế độ dự đoán liên đới hoặc chế độ dự đoán hai chiều, bộ mã hóa video 703 có thể sử dụng việc dự đoán liên đới hoặc kỹ thuật dự đoán hai chiều, một cách lần lượt, để mã hóa khối xử lý thành ảnh được mã hóa. Trong các kỹ thuật mã hóa video định trước, chế độ hợp nhất có thể là chế độ con dự đoán ảnh liên đới trong đó vectơ chuyển động thu được từ một hoặc nhiều biến dự đoán vectơ chuyển động mà không có ưu điểm của thành phần vectơ chuyển động được mã hóa phía ngoài các biến dự đoán. Trong các kỹ thuật mã hóa video khác định trước, thành phần vectơ chuyển động có thể áp dụng tới khối mục tiêu có thể được biểu diễn. Trong ví dụ của sáng chế, bộ mã hóa video 703 bao gồm các thành phần khác, như môđun quyết định chế độ (không được thể hiện) để xác định chế độ của các khối xử lý.

Trong ví dụ trên FIG.7, bộ mã hóa video 703 bao gồm bộ mã hóa liên đới 730, bộ mã hóa nội dự đoán 722, bộ tính toán phần dư 723, bộ chuyển đổi 726, bộ mã hóa phần dư 724, bộ điều khiển chung 721, và bộ mã hóa entropy 725 được ghép nối với nhau như được thể hiện trên FIG.7.

Bộ mã hóa liên đới 730 có cấu trúc để thu các mẫu của khối hiện tại (ví dụ, khối xử lý), so sánh khối đối với một hoặc nhiều khối tham chiếu trong các ảnh tham chiếu (ví dụ, các khối trong các ảnh trước và các ảnh sau), tạo ra thông tin dự đoán liên đới (ví dụ, phần mô tả về thông tin dư thừa theo kỹ thuật mã hóa

liên đới, các vector chuyển động, thông tin chế độ hợp nhất), và tính toán các kết quả dự đoán liên đới (ví dụ, khối được dự đoán) dựa trên thông tin dự đoán liên đới nhờ sử dụng kỹ thuật thích hợp. Trong một vài ví dụ, các ảnh tham chiếu là các ảnh tham chiếu được giải mã mà được giải mã dựa trên thông tin video được mã hóa.

Bộ mã hóa nội dự đoán 722 có cấu trúc để thu các mẫu của khối hiện tại (ví dụ, khối xử lý), trong một vài trường hợp so sánh lần lượt từng khối đã được mã hóa trong cùng ảnh, tạo ra các hệ số biến được lượng tử hóa sau khi biến đổi, và trong một vài trường hợp còn là thông tin nội dự đoán (ví dụ, thông tin chiều nội dự đoán theo một hoặc nhiều kỹ thuật mã hóa nội dự đoán). Trong ví dụ của sáng chế, bộ mã hóa nội dự đoán 722 cũng tính toán các kết quả nội dự đoán (ví dụ, khối được dự đoán) dựa trên thông tin nội dự đoán và các khối tham chiếu trong cùng ảnh.

Bộ điều khiển chung 721 có cấu trúc để xác định dữ liệu điều khiển chung và điều khiển các thành phần khác của bộ mã hóa video 703 dựa trên dữ liệu điều khiển chung. Trong ví dụ của sáng chế, bộ điều khiển chung 721 xác định chế độ của khối, và cấp tín hiệu điều khiển tới bộ chuyển đổi 726 dựa trên chế độ. Ví dụ, khi chế độ là chế độ nội dự đoán, bộ điều khiển chung 721 điều khiển bộ chuyển đổi 726 để lựa chọn kết quả chế độ nội dự đoán để sử dụng bởi bộ tính toán phần dư 723, và điều khiển bộ mã hóa entropy 725 để lựa chọn thông tin nội dự đoán và bao gồm thông tin nội dự đoán trong dòng bit; và khi chế độ là chế độ dự đoán liên đới, bộ điều khiển chung 721 điều khiển bộ chuyển đổi 726 để lựa chọn kết quả dự đoán liên đới để sử dụng bởi bộ tính toán phần dư 723, và điều khiển bộ mã hóa entropy 725 để lựa chọn thông tin dự đoán liên đới và bao gồm thông tin dự đoán liên đới trong dòng bit.

Bộ tính toán phần dư 723 có cấu trúc để tính toán độ chênh lệch (dữ liệu phần dư) giữa khối thu được và các kết quả dự đoán được lựa chọn từ bộ mã hóa nội dự đoán 722 hoặc bộ mã hóa liên đới 730. Bộ mã hóa phần dư 724 có cấu trúc để hoạt động dựa trên dữ liệu phần dư để mã hóa dữ liệu phần dư để tạo ra các hệ số biến đổi. Trong ví dụ của sáng chế, bộ mã hóa phần dư 724 có cấu

trúc để chuyển đổi dữ liệu phần dư từ miền không gian thành miền tần số, và tạo ra các hệ số biến đổi. Sau đó, các hệ số biến đổi được đưa vào xử lý lượng tử hóa để thu nhận các hệ số biến đổi được lượng tử hóa. Trong các phương án khác nhau, bộ mã hóa video 703 cũng bao gồm bộ giải mã phần dư 728. Bộ giải mã phần dư 728 có cấu trúc để thực hiện việc biến đổi ngược, và tạo ra dữ liệu phần dư được giải mã. Dữ liệu phần dư được giải mã có thể được sử dụng một cách thích hợp bởi bộ mã hóa nội dự đoán 722 và bộ mã hóa liên đới 730. Ví dụ, bộ mã hóa liên đới 730 có thể tạo ra các khối được giải mã dựa trên dữ liệu phần dư được giải mã và thông tin dự đoán liên đới, và bộ mã hóa nội dự đoán 722 có thể tạo ra các khối được giải mã dựa trên dữ liệu phần dư được giải mã và thông tin nội dự đoán. Các khối được giải mã được xử lý một cách thích hợp để tạo ra các ảnh được giải mã và các ảnh được giải mã có thể được lưu trữ đệm trong mạch bộ nhớ (không được thể hiện) và được sử dụng như là các ảnh tham chiếu trong một vài ví dụ.

Bộ mã hóa entropy 725 có cấu trúc để tạo khuôn dạng dòng bit để bao gồm khối được mã hóa. Bộ mã hóa entropy 725 có cấu trúc để bao gồm thông tin khác nhau theo tiêu chuẩn thích hợp, như tiêu chuẩn HEVC. Trong ví dụ của sáng chế, bộ mã hóa entropy 725 có cấu trúc để bao gồm dữ liệu điều khiển chung, thông tin dự đoán được lựa chọn (ví dụ, thông tin nội dự đoán hoặc thông tin dự đoán liên đới), thông tin phần dư, và thông tin thích hợp khác trong dòng bit. Lưu ý rằng, theo đối tượng được bộc lộ, khi mã hóa khối trong chế độ con hợp nhất của chế độ dự đoán liên đới hoặc chế độ dự đoán hai chiều, không có thông tin phần dư.

Fig.8 thể hiện sơ đồ của bộ giải mã video 810 theo phương án khác của sáng chế. Bộ giải mã video 810 có cấu trúc để thu các ảnh được mã hóa mà là một phần của chuỗi video được mã hóa, và giải mã các ảnh được mã hóa để tạo ra các ảnh được khôi phục. Trong ví dụ của sáng chế, bộ giải mã video 810 được sử dụng trong vị trí của bộ giải mã video 410 trong ví dụ trên Fig.4.

Trong ví dụ trên Fig.8, bộ giải mã video 810 bao gồm bộ giải mã entropy 871, bộ giải mã liên đới 880, bộ giải mã phần dư 873, môđun khôi phục 874, và bộ nội giải mã 872 được ghép nối với nhau như được thể hiện trên Fig.8.

Bộ giải mã entropy 871 có thể có cấu trúc để khôi phục, từ ảnh được mã hóa, các ký tự định trước mà biểu diễn các phần tử cú pháp mà ảnh được mã hóa của nó được tạo thành. Các ký tự này có thể bao gồm, ví dụ, chế độ trong đó khối được mã hóa (như, ví dụ, chế độ nội dự đoán, chế độ dự đoán liên đới, chế độ dự đoán hai chiều, hai chế độ phía sau trong chế độ con hợp nhất hoặc chế độ con khác), thông tin dự đoán (như, ví dụ, thông tin nội dự đoán hoặc thông tin dự đoán liên đới) mà có thể nhận dạng mẫu hoặc siêu dữ liệu định trước mà được sử dụng cho việc dự đoán bởi bộ nội giải mã 872 hoặc bộ giải mã liên đới 880, một cách lần lượt, thông tin phần dư trong dạng của, ví dụ, các hệ số biến đổi được lượng tử hóa, và loại tương tự. Trong ví dụ của sáng chế, khi chế độ dự đoán là chế độ dự đoán liên đới hoặc dự đoán hai chiều, thông tin dự đoán liên đới được cấp tới bộ giải mã liên đới 880; và khi loại dự đoán là loại nội dự đoán, thông tin nội dự đoán được cấp tới bộ nội giải mã 872. Thông tin phần dư có thể được đưa vào lượng tử hóa ngược và được cấp tới bộ giải mã phần dư 873.

Bộ giải mã liên đới 880 có cấu trúc để thu thông tin dự đoán liên đới, và tạo ra các kết quả dự đoán liên đới dựa trên thông tin dự đoán liên đới.

Bộ nội giải mã 872 có cấu trúc để thu thông tin nội dự đoán, và tạo ra các kết quả dự đoán dựa trên thông tin nội dự đoán.

Bộ giải mã phần dư 873 có cấu trúc để thực hiện việc lượng tử hóa ngược để tách các hệ số biến đổi được giải lượng tử, và xử lý các hệ số biến đổi được giải lượng tử để chuyển đổi phần dư từ miền tần số thành miền không gian. Bộ giải mã phần dư 873 có thể cũng yêu cầu thông tin điều khiển định trước (để bao gồm tham số lượng tử hóa (QP)), và thông tin này có thể được cấp bởi bộ giải mã entropy 871 (đường truyền dữ liệu không được thể hiện đây có thể chỉ là thông tin điều khiển dung lượng thấp).

Môđun khôi phục 874 có cấu trúc để kết hợp, trong miền không gian, phần dư như được xuất ra bởi bộ giải mã phần dư 873 và các kết quả dự đoán (như được xuất ra bởi các môđun dự đoán liên đới hoặc nội dự đoán như là trường hợp có thể có) để tạo ra khối được khôi phục, mà có thể là một phần của ảnh được khôi phục, mà có thể là một phần của video được khôi phục. Lưu ý rằng các hoạt động thích hợp khác, như hoạt động giải khối và loại tương tự, có thể được thực hiện để cải thiện chất lượng trực quan.

Lưu ý rằng các bộ mã hóa video 403, 603, và 703, và các bộ giải mã video 410, 510, và 810 có thể được thực hiện nhờ sử dụng bất kỳ kỹ thuật thích hợp. Trong phương án của sáng chế, các bộ mã hóa video 403, 603, và 703, và các bộ giải mã video 410, 510, và 810 có thể được thực hiện nhờ sử dụng một hoặc nhiều mạch tích hợp. Trong phương án khác, các bộ mã hóa video 403, 603, và 603, và các bộ giải mã video 410, 510, và 810 có thể được thực hiện nhờ sử dụng một hoặc nhiều bộ xử lý mà thực thi các lệnh phần mềm.

Các khía cạnh của sáng chế bao gồm việc điều khiển của các công cụ mã hóa và các chức năng với các cờ điều kiện ràng buộc trong dòng video được mã hóa.

Các bộ mã hóa và giải mã video có thể hoạt động trên kích cỡ ảnh định trước mà được xác định và cố định đối với chuỗi video được mã hóa (CVS-coded chuỗi video), Nhóm của các ảnh (GOP-Group of Pictures), hoặc khung thời gian đa ảnh. Trong ví dụ của sáng chế, như trong MPEG-2, các thiết kế hệ thống đã được biết đến để thay đổi độ phân giải chiều ngang (và, liên quan đến, kích cỡ ảnh) phụ thuộc vào các yếu tố như hoạt động của các cảnh, nhưng chỉ tại các ảnh I, một cách điển hình đối với GOP. Việc tái lấy mẫu các ảnh tham chiếu để sử dụng cho các độ phân giải khác nhau trong CVS đã được biết đến, ví dụ, từ Khuyến nghị ITU-T H.263 Phụ lục P. Tuy nhiên, kích cỡ ảnh không thay đổi, chỉ các ảnh tham chiếu được tái lấy mẫu, có khả năng thu được chỉ trong các phần của ảnh đang được sử dụng (ví dụ, trong trường hợp lấy mẫu xuống), hoặc chỉ các phần của cảnh được chụp (ví dụ, trong trường hợp của lấy mẫu lên). Trong một vài ví dụ, như trong H.263 phụ lục Q, việc tái lấy mẫu của khối

macrô riêng biệt bởi hệ số của hai trong mỗi chiều (ví dụ, hướng lên hoặc hướng xuống) được cho phép. Kích cỡ ảnh có thể vẫn giữ nguyên. Kích cỡ của khối macrô có thể được cố định, ví dụ, trong H.263, và do đó không cần được báo hiệu.

Kích cỡ ảnh trong các ảnh được dự đoán có thể được thay đổi. Trong ví dụ của sáng chế, như VP9, việc tái lấy mẫu ảnh tham chiếu và thay đổi của độ phân giải đối với toàn bộ ảnh được cho phép. Trong ví dụ của sáng chế, (bao gồm, ví dụ, tác giả Hendry, và các tác giả khác, "Thay đổi độ phân giải thích nghi (ARC) đối với VVC", tài liệu nhóm Video hợp tác JVET- M0135-v1, 9-19 tháng 1 năm 2019, toàn bộ nội dung được kết hợp ở đây) việc tái lấy mẫu của toàn bộ ảnh tham chiếu thành độ phân giải khác (ví dụ, độ phân giải cao hơn hoặc độ phân giải thấp hơn) được cho phép. Các độ phân giải ứng viên khác nhau có thể được mã hóa trong tập hợp tham số chuỗi (SPS) và có thể được viện dẫn tới bởi các phân tử cú pháp dựa trên ảnh trong tập hợp tham số ảnh (PPS-picture parameter set).

Các kỹ thuật để báo hiệu kích cỡ ảnh thích ứng trong dòng bit video được bộc lộ trong sáng chế.

Khi các ảnh được mã hóa thành dòng bit mà bao gồm một hoặc nhiều lớp với các chất lượng khác nhau, dòng bit có thể có các phân tử cú pháp mà chỉ rõ các lớp (hoặc tập hợp các lớp) nào có thể được xuất ra tại phía bộ giải mã. Tập hợp của các lớp cần được xuất ra có thể được xác định là tập hợp lớp được xuất ra. Trong mã hóa-giải mã video ví dụ mà hỗ trợ nhiều lớp và các khả năng biến đổi tỷ lệ, một hoặc nhiều tập hợp lớp đầu ra có thể được báo hiệu trong tập tham số video (VPS-video parameter set). Các phân tử cú pháp mà chỉ rõ mức bậc mẫu (PTL-profile tier level) đối với toàn bộ dòng bit hoặc một hoặc nhiều tập hợp lớp đầu ra có thể được báo hiệu trong VPS, DPS mà đồng nhất với thông tin khả năng bộ giải mã (DCI-Decoder Capability Information), SPS, PPS, bản tin SEI, hoặc loại tương tự. Trong thông tin PTL, thông tin điều kiện ràng buộc chung mà có thể chỉ rõ các điều kiện ràng buộc về các công cụ mã hóa hoặc các khả năng có thể được hiện diện. Có mong muốn biểu diễn và báo hiệu

một cách hiệu quả thông tin điều kiện ràng buộc đối với các công cụ mã hóa và các chức năng khác nhau.

Ảnh con có thể liên quan đến, ví dụ, sắp xếp hình chữ nhật của các mẫu, các khối, các khối macrô, các đơn vị mã hóa, hoặc các thực thể tương tự mà được nhóm lại về mặt ngữ nghĩa học, và có thể được mã hóa độc lập trong độ phân giải được thay đổi. Một hoặc nhiều ảnh con có thể tạo thành ảnh. Một hoặc nhiều ảnh con được mã hóa có thể tạo thành ảnh được mã hóa. Một hoặc nhiều ảnh con có thể được ghép thành ảnh, và một hoặc nhiều ảnh con có thể được tách từ ảnh. Trong một vài ví dụ, một hoặc nhiều ảnh con được mã hóa có thể được ghép trong miền được nén mà không chuyển mã tới mức mẫu thành ảnh được mã hóa. Trong một vài ví dụ, một hoặc nhiều ảnh con được mã hóa có thể được tách từ ảnh được mã hóa trong miền được nén.

Thay đổi độ phân giải thích nghi (ARC-Adaptive Resolution Change) có thể liên quan đến cơ chế mà cho phép sự thay đổi độ phân giải của ảnh hoặc ảnh con trong chuỗi video được mã hóa (CVS), ví dụ, bằng cách tái lấy mẫu ảnh tham chiếu. Các tham số ARC có thể liên quan đến thông tin điều khiển được sử dụng để thực hiện thay đổi độ phân giải thích nghi. Các tham số ARC có thể bao gồm các tham số lọc, các hệ số biến đổi tỷ lệ, các độ phân giải của đầu ra và/hoặc ảnh tham chiếu, các cờ điều khiển khác nhau, và/hoặc loại tương tự.

Phần mô tả nêu có thể được áp dụng tới việc mã hóa và giải mã ảnh video đơn được mã hóa độc lập về mặt ngữ nghĩa. Trước khi mô tả việc mã hóa/giải mã của các ảnh con với các tham số ARC độc lập và tính phức tạp bổ sung của việc này, báo hiệu các tham số ARC sẽ được mô tả.

FIG.9 thể hiện các ví dụ (ví dụ, các tùy chọn) để báo hiệu các tham số ARC theo các phương án của sáng chế. Hiệu quả mã hóa, độ phức tạp, và cấu trúc có thể thay đổi giữa các ví dụ. Tiêu chuẩn hoặc kỹ thuật mã hóa video có thể lựa chọn một hoặc nhiều trong số ví dụ, hoặc các biến thể khác, để báo hiệu các tham số ARC. Các ví dụ có thể không loại trừ lẫn nhau, và có thể được hoán đổi dựa trên các nhu cầu ứng dụng, kỹ thuật tiêu chuẩn liên quan, hoặc sự lựa chọn của bộ mã hóa và/hoặc loại tương tự.

Các lớp của các tham số ARC có thể bao gồm:

- Các hệ số lấy mẫu lên và/hoặc lấy mẫu xuống, tách biệt hoặc được kết hợp trong chiều X và chiều Y trong đó một hoặc nhiều phần tử cú pháp ngăn mà có thể trở tới Bảng bao gồm các hệ số lấy mẫu lên và/hoặc lấy mẫu xuống có thể được mã hóa;
- Các hệ số lấy mẫu lên và/hoặc lấy mẫu xuống, với việc bổ sung của chiều thời gian, mà chỉ báo sự thu phóng tốc độ cố định trong và/hoặc ngoài số lượng ảnh định trước trong đó một hoặc nhiều phần tử cú pháp ngăn mà có thể trở tới Bảng bao gồm các hệ số lấy mẫu lên và/hoặc lấy mẫu xuống với việc bổ sung của chiều thời gian có thể được mã hóa;
- Độ phân giải, trong chiều X hoặc Y, trong các đơn vị của các mẫu, các khối, các khối macro, các CU, hoặc mức chi tiết phù hợp bất kỳ khác, của ảnh đầu vào, hình ảnh đầu ra, ảnh tham chiếu, ảnh được mã hóa, được kết hợp hoặc được tách biệt. Nếu có nhiều hơn một độ phân giải (ví dụ, một độ phân giải đối với ảnh đầu vào, độ phân giải khác đối với ảnh tham chiếu), tập hợp của các giá trị có thể được giả định từ tập hợp giá trị khác, mà có thể được tạo công, ví dụ, bằng cách sử dụng các cờ, như được mô tả dưới đây;
- –các tọa độ “cong” mà tương tự như được sử dụng trong Phụ lục P H.263, trong mức chi tiết phù hợp như được mô tả ở trên. H.263 phụ lục P xác định cách thức hiệu quả để mã hóa các tọa độ cong. Các cách thức hiệu quả khác có thể được suy ra. Ví dụ, độ dài biên có thể đảo ngược, mã hóa kiểu “Huffman” của các tọa độ cong theo Phụ lục P có thể được thay thế bởi mã hóa nhị phân độ dài thích hợp, trong đó độ dài của từ mã nhị phân có thể thu được từ kích cỡ ảnh lớn nhất mà được nhân bởi hệ số và độ dịch định trước với giá định trước, để “làm cong” phía ngoài của các biên của kích cỡ ảnh lớn nhất.

- -Các tham số lọc lấy mẫu lên và/hoặc lấy mẫu xuống. Trong ví dụ của sáng chế, có thể chỉ có một bộ lọc cho việc lấy mẫu lên và/hoặc lấy mẫu xuống. Trong ví dụ của sáng chế, việc cho phép tính linh hoạt hơn trong thiết kế bộ lọc là ưu điểm. Trong ví dụ của sáng chế, các tham số lọc được bảo hiệu. Các tham số lọc có thể được lựa chọn bằng cách sử dụng chỉ số trong danh sách của các thiết kế lọc khả năng. Bộ lọc có thể được chỉ rõ đầy đủ (ví dụ, bằng cách chỉ rõ danh sách của các hệ số lọc, nhờ sử dụng các kỹ thuật mã hóa entropy thích hợp), bộ lọc có thể được lựa chọn ẩn thông qua các tỷ lệ lấy mẫu lên hoặc lấy mẫu xuống mà được bảo hiệu theo bất kỳ trong số các cơ chế nêu trên, và/hoặc loại tương tự.

Phần mô tả có thể được áp dụng tới việc mã hóa của tập hợp hữu hạn các hệ số lấy mẫu lên và/hoặc xuống (hệ số giống nhau cần được sử dụng trong cả chiều X và Y), được chỉ báo thông qua từ mã. Từ mã này có thể một cách hiệu quả được mã hóa độ dài biến thiên, ví dụ nhờ sử dụng mã Golomb số mũ cho các phần tử cú pháp nhất định trong các tiêu chuẩn kỹ thuật mã hóa video (ví dụ H.264 và H.265). Việc ánh xạ thích hợp của các giá trị tới các hệ số lấy mẫu lên hoặc lấy mẫu xuống có thể, ví dụ, được dựa trên Bảng 1.

Bảng 1: Ánh xạ các hệ số lấy mẫu lên hoặc lấy mẫu xuống, các từ mã, và các mã Golomb số mũ

Từ mã	Mã số mũ Golomb	Độ phân giải gốc/đích
0	1	1/1
1	010	1/1,5 (biến đổi tỷ lệ lên bởi 50%)
2	011	1,5/1 (biến đổi tỷ lệ xuống bởi 50%)
3	00100	1/2 (biến đổi tỷ lệ lên bởi 100%)
4	00101	2/1 (biến đổi tỷ lệ xuống bởi 100%)

Các cách ánh xạ tương tự có thể được suy ra theo ứng dụng và các khả năng của các cơ chế biến đổi tỷ lệ lên và biến đổi tỷ lệ xuống khả dụng trong kỹ

thuật hoặc tiêu chuẩn nén video. Bảng 1 có thể được mở rộng thích hợp tới các giá trị bổ sung. Các giá trị có thể được biểu diễn bởi các cơ chế mã hóa entropy khác ngoài các mã Golomb số mũ, ví dụ bằng cách sử dụng mã hóa nhị phân. Trong ví dụ của sáng chế, các cơ chế mã hóa entropy ngoài mã Golomb số mũ có thể có các hiệu quả nhất định khi các hệ số tái lấy mẫu nằm ngoài các cơ cấu xử lý video (ví dụ, bộ mã hóa và bộ giải mã), ví dụ, bởi các phần tử mạng nhận biết đa phương tiện (MANE-media-aware network element). Khi không có sự thay đổi độ phân giải được yêu cầu (ví dụ, độ phân giải gốc/đích bằng 1 trong Bảng 1), mã Golomb số mũ (ví dụ, chỉ một bit được thể hiện trong Bảng 1) có thể được lựa chọn, mà có thể có ưu điểm hiệu quả mã hóa, ví dụ, so với sử dụng các mã nhị phân đối với trường hợp chung nhất.

Số lượng mục trong Bảng 1, và các ngữ nghĩa học tương ứng có thể được cấu hình đầy đủ hoặc một phần. Ví dụ, tổng quát cơ bản của Bảng 1 có thể được truyền trong tập hợp tham số mức cao như SPS hoặc DPS. Ngoài ra, một hoặc nhiều bảng tương tự bảng 1 có thể được xác định trong kỹ thuật hoặc tiêu chuẩn mã hóa video, và có thể được lựa chọn thông qua, ví dụ, SPS hoặc DPS.

Việc làm thế nào hệ số lấy mẫu lên hoặc lấy mẫu xuống (thông tin ARC), được mã hóa như được mô tả nêu trên, có thể được chứa trong cú pháp kỹ thuật hoặc tiêu chuẩn mã hóa video được mô tả nêu trên. Các xem xét tương tự có thể áp dụng tới một, hoặc một ít, từ mã mà điều khiển các bộ lọc lấy mẫu lên hoặc lấy mẫu xuống. Khi lượng dữ liệu tương đối lớn được yêu cầu đối với bộ lọc hoặc các cấu trúc dữ liệu khác, phần mô tả là như sau.

Viện dẫn tới FIG.9, trong ví dụ của sáng chế, như trong H.263 phụ lục P, thông tin ARC 902 trong dạng của bốn tọa độ cong được chứa trong thông tin tiêu đề ảnh 901, cụ thể trong phần mở rộng thông tin tiêu đề H.263 PLUSPTYPE 903. Thông tin tiêu đề ảnh 901 bao gồm thông tin ARC 902 và phần mở rộng thông tin tiêu đề H.263 PLUSPTYPE 903 có thể được áp dụng khi i) thông tin tiêu đề ảnh là khả dụng, và ii) sự thay đổi thường xuyên của thông tin ARC được kỳ vọng. Tuy nhiên, thông tin tiêu đề khi sử dụng báo hiệu

kiểu H.263 có thể cao, và các hệ số biến đổi tỷ lệ có thể không được áp dụng trong số các biên ảnh khi thông tin tiêu đề ảnh có thể ở trạng thái chuyển tiếp.

Viện dẫn tới FIG.9, trong ví dụ như JVCET-M135-v1, thông tin tham chiếu ARC 905 (ví dụ, chỉ số) nằm trong PPS 904 có thể trở tới Bảng (hoặc Bảng độ phân giải đích) 906 bao gồm các độ phân giải đích (ví dụ, các độ phân giải 1-3) trong đó Bảng 906 nằm trong SPS 907. Việc đặt các độ phân giải đích trong Bảng 906 trong SPS 907 có thể được điều chỉnh bằng cách sử dụng SPS như là điểm thỏa thuận khả năng tương tác trong khi trao đổi khả năng. Độ phân giải có thể thay đổi, trong giới hạn được thiết lập bởi các giá trị (ví dụ, các độ phân giải 1-3) trong Bảng 906 từ một ảnh tới ảnh khác bằng cách tham chiếu PPS thích hợp 904.

Viện dẫn lại tới FIG.9, các ví dụ hoặc các tùy chọn bổ sung có thể được sử dụng để truyền thông tin ARC trong dòng bit video. Các tùy chọn có thể được đồng thời thể hiện trong cùng một kỹ thuật hoặc tiêu chuẩn mã hóa video.

Trong phương án của sáng chế, thông tin ARC 909 như hệ số tái lấy mẫu (hệ số thu phóng) có thể được thể hiện trong thông tin tiêu đề như tiêu đề lát, tiêu đề GOB, tiêu đề ô, hoặc tiêu đề nhóm ô hoặc loại tương tự. Thông tin tiêu đề nhóm ô 908 được minh họa trong FIG.9 chẳng hạn. Phương án này có thể thích hợp khi không gian bộ nhớ đối với thông tin ARC 909 là nhỏ, như một độ dài biến ue(v) hoặc từ mã độ dài cố định có ít bit, như được thể hiện nêu trên.

Việc có thông tin ARC 909 trong thông tin tiêu đề (ví dụ, thông tin tiêu đề nhóm ô 908 trong FIG.9, thông tin tiêu đề lát, hoặc thông tin tiêu đề ô) một cách trực tiếp có thể có các ưu điểm bổ sung trong đó thông tin ARC 909 có thể được áp dụng tới ảnh con được biểu diễn bởi, ví dụ, nhóm ô (hoặc lát, ô) tương ứng, ngoài toàn bộ ảnh. Ngoài ra, trong ví dụ của sáng chế, ngay cả nếu tiêu chuẩn hoặc kỹ thuật nén video liên quan đến chỉ các thay đổi độ phân giải thích nghi dựa trên toàn bộ ảnh (trái ngược với, ví dụ, các thay đổi độ phân giải thích nghi dựa trên nhóm ô), việc đặt thông tin ARC 909 vào thông tin tiêu đề

nhóm ô 908 so với đặt thông tin ARC 909 vào thông tin tiêu đề ảnh loại H.263 có thể có các ưu điểm nhất định từ khía cạnh phục hồi lỗi.

Viện dẫn tới FIG.9, thông tin ARC 912 có thể được hiện diện trong tập hợp tham số 911 như PPS, tập hợp tham số thông tin tiêu đề, tập hợp tham số ô, tập hợp tham số thích nghi (APS-adaptation parameter set), hoặc loại tương tự. APS 911 được minh họa trong FIG.9 chẳng hạn. Phạm vi của tập hợp tham số 911 có thể có ưu điểm không lớn hơn ảnh, ví dụ nhóm ô. Việc sử dụng thông tin ARC (ví dụ, thông tin ARC 912) có thể là ẩn thông qua việc kích hoạt của tập hợp tham số liên quan (ví dụ, APS 911). Ví dụ, khi kỹ thuật hoặc tiêu chuẩn mã hóa video nhắm tới chỉ ARC dựa trên ảnh, tập hợp tham số ảnh (PPS) hoặc tương đương có thể là thích hợp.

Trong một vài phương án, thông tin tham chiếu ARC 913 có thể được hiện diện trong thông tin tiêu đề nhóm ô 914 hoặc cấu trúc dữ liệu tương tự (ví dụ, thông tin tiêu đề ảnh, thông tin tiêu đề lát, thông tin tiêu đề ô, hoặc thông tin tiêu đề GOP) như được mô tả nêu trên. Thông tin tiêu đề nhóm ô 914 được minh họa trong FIG.9 chẳng hạn. Thông tin tham chiếu 913 này có thể liên quan đến tập con của thông tin ARC 915 khả dụng trong tập hợp tham số 916 với phạm vi ngoài ảnh đơn, ví dụ SPS, DPS hoặc loại tương tự. SPS 916 được minh họa trong FIG.9 chẳng hạn.

FIG.10 thể hiện các ví dụ về báo hiệu các tham số ARC theo các phương án của sáng chế. FIG.10 thể hiện các ví dụ sơ đồ cú pháp được sử dụng trong các tiêu chuẩn mã hóa video. Trong ví dụ của sáng chế, ký hiệu của các sơ đồ cú pháp này thường tuân theo lập trình loại C. Các đường in đậm có thể chỉ báo các phần tử cú pháp thể hiện trong dòng bit, và các đường nhạt có thể chỉ báo các dòng điều khiển hoặc thiết lập của các biến.

Viện dẫn tới FIG.10, thông tin tiêu đề nhóm ô 1001 như cấu trúc cú pháp mẫu của tiêu đề ứng dụng được cho phần (ví dụ, phần chữ nhật) của hình ảnh có thể chứa có điều kiện, độ dài biến, phần tử cú pháp được mã hóa Golomb số mũ `dec_pic_size_idx` 1002 (được mô tả in đậm). Sự có mặt của phần tử cú pháp (ví dụ, `dec_pic_size_idx` 1002) trong thông tin tiêu đề nhóm ô 1001 có thể

được tạo công dựa trên độ phân giải thích nghi, ví dụ, được biểu diễn bởi cờ (ví dụ, `adaptive_pic_resolution_change_flag`) 1003. Giá trị của cờ (ví dụ, `adaptive_pic_resolution_change_flag`) 1003 không được thể hiện in đậm, và do đó cờ được hiện diện trong dòng bit tại điểm trong đó cờ xuất hiện trong sơ đồ cú pháp. Việc độ phân giải thích ứng được sử dụng cho hình ảnh hay một phần của hình ảnh có thể được báo hiệu trong cấu trúc cú pháp bậc cao (ví dụ, SPS 1010) bên trong hoặc bên ngoài dòng bit.

Viện dẫn tới FIG.10, phần trích của SPS 1010 được thể hiện. Phần tử cú pháp thứ nhất được thể hiện là cờ (ví dụ, `adaptive_pic_resolution_change_flag`) 1011. Khi cờ là đúng, cờ này có thể chỉ báo việc sử dụng của độ phân giải thích nghi mà có thể yêu cầu thông tin điều khiển định trước. Trong ví dụ của sáng chế, thông tin điều khiển định trước được hiện diện có điều kiện dựa trên giá trị của cờ 1011 như được thể hiện bởi câu lệnh nếu (if) () 1012 trong SPS 1010 và thông tin tiêu đề nhóm ô 1001.

Khi độ phân giải thích nghi được sử dụng, như được thể hiện trong ví dụ trong FIG.10, độ phân giải đầu ra trong các đơn vị của các mẫu (hoặc độ phân giải của ảnh đầu ra) 1013 có thể được mã hóa. Độ phân giải đầu ra 1013 có thể liên quan đến cả độ phân giải độ rộng (ví dụ, `output_pic_width_in_luma_samples`) và độ phân giải độ cao (ví dụ, `output_pic_height_in_luma_samples`). Độ phân giải độ rộng và độ phân giải độ cao có thể xác định độ phân giải đầu ra 1013. Trong kỹ thuật hoặc tiêu chuẩn mã hóa video, các độ phân giải nhất định đối với các giá trị của độ phân giải đầu ra 1013 có thể được xác định. Ví dụ, định nghĩa mức có thể giới hạn số lượng của tổng mẫu đầu ra (ví dụ, tích của `output_pic_width_in_luma_samples` và `output_pic_height_in_luma_samples`). Tiêu chuẩn hoặc kỹ thuật mã hóa video, hoặc tiêu chuẩn hoặc kỹ thuật phía ngoài (ví dụ, tiêu chuẩn hệ thống) có thể giới hạn phạm vi (ví dụ, độ phân giải độ rộng và/hoặc độ phân giải độ cao có thể chia cho lũy thừa của 2), tỷ lệ co (ví dụ, tỷ lệ của độ phân giải độ rộng trên độ phân giải độ cao là 4:3 hoặc 16:9), hoặc loại tương tự. Trong ví dụ của sáng chế, các giới hạn nêu trên có thể được đưa ra để hỗ trợ thực hiện phần cứng.

Trong các ứng dụng nhất định, bộ mã hóa chỉ dẫn bộ giải mã để sử dụng kích cỡ ảnh tham chiếu định trước mà không giả định rằng kích cỡ là kích cỡ ảnh đầu ra. Ví dụ, phần tử cú pháp (ví dụ, `reference_pic_size_present_flag`) 1014 tạo công sự hiện diện có điều kiện của các chiều ảnh tham chiếu 1015. Các chiều ảnh tham chiếu 1015 có thể liên quan đến cả độ rộng (ví dụ, `reference_pic_width_in_luma_samples`) và độ cao (ví dụ, `reference_pic_height_in_luma_samples`).

FIG.10 thể hiện bảng của các độ rộng và độ cao ảnh giải mã có thể được áp dụng. Bảng có thể được biểu diễn bởi chỉ báo bảng (ví dụ, phần tử cú pháp `num_dec_pic_size_in_luma_samples_minus1`) 1016. “minus1” có thể liên quan đến diễn dịch của giá trị phần tử cú pháp 1016. Ví dụ, nếu giá trị được mã hóa là 0, một mục bảng được biểu diễn. Nếu giá trị được mã hóa bằng năm, sáu mục bảng được biểu diễn. Đối với mỗi “dòng” trong Bảng, độ rộng và độ cao ảnh được giải mã được chứa trong các phần tử cú pháp 1017.

Các mục bảng được biểu diễn bởi các phần tử cú pháp 1017 có thể được đánh chỉ số nhờ sử dụng các phần tử cú pháp `dec_pic_size_idx` 1002 trong thông tin tiêu đề nhóm ô 1001, nhờ đó cho phép các kích cỡ giải mã khác nhau, các hệ số thu phóng theo nhóm ô.

Các chuẩn và kỹ thuật mã hóa video (ví dụ, VP9) có thể hỗ trợ khả năng biến đổi tỷ lệ không gian bằng cách thực hiện các dạng định trước của việc tái lấy mẫu ảnh tham chiếu kết hợp với khả năng biến đổi tỷ lệ thời gian, và do đó có thể cho phép khả năng biến đổi tỷ lệ theo không gian. Trong phương án của sáng chế, các ảnh tham chiếu được lấy mẫu lên nhờ sử dụng các kỹ thuật kiểu ARC tới độ phân giải cao hơn để tạo thành cơ sở của lớp tăng cường không gian. Ảnh được lấy mẫu lên có thể được tinh chỉnh, nhờ sử dụng các cơ chế dự đoán thường (ví dụ, dự đoán bù chuyển động đối với dự đoán liên đới từ các ảnh tham chiếu) tại độ phân giải cao, ví dụ để thêm độ chi tiết.

Sáng chế có thể được sử dụng trong môi trường này. Trong một vài ví dụ, giá trị trong thông tin tiêu đề đơn vị Lớp trừu tượng mạng (NAL-Network Abstraction Layer), ví dụ, trường ID thời gian, được sử dụng để chỉ báo lớp thời

gian và không gian. Các ưu điểm có thể bao gồm các đơn vị chuyển tiếp được lựa chọn (SFU-Selected Forwarding Unit) hiện tại có thể được sử dụng mà không cải biến đối với các môi trường có thể biến đổi tỷ lệ. Các SFU có thể, dựa trên giá trị ID thời gian thông tin tiêu đề đơn vị NAL, được tạo ra và được tối ưu hóa đối với lớp thời gian được lựa chọn chuyển tiếp. Để cho phép điều này, việc ánh xạ có thể được thực hiện giữa kích cỡ ảnh được mã hóa và lớp thời gian như được chỉ báo bởi trường ID thời gian trong thông tin tiêu đề đơn vị NAL.

Khi các ảnh được mã hóa thành dòng bit mà bao gồm một hoặc nhiều lớp với các chất lượng khác nhau, dòng bit có thể có các phần tử cú pháp mà chỉ rõ các lớp nào có thể được xuất ra tại phía bộ giải mã. Tập hợp của các lớp cần được xuất ra có thể được xác định là tập hợp lớp được xuất ra. Trong mã hóa-giải mã video mà hỗ trợ nhiều lớp và các khả năng biến đổi tỷ lệ, một hoặc nhiều tập hợp lớp đầu ra có thể được báo hiệu trong một hoặc nhiều tập tham số video (VPS-video parameter set). Một hoặc nhiều phần tử cú pháp mà chỉ rõ thông tin PTL đối với toàn bộ dòng bit, CVS, mỗi tập hợp lớp đầu ra, và/hoặc loại tương tự có thể được báo hiệu trong cấu trúc cú pháp mức cao, như VPS, DPS, DCI, SPS, PPS, APS, GOP, chuỗi, thông tin tiêu đề, bản tin SEI, hoặc loại tương tự.

Các công cụ mã hóa mức khối có thể được sử dụng để giải mã các điểm ảnh hoặc các mẫu trong ảnh để khôi phục ảnh. Các công cụ mã hóa mức khối có thể bao gồm bất kỳ các công cụ mã hóa thích hợp được sử dụng trong việc khôi phục của khối mã hóa, như các công cụ mã hóa cho việc dự đoán liên đới (hoặc các công cụ mã hóa liên đới), các công cụ mã hóa cho việc nội dự đoán (hoặc intra các công cụ nội dự đoán), bộ lọc vòng thích nghi (ALF), bộ lọc giải khối (DBF), mã hóa entropy, biến đổi, và loại tương tự.

Cú pháp mức cao (HLS-High-level syntax) có thể chỉ rõ thông tin về chức năng, giao diện hệ thống, điều khiển mức ảnh của các công cụ và điều khiển đệm, và loại tương tự. Ví dụ, HLS có thể chỉ rõ phân vùng (ví dụ, ô, lát, ảnh con), quản lý đệm, truy nhập ngẫu nhiên (ví dụ, IDR, truy nhập ngẫu nhiên sạch (CRA-clean random access)), các tập hợp tham số (ví dụ, VPS, SPS, PPS,

APS), tái lấy mẫu ảnh tham chiếu (RPR-reference picture resampling), khả năng biến đổi tỷ lệ, và/hoặc loại tương tự. Cú pháp mức cao có thể nằm phía trên mức khối.

Thông tin điều khiển có thể bao gồm thông tin điều khiển công cụ mức SPS, thông tin điều khiển công cụ mức PPS, thông tin điều khiển mức chuỗi, thông tin điều khiển mức dòng bit, và/hoặc loại tương tự.

Các cờ điều kiện ràng buộc có thể là một phần của cấu trúc HLS.

Trong ví dụ của sáng chế, các cờ điều kiện ràng buộc chỉ báo điều khiển của các công cụ. Các cờ điều kiện ràng buộc có thể được bố trí trong một trong số: thông tin điều khiển mức chuỗi và thông tin điều khiển mức dòng bit. Trong ví dụ của sáng chế, nếu các công cụ nhất định được vô hiệu hóa bởi các cờ điều kiện ràng buộc, không có thông tin điều khiển được hiện diện trong HLS và các công cụ không được sử dụng, ví dụ, đối với các khối mã hóa trong phạm vi tương ứng với HLS.

Các cờ điều kiện ràng buộc có thể được báo hiệu trong cấu trúc cú pháp mẫu.

Thông tin điều kiện ràng buộc có thể chỉ báo rằng công cụ cụ thể, chức năng hoặc các phần tử cú pháp có được hiện diện hay không hoặc được sử dụng trong dòng video tương ứng hoặc dòng bit video được mã hóa. Ví dụ, các cờ điều kiện ràng buộc có thể chỉ báo rằng các công cụ mã hóa liên đới, các công cụ mã hóa nội bộ, DBF, mã hóa entropy, biến đổi, phân chia (ví dụ, ô, lát), quản lý đệm, truy nhập ngẫu nhiên (ví dụ, IDR), tập hợp tham số (ví dụ, SPS, PPS), và/hoặc loại tương tự được hiện diện hoặc được sử dụng trong dòng bit được mã hóa. Thông tin điều kiện ràng buộc có thể được báo hiệu trong các tập hợp tham số (ví dụ, SPS, VPS, hoặc DCI). Các cờ điều kiện ràng buộc có thể được báo hiệu trong cấu trúc cú pháp mức cao (ví dụ, SPS, VPS, DCI).

FIG.11 thể hiện ví dụ về tập hợp của các phần tử cú pháp PTL theo phương án của sáng chế. Tập hợp của các phần tử cú pháp PTL có thể bao gồm `general_profile_idc`, `general_tier_flag`, `general_level_idc`, `num_sub_profiles`,

`general_sub_profile_idc`, `sublayer_level_present_flag`, `ptl_alignment_zero_bit`, `sublayer_level_idc`, và thông tin điều kiện ràng buộc chung.

Thông tin điều kiện ràng buộc chung có thể bao gồm thông tin điều kiện ràng buộc về loại nguồn video, các công cụ mã hóa, các chức năng, và/hoặc loại tương tự. FIG.12 thể hiện ví dụ về thông tin điều kiện ràng buộc chung theo phương án của sáng chế. Thông tin điều kiện ràng buộc chung có thể bao gồm các cờ điều kiện ràng buộc, như được thể hiện trên FIG.12. Việc dẫn tới FIG.12, cờ điều kiện ràng buộc (ví dụ, `intra_only_constraint_flag`) 1205 bằng 1 chỉ báo `sh_slice_type` bằng I (tức là, lát là lát nội bộ). Cờ điều kiện ràng buộc (ví dụ, `intra_only_constraint_flag`) 1205 bằng 0 không đặt ra điều kiện ràng buộc (ví dụ, `sh_slice_type` bằng I) đối với tất cả các ảnh được mã hóa trong phạm vi của thông tin PTL trong đó thông tin khác (ví dụ, `profile_idc`) có thể cho phép các lát không nội bộ. Trong ví dụ của sáng chế, cờ điều kiện ràng buộc (ví dụ, `no_alf_constraint_flag`) 1206 bằng 1 có thể chỉ báo rằng `sps_alf_enabled_flag` bằng 0 đối với tất cả CVS trong phạm vi của thông tin PTL, và do đó việc lọc vòng thích nghi không được sử dụng ngay cả nếu việc lọc vòng thích nghi được cho phép, ví dụ, dựa trên `profile_idc`. Cờ điều kiện ràng buộc (ví dụ, `no_alf_constraint_flag`) 1206 bằng 0 không đặt ra điều kiện ràng buộc nêu trên.

Cờ điều kiện ràng buộc (ví dụ, `no_lossless_coding_tool_constraint_flag`) 1201 có thể được báo hiệu trong thông tin điều kiện ràng buộc chung, như được thể hiện trong FIG 12. Cờ điều kiện ràng buộc (ví dụ, `no_lossless_coding_tool_constraint_flag`) 1201 bằng 1 có thể chỉ báo rằng các công cụ mã hóa liên quan đến mã hóa không tổn hao không thể được sử dụng trong phạm vi của thông tin PTL bao gồm cờ điều kiện ràng buộc 1201. Cờ điều kiện ràng buộc (ví dụ, `no_lossless_coding_tool_constraint_flag`) 1201 bằng 0 không đặt ra điều kiện ràng buộc nêu trên.

Cờ điều kiện ràng buộc (ví dụ, `no_lossy_coding_tool_constraint_flag`) 1202 có thể được báo hiệu trong thông tin điều kiện ràng buộc chung, như được thể hiện trong FIG 12. Cờ điều kiện ràng buộc (ví dụ, `no_lossy_coding_tool_constraint_flag`) 1202 bằng 1 có thể chỉ báo rằng các

công cụ mã hóa liên quan đến mã hóa tổn hao không thể được sử dụng trong phạm vi của thông tin PTL bao gồm cờ điều kiện ràng buộc 1202. Cờ điều kiện ràng buộc (ví dụ, `no_lossy_coding_tool_constraint_flag`) 1202 bằng 0 không đặt ra điều kiện ràng buộc nêu trên.

Trong phương án của sáng chế, cờ điều kiện ràng buộc (ví dụ, `no_lossless_coding_tool_constraint_flag`) 1201 có thể không bằng 1 khi cờ điều kiện ràng buộc (ví dụ, `no_lossy_coding_tool_constraint_flag`) 1202 bằng 1. Ngoài ra, cờ điều kiện ràng buộc (ví dụ, `no_lossy_coding_tool_constraint_flag`) 1202 có thể không bằng 1 khi cờ điều kiện ràng buộc (ví dụ, `no_lossless_coding_tool_constraint_flag`) 1201 bằng 1.

Các cờ điều kiện ràng buộc trong thông tin điều kiện ràng buộc chung có thể được sắp xếp trong thứ tự định trước. Thứ tự này có thể được thiết lập dựa trên, ví dụ, tính hợp lệ của các cơ chế tương ứng và/hoặc các công cụ không được sử dụng trong phạm vi của PTL. Thứ tự này có thể được gọi là thứ tự ưu tiên. Thứ tự này có thể được trình diễn trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung từ mức ưu tiên cao tới mức ưu tiên thấp trong đó mức ưu tiên cao chỉ báo rằng việc không sử dụng công cụ (hoặc cơ chế) có tính hợp lệ cao và mức ưu tiên thấp chỉ báo rằng việc không sử dụng công cụ (hoặc cơ chế) có tính hợp lệ thấp. Các hệ số bổ sung ảnh hưởng tới thứ tự có thể bao gồm các công cụ có thể được sử dụng chỉ cho các trường hợp sử dụng cụ thể (ví dụ, các công cụ đối với các ảnh con, khả năng biến đổi tỷ lệ, và/hoặc hỗ trợ đan xen), ảnh hưởng của công cụ đối với bộ mã hóa/bộ giải mã/độ phức tạp thực hiện, và loại tương tự.

Các Fig.13A-13B thể hiện ví dụ về thông tin PTL (bao gồm ví dụ tập hợp của các phần tử cú pháp PTL) theo phương án của sáng chế. Phần tử cú pháp mà chỉ báo số lượng của các cờ điều kiện ràng buộc (ví dụ, `num_available_constraint_flags`) 1301 có thể được báo hiệu trong thông tin PTL. Ví dụ, số lượng cờ điều kiện ràng buộc có thể được báo hiệu phía ngoài của dấu ngoặc thông tin điều kiện ràng buộc chung, như được thể hiện trên FIG.13A. Ngoài ra, phần tử cú pháp mà chỉ báo số lượng của các cờ điều kiện

ràng buộc có thể được báo hiệu trong phần đầu của cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Khi phần tử cú pháp (ví dụ, `num_available_constraint_flags`) được hiện diện và giá trị của phần tử cú pháp (ví dụ, `num_available_constraint_flags`) bằng N, N còn điều kiện ràng buộc thứ nhất có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Ngoài ra, các cờ điều kiện ràng buộc khác có thể không được hiện diện và có thể được giả định bằng giá trị cụ thể. N có thể số nguyên không âm.

Trong phương án của sáng chế, giá trị N (ví dụ, `num_available_constraint_flags`) nằm trong đoạn từ 0 đến số lượng lớn nhất của các cờ điều kiện ràng buộc (ví dụ, `MaxNumConstraintFlags`). Số lượng lớn nhất của các cờ điều kiện ràng buộc có thể là số nguyên dương bất kỳ. Giá trị của số lượng lớn nhất của các cờ điều kiện ràng buộc (ví dụ, `MaxNumConstraintFlags`) có thể được xác định trước là 16, 32, 64, 128, hoặc loại tương tự. Khi giá trị N (ví dụ, `num_available_constraint_flags`) bằng 0, không có các cờ điều kiện ràng buộc được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Việc mã hóa của giá trị N (ví dụ, `num_available_constraint_flags`) có thể được lựa chọn sao cho biểu diễn được mã hóa entropy tương ứng đối với giá trị N và các cờ điều kiện ràng buộc có thể cộng tối đa số có thể chia cho 8 để đảm bảo việc căn thẳng byte.

Theo các khía cạnh của sáng chế, các cờ điều kiện ràng buộc có thể được phân loại thành một hoặc nhiều nhóm thông tin điều kiện ràng buộc. Mỗi nhóm thông tin điều kiện ràng buộc có thể bao gồm một hoặc nhiều cờ điều kiện ràng buộc và có thể có cờ cổng tương ứng. Cờ cổng của nhóm thông tin điều kiện ràng buộc tương ứng có thể chỉ báo rằng các cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc tương ứng có thể được hiện diện hay không. Trong ví dụ của sáng chế, cờ cổng có thể được gọi là cờ hiện diện nhóm điều kiện ràng buộc. Nói chung, cờ cổng được kết hợp với nhóm thông tin điều kiện ràng buộc tương ứng, và được kết hợp với các cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc tương ứng. Trong phương án của sáng chế, cờ cổng kiểm soát rằng các cờ điều kiện ràng buộc trong nhóm thông tin điều

kiện ràng buộc tương ứng có được hiện diện (hoặc được báo hiệu) trong thông tin điều kiện ràng buộc hay không. Ví dụ, nếu cờ cổng của nhóm thông tin điều kiện ràng buộc tương ứng bằng 1, các cờ điều kiện ràng buộc tương ứng với nhóm thông tin điều kiện ràng buộc có thể được hiện diện, ví dụ, trong thông tin điều kiện ràng buộc. Nếu cờ cổng của nhóm thông tin điều kiện ràng buộc tương ứng bằng 0, các cờ điều kiện ràng buộc tương ứng với nhóm thông tin điều kiện ràng buộc có thể không được hiện diện, ví dụ, trong thông tin điều kiện ràng buộc. Trong ví dụ của sáng chế, nếu tất cả các cờ cổng bằng 0, không có các cờ điều kiện ràng buộc được hiện diện.

Các cờ điều kiện ràng buộc có thể có các phạm vi khác nhau. Ví dụ, phạm vi của các cờ điều kiện ràng buộc trong DCI có thể là dòng bit video được mã hóa. Phạm vi của các cờ điều kiện ràng buộc trong VPS có thể là các CLVS với nhiều lớp. Phạm vi của các cờ điều kiện ràng buộc trong SPS có thể là một CLVS.

Cờ (ví dụ, `gate_flag_all`) có thể chỉ báo rằng các cờ cổng có được hiện diện hay không. Trong ví dụ của sáng chế, cờ là sai mà chỉ báo rằng không có các cờ cổng được hiện diện. Trong ví dụ của sáng chế, khi không có các cờ cổng được hiện diện, không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc. Trong ví dụ của sáng chế, cờ là đúng mà chỉ báo rằng một hoặc nhiều cờ cổng được hiện diện.

Theo các khía cạnh của sáng chế, cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa có thể được thu nhận. Mỗi cờ cổng có thể chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không. Thông tin điều kiện ràng buộc có thể dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa. Việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không có thể được xác định dựa trên cờ cổng của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều

kiện ràng buộc. Nhóm thông tin điều kiện ràng buộc của cờ công có thể bao gồm ít nhất một cờ điều kiện ràng buộc đối với tập con của các khối mã hóa. Thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ công có được hiện diện trong thông tin điều kiện ràng buộc hay không. Tập con của các khối mã hóa có thể được khôi phục dựa trên thông tin dự đoán.

Trong phương án của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ công được xác định là được hiện diện trong thông tin điều kiện ràng buộc. Thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa trên ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công. Trong phương án của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ công được xác định là không được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, mỗi cờ công chỉ báo rằng nhóm thông tin điều kiện ràng buộc tương ứng với cờ công tương ứng không được hiện diện trong thông tin điều kiện ràng buộc, và không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, thông tin điều kiện ràng buộc được hiện diện trong cú pháp mức cao đối với ít nhất một ảnh, và tập con của các khối mã hóa bao gồm các khối mã hóa trong ít nhất một ảnh. Cú pháp mức cao có thể là VPS, DPS, DCI, SPS, PPS, APS, GOP, chuỗi, hoặc loại tương tự. Trong ví dụ của sáng chế, cú pháp mức cao là DCI hoặc DPS, phạm vi của thông tin điều kiện ràng buộc bao gồm dòng bit được mã hóa, và do đó bao gồm các khối mã hóa trong dòng bit được mã hóa. Trong ví dụ của sáng chế, cú pháp mức cao là VPS, phạm vi của thông tin điều kiện ràng buộc bao gồm các CLVS với nhiều lớp, và do đó bao gồm các khối mã hóa trong các CLVS. Trong ví dụ của sáng chế, cú pháp mức cao là SPS, phạm vi của thông tin điều kiện ràng buộc bao gồm một CLVS, và do đó bao gồm các khối mã hóa trong một CLVS.

Trong phương án của sáng chế, mỗi cờ điều kiện ràng buộc mà có thể được hiện diện (hoặc được báo hiệu) trong thông tin điều kiện ràng buộc được

phân loại thành một hoặc nhiều nhóm thông tin điều kiện ràng buộc, và do đó có thể được kiểm soát bởi cờ công tương ứng. Trong ví dụ của sáng chế, một hoặc nhiều nhóm thông tin điều kiện ràng buộc bao gồm nhóm thông tin điều kiện ràng buộc thứ nhất và nhóm thông tin điều kiện ràng buộc thứ hai. Các cờ điều kiện ràng buộc bao gồm ít nhất một cờ điều kiện ràng buộc thứ nhất và ít nhất một cờ điều kiện ràng buộc thứ hai. Ít nhất một cờ điều kiện ràng buộc thứ nhất được phân loại thành nhóm thông tin điều kiện ràng buộc thứ nhất, và ít nhất một cờ điều kiện ràng buộc thứ hai được phân loại thành nhóm thông tin điều kiện ràng buộc thứ hai. Cờ công thứ nhất của nhóm thông tin điều kiện ràng buộc thứ nhất có thể chỉ báo rằng ít nhất một cờ điều kiện ràng buộc thứ nhất (hoặc nhóm thông tin điều kiện ràng buộc thứ nhất) có được hiện diện (ví dụ, được báo hiệu) trong thông tin điều kiện ràng buộc hay không. Cờ công thứ hai của nhóm thông tin điều kiện ràng buộc thứ hai có thể chỉ báo rằng ít nhất một cờ điều kiện ràng buộc thứ hai (hoặc nhóm thông tin điều kiện ràng buộc thứ hai) có được hiện diện (ví dụ, được báo hiệu) trong thông tin điều kiện ràng buộc hay không. Do đó, trong ví dụ của sáng chế, thông tin điều kiện ràng buộc không bao gồm các cờ điều kiện ràng buộc bổ sung tới ít nhất một cờ điều kiện ràng buộc thứ nhất và ít nhất một cờ điều kiện ràng buộc thứ hai.

Trong phương án của sáng chế, tập con thứ nhất của các cờ điều kiện ràng buộc được phân loại thành một hoặc nhiều nhóm thông tin điều kiện ràng buộc, và do đó có thể được kiểm soát bởi cờ công tương ứng. Ngoài ra, tập con thứ hai của các cờ điều kiện ràng buộc nằm phía ngoài một hoặc nhiều nhóm thông tin điều kiện ràng buộc, và do đó không được kiểm soát bởi các cờ công. Trong ví dụ của sáng chế, một hoặc nhiều nhóm thông tin điều kiện ràng buộc bao gồm nhóm thông tin điều kiện ràng buộc thứ ba và nhóm thông tin điều kiện ràng buộc thứ tư. Tập con thứ nhất của các cờ điều kiện ràng buộc bao gồm ít nhất một cờ điều kiện ràng buộc thứ ba và ít nhất một cờ điều kiện ràng buộc thứ tư. Ít nhất một cờ điều kiện ràng buộc thứ ba được phân loại thành nhóm thông tin điều kiện ràng buộc thứ ba, và ít nhất một cờ điều kiện ràng buộc thứ tư được phân loại thành nhóm thông tin điều kiện ràng buộc thứ tư. Cờ công thứ

ba của nhóm thông tin điều kiện ràng buộc thứ ba có thể chỉ báo rằng ít nhất một cờ điều kiện ràng buộc thứ ba (hoặc nhóm thông tin điều kiện ràng buộc thứ ba) có được hiện diện (ví dụ, được báo hiệu) trong thông tin điều kiện ràng buộc hay không. Cờ công thứ tư của nhóm thông tin điều kiện ràng buộc thứ tư có thể chỉ báo rằng ít nhất một cờ điều kiện ràng buộc thứ tư (hoặc nhóm thông tin điều kiện ràng buộc thứ tư) có được hiện diện (ví dụ, được báo hiệu) trong thông tin điều kiện ràng buộc hay không. Ngoài ra, thông tin điều kiện ràng buộc có thể bao gồm tập con thứ hai của các cờ điều kiện ràng buộc. Trong ví dụ của sáng chế, cờ công thứ ba và cờ công thứ tư chỉ báo rằng nhóm thông tin điều kiện ràng buộc thứ ba và nhóm thông tin điều kiện ràng buộc thứ tư không được báo hiệu trong thông tin điều kiện ràng buộc, và do đó thông tin điều kiện ràng buộc bao gồm tập con thứ hai của các cờ điều kiện ràng buộc và không bao gồm ít nhất một cờ điều kiện ràng buộc thứ ba và ít nhất một cờ điều kiện ràng buộc thứ tư. Trong ví dụ của sáng chế, cờ công thứ ba và cờ công thứ tư chỉ báo rằng nhóm thông tin điều kiện ràng buộc thứ ba và nhóm thông tin điều kiện ràng buộc thứ tư được báo hiệu trong thông tin điều kiện ràng buộc, và do đó thông tin điều kiện ràng buộc bao gồm tập con thứ hai của các cờ điều kiện ràng buộc, ít nhất một cờ điều kiện ràng buộc thứ ba và ít nhất một cờ điều kiện ràng buộc thứ tư.

Các Fig.14A-14B thể hiện cấu trúc cú pháp thông tin điều kiện ràng buộc chung ví dụ 1400 theo phương án của sáng chế. Cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1400 có thể biểu diễn thông tin điều kiện ràng buộc (ví dụ, được gọi là thông tin điều kiện ràng buộc chung). Thông tin điều kiện ràng buộc chung (hoặc cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1400) có thể bao gồm một hoặc nhiều cờ công, như cờ công (ví dụ, `general_frame_structure_constraint_group_flag`) 1401, cờ công (ví dụ, `high_level_functionality_constraint_group_flag`) 1402, cờ công (ví dụ, `scalability_constraint_group_flag`) 1403, cờ công (ví dụ, `partitioning_constraint_group_flag`) 1404, cờ công (ví dụ, `intra_coding_tool_constraint_group_flag`) 1405, cờ công (ví dụ,

inter_coding_tool_constraint_group_flag) 1406, cờ công (ví dụ, transform_constraint_group_flag) 1407, cờ công (ví dụ, inloop_filtering_constraint_group_flag) 1408 trong FIG.14A. Một hoặc nhiều cờ công (ví dụ, các cờ công 1401-1408) có thể được hiện diện tại phần đầu của cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1400, như được thể hiện trên FIG.14A.

Cờ công (ví dụ, general_frame_structure_constraint_group_flag) 1401 được kết hợp với nhóm thông tin điều kiện ràng buộc 1410, và được kết hợp với các cờ điều kiện ràng buộc 1411-1414 mà nằm trong nhóm thông tin điều kiện ràng buộc 1410. Cờ công (ví dụ, general_frame_structure_constraint_group_flag) 1401 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc 1411-1414 mà nằm trong nhóm thông tin điều kiện ràng buộc 1410 có thể được hiện diện.

Nhóm thông tin điều kiện ràng buộc 1410 (hoặc các cờ điều kiện ràng buộc 1411-1414) có thể liên quan đến nguồn đầu vào và gói khung (ví dụ khung được đóng gói hoặc được dự đoán). Viện dẫn tới FIG.14A, các cờ điều kiện ràng buộc 1411-1414 tương ứng với general_non_packed_constraint_flag 1411, general_frame_only_constraint_flag 1412, general_non_projected_constraint_flag 1413, và general_one_picture_only_constraint_flag 1414. Nếu không phải, cờ công (ví dụ, general_frame_structure_constraint_group_flag) 1401 bằng 0 có thể chỉ rõ rằng các cờ điều kiện ràng buộc 1411-1414 mà nằm trong nhóm thông tin điều kiện ràng buộc 1410 có thể không được hiện diện.

Viện dẫn tới FIG.14B, cờ công (ví dụ, high_level_functionality_constraint_group_flag) 1402 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến chức năng mức cao (ví dụ tái lấy mẫu ảnh tham chiếu) mà nằm trong nhóm thông tin điều kiện ràng buộc 1420 có thể được hiện diện. Nếu không phải, cờ công (ví dụ, high_level_functionality_constraint_group_flag) 1402 bằng 0 có thể chỉ rõ rằng

các cờ điều kiện ràng buộc mà nằm trong nhóm thông tin điều kiện ràng buộc 1420 có thể không được hiện diện.

Viện dẫn lại FIG.14A, cờ công (ví dụ, scalability_constraint_group_flag) 1403 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến khả năng biến đổi tỷ lệ (ví dụ dự đoán liên lớp) có thể được hiện diện. Nếu không phải, các cờ điều kiện ràng buộc liên quan đến khả năng biến đổi tỷ lệ có thể không được hiện diện.

Cờ công (ví dụ, partitioning_constraint_group_flag) 1404 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến phân chia mức cao (ví dụ ảnh con hoặc ô) có thể được hiện diện. Nếu không phải, các cờ điều kiện ràng buộc liên quan đến phân chia mức cao có thể không được hiện diện.

Cờ công (ví dụ, intra_coding_tool_constraint_group_flag) 1405 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến mã hóa nội bộ (ví dụ việc nội dự đoán) có thể được hiện diện. Nếu không phải, các cờ điều kiện ràng buộc liên quan đến mã hóa nội bộ có thể không được hiện diện.

Cờ công (ví dụ, inter_coding_tool_constraint_group_flag) 1406 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến mã hóa liên đới (ví dụ bù chuyển động đối với dự đoán liên ảnh) có thể được hiện diện. Nếu không phải, các cờ điều kiện ràng buộc liên quan đến mã hóa liên đới có thể không được hiện diện.

Cờ công (ví dụ, transfrom_constraint_group_flag) 1407 bằng 1 có thể chỉ rõ rằng các cờ điều kiện ràng buộc liên quan đến việc mã hóa biến đổi (ví dụ ma trận đa biến đổi) có thể được hiện diện. Nếu không phải, các cờ điều kiện ràng buộc liên quan đến mã hóa biến đổi có thể không được hiện diện.

Trong phương án của sáng chế, khi tất cả các cờ công (ví dụ, các cờ công 1401-1408 trong FIG.14A) bằng 0, không có các cờ điều kiện ràng buộc được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc (ví dụ, cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1400).

Theo các khía cạnh của sáng chế, cú pháp có thể được thiết kế sao cho thông tin điều khiển bao gồm các cờ công (ví dụ, các cờ công 1401-1408), các

cờ điều kiện ràng buộc được kết hợp (ví dụ, các cờ điều kiện ràng buộc 1411-1412 và các cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc 1420), thông tin điều khiển bổ sung, và/hoặc loại tương tự có thể được căn thẳng theo byte, ví dụ, số lượng cờ có thể chia cho 8 để đảm bảo việc căn thẳng theo byte. Trong ví dụ của sáng chế, số lượng cờ cổng và các cờ điều kiện ràng buộc trong thông tin điều kiện ràng buộc (ví dụ, cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1400) có thể chia cho 8. Cơ chế căn thẳng theo byte có thể được sử dụng để thực hiện việc căn thẳng theo byte của thông tin điều khiển. Việc dẫn tới FIG.14B, cú pháp (ví dụ, vòng lặp “while”) 1430 có thể được sử dụng cho việc căn thẳng theo byte.

Trong phương án của sáng chế, thông tin độ dịch như độ dịch (ví dụ, phần tử cú pháp `constraint_info_offset[]`) và thông tin độ dài như độ dài (ví dụ, phần tử cú pháp `constraint_info_length[]`) được hiện diện trong thông tin điều kiện ràng buộc (ví dụ, tại phần đầu của cấu trúc cú pháp thông tin điều kiện ràng buộc chung) mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng được hiện diện trong thông tin điều kiện ràng buộc. Trong phương án của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa. Độ dịch và độ dài có thể được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong mỗi một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc. Số lượng độ dịch có thể được báo hiệu trong dòng bit video được mã hóa.

Trong phương án của sáng chế, 0 độ dịch hoặc nhiều hơn (hoặc các độ dịch thông tin điều kiện ràng buộc) (ví dụ, được chỉ báo bởi phần tử cú pháp `constraint_info_offset[]`) và 0 độ dài hoặc nhiều hơn (hoặc các độ dài thông tin điều kiện ràng buộc) tương ứng (ví dụ, được chỉ báo bởi phần tử cú pháp `constraint_info_length[]`) có thể được hiện diện trong thông tin điều kiện ràng buộc, như tại phần đầu của cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Số lượng của 0 độ dịch thông tin điều kiện ràng buộc hoặc nhiều hơn và

số lượng của 0 độ dài thông tin điều kiện ràng buộc hoặc nhiều hơn có thể được chỉ báo rõ ràng bởi các phân tử cú pháp.

Trong ví dụ của sáng chế, số lượng của 0 độ dịch thông tin điều kiện ràng buộc hoặc nhiều hơn và số lượng của 0 độ dài thông tin điều kiện ràng buộc hoặc nhiều hơn là đồng nhất, và số lượng của 0 độ dịch thông tin điều kiện ràng buộc hoặc nhiều hơn (được biểu diễn bởi M) được chỉ báo rõ ràng (hoặc được báo hiệu) bởi phân tử cú pháp (ví dụ, `num_constraint_info_set`). M có thể là số nguyên mà lớn hơn hoặc bằng 0. M bằng 0 có thể chỉ báo rằng không có độ dịch thông tin điều kiện ràng buộc và không có độ dài thông tin điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, độ dịch thông tin điều kiện ràng buộc (ví dụ, phân tử cú pháp `constraint_info_offset[i]`) và độ dài thông tin điều kiện ràng buộc (ví dụ, phân tử cú pháp `constraint_info_length[i]`) có thể chỉ rõ các cờ điều kiện ràng buộc khả dụng mà được hiện diện trong thông tin điều kiện ràng buộc (ví dụ, cấu trúc cú pháp thông tin điều kiện ràng buộc chung). Trong ví dụ của sáng chế, khi giá trị của độ dịch thông tin điều kiện ràng buộc (ví dụ, phân tử cú pháp `constraint_info_offset[i]`) bằng 5, và giá trị của độ dài thông tin điều kiện ràng buộc (ví dụ, phân tử cú pháp `constraint_info_length[i]`) bằng 3, các cờ điều kiện ràng buộc thứ năm, thứ sáu và thứ bảy được hiện diện trong thông tin điều kiện ràng buộc (ví dụ, cấu trúc cú pháp thông tin điều kiện ràng buộc chung).

Trong ví dụ của sáng chế, nếu M bằng 0, không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc (ví dụ, cấu trúc cú pháp thông tin điều kiện ràng buộc chung).

Trong ví dụ của sáng chế, mã hóa độ dài chạy có thể được sử dụng để mã hóa các cờ điều kiện ràng buộc mà được chỉ rõ trong thứ tự định trước (hoặc thứ tự đã định).

Trong phương án của sáng chế, việc mã hóa chạy có thể được sử dụng trong đó các cờ điều kiện ràng buộc được chỉ rõ trong thứ tự định trước (hoặc thứ tự đã định). Thay vì mã hóa cờ điều kiện ràng buộc một cách trực tiếp, danh

sách được mã hóa thích hợp của các giá trị "bỏ qua" có thể chỉ báo các cờ điều kiện ràng buộc mà bằng 0, với cờ điều kiện ràng buộc sau đây được giả định bằng 1. Việc mã hóa chạy nêu trên có thể là đặc biệt hiệu quả nếu (i) số lượng của các cờ điều kiện ràng buộc là lớn và (ii) phần trăm nhỏ của các cờ điều kiện ràng buộc bằng 1.

Trong phương án của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit được mã hóa. Các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được báo hiệu theo thứ tự định trước. Do đó, các cờ điều kiện ràng buộc có thể được mã hóa chạy (ví dụ, được mã hóa chạy hoặc được giải mã chạy). Ngoài ra, thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa trên các cờ điều kiện ràng buộc.

Trong phương án của sáng chế, ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công bao gồm các cờ điều kiện ràng buộc được báo hiệu theo thứ tự định trước. Do đó, các cờ điều kiện ràng buộc có thể được mã hóa chạy (ví dụ, được mã hóa chạy hoặc được giải mã chạy).

Trong phương án của sáng chế, danh sách đầy đủ của các cờ điều kiện ràng buộc có thể được chỉ rõ trong tiêu chuẩn mã hóa video (ví dụ, tiêu chuẩn kỹ thuật VVC), Bảng phía ngoài, hoặc loại tương tự. Trong ví dụ của sáng chế, chỉ các cờ điều kiện ràng buộc khả dụng của các cờ điều kiện ràng buộc được chỉ báo, ví dụ, bởi một hoặc nhiều trong số phần sau đây: số lượng các cờ điều kiện ràng buộc khả dụng (ví dụ, `num_available_constraint_flags`), các cờ công (hoặc các cờ hiện diện nhóm điều kiện ràng buộc), thông tin độ dịch thông tin điều kiện ràng buộc và thông tin độ dài thông tin điều kiện ràng buộc, hoặc loại tương tự được hiện diện trong dòng video được mã hóa.

Trong ví dụ của sáng chế, danh sách đầy đủ của các cờ điều kiện ràng buộc được chỉ rõ và là khả dụng đối với bộ mã hóa và bộ giải mã. Danh sách đầy đủ của các cờ điều kiện ràng buộc có thể được lưu trữ tại bộ giải mã. Danh sách đầy đủ của các cờ điều kiện ràng buộc có thể bao gồm 100 cờ điều kiện

ràng buộc. 10 trong số 100 cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc đối với CLVS và do đó là khả dụng đối với tập con của các khối mã hóa trong CLVS. 10 trong số 100 cờ điều kiện ràng buộc được gọi là 10 cờ điều kiện ràng buộc khả dụng. Trong ví dụ của sáng chế, số lượng các cờ điều kiện ràng buộc khả dụng (ví dụ, 10) được báo hiệu. Trong ví dụ của sáng chế, 10 cờ điều kiện ràng buộc khả dụng nằm trong hai nhóm thông tin điều kiện ràng buộc và được kiểm soát bởi cờ cổng thứ nhất và cờ cổng thứ hai. Do đó, cờ cổng thứ nhất và cờ cổng thứ hai có thể được báo hiệu để chỉ báo 10 cờ điều kiện ràng buộc khả dụng.

Trong ví dụ của sáng chế, độ dịch thông tin điều kiện ràng buộc thứ nhất (ví dụ, phần tử cú pháp `constraint_info_offset[0]`) và độ dài thông tin điều kiện ràng buộc thứ nhất (ví dụ, phần tử cú pháp `constraint_info_length[0]`) được báo hiệu. Độ dịch thông tin điều kiện ràng buộc thứ hai (ví dụ, phần tử cú pháp `constraint_info_offset[1]`) và độ dài thông tin điều kiện ràng buộc thứ hai (ví dụ, phần tử cú pháp `constraint_info_length[1]`) được báo hiệu. Ví dụ, phần tử cú pháp `constraint_info_offset[0]` là 15 và phần tử cú pháp `constraint_info_length[0]` là 3, và phần tử cú pháp `constraint_info_offset[1]` là 82 và phần tử cú pháp `constraint_info_length[1]` là 7, và do đó chỉ báo rằng các cờ điều kiện ràng buộc thứ 15 đến thứ 17 và các cờ điều kiện ràng buộc thứ 82 đến thứ 88 trong danh sách đầy đủ (ví dụ, 100 cờ điều kiện ràng buộc) là khả dụng hoặc hiện diện trong thông tin điều kiện ràng buộc.

Trong phương án của sáng chế, bất kỳ các kỹ thuật khác nhau (hoặc các phương pháp, các phương án, các ví dụ) để mã hóa hiệu quả các cờ điều kiện ràng buộc có thể được kết hợp, mà sử dụng thông tin điều khiển thích hợp. Việc kết hợp có thể là kết hợp thích hợp của hai hoặc nhiều hơn trong số các kỹ thuật này. Ngoài ra, một trong số các kỹ thuật khác nhau (hoặc các phương pháp, các phương án, các ví dụ) có thể được sử dụng một cách độc lập. Các cờ điều kiện ràng buộc có thể được nhóm lại. Trong các nhóm định trước, việc mã hóa chạy có thể được sử dụng trong khi các nhóm khác có thể sử dụng mã hóa nhị phân thẳng.

Giá trị của số lượng lớn nhất của các cờ điều kiện ràng buộc (ví dụ, `MaxNumConstraintFlags`) có thể được xác định trước là 16, 32, 64, 128, hoặc loại tương tự.

Giá trị của số lượng lớn nhất của các cờ điều kiện ràng buộc (ví dụ, `MaxNumConstraintFlags`) có thể được xác định bởi thông tin mẫu, như `general_profile_idc` hoặc `general_sub_profile_idc`, hoặc thông tin phiên bản mã hóa-giải mã, sao cho phạm vi của số lượng các cờ điều kiện ràng buộc (ví dụ, `num_available_constraint_flags` 1301) có thể bị giới hạn bởi thông tin mẫu hoặc thông tin phiên bản. Ví dụ, giá trị của số lượng của các cờ điều kiện ràng buộc (ví dụ, `num_available_constraint_flags` 1301) trong mẫu chính (ví dụ, trong đó `MaxNumConstraintFlags` = 64) có thể nằm trong khoảng từ 0 đến 64, trong khi giá trị của số lượng của các cờ điều kiện ràng buộc (ví dụ, `num_available_constraint_flags` 1301) trong mẫu cải tiến (ví dụ, trong đó `MaxNumConstraintFlags` = 128) có thể nằm trong khoảng từ 0 đến 128.

Trong phương án của sáng chế, giá trị của số lượng của các cờ điều kiện ràng buộc (ví dụ, `num_available_constraint_flags`) có thể được giả định bằng giá trị được xác định trước bởi thông tin mẫu, như `general_profile_idc` hoặc `general_sub_profile_idc`, hoặc thông tin phiên bản mã hóa-giải mã, sao cho giá trị của `num_available_constraint_flags` có thể được xác định mà không được báo hiệu rõ ràng.

Trong phương án của sáng chế, trong FIG.12, thông tin byte dành riêng có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Ví dụ, các cờ `gci_num_reserved_bytes` 1203 và `gci_reserved_bytes[]` 1204 có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung đối với phần mở rộng của cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Cờ `gci_num_reserved_bytes` có thể chỉ rõ số lượng byte điều kiện ràng buộc dành riêng. Trong ví dụ của sáng chế, các byte điều kiện ràng buộc dành riêng dùng để báo hiệu các cờ bổ sung (ví dụ, các cờ điều kiện ràng buộc bổ sung). Cờ `gci_reserved_byte[ ]` có thể có giá trị thích hợp bất kỳ.

Trong phương án của sáng chế, giá trị của `gci_num_reserved_bytes` có thể được giới hạn hoặc được xác định bởi thông tin mẫu, như `general_profile_idc` hoặc `general_sub_profile_idc`, hoặc thông tin phiên bản mã hóa-giải mã. Với mẫu cơ sở (hoặc mẫu chính), giá trị của `gci_num_reserved_bytes` có thể bằng 0. Với mẫu mở rộng (hoặc mẫu cải tiến), giá trị của `gci_num_reserved_bytes` có thể lớn hơn 0.

Cờ chuỗi trường có thể được báo hiệu trong dòng bit video được mã hóa. Cờ chuỗi trường có thể chỉ báo rằng các ảnh trong lớp đầu ra có được mã hóa với mã hóa trường hay không. Cờ chuỗi trường có thể được báo hiệu trong SPS như là `sps_field_seq_flag`. Trong phương án của sáng chế, cờ `sps_field_seq_flag` có thể được hiện diện trong SPS. Cờ `sps_field_seq_flag` bằng 1 có thể chỉ báo rằng CLVS mà các ảnh mà biểu diễn các trường. Cờ `sps_field_seq_flag` bằng 0 có thể chỉ báo rằng CLVS mang các ảnh mà biểu diễn khung.

Trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12, cờ `general_frame_only_constraint_flag` có thể được hiện diện. Cờ `general_frame_only_constraint_flag` bằng 1 có thể chỉ rõ rằng phạm vi đối với tập hợp lớp đầu ra (ví dụ, `OlsInScope`) mà các ảnh mà biểu diễn khung. Cờ `general_frame_only_constraint_flag` bằng 0 chỉ rõ rằng phạm vi đối với tập hợp lớp đầu ra (ví dụ, `OlsInScope`) mang các ảnh mà có thể có hoặc có thể không biểu diễn các khung. Trong phương án của sáng chế, cờ `general_frame_only_constraint_flag` chỉ báo rằng các ảnh trong tập hợp lớp đầu ra có được mã hóa với mã hóa trường hay không. Tập hợp lớp đầu ra có thể bao gồm tập con của các khối mã hóa. Cờ `sps_field_seq_flag` có thể là sai dựa trên cờ `general_frame_only_constraint_flag` (ví dụ, bằng 1) mà chỉ báo rằng tập con của các ảnh không được mã hóa với mã hóa trường. Tập con của các ảnh có thể nằm trong một lớp của tập hợp lớp đầu ra.

Khi cờ `general_frame_only_constraint_flag` bằng 1, giá trị của cờ `sps_field_seq_flag` có thể bằng 0.

Trong phương án của sáng chế, cờ `pps_mixed_nalu_types_in_pic_flag` có thể được hiện diện trong PPS. Cờ `pps_mixed_nalu_types_in_pic_flag` bằng 1 có thể chỉ rõ rằng mỗi ảnh mà viện dẫn tới PPS có nhiều hơn một đơn vị VCL NAL và các đơn vị VCL NAL không có cùng giá trị của `nal_unit_type`. Cờ `pps_mixed_nalu_types_in_pic_flag` bằng 0 có thể chỉ rõ rằng mỗi ảnh mà viện dẫn tới PPS có một hoặc nhiều đơn vị VCL NAL và các đơn vị VCL NAL của mỗi ảnh mà viện dẫn tới PPS có cùng giá trị của `nal_unit_type`. Trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12, cờ `no_mixed_nalu_types_in_pic_constraint_flag` có thể được hiện diện. Cờ `no_mixed_nalu_types_in_pic_constraint_flag` bằng 1 có thể chỉ rõ rằng giá trị của `pps_mixed_nalu_types_in_pic_flag` sẽ bằng 0. Cờ `no_mixed_nalu_types_in_pic_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này.

Trong phương án của sáng chế, cờ `general_one_picture_only_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. `general_one_picture_only_constraint_flag` bằng 1 có thể chỉ rõ rằng chỉ có một ảnh được mã hóa trong dòng bit. Cờ `general_one_picture_only_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này.

Trong phương án của sáng chế, cờ `single_layer_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ `single_layer_constraint_flag` bằng 1 có thể chỉ rõ rằng `sps_video_parameter_set_id` sẽ bằng 0. Cờ `single_layer_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ `general_one_picture_only_constraint_flag` bằng 1, giá trị của cờ `single_layer_constraint_flag` có thể bằng 1.

Trong phương án của sáng chế, cờ `all_layers_independent_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ `all_layers_independent_constraint_flag` bằng 1 có thể chỉ rõ rằng cờ

vps_all_independent_layers_flag có thể bằng 1. Còn all_layers_independent_constraint_flag bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ single_layer_constraint_flag bằng 1, giá trị của cờ all_layers_independent_constraint_flag có thể bằng 1.

Trong phương án của sáng chế, cờ no_res_change_in_clvs_constraint_flag có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ no_res_change_in_clvs_constraint_flag bằng 1 có thể chỉ rõ rằng cờ sps_res_change_in_clvs_allowed_flag có thể bằng 0. Còn no_res_change_in_clvs_constraint_flag bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ no_ref_pic_resampling_constraint_flag bằng 1, giá trị của cờ no_res_change_in_clvs_constraint_flag có thể bằng 1.

Trong phương án của sáng chế, cờ no_mixed_nalu_types_in_pic_constraint_flag có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ no_mixed_nalu_types_in_pic_constraint_flag bằng 1 chỉ rõ rằng giá trị của cờ pps_mixed_nalu_types_in_pic_flag có thể bằng 0. Còn no_mixed_nalu_types_in_pic_constraint_flag bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ one_subpic_per_pic_constraint_flag bằng 1, giá trị của cờ no_mixed_nalu_types_in_pic_constraint_flag có thể bằng 1.

Trong phương án của sáng chế, cờ no_trail_constraint_flag có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ no_trail_constraint_flag bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với nuh_unit_type bằng TRAIL_NUT được hiện diện trong OlsInScope. Còn no_trail_constraint_flag bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ general_one_picture_only_constraint_flag bằng 1, cờ no_trail_constraint_flag có thể bằng 1.

Trong phương án của sáng chế, cờ no_stsa_constraint_flag có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ no_stsa_constraint_flag bằng 1 có thể chỉ rõ rằng có thể không có đơn vị

NAL với `nuh_unit_type` bằng `STSA_NUT` được hiện diện trong `OlsInScope`. Còn `no_stsa_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi `general_one_picture_only_constraint_flag` bằng 1, `no_stsa_constraint_flag` có thể bằng 1.

Trong phương án của sáng chế, `no_trail_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Còn `no_trail_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `TRAIL_NUT` được hiện diện trong `OlsInScope`. Còn `no_trail_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi `general_one_picture_only_constraint_flag` bằng 1, `no_trail_constraint_flag` có thể bằng 1.

Trong phương án của sáng chế, `no_stsa_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Còn `no_stsa_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `STSA_NUT` được hiện diện trong `OlsInScope`. Còn `no_stsa_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi `general_one_picture_only_constraint_flag` bằng 1, `no_stsa_constraint_flag` có thể bằng 1.

Trong phương án của sáng chế, `no_idr_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. `no_idr_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `IDR_W_RADL` hoặc `IDR_N_LP` được hiện diện trong `OlsInScope`. Còn `no_idr_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này.

Trong phương án của sáng chế, `no_cra_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Còn `no_cra_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `CRA_NUT` được hiện diện trong `OlsInScope`. Còn `no_cra_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này.

Trong phương án của sáng chế, cờ `no_rasl_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ `no_rasl_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `RASL_NUT` được hiện diện trong `OlsInScope`. Cờ `no_rasl_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ `no_cra_constraint_flag` bằng 1, giá trị của cờ `no_rasl_constraint_flag` có thể bằng 1.

Trong phương án của sáng chế, cờ `no_radl_constraint_flag` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung trong FIG 12. Cờ `no_radl_constraint_flag` bằng 1 có thể chỉ rõ rằng có thể không có đơn vị NAL với `nuh_unit_type` bằng `RADL_NUT` được hiện diện trong `OlsInScope`. Cờ `no_radl_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi cờ `no_idr_constraint_flag` bằng 1 và cờ `no_cra_constraint_flag` bằng 1, giá trị của cờ `no_rasl_constraint_flag` có thể bằng 1.

Các Fig.15A-15D thể hiện cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 ví dụ theo phương án của sáng chế. Cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 có thể biểu diễn thông tin điều kiện ràng buộc (ví dụ, được gọi là thông tin điều kiện ràng buộc chung). Thông tin điều kiện ràng buộc chung (hoặc cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500) có thể bao gồm cờ (ví dụ, `gci_present_flag`) mà chỉ báo rằng các cờ điều kiện ràng buộc có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không. Trong ví dụ của sáng chế, nếu cờ (ví dụ, `gci_present_flag`) là đúng, một hoặc nhiều cờ điều kiện ràng buộc có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500. Nếu cờ (ví dụ, `gci_present_flag`) là sai, không có các cờ điều kiện ràng buộc có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500.

Như được mô tả nêu trên có viện dẫn tới các Fig.14A-14B, các cờ điều kiện ràng buộc trong các Fig.15A-15D có thể được phân loại thành các nhóm (ví dụ, các nhóm thông tin điều kiện ràng buộc). Việc các cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc có thể được hiện diện trong cấu trúc

cú pháp thông tin điều kiện ràng buộc chung 1500 hay không có thể được điều khiển, ví dụ, bởi cờ cổng tương ứng đối với nhóm thông tin điều kiện ràng buộc.

Cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 có thể bao gồm bất kỳ số lượng thích hợp của các nhóm thông tin điều kiện ràng buộc. Trong ví dụ của sáng chế, cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 có thể còn bao gồm các cờ điều kiện ràng buộc mà không được chứa trong các nhóm thông tin điều kiện ràng buộc, và do đó các cờ điều kiện ràng buộc nằm ngoài các nhóm thông tin điều kiện ràng buộc.

Trong ví dụ được thể hiện trong các Fig.15A-15D, cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 bao gồm chín nhóm thông tin điều kiện ràng buộc 1510-1518. Nhóm thông tin điều kiện ràng buộc 1510 liên quan đến thông tin điều kiện ràng buộc chung và bao gồm ba cờ điều kiện ràng buộc (ví dụ, `gci_intra_only_constraint_flag`, `gci_all_layers_independent_constraint_flag`, `gci_one_a_u_only_constraint_flag`). Trong ví dụ của sáng chế, cờ cổng có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1510 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1511 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc khuôn dạng ảnh. Trong ví dụ của sáng chế, cờ cổng có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1511 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1512 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc loại đơn vị NAL. Trong ví dụ của sáng chế, cờ cổng có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1512 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1513 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc phân chia ảnh con lát hoặc ô. Trong ví dụ của sáng chế, cờ cổng có thể kiểm soát hoặc điều khiển

rằng nhóm thông tin điều kiện ràng buộc 1513 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1514 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc phân chia khối và CTU. Trong ví dụ của sáng chế, cờ công có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1514 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1515 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc mã hóa nội bộ. Trong ví dụ của sáng chế, cờ công có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1515 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1516 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc mã hóa liên đới. Trong ví dụ của sáng chế, cờ công có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1516 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1517 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc biến đổi, lượng tử hóa và phần dư. Trong ví dụ của sáng chế, cờ công có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1517 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Nhóm thông tin điều kiện ràng buộc 1518 và các cờ điều kiện ràng buộc tương ứng liên quan đến thông tin điều kiện ràng buộc bộ lọc vòng. Trong ví dụ của sáng chế, cờ công có thể kiểm soát hoặc điều khiển rằng nhóm thông tin điều kiện ràng buộc 1518 có được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 hay không.

Trong phương án của sáng chế, trong FIG.15D, các phần tử cú pháp `gci_num_reserved_bits` và `gci_reserved_zero_bit[]` có thể được hiện diện trong cấu trúc cú pháp thông tin điều kiện ràng buộc chung 1500 đối với phần mở

rộng của cấu trúc cú pháp thông tin điều kiện ràng buộc chung. Phần tử cú pháp `gci_num_reserved_bits` có thể chỉ rõ số lượng bit điều kiện ràng buộc dành riêng. Phần tử cú pháp `gci_reserved_zero_bit[ ]` có thể có giá trị thích hợp bất kỳ. Trong ví dụ của sáng chế, sự hiện diện của phần tử cú pháp `gci_reserved_zero_bit[ ]` và giá trị của `gci_reserved_zero_bit[ ]` không ảnh hưởng tới xử lý giải mã được chỉ rõ trong tiêu chuẩn mã hóa video định trước (ví dụ, tiêu chuẩn kỹ thuật VVC). Sự hiện diện của phần tử cú pháp `gci_reserved_zero_bit[ ]` và giá trị của `gci_reserved_zero_bit[ ]` có thể được sử dụng đối với một vài tiêu chuẩn mã hóa video.

FIG.11 thể hiện ví dụ về tập hợp của các phần tử cú pháp PTL trong cấu trúc cú pháp PTL theo phương án của sáng chế. Tập hợp của các phần tử cú pháp PTL có thể bao gồm `general_profile_idc`, `general_tier_flag`, `general_level_idc`, `ptl_frame_only_constraint_flag`, `ptl_multilayer_enabled_flag`, `plt_sublayer_level_present_flag[ ]`, `ptl_reserved_zero_bit`, `sublayer_level_idc[ ]`, `ptl_num_sub_profiles`, `general_sub_profile_idc`, và/hoặc thông tin điều kiện ràng buộc chung. Việc một vài (ví dụ, `general_profile_idc`, `general_tier_flag`, thông tin điều kiện ràng buộc chung, `ptl_num_sub_profiles`) tập hợp của phần tử cú pháp PTL có thể được hiện diện hay không có thể được dựa trên các giá trị của cờ (ví dụ, `profileTierPresentFlag`).

Chuỗi video với một hoặc nhiều lớp có thể được mã hóa như là cấu trúc mã hóa trường trong đó hai hoặc nhiều hơn các ảnh trường liên tiếp có thể bao gồm ảnh khung toàn phần. Thông tin mã hóa trường có thể được báo hiệu trong cấu trúc cú pháp mức cao, như các tập hợp tham số hoặc các bản tin SEI.

Trong phương án của sáng chế, mọi khung video được đan xen có hai trường đối với mỗi khung. Trong một vài ví dụ, việc giảm ba trên hai (giảm 3:2) có thể được sử dụng để chuyển đổi 24 khung trên giây thành 29,97 hoặc 30 khung trên giây. Việc giảm 3:2 có thể chuyển đổi bốn khung thành năm khung cộng việc giảm tốc độ, một cách xấp xỉ.

Trong phương án của sáng chế, cờ (ví dụ, `sps_field_seq_flag`) có thể được báo hiệu trong SPS, như được thể hiện trên FIG.17, để chỉ báo rằng việc

mã hóa trường có được sử dụng hay không. Cờ (ví dụ, `sps_field_seq_flag`) bằng 1 có thể chỉ báo rằng CLVS mà các ảnh mà biểu diễn các trường. Cờ (ví dụ, `sps_field_seq_flag` bằng 0) có thể chỉ báo rằng CLVS mang các ảnh mà biểu diễn khung. Khi cờ (ví dụ, `general_frame_only_constraint_flag`) bằng 1, giá trị của cờ (ví dụ, `sps_field_seq_flag`) có thể bằng 0. Trong ví dụ của sáng chế, cờ (ví dụ, `general_frame_only_constraint_flag`) bằng 1 chỉ báo rằng giá trị của cờ (ví dụ, `sps_field_seq_flag`) bằng 0.

Trong phương án của sáng chế, khi cờ (ví dụ, `sps_field_seq_flag`) bằng 1, bản tin SEI thông tin trường khung có thể được hiện diện đối với mọi ảnh được mã hóa trong CLVS. Tiêu chuẩn hoặc kỹ thuật mã hóa video có thể yêu cầu sự hiện diện này.

Trong phương án của sáng chế, xử lý giải mã có thể không xử lý các ảnh mà biểu diễn các trường hoặc các khung khác với các ảnh khác. Chuỗi của các ảnh mà biểu diễn các trường có thể được mã hóa với các chiều ảnh của trường riêng biệt. Ví dụ, các ảnh mà biểu diễn các trường 1080i có thể có các chiều đầu ra được cắt bằng 1920x540, trong khi tốc độ ảnh chuỗi có thể biểu diễn tốc độ của các trường nguồn (ví dụ, thông thường nằm giữa 50 và 60 Hz), thay vì tốc độ khung nguồn (ví dụ, thông thường nằm giữa 25 và 30 Hz).

Trong phương án của sáng chế, giá trị của cờ (ví dụ, `sps_field_seq_flag`) có thể là giống nhau qua các lớp. Trong ví dụ của sáng chế, giá trị của cờ (ví dụ, `sps_field_seq_flag`) có thể là giống nhau trong tất cả các SPS mà được viện dẫn bởi các CLVS trong CVS, ví dụ, do tất cả các lớp có thể có cùng cấu trúc trường hoặc cùng cấu trúc khung.

Trong phương án của sáng chế, giá trị của cờ (ví dụ, `sps_field_seq_flag`) của lớp A bằng giá trị của cờ (ví dụ, `sps_field_seq_flag`) của lớp phụ thuộc của lớp A trong tập hợp lớp đầu ra.

Trong phương án của sáng chế, khi cờ (ví dụ, `sps_field_seq_flag`) bằng 1, tất cả các trường mà bao gồm cùng khung có thể viện dẫn tới cùng PPS sao cho tất cả các trường của cùng khung có thể cùng kích cỡ ảnh, cửa sổ tương thích và phân chia ô.

Trong phương án của sáng chế, hai trường của khung có thể có cùng cấu trúc phân chia với cùng phần tử cú pháp ô hoặc cùng phần tử cú pháp ảnh con.

Trong phương án của sáng chế, hai trường của khung có thể có cấu trúc lớp đồng nhất với cùng số lượng lớp và lớp con.

Trong phương án của sáng chế, như được thể hiện trên FIG.18, cờ (ví dụ, `vps_field_seq_flag`) có thể được báo hiệu trong VPS để chỉ báo rằng các lớp mà viện dẫn tới VPS có được mã hóa như là trường hoặc khung hay không. Cờ (ví dụ, `vps_field_seq_flag`) bằng 1 có thể chỉ báo rằng CVS mà viện dẫn tới VPS mang các ảnh mà biểu diễn các trường. Cờ (ví dụ, `vps_field_seq_flag`) bằng 0 có thể chỉ báo rằng CVS mà viện dẫn tới VPS mang các ảnh mà biểu diễn các khung. Khi cờ (ví dụ, `general_frame_only_constraint_flag`) bằng 1, giá trị của cờ (ví dụ, `sps_field_seq_flag`) sẽ bằng 0.

Khi cờ (ví dụ, `vps_field_seq_flag`) được hiện diện trong VPS, giá trị của cờ (ví dụ, `sps_field_seq_flag`) trong SPS mà viện dẫn tới VPS có thể bằng giá trị của cờ (ví dụ, `vps_field_seq_flag`).

Xóa Cú pháp mức cao trong SPS

Việc xóa cú pháp mức cao (HLS-high level syntax) sau đây có thể được thực hiện trong tập tham số chuỗi (SPS-Sequence Parameter Set) theo các phương án của sáng chế: (1) Báo hiệu `sps_picture_header_in_slice_header_flag`; (2) Căn chỉnh trên giá trị `subpic_treated_as_pic_flag` qua các lớp; (3) Điều kiện ràng buộc trên số lượng ảnh con đối với ảnh nhỏ; (4) Hiệu chỉnh trên khoảng giá trị của `sps_subpic_id_len_minus1`.

1. Báo hiệu `picture_header_in_slice_header_flag`

Khi thông tin tiêu đề ảnh được chứa trong thông tin tiêu đề lát, giả thiết rằng thông tin ảnh con không được sử dụng và chỉ một lát được hiện diện theo ảnh trong CLVS mà viện dẫn tới tập hợp tham số chuỗi (SPS-sequence parameter set). Giả định này được thỏa thuận theo giả thiết, nhưng không được mô tả rõ ràng. Để thực hiện việc xóa dòng trực tiếp HTTP, cờ mới `sps_picture_header_in_slice_header_flag` có thể được xác định và được sử dụng

trong SPS. `sps_picture_header_in_slice_header_flag` bằng 1 chỉ rõ rằng cấu trúc cú pháp thông tin tiêu đề ảnh (PH) được hiện diện trong thông tin tiêu đề lát và chỉ một lát được hiện diện trên ảnh trong CLVS. Ngoài ra, không có thông tin ảnh con được hiện diện khi `sps_picture_header_in_slice_header_flag` bằng 1.

Phần sau đây là ngữ nghĩa ví dụ và các định nghĩa cú pháp đối với cờ mới `sps_picture_header_in_slice_header_flag`:

```
seq_parameter_set_rbsp( ) { Descriptor
    ...
    sps_picture_header_in_slice_header_flag    u(1)
    if( !sps_picture_header_in_slice_header_flag )
        subpic_info_present_flag    u(1)
    if( subpic_info_present_flag ) {
        ...
    }
    ...
}
```

`sps_picture_header_in_slice_header_flag` bằng 1 chỉ rõ rằng cấu trúc cú pháp PH được hiện diện trong thông tin tiêu đề lát và chỉ một lát được hiện diện trên ảnh trong CLVS. `sps_picture_header_in_slice_header_flag` bằng 0 chỉ rõ rằng cấu trúc cú pháp PH không được hiện diện trong thông tin tiêu đề lát trong CLVS.

`subpic_info_present_flag` bằng 1 chỉ rõ rằng thông tin ảnh con được hiện diện đối với CLVS và có thể có một hoặc nhiều hơn một ảnh con trong mỗi ảnh của CLVS. `subpic_info_present_flag` bằng 0 chỉ rõ rằng thông tin ảnh con không được hiện diện đối với CLVS và chỉ có một ảnh con trong mỗi ảnh của CLVS. Khi không được hiện diện, giá trị của `subpic_info_present_flag` được giả định bằng 0.

Khi `res_change_in_clvs_allowed_flag` bằng 1, giá trị của `subpic_info_present_flag` sẽ bằng 0.

Trong một phương án, `picture_header_in_slice_header_flag` được báo hiệu trong thông tin tiêu đề ảnh.

`picture_header_in_slice_header_flag` bằng 1 chỉ rõ rằng cấu trúc cú pháp PH được hiện diện trong thông tin tiêu đề lát. `picture_header_in_slice_header_flag` bằng 0 chỉ rõ rằng cấu trúc cú pháp PH không được hiện diện trong thông tin tiêu đề lát.

Giá trị của `picture_header_in_slice_header_flag` sẽ bằng `sps_picture_header_in_slice_header_flag`.

Khi `picture_header_in_slice_header_flag` bằng 1 đối với lát được mã hóa, đây là yêu cầu của tương thích dòng bit rằng không có đơn vị VCL NAL với `nal_unit_type` bằng PH_NUT sẽ được hiện diện trong CLVS.

Khi `picture_header_in_slice_header_flag` bằng 0, tất cả các lát được mã hóa trong ảnh hiện tại sẽ có `picture_header_in_slice_header_flag` bằng 0, và PU hiện tại sẽ có đơn vị PH NAL.

Ngoài ra, còn điều kiện rằng buộc `pic_header_in_slice_header_constraint_flag` có thể được thêm vào trong `general_constraint_info()`. Khi `pic_header_in_slice_header_constraint_flag` bằng 1, giá trị của `picture_header_in_slice_header_flag` trong PH sẽ bằng 1.

<code>general_constraint_info() {</code>	Mô tả
...	
<code>one_tile_per_pic_constraint_flag</code>	u(1)
<code>one_slice_per_pic_constraint_flag</code>	u(1)
<code>one_subpic_per_pic_constraint_flag</code>	u(1)
<code>pic_header_in_slice_header_constraint_flag</code>	u(1)
...	
<code>}</code>	

`pic_header_in_slice_header_constraint_flag` bằng 1 chỉ rõ rằng cấu trúc cú pháp PH được hiện diện trong thông tin tiêu đề lát. `pic_header_in_slice_header_constraint_flag` bằng 0 không đặt ra điều kiện ràng

buộc này. Khi `pic_header_in_slice_header_constraint_flag` bằng 1, giá trị của `picture_header_in_slice_header_flag` sẽ bằng 1.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `rpl_info_in_ph_flag`, `dbf_info_in_ph_flag`, `sao_info_in_ph_flag`, `wp_info_in_ph_flag`, `qp_delta_info_in_ph_flag` sẽ bằng 0.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `rect_slice_flag` sẽ bằng 1.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `subpic_info_present_flag` sẽ bằng 0.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `separate_colour_plane_flag` sẽ bằng 0.

Phần sau đây là định nghĩa cú pháp ví dụ đối với cờ điều kiện ràng buộc mới:

<code>general_constraint_info()</code> {	Mô tả
<code>general_progressive_source_flag</code>	u(1)
<code>general_interlaced_source_flag</code>	u(1)
<code>general_non_packed_constraint_flag</code>	u(1)
<code>general_frame_only_constraint_flag</code>	u(1)
<code>general_non_projected_constraint_flag</code>	u(1)
<code>intra_only_constraint_flag</code>	u(1)
<code>max_bitdepth_constraint_idc</code>	u(4)
<code>max_chroma_format_constraint_idc</code>	u(2)
<code>no_res_change_in_clvs_constraint_flag</code>	u(1)
<code>one_tile_per_pic_constraint_flag</code>	u(1)
<code>one_slice_per_pic_constraint_flag</code>	u(1)
<code>one_subpic_per_pic_constraint_flag</code>	u(1)
<code>pic_header_in_slice_header_constraint_flag</code>	u(1)

no_qbtt_dual_tree_intra_constraint_flag	u(1)
no_partition_constraints_override_constraint_flag	u(1)
no_sao_constraint_flag	u(1)
no_alf_constraint_flag	u(1)
no_ccalf_constraint_flag	u(1)
no_joint_cbr_constraint_flag	u(1)
no_ref_wraparound_constraint_flag	u(1)
no_temporal_mvp_constraint_flag	u(1)
no_sbtmvp_constraint_flag	u(1)
no_amvr_constraint_flag	u(1)
no_bdof_constraint_flag	u(1)
no_dmvr_constraint_flag	u(1)
no_cclm_constraint_flag	u(1)
no_mts_constraint_flag	u(1)
no_sbt_constraint_flag	u(1)
no_affine_motion_constraint_flag	u(1)
no_bcw_constraint_flag	u(1)
no_ibc_constraint_flag	u(1)
no_ciip_constraint_flag	u(1)
no_fpel_mmvd_constraint_flag	u(1)
no_gpm_constraint_flag	u(1)
no_ladf_constraint_flag	u(1)
no_transform_skip_constraint_flag	u(1)
no_bdpem_constraint_flag	u(1)
no_qp_delta_constraint_flag	u(1)
no_dep_quant_constraint_flag	u(1)

no_sign_data_hiding_constraint_flag	u(1)
no_mixed_nalu_types_in_pic_constraint_flag	u(1)
no_trail_constraint_flag	u(1)
no_stsa_constraint_flag	u(1)
no_rasl_constraint_flag	u(1)
no_radl_constraint_flag	u(1)
no_idr_constraint_flag	u(1)
no_cra_constraint_flag	u(1)
no_gdr_constraint_flag	u(1)
no_aps_constraint_flag	u(1)
while(!byte_aligned())	
gci_alignment_zero_bit	f(1)
num_reserved_constraint_bytes	u(8)
for(i = 0; i < num_reserved_constraint_bytes; i++)	
gci_reserved_constraint_byte[i]	u(8)
}	

Phần sau đây là định nghĩa ngữ nghĩa ví dụ đối với cờ điều kiện ràng buộc mới:

separate_colour_plane_flag bằng 1 chỉ rõ rằng ba thành phần màu của khuôn dạng sắc độ 4:4:4 được mã hóa riêng biệt. **separate_colour_plane_flag** bằng 0 chỉ rõ rằng các thành phần màu không được mã hóa riêng biệt. Khi **separate_colour_plane_flag** không được hiện diện, tham số này được xem là bằng 0. Khi **separate_colour_plane_flag** bằng 1, ảnh được mã hóa bao gồm ba thành phần riêng biệt, mỗi thành phần bao gồm các mẫu được mã hóa của một mặt phẳng màu (Y, Cb, hoặc Cr) và sử dụng cú pháp mã hóa đơn sắc. Trong trường hợp này, mỗi mặt phẳng màu được kết hợp với giá trị **colour_plane_id** cụ thể.

Lưu ý - Không có sự phụ thuộc trong các xử lý giải mã giữa các mặt phẳng màu mà có các giá trị `colour_plane_id` khác nhau. Ví dụ, xử lý giải mã của ảnh đơn sắc với một giá trị của `colour_plane_id` không sử dụng bất kỳ dữ liệu từ các ảnh đơn sắc mà có các giá trị khác nhau của `colour_plane_id` đối với việc dự đoán liên đới.

Phụ thuộc vào giá trị của `separate_colour_plane_flag`, giá trị của biến `ChromaArrayType` được gán như sau:

- Nếu `separate_colour_plane_flag` bằng 0, `ChromaArrayType` được thiết lập bằng `chroma_format_idc`.
- Nếu không phải (`separate_colour_plane_flag` bằng 1), `ChromaArrayType` được thiết lập bằng 0.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `separate_colour_plane_flag` sẽ bằng 0.

`subpic_info_present_flag` bằng 1 chỉ rõ rằng thông tin ảnh con được hiện diện đối với CLVS và có thể có một hoặc nhiều hơn một ảnh con trong mỗi ảnh của CLVS. `subpic_info_present_flag` bằng 0 chỉ rõ rằng thông tin ảnh con không được hiện diện đối với CLVS và chỉ có một ảnh con trong mỗi ảnh của CLVS.

Khi `res_change_in_clvs_allowed_flag` bằng 1, giá trị của `subpic_info_present_flag` sẽ bằng 0.

NOTE – khi dòng bit là kết quả của quá trình trích xuất dòng bit con và chứa chỉ tập con của các ảnh con của dòng bit đầu vào cho quá trình trích xuất dòng bit con, có thể được yêu cầu để thiết lập giá trị của `subpic_info_present_flag` bằng 1 trong RBSP của các SPS.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `subpic_info_present_flag` sẽ bằng 0.

`rect_slice_flag` bằng 0 chỉ rõ rằng các ô trong mỗi lát nằm trong thứ tự quét mảnh và thông tin lát không được báo hiệu trong PPS. `rect_slice_flag` bằng 1 chỉ rõ rằng các ô trong mỗi lát bao phủ vùng chữ nhật của ảnh và thông tin lát được báo hiệu trong PPS. Khi không được hiện diện, `rect_slice_flag` được giả

định bằng 1. Khi `subpic_info_present_flag` bằng 1, giá trị của `rect_slice_flag` sẽ bằng 1.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `rect_slice_flag` sẽ bằng 1.

`qp_delta_info_in_ph_flag` bằng 1 chỉ rõ rằng thông tin delta QP được hiện diện trong cấu trúc cú pháp PH và không được hiện diện trong các thông tin tiêu đề lát mà viện dẫn tới PPS mà không chứa cấu trúc cú pháp PH. `qp_delta_info_in_ph_flag` bằng 0 chỉ rõ rằng thông tin delta QP không được hiện diện trong cấu trúc cú pháp PH và có thể được hiện diện trong các thông tin tiêu đề lát mà viện dẫn tới PPS mà không chứa cấu trúc cú pháp PH.

Khi `pic_header_in_slice_header_constraint_flag` bằng 1, `rpl_info_in_ph_flag`, `dbf_info_in_ph_flag`, `sao_info_in_ph_flag`, `wp_info_in_ph_flag`, `qp_delta_info_in_ph_flag` sẽ bằng 0.

`one_subpic_per_pic_constraint_flag` bằng 1 chỉ rõ rằng mỗi ảnh sẽ chứa chỉ một ảnh con. `one_subpic_per_pic_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi `one_slice_per_pic_constraint_flag` bằng 1, giá trị của `one_subpic_per_pic_constraint_flag` sẽ bằng 1.

`pic_header_in_slice_header_constraint_flag` bằng 1 chỉ rõ rằng cấu trúc cú pháp PH được hiện diện trong thông tin tiêu đề lát. `pic_header_in_slice_header_constraint_flag` bằng 0 không đặt ra điều kiện ràng buộc này. Khi `pic_header_in_slice_header_constraint_flag` bằng 1, giá trị của `picture_header_in_slice_header_flag` sẽ bằng 1.

2. Căn chỉnh trên giá trị `subpic_treated_as_pic_flag` qua các lớp

Đối với việc tách ảnh con từ dòng bit đa lớp, khi `subpic_treated_as_pic_flag[ i ]` bằng 1, đối với mỗi lớp đầu ra và các lớp tham chiếu của nó trong OLS, tất cả ảnh trong lớp đầu ra và các lớp tham chiếu của nó sẽ có giá trị của `subpic_treated_as_pic_flag[ j ]` bằng 1 đối với mỗi giá trị của `j` trong khoảng từ 0 đến `sps_num_subpics_minus1`. Do đó, theo các phương án của sáng chế, điều kiện ràng buộc sau đây được bổ sung.

`subpic_treated_as_pic_flag[ i ]` bằng 1 chỉ rõ rằng ảnh con thứ i của mỗi ảnh được mã hóa trong CLVS được coi là ảnh trong quá trình giải mã ngoại trừ các thao tác lọc nội vòng. `subpic_treated_as_pic_flag[ i ]` bằng 0 chỉ rõ rằng ảnh con thứ i của mỗi ảnh được mã hóa trong CLVS không được coi là ảnh trong quá trình giải mã ngoại trừ các thao tác lọc nội vòng. Khi không được hiện diện, giá trị của `subpic_treated_as_pic_flag[ i ]` được giả định bằng `sps_independent_subpics_flag`.

Khi `subpic_treated_as_pic_flag[ i ]` bằng 1, đây là yêu cầu của tính tương thích dòng bit rằng tất cả các điều kiện sau đây là đúng đối với mỗi lớp đầu ra và các lớp tham chiếu của nó trong OLS mà bao gồm lớp mà chứa ảnh con thứ i như là lớp đầu ra:

- Tất cả ảnh trong lớp đầu ra và các lớp tham chiếu của nó sẽ có cùng giá trị của `pic_width_in_luma_samples` và cùng giá trị của `pic_height_in_luma_samples`.
- Tất cả các SPS được viện dẫn tới bởi lớp đầu ra và các lớp tham chiếu của nó sẽ có cùng giá trị của `sps_num_subpics_minus1` và sẽ có cùng các giá trị của `subpic_ctu_top_left_x[ j ]`, `subpic_ctu_top_left_y[ j ]`, `subpic_width_minus1[ j ]`, `subpic_height_minus1[ j ]`, và `loop_filter_across_subpic_enabled_flag[ j ]`, một cách lần lượt, đối với mỗi giá trị của j trong đoạn từ 0 đến `sps_num_subpics_minus1`.
- Tất cả ảnh trong mỗi đơn vị truy nhập trong lớp đầu ra và các lớp tham chiếu của nó sẽ có cùng giá trị của `SubpicIdVal[ j ]` đối với mỗi giá trị của j trong đoạn từ 0 đến `sps_num_subpics_minus1`.
- Tất cả ảnh trong lớp đầu ra và các lớp tham chiếu của nó sẽ có cùng giá trị của `subpic_treated_as_pic_flag[ j ]` bằng 1 đối với mỗi giá trị của j trong đoạn từ 0 đến `sps_num_subpics_minus1`.

3. Điều kiện ràng buộc về số lượng ảnh con đối với ảnh nhỏ

Trong một phương án, điều kiện ràng buộc bổ sung có thể được đặt ra sao cho số lượng ảnh con sẽ bằng 1, khi kích cỡ ảnh không lớn hơn `CtbSizeY`.

Phần sau đây là các định nghĩa cú pháp và/hoặc ngữ nghĩa ví dụ đối với điều kiện ràng buộc bổ sung:

Khi `pic_width_max_in_luma_samples` không lớn hơn `CtbSizeY` và `pic_height_max_in_luma_samples` không lớn hơn `CtbSizeY`, giá trị của `sps_num_subpics_minus1` sẽ bằng 0.

4. Hiệu chỉnh trên khoảng giá trị của `sps_subpic_id_len_minus1`

Trong một phương án, các ngữ nghĩa của `sps_subpic_id_len_minus1` có thể được cải biến như sau sao cho phần tử cú pháp có thể có khoảng giá trị chính xác hơn.

`sps_subpic_id_len_minus1` plus 1 chỉ rõ số lượng bit được sử dụng để biểu diễn phần tử cú pháp `sps_subpic_id[ i ]`, các phần tử cú pháp `pps_subpic_id[ i ]`, khi được hiện diện, và phần tử cú pháp `slice_subpic_id`, khi được hiện diện. Giá trị của `sps_subpic_id_len_minus1` sẽ nằm trong đoạn từ 0 đến 15. Giá trị của $1 \ll (\text{sps_subpic_id_len_minus1})$ sẽ lớn hơn hoặc bằng `sps_num_subpics_minus1 + 1`.

Các phương án được đề cập nêu trên có thể được sử dụng một cách riêng biệt hoặc được kết hợp trong thứ tự bất kỳ. Ngoài ra, mỗi phương án, bộ mã hóa và/hoặc bộ giải mã có thể được thực hiện bởi hệ mạch xử lý (ví dụ, một hoặc nhiều bộ xử lý hoặc một hoặc nhiều mạch tích hợp). Trong một ví dụ, một hoặc nhiều bộ xử lý thực thi chương trình mà được lưu trữ trong phương tiện bất biến đọc được bởi máy tính.

Fig.19 thể hiện lưu đồ mô tả xử lý 1900 theo phương án của sáng chế. Xử lý 1900 có thể được sử dụng trong việc khôi phục khối để tạo ra khối dự đoán đối với khối được khôi phục. Thuật ngữ khối trong sáng chế có thể được hiểu là khối dự đoán, CB, CU, hoặc loại tương tự. Trong các phương án khác nhau, xử lý 1900 được thực hiện bởi hệ mạch xử lý, như hệ mạch xử lý trong các thiết bị đầu cuối 310, 320, 330 và 340, hệ mạch xử lý mà thực hiện các chức năng của bộ mã hóa video 403, hệ mạch xử lý mà thực hiện các chức năng của bộ giải mã video 410, hệ mạch xử lý mà thực hiện các chức năng của bộ giải mã video 510, hệ mạch xử lý mà thực hiện các chức năng của bộ mã hóa video 603,

và loại tương tự. Trong một vài phương án, xử lý 1900 được thực hiện trong các lệnh phần mềm, nhờ đó khi hệ mạch xử lý thực thi các lệnh phần mềm, hệ mạch xử lý thực hiện xử lý 1900. Xử lý bắt đầu tại bước S1901 và chuyển sang bước S1910.

Tại bước S1910, cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa có thể được thu nhận. Mỗi cờ cổng có thể chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không. Thông tin điều kiện ràng buộc có thể dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa.

Thông tin điều kiện ràng buộc có thể được hiện diện trong cú pháp mức cao (ví dụ, SPS, VPS, DCI, DPS) đối với ít nhất một ảnh, và tập con của các khối mã hóa có thể bao gồm các khối mã hóa trong ít nhất một ảnh.

Trong ví dụ của sáng chế, mỗi cờ cổng chỉ báo rằng nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng không được hiện diện trong thông tin điều kiện ràng buộc, và không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

Tại bước 1920, việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không có thể được xác định dựa trên cờ cổng của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc. Nhóm thông tin điều kiện ràng buộc của cờ cổng có thể bao gồm ít nhất một cờ điều kiện ràng buộc đối với tập con của các khối mã hóa.

Tại bước S1930, thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không.

Trong ví dụ của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ cổng được xác định là được hiện diện trong thông tin điều kiện ràng buộc. Thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa

trên ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công.

Trong ví dụ của sáng chế, nhóm thông tin điều kiện ràng buộc của cờ công được xác định là không được hiện diện trong thông tin điều kiện ràng buộc. Thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định độc lập với nhóm thông tin điều kiện ràng buộc của cờ công.

Tại bước S1940, tập con của các khối mã hóa có thể được khôi phục dựa trên thông tin dự đoán.

Xử lý 1900 có thể được áp dụng một cách thích hợp. Các bước trong xử lý 1900 có thể được cải biến và/hoặc được bỏ qua. Các bước bổ sung có thể được thêm vào. Thứ tự thích hợp bất kỳ của các phương án có thể được sử dụng. Ví dụ, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa. Các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc có thể được báo hiệu theo thứ tự định trước. Các cờ điều kiện ràng buộc có thể được giải mã chạy, và do đó thông tin dự đoán đối với tập con của các khối mã hóa có thể được xác định dựa trên các cờ điều kiện ràng buộc.

Trong ví dụ của sáng chế, ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công bao gồm các cờ điều kiện ràng buộc được báo hiệu theo thứ tự định trước, và các cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công có thể được giải mã chạy.

Trong ví dụ của sáng chế, độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công được hiện diện trong thông tin điều kiện ràng buộc.

Trong ví dụ của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa. Độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong mỗi một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong thông tin

điều kiện ràng buộc. Số lượng độ dịch có thể được báo hiệu trong dòng bit video được mã hóa.

Trong ví dụ của sáng chế, một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa. Một hoặc nhiều phần tử cú pháp trong thông tin điều kiện ràng buộc có thể được căn thẳng theo byte. Một hoặc nhiều phần tử cú pháp có thể bao gồm cờ công đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc và các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc.

Trong ví dụ của sáng chế, ít nhất một cờ công bao gồm cờ điều kiện ràng buộc chỉ khung (ví dụ, `general_frame_only_constraint_flag`) mà chỉ báo rằng các ảnh trong tập hợp lớp đầu ra có được mã hóa với mã hóa trường hay không. Tập hợp lớp đầu ra có thể bao gồm tập con của các khối mã hóa. Cờ chuỗi trường (ví dụ, `sps_field_seq_flag`) là sai dựa trên cờ điều kiện ràng buộc chỉ khung mà chỉ báo rằng tập con của các ảnh không được mã hóa với mã hóa trường trong đó tập con của các ảnh nằm trong một lớp của tập hợp lớp đầu ra.

Trong ví dụ của sáng chế, cờ chuỗi trường được báo hiệu trong dòng bit được mã hóa và cờ chuỗi trường chỉ báo rằng các ảnh trong lớp đầu ra có được mã hóa với mã hóa trường hay không.

Các phương án trong sáng chế có thể được sử dụng một cách riêng biệt hoặc được kết hợp trong thứ tự bất kỳ. Ngoài ra, mỗi phương pháp (hoặc các phương án), bộ mã hóa, và bộ giải mã có thể được thực hiện bởi hệ mạch xử lý (ví dụ, một hoặc nhiều bộ xử lý hoặc một hoặc nhiều mạch tích hợp). Trong một ví dụ, một hoặc nhiều bộ xử lý thực thi chương trình mà được lưu trữ trong phương tiện bất biến đọc được bởi máy tính.

Các khía cạnh của sáng chế liên quan đến việc báo hiệu thông tin mã hóa trường trong dòng video được mã hóa, ví dụ, với nhiều lớp.

Khi các ảnh được mã hóa thành dòng bit mà bao gồm nhiều lớp với các chất lượng khác nhau, dòng bit có thể có các phần tử cú pháp mà chỉ rõ các lớp nào có thể được xuất ra tại phía bộ giải mã. Trong mã hóa-giải mã video

định trước mà hỗ trợ nhiều lớp và các khả năng biến đổi tỷ lệ, một hoặc nhiều tập hợp lớp đầu ra có thể được báo hiệu trong tập tham số video (VPS-video parameter set). Các phân tử cú pháp mà chỉ rõ các tập hợp lớp đầu ra và phần phụ thuộc của chúng, mẫu/lớp/mức và các tham số mô hình tham chiếu bộ giải mã giả thiết cần được báo hiệu một cách hiệu quả trong tập hợp tham số. Khi các chuỗi video đa lớp được mã hóa như là cấu trúc trường trong đó hai ảnh trường hoặc nhiều hơn bao gồm một ảnh khung, thông tin mã hóa trường có thể được báo hiệu trong cấu trúc cú pháp mức cao, như SPS, VPS, hoặc bản tin SEI.

Trong một số kỹ thuật mã hóa video, Đơn vị truy nhập (AU) có thể liên quan đến (các) hình ảnh được mã hóa, (các) lát, (các) ô, (các) đơn vị NAL, và v.v., mà được chụp và được xử lý bên trong ảnh/lát/ô/dòng bit đơn vị NAL tương ứng ở thời điểm cụ thể. Thời điểm cụ thể này có thể là thời điểm xử lý.

Trong một vài ví dụ, như trong HEVC, và các kỹ thuật mã hóa video định trước khác, giá trị đếm thứ tự ảnh (POC) có thể được sử dụng để chỉ báo ảnh tham chiếu được lựa chọn trong số nhiều ảnh tham chiếu được lưu trữ trong bộ đệm ảnh được giải mã (DPB-Decoded picture buffer). Khi đơn vị truy nhập (AU) bao gồm một hoặc nhiều ảnh, lát, hoặc ô, mỗi ảnh, lát, hoặc ô thuộc về cùng một AU có thể có cùng giá trị POC, mà có thể được thu nhận từ đó rằng một hoặc nhiều ảnh, lát hoặc ô được tạo ra từ nội dung của cùng thời điểm xử lý. Do đó, hai ảnh/lát/ô mang cùng giá trị POC định trước có thể chỉ báo rằng hai ảnh/lát/ô thuộc về cùng AU và có cùng thời điểm xử lý. Ngược lại, hai ảnh/ô/lát mà có các giá trị POC khác nhau có thể chỉ báo hai ảnh/lát/ô thuộc về các AU khác nhau và có các thời điểm xử lý khác nhau.

Theo phương án của sáng chế, quan hệ cứng nêu trên có thể được làm mềm trong đó đơn vị truy nhập có thể bao gồm các ảnh, các lát hoặc các ô với các giá trị POC khác nhau. Bằng cách cho phép các giá trị POC khác nhau trong AU, các giá trị POC có thể được sử dụng để nhận dạng các ảnh/lát/ô có khả năng giải mã được với thời gian thể hiện đồng nhất. Do đó, các lớp có thể biến đổi tỷ lệ có thể được hỗ trợ mà không cần thay đổi báo hiệu lựa chọn hình

ảnh tham chiếu (ví dụ như báo hiệu tập hình ảnh tham chiếu hoặc báo hiệu danh sách hình ảnh tham chiếu), như được mô tả chi tiết hơn dưới đây.

Trong một vài ví dụ, có mong muốn nhận dạng AU mà hình ảnh/lát/ô thuộc về, đối với hình ảnh/các lát/các ô khác mà có các giá trị POC khác nhau, chỉ từ giá trị POC. Việc này có thể được thực hiện, như được mô tả dưới đây.

Trong phương án của sáng chế, số đếm đơn vị truy nhập (AUC) có thể được báo hiệu trong cấu trúc cú pháp bậc cao, như là tiêu đề đơn vị NAL, tiêu đề lát, tiêu đề nhóm ô, bản tin SEI, tập tham số hoặc dấu phân tách AU. Giá trị của AUC (hoặc giá trị AUC) có thể được sử dụng để nhận dạng các đơn vị NAL, các ảnh, các lát hoặc các ô nào mà thuộc về AU định trước. Giá trị của AUC có thể tương ứng với khoảng thời điểm xử lý riêng biệt. Giá trị AUC có thể bằng nhiều giá trị POC. Bằng cách chia giá trị POC cho giá trị nguyên, giá trị AUC có thể được tính toán. Trong một vài ví dụ, các thao tác phân chia có thể đặt một gánh nặng lên các cách thực hiện của bộ giải mã. Do đó, các giới hạn nhỏ trong khoảng đếm số của các giá trị AUC có thể cho phép thay thế thao tác phân chia thành các thao tác dịch. Ví dụ, giá trị AUC có thể bằng giá trị Bit quan trọng nhất (MSB-Most Significant Bit) của khoảng giá trị POC.

Trong phương án của sáng chế, giá trị của POC trên mỗi AU (ví dụ, poc_cycle-au) có thể được báo hiệu trong cấu trúc cú pháp bậc cao, như là tiêu đề đơn vị NAL, tiêu đề lát, tiêu đề nhóm ô, bản tin SEI, tập tham số hoặc dấu phân tách AU. Giá trị của chu kỳ POC theo AU (ví dụ, poc_cycle_au) có thể chỉ báo có bao nhiêu giá trị POC khác nhau và liên tiếp có thể được kết hợp với cùng AU. Ví dụ, nếu giá trị của chu kỳ POC theo AU (ví dụ, poc_cycle_au) bằng 4, các ảnh, các lát hoặc các ô có giá trị POC bằng 0 đến 3, được kết hợp với AU với giá trị AUC bằng 0, và các ảnh, các lát hoặc các ô với giá trị POC bằng 4 đến 7, được kết hợp với AU với giá trị AUC bằng 1. Do đó, giá trị của AUC có thể được giả định bằng cách chia giá trị POC cho giá trị của giá trị chu kỳ POC theo AU (ví dụ, poc_cycle_au).

Trong phương án của sáng chế, giá trị của chu kỳ POC theo AU (ví dụ, poc_cyle_au) có thể được thu nhận từ thông tin, ví dụ, nằm trong VPS, mà nhận dạng số lượng lớp SNR không gian trong chuỗi video được mã hóa. Trong khi việc thu nhận như được mô tả nêu trên có thể tiết kiệm số ít bit trong VPS và do đó có thể cải thiện hiệu quả mã hóa, có thể là ưu điểm để mã hóa rõ ràng giá trị của chu kỳ POC theo AU (ví dụ, poc_cycle_au) trong cấu trúc cú pháp mức cao thích hợp được phân cấp phía dưới VPS để tối thiểu hóa giá trị của chu kỳ POC theo AU (ví dụ, poc_cycle_au) đối với phần nhỏ định trước của dòng bit (ví dụ, ảnh). Việc tối ưu hóa này có thể tiết kiệm nhiều bit hơn mà có thể được tiết kiệm thông qua quá trình thu nhận ở trên vì các giá trị POC (và/hoặc các giá trị của các phần tử cú pháp viện dẫn không trực tiếp đến POC) có thể được mã hóa trong các cấu trúc cú pháp cấp độ thấp.

Trong phương án của sáng chế, FIG.20 thể hiện ví dụ của các bảng cú pháp của các bảng cú pháp để báo hiệu phần tử cú pháp của vps_poc_cycle_au trong VPS (hoặc SPS), mà chỉ báo poc_cycle_au được sử dụng cho toàn bộ ảnh/các lát trong chuỗi video được mã hóa, và phần tử cú pháp của slice_poc_cycle_au, mà chỉ báo poc_cycle_au của lát hiện thời, trong tiêu đề lát.

Nếu giá trị POC tăng lên một cách thống nhất trên mỗi AU, phần tử cú pháp vps_contant_poc_cycle_per_au trong VPS được thiết đặt bằng 1 và phần tử cú pháp vps_poc_cycle_au được báo hiệu trong VPS. Do đó, phần tử cú pháp slice_poc_cycle_au không được báo hiệu rõ ràng, và giá trị của AUC cho mỗi AU có thể được tính toán bằng cách chia giá trị của POC bởi phần tử cú pháp vps_poc_cycle_au.

Nếu giá trị POC không làm tăng một cách thống nhất trên mỗi AU, phần tử cú pháp vps_contant_poc_cycle_per_au trong VPS được thiết đặt bằng 0. Do đó, phần tử cú pháp vps_access_unit_cnt không được báo hiệu, trong khi đó phần tử cú pháp slice_access_unit_cnt được báo hiệu trong tiêu đề lát cho mỗi lát hoặc ảnh. Mỗi lát hoặc ảnh có thể có giá trị khác nhau của

slice_access_unit_cnt. Giá trị của AUC cho mỗi AU có thể được tính toán bằng cách chia giá trị của POC cho phần tử cú pháp slice_poc_cycle_au.

FIG.21 thể hiện sơ đồ khối minh họa dòng hoạt động liên quan được mô tả nêu trên. Dòng hoạt động này bắt đầu tại bước S2101 và chuyển sang bước S2110.

Tại bước S2110, VPS hoặc SPS có thể được phân tích và việc chu kỳ POC theo AU có cố định hay không có thể được nhận dạng.

Tại bước S2120, việc chu kỳ POC theo AU có cố định trong chuỗi video được mã hóa hay không có thể được xác định. Nếu chu kỳ POC theo AU được xác định là cố định trong chuỗi video được mã hóa, dòng xử lý chuyển sang bước S2130. Nếu không phải, dòng xử lý chuyển sang bước S2140.

Tại bước S2130, giá trị của số đếm đơn vị truy nhập có thể được tính toán từ giá trị poc_cycle_au mức chuỗi và giá trị POC. Sau đó, dòng xử lý then chuyển sang bước S2150.

Tại bước S2140, giá trị của số đếm đơn vị truy nhập có thể được tính toán từ giá trị poc_cycle_au mức ảnh và giá trị POC. Sau đó, dòng xử lý then chuyển sang bước S2150.

Tại bước S2150, VPS hoặc SPS có thể được phân tích và việc chu kỳ POC theo AU có cố định hay không có thể được nhận dạng.

Trong phương án của sáng chế, ngay cả khi giá trị của POC của ảnh, lát, hoặc ô có thể là khác nhau, hình ảnh, lát, hoặc ô tương ứng với AU có cùng một giá trị AUC có thể được kết hợp với cùng một thời điểm giải mã hoặc xuất ra. Do đó, không cần có sự phụ thuộc phân tích liên đới/giải mã qua các ảnh, các lát hoặc các ô trong cùng một AU, tất cả hoặc tập con của các ảnh, các lát hoặc các ô được kết hợp với cùng một AU có thể được giải mã song song, và có thể được xuất ra tại cùng thời điểm.

Trong phương án của sáng chế, ngay cả khi giá trị của POC của ảnh, lát, hoặc ô có thể là khác nhau, ảnh, lát, hoặc ô tương ứng với AU có cùng một giá trị AUC có thể được kết hợp với cùng một thời điểm xử lý/hiển thị. Khi thời gian xử lý được chứa theo khuôn dạng bộ chứa, ngay cả khi các ảnh tương

ứng với các AU khác nhau, nếu các ảnh có cùng một thời điểm xử lý, các ảnh có thể được hiển thị cùng một đối tượng thời gian.

Trong phương án của sáng chế, mỗi ảnh, lát, hoặc ô có thể có cùng một ký hiệu nhận dạng theo thời gian (ví dụ, `temporal_id`) trong cùng một AU. Tất cả hoặc tập con của các ảnh, các lát hoặc các ô tương ứng với đối tượng thời gian có thể được kết hợp với cùng một lớp con theo thời gian. Trong phương án của sáng chế, mỗi ảnh, lát, hoặc ô có thể có các ký hiệu nhận dạng lớp không gian giống hoặc khác nhau (ví dụ, `layer_id`) trong cùng AU. Tất cả hoặc tập con của các ảnh, các lát hoặc các ô tương ứng với đối tượng thời gian có thể được kết hợp với cùng một lớp hoặc lớp theo không gian khác nhau.

FIG.22 thể hiện ví dụ của cấu trúc chuỗi video với sự kết hợp của `temporal_id`, `layer_id`, các giá trị POC và AUC có sự thay đổi độ phân giải thích ứng. Trong ví dụ được thể hiện trên FIG.22, hình ảnh, lát hoặc ô trong AU thứ nhất với $AUC = 0$ có thể có `temporal_id = 0` và `layer_id = 0` hoặc 1, trong khi đó hình ảnh, lát hoặc ô trong AU thứ hai với $AUC = 1$ có thể có `temporal_id = 1` và `layer_id = 0` hoặc 1, theo cách tương ứng. Giá trị của POC tăng lên 1 trên mỗi ảnh bất kể các giá trị của `temporal_id` và `layer_id`. Việc dẫn tới FIG.22, giá trị của `poc_cycle_au` có thể bằng 2. Trong ví dụ của sáng chế, giá trị của `poc_cycle_au` có thể được ưu tiên thiết lập bằng số lượng các lớp (có khả năng biến đổi tỷ lệ theo không gian). Trong ví dụ này, giá trị của POC được tăng thêm 2, trong khi đó giá trị của AUC tăng thêm 1.

Trong các phương án nêu trên, tất cả hoặc tập con của cấu trúc dự đoán ảnh liên đới hoặc cấu trúc dự đoán lớp liên đới và sự chỉ báo hình ảnh tham chiếu có thể được hỗ trợ bằng cách sử dụng báo hiệu tập hình ảnh tham chiếu (RPS) trong các tiêu chuẩn video, như HEVC hoặc báo hiệu danh sách hình ảnh tham chiếu (RPL). Trong RPS hoặc RPL, hình ảnh tham chiếu được chọn có thể được chỉ báo bởi báo hiệu giá trị của POC hoặc giá trị delta của POC giữa hình ảnh hiện thời và hình ảnh tham chiếu được lựa chọn. Trong các ví dụ khác nhau của sáng chế, RPS và RPL có thể được sử dụng để chỉ báo cấu trúc dự đoán ảnh liên đới hoặc dự đoán lớp liên đới mà

không cần thay đổi báo hiệu, và có các giới hạn sau đây. Nếu giá trị của `temporal_id` của hình ảnh tham chiếu lớn hơn giá trị của `temporal_id` của hình ảnh hiện tại, hình ảnh hiện tại có thể không sử dụng hình ảnh tham chiếu cho việc bù chuyển động hoặc các dự đoán khác. Nếu giá trị của `layer_id` của hình ảnh tham chiếu lớn hơn giá trị `layer_id` của hình ảnh hiện tại, hình ảnh hiện tại có thể không sử dụng hình ảnh tham chiếu cho việc bù chuyển động hoặc các dự đoán khác.

Trong phương án của sáng chế, biến đổi tỷ lệ vector chuyển động dựa trên độ chênh lệch POC đối với dự đoán vector chuyển động theo thời gian có thể được vô hiệu qua các ảnh trong đơn vị truy nhập. Do đó, mặc dù mỗi ảnh có thể có giá trị POC khác nhau trong AU, vector chuyển động không được biến đổi tỷ lệ và được sử dụng cho dự đoán vector chuyển động theo thời gian trong AU do ảnh tham chiếu với POC khác nhau trong cùng AU được xem xét ảnh tham chiếu mà có cùng thời điểm. Do đó, chức năng biến đổi tỷ lệ vector chuyển động có thể trở về 1, khi ảnh tham chiếu thuộc về AU được kết hợp với ảnh hiện tại.

Trong phương án của sáng chế, việc biến đổi tỷ lệ vector chuyển động dựa trên độ chênh lệch POC cho dự đoán vector chuyển động theo thời gian có thể một cách tùy chọn được vô hiệu qua nhiều ảnh, khi độ phân giải theo không gian của hình ảnh tham chiếu khác với độ phân giải theo không gian của ảnh hiện tại. Khi việc biến đổi tỷ lệ vector chuyển động được cho phép, vector chuyển động được biến đổi tỷ lệ dựa trên cả độ chênh lệch POC và tỷ lệ độ phân giải theo không gian giữa ảnh hiện tại và ảnh tham chiếu.

Trong phương án của sáng chế, vector chuyển động có thể được lấy tỷ lệ dựa trên độ chênh lệch AUC thay vì độ chênh lệch POC, cho dự đoán vector chuyển động theo thời gian, đặc biệt khi `poc_cycle_au` có giá trị không đồng nhất (ví dụ, khi `vps_contant_poc_cycle_per_au == 0`). Ngược lại (ví dụ, khi `vps_contant_poc_cycle_per_au == 1`), việc lấy tỷ lệ vector chuyển động dựa trên độ chênh lệch AUC có thể giống với việc lấy tỷ lệ vector chuyển động dựa trên độ chênh lệch POC.

Trong phương án của sáng chế, khi vector chuyển động được biến đổi tỷ lệ dựa trên độ chênh lệch AUC, vector chuyển động tham chiếu trong cùng một AU (với cùng một giá trị AUC) với ảnh hiện tại không được biến đổi tỷ lệ dựa trên độ chênh lệch AUC và được sử dụng cho dự đoán vector chuyển động mà không cần biến đổi tỷ lệ hoặc có biến đổi tỷ lệ dựa trên tỷ lệ độ phân giải theo không gian giữa ảnh hiện tại và ảnh tham chiếu.

Trong phương án của sáng chế, giá trị AUC được sử dụng để nhận dạng biên của AU và được sử dụng cho thao tác của bộ giải mã tham chiếu giả định (HRD), mà cần sự định thời đầu vào và đầu ra với độ chi tiết AU. Trong các ví dụ khác nhau, ảnh được giải mã với lớp cao nhất trong AU có thể được xuất ra để hiển thị. Giá trị AUC và giá trị layer_id có thể được sử dụng để nhận dạng ảnh đầu ra.

Trong phương án của sáng chế, ảnh có thể bao gồm một hoặc nhiều ảnh con. Mỗi ảnh con này có thể phủ vùng cục bộ hoặc toàn bộ vùng của ảnh. Vùng được hỗ trợ bởi ảnh con có thể có hoặc có thể không bị chồng lấn với vùng được hỗ trợ bởi ảnh con khác. Vùng được xử lý bởi một hoặc nhiều ảnh con có thể có hoặc có thể không phủ toàn bộ vùng của ảnh. Nếu ảnh bao gồm ảnh con, vùng được hỗ trợ bởi ảnh con có thể giống với vùng được hỗ trợ bởi ảnh.

Trong phương án của sáng chế, ảnh con có thể được mã hóa bằng phương pháp mã hóa giống với phương pháp mã hóa được sử dụng cho ảnh được mã hóa. Ảnh con có thể được mã hóa độc lập hoặc có thể được mã hóa phụ thuộc vào một con khác hoặc ảnh được mã hóa. Ảnh con có thể có hoặc có thể không có phụ thuộc phân tích bất kỳ từ một ảnh con khác hoặc ảnh được mã hóa.

Trong phương án của sáng chế, ảnh con được mã hóa có thể được chứa trong một hoặc nhiều lớp. Ảnh con được mã hóa trong lớp có thể có độ phân giải theo không gian khác nhau. Ảnh con gốc có thể được tái lấy mẫu theo không gian (ví dụ, được lấy mẫu lên hoặc lấy mẫu xuống), được mã hóa

với các tham số độ phân giải không gian khác nhau, và được chứa trong dòng bit tương ứng với lớp.

Trong phương án của sáng chế, ảnh con với (W, H) có thể được mã hóa và được chứa trong dòng bit được mã hóa tương ứng với lớp 0. W chỉ báo độ rộng của ảnh con và H chỉ báo độ cao của ảnh con, một cách lần lượt. Ảnh con được lấy mẫu lên (hoặc được lấy mẫu xuống) có thể được lấy mẫu lên (hoặc được lấy mẫu xuống) từ ảnh con với độ phân giải không gian gốc. Ảnh con được lấy mẫu lên (hoặc được lấy mẫu xuống) có thể có độ phân giải không gian $(W^*S_{w,k}, H^*S_{h,k})$ và có thể được mã hóa và được chứa trong dòng bit được mã hóa tương ứng với lớp k . Các tham số $S_{w,k}$ và $S_{h,k}$ có thể chỉ báo các tỷ lệ tái lấy mẫu dọc hoặc ngang, một cách lần lượt. Nếu các giá trị của $S_{w,k}, S_{h,k}$ lớn hơn 1, việc tái lấy mẫu là lấy mẫu lên. Nếu các giá trị của $S_{w,k}, S_{h,k}$ nhỏ hơn 1, việc tái lấy mẫu là lấy mẫu xuống.

Trong phương án của sáng chế, ảnh con được mã hóa trong lớp có thể có chất lượng quan sát khác nhau từ chất lượng quan sát của ảnh con được mã hóa trong lớp khác trong cùng ảnh con hoặc các ảnh con khác nhau. Ví dụ, ảnh con i trong lớp n , được mã hóa với tham số lượng tử hóa (ví dụ, Q_i, n) trong khi đó ảnh con j trong lớp m được mã hóa với tham số lượng tử hóa (ví dụ Q_j, m).

Trong phương án của sáng chế, ảnh con được mã hóa trong lớp có thể giải mã được một cách độc lập, mà không cần bất kỳ phân tích hoặc giải mã phần phụ thuộc nào từ ảnh con được mã hóa trong một lớp khác của cùng vùng cục bộ. Lớp ảnh con, mà có thể giải mã được một cách độc lập mà không cần tham chiếu đến một lớp ảnh con khác của cùng vùng cục bộ, là lớp ảnh con độc lập. Ảnh con được mã hóa trong lớp ảnh con độc lập có thể hoặc có thể không có sự phụ thuộc phân tích hoặc giải mã từ ảnh con được mã hóa trước đó trong cùng lớp ảnh con. Ảnh con được mã hóa có thể không có bất kỳ sự phụ thuộc từ ảnh được mã hóa trong lớp ảnh con khác.

Trong phương án của sáng chế, ảnh con được mã hóa trong lớp có thể giải mã được một cách phụ thuộc, với bất kỳ sự phụ thuộc phân tích hoặc giải mã từ ảnh con được mã hóa trong lớp khác của cùng vùng cục bộ. Lớp ảnh con, mà có thể giải mã được một cách phụ thuộc với tham chiếu đến một lớp ảnh con khác của cùng vùng cục bộ, là lớp ảnh con phụ thuộc. Ảnh con được mã hóa trong ảnh con phụ thuộc có thể tham chiếu đến ảnh con được mã hóa thuộc về cùng một ảnh con, ảnh con được mã hóa trước trong cùng một lớp ảnh con, hoặc cả hai ảnh con tham chiếu.

Trong phương án của sáng chế, ảnh con được mã hóa bao gồm một hoặc nhiều lớp ảnh con độc lập và một hoặc nhiều lớp ảnh con phụ thuộc. Tuy nhiên, ít nhất một lớp ảnh con độc lập có thể được biểu diễn cho ảnh con được mã hóa. Lớp ảnh con độc lập có thể có giá trị của ký hiệu nhận dạng lớp (*layer_id*), mà có thể được thể hiện trong tiêu đề đơn vị NAL hoặc một cấu trúc cú pháp bậc cao khác, bằng 0. Lớp ảnh con với *layer_id* bằng 0 là lớp ảnh con gốc.

Trong phương án của sáng chế, ảnh có thể bao gồm một hoặc nhiều ảnh con phía trước và một ảnh con phía nền. Vùng được hỗ trợ bởi ảnh con phía nền có thể bằng vùng của ảnh. Vùng được hỗ trợ bởi ảnh con phía trước có thể chồng lấn với vùng được hỗ trợ bởi ảnh con phía nền. Ảnh con phía nền có thể là lớp ảnh con cơ sở, trong khi đó ảnh con phía trước có thể là lớp ảnh con không phải cơ sở (ví dụ, lớp ảnh con nâng cao). Một hoặc nhiều lớp ảnh con không phải cơ sở có thể tham chiếu cùng lớp cơ sở để giải mã. Mỗi lớp ảnh con không phải cơ sở với *layer_id* bằng *a* có thể tham chiếu lớp ảnh con không phải cơ sở với *layer_id* bằng *b*, trong đó *a* lớn hơn *b*.

Trong phương án của sáng chế, ảnh có thể bao gồm một hoặc nhiều ảnh con phía trước mà có hoặc không có ảnh con phía nền. Mỗi ảnh con có thể có lớp ảnh con cơ sở của nó và một hoặc nhiều lớp không phải cơ sở (nâng cao). Mỗi lớp ảnh con cơ sở có thể được tham chiếu bởi một hoặc nhiều lớp ảnh con không phải cơ sở. Mỗi lớp ảnh con không phải cơ

sở với `layer_id` bằng a có thể tham chiếu lớp ảnh con không phải cơ sở với `layer_id` bằng b , trong đó a lớn hơn b .

Trong phương án của sáng chế, ảnh có thể bao gồm một hoặc nhiều ảnh con phía trước mà có hoặc không có ảnh con phía nền. Mỗi ảnh con được mã hóa trong lớp ảnh con (cơ sở hoặc không phải cơ sở) có thể được tham chiếu bởi một hoặc nhiều lớp ảnh con không phải cơ sở thuộc về cùng một ảnh con và một hoặc nhiều lớp ảnh con không phải cơ sở, mà không thuộc về cùng một ảnh con.

Trong phương án của sáng chế, ảnh có thể bao gồm một hoặc nhiều ảnh con phía trước mà có hoặc không có ảnh con phía nền. Ảnh con trong lớp a có thể còn được phân chia thành nhiều ảnh con trong cùng một lớp. Một hoặc nhiều ảnh con được mã hóa trong lớp b có thể tham chiếu ảnh con được phân chia trong lớp a .

Trong phương án của sáng chế, chuỗi video được mã hóa (CVS) có thể là nhóm của các ảnh được mã hóa. CVS có thể gồm có một hoặc nhiều chuỗi ảnh con được mã hóa (CSPS), trong đó CSPS có thể là nhóm ảnh con được mã hóa mà phủ cùng vùng cục bộ của ảnh. CSPS có thể có độ phân giải theo thời gian giống hoặc khác nhau so với của chuỗi video được mã hóa.

Trong phương án của sáng chế, CSPS có thể được mã hóa và được chứa trong một hoặc nhiều lớp. CSPS có thể gồm có một hoặc nhiều lớp CSPS. Giải mã một hoặc nhiều lớp CSPS tương ứng với CSPS có thể khôi phục chuỗi của ảnh con tương ứng với cùng vùng cục bộ.

Trong phương án của sáng chế, số lượng lớp CSPS tương ứng với CSPS có thể giống với hoặc khác với số lượng lớp CSPS tương ứng với một CSPS khác.

Trong phương án của sáng chế, lớp CSPS có thể có độ phân giải theo thời gian khác nhau (ví dụ như tốc độ khung) từ một lớp CSPS khác. Chuỗi ảnh con (ví dụ, không được nén) gốc có thể được tái lấy mẫu theo

thời gian (ví dụ, được lấy mẫu lên hoặc lấy mẫu xuống), được mã hóa với các tham số độ phân giải theo thời gian khác nhau, và được chứa trong dòng bit tương ứng với lớp.

Trong phương án của sáng chế, chuỗi ảnh con với tốc độ khung, F , có thể được mã hóa và được chứa trong dòng bit được mã hóa tương ứng với lớp 0, trong khi đó chuỗi ảnh con được lấy mẫu lên (hoặc lấy mẫu xuống) theo thời gian từ chuỗi ảnh con gốc, với $F^* St,k$, có thể được mã hóa và được chứa trong dòng bit được mã hóa tương ứng với lớp k . St,k chỉ báo tỷ lệ lấy mẫu theo thời gian đối với lớp k . Nếu giá trị của St,k lớn hơn 1, quá trình tái lấy mẫu theo thời gian tương ứng với sự chuyển đổi lên tốc độ khung. Nếu giá trị của St,k nhỏ hơn 1, quá trình tái lấy mẫu theo thời gian tương ứng với sự chuyển đổi xuống tốc độ khung.

Trong phương án của sáng chế, khi ảnh con với lớp CSPS a được tham chiếu bởi ảnh con với lớp CSPS b cho việc bù chuyển động hoặc dự đoán lớp liên đới bất kỳ, nếu độ phân giải theo không gian của lớp CSPS a khác với độ phân giải theo không gian của lớp CSPS b , các điểm ảnh được giải mã trong lớp CSPS a được tái lấy mẫu và được sử dụng cho việc tham chiếu. Quá trình tái lấy mẫu có thể cần lọc lấy mẫu lên hoặc lọc lấy mẫu xuống.

FIG. 23 thể hiện ví dụ về dòng video bao gồm CSPS video phía nền với `layer_id` bằng 0 và nhiều lớp CSPS phía trước. Trong khi ảnh con được mã hóa có thể gồm có một hoặc nhiều lớp CSPS, vùng phía nền, mà không thuộc về bất kỳ lớp CSPS phía trước nào, có thể gồm có lớp cơ sở. Lớp cơ sở có thể chứa vùng phía nền và các vùng phía trước. Lớp CSPS nâng cao có thể chứa vùng phía trước. Lớp CSPS nâng cao có thể có chất lượng trực quan tốt hơn lớp cơ sở, tại cùng một vùng. Lớp CSPS nâng cao có thể tham chiếu các điểm ảnh được khôi phục và các vector chuyển động của lớp cơ sở, tương ứng với cùng một vùng.

Trong phương án của sáng chế, dòng bit video tương ứng với lớp cơ sở được chứa trong phần theo dõi, trong khi đó các lớp CSPS tương ứng với mỗi ảnh con được chứa trong phần theo dõi riêng, trong tệp video.

Trong phương án của sáng chế, dòng bit video tương ứng với lớp cơ sở được chứa trong phần theo dõi, trong khi đó các lớp CSPS với cùng `layer_id` được chứa trong phần theo dõi riêng. Trong ví dụ của sáng chế, phần theo dõi tương ứng với lớp k chỉ bao gồm các lớp CSPS tương ứng với lớp k .

Trong phương án của sáng chế, mỗi lớp CSPS của mỗi ảnh con được lưu trữ trong phần theo dõi riêng. Mỗi phần theo dõi có thể hoặc có thể không có bất kỳ phân tích hoặc giải mã phân phụ thuộc từ một hoặc nhiều phần theo dõi khác.

Trong phương án của sáng chế, mỗi phần theo dõi có thể chứa các dòng bit tương ứng với lớp i đến lớp j của các lớp CSPS của tất cả hoặc tập con của các ảnh con, trong đó $0 < i \leq j \leq k$, và k là lớp cao nhất của CSPS.

Trong phương án của sáng chế, ảnh bao gồm một hoặc nhiều dữ liệu đa phương tiện được kết hợp bao gồm bản đồ độ sâu, bản đồ alpha, 3D dữ liệu hình học, bản đồ chiếm giữ, và loại tương tự. Dữ liệu đa phương tiện được kết hợp và được định thời có thể được phân chia thành một hoặc nhiều dòng con dữ liệu mà mỗi chúng tương ứng với một ảnh con.

Trong phương án của sáng chế, FIG.24 thể hiện ví dụ về hội nghị video dựa trên phương pháp ảnh con đa lớp. Trong dòng video, một dòng bit video lớp cơ sở tương ứng với ảnh phía nền và một hoặc nhiều dòng bit video lớp nâng cao tương ứng với ảnh con phía trước được chứa. Mỗi dòng bit video lớp nâng cao tương ứng với lớp CSPS. Trong phần hiển thị, ảnh tương ứng với lớp cơ sở được hiển thị mặc định. Nó chứa một hoặc nhiều ảnh trong ảnh (PIP-picture in picture) của người dùng. Khi người dùng cụ thể được lựa chọn bằng sự điều khiển của máy khách,

lớp CSPA nâng cao tương ứng với người dùng được lựa chọn được giải mã và được hiển thị với chất lượng hoặc độ phân giải không gian nâng cao.

FIG.25 thể hiện sơ đồ đối với hoạt động được mô tả nêu trên.

Hoạt động này bắt đầu tại bước S1501 và chuyển sang bước S1510.

Tại bước S2510, dòng bit video với đa lớp có thể được giải mã.

Tại bước S1220, vùng phía nền và một hoặc nhiều ảnh con phía trước có thể được nhận dạng.

Tại bước S2530, việc vùng ảnh con cụ thể có được lựa chọn hay không có thể được xác định. Nếu vùng ảnh con cụ thể được xác định là được lựa chọn, hoạt động chuyển sang bước S2540. Nếu không phải, hoạt động chuyển sang bước S2550.

Tại S2540, ảnh con nâng cao có thể được giải mã và được hiển thị. Sau đó hoạt động chuyển sang bước S2599, và kết thúc.

Tại bước S2550, vùng phía nền có thể được giải mã và được hiển thị. Sau đó hoạt động chuyển sang bước S2599, và kết thúc.

Trong phương án của sáng chế, hộp trung tâm mạng (như bộ định tuyến) có thể lựa chọn tập con của các lớp để gửi đến người dùng phụ thuộc vào băng thông. Sự sắp xếp ảnh/ảnh con có thể được sử dụng cho việc thích ứng băng thông. Ví dụ, nếu người dùng không có băng thông, các dải bộ định tuyến của các lớp hoặc lựa chọn một số ảnh con do tính quan trọng của chúng hoặc dựa trên cài đặt được sử dụng và thao tác này có thể được thực hiện động để thích ứng với băng thông.

Fig.26 thể hiện trường hợp sử dụng 1400 của video 360. Khi ảnh 360 hình cầu được chiếu lên ảnh phẳng, ảnh 360 trình chiếu có thể được phân chia thành nhiều ảnh con như lớp cơ sở. Lớp nâng cao của ảnh con cụ thể có thể được mã hóa và được truyền đến máy khách. Bộ giải mã có thể giải mã cả lớp cơ sở bao gồm tất cả ảnh con và lớp nâng cao của ảnh con được chọn. Khi khung nhìn hiện tại giống với ảnh con được chọn, ảnh được hiển thị có thể có

chất lượng cao hơn với ảnh con được giải mã với lớp nâng cao. Ngược lại, ảnh được giải mã với lớp cơ sở có thể được hiển thị với chất lượng thấp.

Trong phương án của sáng chế, thông tin bố trí thích hợp bất kỳ để hiển thị có thể được biểu diễn trong tệp tin, như là thông tin bổ sung (như là bản tin SEI hoặc siêu dữ liệu). Một hoặc nhiều ảnh con được giải mã có thể được bố trí lại và được hiển thị phụ thuộc vào thông tin bố cục được báo hiệu. Thông tin bố trí có thể được báo hiệu bởi máy chủ phát trực tuyến hoặc máy truyền thanh, hoặc có thể được tạo lại bởi thực thể mạng hoặc máy chủ đám mây, hoặc có thể được xác định bởi thiết lập tùy chỉnh của người dùng.

Trong phương án của sáng chế, khi ảnh đầu ra được chia thành một hoặc nhiều vùng con (ví dụ, các vùng con chữ nhật), mỗi vùng con có thể được mã hóa như là lớp độc lập. Mỗi lớp độc lập tương ứng với vùng cục bộ có thể có giá trị `layer_id` duy nhất. Đối với mỗi lớp độc lập, kích cỡ ảnh con và thông tin vị trí có thể được báo hiệu. Ví dụ, kích cỡ ảnh (ví dụ độ rộng và/hoặc độ cao của ảnh), thông tin dịch vị của góc trên cùng-bên trái (ví dụ, `x_offset`, `y_offset`). Fig.27 thể hiện ví dụ của bố trí của ảnh con được chia, kích cỡ ảnh con và thông tin vị trí và cấu trúc dự đoán ảnh tương ứng. Thông tin bố trí bao gồm (các) kích cỡ ảnh con và (các) vị trí ảnh con có thể được báo hiệu trong cấu trúc cú pháp mức cao, như là (các) tập tham số, tiêu đề của lát hoặc nhóm ô, hoặc bản tin SEI.

Trong phương án của sáng chế, mỗi ảnh con tương ứng với lớp độc lập có thể có giá trị POC duy nhất trong AU. Khi ảnh tham chiếu trong số các ảnh được lưu trữ trong DPB được chỉ báo bằng cách sử dụng (các) phân tử cú pháp trong cấu trúc RPS hoặc RPL, (các) giá trị POC của mỗi ảnh con tương ứng với lớp có thể được sử dụng.

Trong phương án của sáng chế, để chỉ báo cấu trúc dự đoán (ví dụ, cấu trúc dự đoán liên lớp), `layer_id` có thể không được sử dụng và giá trị POC (ví dụ, giá trị delta POC) có thể được sử dụng.

Trong phương án của sáng chế, ảnh con với giá trị POC bằng N tương ứng với lớp (hoặc vùng cục bộ) có thể hoặc có thể không được sử dụng làm ảnh tham chiếu của ảnh con với giá trị POC bằng N+K, tương ứng với cùng một lớp (hoặc cùng vùng cục bộ) cho dự đoán bù chuyển động. Trong các trường hợp khác nhau, giá trị của số K có thể bằng số lượng tối đa của các lớp (ví dụ, các lớp độc lập), mà có thể đồng nhất với số lượng các vùng con.

Trong phương án của sáng chế, FIG.28 thể hiện trường hợp mở rộng của FIG.27. Khi ảnh đầu vào được chia thành nhiều (ví dụ như bốn) vùng con, mỗi vùng cục bộ có thể được mã hóa với một hoặc nhiều lớp. Việc dẫn tới FIG.28, số lượng các lớp độc lập có thể bằng số lượng các vùng con, và một hoặc nhiều lớp có thể tương ứng với vùng con. Do đó, mỗi vùng con có thể được mã hóa với một hoặc nhiều lớp độc lập và 0 lớp phụ thuộc hoặc nhiều hơn.

Trong phương án của sáng chế, trên FIG.28, ảnh đầu vào có thể được chia thành bốn vùng con. Vùng con trên cùng bên phải có thể được mã hóa thành hai lớp (ví dụ lớp 1 và lớp 4) trong khi đó vùng con dưới cùng bên phải có thể được mã hóa thành hai lớp (ví dụ, lớp 3 và lớp 5). Trong FIG.28, lớp 4 có thể tham chiếu lớp 1 cho dự đoán bù chuyển động, và lớp 5 có thể tham chiếu lớp 3 cho việc bù chuyển động.

Trong phương án của sáng chế, việc lọc trong vòng (như là lọc giải khối, lọc trong vòng thích ứng, tạo lại dạng, lọc song hướng hoặc lọc dựa trên học sâu bất kỳ) qua biên lớp có thể (một cách tùy chọn) được vô hiệu.

Trong phương án của sáng chế, dự đoán bù chuyển động hoặc sao chép nội khối qua biên lớp có thể (một cách tùy chọn) được vô hiệu.

Trong phương án của sáng chế, việc đệm biên cho dự đoán bù chuyển động hoặc lọc trong vòng ở biên của ảnh con có thể được xử lý một cách tùy chọn. Cờ chỉ báo để xem việc đệm biên được xử lý hay không có thể được báo hiệu trong cấu trúc cú pháp mức cao, như là (các) tập tham số

(VPS, SPS, PPS, hoặc APS), thông tin tiêu đề lát hoặc nhóm ô, hoặc bản tin SEI.

Trong phương án của sáng chế, thông tin bố cục của (các) ảnh con (hoặc (các) ảnh con) có thể được báo hiệu trong VPS hoặc SPS. Các Fig.29A-29B thể hiện ví dụ về các phần tử cú pháp trong VPS và SPS. Việc dẫn tới FIG.29A, cờ (ví dụ, `vps_sub_picture_dividing_flag`) được báo hiệu trong VPS. Cờ (ví dụ, `vps_sub_picture_dividing_flag`) có thể chỉ báo xem (các) ảnh đầu vào được chia thành nhiều vùng con hay không. Khi giá trị của cờ (ví dụ, `vps_sub_picture_dividing_flag`) bằng 0, (các) hình ảnh đầu vào trong (các) chuỗi video được mã hóa tương ứng với VPS hiện tại có thể không được chia thành nhiều vùng con. Do đó, kích cỡ ảnh đầu vào có thể bằng kích cỡ ảnh được mã hóa (ví dụ, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`), mà được báo hiệu trong SPS. Khi giá trị của cờ (ví dụ, `vps_sub_picture_dividing_flag`) bằng 1, (các) ảnh đầu vào có thể được chia thành nhiều vùng con. Do đó, các phần tử cú pháp `vps_full_pic_width_in_luma_samples` và `vps_full_pic_height_in_luma_samples` được báo hiệu trong VPS. Các giá trị của `vps_full_pic_width_in_luma_samples` và `vps_full_pic_height_in_luma_samples` có thể bằng độ rộng và độ cao của (các) ảnh đầu vào, một cách tương ứng.

Trong phương án của sáng chế, các giá trị của `vps_full_pic_width_in_luma_samples` và `vps_full_pic_height_in_luma_samples` có thể không được sử dụng để giải mã, nhưng có thể được sử dụng cho xử lý và hiển thị.

Trong phương án của sáng chế, khi giá trị của cờ (ví dụ `vps_sub_picture_dividing_flag`) bằng 1, các phần tử cú pháp `pic_offset_x` và `pic_offset_y` có thể được báo hiệu trong SPS, mà tương ứng với (các) lớp cụ thể. Do đó, ảnh được mã hóa kích cỡ (ví dụ, `pic_width_in_luma_samples`, `pic_height_in_luma_samples`) được báo hiệu trong SPS có thể bằng độ rộng và độ cao của vùng con tương ứng với lớp cụ thể. Vị trí (ví dụ, các phần tử cú

pháp `pic_offset_x` và `pic_offset_y`) của góc trên cùng bên trái của vùng con có thể được báo hiệu trong SPS.

Trong phương án của sáng chế, thông tin vị trí (ví dụ, các phần tử cú pháp `pic_offset_x`, `pic_offset_y`) của góc trên cùng bên trái của vùng con có thể không được sử dụng để giải mã, nhưng có thể được sử dụng cho việc cấu thành và hiển thị.

Trong phương án của sáng chế, thông tin bố cục (ví dụ, kích cỡ và vị trí) của tất cả hoặc (các) vùng con tập con của (các) ảnh đầu vào và thông tin phụ thuộc giữa (các) lớp có thể được báo hiệu trong tập tham số hoặc bản tin SEI. Fig.30 thể hiện ví dụ 1800 của các phần tử cú pháp để chỉ báo thông tin của bố cục của các vùng con, sự phụ thuộc giữa các lớp, và quan hệ giữa vùng con và một hoặc nhiều các lớp. Viền dẫn tới FIG.30, phần tử cú pháp `num_sub_region` có thể chỉ báo số lượng vùng con (chữ nhật) trong chuỗi video được mã hóa hiện tại. Phần tử cú pháp `num_layers` có thể chỉ báo số lượng lớp trong chuỗi video được mã hóa hiện tại. Giá trị của `num_layers` có thể bằng hoặc lớn hơn giá trị của `num_sub_region`. Khi vùng con bất kỳ được mã hóa dưới dạng lớp đơn, giá trị của `num_layers` có thể bằng giá trị của `num_sub_region`. Khi một hoặc nhiều các vùng con được mã hóa dưới dạng đa lớp, giá trị của `num_layers` có thể lớn hơn giá trị của `num_sub_region`. Phần tử cú pháp `direct_dependency_flag[i][j]` có thể chỉ báo sự phụ thuộc từ lớp thứ *j* tới lớp thứ *i*. Phần tử cú pháp `num_layers_for_region[i]` có thể chỉ báo số lượng lớp được kết hợp với vùng con thứ *i*. Phần tử cú pháp `sub_region_layer_id[i][j]` có thể chỉ báo `layer_id` của lớp thứ *j* được kết hợp với vùng con thứ *i*. `sub_region_offset_x[i]` và `sub_region_offset_y[i]` có thể chỉ báo vị trí ngang và dọc của góc trên cùng bên trái của vùng con thứ *i*, một cách lần lượt. `sub_region_width[i]` và `sub_region_height[i]` có thể chỉ báo độ rộng và độ cao của vùng con thứ *i*, một cách lần lượt.

Trong phương án của sáng chế, một hoặc nhiều phần tử cú pháp mà chỉ rõ tập hợp lớp đầu ra để chỉ báo một trong nhiều lớp cần được xuất ra với

hoặc mà không cần có thông tin PTL có thể được báo hiệu trong cấu trúc cú pháp mức cao, ví dụ như VPS, DPS, SPS, PPS, APS hoặc bản tin SEI.

Viện dẫn tới Fig.31, phần tử cú pháp `num_output_layer_sets` mà chỉ báo số lượng tập hợp lớp đầu ra (OLS) trong chuỗi video được mã hóa liên quan đến VPS có thể được báo hiệu trong VPS. Đối với mỗi tập hợp lớp đầu ra, cờ (ví dụ, `output_layer_flag`) có thể được báo hiệu đối với mỗi lớp đầu ra.

Trong phương án của sáng chế, `output_layer_flag[i]` bằng 1 có thể chỉ rõ rằng lớp thứ *i* là đầu ra. `Vps_output_layer_flag[i]` bằng 0 có thể chỉ rõ rằng lớp thứ *i* không phải đầu ra.

Trong phương án của sáng chế, một hoặc nhiều phần tử cú pháp mà chỉ định thông tin PTL cho mỗi tập hợp lớp đầu ra có thể được báo hiệu trong cấu trúc cú pháp mức cao, ví dụ như VPS, DPS, SPS, PPS, APS hoặc bản tin SEI. Viện dẫn tới Fig.31, phần tử cú pháp `num_profile_tile_level` chỉ báo số lượng thông tin cấp độ mẫu trên mỗi OLS trong chuỗi video được mã hóa liên quan đến VPS có thể được báo hiệu trong VPS. Đối với mỗi tập hợp lớp đầu ra, tập hợp của các phần tử cú pháp đối với thông tin PTL hoặc chỉ số mà chỉ báo thông tin PTL cụ thể trong số các mục trong thông tin PTL có thể được báo hiệu đối với mỗi lớp đầu ra.

Trong phương án của sáng chế, `profile_tier_level_idx[i][j]` chỉ rõ chỉ số, vào danh sách của các cấu trúc cú pháp `profile_tier_level()` trong VPS, của cấu trúc cú pháp `profile_tier_level()` mà áp dụng cho lớp thứ *j* của OLS thứ *i*.

Trong phương án của sáng chế, viện dẫn tới FIG.32, các phần tử cú pháp `num_profile_tile_level` và/hoặc `num_output_layer_sets` có thể được báo hiệu khi số lượng lớp lớn nhất lớn hơn 1 (`vps_max_layers_minus1 > 0`).

Trong phương án của sáng chế, viện dẫn tới Fig.32, phần tử cú pháp `vps_output_layers_mode[i]` chỉ báo chế độ báo hiệu lớp đầu ra cho tập hợp lớp đầu ra thứ *i* có thể được biểu diễn trong VPS.

Trong phương án của sáng chế, `vps_output_layers_mode[ i ]` bằng 0 chỉ rõ rằng chỉ lớp cao nhất là đầu ra với tập hợp lớp đầu ra thứ *i*. `vps_output_layer_mode[ i ]` bằng 1 chỉ rõ rằng tất cả các lớp là đầu ra với tập hợp lớp đầu ra thứ *i*. `vps_output_layer_mode[ i ]` bằng 2 chỉ rõ rằng các lớp mà là đầu ra là các lớp với `vps_output_layer_flag[ i ][ j ]` bằng 1 với tập hợp lớp đầu ra thứ *i*. Nhiều giá trị hơn có thể được dành riêng.

Trong phương án của sáng chế, `output_layer_flag[ i ][ j ]` có thể có hoặc có thể không được báo hiệu phụ thuộc vào giá trị của `vps_output_layers_mode[ i ]` cho tập hợp lớp đầu ra thứ *i*.

Trong phương án của sáng chế, viện dẫn tới Fig.32, `vps_ptl_signal_flag[ i ]` có thể được biểu diễn cho tập hợp lớp đầu ra thứ *i*. Phụ thuộc vào giá trị của `vps_ptl_signal_flag[ i ]`, thông tin PTL cho tập hợp lớp đầu ra thứ *i* có thể có hoặc có thể không được báo hiệu.

Trong phương án của sáng chế, viện dẫn tới FIG.33, số lượng ảnh con (ví dụ, `max_subpics_minus1`) trong CVS hiện tại có thể được báo hiệu trong cấu trúc cú pháp mức cao, ví dụ VPS, DPS, SPS, PPS, APS hoặc bản tin SEI.

Trong phương án của sáng chế, viện dẫn tới Fig.33, ký hiệu nhận dạng ảnh con (ví dụ, `sub_pic_id[i]`) cho ảnh con thứ *i* có thể được báo hiệu, khi số lượng các ảnh con lớn hơn 1 (`max_subpics_minus1 > 0`).

Trong phương án của sáng chế, một hoặc nhiều phần tử cú pháp chỉ báo ký hiệu nhận dạng ảnh con thuộc về mỗi lớp của mỗi tập hợp lớp đầu ra có thể được báo hiệu trong VPS. Viện dẫn tới Fig.34, `sub_pic_id_layer[i][j][k]`, mà chỉ báo ảnh con thứ *k* được biểu diễn trong lớp thứ *j* của tập hợp lớp đầu ra thứ *i*. Với thông tin nêu trên, bộ giải mã có thể nhận biết ảnh con nào có thể được giải mã và được xuất ra đối với mỗi lớp của tập hợp lớp đầu ra cụ thể.

Trong phương án của sáng chế, thông tin tiêu đề ảnh (PH-picture header) là cấu trúc cú pháp mà chứa các phần tử cú pháp mà áp dụng tới tất cả lát của ảnh được mã hóa. Thông tin tiêu đề lát có thể là một phần của lát được

mã hóa mà chứa các phần tử dữ liệu liên quan đến tất cả ô hoặc hàng CTU trong ô được biểu diễn trong lát. Đơn vị ảnh (PU) là tập hợp của các đơn vị NAL mà được kết hợp với nhau theo quy tắc phân loại được chỉ rõ, là liên tiếp trong thứ tự giải mã, và chứa, ví dụ, chính xác một ảnh được mã hóa. PU có thể chứa thông tin tiêu đề ảnh (PH) và một hoặc nhiều đơn vị VCL NAL mà cấu thành ảnh được mã hóa.

Trong phương án của sáng chế, SPS (RBSP) có thể khả dụng đối với xử lý giải mã trước SPS đang được viện dẫn, được chứa trong ít nhất một AU với TemporalId bằng 0 hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, SPS (RBSP) có thể khả dụng đối với xử lý trước SPS đang được tham chiếu, được chứa trong ít nhất một AU với TemporalId bằng 0 trong CVS, mà chứa một hoặc nhiều PPS liên quan đến SPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, SPS (RBSP) có thể khả dụng đối với xử lý giải mã trước SPS đang được tham chiếu bởi một hoặc nhiều PPS, được chứa trong ít nhất một PU với nuh_layer_id bằng giá trị nuh_layer_id thấp nhất của các đơn vị PPS NAL mà liên quan đến đơn vị SPS NAL trong CVS, mà chứa một hoặc nhiều PPS liên quan đến SPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, SPS (RBSP) có thể khả dụng đối với xử lý giải mã trước SPS đang được tham chiếu bởi một hoặc nhiều PPS, được chứa trong ít nhất một PU với TemporalId bằng 0 và nuh_layer_id bằng giá trị nuh_layer_id thấp nhất của các đơn vị PPS NAL mà liên quan đến đơn vị SPS NAL hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, SPS (RBSP) có thể khả dụng đối với xử lý giải mã trước SPS đang được tham chiếu bởi một hoặc nhiều PPS, được chứa trong ít nhất một PU với TemporalId bằng 0 và nuh_layer_id bằng giá trị nuh_layer_id thấp nhất của các đơn vị PPS NAL mà liên quan đến đơn vị SPS NAL trong CVS, mà chứa một hoặc nhiều PPS liên quan đến SPS, hoặc được

cấp thông qua phương tiện phía ngoài hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, `pps_seq_parameter_set_id` chỉ rõ giá trị của `sps_seq_parameter_set_id` đối với SPS được tham chiếu. Trong ví dụ của sáng chế, giá trị của `pps_seq_parameter_set_id` có thể giống nhau trong tất cả PPS mà được viện dẫn tới bởi các ảnh được mã hóa trong CLVS.

Trong phương án của sáng chế, tất cả đơn vị SPS NAL với giá trị cụ thể của `sps_seq_parameter_set_id` trong CVS có thể có cùng nội dung.

Trong phương án của sáng chế, bất kể các giá trị `nuh_layer_id`, các đơn vị SPS NAL có thể chia sẻ cùng không gian giá trị của `sps_seq_parameter_set_id`.

Trong phương án của sáng chế, giá trị `nuh_layer_id` của đơn vị SPS NAL có thể bằng giá trị `nuh_layer_id` thấp nhất của các đơn vị PPS NAL mà liên quan đến đơn vị SPS NAL.

Trong phương án của sáng chế, khi SPS với `nuh_layer_id` bằng *m* được viện dẫn tới bởi một hoặc nhiều PPS với `nuh_layer_id` bằng *n*. Lớp với `nuh_layer_id` bằng *m* có thể tương tự lớp với `nuh_layer_id` bằng *n* hoặc lớp tham chiếu (trực tiếp hoặc gián tiếp) của lớp với `nuh_layer_id` bằng *m*.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn, được chứa trong ít nhất một AU với TemporalId bằng TemporalId của đơn vị PPS NAL hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn, được chứa trong ít nhất một AU với TemporalId bằng TemporalId của đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn NAL lát được mã hóa) hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn bởi một hoặc nhiều PH (hoặc các

đơn vị NAL lát được mã hóa), được chứa trong ít nhất một PU với `nuh_layer_id` bằng giá trị `nuh_layer_id` thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa) liên quan đến PPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được tham chiếu bởi một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa), được chứa trong ít nhất một PU với `TemporalId` bằng `TemporalId` của đơn vị PPS NAL và `nuh_layer_id` bằng giá trị `nuh_layer_id` thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa) liên quan đến PPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, `ph_pic_parameter_set_id` trong PH chỉ rõ giá trị của `pps_pic_parameter_set_id` đối với PPS được tham chiếu được sử dụng. Giá trị của `pps_seq_parameter_set_id` có thể giống nhau trong tất cả PPS mà được viện dẫn tới bởi các ảnh được mã hóa trong CLVS.

Trong phương án, tất cả các đơn vị PPS NAL với giá trị cụ thể của `pps_pic_parameter_set_id` trong PU có thể có cùng nội dung.

Trong phương án của sáng chế, bất kể các giá trị `nuh_layer_id`, các đơn vị PPS NAL có thể chia sẻ cùng không gian giá trị của `pps_pic_parameter_set_id`.

Trong phương án của sáng chế, giá trị `nuh_layer_id` của đơn vị PPS NAL có thể bằng giá trị `nuh_layer_id` thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị NAL. Đơn vị NAL liên quan đến đơn vị PPS NAL.

Trong phương án của sáng chế, khi PPS với `nuh_layer_id` bằng m được viện dẫn tới bởi một hoặc nhiều đơn vị NAL lát được mã hóa với `nuh_layer_id` bằng n . Lớp với `nuh_layer_id` bằng m có thể tương tự lớp với

nuh_layer_id bằng n hoặc lớp tham chiếu (trực tiếp hoặc gián tiếp) của lớp với nuh_layer_id bằng m .

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn, được chứa trong ít nhất một AU với TemporalId bằng TemporalId của đơn vị PPS NAL hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn, được chứa trong ít nhất một AU với TemporalId bằng TemporalId của đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn NAL lát được mã hóa) hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được viện dẫn bởi một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa), được chứa trong ít nhất một PU với nuh_layer_id bằng giá trị nuh_layer_id thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa) liên quan đến PPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, PPS (RBSP) có thể khả dụng đối với xử lý giải mã trước PPS đang được tham chiếu bởi một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa), được chứa trong ít nhất một PU với TemporalId bằng TemporalId của đơn vị PPS NAL và nuh_layer_id bằng giá trị nuh_layer_id thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị PPS NAL trong CVS, mà chứa một hoặc nhiều PH (hoặc các đơn vị NAL lát được mã hóa) liên quan đến PPS, hoặc được cấp thông qua phương tiện phía ngoài.

Trong phương án của sáng chế, ph_pic_parameter_set_id trong PH chỉ rõ giá trị của pps_pic_parameter_set_id đối với PPS được tham chiếu được

sử dụng. Giá trị của `pps_seq_parameter_set_id` có thể giống nhau trong tất cả PPS mà được viện dẫn tới bởi các ảnh được mã hóa trong CLVS.

Trong phương án, tất cả các đơn vị PPS NAL với giá trị cụ thể của `pps_pic_parameter_set_id` trong PU có thể có cùng nội dung.

Trong phương án của sáng chế, bất kể các giá trị `nuh_layer_id`, các đơn vị PPS NAL có thể chia sẻ cùng không gian giá trị của `pps_pic_parameter_set_id`.

Trong phương án của sáng chế, giá trị `nuh_layer_id` của đơn vị PPS NAL có thể bằng giá trị `nuh_layer_id` thấp nhất của các đơn vị NAL lát được mã hóa mà liên quan đến đơn vị NAL mà liên quan đến đơn vị PPS NAL.

Trong phương án của sáng chế, khi PPS với `nuh_layer_id` bằng m được viện dẫn tới bởi một hoặc nhiều đơn vị NAL lát được mã hóa với `nuh_layer_id` bằng n . Lớp với `nuh_layer_id` bằng m có thể tương tự lớp với `nuh_layer_id` bằng n hoặc lớp tham chiếu (trực tiếp hoặc gián tiếp) của lớp với `nuh_layer_id` bằng m .

Lớp đầu ra chỉ báo lớp của tập hợp lớp đầu ra mà là đầu ra. Tập hợp lớp đầu ra (OLS) chỉ báo tập hợp của các lớp bao gồm tập hợp của các lớp được chỉ rõ, trong đó một hoặc nhiều lớp trong tập hợp của các lớp được chỉ rõ là các lớp đầu ra. Chỉ số lớp OLS là chỉ số của lớp trong OLS. Chỉ số lớp OLS có thể trỏ tới danh sách của các lớp trong OLS.

Lớp con chỉ báo lớp có khả năng biến đổi theo thời gian của dòng bit có khả năng biến đổi theo thời gian bao gồm các đơn vị VCL NAL với giá trị cụ thể của biến `TemporalId` và các đơn vị không phải VCL NAL được kết hợp. Biểu diễn lớp con chỉ báo tập con của dòng bit bao gồm các đơn vị NAL của lớp con cụ thể và lớp con thấp hơn.

VPS RBSP có thể khả dụng đối với xử lý giải mã trước VPS RBSP đang được viện dẫn, được chứa trong ít nhất một AU với `TemporalId` bằng 0 hoặc được cấp thông qua phương tiện phía ngoài. Tất cả đơn vị VPS NAL với

giá trị cụ thể của `vps_video_parameter_set_id` trong CVS có thể có cùng nội dung.

`vps_video_parameter_set_id` cấp ký hiệu nhận dạng đối với VPS để tham chiếu bởi các phần tử cú pháp khác. Giá trị của `vps_video_parameter_set_id` có thể lớn hơn 0.

`vps_max_layers_minus1` cộng 1 chỉ rõ số lượng lớp được cho phép lớn nhất trong mỗi CVS mà viện dẫn tới VPS.

`vps_max_sublayers_minus1` cộng 1 chỉ rõ số lượng lớp con theo thời gian lớn nhất mà có thể được hiện diện trong lớp trong mỗi CVS mà viện dẫn tới VPS. Giá trị của `vps_max_sublayers_minus1` có thể nằm trong đoạn từ 0 đến 6.

Trong ví dụ của sáng chế, cờ (ví dụ, `vps_all_layers_same_num_sublayers_flag`) bằng 1 chỉ rõ rằng số lượng lớp con theo thời gian là giống nhau đối với tất cả lớp trong mỗi CVS mà viện dẫn tới VPS. Cờ (ví dụ, `vps_all_layers_same_num_sublayers_flag`) bằng 0 chỉ rõ rằng các lớp trong mỗi CVS mà viện dẫn tới VPS có thể có hoặc có thể không có cùng số lượng lớp con theo thời gian. Khi không được hiện diện, giá trị của cờ (ví dụ, `vps_all_layers_same_num_sublayers_flag`) được giả định bằng 1.

Cờ (ví dụ, `vps_all_independent_layers_flag`) bằng 1 chỉ rõ rằng tất cả các lớp trong CVS được mã hóa độc lập mà không sử dụng dự đoán liên lớp. Cờ (ví dụ, `vps_all_independent_layers_flag`) bằng 0 chỉ rõ rằng một hoặc nhiều lớp trong CVS có thể sử dụng dự đoán liên lớp. Khi không được hiện diện, giá trị của cờ (ví dụ, `vps_all_independent_layers_flag`) được giả định bằng 1.

`vps_layer_id[i]` có thể chỉ rõ giá trị `nuh_layer_id` của lớp thứ *i*. Đối với bất kỳ hai giá trị nguyên không âm của *m* và *n*, giá trị của `vps_layer_id[m]` có thể nhỏ hơn `vps_layer_id[n]` khi *m* nhỏ hơn *n*.

`vps_independent_layer_flag[i]` bằng 1 có thể chỉ rõ rằng lớp với chỉ số *i* không sử dụng dự đoán liên lớp. `vps_independent_layer_flag[i]` bằng 0 có

thể chỉ rõ rằng lớp với chỉ số i có thể sử dụng dự đoán liên lớp và các phần tử cú pháp $vps_direct_ref_layer_flag[i][j]$ với j trong đoạn từ 0 đến $i - 1$, được hiện diện trong VPS. Khi không được hiện diện, các giá trị của $vps_independent_layer_flag[i]$ được giả định bằng 1.

$vps_direct_ref_layer_flag[i][j]$ bằng 0 có thể chỉ rõ rằng lớp với chỉ số j không phải là lớp tham chiếu trực tiếp đối với lớp có chỉ số i . $vps_direct_ref_layer_flag[i][j]$ bằng 1 có thể chỉ rõ rằng lớp với chỉ số j là lớp tham chiếu trực tiếp đối với lớp có chỉ số i . Khi $vps_direct_ref_layer_flag[i][j]$ không được hiện diện với i và j trong đoạn từ 0 đến $vps_max_layers_minus1$, tham số này được giả định bằng 0. Khi $vps_independent_layer_flag[i]$ bằng 0, có thể có ít nhất một giá trị của j trong đoạn từ 0 đến $i - 1$, sao cho giá trị của $vps_direct_ref_layer_flag[i][j]$ bằng 1.

Các biến $NumDirectRefLayers[i]$, $DirectRefLayerIdx[i][d]$, $NumRefLayers[i]$, $RefLayerIdx[i][r]$, và $LayerUsedAsRefLayerFlag[j]$ có thể được thu nhận 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
            LayerUsedAsRefLayerF
            lag[ j ] = 1
        }
        if( dependencyFlag[
            i ][ j ] )
            RefLayerIdx[ i ][
                r++ ] = j
    }
    NumDirectRefLayers[ i ] = d
    NumRefLayers[ i ] = r
}

```

Biến GeneralLayerIdx [i], mà chỉ rõ chỉ số lớp của lớp có nuh_layer_id bằng vps_layer_id [i], có thể được thu nhận như sau:

```

for( i = 0; i <= vps_max_layers_minus1; i++ )
    GeneralLayerIdx[ vps_layer_id[ i ] ] = i

```

Trong ví dụ của sáng chế, đối với bất kỳ hai giá trị khác nhau của i và j, cả trong đoạn từ 0 đến vps_max_layers_minus1, khi cờ (ví dụ, dependencyFlag [i][j]) bằng 1. Yêu cầu củ tính tương thích dòng bit có thể là các giá trị của chroma_format_idc và bit_depth_minus8 mà áp dụng tới lớp thứ i có thể bằng các giá trị của chroma_format_idc và bit_depth_minus8, một cách lần lượt, mà áp dụng tới lớp thứ j.

max_tid_ref_present_flag [i] bằng 1 có thể chỉ rõ rằng phần tử cú pháp max_tid_il_ref_pics_plus1[i] được hiện diện. max_tid_ref_present_flag[i] bằng 0 có thể chỉ rõ rằng phần tử cú pháp max_tid_il_ref_pics_plus1[i] không được hiện diện.

`max_tid_il_ref_pics_plus1[i]` bằng 0 có thể chỉ rõ rằng dự đoán liên lớp không được sử dụng bởi các ảnh không phải IRAP của lớp thứ *i*. `max_tid_il_ref_pics_plus1[i]` lớn hơn 0 có thể chỉ rõ rằng, để giải mã các ảnh của lớp thứ *i*, không có ảnh với `TemporalId` lớn hơn `max_tid_il_ref_pics_plus1[i]-1` được sử dụng như là ILRP. Khi không được hiện diện, giá trị của `max_tid_il_ref_pics_plus1[i]` được giả định bằng 7.

`each_layer_is_an_ols_flag` bằng 1 có thể chỉ rõ rằng mỗi OLS chỉ chứa một lớp và bản thân mỗi lớp trong CVS mà viện dẫn tới VPS là OLS với một lớp được bao gồm là chỉ lớp đầu ra. `each_layer_is_an_ols_flag` bằng 0 có thể chỉ rõ rằng OLS có thể chứa nhiều hơn một lớp. Nếu `vps_max_layers_minus1` bằng 0, giá trị của `each_layer_is_an_ols_flag` được giả định bằng 1. Nếu không phải, khi `vps_all_independent_layers_flag` bằng 0, giá trị của `each_layer_is_an_ols_flag` được giả định bằng 0.

`ols_mode_idc` bằng 0 có thể chỉ rõ rằng tổng số lượng OLS được chỉ rõ bởi VPS bằng `vps_max_layers_minus1 + 1`, OLS thứ *i* bao gồm các lớp với các chỉ số lớp từ 0 đến *i*, và đối với mỗi OLS chỉ lớp cao nhất trong OLS được xuất ra.

`ols_mode_idc` bằng 1 có thể chỉ rõ rằng tổng số lượng OLS được chỉ rõ bởi VPS bằng `vps_max_layers_minus1 + 1`, OLS thứ *i* bao gồm các lớp với các chỉ số lớp từ 0 đến *i*. Đối với mỗi OLS, tất cả lớp trong OLS được xuất ra.

`ols_mode_idc` bằng 2 có thể chỉ rõ rằng tổng số lượng OLS được chỉ rõ bởi VPS được báo hiệu rõ ràng và đối với mỗi OLS, các lớp đầu ra được báo hiệu rõ ràng và các lớp khác là các lớp mà là các lớp tham chiếu trực tiếp hoặc gián tiếp của các lớp đầu ra của OLS.

Giá trị của `ols_mode_idc` có thể nằm trong đoạn từ 0 đến 2. Giá trị 3 của `ols_mode_idc` có thể được dành riêng đối với các kỹ thuật hoặc tiêu chuẩn mã hóa video định trước, ví dụ, bởi ITU-T hoặc ISO/IEC.

Khi cờ (ví dụ, `vps_all_independent_layers_flag`) bằng 1 và cờ (ví dụ, `each_layer_is_an_ols_flag`) bằng 0, giá trị của `ols_mode_idc` được giả định bằng 2.

`num_output_layer_sets_minus1` cộng 1 có thể chỉ rõ tổng số lượng OLS được chỉ rõ bởi VPS khi `ols_mode_idc` bằng 2.

Biến `TotalNumOlss`, mà chỉ rõ tổng số lượng OLS được chỉ rõ bởi VPS, có thể được thu nhận như sau:

```
if( vps_max_layers_minus1 == 0 )
    TotalNumOlss = 1
else if( each_layer_is_an_ols_flag == 1 && ols_mode_idc == 0 )
    TotalNumOlss = vps_max_layers_minus1 + 1
else if( ols_mode_idc == 1 )
    TotalNumOlss = vps_max_layers_minus1 + 1
else if( ols_mode_idc == 2 )
```

`TotalNumOlss = num_output_layer_sets_minus1 + 1`
`ols_output_layer_flag[i][j]` bằng 1 có thể chỉ rõ rằng lớp với `nuh_layer_id` bằng `vps_layer_id[j]` là lớp đầu ra của OLS thứ *i* khi `ols_mode_idc` bằng 2. `ols_output_layer_flag[i][j]` bằng 0 chỉ rõ rằng lớp với `nuh_layer_id` bằng `vps_layer_id[j]` không phải lớp đầu ra của OLS thứ *i* khi `ols_mode_idc` bằng 2.

Biến `NumOutputLayersInOls[i]`, mà chỉ rõ số lượng lớp đầu ra trong OLS thứ *i*, biến `NumSubLayersInLayerInOLS[i][j]`, mà chỉ rõ số lượng lớp con trong lớp thứ *j* trong OLS thứ *i*, biến `OutputLayerIdInOls[i][j]`, mà chỉ rõ giá trị `nuh_layer_id` của lớp đầu ra thứ *j* trong OLS thứ *i*, và biến `LayerUsedAsOutputLayerFlag[k]`, mà chỉ rõ rằng lớp thứ *k* có được sử dụng như là lớp đầu ra trong ít nhất một OLS hay không, có thể được thu nhận như sau:

```
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 == 1 && 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 == 1 &&
        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
    LayerUsedAsOutputLayerF
    lag[ k ] = 1
    OutputLayerIdx[ i ][ j ] = k
    OutputLayerIdInOls[ i ][ j++ ] = vps_layer_id[ k ]
    NumSubLayersInLayerInOLS[ i ][ j ] =
        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[
        OutputLayerIdInOls[ i ][ j ] ] )
        NumSubLayersInLayerInOLS[ i ][ RefLayerIdx[
            idx ][ k ] ] =
            max_tid_il_ref_pics_plus1[ OutputLayerIdInOls[ i ][ j
                ] ]
    }
}
}
}

```

Đối với mỗi giá trị của i trong đoạn từ 0 đến $vps_max_layers_minus1$, các giá trị của $LayerUsedAsRefLayerFlag[i]$ và

LayerUsedAsOutputLayerFlag [i] có thể đều không bằng 0. Do đó, có thể không có lớp mà không phải lớp đầu ra của ít nhất một OLS hoặc lớp tham chiếu trực tiếp của bất kỳ lớp khác.

Đối với mỗi OLS, có thể có ít nhất một lớp mà là lớp đầu ra. Do đó, đối với bất kỳ giá trị của i trong đoạn từ 0 đến TotalNumOlss - 1, giá trị của NumOutputLayersInOls [i] có thể lớn hơn hoặc bằng 1.

Biến NumLayersInOls[i], mà chỉ rõ số lượng lớp trong OLS thứ i, và biến LayerIdInOls[i][j], mà chỉ rõ giá trị nuh_layer_id của lớp thứ j trong OLS thứ i, được thu nhận như sau:

```

NumLayersInOls[ 0 ] = 1
LayerIdInOls[ 0 ][ 0 ] =
vps_layer_id[ 0 ] for( i = 1; i <
TotalNumOlss; i++ ) {
    if(
        each_layer_is_an_
        ols_flag      ) {
        NumLayersInOls[ i
        ] = 1
        LayerIdInOls[ i ][ 0 ] = vps_layer_id[ i ]
    } else if( ols_mode_idc == 0 1 1 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
    }
}

```

```
}
```

Biến `OlsLayerIdx [i][j]`, mà chỉ rõ chỉ số lớp OLS của lớp với `nuh_layer_id` bằng `LayerIdInOls [i][j]`, có thể được thu nhận như sau:

```
for( i = 0; i < TotalNumOlss; i++ )
  for j = 0; j < NumLayersInOls[i]; j++ )
    OlsLayerIdx[ i ][ LayerIdInOls[i][j]] = j
```

Lớp thấp nhất trong mỗi OLS có thể là lớp độc lập. Do đó, đối với mỗi `i` trong đoạn từ 0 đến `TotalNumOlss - 1`, giá trị của `vps_independent_layer_flag[ GeneralLayerIdx[ LayerIdInOls [i][0]]]` có thể bằng 1.

Mỗi lớp có thể được chứa trong ít nhất một OLS được chỉ rõ bởi VPS. Do đó, đối với mỗi lớp với giá trị cụ thể của `nuh_layer_id` `nuhLayerId` bằng một trong số `vps_layer_id [k]` với `k` trong đoạn từ 0 đến `vps_max_layers_minus1`, có thể có ít nhất một cặp giá trị của `i` và `j`, trong đó `i` nằm trong đoạn từ 0 đến `TotalNumOlss - 1`, và `j` nằm trong đoạn từ `NumLayersInOls [i] - 1`, sao cho giá trị của `LayerIdInOls [i][j]` bằng `nuhLayerId`.

Trong phương án của sáng chế, xử lý giải mã hoạt động như sau đối với ảnh hiện tại `CurrPic`:

- `PictureOutputFlag` được thiết lập như sau:
 - Nếu một trong số các điều kiện sau đây là đúng, `PictureOutputFlag` được thiết lập bằng 0:
 - ảnh hiện tại là ảnh RASL và `NoOutputBeforeRecoveryFlag` của ảnh IRAP được kết hợp bằng 1.

- gdr_enabled_flag bằng 1 và ảnh hiện tại của ảnh GDR với NoOutputBeforeRecoveryFlag bằng 1.
 - gdr_enabled_flag bằng 1, ảnh hiện tại được kết hợp với ảnh GDR với NoOutputBeforeRecoveryFlag bằng 1, và PicOrderCntVal của ảnh hiện tại nhỏ hơn RpPicOrderCntVal của ảnh GDR được kết hợp.
 - sps_video_parameter_set_id lớn hơn 0, ols_mode_idc bằng 0 và AU hiện tại chứa ảnh picA mà thỏa mãn tất cả các điều kiện sau đây
 - PicA có PictureOutputFlag bằng 1.
 - PicA có nuh_layer_id nuhLid lớn hơn của ảnh hiện tại.
 - PicA thuộc về lớp đầu ra của OLS (tức là, OutputLayerIdInOls[TargetOlsIdx][0] bằng nuhLid).
 - sps_video_parameter_set_id lớn hơn 0, ols_mode_idc bằng 2, và
 - ols_output_layer_flag[TargetOlsIdx][GeneralLayerIdx[nuh_layer_id]] bằng 0.
- Nếu không phải, PictureOutputFlag được thiết lập bằng pic_output_flag.

Sau khi tất cả lát của ảnh hiện tại được giải mã, ảnh được giải mã hiện tại được đánh dấu là "được sử dụng cho việc tham chiếu ngắn hạn", và mỗi mục ILRP trong RefPicList [0] hoặc RefPicList [1] được đánh dấu là "được sử dụng cho việc tham chiếu ngắn hạn".

Trong phương án của sáng chế, khi mỗi lớp là tập hợp lớp đầu ra, PictureOutputFlag được thiết lập bằng pic_output_flag, bất kể giá trị của ols_mode_idc.

Trong phương án của sáng chế, PictureOutputFlag được thiết lập bằng 0 khi sps_video_parameter_set_id lớn hơn 0, each_layer_is_an_ols_flag bằng 0, ols_mode_idc bằng 0 và AU hiện

tại chứa ảnh picA mà thỏa mãn tất cả các điều kiện sau đây: PicA có PictureOutputFlag bằng 1, PicA có nuh_layer_id nuhLid lớn hơn so với của ảnh hiện tại và PicA thuộc về lớp đầu ra của OLS (tức là, OutputLayerIdInOls [TargetOlsIdx][0] bằng nuhLid).

Trong phương án của sáng chế, PictureOutputFlag được thiết lập bằng 0 khi sps_video_parameter_set_id lớn hơn 0, each_layer_is_an_ols_flag bằng 0, ols_mode_idc bằng 2, và ols_output_layer_flag [TargetOlsIdx][GeneralLayerIdx[nuh_layer_id]] bằng 0.

FIG.35 thể hiện ví dụ về cấu trúc cú pháp của VPS RBSP. Cấu trúc cú pháp được thể hiện trên FIG.35 là tương tự cấu trúc cú pháp được thể hiện trên FIG.34, và do đó phần mô tả chi tiết được bỏ qua nhằm mục đích ngắn gọn.

Các kỹ thuật được mô tả nêu trên (ví dụ, các kỹ thuật báo hiệu các cờ điều kiện ràng buộc, các tham số độ phân giải thích nghi, và/hoặc loại tương tự) có thể được thực hiện như là phần mềm máy tính sử dụng các lệnh có thể đọc được bởi máy tính và được lưu trữ vật lý trong một hoặc nhiều phương tiện đọc được bởi máy tính. Ví dụ, Fig.36 thể hiện hệ thống máy tính 3600 thích hợp để thực hiện các phương án của đối tượng được bộc lộ.

Phần mềm máy tính có thể được mã hóa nhờ sử dụng bất kỳ mã máy hoặc ngôn ngữ máy tính thích hợp, mà có thể được tập hợp, biên dịch, liên kết, hoặc cơ chế tương tự để tạo ra mã bao gồm các lệnh mà có thể được thực thi trực tiếp, hoặc thông qua diễn dịch, thực thi vi mã, và loại tương tự, bởi một hoặc nhiều bộ xử lý trung tâm (CPU) máy tính, các bộ xử lý đồ họa (GPU), và loại tương tự.

Các lệnh có thể được thực thi trên các loại máy tính khác nhau hoặc các thành phần của chúng, bao gồm, ví dụ, các máy tính cá nhân, máy tính bảng, máy chủ, điện thoại thông minh, thiết bị chơi trò chơi, thiết bị Internet vạn vật, và loại tương tự.

Các bộ phận được thể hiện trên Fig.36 đối với hệ thống máy tính 3600 là ví dụ về mặt bản chất và không dự định đề xuất bất kỳ giới hạn nào đối với phạm vi sử dụng hoặc chức năng của phần mềm máy tính mà thực hiện các phương án của sáng chế. Cấu trúc của các bộ phận không được hiểu là có bất kỳ sự phụ thuộc hoặc yêu cầu liên quan đến bất kỳ một hoặc kết hợp của các bộ phận được minh họa trong phương án ví dụ của hệ thống máy tính 3600.

Hệ thống máy tính 3600 có thể bao gồm các thiết bị đầu vào giao diện người dùng định trước. Thiết bị đầu vào giao diện người dùng này có thể phản hồi lại việc nhập vào bởi một hoặc nhiều người dùng thông qua, ví dụ, việc nhập xúc giác (như: nhấn phím, trượt, di chuyển bao dữ liệu), nhập audio (như: thoại, vỗ tay), nhập trực quan (như: các cử chỉ), nhập khứu giác (không được thể hiện). Các thiết bị giao diện người dùng có thể cũng được sử dụng để quay nội dung đa phương tiện không cần thiết liên quan trực tiếp đến việc nhập nhận thức bởi người dùng, như audio (như: âm nhạc xung quanh, âm nhạc thoại), các ảnh (như: các ảnh được quét, các ảnh chụp thu nhận từ camera ảnh tĩnh), video (như video hai chiều, video ba chiều bao gồm video lập thể).

Các thiết bị giao diện người dùng đầu vào có thể bao gồm một hoặc nhiều trong số (chỉ một trong số các thiết bị được thể hiện): bàn phím 3601, chuột 3602, bàn di chuột 3603, màn chạm 3610, bao dữ liệu (không được thể hiện), cần điều khiển 3605, microphôn 3606, máy quét 3607, camera 3608.

Hệ thống máy tính 3600 có thể cũng bao gồm các thiết bị đầu ra giao diện người dùng định trước. Các thiết bị đầu ra giao diện người dùng này có thể là sự mô phỏng các giác quan của một hoặc nhiều người dùng thông qua, ví dụ, xúc giác, âm thanh, ánh sáng, và mùi/vị giác. Các thiết bị đầu ra giao diện người dùng này có thể bao gồm các thiết bị đầu ra xúc giác (ví dụ xúc giác phản hồi bởi màn chạm 3610, bao dữ liệu (không được thể hiện), hoặc cần điều khiển 3605, nhưng có thể cũng có các thiết bị phản hồi xúc giác mà không đóng vai trò là các thiết bị đầu vào), các thiết bị xuất audio (như: các loa 3609, các tai nghe (không được thể hiện)), các thiết bị đầu ra trực quan (như các màn hình 3610 bao gồm các màn hình CRT, màn hình LCD, màn hình plasma, màn hình

OLED, mỗi trong số chúng có hoặc không có khả năng nhập màn chạm, có hoặc không có khả năng phản hồi xúc giác—một vài trong số chúng có thể có khả năng xuất ra đầu ra trực quan hai chiều hoặc đầu ra nhiều hơn ba chiều thông qua phương tiện như đầu ra lập thể; các kính thực tế ảo (không được thể hiện), các màn hình toàn ký và bề khối (không được thể hiện), và các máy in (không được thể hiện).

Hệ thống máy tính 3600 có thể cũng bao gồm các thiết bị lưu trữ có thể truy nhập được bởi người dùng và phương tiện được kết hợp của chúng như phương tiện quang bao gồm CD/DVD ROM/RW 3620 với CD/DVD hoặc phương tiện tương tự 3621, ổ chóp 3622, ổ cứng tháo rời hoặc ổ bán dẫn 3623, phương tiện từ truyền thống như băng và đĩa mềm (không được thể hiện), các thiết bị dựa trên ROM/ASIC/PLD chuyên dụng như các khóa điện tử bảo mật (không được thể hiện), và loại tương tự.

Người có hiểu biết trung bình trong lĩnh vực kỹ thuật cũng sẽ hiểu rằng thuật ngữ “phương tiện đọc được bởi máy tính” như được sử dụng kết hợp với đối tượng được bộc lộ không bao hàm môi trường truyền, các sóng mang, hoặc các tín hiệu chuyển tiếp khác.

Hệ thống máy tính 3600 có thể cũng bao gồm giao diện 3654 tới một hoặc nhiều mạng truyền thông 3655. Các mạng có thể ví dụ là mạng không dây, nối dây, quang học. Các mạng có thể là mạng cục bộ, mạng diện rộng, mạng đô thị, mạng phương tiện giao thông và công nghiệp, thời gian thực, dung sai trễ, và v.v. Các ví dụ của các mạng bao gồm các mạng cục bộ như Ethernet, LAN không dây, mạng tế bào bao gồm GSM, 3G, 4G, 5G, LTE và loại tương tự, TV các mạng số diện rộng có dây hoặc không dây bao gồm TV cáp, TV vệ tinh, và TV quảng bá mặt đất, phương tiện giao thông và công nghiệp bao gồm CANBus, và v.v. Các mạng nhất định thường yêu cầu các bộ thích ứng giao diện mạng phía ngoài mà được gắn vào cổng dữ liệu mục đích chung định trước hoặc các kênh truyền ngoại vi 3649 (như, ví dụ các cổng USB của hệ thống máy tính 3600); các phần khác thường được tích hợp vào lõi của hệ thống máy tính 3600 bằng cách gắn tới kênh truyền hệ thống như được mô tả dưới đây (ví dụ giao

diện Ethernet vào hệ thống máy tính PC hoặc giao diện mạng tế bào vào hệ thống máy tính điện thoại thông minh). Nhờ sử dụng bất kỳ các mạng này, hệ thống máy tính 3600 có thể truyền thông với các thực thể khác. Việc truyền thông này có thể là chỉ thu đơn hướng (ví dụ, TV quảng bá), chỉ gửi đơn hướng (ví dụ các thiết bị CANbus tới CANbus định trước), hoặc hai chiều, ví dụ tới các hệ thống máy tính khác nhờ sử dụng các mạng số diện rộng hoặc cục bộ. Các giao thức định trước và các ngăn giao thức có thể được sử dụng trên mỗi mạng và giao diện mạng này như được mô tả nêu trên.

Các thiết bị giao diện người dùng, các thiết bị lưu trữ có thể truy nhập được bởi người dùng, và các giao diện mạng nêu trên có thể được gắn vào lõi 3640 của hệ thống máy tính 3600.

Lõi 3640 có thể bao gồm một hoặc nhiều bộ xử lý trung tâm (CPU) 3641, các bộ xử lý đồ họa (GPU) 3642, các bộ xử lý khả trình chuyên dụng dưới dạng của mảng cổng khả trình dạng trường (FPGA) 3643, các bộ tăng tốc phần cứng đối với các tác vụ định trước 3644, các bộ điều chỉnh thích ứng đồ họa (~~50), và v.v. Các thiết bị này, cùng với bộ nhớ chỉ đọc (ROM) 3645, bộ nhớ truy nhập ngẫu nhiên 3646, bộ lưu trữ khối bên trong như các ổ cứng không thể truy nhập bởi người dùng bên trong, SSD, và loại tương tự 3647, có thể được kết nối thông qua kênh truyền hệ thống 3648. Trong một vài hệ thống máy tính, kênh truyền hệ thống 3648 có thể truy nhập được trong dạng của một hoặc nhiều đầu cắm vật lý để cho phép các phần mở rộng bởi các CPU, GPU bổ sung, và loại tương tự. Các thiết bị ngoại vi có thể được gắn trực tiếp tới kênh truyền hệ thống của lõi 3648, hoặc thông qua kênh truyền ngoại vi 3649. Trong ví dụ của sáng chế, màn hình 3610 có thể được kết nối tới bộ điều chỉnh thích ứng đồ họa 3650. Các cấu trúc của các kênh truyền ngoại vi bao gồm PCI, USB, và loại tương tự.

Các CPU 3641, GPU 3642, FPGA 3643, và các bộ tăng tốc 3644 có thể thực thi các lệnh định trước mà, một cách kết hợp, có thể thực thi mã máy tính nêu trên. Mã máy tính có thể được lưu trữ trong ROM 3645 hoặc RAM 3646. Dữ liệu chuyển tiếp có thể cũng được lưu trữ trong RAM 3646, trong khi

dữ liệu cố định có thể được lưu trữ ví dụ, trong bộ nhớ khối bên trong 3647. Việc lưu trữ nhanh và truy xuất tới bất kỳ thiết bị bộ nhớ có thể được cho phép thông qua việc sử dụng của bộ nhớ lưu trữ đệm, mà có thể được kết hợp mật thiết với một hoặc nhiều CPU 3641, GPU 3642, bộ lưu trữ cỡ lớn 3647, ROM 3645, RAM 3646, và loại tương tự.

Phương tiện đọc được bởi máy tính có thể có mã máy tính trên đó để thực hiện các hoạt động được thực hiện bởi máy tính khác nhau. Phương tiện và mã máy tính có thể được thiết kế và xây dựng đặc biệt cho mục đích của sáng chế, hoặc chúng có thể là loại được biết đến rộng rãi và khả dụng đối với người có hiểu biết trung bình trong lĩnh vực phần mềm máy tính.

Theo ví dụ của sáng chế và không bị giới hạn, hệ thống máy tính có cấu trúc 3600, và cụ thể, lõi 3640 có thể cung cấp chức năng như là kết quả của các bộ xử lý (bao gồm các CPU, GPU, FPGA, các bộ tăng tốc, và loại tương tự) mà thực thi phần mềm trong một hoặc nhiều phương tiện hữu hình đọc được bởi máy tính. Các phương tiện đọc được bởi máy tính này có thể là phương tiện được kết hợp với bộ lưu trữ khối có thể truy nhập bởi người dùng như được giới thiệu nêu trên, cũng như bộ lưu trữ định trước của lõi 3640 mà có bản chất bất biến, như bộ lưu trữ khối bên trong của lõi 3647 hoặc ROM 3645. Phần mềm mà thực hiện các phương án của sáng chế có thể được lưu trữ trong các thiết bị này và được thực thi bởi lõi 3640. Phương tiện đọc được bởi máy tính có thể bao gồm một hoặc nhiều thiết bị bộ nhớ hoặc chip, theo nhu cầu cụ thể. Phần mềm có thể làm cho lõi 3640 và một cách cụ thể, các bộ xử lý trong đó (bao gồm CPU, GPU, FPGA, và loại tương tự) thực hiện các xử lý cụ thể hoặc các phần cụ thể của các xử lý cụ thể được mô tả ở đây, bao gồm xác định các cấu trúc dữ liệu được lưu trữ trong RAM 3646 và cải biến các cấu trúc dữ liệu này theo các xử lý được xác định bởi phần mềm. Ngoài ra hoặc như là tùy chọn khác, hệ thống máy tính có thể cung cấp chức năng như là kết quả của mạng cứng logic hoặc được lắp trong mạch (ví dụ: bộ tăng tốc 3644), mà có thể hoạt động trong vị trí của hoặc cùng với phần mềm để thực hiện các xử lý cụ thể hoặc các phần cụ thể của các xử lý cụ thể được mô tả ở đây. Việc tham chiếu tới phần mềm có thể bao

hàm logic, và ngược lại, nếu cần. Việc tham chiếu tới phương tiện đọc được bởi máy tính có thể bao hàm mạch (như mạch tích hợp (IC)) mà lưu trữ phần mềm để thực thi, mạch thực hiện logic để thực thi, hoặc cả hai, nếu cần. Sáng chế bao hàm kết hợp thích hợp của phần cứng và phần mềm.

Phụ lục A: Ký hiệu viết tắt

JEM: joint exploration model - mô hình khảo sát chung

VVC: versatile video coding - mã hóa video đa năng

BMS: benchmark set - tập hợp điểm tiêu chuẩn

MV: Motion Vector - Vector chuyển động

HEVC High Efficiency Video Coding-Mã hóa video hiệu quả cao

SEI: Supplementary Enhancement Information - Thông tin tăng cường bổ sung

VUI: Video Usability Information - Thông tin có khả năng sử dụng video

GOP: Groups of Pictures - Nhóm của các ảnh

TU: Transform Units - Đơn vị biến đổi

PU: Prediction Units - Đơn vị dự đoán

CTU: Coding Tree Units - Đơn vị cây mã hóa

CTB: Coding Tree Blocks - Khối cây mã hóa

PB: Prediction Blocks - Khối dự đoán

HRD: Hypothetical Reference Decoder - Bộ giải mã tham chiếu giả thiết

SNR: Signal Noise Ratio - Tỷ lệ tín hiệu trên tạp âm

CPU: Central Processing Units - Bộ xử lý trung tâm

GPU: Graphics Processing Units - Bộ xử lý đồ họa

CRT: Cathode Ray Tube - Ống tia âm cực

LCD: Liquid-Crystal Display - Màn hình tinh thể lỏng

OLED: Organic Light-Emitting Diode - Điốt phát quang hữu cơ

CD: Compact Disc - Đĩa nén

DVD: Digital Video Disc - Đĩa video số

ROM: Read-only Memory - Bộ nhớ chỉ đọc

RAM: Random Access Memory - Bộ nhớ truy nhập ngẫu nhiên

ASIC: Application-Specific Integrated Circuit - Mạch tích hợp ứng dụng riêng

PLD: Programmable Logic Device - Thiết bị logic khả trình

LAN: Local Area Network - Mạng vùng cục bộ

GSM: Global System for Mobile communications - Hệ thống truyền thông di động toàn cầu

LTE: Long Term Evolution - Phát triển dài hạn

CANBus: Controller Area Network Bus - Kênh truyền mạng vùng điều khiển

USB: Universal Serial Bus - Kênh truyền nối tiếp đa năng

PCI: Peripheral Component Interconnect - Liên kết thành phần ngoại vi

FPGA: Field Programmable Gate Areas - Mảng cổng khả trình dạng trường

SSD: solid-state drive - Ổ bán dẫn

IC: Integrated Circuit - Mạch tích hợp

CU: Coding Unit - Đơn vị mã hóa

Trong khi sáng chế đã mô tả một vài phương án ví dụ, có các phương án khác, các hoán vị, và các thay thế tương đương khác nhau, mà nằm trong phạm vi của sáng chế. Do đó, sẽ được hiểu rằng người có hiểu biết trung bình trong lĩnh vực kỹ thuật sẽ có thể nghĩ ra các hệ thống và phương pháp, mặc dù không được thể hiện cụ thể hoặc được mô tả ở đây, áp dụng các nguyên tắc của sáng chế và do đó nằm trong bản chất và phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã video trong bộ giải mã, bao gồm:

thu nhận cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa, mỗi cờ cổng mà chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không, thông tin điều kiện ràng buộc dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa;

xác định, dựa trên cờ cổng của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc, rằng nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không, nhóm thông tin điều kiện ràng buộc của cờ cổng bao gồm ít nhất một cờ điều kiện ràng buộc dùng cho tập con của các khối mã hóa;

xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không; và

khôi phục tập con của các khối mã hóa dựa trên thông tin dự đoán.

2. Phương pháp theo điểm 1, trong đó

nhóm thông tin điều kiện ràng buộc của cờ cổng được xác định là được hiện diện trong thông tin điều kiện ràng buộc; và

bước xác định thông tin dự đoán bao gồm xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng.

3. Phương pháp theo điểm 1, trong đó

nhóm thông tin điều kiện ràng buộc của cờ cổng được xác định là không được hiện diện trong thông tin điều kiện ràng buộc.

4. Phương pháp theo điểm 1, trong đó mỗi cờ cổng chỉ báo rằng nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng không được hiện diện

trong thông tin điều kiện ràng buộc, và không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

5. Phương pháp theo điểm 1, trong đó thông tin điều kiện ràng buộc được hiện diện trong cú pháp mức cao đối với ít nhất một ảnh, và tập con của các khối mã hóa bao gồm các khối mã hóa trong ít nhất một ảnh.

6. Phương pháp theo điểm 1, trong đó

một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa,

các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được báo hiệu theo thứ tự định trước, và

bước xác định thông tin dự đoán còn bao gồm

giải mã chạy các cờ điều kiện ràng buộc; và

xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên các cờ điều kiện ràng buộc.

7. Phương pháp theo điểm 2, trong đó

ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng bao gồm các cờ điều kiện ràng buộc được báo hiệu theo thứ tự định trước, và

bước xác định thông tin dự đoán còn bao gồm giải mã chạy các cờ điều kiện ràng buộc.

8. Phương pháp theo điểm 2, trong đó

độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng được hiện diện trong thông tin điều kiện ràng buộc.

9. Phương pháp theo điểm 1, trong đó

một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa;

đối với mỗi một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc, độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong một trong số một hoặc

nhiều của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng được hiện diện trong thông tin điều kiện ràng buộc; và

số lượng độ dịch được báo hiệu trong dòng bit được mã hóa.

10. Phương pháp theo điểm 1, trong đó

một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa,

một hoặc nhiều phần tử cú pháp trong thông tin điều kiện ràng buộc được căn chỉnh theo byte, một hoặc nhiều phần tử cú pháp bao gồm cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc và các cờ điều kiện ràng buộc trong một hoặc nhiều của ít nhất một nhóm thông tin điều kiện ràng buộc.

11. Thiết bị giải mã video, bao gồm hệ mạch xử lý có cấu trúc để:

thu nhận cờ cổng đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa, mỗi cờ cổng mà chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không, thông tin điều kiện ràng buộc dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa;

xác định, dựa trên cờ cổng của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc, rằng nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không, nhóm thông tin điều kiện ràng buộc của cờ cổng bao gồm ít nhất một cờ điều kiện ràng buộc dùng cho tập con của các khối mã hóa;

xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ cổng có được hiện diện trong thông tin điều kiện ràng buộc hay không; và

khôi phục tập con của các khối mã hóa dựa trên thông tin dự đoán.

12. Thiết bị theo điểm 11, trong đó

nhóm thông tin điều kiện ràng buộc của cờ cổng được xác định là được hiện diện trong thông tin điều kiện ràng buộc; và

mạch xử lý có cấu trúc để xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng.

13. Thiết bị theo điểm 11, trong đó

nhóm thông tin điều kiện ràng buộc của cờ cổng được xác định là không được hiện diện trong thông tin điều kiện ràng buộc.

14. Thiết bị theo điểm 11, trong đó mỗi cờ cổng chỉ báo rằng nhóm thông tin điều kiện ràng buộc tương ứng với cờ cổng tương ứng không được hiện diện trong thông tin điều kiện ràng buộc, và không có các cờ điều kiện ràng buộc được hiện diện trong thông tin điều kiện ràng buộc.

15. Thiết bị theo điểm 11, trong đó thông tin điều kiện ràng buộc được hiện diện trong cú pháp mức cao đối với ít nhất một ảnh, và tập con của các khối mã hóa bao gồm các khối mã hóa trong ít nhất một ảnh.

16. Thiết bị theo điểm 11, trong đó

một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa,

các cờ điều kiện ràng buộc trong một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được báo hiệu theo thứ tự định trước, và

mạch xử lý có cấu trúc để:

giải mã chạy các cờ điều kiện ràng buộc; và

xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên các cờ điều kiện ràng buộc.

17. Thiết bị theo điểm 12, trong đó

ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ cổng bao gồm các cờ điều kiện ràng buộc được báo hiệu theo thứ tự định trước, và

mạch xử lý có cấu trúc để giải mã-chạy các cờ điều kiện ràng buộc.

18. Thiết bị theo điểm 12, trong đó

độ dịch và độ dài được hiện diện trong thông tin điều kiện ràng buộc mà chỉ báo rằng ít nhất một cờ điều kiện ràng buộc trong nhóm thông tin điều kiện ràng buộc của cờ công được hiện diện trong thông tin điều kiện ràng buộc.

19. Thiết bị theo điểm 11, trong đó

một hoặc nhiều trong số ít nhất một nhóm thông tin điều kiện ràng buộc được hiện diện trong dòng bit video được mã hóa,

một hoặc nhiều phần tử cú pháp trong thông tin điều kiện ràng buộc được căn thẳng theo byte, một hoặc nhiều phần tử cú pháp bao gồm cờ công đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc và các cờ điều kiện ràng buộc trong một hoặc nhiều của ít nhất một nhóm thông tin điều kiện ràng buộc.

20. Phương tiện lưu trữ bất biến đọc được bởi máy tính mà lưu trữ các lệnh có thể thực thi được bởi ít nhất một bộ xử lý để thực hiện:

thu nhận cờ công đối với mỗi ít nhất một nhóm thông tin điều kiện ràng buộc trong thông tin điều kiện ràng buộc được báo hiệu trong dòng bit video được mã hóa, mỗi cờ công mà chỉ báo rằng nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc tương ứng với cờ công tương ứng có được hiện diện trong thông tin điều kiện ràng buộc hay không, thông tin điều kiện ràng buộc dùng cho tập con của các khối mã hóa trong dòng bit được mã hóa;

xác định, dựa trên cờ công của nhóm thông tin điều kiện ràng buộc của ít nhất một nhóm thông tin điều kiện ràng buộc, rằng nhóm thông tin điều kiện ràng buộc của cờ công có được hiện diện trong thông tin điều kiện ràng buộc hay không, nhóm thông tin điều kiện ràng buộc của cờ công bao gồm ít nhất một cờ điều kiện ràng buộc dùng cho tập con của các khối mã hóa;

xác định thông tin dự đoán đối với tập con của các khối mã hóa dựa trên việc nhóm thông tin điều kiện ràng buộc của cờ công có được hiện diện trong thông tin điều kiện ràng buộc hay không; và

khôi phục tập con của các khối mã hóa dựa trên thông tin dự đoán.

1/29

	R01	R02	R03	R04	R05	R06	R07	R08	R09
R10	S11	S12	S13	S14					
R20	S21	S22	S23	S24					
R30	S31	S32	S33	S34					
R40	S41	S42	S43	S44					
R50									
R60									
R70									

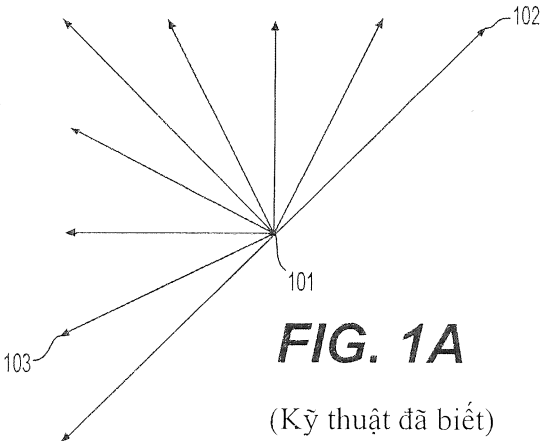


FIG. 1A

(Kỹ thuật đã biết)

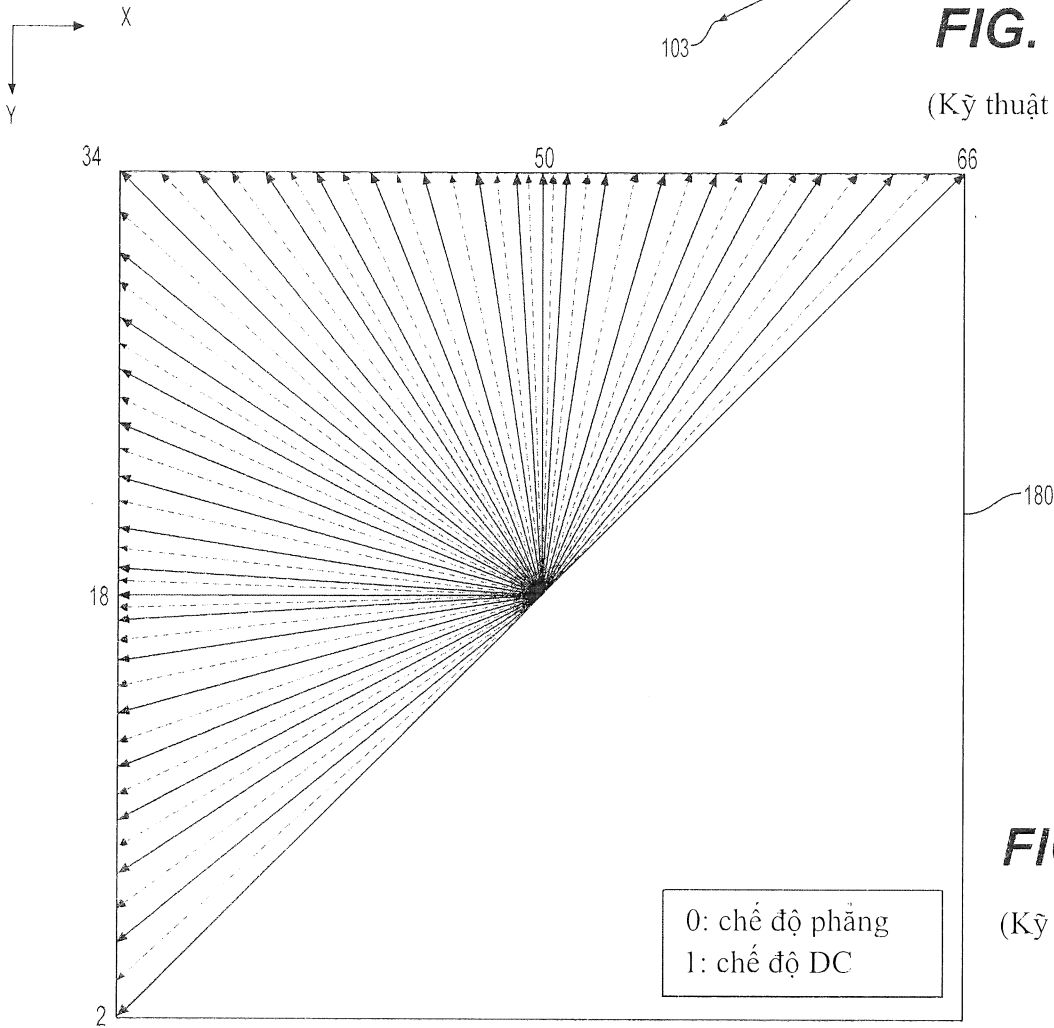
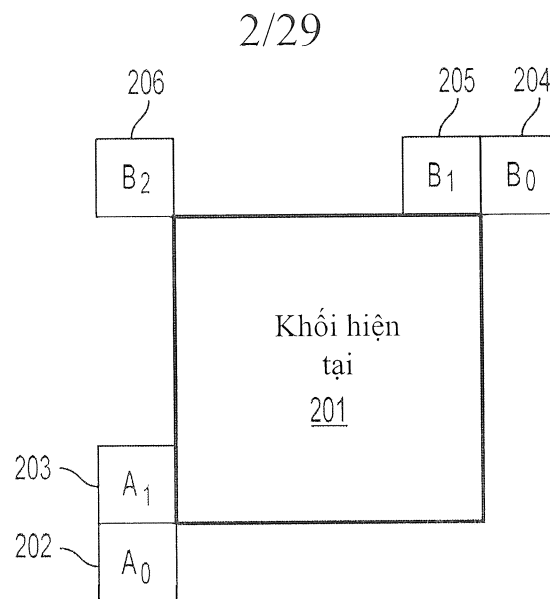


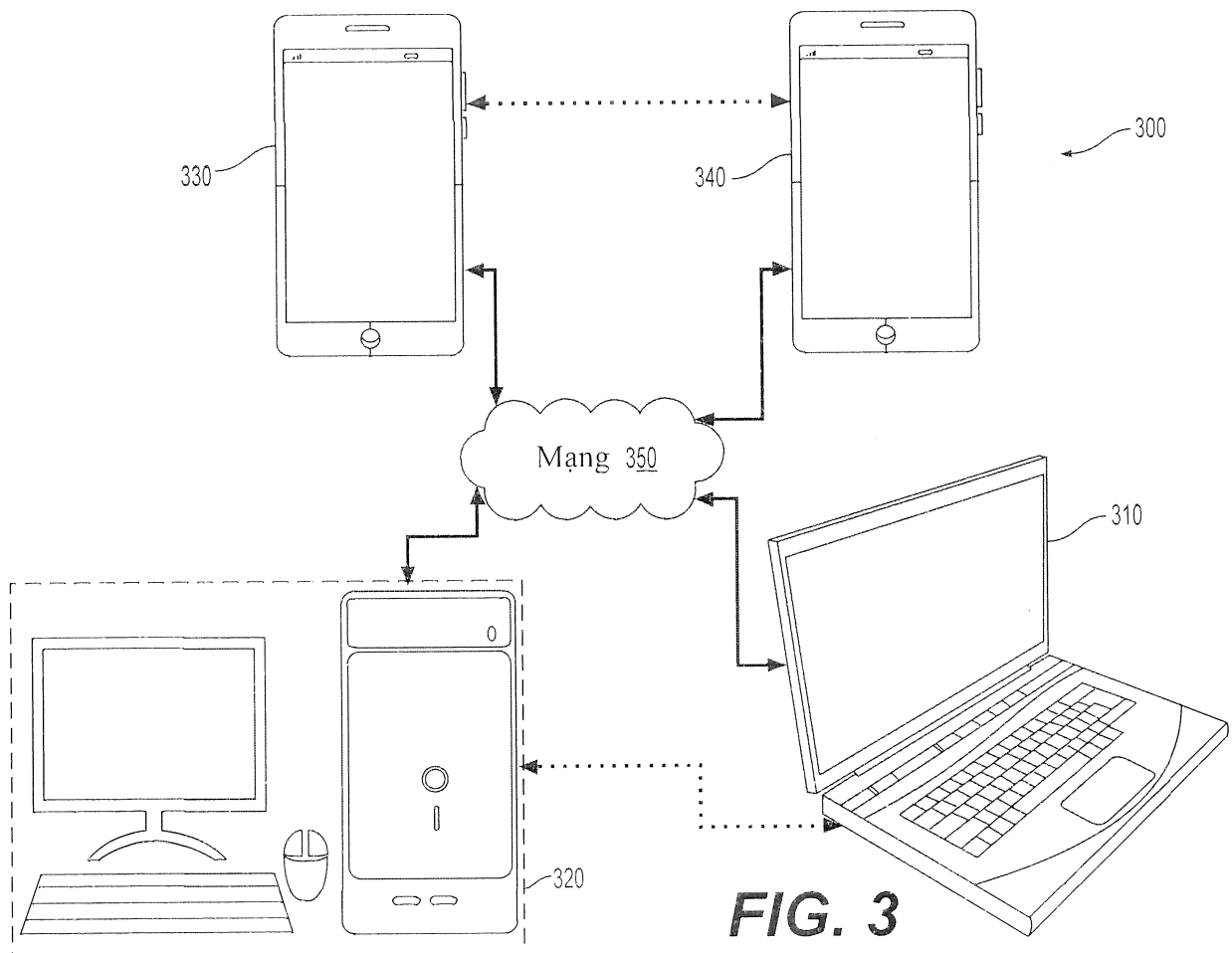
FIG. 1B

(Kỹ thuật đã biết)

0: chế độ phẳng
1: chế độ DC

**FIG. 2**

(Kỹ thuật đã biết)

**FIG. 3**

3/29

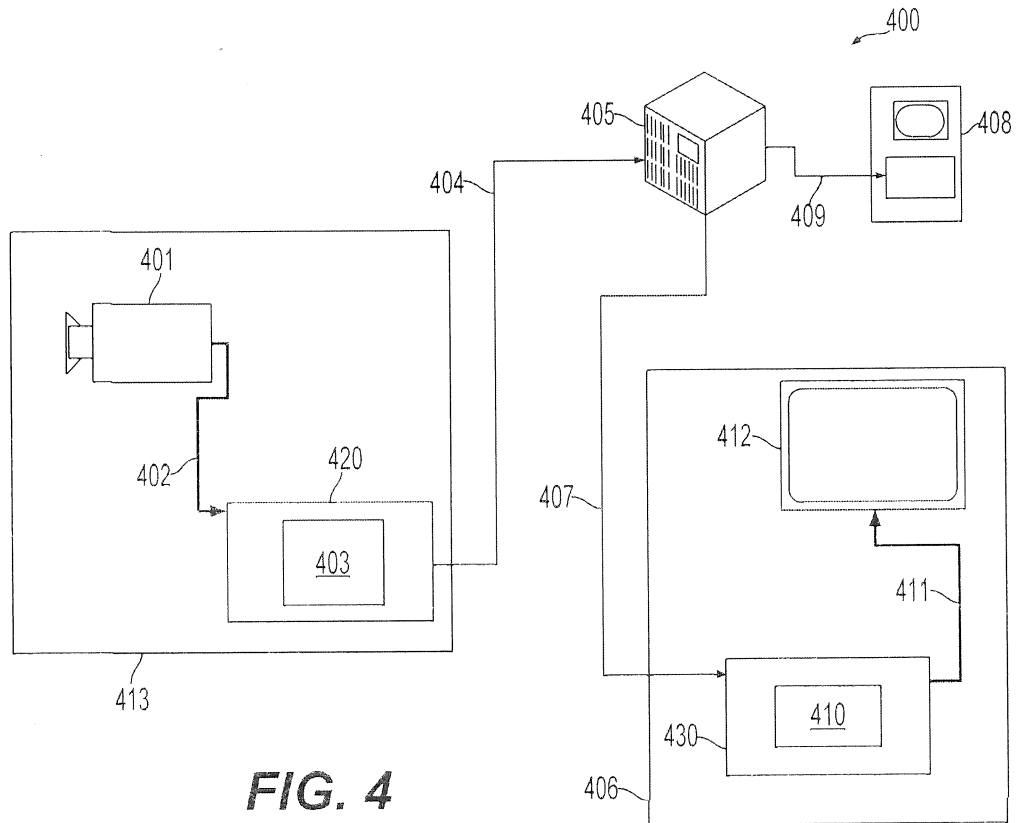


FIG. 4

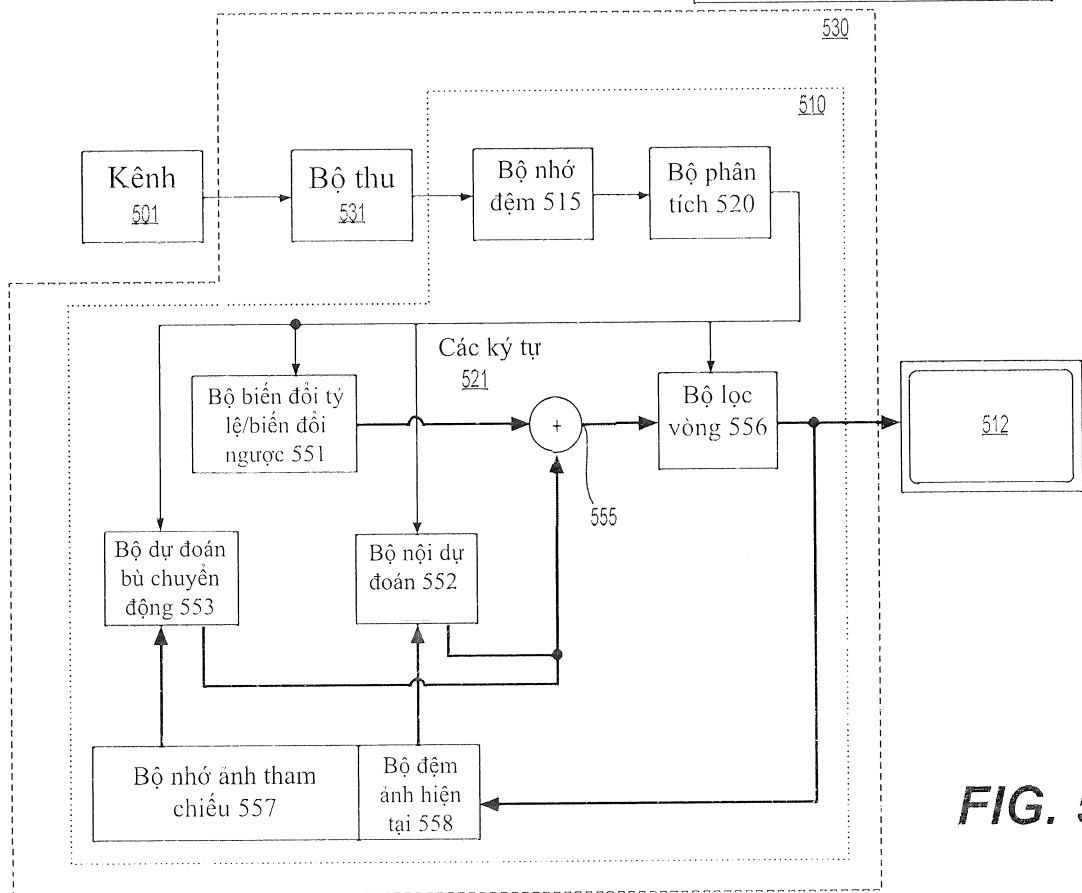


FIG. 5

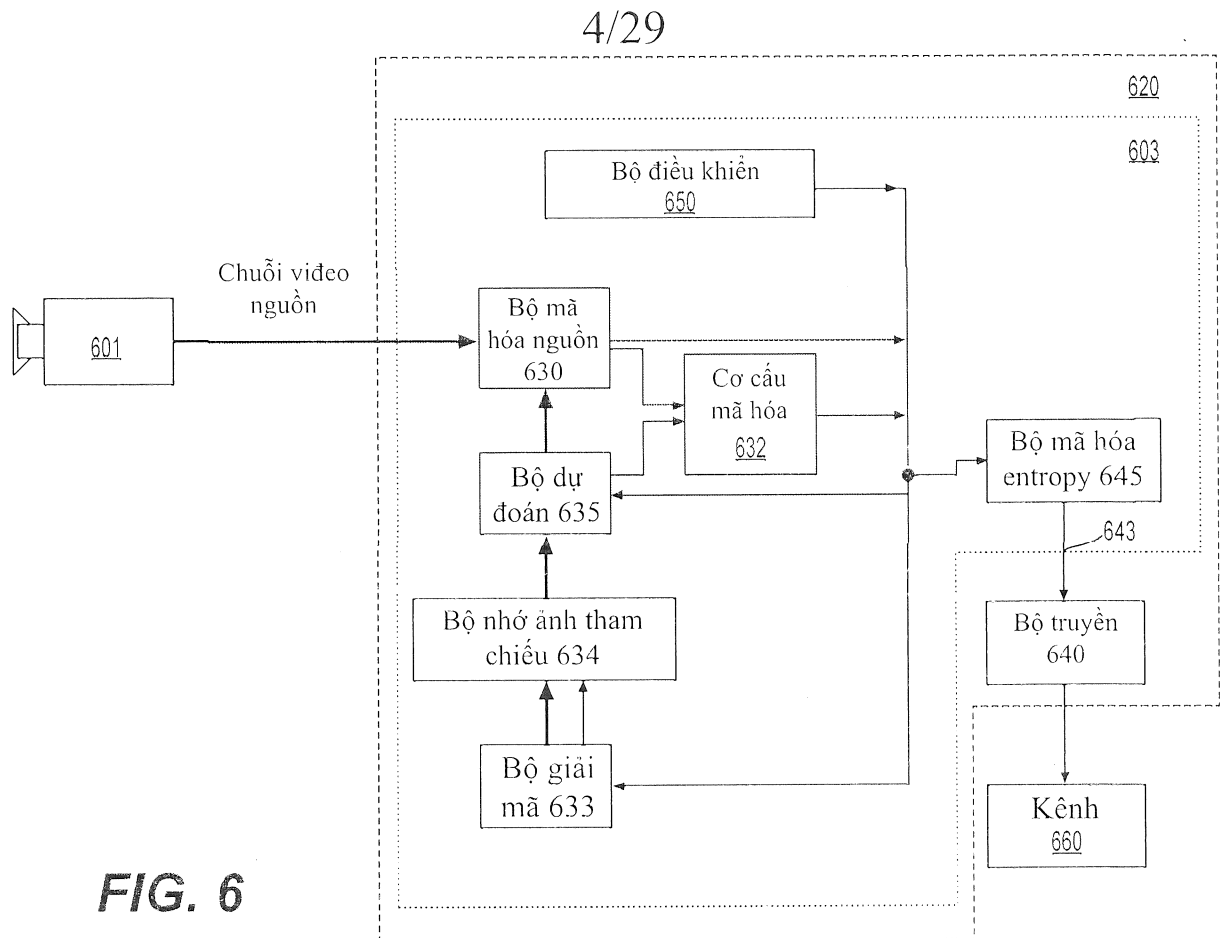


FIG. 6

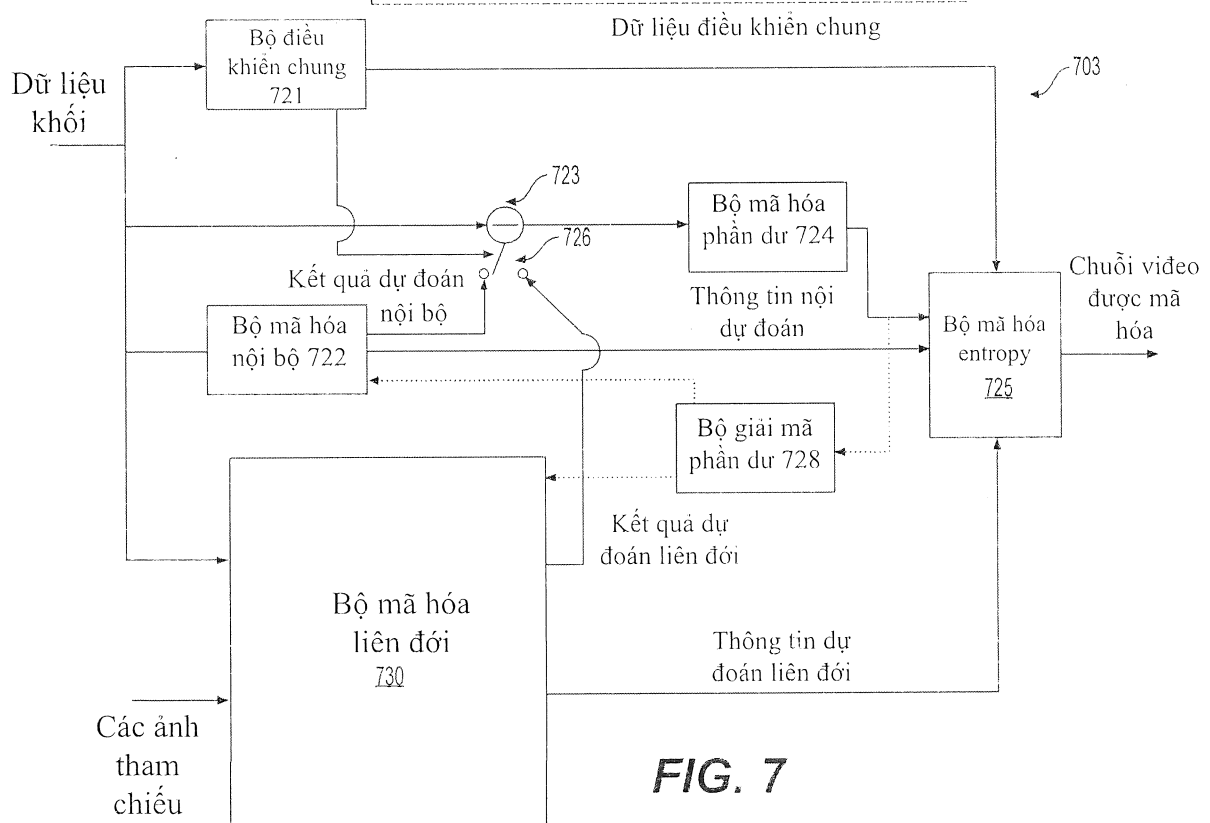
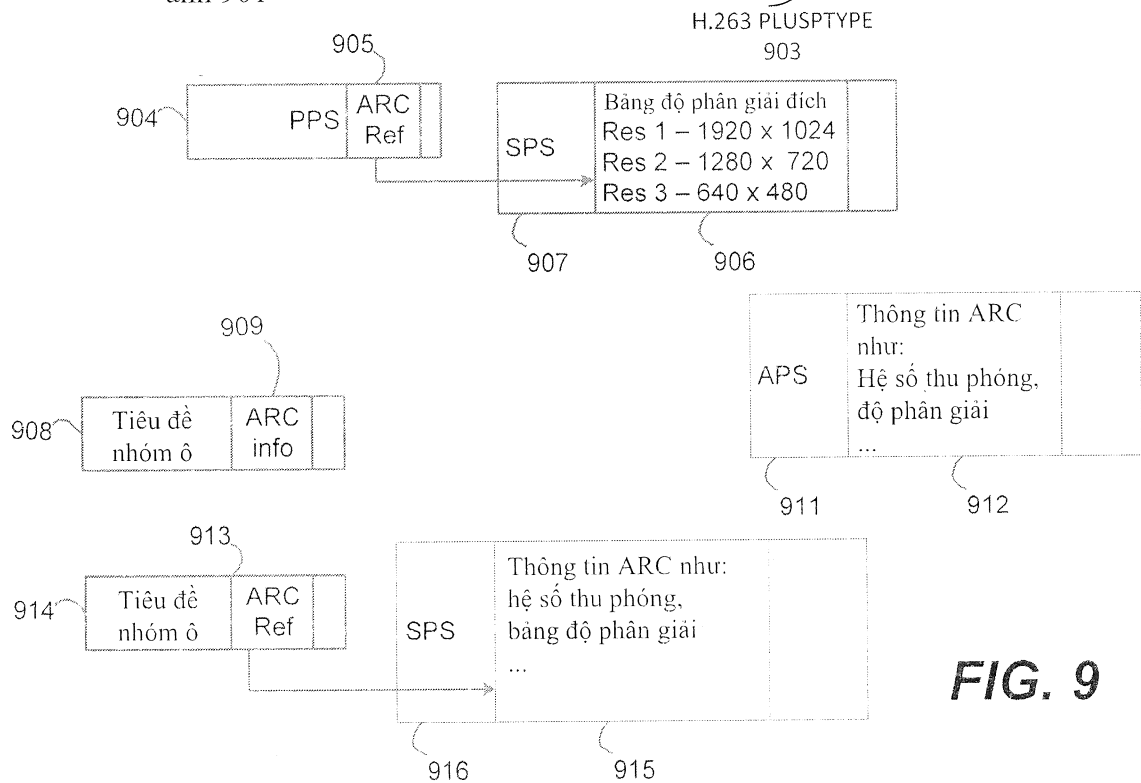
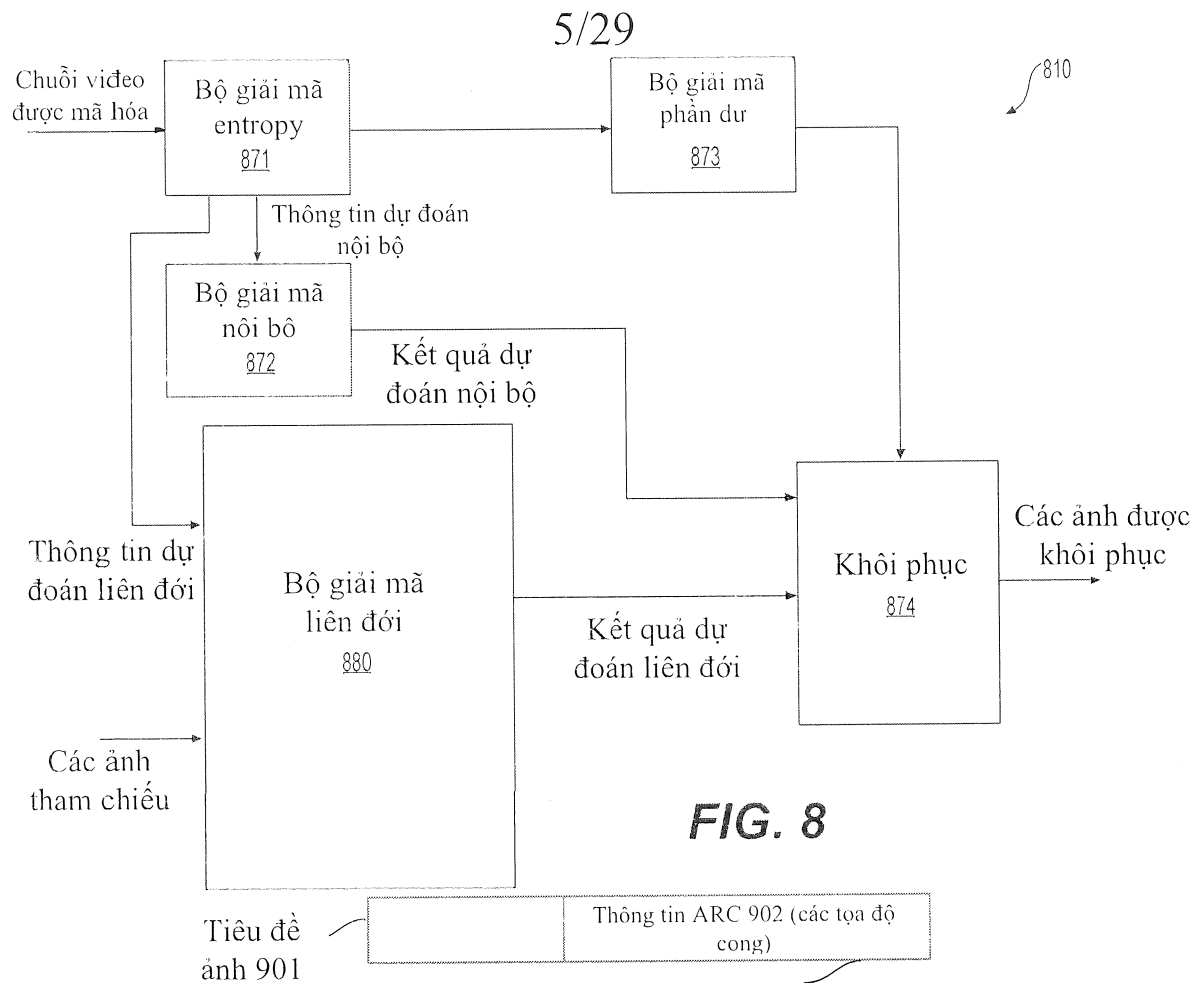


FIG. 7



6/29

1001	tile_group_header() {	
	...	
1003	if(adaptive_pic_resolution_change_flag) {	
1002	dec_pic_size_idx	u(1)
	}	
	...	
	}	

1010	seq_parameter_set_rbsp() {	Mô tả
	...	
1011	adaptive_pic_resolution_change_flag	u(1)
1012	if(adaptive_pic_resolution_change_flag) {	
1013 [	output_pic_width_in_luma_samples	ue(v)
	output_pic_height_in_luma_samples	ue(v)
1014	reference_pic_size_present_flag	u(1)
	if(reference_pic_size_present_flag)	
	{	
1015 [	reference_pic_width_in_luma_samples	ue(v)
	reference_pic_height_in_luma_samples	ue(v)
	}	
1016	num_dec_pic_size_in_luma_samples_minus1	ue(v)
	for(i = 0; i <=	
	num_dec_pic_size_in_luma_samples_minus1;	
	i++) {	
1017 [	dec_pic_width_in_luma_samples[i]	ue(v)
	dec_pic_height_in_luma_samples[i]	ue(v)
	}	
	}	
	...	
	}	

FIG. 10

7/29

profile tier level(profileTierPresentFlag, maxNumSubLayersMinus1) {	Mô tả
if(profileTierPresentFlag) {	
general profile idc	u(7)
general tier flag	u(1)
general constraint info()	
}	
general level idc	u(8)
if(profileTierPresentFlag) {	
num sub profiles	u(8)
for(i = 0; i < num sub profiles; i++)	
general sub profile idc[i]	u(32)
}	
for(i = 0; i < maxNumSubLayersMinus1; i++)	
sublayer level present flag[i]	u(1)
while(!byte aligned())	
ptl alignment zero bit	f(1)
for(i = 0; i < maxNumSubLayersMinus1; i++)	
if(sublayer level present flag[i])	
sublayer level idc[i]	u(8)
}	

FIG. 11

8/29

	general_constraint_info() {	Mô tả
	general_non_packed_constraint_flag	u(1)
	general_frame_only_constraint_flag	u(1)
	general_non_projected_constraint_flag	u(1)
	general_one_picture_only_constraint_flag	u(1)
1201	no_lossless_coding_tool_constraint_flag	u(1)
1202	no_lossy_coding_tool_constraint_flag	u(1)
1205	intra_only_constraint_flag	u(1)
	max_bitdepth_constraint_idc	u(4)
	max_chroma_format_constraint_idc	u(2)
	single_layer_constraint_flag	u(1)
	all_layers_independent_constraint_flag	u(1)
	no_ref_pic_resampling_constraint_flag	u(1)
	no_res_change_in_clvs_constraint_flag	u(1)
	one_tile_per_pic_constraint_flag	u(1)
	pic_header_in_slice_header_constraint_flag	u(1)
	one_slice_per_pic_constraint_flag	u(1)
	one_subpic_per_pic_constraint_flag	u(1)
	...	
	no_partition_constraints_override_constraint_flag	u(1)
	no_sao_constraint_flag	u(1)
1206	no_alf_constraint_flag	u(1)
	no_radl_constraint_flag	u(1)
	no_idr_constraint_flag	u(1)
	no_cra_constraint_flag	u(1)
	no_gdr_constraint_flag	u(1)
	no_aps_constraint_flag	u(1)
	while(!byte_aligned())	
	gci_alignment_zero_bit	f(1)
1203	gci_num_reserved_bytes	u(8)
	for(i = 0; i < gci_num_reserved_bytes; i++)	
1204	gci_reserved_byte[i]	u(8)
	}	

FIG. 12

9/29

1301	profile tier level(profileTierPresentFlag, maxNumSubLayersMinus1) {	Mô tả
	if(profileTierPresentFlag) {	
	general profile idc	u(7)
	general tier flag	u(1)
	num available constraint flags	u(8)
	general constraint info(num available constraint flags)	
	}	
	general level idc	u(8)
	if(profileTierPresentFlag) {	
	num sub profiles	u(8)
	for(i = 0; i < num sub profiles; i++)	
	general sub profile idc[i]	u(32)
	}	
	for(i = 0; i < maxNumSubLayersMinus1; i++)	
	sublayer level present flag[i]	u(1)
	while(!byte aligned())	
	ptl alignment zero bit	f(1)
	for(i = 0; i < maxNumSubLayersMinus1; i++)	
	if(sublayer level present flag[i])	
	sublayer level idc[i]	u(8)
	}	

FIG. 13A

tiếp tục từ FIG. 13A

1302	general constraint info(num available constraint flag	{	Mô tả
	count = 0		
	if(count < num available constraint flags)		
	general non packed constraint flag		u(1)
	count++		
	if(count < num available constraint flags)		
	general frame only constraint flag		u(1)
	count++		
	if(count < num available constraint flags)		
	general non projected constraint flag		u(1)
	count++		
	...		
	while(!byte aligned())		
	gci alignment zero bit		f(1)
	gci num reserved bytes		u(8)
	for(i = 0; i < gci num reserved bytes; i++)		
	gci reserved byte[i]		u(8)
	}		
	}		

FIG. 13B

10/29

1400

	general_constraint_info() {	Mô tả
1401	general_frame_structure_constraint_group_flag	u(1)
1402	high_level_functionality_constraint_group_flag	u(1)
1403	scalability_constraint_group_flag	u(1)
1404	partitioning_constraint_group_flag	u(1)
1405	intra_coding_tool_constraint_group_flag	u(1)
1406	inter_coding_tool_constraint_group_flag	u(1)
1407	transform_constraint_group_flag	u(1)
1408	inloop_filtering_constraint_group_flag	u(1)
	...	
	if(general_frame_structure_constraint_group_flag) {	
1411	general_non_packed_constraint_flag	u(1)
1412	general_frame_only_constraint_flag	u(1)
1413	general_non_projected_constraint_flag	u(1)
1414	general_one_picture_only_constraint_flag	u(1)
	}	
	}	

1410

FIG. 14A

1400

tiếp tục từ FIG. 14A

	if(high_level_functionality_constraint_group_flag) {	
1420	pic_header_in_slice_header_constraint_flag	u(1)
	no_rasl_constraint_flag	u(1)
	no_idr_constraint_flag	u(1)
	no_cra_constraint_flag	u(1)
	no_gdr_constraint_flag	u(1)
	no_aps_constraint_flag	u(1)
	...	
	}	
	if(intra_coding_tool_constraint_group_flag) {	
	...	
	}	
	if(inter_coding_tool_constraint_group_flag) {	
	...	
	}	
	while(!byte_aligned())	
1430	gci_alignment_zero_bit	f(1)
	gci_num_reserved_bytes	u(8)
	for(i = 0; i < gci_num_reserved_bytes; i++)	
	gci_reserved_byte[i]	u(8)
	}	

FIG. 14B

11/29

1500

	general_constraints_info() {	Mô tả
	gci_present_flag	u(1)
	if(gci_present_flag) {	
	/* general */	
	gci_intra_only_constraint_flag	u(1)
1510	gci_all_layers_independent_constraint_flag	u(1)
	gci_one_au_only_constraint_flag	u(1)
	/* picture format */	
1511	gci_sixteen_minus_max_bitdepth_constraint_idc	u(4)
	gci_three_minus_max_chroma_format_constraint_idc	u(2)
	/* NAL unit type related */	
	gci_no_mixed_nalu_types_in_pic_constraint_flag	u(1)
	gci_no_trail_constraint_flag	u(1)
	gci_no_stsa_constraint_flag	u(1)
	gci_no_rasl_constraint_flag	u(1)
1512	gci_no_radl_constraint_flag	u(1)
	gci_no_idr_constraint_flag	u(1)
	gci_no_cra_constraint_flag	u(1)
	gci_no_gdr_constraint_flag	u(1)
	gci_no_aps_constraint_flag	u(1)
	gci_no_idr_rpl_constraint_flag	u(1)

FIG. 15A

tiếp tục từ FIG. 15A

1500

	/* tile, slice, subpicture partitioning */	
	gci_one_tile_per_pic_constraint_flag	u(1)
	gci_pic_header_in_slice_header_constraint_flag	u(1)
1513	gci_one_slice_per_pic_constraint_flag	u(1)
	gci_no_rectangular_slice_constraint_flag	u(1)
	gci_one_slice_per_subpic_constraint_flag	u(1)
	gci_no_subpic_info_constraint_flag	u(1)
	/* CTU and block partitioning */	
	gci_three_minus_max_log2_ctu_size_constraint_idc	u(2)
1514	gci_no_partition_constraints_override_constraint_flag	u(1)
	gci_no_mtt_constraint_flag	
	gci_no_qtbtt_dual_tree_intra_constraint_flag	u(1)
	/* intra */	
	gci_no_palette_constraint_flag	u(1)
	gci_no_ibc_constraint_flag	u(1)
1515	gci_no_isp_constraint_flag	u(1)
	gci_no_mrl_constraint_flag	u(1)
	gci_no_mip_constraint_flag	u(1)
	gci_no_cclm_constraint_flag	u(1)

FIG. 15B

12/29

tiếp tục từ FIG. 15B

1500

/* inter */	
gei_no_ref_pic_resampling_constraint_flag	u(1)
1516 gei_no_res_change_in_clvs_constraint_flag	u(1)
gei_no_weighted_prediction_constraint_flag	u(1)
gei_no_ref_wraparound_constraint_flag	u(1)
gei_no_temporal_mvp_constraint_flag	u(1)
gei_no_sbtmvp_constraint_flag	u(1)
gei_no_amvr_constraint_flag	u(1)
gei_no_bdof_constraint_flag	u(1)
gei_no_smvd_constraint_flag	u(1)
gei_no_dmvr_constraint_flag	u(1)
gei_no_mmvd_constraint_flag	u(1)
gei_no_affine_motion_constraint_flag	u(1)
gei_no_prof_constraint_flag	u(1)
gei_no_bcw_constraint_flag	u(1)
gei_no_ciip_constraint_flag	u(1)
gei_no_gpm_constraint_flag	u(1)

FIG. 15C

1500

1517

tiếp tục từ
FIG. 15C

/* transform, quantization, residual */	
gei_no_luma_transform_size_64_constraint_flag	u(1)
gei_no_transform_skip_constraint_flag	u(1)
gei_no_bdpcm_constraint_flag	u(1)
gei_no_mts_constraint_flag	u(1)
gei_no_lfnst_constraint_flag	u(1)
gei_no_joint_cbr_constraint_flag	u(1)
gei_no_sbt_constraint_flag	u(1)
gei_no_act_constraint_flag	u(1)
gei_no_explicit_scaling_list_constraint_flag	u(1)
gei_no_dep_quant_constraint_flag	u(1)
gei_no_sign_data_hiding_constraint_flag	u(1)
gei_no_cu_qp_delta_constraint_flag	u(1)
gei_no_chroma_qp_offset_constraint_flag	u(1)
/* loop filter */	
gei_no_sao_constraint_flag	u(1)
gei_no_alf_constraint_flag	u(1)
gei_no_ccalf_constraint_flag	u(1)
gei_no_lmes_constraint_flag	u(1)
gei_no_ladf_constraint_flag	u(1)
gei_no_virtual_boundaries_constraint_flag	u(1)
gei_num_reserved_bits	u(8)
for(i = 0; i < gei_num_reserved_bits; i++)	
gei_reserved_zero_bit[i]	u(1)
{	
while(!byte_aligned())	
gei_alignment_zero_bit	f(1)
}	

1518

FIG. 15D

13/29

profile_tier_level(profileTierPresentFlag, MaxNumSubLayersMinus1) {	Mô tả
if(profileTierPresentFlag) {	
general_profile_idc	u(7)
general_tier_flag	u(1)
}	
general_level_idc	u(8)
ptl_frame_only_constraint_flag	u(1)
ptl_multilayer_enabled_flag	u(1)
if(profileTierPresentFlag)	
general_constraints_info()	
for(i = MaxNumSubLayersMinus1 - 1; i >= 0; i--)	
ptl_sublayer_level_present_flag[i]	u(1)
while(!byte_aligned())	
ptl_reserved_zero_bit	u(1)
for(i = MaxNumSubLayersMinus1 - 1; i >= 0; i--)	
if(ptl_sublayer_level_present_flag[i])	
sublayer_level_idc[i]	u(8)
if(profileTierPresentFlag) {	
ptl_num_sub_profiles	u(8)
for(i = 0; i < ptl_num_sub_profiles; i++)	
general_sub_profile_idc[i]	u(32)
}	
}	

FIG. 16

FIG. 17

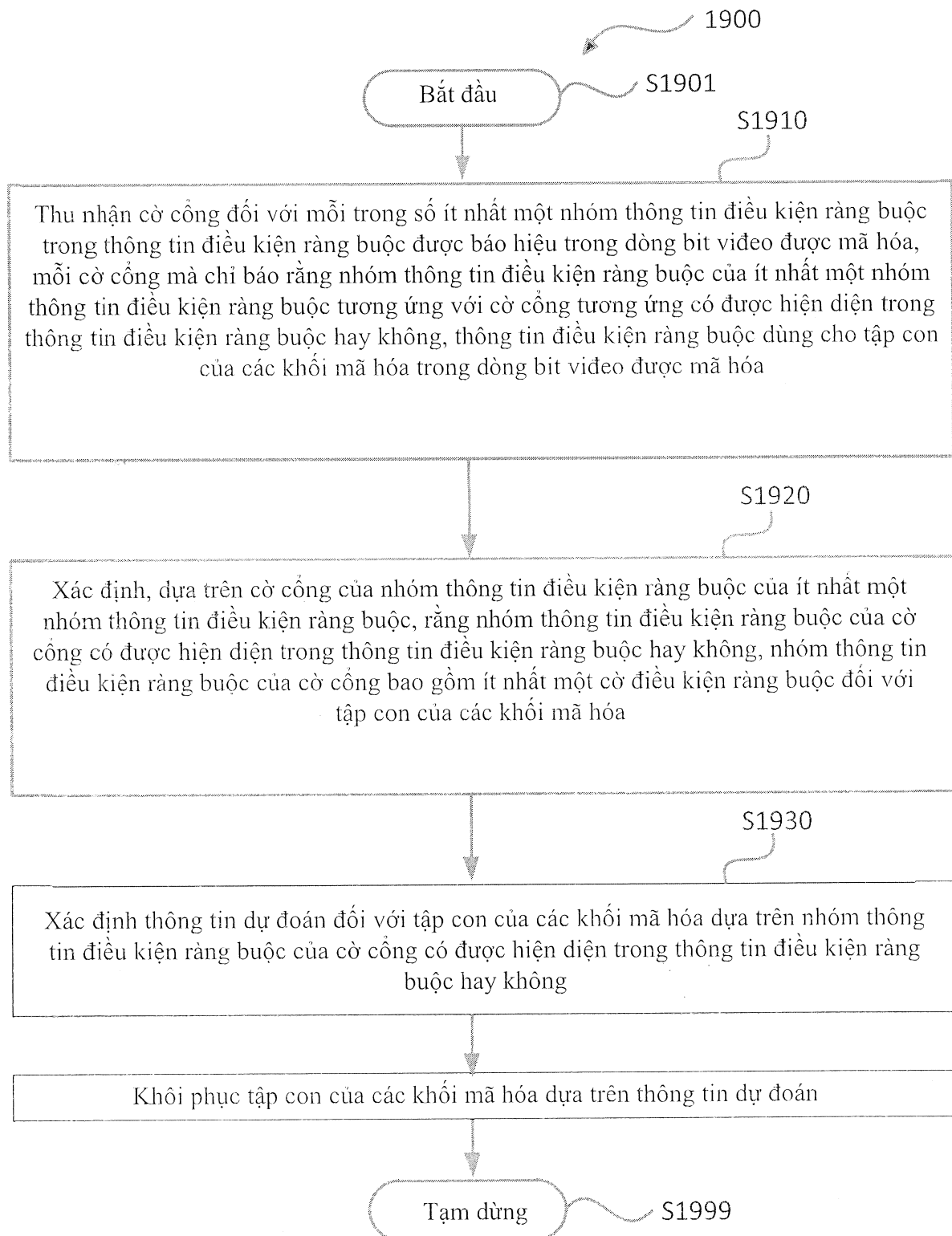
seq_parameter_set_rbsp() {	Mô tả
sps_seq_parameter_set_id	u(4)
...	
sps_field_seq_flag	u(1)
...	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

14/29

FIG. 18

video_parameter_set_rbsp() {	Mô tả
...	
vps_field_seq_flag	u(1)
...	
}	

15/29

**FIG. 19**

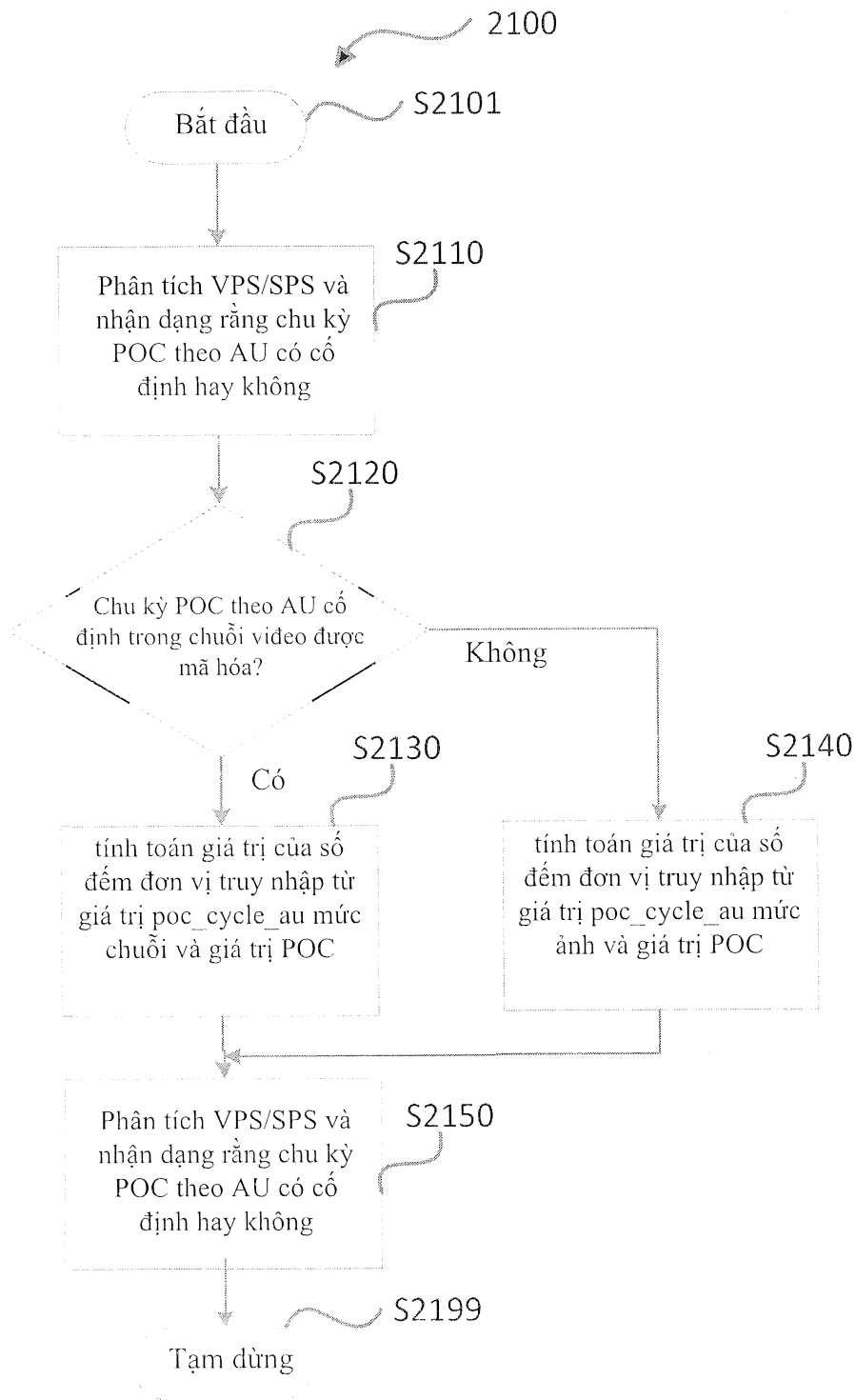
16/29

	Mô tả
video_parameter_set_rbsp() {	
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(8)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_included_layer_id[i]	u(7)
vps_reserved_zero_bit	u(1)
}	
vps_constraint_info_present_flag	u(1)
vps_constant_poc_cycle_per_au	u(1)
if(vps_constant_poc_cycle_per_au)	
vps_poc_cycle_au	u(8)
}	
}	

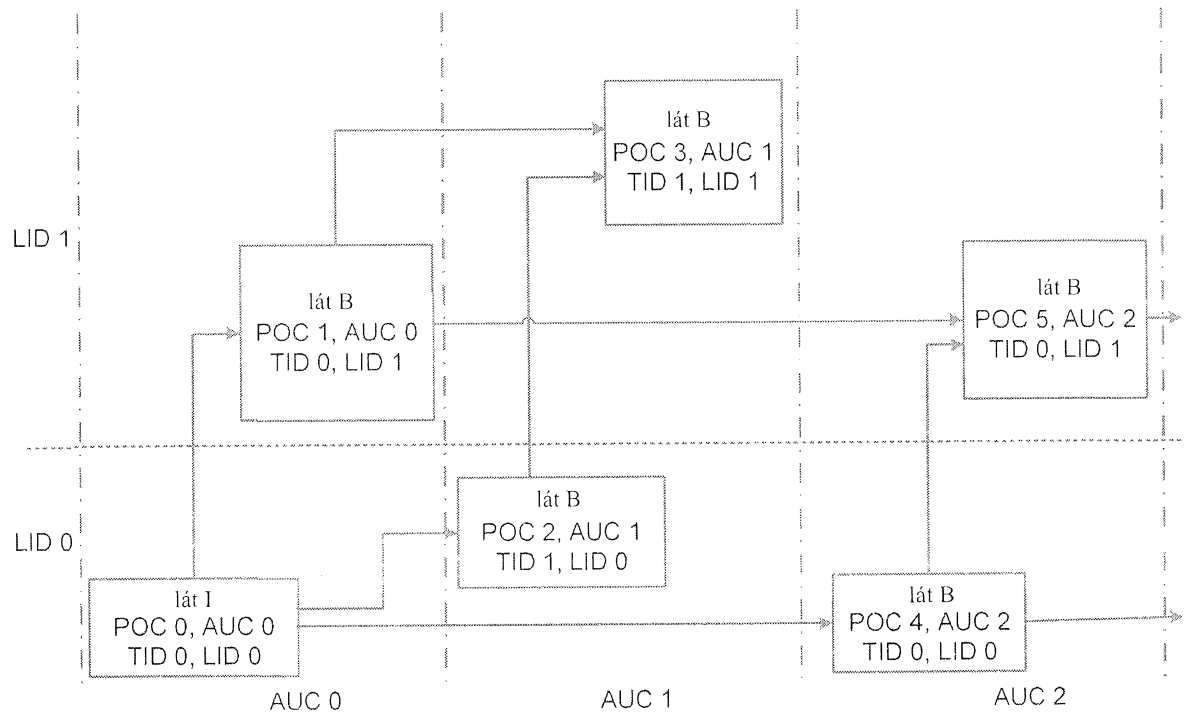
	Mô tả
slice_header() {	
slice_pic_parameter_set_id	ue(4)
if(rect_slice_flag & NumBricksInPic > 1) {	
slice_address	u(1)
if(!rect_slice_flag && !single_brick_per_slice_flag) {	
num_bricks_in_slice_minus1	ue(4)
slice_type	ue(4)
if(NalUnitType == GRA_NUT) {	
recovery_poc_cnt	se(4)
slice_pic_order_cnt_lsb	u(4)
}	
if(!vps_constant_poc_cycle_per_au)	
slice_poc_cycle_au	u(8)
}	
}	

FIG. 20

17/29

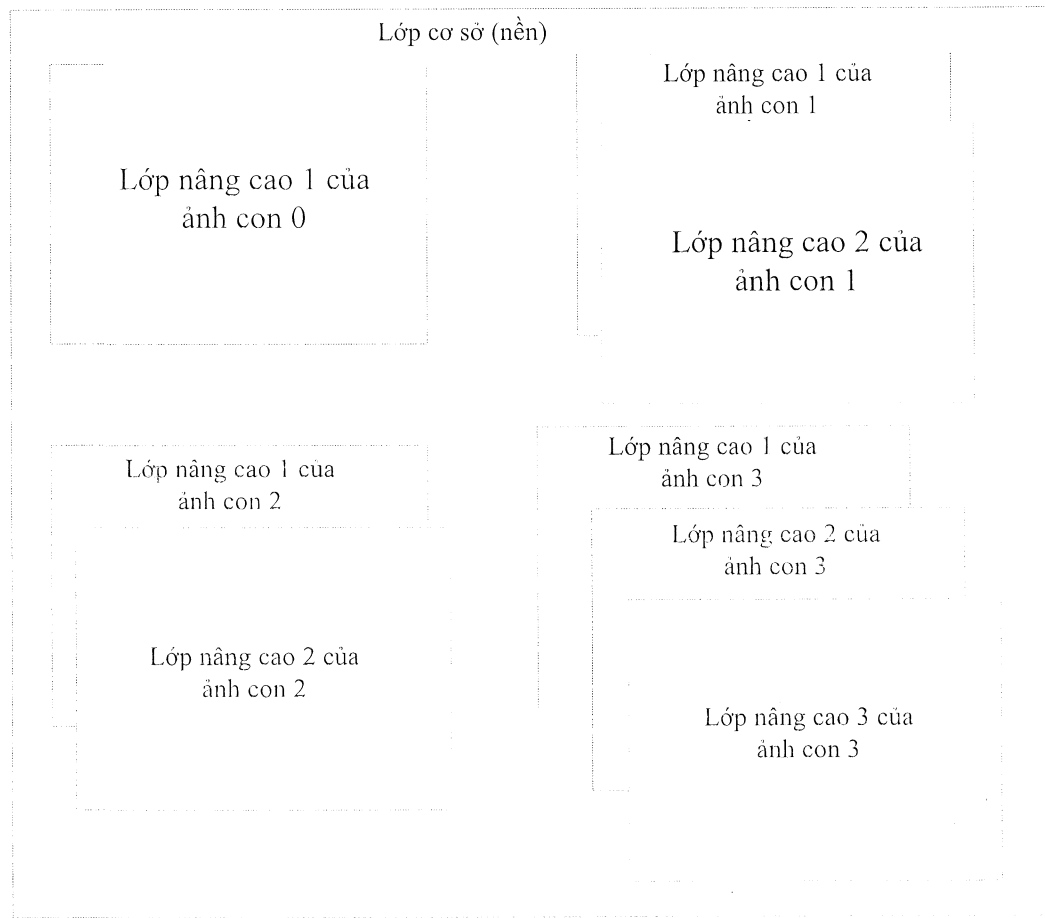
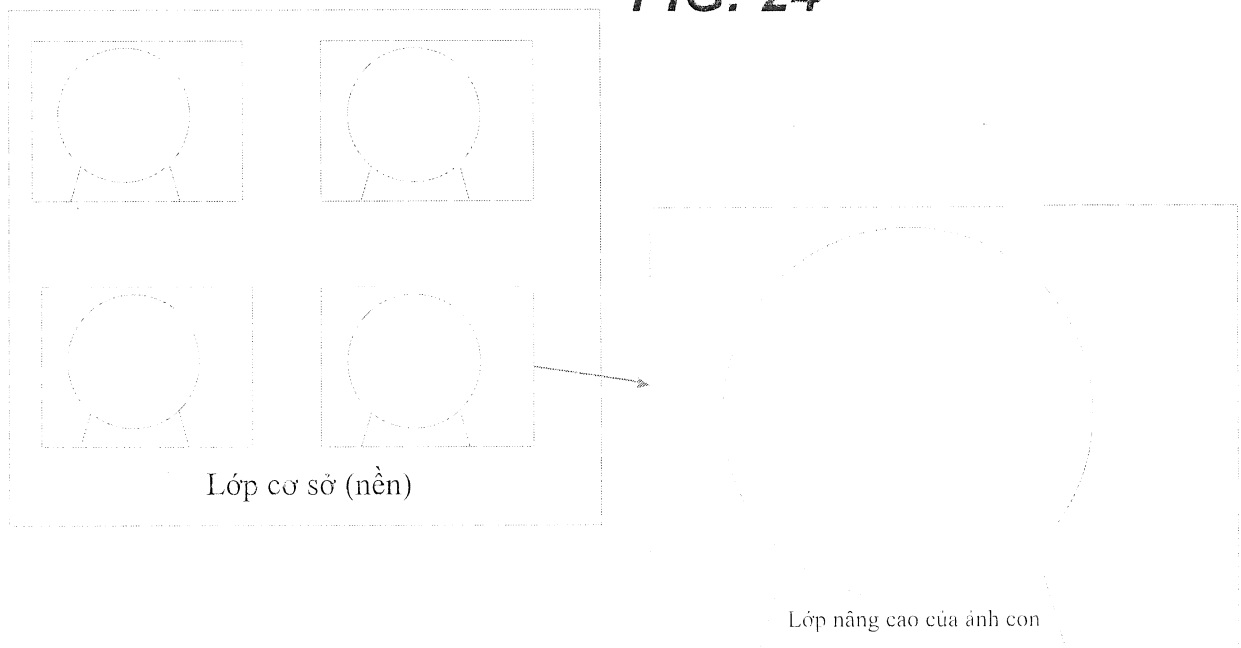
**FIG. 21**

18/29

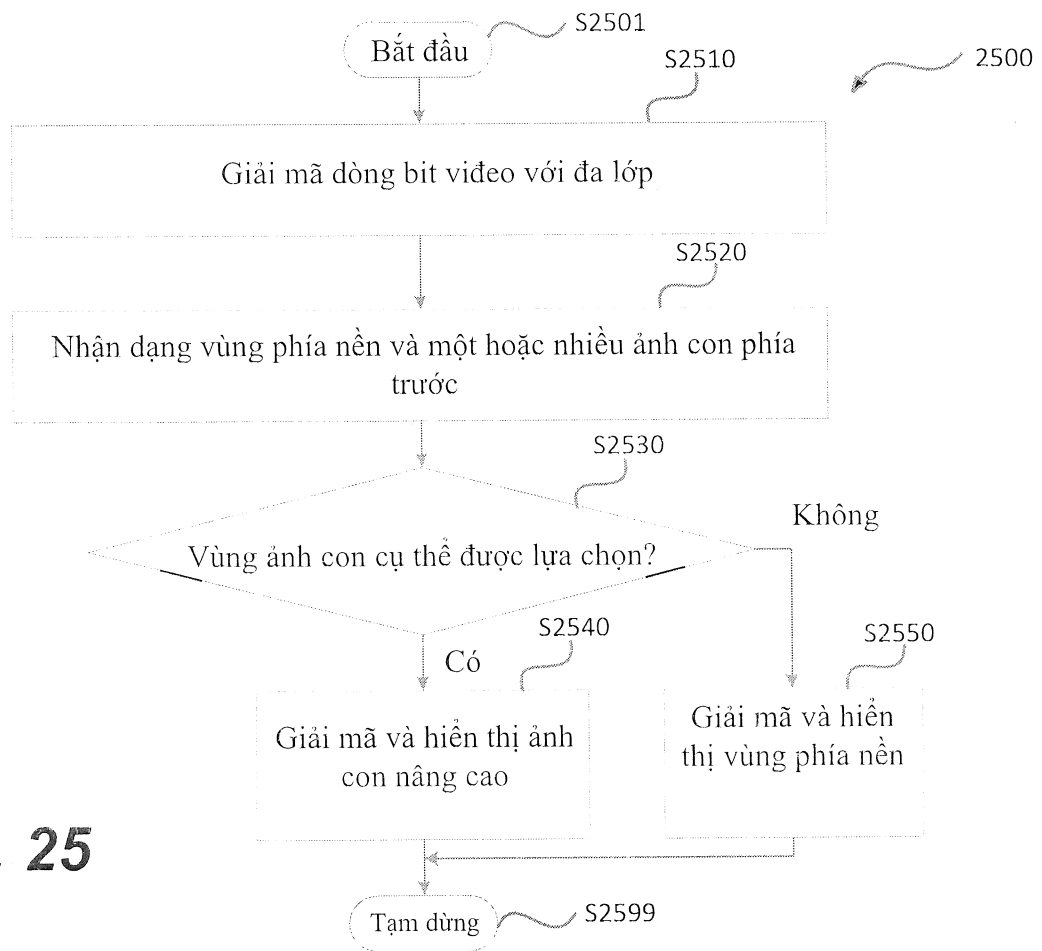
**FIG. 22**

POC: số đếm thứ tự ảnh
AUC: số đếm đơn vị truy nhập
TID: ID thời gian
LID: ID lớp

19/29

**FIG. 23****FIG. 24**

20/29

**FIG. 25**

21/29

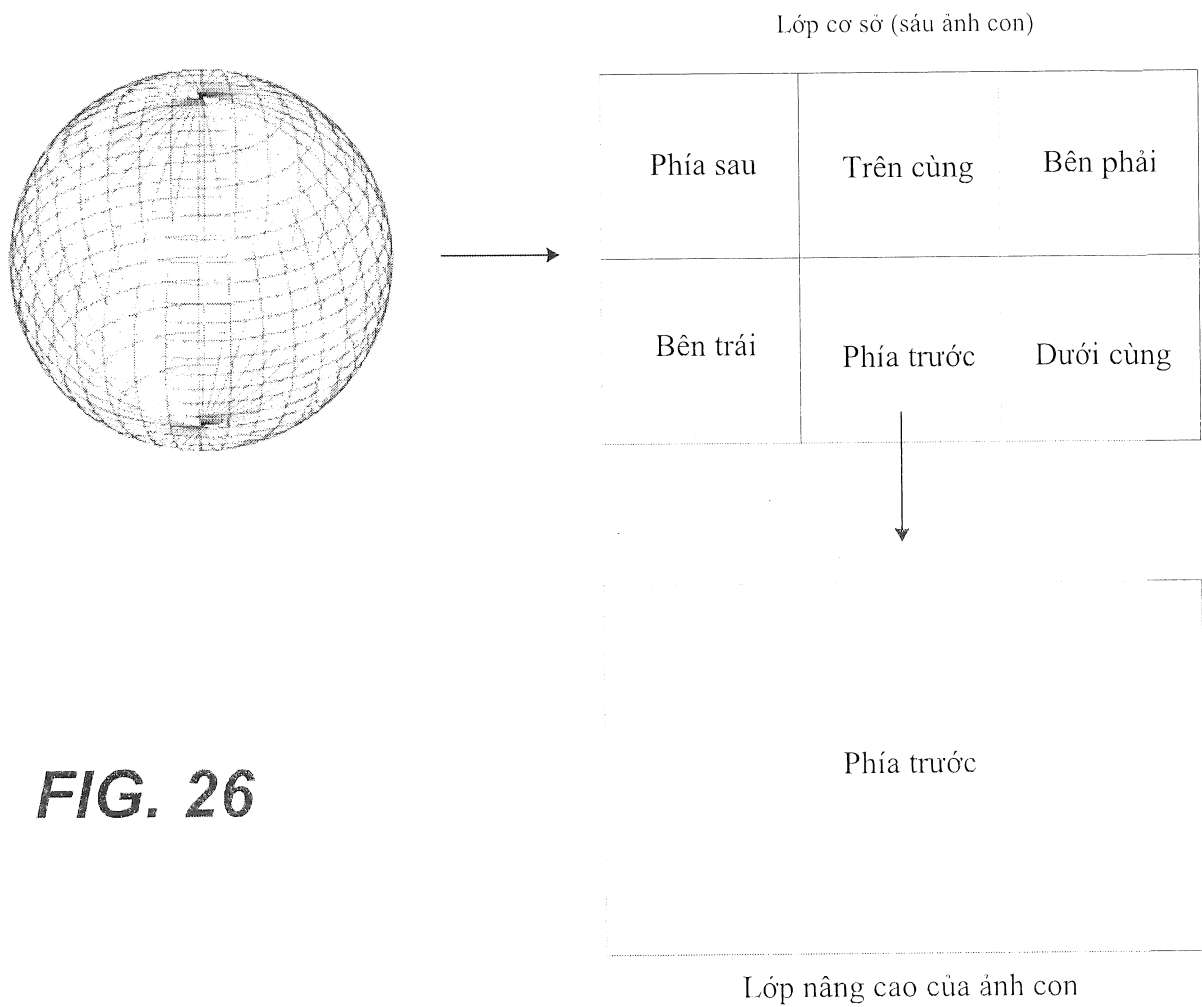
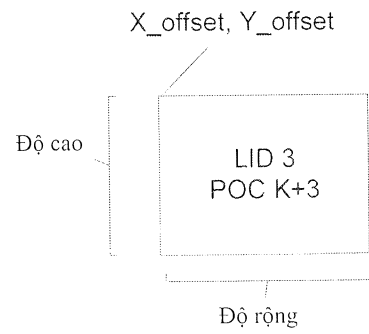
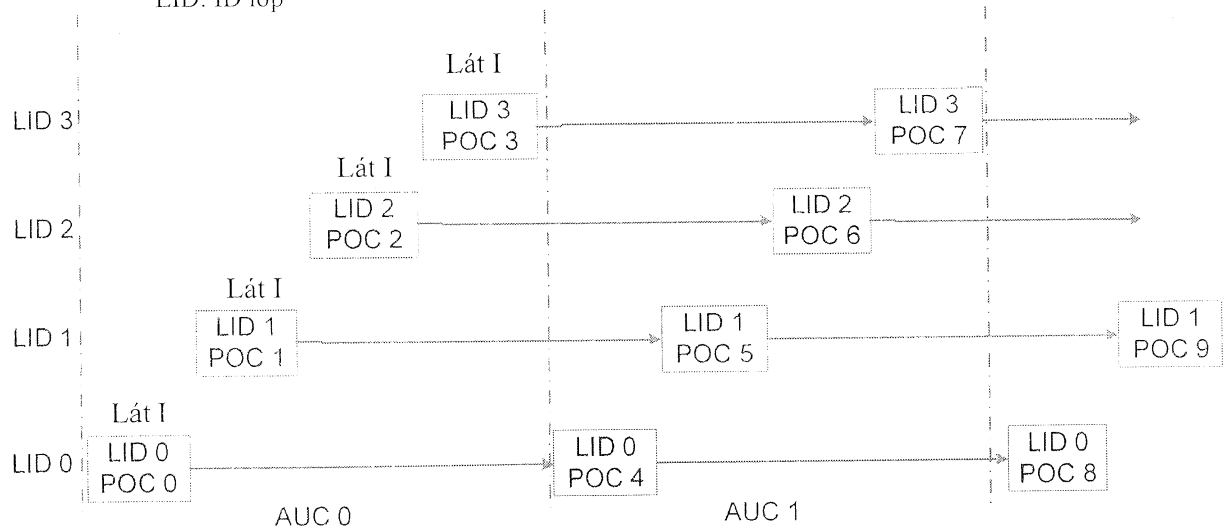


FIG. 26

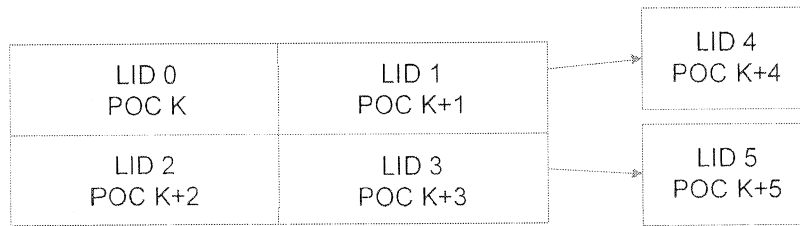
22/29

LID 0 POC K	LID 1 POC K+1
LID 2 POC K+2	LID 3 POC K+3

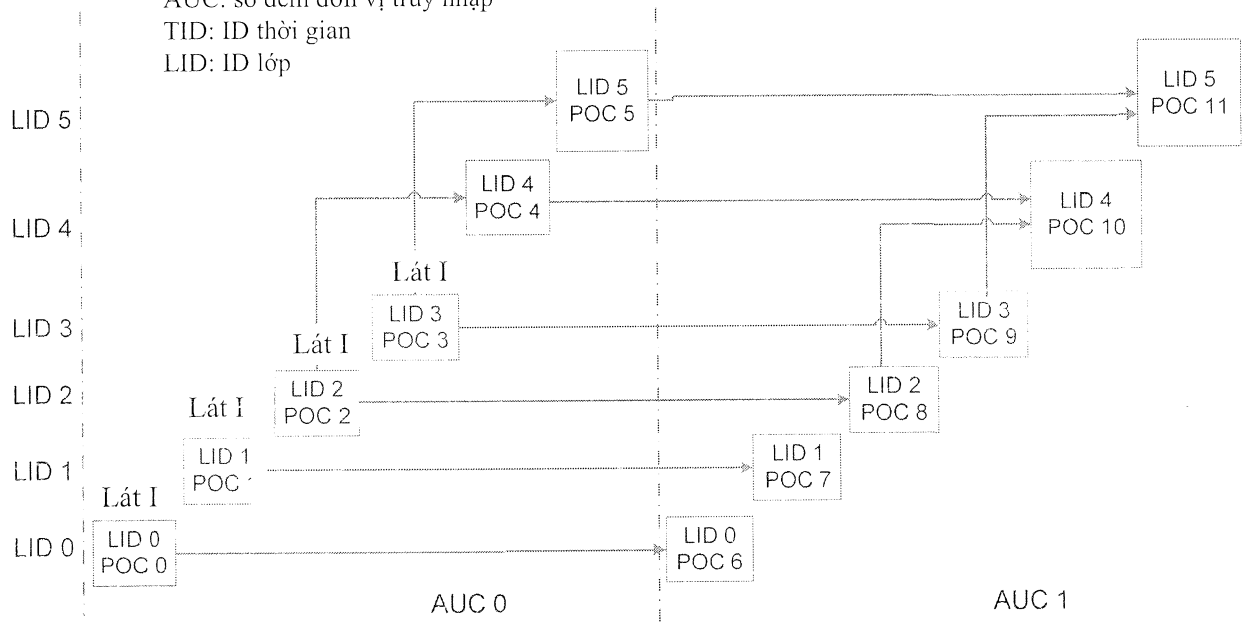
POC: số đếm thứ tự ảnh
AUC: số đếm đơn vị truy nhập
TID: ID thời gian
LID: ID lớp

**FIG. 27**

23/29

**FIG. 28**

POC: số đếm thứ tự ảnh
 AUC: số đếm đơn vị truy nhập
 TID: ID thời gian
 LID: ID lớp



video_parameter_set_rbsp() {	Mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(8)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_included_layer_id[i]	u(7)
vps_reserved_zero_bit	u(1)
}	
vps_sub_picture_dividing_flag	u(1)
if(vps_sub_picture_dividing_flag) {	
vps_full_pic_width_in_luma_samples	ue(v)
vps_full_pic_height_in_luma_samples	ue(v)
}	
}	

FIG. 29A

24/29
tiếp tục từ FIG. 29A

* seq_parameter_set_rbsp() {	Mô tả
sps_decoding_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
sps_max_sub_layers_minus1	u(3)
...	
pic_width_in_luma_samples	ue(v)
pic_height_in_luma_samples	ue(v)
if(vps_sub_picture_dividing_flag) {	
pic_offset_x	ue(v)
pic_offset_y	ue(v)
}	
...	
}	

FIG. 29B

sub_region_partitioning_info(payloadSize) {	Mô tả
num_sub_region	ue(v)
num_layers	ue(v)
for(i = 0; i <= num_layers; i++)	
layer_id [i]	ue(v)
for(i = 0; i < num_layers; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
for(i = 0; i < num_sub_region; i++) {	
num_layers_for_region[i]	ue(v)
for(j = 0; j < num_layers_for_region[i]; j++)	
sub_region_layer_id[i][j]	ue(v)
sub_region_offset_x[i]	ue(v)
sub_region_offset_y[i]	ue(v)
sub_region_width[i]	ue(v)
sub_region_height[i]	ue(v)
}	
...	
}	

FIG. 30

25/29

video parameter set rbsp() {	Mô tả
...	
vps_max_layers_minus1	u(6)
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++)	
for(j = 0; j < NumLayersInIdList[i]; j++) {	
output_layer_flag[i][j]	u(1)
profile_tier_level_idx[i][j]	u(v)
}	
...	
}	

FIG. 31

video parameter set rbsp() {	Mô tả
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
{	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInIdList[i]; j++) {	
if(vps_output_layers_mode[i] == 2)	
output_layer_flag[i][j]	u(1)
if(vps_ptl_signal_flag[i])	
profile_tier_level_idx[i][j]	u(v)
}	
}	
}	
...	
}	

FIG. 32

26/29

FIG. 33

video parameter set rbsp() {	Mô tả
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0) {	
num_output_layer_sets	ue(v)
num_profile_tier_level	ue(v)
}	
max_subpics_minus1	u(8)
for(i = 0; i < max_subpics_minus1; i++) {	
sub_pic_id[i]	u(8)
}	
for(i = 0; i < num_profile_tier_level; i++)	
profile_tier_level(vps_max_sub_layers_minus1)	
for(i = 0; i < num_output_layer_sets; i++) {	
vps_output_layers_mode[i]	u(2)
vps_ptl_signal_flag[i]	u(1)
for(j = 0; j < NumLayersInIdList[i]; j++) {	
num_output_subpic_layer[i][j]	ue(v)
for(k = 0; k < num_output_subpic_layer[i][j]; k++)	
sub_pic_id_layer[i][j][k]	u(8)
if(vps_output_layers_mode[i] == 2)	
output_layer_flag[i][j]	u(1)
if(vps_ptl_signal_flag[i])	
profile_tier_level_idx[i][j]	u(v)
}	
}	
...	
}	

27/29

FIG. 34

video_parameter_set_rbsp() {	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)
}	
}	
}	
}	
}	
}	

28/29

FIG. 35

video parameter set rbsp() {	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 && vps max layers minus1 > 1) {	
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)
}	
}	
}	
}	
...	
}	

