



(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
1-0051185
(51)^{2021.01} H04N 19/105; H04N 19/50; H04N 19/573; H04N 19/159 (13) B

(21) 1-2022-05305 (22) 22/01/2021
(86) PCT/RU2021/050017 22/01/2021 (87) WO2021/054869 25/03/2021
(30) PCT/RU2020/000027 23/01/2020 RU; PCT/RU2020/000035 27/01/2020 RU
(45) 25/09/2025 450 (43) 25/10/2022 415A
(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)
Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, P. R. China
(72) CHERNYAK, Roman Igorevich (RU); SOLOVYEV, Timofey Mikhailovich (RU);
IKONIN, Sergey Yurievich (RU); KARABUTOV, Alexander Alexandrovich (RU);
ALSHINA, Elena Alexandrovna (RU).
(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP GIẢI MÃ, PHƯƠNG PHÁP MÃ HÓA, BỘ GIẢI MÃ VÀ BỘ MÃ HÓA

(21) 1-2022-05305

(57) Sáng chế đề cập đến các kỹ thuật để thực hiện nhiều hơn một sơ đồ quản lý ảnh tham chiếu được cung cấp trong ngữ cảnh của lập mã video. Cụ thể, sáng chế đề cập đến phương pháp giải mã chuỗi video được mã hóa, phương pháp mã hóa chuỗi video, bộ mã hóa, bộ giải mã, thiết bị mã hóa, thiết bị giải mã và phương tiện đọc được bởi máy tính không tạm thời, trong đó phương pháp giải mã chuỗi video được mã hóa, bao gồm các bước thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu; và xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu.

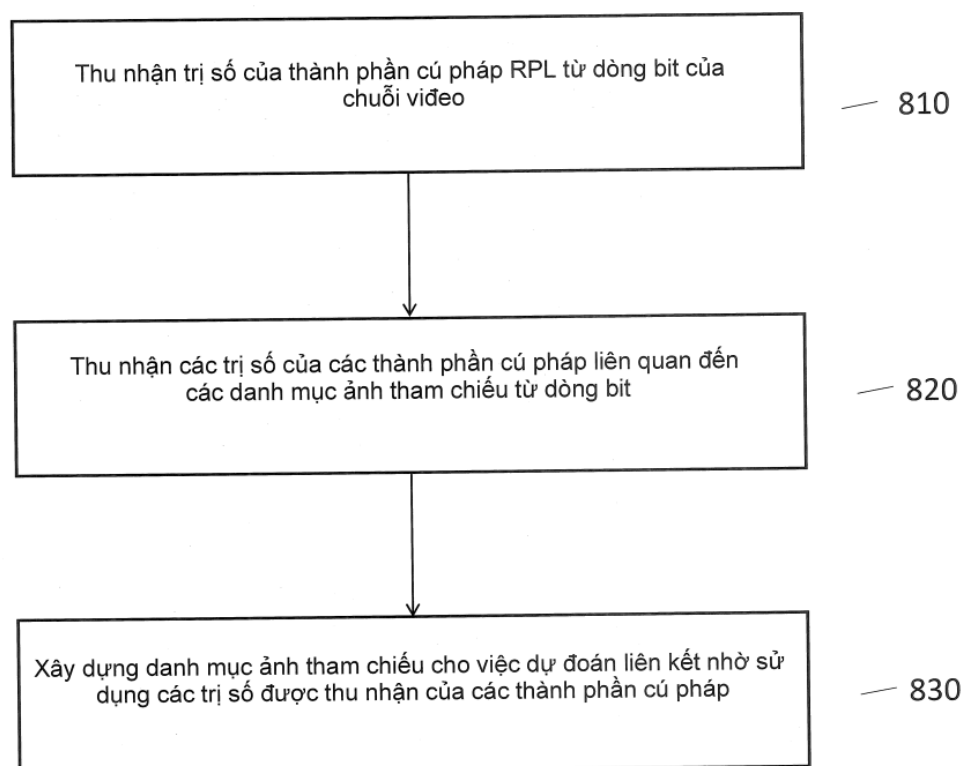


Fig. 8

Lĩnh vực kỹ thuật được đề cập

Các phương án của sáng chế nói chung đề cập đến lĩnh vực xử lý ảnh và cụ thể hơn là đến lập mã video.

Tình trạng kỹ thuật của sáng chế

Lập mã video (mã hóa và giải mã video) được sử dụng trong phạm vi rộng của các ứng dụng video số, ví dụ truyền hình (television, viết tắt là TV) số phát rộng, truyền video qua Internet và các mạng di động, các ứng dụng đàm thoại thời gian thực chẳng hạn như trò chuyện video, hội nghị video, đĩa video số (digital video disc, viết tắt là DVD) và các đĩa Blu-ray, các hệ thống thu nhận và biên tập nội dung video, và các máy quay phim kết hợp (camcorder) của các ứng dụng an ninh.

Lượng dữ liệu video được cần để mô tả ngay cả video tương đối ngắn có thể là đáng kể, điều này có thể dẫn đến các khó khăn khi dữ liệu cần được tạo dòng hoặc mặt khác được truyền thông qua mạng truyền thông với dung lượng độ rộng dải giới hạn. Do đó, dữ liệu video thường được nén trước khi được truyền thông qua các mạng viễn thông hiện nay. Kích thước của video cũng có thể là vấn đề khi video được lưu trữ trên thiết bị lưu trữ bởi vì các tài nguyên bộ nhớ có thể bị giới hạn. Các thiết bị nén video thường sử dụng phần mềm và/hoặc phần cứng ở nguồn để lập mã dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó làm giảm số lượng dữ liệu được cần để biểu diễn các hình ảnh video số. Dữ liệu được nén sau đó được thu ở đích bởi thiết bị giải nén video mà giải mã dữ liệu video. Với các tài nguyên mạng giới hạn và các nhu cầu ngày càng tăng về chất lượng video cao hơn, các kỹ thuật nén và giải nén nâng cao mà nâng cao tỷ số nén với chất lượng ảnh không giảm hoặc giảm không đáng kể là điều mong muốn.

Cụ thể là, cách thực hiện hiệu quả của các sơ đồ quản lý ảnh tham chiếu trong ngữ cảnh của việc xử lý dự đoán liên kết vẫn đặt ra vấn đề khó khăn xét về việc xử lý hiệu quả của và việc lựa chọn từ các sơ đồ khác nhau mà, về nguyên lý, có thể là sẵn có.

Bản chất kỹ thuật của sáng chế

Các phương án của sáng chế đề xuất các máy và các phương pháp mã hóa và giải

mã theo các điểm yêu cầu bảo hộ độc lập.

Các mục đích nêu trên và các mục đích khác đạt được bởi đối tượng của các điểm yêu cầu bảo hộ độc lập. Các cách thức thực hiện thêm khác là rõ ràng từ các điểm yêu cầu bảo hộ phụ thuộc, phần mô tả và các hình vẽ.

Các phương án cụ thể được nêu trong các điểm yêu cầu bảo hộ độc lập đính kèm, với các phương án khác trong các điểm yêu cầu bảo hộ phụ thuộc.

Theo một khía cạnh, sáng chế đề xuất phương pháp giải mã chuỗi video được mã hóa, phương pháp này bao gồm các bước

thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) (ví dụ, `sps_rpl_flag`) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không;

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit: a) thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu và b)

xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu.

Cụ thể, sáng chế đề xuất phương pháp giải mã chuỗi video được mã hóa, phương pháp này bao gồm các bước

thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không; và

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu.

Trị số của thành phần cú pháp RPL có thể được thu nhận bằng cách phân tách dòng bit. Khác với kỹ thuật đã biết, thành phần cú pháp (thành phần cú pháp RPL) miễn là

cho phép phát hiện (ví dụ, bởi thiết bị giải mã) xem cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không và, nhờ đó, xem phương pháp các danh mục ảnh tham chiếu có thể được sử dụng cho quy trình dự đoán liên kết hay không. Nhờ đó, việc quản lý của quy trình dự đoán liên kết, cụ thể là, việc quản lý ảnh tham chiếu có thể được điều khiển một cách hữu hiệu.

Theo cách thực hiện, phương pháp còn bao gồm bước, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit. Sự xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit có thể bao gồm thực hiện phương pháp cửa sổ trượt để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

Nhờ đó, việc quản lý danh mục ảnh tham chiếu mà không được dựa vào danh mục ảnh tham chiếu có thể được thực hiện. Trong trường hợp này, phương pháp cửa sổ trượt có thể được thực hiện. Theo đó, việc quản lý hiệu quả của các sơ đồ quản lý ảnh tham chiếu khác nhau bao gồm các việc quản lý danh mục ảnh tham chiếu như được ứng dụng bởi VVC, một mặt, và phương pháp cửa sổ trượt như được ứng dụng bởi AVC, mặt khác, có thể được đề xuất.

Theo cách thực hiện, phương pháp còn bao gồm bước, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, thu nhận, từ dòng bit, trị số của thành phần cú pháp thứ hai (ví dụ, `max_num_tid0_ref_pics`) liên quan đến việc đánh dấu ảnh tham chiếu (khi nó được ứng dụng bởi AVC). Việc đánh dấu ảnh tham chiếu có thể cung cấp thông tin về số lớn nhất của các ảnh tham chiếu (trong lớp thời gian 0) sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video. Thông tin này có thể đã được sử dụng để xác định kích thước của bộ đệm ảnh được giải mã.

Theo cách thực hiện, các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu (khi nó được ứng dụng bởi VVC). Các thành phần cú pháp thứ nhất này liên quan đến việc xây dựng các danh mục ảnh tham chiếu có thể được bao gồm trong ít

nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một (đơn) danh mục ảnh tham chiếu. Việc ứng dụng cấu trúc cú pháp danh mục ảnh tham chiếu như vậy cho phép đối với việc lập mã hiệu quả của thông tin cần cho sự xây dựng của các danh mục ảnh tham chiếu.

Theo cách thực hiện, thành phần cú pháp RPL là cờ (ví dụ, được đặt tên là “sps_rpl_flag”, xem thêm phần mô tả chi tiết dưới đây) trong đó trị số thứ nhất (ví dụ, 1) của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai (ví dụ, 0) của cờ khác với trị số thứ nhất định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit. Việc phát hiện/báo hiệu xem cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không nhờ cờ như vậy là rất hiệu quả xét về kích thước bit của việc lập mã.

Về nguyên lý, trị số của thành phần cú pháp RPL có thể được thu nhận từ một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh và đoạn đầu lát. Nhờ đó, việc báo hiệu trên các mức phân cấp khác nhau của các tập hợp thông số được dự tính.

Theo cách thực hiện, các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được thu nhận từ tập hợp thông số chuỗi và/hoặc đoạn đầu lát. Hơn nữa, các mức phân cấp khác nhau của các tập hợp thông số có thể được sử dụng một cách thích hợp.

Theo cách thực hiện phương pháp giải mã bao gồm bước, khi thành phần cú pháp RPL định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận trị số của ít nhất một thành phần cú pháp thứ ba (ví dụ, ref_pic_list_sps_flag) từ đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không. Nhờ đó, việc quản lý của quy trình dự đoán liên kết dựa vào các danh mục ảnh tham chiếu có thể được thực hiện một cách hữu hiệu.

Theo cách thực hiện, ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó trị số của một trong số hai thành phần cú pháp thứ ba được thu nhận đối với danh mục ảnh tham chiếu thứ nhất và, trong trường hợp mà thành phần cú pháp thứ tư (ví dụ, rpl1_idx_present_flag) được thu nhận từ dòng bit chỉ báo rằng thành

phần cú pháp thứ ba khác là có mặt trong dòng bit, trị số của thành phần cú pháp khác trong số hai thành phần cú pháp thứ ba được thu nhận đối với danh mục tham chiếu thứ hai.

Thành phần cú pháp thứ tư có thể là cờ khác, trị số là 1 của cờ khác chỉ báo rằng danh mục tham chiếu thứ hai là có mặt trong dòng bit. Ứng dụng của cờ như vậy cho phép đối với việc báo hiệu hiệu quả xét về kích thước bit của việc lập mã.

Theo cách thực hiện, ít nhất một trị số của thành phần cú pháp thứ ba được thu nhận trong trường hợp mà có ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một danh mục ảnh tham chiếu.

Ít nhất một trị số của thành phần cú pháp thứ ba có thể được thu nhận từ đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	

...	
}	

Theo cách thực hiện khác, ít nhất một trị số của thành phần cú pháp thứ ba được thu nhận từ đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

Theo khía cạnh thứ hai, phương pháp mã hóa chuỗi video được đề xuất tương ứng với phương pháp giải mã nêu trên và đưa ra các ưu điểm giống nhau hoặc tương tự như được nêu trên. Phương pháp mã hóa chuỗi video theo khía cạnh thứ hai bao gồm các bước

xác định xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng (bởi thiết bị giải mã) của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết hay không;

tạo dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit trong trường hợp mà nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng (bởi thiết bị giải mã) cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Theo cách thực hiện, dòng bit bao gồm thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu không được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu.

Các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu có thể được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu.

Theo cách thực hiện, thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu cung cấp thông tin về số lớn nhất của các ảnh tham chiếu (trong lớp thời gian 0) sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video.

Theo cách thực hiện, thành phần cú pháp RPL là cờ và trị số thứ nhất (ví dụ 1) của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai (ví dụ, 0) của cờ khác với trị số thứ nhất định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit.

Thành phần cú pháp RPL có thể được báo hiệu trong một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh và đoạn đầu lát.

Các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể được bao gồm trong tập hợp thông số chuỗi và/hoặc đoạn đầu lát.

Theo cách thực hiện, ít nhất một thành phần cú pháp thứ ba được báo hiệu trong đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không, khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Theo cách thực hiện, dòng bit bao gồm thành phần cú pháp thứ tư và ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó một trong số hai thành phần cú pháp thứ ba là cho danh mục ảnh tham chiếu thứ nhất và thành phần cú pháp khác trong số hai thành phần cú pháp thứ ba là cho danh mục tham chiếu thứ hai trong trường hợp mà trị số của thành phần cú pháp thứ tư chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

Thành phần cú pháp thứ tư có thể là cờ khác, trị số là 1 của cờ khác chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

Theo cách thực hiện, ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu.

Thành phần cú pháp thứ ba có thể được báo hiệu trong đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	

(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

Theo cách thực hiện khác thành phần cú pháp thứ ba được báo hiệu trong đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	

}	
...	
}	

Hơn nữa, bộ giải mã được tạo cấu hình để thực hiện các phương án nêu trên của phương pháp giải mã sáng tạo và bộ mã hóa được tạo cấu hình để thực hiện các phương án nêu trên của phương pháp mã hóa sáng tạo được đề xuất. Cụ thể là, có đề xuất bộ giải mã bao gồm hệ mạch xử lý để thực hiện các phương án nêu trên của phương pháp giải mã sáng tạo và bộ mã hóa bao gồm hệ mạch xử lý để thực hiện các phương án nêu trên của phương pháp mã hóa sáng tạo.

Hơn nữa, có đề xuất sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện các phương pháp được nêu trên khi được thực hiện trên máy tính hoặc bộ xử lý.

Bộ giải mã miễn là bao gồm một hoặc nhiều bộ xử lý và phương tiện lưu trữ đọc được bởi máy tính không tạm thời được ghép nối với các bộ xử lý và lưu trữ sự lập trình để thực hiện bởi các bộ xử lý, trong đó sự lập trình, khi được thực hiện bởi các bộ xử lý, tạo cấu hình bộ giải mã để thực hiện các phương pháp nêu trên.

Bộ mã hóa miễn là bao gồm một hoặc nhiều bộ xử lý và phương tiện lưu trữ đọc được bởi máy tính không tạm thời được ghép nối với các bộ xử lý và lưu trữ sự lập trình để thực hiện bởi các bộ xử lý, trong đó sự lập trình, khi được thực hiện bởi các bộ xử lý, tạo cấu hình bộ mã hóa để thực hiện các phương pháp nêu trên.

Hơn nữa, có đề xuất phương tiện đọc được bởi máy tính không tạm thời mang mã chương trình mà, khi được thực hiện bởi thiết bị máy tính, khiến thiết bị máy tính thực hiện các phương pháp được nêu trên.

Hơn nữa, các thiết bị mã hóa và giải mã sau đây được đề xuất mà nó đưa ra các ưu điểm giống nhau hoặc tương tự như được nêu trên.

Theo khía cạnh khác, có đề xuất thiết bị giải mã để giải mã chuỗi video được mã hóa, bao gồm:

bộ phận thu nhận được tạo cấu hình để thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không và,

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu; và

bộ phận xây dựng RPL để, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu.

Cụ thể là, có đề xuất thiết bị giải mã để giải mã chuỗi video được mã hóa, bao gồm bộ phận thu nhận được tạo cấu hình để thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không và,

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu.

Theo cách thực hiện, bộ phận xây dựng RPL được tạo cấu hình để, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit.

Việc xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit có thể bao gồm thực hiện phương pháp cửa sổ trượt để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham

chiều là không có mặt trong dòng bit, thu nhận, từ dòng bit, trị số của thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu.

Các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu.

Theo cách thực hiện, các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, và bộ phận xây dựng RPL được tạo cấu hình để sử dụng mỗi cấu trúc cú pháp danh mục ảnh tham chiếu để xây dựng một danh mục ảnh tham chiếu.

Thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu có thể cung cấp thông tin về số lớn nhất của các ảnh tham chiếu sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video.

Thành phần cú pháp RPL có thể là cờ trong đó trị số thứ nhất (ví dụ, 1) của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai (ví dụ, 0) của cờ khác với trị số thứ nhất định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để thu nhận trị số của thành phần cú pháp RPS từ một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh và đoạn đầu lát.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để thu nhận các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu từ tập hợp thông số chuỗi và/hoặc đoạn đầu lát.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để, khi thành phần cú pháp RPL định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận trị số của ít nhất một thành phần cú pháp thứ ba từ đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không.

Theo cách thực hiện, ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó bộ phận thu nhận được tạo cấu hình để thu nhận trị số của một trong số hai thành phần cú pháp thứ ba đối với danh mục ảnh tham chiếu thứ nhất và, trong trường hợp mà thành phần cú pháp thứ tư được thu nhận từ dòng bit chỉ báo rằng

thành phần cú pháp thứ ba khác là có mặt trong dòng bit, trị số của thành phần cú pháp khác trong số hai thành phần cú pháp thứ ba đối với danh mục tham chiếu thứ hai.

Theo cách thực hiện, trong đó thành phần cú pháp thứ tư là cờ khác, trị số là 1 của cờ khác chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để thu nhận ít nhất một trị số của thành phần cú pháp thứ ba trong trường hợp mà có ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, và trong đó bộ phận xây dựng RPL được tạo cấu hình để sử dụng mỗi cấu trúc cú pháp danh mục ảnh tham chiếu để xây dựng một danh mục ảnh tham chiếu.

Theo cách thực hiện, bộ phận thu nhận được tạo cấu hình để thu nhận ít nhất một trị số của thành phần cú pháp thứ ba từ đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	

...	
}	

Hơn nữa, bộ phận thu nhận có thể được tạo cấu hình để thu nhận ít nhất một trị số của thành phần cú pháp thứ ba từ đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 &&	
rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	
...	
}	

Theo khía cạnh khác, có đề xuất thiết bị mã hóa để mã hóa chuỗi video, bao gồm:

bộ phận xác định để xác định xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết hay không;

bộ phận tạo dòng bit được tạo cấu hình để tạo dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit trong trường hợp mà nó được xác định bởi bộ phận xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Dòng bit có thể bao gồm thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu, khi nó được xác định bởi bộ phận xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu không được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu.

Theo cách thực hiện, các thành phần cú pháp thứ nhất liên quan đến việc xây dựng các danh mục ảnh tham chiếu được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu.

Theo cách thực hiện, thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu cung cấp thông tin về số lớn nhất của các ảnh tham chiếu sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video

Theo cách thực hiện, thành phần cú pháp RPL là cờ và trị số thứ nhất (ví dụ, 1) của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai (ví dụ, 0) của cờ khác với trị số thứ nhất định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit.

Thành phần cú pháp RPL có thể được bao gồm trong một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh và đoạn đầu lát.

Các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể được bao gồm trong tập hợp thông số chuỗi và/hoặc đoạn đầu lát.

Theo cách thực hiện, ít nhất một thành phần cú pháp thứ ba được bao gồm trong đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham

chiều có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không, khi nó được xác định bởi bộ phận xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Theo cách thực hiện, dòng bit bao gồm thành phần cú pháp thứ tư và ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó một trong số hai thành phần cú pháp thứ ba là cho danh mục ảnh tham chiếu thứ nhất và thành phần cú pháp khác trong số hai thành phần cú pháp thứ ba là cho danh mục tham chiếu thứ hai trong trường hợp mà trị số của thành phần cú pháp thứ tư chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

Thành phần cú pháp thứ tư có thể là cờ khác, trị số là 1 của cờ khác chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

Theo cách thực hiện, ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được bao gồm trong tập hợp thông số chuỗi, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu.

Thành phần cú pháp thứ ba có thể được bao gồm trong đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	

ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

Thành phần cú pháp thứ ba có thể được bao gồm trong đoạn đầu lát bao gồm

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

Chi tiết về một hoặc nhiều phương án được nêu trong các hình vẽ kèm theo và phần mô tả dưới đây. Các dấu hiệu, các đối tượng, và các ưu điểm khác sẽ rõ ràng từ phần mô tả, các hình vẽ, và các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Trong phần dưới đây, các phương án của sáng chế được mô tả chi tiết hơn dựa vào các hình vẽ kèm theo, trong đó:

Fig.1A là sơ đồ khối thể hiện ví dụ về hệ thống lập mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.1B là sơ đồ khối thể hiện ví dụ khác về hệ thống lập mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.2 là sơ đồ khối thể hiện ví dụ về bộ mã hóa video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.3 là sơ đồ khối thể hiện cấu trúc ví dụ của bộ giải mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.4 là sơ đồ khối minh họa ví dụ về máy mã hóa hoặc máy giải mã;

Fig.5 là sơ đồ khối minh họa ví dụ khác về máy mã hóa hoặc máy giải mã;

Fig.6 là sơ đồ khối thể hiện cấu trúc ví dụ về hệ thống cung cấp nội dung 3100 mà thực hiện dịch vụ phân phối nội dung;

Fig.7 là sơ đồ khối thể hiện cấu trúc của ví dụ về thiết bị đầu cuối;

Fig.8 là lưu đồ minh họa phương pháp giải mã chuỗi video được mã hóa theo phương án;

Fig.9 là lưu đồ minh họa phương pháp mã hóa chuỗi video theo phương án;

Fig.10 là sơ đồ khối thể hiện thiết bị giải mã theo phương án;

Fig.11 là sơ đồ khối thể hiện thiết bị mã hóa theo phương án.

Trong phần dưới đây, các ký hiệu tham chiếu giống nhau đề cập đến các đặc điểm giống nhau hoặc ít nhất là tương đương về mặt chức năng nếu không được định rõ theo cách khác.

Mô tả chi tiết sáng chế

Trong phần mô tả dưới đây, tham chiếu được thực hiện tới các hình vẽ kèm theo, mà nó tạo nên một phần của sáng chế, và mà nó thể hiện, bằng cách minh họa, các khía cạnh cụ thể của các phương án của sáng chế hoặc các khía cạnh cụ thể trong đó các phương án của sáng chế có thể được sử dụng. Được hiểu rằng, các phương án của sáng chế có thể được sử dụng theo các khía cạnh khác và bao gồm các sự thay đổi cấu trúc hoặc logic không được mô tả trên các hình vẽ. Do đó, phần mô tả chi tiết dưới đây không được coi là giới hạn, và phạm vi của sáng chế được định rõ bởi các điểm yêu cầu bảo hộ kèm theo.

Chẳng hạn, được hiểu rằng sáng chế cùng với phương pháp được mô tả có thể cũng đúng với thiết bị hoặc hệ thống tương ứng được tạo cấu hình để thực hiện phương pháp và ngược lại. Ví dụ, nếu một hoặc nhiều bước phương pháp cụ thể được mô tả, thiết bị tương ứng có thể bao gồm một hoặc nhiều bộ phận, ví dụ, các bộ phận chức năng, để thực hiện một hoặc nhiều bước phương pháp được mô tả (ví dụ, một bộ phận thực hiện một hoặc nhiều bước, hoặc nhiều bộ phận đều thực hiện một hoặc nhiều bước), thậm chí nếu một hoặc nhiều bộ phận như vậy không được minh họa hoặc mô tả rõ ràng trên các hình vẽ. Mặt khác, ví dụ, nếu máy cụ thể được mô tả dựa vào một hoặc nhiều bộ phận, ví dụ, các bộ phận chức năng, phương pháp tương ứng có thể bao gồm một bước để thực hiện chức năng của một hoặc nhiều bộ phận (ví dụ, một bước thực hiện chức năng của một hoặc nhiều bộ phận, hoặc nhiều bước đều thực hiện chức năng của một hoặc nhiều bộ phận), thậm chí nếu một hoặc nhiều bước như vậy không được minh họa hoặc mô tả rõ ràng trên các hình vẽ. Hơn nữa, được hiểu rằng các đặc điểm của các phương án lấy làm ví dụ khác nhau và/hoặc các khía cạnh được mô tả ở đây có thể được kết hợp với nhau, ngoại trừ được lưu ý cụ thể theo cách khác.

Lập mã video cụ thể đề cập đến việc xử lý của chuỗi các ảnh, mà nó tạo nên video hoặc chuỗi video. Thay vì thuật ngữ “ảnh (picture)” thuật ngữ “khung (frame)” hoặc “hình ảnh (image)” có thể được sử dụng như từ đồng nghĩa trong lĩnh vực lập mã video. Lập mã video (hoặc việc lập mã nói chung) bao gồm hai phần mã hóa video và giải mã video. Mã hóa video được thực hiện ở phía nguồn, cụ thể là bao gồm việc xử lý (ví dụ, bằng cách nén) các ảnh video ban đầu để làm giảm lượng dữ liệu được yêu cầu để biểu diễn các ảnh video (cho việc truyền và/hoặc lưu trữ hữu hiệu hơn). Giải mã video được

thực hiện ở phía đích và cụ thể bao gồm việc xử lý ngược so với bộ mã hóa để khôi phục các ảnh video. Các phương án đề cập đến “việc lập mã (coding)” của các ảnh video (hoặc các ảnh nói chung) sẽ được hiểu là liên quan đến “mã hóa” hoặc “giải mã” của các ảnh video hoặc các chuỗi video tương ứng. Sự kết hợp của phần mã hóa và phần giải mã cũng được đề cập đến là CODEC (lập mã và giải mã).

Trong trường hợp lập mã video không tổn thất, các ảnh video ban đầu có thể được khôi phục, nghĩa là các ảnh video được khôi phục có chất lượng giống như các ảnh video ban đầu (giả sử không có tổn thất truyền hoặc tổn thất dữ liệu trong khi lưu trữ hoặc truyền). Trong trường hợp lập mã video tổn thất, việc nén thêm, ví dụ, nhờ lượng tử hóa, được thực hiện, để làm giảm lượng dữ liệu biểu diễn các ảnh video, mà không thể được khôi phục hoàn toàn ở bộ giải mã, nghĩa là chất lượng của các ảnh video được khôi phục là thấp hơn hoặc xấu hơn so với chất lượng của các ảnh video ban đầu.

Một vài chuẩn lập mã video thuộc về nhóm của “các bộ mã hóa/giải mã video lai tổn thất” (nghĩa là kết hợp dự đoán theo không gian và thời gian trong việc lập mã biến đổi 2D và miền mẫu dùng để áp dụng lượng tử hóa trong miền biến đổi). Mỗi ảnh của chuỗi video được phân chia cụ thể thành tập hợp của các khối không chồng lấn và việc lập mã được thực hiện một cách cụ thể ở mức khối. Nói cách khác, ở bộ mã hóa video được xử lý một cách cụ thể, nghĩa là được mã hóa, ở mức khối (khối video), ví dụ, nhờ sử dụng dự đoán theo không gian (ảnh bên trong) và/hoặc dự đoán theo thời gian (ảnh liên kết) để tạo khối dự đoán, trừ khối dự đoán từ khối hiện tại (khối hiện được xử lý/cần được xử lý) để thu nhận khối dư, biến đổi khối dư và lượng tử hóa khối dư trong miền biến đổi để làm giảm lượng dữ liệu được truyền (nén), trong khi ở bộ giải mã việc xử lý ngược so với bộ mã hóa được áp dụng tới khối được mã hóa hoặc được nén để khôi phục khối hiện tại cho việc biểu diễn. Hơn nữa, bộ mã hóa lặp lại vòng lập xử lý bộ giải mã sao cho cả hai sẽ tạo các việc dự đoán giống hệt nhau (ví dụ, các việc dự đoán bên trong và liên kết) và/hoặc các việc khôi phục cho việc xử lý, nghĩa là việc lập mã, các khối tiếp theo.

Trong phần dưới đây, các phương án về hệ thống lập mã video 10, bộ mã hóa video 20 và bộ giải mã video 30 được mô tả dựa vào các hình vẽ từ Fig.1 đến Fig.3.

Fig.1A là sơ đồ khối giản lược minh họa hệ thống lập mã ví dụ 10, ví dụ, hệ thống lập mã video 10 (hoặc gọi ngắn gọn là hệ thống lập mã 10) mà có thể ứng dụng các kỹ

thuật của sáng chế này. Bộ mã hóa video 20 (hoặc gọi ngắn gọn là bộ mã hóa 20) và bộ giải mã video 30 (hoặc gọi ngắn gọn là bộ giải mã 30) của hệ thống lập mã video 10 biểu diễn các ví dụ về các thiết bị mà có thể được tạo cấu hình để thực hiện các kỹ thuật phù hợp với các ví dụ khác nhau được mô tả trong sáng chế.

Như được thể hiện trên Fig.1A, hệ thống lập mã 10 bao gồm thiết bị nguồn 12 được tạo cấu hình để cung cấp dữ liệu ảnh được mã hóa 21 ví dụ, tới thiết bị đích 14 để giải mã dữ liệu ảnh được mã hóa 13.

Thiết bị nguồn 12 bao gồm bộ mã hóa 20, và có thể theo cách bổ sung, nghĩa là một cách tùy chọn, bao gồm nguồn ảnh 16, bộ tiền xử lý (hoặc bộ phận tiền xử lý) 18, ví dụ, bộ tiền xử lý ảnh 18, và giao diện truyền thông hoặc bộ phận truyền thông 22.

Nguồn ảnh 16 có thể bao gồm hoặc là loại bất kỳ của thiết bị chụp ảnh, ví dụ camera để chụp ảnh thế giới thực, và/hoặc loại bất kỳ của thiết bị tạo ảnh, ví dụ bộ xử lý đồ họa máy tính để tạo ảnh động máy tính, hoặc loại bất kỳ của thiết bị khác để thu nhận và/hoặc cung cấp ảnh thế giới thực, ảnh được tạo bởi máy tính (ví dụ, nội dung màn hình, ảnh thực tế ảo (virtual reality, viết tắt là VR) và/hoặc sự kết hợp bất kỳ của nó (ví dụ, ảnh thực tế tăng cường (augmented reality, viết tắt là AR). Nguồn ảnh có thể là loại bất kỳ của bộ nhớ hoặc bộ phận lưu trữ lưu trữ ảnh bất kỳ trong số các ảnh nêu trên.

Phân biệt với bộ tiền xử lý 18 và việc xử lý được thực hiện bởi bộ phận tiền xử lý 18, ảnh hoặc dữ liệu ảnh 17 có thể cũng được đề cập đến là ảnh thô hoặc dữ liệu ảnh thô 17.

Bộ tiền xử lý 18 được tạo cấu hình để thu dữ liệu ảnh (thô) 17 và để thực hiện việc tiền xử lý trên dữ liệu ảnh 17 để thu nhận ảnh được tiền xử lý 19 hoặc dữ liệu ảnh được tiền xử lý 19. Việc tiền xử lý được thực hiện bởi bộ tiền xử lý 18 có thể, ví dụ, bao gồm tinh chỉnh, chuyển đổi định dạng màu (ví dụ, từ RGB thành YCbCr), chỉnh màu, hoặc khử nhiễu. Có thể hiểu rằng bộ phận tiền xử lý 18 có thể là thành phần tùy chọn.

Bộ mã hóa video 20 được tạo cấu hình để thu dữ liệu ảnh được tiền xử lý 19 và cung cấp dữ liệu ảnh được mã hóa 21 (chi tiết hơn sẽ được mô tả dưới đây, ví dụ, dựa vào Fig.2).

Giao diện truyền thông 22 của thiết bị nguồn 12 có thể được tạo cấu hình để thu dữ liệu ảnh được mã hóa 21 và để truyền dữ liệu ảnh được mã hóa 21 (hoặc phiên bản được

xử lý thêm bất kỳ của nó) qua kênh truyền thông 13 tới thiết bị khác, ví dụ, thiết bị đích 14 hoặc thiết bị bất kỳ khác, cho việc lưu trữ hoặc việc khôi phục trực tiếp.

Thiết bị đích 14 bao gồm bộ giải mã 30 (ví dụ, bộ giải mã video 30), và có thể theo cách bổ sung, nghĩa là một cách tùy chọn, bao gồm giao diện truyền thông hoặc bộ phận truyền thông 28, bộ hậu xử lý 32 (hoặc bộ phận hậu xử lý 32) và thiết bị hiển thị 34.

Giao diện truyền thông 28 của thiết bị đích 14 được tạo cấu hình thu dữ liệu ảnh được mã hóa 21 (hoặc phiên bản được xử lý thêm bất kỳ của nó), ví dụ, một cách trực tiếp từ thiết bị nguồn 12 hoặc từ nguồn bất kỳ khác, ví dụ, thiết bị lưu trữ, ví dụ, dữ liệu ảnh được mã hóa thiết bị lưu trữ, và cung cấp dữ liệu ảnh được mã hóa 21 tới bộ giải mã 30.

Giao diện truyền thông 22 và giao diện truyền thông 28 có thể được tạo cấu hình để truyền hoặc thu dữ liệu ảnh được mã hóa 21 hoặc dữ liệu được mã hóa 13 qua liên kết truyền thông trực tiếp giữa thiết bị nguồn 12 và thiết bị đích 14, ví dụ, kết nối không dây hoặc nối dây trực tiếp, hoặc qua loại bất kỳ của mạng, ví dụ, mạng không dây hoặc nối dây hoặc sự kết hợp bất kỳ của nó, hoặc loại bất kỳ của mạng cá nhân và công cộng, hoặc loại bất kỳ của sự kết hợp của nó.

Giao diện truyền thông 22 có thể, ví dụ, được tạo cấu hình để bao gói dữ liệu ảnh được mã hóa 21 thành định dạng thích hợp, ví dụ, gói, và/hoặc xử lý dữ liệu ảnh được mã hóa nhờ sử dụng loại bất kỳ của việc mã hóa truyền hoặc xử lý để truyền qua liên kết truyền thông hoặc mạng truyền thông.

Giao diện truyền thông 28, tạo nên bản sao của giao diện truyền thông 22, có thể, ví dụ, được tạo cấu hình để thu dữ liệu được truyền và xử lý dữ liệu truyền nhờ sử dụng loại bất kỳ của việc xử lý hoặc giải mã truyền tương ứng và/hoặc tách gói để thu nhận dữ liệu ảnh được mã hóa 21.

Cả giao diện truyền thông 22 và giao diện truyền thông 28 đều có thể được tạo cấu hình như các giao diện truyền thông một chiều như được chỉ báo bởi mũi tên dùng cho kênh truyền thông 13 trên Fig.1A trở từ thiết bị nguồn 12 tới thiết bị đích 14, hoặc các giao diện truyền thông hai chiều, và có thể được tạo cấu hình, ví dụ, để gửi và thu các thông báo, ví dụ, để thiết đặt sự kết nối, để báo nhận và trao đổi thông tin bất kỳ khác

liên quan đến liên kết truyền thông và/hoặc truyền dữ liệu, ví dụ, truyền dữ liệu ảnh được mã hóa.

Bộ giải mã 30 được tạo cấu hình để thu dữ liệu ảnh được mã hóa 21 và cung cấp dữ liệu ảnh được giải mã 31 hoặc ảnh được giải mã 31 (chi tiết hơn sẽ được mô tả dưới đây, ví dụ, dựa vào Fig.3 hoặc Fig.5).

Bộ hậu xử lý 32 của thiết bị đích 14 được tạo cấu hình để hậu xử lý dữ liệu ảnh được giải mã 31 (cũng được gọi là dữ liệu ảnh được khôi phục), ví dụ, ảnh được giải mã 31, để thu nhận dữ liệu ảnh được hậu xử lý 33, ví dụ, ảnh được hậu xử lý 33. Việc hậu xử lý được thực hiện bởi bộ phận hậu xử lý 32 có thể bao gồm, ví dụ, chuyển đổi định dạng màu (ví dụ, từ YCbCr thành RGB), chỉnh màu, tinh chỉnh, hoặc lấy mẫu lại, hoặc việc xử lý bất kỳ khác, ví dụ, để chuẩn bị dữ liệu ảnh được giải mã 31 cho việc hiển thị, ví dụ, bởi thiết bị hiển thị 34.

Thiết bị hiển thị 34 của thiết bị đích 14 được tạo cấu hình để thu dữ liệu ảnh được hậu xử lý 33 cho việc hiển thị ảnh, ví dụ, tới người dùng hoặc người xem. Thiết bị hiển thị 34 có thể là hoặc bao gồm loại bất kỳ của màn hình để biểu diễn ảnh được khôi phục, ví dụ, màn hình hoặc màn hiển thị bên ngoài hoặc được tích hợp. Các màn hình có thể, ví dụ, bao gồm các màn hình tinh thể lỏng (liquid crystal display, viết tắt là LCD), các màn hình điốt phát sáng hữu cơ (organic light emitting diode, viết tắt là OLED), các màn hình plasma, các máy chiếu, các màn hình vi LED, tinh thể lỏng trên silicon (liquid crystal on silicon, viết tắt là LCoS), bộ xử lý ánh sáng số (digital light processor, viết tắt là DLP) hoặc loại bất kỳ của màn hình khác.

Mặc dù Fig.1A mô tả thiết bị nguồn 12 và thiết bị đích 14 như các thiết bị tách biệt, các phương án về các thiết bị có thể cũng bao gồm cả hai hoặc cả hai chức năng, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng. Theo các phương án như vậy, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng có thể được thực hiện nhờ sử dụng cùng phần cứng và/hoặc phần mềm hoặc bởi phần cứng và/hoặc phần mềm riêng biệt hoặc sự kết hợp bất kỳ của nó.

Như sẽ thấy rõ đối với người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng dựa vào phần mô tả, sự tồn tại và phân tách (chính xác) của các chức năng của các

bộ phận hoặc các chức năng khác nhau bên trong thiết bị nguồn 12 và/hoặc thiết bị đích 14 như được thể hiện trên Fig.1A có thể thay đổi tùy thuộc vào thiết bị và ứng dụng thực tế.

Bộ mã hóa 20 (ví dụ, bộ mã hóa video 20) hoặc bộ giải mã 30 (ví dụ, bộ giải mã video 30) hoặc cả bộ mã hóa 20 và bộ giải mã 30 có thể được thực hiện qua hệ mạch xử lý như được thể hiện trên Fig.1B, chẳng hạn như một hoặc nhiều bộ vi xử lý, các bộ xử lý tín hiệu số (digital signal processor, viết tắt là DSP), các mạch tích hợp chuyên dụng (application-specific integrated circuit, viết tắt là ASIC), các mảng cổng lập trình được dạng trường (field-programmable gate array, viết tắt là FPGA), logic rời rạc, phần cứng, lập mã video chuyên dụng hoặc các sự kết hợp bất kỳ của nó. Bộ mã hóa 20 có thể được thực hiện qua hệ mạch xử lý 46 để bao gồm các môđun khác nhau như được thảo luận đối với bộ mã hóa 20 trên Fig.2 và/hoặc hệ thống phụ hoặc hệ thống bộ mã hóa bất kỳ khác được mô tả ở đây. Bộ giải mã 30 có thể được thực hiện qua hệ mạch xử lý 46 để bao gồm các môđun khác nhau như được thảo luận đối với bộ giải mã 30 trên Fig.3 và/hoặc hệ thống phụ hoặc hệ thống bộ giải mã bất kỳ khác được mô tả ở đây. Hệ mạch xử lý có thể được tạo cấu hình để thực hiện các thao tác khác nhau như được thảo luận sau đây. Như được thể hiện trên Fig.5, nếu các kỹ thuật được thực hiện một phần trong phần mềm, thiết bị có thể lưu trữ các lệnh dùng cho phần mềm trong phương tiện lưu trữ đọc được bởi máy tính không tạm thời, thích hợp và có thể thực hiện các lệnh trong phần cứng nhờ sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế này. Bộ mã hóa video 20 hoặc bộ giải mã video 30 có thể được tích hợp như phần của bộ mã hóa/bộ giải mã được kết hợp (CODEC) trong thiết bị đơn lẻ, ví dụ, như được thể hiện trên Fig.1B.

Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm bất kỳ trong số phạm vi rộng của các thiết bị, bao gồm loại bất kỳ của các thiết bị cầm tay hoặc tĩnh, ví dụ, máy tính cầm tay (notebook) hoặc máy tính xách tay (laptop), các điện thoại di động, các điện thoại thông minh, máy tính bảng (tablet) hoặc máy tính dạng bảng (tablet computer), các camera, các máy tính để bàn, các thiết bị giải mã tín hiệu truyền hình (set-top box), các máy thu hình, các thiết bị hiển thị, các máy chơi phương tiện số (digital media player), các bảng trò chơi video (video gaming console), các thiết bị tạo dòng video (chẳng hạn như các máy chủ dịch vụ nội dung (content services server) hoặc các máy

chủ phân phối nội dung (content delivery server)), thiết bị thu phát rộng, thiết bị truyền phát rộng, hoặc tương tự và có thể không sử dụng hoặc loại bất kỳ của hệ điều hành. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị cho việc truyền thông không dây. Do đó, thiết bị nguồn 12 và thiết bị đích 14 có thể là các thiết bị truyền thông không dây.

Trong một số trường hợp, hệ thống lập mã video 10 được minh họa trên Fig.1A chỉ là ví dụ và các kỹ thuật của sáng chế có thể áp dụng tới các việc thiết đặt lập mã video (ví dụ, mã hóa video hoặc giải mã video) mà không nhất thiết bao gồm việc truyền thông dữ liệu bất kỳ giữa các thiết bị mã hóa và giải mã. Trong các ví dụ khác, dữ liệu được truy xuất từ bộ nhớ cục bộ, được tạo dòng qua mạng, hoặc tương tự. Thiết bị mã hóa video có thể mã hóa và lưu trữ dữ liệu tới bộ nhớ, và/hoặc thiết bị giải mã video có thể truy xuất và giải mã dữ liệu từ bộ nhớ. Trong một số ví dụ khác, mã hóa và giải mã được thực hiện bởi các thiết bị mà không truyền thông với thiết bị khác, nhưng dễ dàng mã hóa dữ liệu tới bộ nhớ và/hoặc truy xuất và giải mã dữ liệu từ bộ nhớ.

Nhằm thuận tiện cho việc mô tả, các phương án của sáng chế được mô tả ở đây, ví dụ, nhờ tham chiếu tới lập mã video hiệu quả cao (High-Efficiency Video Coding, viết tắt là HEVC) hoặc tới phần mềm tham chiếu của lập mã video đa năng (Versatile Video Coding, viết tắt là VVC), chuẩn lập mã video thế hệ tiếp theo được phát triển bởi nhóm cộng tác chung về lập mã video (Joint Collaboration Team on Video Coding, viết tắt là JCT-VC) của nhóm chuyên gia lập mã video (Video Coding Experts Group, viết tắt là VCEG) ITU-T và nhóm chuyên gia ảnh động (Motion Picture Experts Group, viết tắt là MPEG) ISO/IEC. Người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng sẽ hiểu rằng các phương án của sáng chế là không giới hạn ở HEVC hoặc VVC.

Bộ mã hóa và Phương pháp mã hóa

Fig.2 thể hiện sơ đồ khối giản lược của bộ mã hóa video ví dụ 20 mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Trong ví dụ trên Fig.2, bộ mã hóa video 20 bao gồm đầu vào 201 (hoặc giao diện đầu vào 201), bộ phận tính phần dư 204, bộ phận xử lý biến đổi 206, bộ phận lượng tử hóa 208, bộ phận lượng tử hóa ngược 210, và bộ phận xử lý biến đổi ngược 212, bộ phận khôi phục 214, bộ phận bộ lọc vòng lặp 220, bộ đệm ảnh được giải mã (decoded picture buffer, viết tắt là DPB) 230, bộ phận lựa chọn chế độ 260, bộ phận mã hóa entropi 270 và đầu ra 272 (hoặc giao diện đầu ra 272).

Bộ phận lựa chọn chế độ 260 có thể bao gồm bộ phận dự đoán liên kết 244, bộ phận dự đoán bên trong 254 và bộ phận phân chia 262. Bộ phận dự đoán liên kết 244 có thể bao gồm bộ phận đánh giá chuyển động và bộ phận bù chuyển động (không được thể hiện). Bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể cũng được đề cập đến là bộ mã hóa video lai hoặc bộ mã hóa video theo bộ mã hóa/giải mã video lai.

Bộ phận tính phần dư 204, bộ phận xử lý biến đổi 206, bộ phận lượng tử hóa 208, bộ phận lựa chọn chế độ 260 có thể được đề cập đến là việc tạo nên đường tín hiệu thuận của bộ mã hóa 20, trong khi bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận khôi phục 214, bộ đệm 216, bộ lọc vòng lặp 220, bộ đệm ảnh được giải mã (DPB) 230, bộ phận dự đoán liên kết 244 và bộ phận dự đoán bên trong 254 có thể được đề cập đến là tạo nên đường tín hiệu ngược của bộ mã hóa video 20, trong đó đường tín hiệu ngược của bộ mã hóa video 20 tương ứng với đường tín hiệu của bộ giải mã (xem bộ giải mã video 30 trên Fig.3). Bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận khôi phục 214, bộ lọc vòng lặp 220, bộ đệm ảnh được giải mã (DPB) 230, bộ phận dự đoán liên kết 244 và bộ phận dự đoán bên trong 254 cũng được đề cập đến việc tạo nên “bộ giải mã lắp sẵn” của bộ mã hóa video 20.

Cú pháp mức cao (High Level Syntax, viết tắt là HLS)

Có một số tập hợp mức cao của các thành phần cú pháp mà được thiết kế để truyền các thành phần cú pháp mức cao khác nhau chẳng hạn như các cờ điều khiển của công cụ, các thông số chuỗi video cụ thể (ví dụ, độ phân giải, độ sâu bit bên trong và bên ngoài, v.v.) và một số đoạn chương trình mã hóa/giải mã, mà có thể được tái sử dụng, ví dụ, bởi các ảnh, các lát hoặc các khối cụ thể của ảnh.

Trong các bộ mã hóa/giải mã hiện nay, chẳng hạn như 3 tập hợp thông số VVC và EVC sau đây thường được sử dụng: 1) tập hợp thông số chuỗi (SPS); 2) tập hợp thông số ảnh (PPS); và 3) đoạn đầu lát (SH). SPS là tập hợp chung nhất và bao gồm thông tin, mà không thay đổi đối với hoặc bên trong một số ảnh được lập mã, ví dụ, độ phân giải video, độ sâu bit bên trong và bên ngoài thường được báo hiệu trong SPS (hoặc ở mức SPS). PPS bao gồm thông tin, mà có thể thay đổi từ ảnh tới ảnh, ví dụ, các mảnh hoặc thông tin QP delta mức PPS. SH chứa thông tin mà có thể thay đổi từ lát tới lát, cũng như thông tin ảnh liên quan như vậy mà là tương đối nhỏ hoặc chỉ liên quan đối với các

loại ảnh hoặc lát nhất định. SPS, PPS và SH có thể được xét đến tạo nên sự phân cấp của các tập hợp cú pháp mức cao, trong đó SPS định rõ mức cao nhất, PPS mức thấp hơn tiếp theo và SH mức thấp hơn nữa nằm trong sự phân cấp của các mức như vậy. Các phương án có thể được tạo cấu hình để thực hiện ba tập hợp thông số này và các tập hợp thông số khác.

Các ảnh & phân chia ảnh (các ảnh & các khối)

Bộ mã hóa 20 có thể được tạo cấu hình để thu, ví dụ, qua đầu vào 201, ảnh 17 (hoặc dữ liệu ảnh 17), ví dụ, ảnh của chuỗi các ảnh tạo nên video hoặc chuỗi video. Dữ liệu ảnh hoặc ảnh được thu có thể cũng là ảnh được tiền xử lý 19 (hoặc dữ liệu ảnh được tiền xử lý 19). Nhằm đơn giản, phần mô tả sau đây đề cập đến là ảnh 17. Ảnh 17 có thể cũng được đề cập đến là ảnh hiện tại hoặc ảnh được lập mã (cụ thể trong lập mã video để phân biệt ảnh hiện tại với các ảnh khác, ví dụ, các ảnh được mã hóa và/hoặc được giải mã trước đó của cùng chuỗi video, nghĩa là chuỗi video mà cũng bao gồm ảnh hiện tại).

Ảnh (kỹ thuật số) là hoặc có thể được coi là mảng hoặc ma trận hai chiều của các mẫu với các trị số cường độ. Mẫu trong mảng có thể cũng được gọi là điểm ảnh (pixel) (dạng ngắn gọn của thành phần ảnh) hoặc ảnh điểm (pel). Số của các mẫu theo chiều ngang và chiều dọc (hoặc trục) của mảng hoặc ảnh định rõ kích thước và/hoặc độ phân giải của ảnh. Đối với việc biểu diễn của màu sắc, cụ thể là ba thành phần màu sắc được ứng dụng, nghĩa là ảnh có thể được biểu diễn hoặc bao gồm ba mảng mẫu. Theo định dạng RGB hoặc không gian màu sắc ảnh bao gồm mảng mẫu đỏ, xanh lá cây và xanh da trời tương ứng. Tuy nhiên, trong lập mã video mỗi điểm ảnh cụ thể được biểu diễn theo định dạng độ chói và sắc độ hoặc không gian màu sắc, ví dụ, YCbCr, mà bao gồm thành phần độ chói được chỉ báo bởi Y (đôi khi L cũng được sử dụng thay thế) và hai thành phần sắc độ được chỉ báo bởi Cb và Cr. Thành phần độ chói (hoặc ngắn gọn là luma) Y biểu diễn cường độ mức xám hoặc độ sáng (ví dụ, giống như trong ảnh mức xám), trong khi hai thành phần sắc độ (hoặc ngắn gọn là chroma) Cb và Cr biểu diễn các thành phần thông tin màu sắc hoặc sắc độ. Theo đó, ảnh theo định dạng YCbCr bao gồm mảng mẫu độ chói của các trị số mẫu độ chói (Y), và hai hằng mẫu sắc độ của các trị số sắc độ (Cb và Cr). Các ảnh theo định dạng RGB có thể được chuyển đổi hoặc được biến đổi thành định dạng YCbCr và ngược lại, quy trình xử lý cũng được biết đến

là biến đổi hoặc chuyển đổi màu sắc. Nếu ảnh là đơn sắc, ảnh có thể bao gồm chỉ mảng mẫu độ chói. Theo đó, ảnh có thể là, ví dụ, mảng của các mẫu độ chói theo định dạng đơn sắc hoặc mảng của các mẫu độ chói và hai mảng tương ứng của các mẫu sắc độ theo định dạng màu 4:2:0, 4:2:2, và 4:4:4.

Các phương án của bộ mã hóa video 20 có thể bao gồm bộ phận phân chia ảnh (không được mô tả trên Fig.2) được tạo cấu hình để phân chia ảnh 17 thành nhiều khối ảnh (cụ thể là không chồng lấn) 203. Các khối này có thể cũng được đề cập đến là các khối gốc, các khối cỡ lớn (macro) (H.264/AVC) hoặc các khối cây lập mã (coding tree block, viết tắt là CTB) hoặc các bộ phận cây lập mã (coding tree unit, viết tắt là CTU) (H.265/HEVC và VVC). Bộ phận phân chia ảnh có thể được tạo cấu hình để sử dụng cùng kích thước khối đối với tất cả các ảnh của chuỗi video và mạng lưới tương ứng định rõ kích thước khối, hoặc để thay đổi kích thước khối giữa các ảnh hoặc các tập hợp con hoặc các nhóm của các ảnh, và phân chia mỗi ảnh thành các khối tương ứng.

Theo các phương án khác, bộ mã hóa video có thể được tạo cấu hình để thu một cách trực tiếp khối 203 của ảnh 17, ví dụ, một, một vài hoặc tất cả các khối tạo nên ảnh 17. Khối ảnh 203 có thể cũng được đề cập đến là khối ảnh hiện tại hoặc khối ảnh được lập mã.

Giống như ảnh 17, khối ảnh 203 cũng là hoặc có thể được coi là mảng hoặc ma trận hai chiều của các mẫu với các trị số cường độ (các trị số mẫu), mặc dù có kích cỡ nhỏ hơn so với ảnh 17. Nói cách khác, khối 203 có thể bao gồm, ví dụ, một mảng mẫu (ví dụ, mảng độ chói trong trường hợp ảnh đơn sắc 17, hoặc mảng độ chói hoặc sắc độ trong trường hợp ảnh màu) hoặc ba mảng mẫu (ví dụ, mảng độ chói và hai mảng sắc độ trong trường hợp ảnh màu 17) hoặc số và/hoặc loại bất kỳ khác của các mảng tùy thuộc vào định dạng màu sắc được áp dụng. Số của các mẫu theo chiều ngang và chiều dọc (hoặc trục) của khối 203 định rõ kích thước của khối 203. Theo đó, khối có thể, ví dụ, mảng $M \times N$ (M cột nhân N hàng) của các mẫu, hoặc mảng $M \times N$ của các hệ số biến đổi.

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể được tạo cấu hình để mã hóa ảnh 17 theo từng khối, ví dụ, mã hóa và dự đoán được thực hiện trên mỗi khối 203.

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân chia và/hoặc mã hóa ảnh nhờ sử dụng các lát (cũng được đề cập đến là các lát video), trong đó ảnh có thể được phân chia thành hoặc được mã hóa nhờ sử dụng một hoặc nhiều lát (cụ thể là không chồng lấn), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều nhóm của các khối (ví dụ, các mảnh (H.265/HEVC và VVC) hoặc các miếng (VVC)).

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân chia và/hoặc mã hóa ảnh nhờ sử dụng các lát/các nhóm mảnh (cũng được đề cập đến là các nhóm mảnh video) và/hoặc các mảnh (cũng được đề cập đến là các mảnh video), trong đó ảnh có thể được phân chia thành hoặc được mã hóa nhờ sử dụng một hoặc nhiều lát/các nhóm mảnh (cụ thể là không chồng lấn), và mỗi lát/nhóm mảnh có thể bao gồm, ví dụ, một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều mảnh, trong đó mỗi mảnh, ví dụ, có thể có dạng hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ, các khối hoàn thiện hoặc một phần.

Tính phần dư

Bộ phận tính phần dư 204 có thể được tạo cấu hình để tính khối dư 205 (cũng được đề cập đến là phần dư 205) dựa vào khối ảnh 203 và khối dự đoán 265 (chi tiết hơn về khối dự đoán 265 được đưa ra dưới đây), ví dụ, bằng cách trừ các trị số mẫu của khối dự đoán 265 từ các trị số mẫu của khối ảnh 203, theo từng mẫu (từng điểm ảnh) để thu nhận khối dư 205 trong miền mẫu.

Biến đổi

Bộ phận xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng việc biến đổi, ví dụ, biến đổi côsin rời rạc (discrete cosine transform, viết tắt là DCT) hoặc biến đổi sin rời rạc (discrete sine transform, viết tắt là DST), trên các trị số mẫu của khối dư 205 để thu nhận các hệ số biến đổi 207 trong miền biến đổi. Các hệ số biến đổi 207 có thể cũng được gọi là các hệ số phần dư biến đổi và biểu diễn khối dư 205 trong miền biến đổi.

Bộ phận xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng các phép xấp xỉ số nguyên của DCT/DST, chẳng hạn như các sự biến đổi được định rõ đối với H.265/HEVC. So với biến đổi DCT trực giao, các phép xấp xỉ số nguyên như vậy cụ thể được gọi là hệ số cố định. Để duy trì tiêu chuẩn của khối dư mà được xử lý bởi các

sự biến đổi thuận và ngược, các hệ số mở rộng bổ sung được áp dụng như phần của quy trình biến đổi. Các hệ số mở rộng cụ thể được chọn lựa dựa vào các ràng buộc nhất định giống như các hệ số mở rộng là lũy thừa của hai đối với các thao tác dịch vị, độ sâu bit của các hệ số biến đổi, sự cân bằng giữa độ chính xác và các chi phí thực hiện, v.v.. Các hệ số mở rộng cụ thể, ví dụ, được định rõ đối với biến đổi ngược, ví dụ, bởi bộ phận xử lý biến đổi ngược 212 (và biến đổi ngược tương ứng, ví dụ, bởi bộ phận xử lý biến đổi ngược 312 ở bộ giải mã video 30) và các hệ số mở rộng tương ứng đối với biến đổi thuận, ví dụ, bởi bộ phận xử lý biến đổi 206, ở bộ mã hóa 20 có thể được định rõ theo đó.

Các phương án về bộ mã hóa video 20 (bộ phận xử lý biến đổi 206 tương ứng) có thể được tạo cấu hình để đưa ra các thông số biến đổi, ví dụ, loại biến đổi hoặc các sự biến đổi, ví dụ, trực tiếp hoặc được mã hóa hoặc được nén qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã video 30 có thể thu và sử dụng các thông số biến đổi cho việc giải mã.

Lượng tử hóa

Bộ phận lượng tử hóa 208 có thể được tạo cấu hình để lượng tử hóa các hệ số biến đổi 207 để thu nhận các hệ số được lượng tử hóa 209, ví dụ, bằng cách áp dụng lượng tử hóa vô hướng hoặc lượng tử hóa vector. Các hệ số được lượng tử hóa 209 có thể cũng được gọi là các hệ số biến đổi được lượng tử hóa 209 hoặc các hệ số phân dư được lượng tử hóa 209.

Quy trình lượng tử hóa có thể làm giảm độ sâu bit được kết hợp với một số hoặc tất cả trong số các hệ số biến đổi 207. Ví dụ, hệ số biến đổi n -bit có thể được làm tròn xuống tới hệ số biến đổi m -bit trong khi lượng tử hóa, trong đó n lớn hơn m . Mức độ lượng tử hóa có thể được biến đổi bằng cách điều chỉnh thông số lượng tử hóa (quantization parameter, viết tắt là QP). Ví dụ đối với lượng tử hóa vô hướng, việc mở rộng khác nhau có thể được áp dụng để đạt được việc lượng tử hóa mịn hơn hoặc thô hơn. Các kích thước bước lượng tử hóa nhỏ hơn tương ứng với việc lượng tử hóa mịn hơn, trong khi các kích thước bước lượng tử hóa lớn hơn tương ứng với việc lượng tử hóa thô hơn. Kích thước bước lượng tử hóa áp dụng được có thể được chỉ báo bởi thông số lượng tử hóa (QP). Thông số lượng tử hóa có thể ví dụ là chỉ số tập hợp được định rõ trước của các kích thước bước lượng tử hóa áp dụng được. Ví dụ, các thông số

lượng tử hóa nhỏ có thể tương ứng với việc lượng tử hóa mịn (các kích thước bước lượng tử hóa nhỏ) và các thông số lượng tử hóa lớn có thể tương ứng với việc lượng tử hóa thô (các kích thước bước lượng tử hóa lớn) hoặc ngược lại. Việc lượng tử hóa có thể bao gồm phép chia bởi kích thước bước lượng tử hóa và tương ứng và/hoặc việc giải lượng tử hóa ngược, ví dụ, bởi bộ phận lượng tử hóa ngược 210, có thể bao gồm phép nhân với kích thước bước lượng tử hóa. Các phương án theo một số chuẩn, ví dụ, HEVC, có thể được tạo cấu hình để sử dụng thông số lượng tử hóa để xác định kích thước bước lượng tử hóa. Nói chung, kích thước bước lượng tử hóa có thể được tính dựa vào thông số lượng tử hóa nhờ sử dụng phép xấp xỉ điểm cố định của phương trình bao gồm phép chia. Các hệ số mở rộng bổ sung có thể được đưa vào đối với việc lượng tử hóa và giải lượng tử hóa để khôi phục định chuẩn của khối dư, mà có thể đạt được sự biến đổi bởi việc mở rộng được sử dụng trong phép xấp xỉ điểm cố định của phương trình đối với kích thước bước lượng tử hóa và thông số lượng tử hóa. Theo một cách thực hiện ví dụ, việc mở rộng của giải lượng tử hóa và biến đổi ngược có thể được kết hợp. Theo cách khác, các bảng lượng tử hóa tùy biến có thể được sử dụng và được báo hiệu từ bộ mã hóa tới bộ giải mã, ví dụ, trong dòng bit. Lượng tử hóa là thao tác tổn thất, trong đó tổn thất tăng với việc tăng các kích thước bước lượng tử hóa.

Các phương án về bộ mã hóa video 20 (bộ phận lượng tử hóa 208 tương ứng) có thể được tạo cấu hình để đưa ra các thông số lượng tử hóa (QP), ví dụ, trực tiếp hoặc được mã hóa qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã video 30 có thể thu và áp dụng các thông số lượng tử hóa cho việc giải mã.

Lượng tử hóa ngược

Bộ phận lượng tử hóa ngược 210 được tạo cấu hình để áp dụng lượng tử hóa ngược của bộ phận lượng tử hóa 208 trên các hệ số được lượng tử hóa để thu nhận các hệ số được giải lượng tử hóa 211, ví dụ, bằng cách áp dụng ngược với sơ đồ lượng tử hóa được áp dụng bởi bộ phận lượng tử hóa 208 dựa vào hoặc sử dụng kích thước bước lượng tử hóa giống như bộ phận lượng tử hóa 208. Các hệ số được giải lượng tử hóa 211 có thể cũng được gọi là các hệ số phần dư được giải lượng tử hóa 211 và tương ứng - mặc dù cụ thể là không giống hệt với các hệ số biến đổi do tổn thất bởi lượng tử hóa - với các hệ số biến đổi 207.

Biến đổi ngược

Bộ phận xử lý biến đổi ngược 212 được tạo cấu hình để áp dụng biến đổi ngược của việc biến đổi được áp dụng bởi bộ phận xử lý biến đổi 206, ví dụ, biến đổi cosin rời rạc (discrete cosine transform, viết tắt là DCT) ngược hoặc biến đổi sin rời rạc (discrete sine transform, viết tắt là DST) ngược hoặc các sự biến đổi ngược khác, để thu nhận khối dư được khôi phục 213 (hoặc các hệ số được giải lượng tử hóa tương ứng 213) trong miền mẫu. Khối dư được khôi phục 213 có thể cũng được gọi là khối biến đổi 213.

Khôi phục

Bộ phận khôi phục 214 (ví dụ, bộ cộng hoặc bộ lấy tổng 214) được tạo cấu hình để cộng khối biến đổi 213 (nghĩa là khối dư được khôi phục 213) với khối dự đoán 265 để thu nhận khối được khôi phục 215 trong miền mẫu, ví dụ, bằng cách cộng – theo từng mẫu - các trị số mẫu của khối dư được khôi phục 213 và các trị số mẫu của khối dự đoán 265.

Lọc

Bộ phận lọc vòng lặp 220 (hoặc ngắn gọn là “bộ lọc vòng lặp” 220), được tạo cấu hình để lọc khối được khôi phục 215 để thu nhận khối được lọc 221, hoặc nói chung, để lọc các mẫu được khôi phục để thu nhận các trị số mẫu được lọc. Bộ phận lọc vòng lặp, ví dụ, được tạo cấu hình để làm nhẵn các sự chuyển đổi điểm ảnh, hoặc cách khác là nâng cao chất lượng video. Bộ phận lọc vòng lặp 220 có thể bao gồm một hoặc nhiều bộ lọc vòng lặp chẳng hạn như bộ lọc tách khối, bộ lọc dịch vị thích ứng mẫu (sample-adaptive offset, viết tắt là SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc vòng lặp thích ứng (adaptive loop filter, viết tắt là ALF), bộ lọc triệt nhiễu (noise suppression filter, viết tắt là NSF), hoặc sự kết hợp bất kỳ của nó. Theo ví dụ, bộ phận lọc vòng lặp 220 có thể bao gồm bộ lọc tách khối, bộ lọc SAO và bộ lọc ALF. Thứ tự của quy trình lọc có thể là lọc tách khối, SAO và ALF. Theo ví dụ khác, quy trình xử lý được gọi là ánh xạ độ chói với việc mở rộng sắc độ (luma mapping with chroma scaling, viết tắt là LMCS) (chẳng hạn như, bộ tạo lại dạng trong vòng lặp thích ứng (adaptive in-loop reshaper)) được bổ sung. Quy trình này được thực hiện trước khi tách khối. Theo ví dụ khác, quy trình lọc tách khối có thể cũng được áp dụng tới các mép khối con bên trong, ví dụ, các mép các khối con afin, các mép các khối con ATMVP, các mép biến đổi khối con (sub-block transform, viết tắt là SBT) và các mép phân chia con bên trong (intra

sub-partition, viết tắt là ISP). Mặc dù bộ phận lọc vòng lặp 220 được thể hiện trên Fig.2 là bộ lọc trong vòng lặp, theo các cấu hình khác, bộ phận lọc vòng lặp 220 có thể được thực hiện như bộ lọc vòng lặp sau. Khối được lọc 221 có thể cũng được gọi là khối được khôi phục được lọc 221.

Các phương án về bộ mã hóa video 20 (tương ứng bộ phận lọc vòng lặp 220) có thể được tạo cấu hình để đưa ra các thông số bộ lọc vòng lặp (chẳng hạn như các thông số bộ lọc SAO hoặc các thông số bộ lọc ALF hoặc các thông số LMCS), ví dụ, trực tiếp hoặc được mã hóa qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã 30 có thể thu và áp dụng cùng các thông số bộ lọc vòng lặp hoặc các bộ lọc vòng lặp tương ứng cho việc giải mã.

Bộ đệm ảnh được giải mã

Bộ đệm ảnh được giải mã (DPB) 230 có thể là bộ nhớ mà lưu trữ các ảnh tham chiếu, hoặc dữ liệu ảnh tham chiếu nói chung, cho việc mã hóa dữ liệu video bởi bộ mã hóa video 20. DPB 230 có thể được tạo nên bởi bất kỳ trong số các loại thiết bị nhớ khác nhau, chẳng hạn như bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory, viết tắt là DRAM), bao gồm DRAM đồng bộ (synchronous DRAM, viết tắt là SDRAM), RAM từ điện trở (magnetoresistive RAM, viết tắt là MRAM), RAM điện trở (resistive RAM, viết tắt là RRAM), hoặc các loại khác của các thiết bị nhớ. Bộ đệm ảnh được giải mã (DPB) 230 có thể được tạo cấu hình để lưu trữ một hoặc nhiều khối được lọc 221. Bộ đệm ảnh được giải mã 230 có thể còn được tạo cấu hình để lưu trữ các khối được lọc trước đó khác, ví dụ, các khối được lọc và được khôi phục trước đó 221, của cùng ảnh hiện tại hoặc của các ảnh khác nhau, ví dụ, các ảnh được khôi phục trước đó, và có thể cung cấp các ảnh (và các khối và các mẫu tham chiếu tương ứng) được khôi phục hoàn toàn trước đó, nghĩa là được giải mã, và/hoặc ảnh hiện tại được khôi phục một phần (và các khối và các mẫu tham chiếu tương ứng), ví dụ đối với dự đoán liên kết. Bộ đệm ảnh được giải mã (DPB) 230 có thể cũng được tạo cấu hình để lưu trữ một hoặc nhiều khối được khôi phục chưa được lọc 215, hoặc các mẫu được khôi phục chưa được lọc nói chung, ví dụ, nếu khối được khôi phục 215 không được lọc bởi bộ phận lọc vòng lặp 220, hoặc phiên bản được xử lý thêm bất kỳ khác của các khối được khôi phục hoặc các mẫu.

Trong phần mô tả bộ mã hóa/giải mã video, các ảnh được nhận dạng cho nhiều mục đích, bao gồm để sử dụng như ảnh tham chiếu trong việc dự đoán liên kết, cho việc đưa ra của các ảnh từ bộ đệm ảnh được giải mã (DPB), cho việc mở rộng của các vector chuyển động, cho việc dự đoán được lấy trọng số, v.v.. Trong AVC và HEVC, các ảnh có thể được nhận dạng bởi số đếm thứ tự ảnh (picture order count, viết tắt là POC). Trong AVC và HEVC, các ảnh trong DPB có thể được đánh dấu là “được sử dụng cho việc tham chiếu ngắn hạn,” “được sử dụng cho việc tham chiếu dài hạn,” hoặc “không được sử dụng cho việc tham chiếu”. Một khi ảnh đã được đánh dấu “không được sử dụng cho việc tham chiếu,” ảnh có thể không còn được sử dụng cho việc dự đoán. Khi ảnh không còn được cần cho việc đưa ra, ảnh có thể được loại bỏ khỏi DPB.

Trong AVC, có hai loại của các ảnh tham chiếu, ngắn hạn và dài hạn. Ảnh tham chiếu có thể được đánh dấu là “không được sử dụng cho việc tham chiếu” khi nó trở nên không được cần đến nữa cho việc tham chiếu dự đoán. Việc chuyển đổi trong số ba trạng thái này (ngắn hạn, dài hạn, và không được sử dụng cho việc tham chiếu) được điều khiển bởi quy trình đánh dấu ảnh tham chiếu được giải mã. Có hai cơ chế đánh dấu ảnh tham chiếu được giải mã đan xen, quy trình của sổ trượt ngầm và thao tác điều khiển quản lý bộ nhớ (memory management control operation, viết tắt là MMCO) rõ ràng. Quy trình của sổ trượt đánh dấu ảnh tham chiếu ngắn hạn là “không được sử dụng cho việc tham chiếu” khi số các khung tham chiếu là bằng số lớn nhất đưa ra (max_num_ref_frames trong tập hợp thông số chuỗi (SPS)). Các ảnh tham chiếu ngắn hạn được lưu trữ theo cách thức vào trước, ra trước sao cho các ảnh được giải mã ngắn hạn gần nhất được giữ trong DPB.

Quy trình MMCO rõ ràng có thể bao gồm nhiều lệnh MMCO. Lệnh MMCO có thể đánh dấu một hoặc nhiều các ảnh tham chiếu ngắn hạn hoặc dài hạn là “không được sử dụng cho việc tham chiếu,” có thể đánh dấu tất cả các ảnh là “không được sử dụng cho việc tham chiếu,” hoặc có thể đánh dấu ảnh tham chiếu hiện tại hoặc ảnh tham chiếu ngắn hạn hiện có là dài hạn và sau đó gán chỉ số ảnh dài hạn tới ảnh tham chiếu dài hạn đó.

Trong AVC, các thao tác đánh dấu ảnh tham chiếu cũng như các quy trình xử lý để đưa ra và loại bỏ các ảnh từ DPB được thực hiện sau khi ảnh đã được giải mã. Các phương án của sáng chế có thể được tạo cấu hình để thực hiện phương pháp bất kỳ trong

số các phương pháp này và có thể được tạo cấu hình để sử dụng quy trình của sổ trượt ngầm, thao tác điều khiển quản lý bộ nhớ rõ ràng (MMCO), hoặc cả hai.

Phương pháp khác dùng cho các việc quản lý ảnh tham chiếu được sử dụng trong VVC. Nó được gọi là phương pháp các danh mục ảnh tham chiếu (Reference Picture List, viết tắt là RPL) và bao gồm các khía cạnh sau, mỗi trong số đó có thể được áp dụng riêng lẻ, và một vài trong số đó có thể được áp dụng theo sự kết hợp. 1) Việc đánh dấu ảnh tham chiếu là trực tiếp dựa vào hai danh mục ảnh tham chiếu, chẳng hạn danh mục ảnh tham chiếu 0 và danh mục ảnh tham chiếu 1. 1a) Thông tin cho việc dẫn ra của hai danh mục ảnh tham chiếu được báo hiệu dựa vào các thành phần cú pháp và các cấu trúc cú pháp trong SPS, PPS, và/hoặc đoạn đầu lát. 1b) mỗi trong số hai danh mục ảnh tham chiếu đối với ảnh được báo hiệu rõ ràng trong cấu trúc danh mục ảnh tham chiếu. 1b.i) một hoặc nhiều cấu trúc danh mục ảnh tham chiếu có thể được báo hiệu trong SPS và mỗi trong số chúng có thể được tham chiếu tới bởi chỉ số từ đoạn đầu lát. 1b.ii) mỗi trong số danh mục ảnh tham chiếu 0 và 1 có thể được báo hiệu trực tiếp trong đoạn đầu lát. 2) Thông tin cho việc dẫn ra của hai danh mục ảnh tham chiếu được báo hiệu đối với tất cả các loại của các lát, nghĩa là, các lát B (dự đoán hai chiều), P (dự đoán một chiều), và I (bên trong). Thuật ngữ lát đề cập đến sự tập hợp của các bộ phận cây lập mã chẳng hạn như lát trong HEVC hoặc VVC WD hoặc EVC WD gần nhất; nó có thể cũng đề cập đến một số sự tập hợp khác của các bộ phận cây lập mã chẳng hạn như mảnh trong HEVC. 3) hai danh mục ảnh tham chiếu được tạo cho tất cả các loại của các lát, nghĩa là, các lát B, P, và I. 4) hai danh mục ảnh tham chiếu được xây dựng trực tiếp mà không sử dụng quy trình khởi tạo danh mục ảnh tham chiếu và quy trình sửa đổi danh mục ảnh tham chiếu. 5) Trong mỗi trong số hai danh mục ảnh tham chiếu, các ảnh tham chiếu mà có thể được sử dụng cho việc dự đoán liên kết của ảnh hiện tại có thể chỉ được tham chiếu tới bởi số các mục nhập ở đầu của danh mục. Các mục nhập này được tham chiếu tới như các mục nhập hoạt động trong danh mục, trong khi các mục nhập khác được tham chiếu tới như các mục nhập không hoạt động trong danh mục. Số của tổng số các mục nhập và số của các mục nhập hoạt động trong danh mục có thể đều được dẫn ra. 6) ảnh được tham chiếu tới bởi mục nhập không hoạt động trong danh mục ảnh tham chiếu không được phép được tham chiếu tới bởi mục nhập khác trong danh mục ảnh tham chiếu hoặc mục nhập bất kỳ trong danh mục ảnh tham chiếu khác. 7) Các

ảnh tham chiếu dài hạn chỉ được nhận dạng bởi số nhất định của các POC LSB, trong đó số này có thể lớn hơn so với số của các POC LSB được báo hiệu trong các đoạn đầu lát cho việc dẫn ra của các trị số POC, và số này được chỉ báo trong SPS. 8) các cấu trúc danh mục ảnh tham chiếu được báo hiệu chỉ trong các đoạn đầu lát, cả các ảnh tham chiếu ngắn hạn và các ảnh tham chiếu dài hạn được nhận dạng bởi các POC LSB của chúng, mà có thể được biểu diễn bởi các số của các bit mà là khác với số của các bit được sử dụng để biểu diễn các POC LSB được báo hiệu trong các đoạn đầu lát cho việc dẫn ra của các trị số POC, và các số của các bit được sử dụng để biểu diễn các POC LSB cho việc nhận dạng các ảnh tham chiếu ngắn hạn và các ảnh tham chiếu dài hạn có thể là khác nhau. 9) các cấu trúc danh mục ảnh tham chiếu được báo hiệu chỉ trong các đoạn đầu lát, không có sự phân biệt được thực hiện giữa các ảnh tham chiếu ngắn hạn và dài hạn, tất cả các ảnh tham chiếu chỉ được đặt tên là các ảnh tham chiếu, và các ảnh tham chiếu được nhận dạng bởi các POC LSB của chúng, mà có thể được biểu diễn bởi số của các bit mà là khác với số của các bit được sử dụng để biểu diễn các POC LSB được báo hiệu trong các đoạn đầu lát cho việc dẫn ra của các trị số POC. Các phương án của sáng chế có thể được tạo cấu hình để thực hiện phương pháp bất kỳ trong số các phương pháp này.

Cách thực hiện chung của phương pháp RPL giả sử rằng các thành phần cú pháp sau đây là có mặt trong dòng bit.

seq_parameter_set_rbsp() {	Ký hiệu mô tả
...	
rpl1_same_as_rpl0_flag	u(1)
for(i = 0; i < !rpl1_same_as_rpl0_flag ? 2 : 1; i++) {	
num_ref_pic_lists_in_sps[i]	ue(v)
for(j = 0; j < num_ref_pic_lists_in_sps[i]; j++)	
ref_pic_list_struct(i, j, long_term_ref_pics_flag)	
}	
...	
}	

pic_parameter_set_rbsp() {	Ký hiệu mô tả
...	
rpl1_idx_present_flag	u(1)
...	
}	

slice_header() {	Ký hiệu mô tả
...	
if(nal_unit_type != IRAP_NUT) {	
...	
for(i = 0; i < 2; i++) {	
if((i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	

ref_pic_list_struct(listIdx, rplsIdx, ltrpFlag) {	Ký hiệu mô tả
num_strp_entries[listIdx][rplsIdx]	ue(v)
if(ltrpFlag)	
num_ltrp_entries[listIdx][rplsIdx]	ue(v)
for(i = 0; i < NumEntriesInList[listIdx][rplsIdx]; i++) {	
if(num_ltrp_entries[listIdx][rplsIdx] > 0)	
lt_ref_pic_flag[listIdx][rplsIdx][i]	u(1)
if(!lt_ref_pic_flag[listIdx][rplsIdx][i]) {	
delta_poc_st[listIdx][rplsIdx][i]	ue(v)
if(delta_poc_st[listIdx][rplsIdx][i] > 0)	
strp_entry_sign_flag[listIdx][rplsIdx][i]	u(1)
} else	
poc_lsb_lt[listIdx][rplsIdx][i]	u(v)
}	
}	

Trong đó các ngữ nghĩa về các thành phần cú pháp mới được bổ sung liên quan đến RPL được đưa ra dưới đây. rpl1_same_as_rpl0_flag bằng 1 định rõ rằng các cấu trúc cú pháp num_ref_pic_lists_in_sps[1] và ref_pic_list_struct(1, rplsIdx, ltrpFlag) là không có mặt.

num_ref_pic_lists_in_sps[i] định rõ số của các cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx, ltrpFlag) với listIdx bằng i được bao gồm trong SPS. Trị số của num_ref_pic_lists_in_sps[i] sẽ nằm trong khoảng từ 0 đến 64.

ref_pic_list_struct(listIdx, rplsIdx, ltrpFlag) cấu trúc cú pháp được báo hiệu trực tiếp trong các đoạn đầu lát của ảnh hiện tại.

rpl1_idx_present_flag bằng 0 định rõ rằng ref_pic_list_sps_flag[1] và ref_pic_list_idx[1] là không có mặt trong các đoạn đầu lát. rpl1_idx_present_flag bằng 1 định rõ rằng ref_pic_list_sps_flag[1] và ref_pic_list_idx[1] có thể có mặt trong các đoạn đầu lát.

`ref_pic_list_sps_flag[ i ]` bằng 1 định rõ rằng danh mục ảnh tham chiếu i của ảnh hiện tại được dẫn ra dựa vào một trong số các cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )` với `listIdx` bằng i trong SPS hoạt động. `ref_pic_list_sps_flag[ i ]` bằng 0 định rõ rằng danh mục ảnh tham chiếu i của ảnh hiện tại được dẫn ra dựa vào cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )` với `listIdx` bằng i mà được bao gồm trực tiếp trong các đoạn đầu lát của ảnh hiện tại. Khi `num_ref_pic_lists_in_sps[ i ]` bằng 0, trị số của `ref_pic_list_sps_flag[ i ]` sẽ bằng 0. Khi `rpl1_idx_present_flag` bằng 0 và `ref_pic_list_sps_flag[ 0 ]` là có mặt, trị số của `ref_pic_list_sps_flag[ 1 ]` được suy ra là bằng `ref_pic_list_sps_flag[ 0 ]`. Khi không có mặt, trị số của `ref_pic_list_sps_flag[ i ]` được suy ra là bằng 0.

`ref_pic_list_idx[ i ]` định rõ chỉ số, trong danh mục của các cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )` với `listIdx` bằng i được bao gồm trong SPS hoạt động, của cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )` với `listIdx` bằng i mà được sử dụng cho việc dẫn ra của danh mục ảnh tham chiếu i của ảnh hiện tại. Thành phần cú pháp `ref_pic_list_idx[ i ]` được biểu diễn bởi $\text{Ceil}(\text{Log}_2(\text{num_ref_pic_lists_in_sps}[i]))$ các bit. Khi không có mặt, trị số của `ref_pic_list_idx[ i ]` được suy ra là bằng 0. Trị số của `ref_pic_list_idx[ i ]` sẽ nằm trong khoảng từ 0 đến `num_ref_pic_lists_in_sps[ i ] - 1`. Khi `rpl1_idx_present_flag` bằng 0 và `ref_pic_list_sps_flag[ 0 ]` là có mặt, trị số của `ref_pic_list_idx[ 1 ]` được suy ra là bằng `ref_pic_list_idx[ 0 ]`.

`num_strp_entries[ listIdx ][ rplIdx ]` định rõ số của các mục nhập STRP trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )`. Khi không có mặt, trị số của `num_ltrp_entries[ listIdx ][ rplIdx ]` được suy ra là bằng 0. Trị số của `num_strp_entries[ listIdx ][ rplIdx ]` sẽ nằm trong khoảng từ 0 đến `sps_max_dec_pic_buffering_minus1`.

`num_ltrp_entries[ listIdx ][ rplIdx ]` định rõ số của các mục nhập LTRP trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag )`. Khi không có mặt, trị số của `num_ltrp_entries[ listIdx ][ rplIdx ]` được suy ra là bằng 0.

biến `NumEntriesInList[ listIdx ][ rplIdx ]` được dẫn ra như sau:

$$\text{NumEntriesInList}[\text{listIdx}][\text{rplsIdx}] = \text{num_strp_entries}[\text{listIdx}][\text{rplsIdx}] + \text{num_ltrp_entries}[\text{listIdx}][\text{rplsIdx}] \quad (123)$$

Trị số của $\text{NumEntriesInList}[\text{listIdx}][\text{rplsIdx}]$ sẽ nằm trong khoảng từ 0 đến $\text{sps_max_dec_pic_buffering_minus1}$.

$\text{lt_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 1 định rõ rằng mục nhập thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx}, \text{ltrpFlag})$ là mục nhập LTRP. $\text{lt_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 0 định rõ rằng mục nhập thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx}, \text{ltrpFlag})$ là mục nhập STRP. Khi không có mặt, trị số của $\text{lt_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i]$ được suy ra là bằng 0.

Tổng của $\text{lt_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i]$ đối với tất cả các trị số của i trong khoảng từ 0 đến $\text{NumEntriesInList}[\text{listIdx}][\text{rplsIdx}] - 1$, sẽ bằng $\text{num_ltrp_entries}[\text{listIdx}][\text{rplsIdx}]$.

$\text{delta_poc_st}[\text{listIdx}][\text{rplsIdx}][i]$, khi mục nhập thứ i là mục nhập STRP thứ nhất trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{rplsIdx}, \text{ltrpFlag})$, định rõ độ chênh lệch tuyệt đối giữa các trị số số đếm thứ tự ảnh của ảnh hiện tại và ảnh được tham chiếu tới bởi mục nhập thứ i , hoặc, khi mục nhập thứ i là mục nhập STRP nhưng không phải mục nhập STRP thứ nhất trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{rplsIdx}, \text{ltrpFlag})$, định rõ độ chênh lệch tuyệt đối giữa các trị số số đếm thứ tự ảnh của các ảnh được tham chiếu tới bởi mục nhập thứ i và bởi mục nhập STRP trước đó trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx}, \text{ltrpFlag})$.

Trị số của $\text{delta_poc_st}[\text{listIdx}][\text{rplsIdx}][i]$ sẽ nằm trong khoảng từ -2^{15} đến $2^{15} - 1$.

$\text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 1 định rõ rằng mục nhập thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx}, \text{ltrpFlag})$ có trị số lớn hơn so với hoặc bằng 0. $\text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 0 định rõ rằng mục nhập thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx}, \text{ltrpFlag})$ có trị số nhỏ hơn 0. Khi không có mặt, trị số của $\text{strp_entry_sign_flag}[i][j]$ được suy ra là bằng 1.

Danh mục $\text{DeltaPocSt}[\text{listIdx}][\text{rplsIdx}]$ được dẫn ra như sau:

```

for( i = 0; i < NumEntriesInList[ listIdx ][ rplsIdx ]; i++ ) {
    if( !lt_ref_pic_flag[ i ][ RplsIdx[ i ] ][ j ]
)
    (124)
        DeltaPocSt[ listIdx ][ rplsIdx ][ i ] = (
            strp_entry_sign_flag[ listIdx ][ rplsIdx ][ i ] ) ?
                delta_poc_st[ listIdx ][ rplsIdx ][ i ] : 0 -
            delta_poc_st[ listIdx ][ rplsIdx ][ i ]
    }

```

poc_lsb_lt[listIdx][rplsIdx][i] định rõ trị số của môđun số đếm thứ tự ảnh MaxPicOrderCntLsb của ảnh được tham chiếu tới bởi mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx, ltrpFlag). Độ dài poc_lsb_lt[listIdx][rplsIdx][i] thành phần cú pháp là $\log_2_max_pic_order_cnt_lsb_minus4 + 4$ bit.

Các phương án của sáng chế có thể được tạo cấu hình để thực hiện phương pháp bất kỳ trong số các phương pháp này.

Lựa chọn chế độ (phân chia & dự đoán)

Bộ phận lựa chọn chế độ 260 bao gồm bộ phận phân chia 262, bộ phận dự đoán liên kết 244 và bộ phận dự đoán bên trong 254, và được tạo cấu hình để thu hoặc thu nhận dữ liệu ảnh ban đầu, ví dụ, khối ban đầu 203 (khối hiện tại 203 của ảnh hiện tại 17), và dữ liệu ảnh được khôi phục, ví dụ, các khối hoặc các mẫu được khôi phục được lọc và/hoặc chưa được lọc hoặc các khối của cùng ảnh (hiện tại) và/hoặc từ một hoặc nhiều ảnh được giải mã trước đó, ví dụ, từ bộ đệm ảnh được giải mã 230 hoặc các bộ đệm khác (ví dụ, bộ đệm tuyến tính, không được thể hiện). Dữ liệu ảnh được khôi phục được sử dụng như dữ liệu ảnh tham chiếu cho việc dự đoán, ví dụ, việc dự đoán liên kết hoặc dự đoán bên trong, để thu nhận khối dự đoán 265 hoặc bộ dự đoán 265.

Bộ phận lựa chọn chế độ 260 có thể được tạo cấu hình để xác định hoặc lựa chọn việc phân chia đối với khối hiện tại chế độ dự đoán (bao gồm không phân chia) và chế độ dự đoán (ví dụ, chế độ dự đoán bên trong hoặc liên kết) và tạo khối dự đoán tương ứng 265, mà được sử dụng cho việc tính của khối dự 205 và cho việc khôi phục của khối được khôi phục 215.

Các phương án về bộ phận lựa chọn chế độ 260 có thể được tạo cấu hình để lựa chọn việc phân chia và chế độ dự đoán (ví dụ, từ các phương án được hỗ trợ bởi hoặc sẵn có đối với bộ phận lựa chọn chế độ 260), mà cung cấp sự so khớp tốt nhất hoặc nói cách khác phần dư nhỏ nhất (phần dư nhỏ nhất nghĩa là việc nén tốt hơn cho việc truyền hoặc lưu trữ), hoặc tổn hao báo hiệu nhỏ nhất (tổn hao báo hiệu nhỏ nhất nghĩa là việc nén tốt hơn đối với việc truyền hoặc lưu trữ), hoặc cân nhắc hoặc cân bằng cả hai. Bộ phận lựa chọn chế độ 260 có thể được tạo cấu hình để xác định việc phân chia và chế độ dự đoán dựa vào tối ưu hóa tỷ lệ méo (rate distortion optimization, viết tắt là RDO), nghĩa là lựa chọn chế độ dự đoán mà cung cấp tỷ lệ méo nhỏ nhất. Các thuật ngữ như “tốt nhất”, “nhỏ nhất”, “tối ưu” v.v., trong ngữ cảnh này không nhất thiết đề cập đến toàn bộ “tốt nhất”, “nhỏ nhất”, “tối ưu”, v.v., nhưng có thể cũng đề cập đến sự hoàn chỉnh của tiêu chuẩn kết thúc hoặc lựa chọn giống như trị số vượt quá hoặc giảm xuống dưới ngưỡng hoặc các ràng buộc khác có khả năng dẫn đến “lựa chọn dưới mức tối ưu” nhưng làm giảm độ phức tạp và thời gian xử lý.

Nói cách khác, bộ phận phân chia 262 có thể được tạo cấu hình để phân chia ảnh từ chuỗi video thành chuỗi của các bộ phận cây lập mã (các CTU), và CTU 203 có thể được phân chia thêm thành các phần chia khối nhỏ hơn hoặc các khối con (mà nó lại tạo nên các khối), ví dụ, sử dụng lặp lại việc phân chia cây tứ phân (quad-tree, viết tắt là QT), phân chia nhị phân (binary tree, viết tắt là BT) hoặc phân chia cây tam phân (triple-tree, viết tắt là TT) hoặc sự kết hợp bất kỳ của nó, và để thực hiện, ví dụ, dự đoán cho mỗi trong số các phần chia khối hoặc các khối con, trong đó lựa chọn chế độ bao gồm sự lựa chọn của cấu trúc cây của khối được phân chia 203 và các chế độ dự đoán được áp dụng tới mỗi trong số các phần chia khối hoặc các khối con.

Trong phần dưới đây, việc phân chia (ví dụ, bởi bộ phận phân chia 260) và việc xử lý dự đoán (bởi bộ phận dự đoán liên kết 244 và bộ phận dự đoán bên trong 254) được thực hiện bởi bộ mã hóa video ví dụ 20 sẽ được giải thích chi tiết hơn.

Phân chia

Bộ phận phân chia 262 có thể được tạo cấu hình để phân chia ảnh từ chuỗi video thành chuỗi của các bộ phận cây lập mã (các CTU), và bộ phận phân chia 262 có thể phân chia (hoặc phân tách) bộ phận cây lập mã (CTU) 203 thành các phần chia nhỏ hơn, ví dụ, các khối nhỏ hơn có kích thước hình vuông hoặc hình chữ nhật. Đối với ảnh mà

có ba mảng mẫu, CTU gồm có khối $N \times N$ của các mẫu độ chói cùng với hai khối tương ứng của các mẫu sắc độ. Kích thước cho phép lớn nhất của khối độ chói trong CTU được định rõ là 128×128 trong lập mã video đa năng (Versatile Video coding, viết tắt là VVC) đang phát triển, nhưng nó có thể được định rõ là trị số khác 128×128 trong tương lai, ví dụ, 256×256 . Các CTU của ảnh có thể được tạo chùm/được tạo nhóm như các lát/các nhóm mảnh, các mảnh hoặc các miếng. Mảnh bao phủ vùng hình chữ nhật của ảnh, và mảnh có thể được chia thành một hoặc nhiều miếng. Miếng gồm có số các hàng CTU bên trong mảnh. Mảnh mà không được phân chia thành nhiều miếng có thể được đề cập đến là miếng. Tuy nhiên, miếng là tập hợp con đúng của mảnh và không được đề cập đến là mảnh. Có hai chế độ của các nhóm mảnh được hỗ trợ trong VVC, chẳng hạn như chế độ nhóm mảnh/lát quét mảnh và chế độ lát hình chữ nhật. Trong chế độ nhóm mảnh quét mảnh, nhóm mảnh/lát chứa chuỗi của các mảnh trong việc quét mảnh mảnh của ảnh. Trong chế độ lát hình chữ nhật, lát chứa số các miếng của ảnh mà tạo nên dưới dạng tập hợp vùng hình chữ nhật của ảnh. Các miếng nằm trong lát hình chữ nhật là theo thứ tự quét mảnh miếng của lát. Các khối nhỏ hơn này (mà có thể cũng được gọi là các khối con) có thể còn được phân chia thậm chí thành các phân chia nhỏ hơn. Điều này cũng được đề cập đến việc phân chia cây hoặc phân chia cây phân cấp, trong đó khối gốc, ví dụ, ở mức cây gốc 0 (mức phân cấp 0, độ sâu 0), có thể được phân chia dưới dạng đệ quy, ví dụ, được phân chia thành hai hoặc nhiều hơn hai khối có mức cây thấp hơn tiếp theo, ví dụ, các nút ở mức cây 1 (mức phân cấp 1, độ sâu 1), trong đó các khối này có thể lại được phân chia thành hai hoặc nhiều hơn hai khối có mức thấp hơn tiếp theo, ví dụ, mức cây 2 (mức phân cấp 2, độ sâu 2), v.v., cho đến khi việc phân chia được kết thúc, ví dụ, bởi vì tiêu chuẩn kết thúc được thỏa mãn, ví dụ, độ sâu cây lớn nhất hoặc kích thước khối nhỏ nhất đạt được. Các khối mà không được phân chia thêm cũng được đề cập đến là các khối lá hoặc các nút lá của cây. Cây sử dụng phân chia thành hai phân chia được đề cập đến là cây nhị phân (binary-tree, viết tắt là BT), cây sử dụng phân chia thành ba phân chia được đề cập đến là cây tam phân (ternary-tree, viết tắt là TT), và cây sử dụng phân chia thành bốn phân chia được đề cập đến là cây tứ phân (quad-tree, viết tắt là QT).

Ví dụ, bộ phận cây lập mã (CTU) có thể là hoặc bao gồm CTB của các mẫu độ chói, hai CTB tương ứng của các mẫu sắc độ của ảnh mà có ba mảng mẫu, hoặc CTB của các

mẫu của ảnh đơn sắc hoặc ảnh mà được lập mã sử dụng ba mặt phẳng màu sắc riêng biệt và các cấu trúc cú pháp được sử dụng để lập mã các mẫu. Do đó, khối cây lập mã (CTB) có thể là khối $N \times N$ của các mẫu cho một số trị số của N sao cho việc chia của thành phần thành các CTB là phân chia. Bộ phận lập mã (coding unit, viết tắt là CU) có thể là hoặc bao gồm khối lập mã của các mẫu độ chói, hai khối lập mã tương ứng của các mẫu sắc độ của ảnh mà có ba mảng mẫu, hoặc khối lập mã của các mẫu của ảnh đơn sắc hoặc ảnh mà được lập mã sử dụng ba mặt phẳng màu sắc riêng biệt và các cấu trúc cú pháp được sử dụng để lập mã các mẫu. Do đó, khối lập mã (CB) có thể là khối $M \times N$ của các mẫu cho một số trị số của M và N sao cho việc chia của CTB thành các khối lập mã là phân chia.

Theo các phương án, ví dụ, theo HEVC, bộ phận cây lập mã (CTU) có thể được phân tách thành các CU nhờ sử dụng cấu trúc cây tứ phân được biểu thị là cây lập mã. Việc quyết định xem có lập mã vùng ảnh nhờ sử dụng dự đoán ảnh liên kết (theo thời gian) hoặc ảnh bên trong (theo không gian) hay không được thực hiện ở mức CU lá. Mỗi CU lá có thể còn được phân tách thành một, hai hoặc bốn PU theo loại phân tách PU. Bên trong một PU, quy trình dự đoán như nhau được áp dụng và thông tin liên quan được truyền tới bộ giải mã trên cơ sở PU. Sau khi thu nhận khối dư bằng cách áp dụng quy trình dự đoán dựa vào loại phân tách PU, CU lá có thể được phân chia thành các bộ phận biến đổi (transform unit, viết tắt là TU) theo cấu trúc cây tứ phân khác tương tự với cây lập mã đối với CU.

Theo các phương án, ví dụ, theo chuẩn lập mã video gần đây nhất hiện đang phát triển, mà được đề cập đến là lập mã video đa năng (VVC), cây đa loại được đặt vào cây tứ phân được kết hợp nhờ sử dụng cấu trúc phân đoạn phân tách nhị phân và tam phân chẳng hạn được sử dụng để phân chia bộ phận cây lập mã. Trong cấu trúc cây lập mã bên trong bộ phận cây lập mã, CU có thể có dạng hình vuông hoặc hình chữ nhật. Ví dụ, bộ phận cây lập mã (CTU) đầu tiên được phân chia bởi cây tứ phân. Sau đó các nút lá cây tứ phân có thể còn được phân chia bởi cấu trúc cây đa loại. Có bốn loại phân tách trong cấu trúc cây đa loại, phân tách nhị phân theo chiều dọc (vertical binary splitting, gọi là SPLIT_BT_VER), phân tách nhị phân theo chiều ngang (horizontal binary splitting, gọi là SPLIT_BT_HOR), phân tách tam phân theo chiều dọc (vertical ternary splitting, gọi là SPLIT_TT_VER), và phân tách tam phân theo chiều ngang (horizontal

ternary splitting, gọi là SPLIT_TT_HOR). Các nút lá cây đa loại được gọi là các bộ phận lập mã (coding unit, viết tắt là CU), và trừ khi CU là quá lớn đối với độ dài biến đổi lớn nhất, việc phân đoạn này được sử dụng cho việc xử lý dự đoán và biến đổi mà không có bất kỳ việc phân chia thêm. Điều này có nghĩa là, trong hầu hết các trường hợp, CU, PU và TU có cùng kích thước khối trong cây tứ phân với cấu trúc khối lập mã cây đa loại được đưa vào. Ngoại lệ xuất hiện khi độ dài biến đổi được hỗ trợ lớn nhất là nhỏ hơn độ rộng hoặc độ cao của thành phần màu sắc của CU. VVC phát triển cơ chế báo hiệu duy nhất của thông tin phân tách phân chia trong cây tứ phân với cấu trúc cây lập mã cây đa loại được đưa vào. Trong cơ chế báo hiệu, bộ phận cây lập mã (CTU) được xử lý như gốc của cây tứ phân và đầu tiên được phân chia bởi cấu trúc cây tứ phân. Mỗi nút lá cây tứ phân (khi đủ lớn để cho phép nó) sau đó còn được phân chia bởi cấu trúc cây đa loại. Trong cấu trúc cây đa loại, cờ thứ nhất (`mtt_split_cu_flag`) được báo hiệu để chỉ báo xem nút còn được phân chia hay không; khi nút còn được phân chia, cờ thứ hai (`mtt_split_cu_vertical_flag`) được báo hiệu để chỉ báo hướng phân tách, và sau đó cờ thứ ba (`mtt_split_cu_binary_flag`) được báo hiệu để chỉ báo xem việc phân tách là phân tách nhị phân hay phân tách tam phân. Dựa vào các trị số của `mtt_split_cu_vertical_flag` và `mtt_split_cu_binary_flag`, chế độ phân tách cây đa loại (multi-type tree splitting mode, viết tắt là `MttSplitMode`) của CU có thể được dẫn ra bởi bộ giải mã dựa vào quy tắc được định rõ trước hoặc bảng. Cần lưu ý, đối với thiết kế nhất định, ví dụ, khối độ chói 64×64 và thiết kế đường ống dẫn sắc độ 32×32 trong các bộ giải mã phần cứng VVC, việc phân tách TT bị cấm khi độ rộng hoặc độ cao của khối lập mã độ chói lớn hơn 64, như được thể hiện trên Fig.6. Việc phân tách TT cũng bị cấm khi độ rộng hoặc độ cao của khối lập mã sắc độ lớn hơn 32. Thiết kế đường ống dẫn sẽ chia ảnh thành các bộ phận dữ liệu đường ống ảo (Virtual pipeline data unit, viết tắt là VPDU) mà được định rõ là các bộ phận không chồng lấn trong ảnh. Trong các bộ giải mã phần cứng, các VPDU liên tiếp được xử lý bởi nhiều tầng đường ống một cách đồng thời. Kích thước VPDU là tương đối tỷ lệ với kích thước bộ đệm trong hầu hết các tầng đường ống, vì vậy đó là điều quan trọng để giữ kích thước VPDU nhỏ. Trong hầu hết các bộ giải mã phần cứng, kích thước VPDU có thể được thiết đặt tới kích thước khối biến đổi (TB) lớn nhất. Tuy nhiên, trong VVC, việc phân chia cây tam phân (TT) và cây nhị phân (BT) có thể dẫn tới làm tăng kích thước của VPDU

Ngoài ra, cần lưu ý rằng, khi phần của khối nút cây vượt quá ranh giới ảnh dưới cùng hoặc bên phải, khối nút cây buộc phải được phân tách cho đến khi tất cả các mẫu của mỗi CU được lập mã được nằm phía trong các ranh giới ảnh.

Bằng ví dụ, công cụ các phần chia nhỏ bên trong (Intra Sub-Partition, viết tắt là ISP) có thể chia các khối được dự đoán bên trong độ chói theo chiều dọc hoặc theo chiều ngang thành 2 hoặc 4 phần chia con tùy thuộc vào kích thước khối.

Theo một ví dụ, bộ phận lựa chọn chế độ 260 của bộ mã hóa video 20 có thể được tạo cấu hình để thực hiện sự kết hợp bất kỳ của các kỹ thuật phân chia được mô tả ở đây.

Như được nêu trên, bộ mã hóa video 20 được tạo cấu hình để xác định hoặc lựa chọn chế độ dự đoán tốt nhất hoặc tối ưu từ tập hợp của (ví dụ, được xác định trước) các chế độ dự đoán. Tập hợp của các chế độ dự đoán có thể bao gồm, ví dụ, các chế độ dự đoán bên trong và/hoặc các chế độ dự đoán liên kết.

Dự đoán bên trong

Tập hợp của các chế độ dự đoán bên trong có thể bao gồm 35 chế độ dự đoán bên trong khác nhau, ví dụ, các chế độ vô hướng giống như chế độ DC (hoặc trung bình) và chế độ mặt phẳng, hoặc các chế độ có hướng, ví dụ, như được định rõ trong HEVC, hoặc có thể bao gồm 67 chế độ dự đoán bên trong khác nhau, ví dụ, các chế độ vô hướng giống như chế độ DC (hoặc trung bình) và chế độ mặt phẳng, hoặc các chế độ có hướng, ví dụ, như được định rõ đối với VVC. Bằng ví dụ, một vài các chế độ dự đoán bên trong góc thông thường được thay thế một cách thích ứng bằng các chế độ dự đoán bên trong góc rộng đối với các khối không phải hình vuông, ví dụ, như được định rõ trong VVC. Với ví dụ khác, để tránh các thao tác chia đôi với dự đoán DC, chỉ phía dài hơn được sử dụng để tính toán mức trung bình đối với các khối không phải hình vuông. Và, các kết quả dự đoán bên trong của chế độ mặt phẳng có thể còn được biến đổi bởi phương pháp kết hợp dự đoán bên trong phụ thuộc vị trí (position dependent intra prediction combination, viết tắt là PDPC).

Bộ phận dự đoán bên trong 254 được tạo cấu hình để sử dụng các mẫu được khôi phục của các khối lân cận của cùng ảnh hiện tại để tạo ra khối dự đoán bên trong 265 theo chế độ dự đoán bên trong của tập hợp của các chế độ dự đoán bên trong.

Bộ phận dự đoán bên trong 254 (hoặc nói chung là bộ phận lựa chọn chế độ 260) còn được tạo cấu hình để đưa ra các thông số dự đoán bên trong (hoặc nói chung là thông tin chỉ báo về chế độ dự đoán bên trong được lựa chọn cho khối) tới bộ phận mã hóa entropi 270 ở dạng các thành phần cú pháp 266 cho việc bao gồm trong dữ liệu ảnh được mã hóa 21, sao cho, ví dụ, bộ giải mã video 30 có thể thu và sử dụng các thông số dự đoán cho việc giải mã.

Dự đoán liên kết

Tập hợp của (hoặc có thể) các chế độ dự đoán liên kết phụ thuộc vào các ảnh tham chiếu khả dụng (nghĩa là các ảnh được giải mã ít nhất một phần trước đó, ví dụ, được lưu trữ trong DBP 230) và các thông số dự đoán liên kết khác, ví dụ, toàn bộ ảnh tham chiếu hay chỉ một phần, ví dụ, vùng cửa sổ tìm kiếm quanh vùng của khối hiện tại, của ảnh tham chiếu được sử dụng để tìm kiếm khối tham chiếu trùng khớp nhất, và/hoặc ví dụ, việc nội suy điểm ảnh được áp dụng hay không, ví dụ, nội suy nửa/bán điểm ảnh, một phần tư điểm ảnh và/hoặc 1/16 điểm ảnh.

Cùng với các chế độ dự đoán nêu trên, chế độ bỏ qua, chế độ trực tiếp và/hoặc chế độ dự đoán liên kết khác có thể được áp dụng.

Ví dụ, dự đoán hợp nhất được mở rộng, danh mục ứng viên hợp nhất của chế độ như vậy được xây dựng bằng cách bao gồm năm loại của các ứng viên theo thứ tự sau đây: MVP theo không gian từ các CU lân cận theo không gian, MVP theo thời gian từ các CU được sắp xếp, MVP trên cơ sở lịch sử từ bảng FIFO, MVP trung bình theo cặp và các MV không. Và tính chỉnh vector chuyển động phía bộ giải mã trên cơ sở so khớp hai chiều (decoder side motion vector refinement, viết tắt là DMVR) có thể được áp dụng để tăng độ chính xác của các MV của chế độ hợp nhất. Chế độ hợp nhất với MVD (MMVD), mà đến từ chế độ hợp nhất với các độ lệch vector chuyển động. Cờ MMVD được báo hiệu ngay sau khi gửi cờ bỏ qua và cờ hợp nhất để định rõ xem chế độ MMVD được sử dụng cho CU hay không. Và sơ đồ độ phân giải vector chuyển động thích ứng mức CU (adaptive motion vector resolution, viết tắt là AMVR) có thể được áp dụng. AMVR cho phép MVD của CU được lập mã theo độ chính xác khác nhau. Tùy thuộc vào chế độ dự đoán đối với CU hiện tại, các MVD của CU hiện tại có thể được lựa chọn một cách thích ứng. Khi CU được lập mã trong chế độ hợp nhất, chế độ dự đoán bên trong/liên kết được kết hợp (combined inter/intra prediction, viết tắt là CIIP) có thể

được áp dụng tới CU hiện tại. Việc lấy trung bình trọng số của các tín hiệu dự đoán bên trong và liên kết được thực hiện để thu nhận việc dự đoán CIIP. Dự đoán được bù chuyển động afin, trường chuyển động afin của khối được mô tả bởi thông tin chuyển động của hai vector chuyển động điểm điều khiển (4 thông số) hoặc ba vector chuyển động điểm điều khiển (6 thông số). Dự đoán vector chuyển động theo thời gian trên cơ sở khối con (subblock-based temporal motion vector prediction, viết tắt là SbTMVP), mà tương tự với dự đoán vector chuyển động theo thời gian (temporal motion vector prediction, viết tắt là TMVP) trong HEVC, nhưng dự đoán các vector chuyển động của các CU con nằm trong CU hiện tại. Dòng quang học hai chiều (Bi-directional optical flow, viết tắt là BDOF), được đề cập đến trước đó là BIO, là phiên bản đơn giản hơn mà yêu cầu phép tính ít hơn nhiều, đặc biệt là xét về số các phép nhân và kích thước của số nhân. Chế độ phân chia hình tam giác, theo chế độ như vậy, CU được phân tách thậm chí thành hai phần chia dạng hình tam giác, nhờ sử dụng phân tách theo đường chéo hoặc phân tách không theo đường chéo. Bên cạnh đó, chế độ dự đoán hai chiều được mở rộng vượt quá mức trung bình đơn giản để cho phép việc lấy trung bình trọng số của hai tín hiệu dự đoán.

Bộ phận dự đoán liên kết 244 có thể bao gồm bộ phận đánh giá chuyển động (motion estimation, viết tắt là ME) và bộ phận bù chuyển động (motion compensation, viết tắt là MC) (đều không được thể hiện trên Fig.2). Bộ phận đánh giá chuyển động có thể được tạo cấu hình để thu hoặc thu nhận khối ảnh 203 (khối ảnh hiện tại 203 của ảnh hiện tại 17) và ảnh được giải mã 231, hoặc ít nhất một hoặc nhiều khối được khôi phục trước đó, ví dụ, các khối được khôi phục của một hoặc nhiều ảnh được giải mã khác/khác nhau trước đó 231, cho việc đánh giá chuyển động. Ví dụ, chuỗi video có thể bao gồm ảnh hiện tại và các ảnh được giải mã trước đó 231, hoặc nói cách khác, ảnh hiện tại và các ảnh được giải mã trước đó 231 có thể là phần của hoặc tạo nên chuỗi các ảnh tạo nên chuỗi video.

Bộ mã hóa 20 có thể, ví dụ, được tạo cấu hình để lựa chọn khối tham chiếu từ nhiều khối tham chiếu của các ảnh giống nhau hoặc khác nhau của nhiều ảnh khác và cung cấp ảnh tham chiếu (hoặc chỉ số ảnh tham chiếu) và/hoặc dịch vị (dịch vị theo không gian) giữa vị trí (các tọa độ x, y) của khối tham chiếu và vị trí của khối hiện tại như các thông số dự đoán liên kết tới bộ phận đánh giá chuyển động. Dịch vị này cũng được gọi

là vector chuyển động (motion vector, viết tắt là MV).

Bộ phận bù chuyển động được tạo cấu hình để thu nhận, ví dụ, thu, thông số dự đoán liên kết và để thực hiện dự đoán liên kết dựa vào hoặc sử dụng thông số dự đoán liên kết để thu nhận khối dự đoán liên kết 265. Bù chuyển động, được thực hiện bởi bộ phận bù chuyển động, có thể bao hàm việc truy xuất hoặc tạo ra khối dự đoán dựa vào vector khối/chuyển động được xác định nhờ việc đánh giá chuyển động, có khả năng thực hiện các việc nội suy tới độ chính xác điểm ảnh con. Việc lọc nội suy có thể tạo ra các mẫu điểm ảnh bổ sung từ các mẫu điểm ảnh đã biết, do đó có khả năng làm tăng số các khối dự đoán ứng viên mà có thể được sử dụng để lập mã khối ảnh. Khi thu vector chuyển động đối với PU của khối ảnh hiện tại, bộ phận bù chuyển động có thể định vị khối dự đoán mà trong đó các điểm vector chuyển động trong một trong số các danh mục ảnh tham chiếu.

Bộ phận bù chuyển động có thể cũng tạo các thành phần cú pháp được kết hợp với các khối và video các lát cho việc sử dụng bởi bộ giải mã video 30 trong việc giải mã các khối ảnh của lát video. Ngoài ra hoặc như sự thay thế cho các lát và các thành phần cú pháp tương ứng, các nhóm mảnh và/hoặc các mảnh tương ứng và các thành phần cú pháp tương ứng có thể được tạo ra hoặc được sử dụng.

Lập mã entropi

Bộ phận mã hóa entropi 270 được tạo cấu hình để áp dụng, ví dụ, sơ đồ hoặc thuật toán mã hóa entropi (ví dụ, sơ đồ lập mã độ dài biến đổi (variable length coding, viết tắt là VLC), sơ đồ VLC thích ứng ngữ cảnh (context adaptive VLC, viết tắt là CAVLC), sơ đồ lập mã số học, nhị phân hóa, lập mã số học nhị phân thích ứng ngữ cảnh (context adaptive binary arithmetic coding, viết tắt là CABAC), lập mã số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (syntax-based context adaptive binary arithmetic coding, viết tắt là SBAC), lập mã entropi phân chia khoảng xác suất (probability interval partitioning entropy, viết tắt là PIPE) hoặc kỹ thuật hoặc phương pháp luận mã hóa entropi khác) hoặc bỏ qua (không nén) trên các hệ số được lượng tử hóa 209, các thông số dự đoán liên kết, các thông số dự đoán bên trong, các thông số bộ lọc vòng lặp và/hoặc các thành phần cú pháp khác để thu nhận dữ liệu ảnh được mã hóa 21 mà có thể được đưa ra qua đầu ra 272, ví dụ, dưới dạng dòng bit được mã hóa 21, sao cho, ví dụ, bộ giải mã video 30 có thể thu và sử dụng các thông số cho việc giải mã. Dòng bit

được mã hóa 21 có thể được truyền tới bộ giải mã video 30, hoặc được lưu trữ trong bộ nhớ cho việc truyền hoặc truy xuất sau đó bởi bộ giải mã video 30.

Các biến thể cấu trúc khác của bộ mã hóa video 20 có thể được sử dụng để mã hóa dòng video. Ví dụ, bộ mã hóa trên cơ sở không biến đổi 20 có thể lượng tử hóa tín hiệu phần dư trực tiếp mà không bộ phận xử lý biến đổi 206 đối với các khối hoặc các khung nhất định. Theo cách thực hiện khác, bộ mã hóa 20 có thể có bộ phận lượng tử hóa 208 và bộ phận lượng tử hóa ngược 210 được kết hợp thành bộ phận đơn.

Bộ giải mã và phương pháp giải mã

Fig.3 thể hiện ví dụ về bộ giải mã video 30 mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Bộ giải mã video 30 được tạo cấu hình để thu dữ liệu ảnh được mã hóa 21 (ví dụ, dòng bit được mã hóa 21), ví dụ, được mã hóa bởi bộ mã hóa 20, để thu nhận ảnh được giải mã 331. Dữ liệu ảnh được mã hóa hoặc dòng bit bao gồm thông tin dùng cho việc giải mã dữ liệu ảnh được mã hóa, ví dụ, dữ liệu mà biểu diễn các khối ảnh của lát video được mã hóa (và/hoặc các nhóm mảnh hoặc các mảnh) và các thành phần cú pháp được kết hợp.

Trong ví dụ trên Fig.3, bộ giải mã 30 bao gồm bộ phận giải mã entropi 304, bộ phận lượng tử hóa ngược 310, bộ phận xử lý biến đổi ngược 312, bộ phận khôi phục 314 (ví dụ, bộ lấy tổng 314), bộ lọc vòng lặp 320, bộ đệm ảnh được giải mã (DBP) 330, bộ phận ứng dụng chế độ 360, bộ phận dự đoán liên kết 344 và bộ phận dự đoán bên trong 354. Bộ phận dự đoán liên kết 344 có thể là hoặc bao gồm bộ phận bù chuyển động. Bộ giải mã video 30 có thể, trong một số ví dụ, thực hiện tiến trình giải mã nói chung ngược lại với tiến trình mã hóa được mô tả đối với bộ mã hóa video 100 từ Fig.2.

Như được giải thích liên quan đến bộ mã hóa 20, bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận khôi phục 214, bộ lọc vòng lặp 220, bộ đệm ảnh được giải mã (DPB) 230, bộ phận dự đoán liên kết 344 và bộ phận dự đoán bên trong 354 cũng được đề cập đến là tạo nên “bộ giải mã cài sẵn” của bộ mã hóa video 20. Theo đó, bộ phận lượng tử hóa ngược 310 có thể giống hệt về chức năng với bộ phận lượng tử hóa ngược 110, bộ phận xử lý biến đổi ngược 312 có thể giống hệt về chức năng với bộ phận xử lý biến đổi ngược 212, bộ phận khôi phục 314 có thể giống hệt về chức năng với bộ phận khôi phục 214, bộ lọc vòng lặp 320 có thể giống hệt về

chức năng với bộ lọc vòng lặp 220, và bộ đệm ảnh được giải mã 330 có thể giống hệt về chức năng với bộ đệm ảnh được giải mã 230. Do đó, các phần giải thích được đưa ra đối với các bộ phận và các chức năng tương ứng của bộ mã hóa video 20 áp dụng tương ứng với các bộ phận và các chức năng tương ứng của bộ giải mã video 30.

Giải mã entropi

Bộ phận giải mã entropi 304 được tạo cấu hình để phân tách dòng bit 21 (hoặc nói chung là dữ liệu ảnh được mã hóa 21) và thực hiện, ví dụ, giải mã entropi tới dữ liệu ảnh được mã hóa 21 để thu nhận, ví dụ, các hệ số được lượng tử hóa 309 và/hoặc các thông số lập mã được giải mã (không được thể hiện trên Fig.3), ví dụ, bất kỳ hoặc tất cả trong số các thông số dự đoán liên kết (ví dụ, chỉ số ảnh tham chiếu và vectơ chuyển động), thông số dự đoán bên trong (ví dụ, chỉ số hoặc chế độ dự đoán bên trong), các thông số biến đổi, các thông số lượng tử hóa, các thông số bộ lọc vòng lặp, và/hoặc các thành phần cú pháp khác. Bộ phận giải mã entropi 304 có thể được tạo cấu hình để áp dụng các sơ đồ hoặc các thuật toán giải mã tương ứng với các sơ đồ mã hóa như được mô tả liên quan đến bộ phận mã hóa entropi 270 của bộ mã hóa 20. Bộ phận giải mã entropi 304 có thể còn được tạo cấu hình để cung cấp các thông số dự đoán liên kết, thông số dự đoán bên trong và/hoặc các thành phần cú pháp khác tới bộ phận ứng dụng chế độ 360 và các thông số khác tới các bộ phận khác của bộ giải mã 30. Bộ giải mã video 30 có thể thu các thành phần cú pháp ở mức lát video và/hoặc mức khối video. Ngoài ra hoặc như sự thay thế cho các lát và các thành phần cú pháp tương ứng, các nhóm mảnh và/hoặc các mảnh và các thành phần cú pháp tương ứng có thể được thu và/hoặc được sử dụng.

Lượng tử hóa ngược

Bộ phận lượng tử hóa ngược 310 có thể được tạo cấu hình để thu các thông số lượng tử hóa (QP) (hoặc nói chung là thông tin liên quan đến lượng tử hóa ngược) và các hệ số được lượng tử hóa từ dữ liệu ảnh được mã hóa 21 (ví dụ, bằng cách phân tách và/hoặc giải mã, ví dụ, bởi bộ phận giải mã entropi 304) và áp dụng dựa vào các thông số lượng tử hóa lượng tử hóa ngược trên các hệ số được lượng tử hóa được giải mã 309 để thu nhận các hệ số được giải lượng tử hóa 311, mà có thể cũng được gọi là các hệ số biến đổi 311. Quy trình lượng tử hóa ngược có thể bao gồm việc sử dụng của thông số lượng tử hóa được xác định bởi bộ mã hóa video 20 đối với mỗi khối video trong lát video

(hoặc mảnh hoặc nhóm mảnh) để xác định mức độ lượng tử hóa và, cũng vậy, mức độ lượng tử hóa ngược mà sẽ được áp dụng.

Biến đổi ngược

Bộ phận xử lý biến đổi ngược 312 có thể được tạo cấu hình để thu các hệ số được giải lượng tử hóa 311, cũng được đề cập đến là các hệ số biến đổi 311, và áp dụng việc biến đổi tới các hệ số được giải lượng tử hóa 311 để thu nhận các khối dư được khôi phục 213 trong miền mẫu. Các khối dư được khôi phục 213 có thể cũng được gọi là các khối biến đổi 313. Việc biến đổi có thể là biến đổi ngược, ví dụ, DCT ngược, DST ngược, biến đổi số nguyên ngược, hoặc tương tự về mặt khái niệm với quy trình biến đổi ngược. Bộ phận xử lý biến đổi ngược 312 có thể còn được tạo cấu hình để thu các thông số biến đổi hoặc thông tin tương ứng từ dữ liệu ảnh được mã hóa 21 (ví dụ, bằng cách phân tách và/hoặc giải mã, ví dụ, bởi bộ phận giải mã entropi 304) để xác định việc biến đổi được áp dụng tới các hệ số được giải lượng tử hóa 311.

Khôi phục

Bộ phận khôi phục 314 (ví dụ, bộ cộng hoặc bộ lấy tổng 314) có thể được tạo cấu hình để cộng khối dư được khôi phục 313, tới khối dự đoán 365 để thu nhận khối được khôi phục 315 trong miền mẫu, ví dụ, bằng cách cộng các trị số mẫu của khối dư được khôi phục 313 và các trị số mẫu của khối dự đoán 365.

Lọc

Bộ phận lọc vòng lặp 320 (hoặc trong vòng lặp lặp mã hoặc sau vòng lặp lặp mã) được tạo cấu hình để lọc khối được khôi phục 315 để thu nhận khối được lọc 321, ví dụ, để làm nhẵn các sự chuyển đổi điểm ảnh, hoặc mặt khác nâng cao chất lượng video. Bộ phận lọc vòng lặp 320 có thể bao gồm một hoặc nhiều bộ lọc vòng lặp chẳng hạn như bộ lọc tách khối, bộ lọc dịch vị thích ứng mẫu (sample-adaptive offset, viết tắt là SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc vòng lặp thích ứng (ALF), bộ lọc triệt nhiễu (noise suppression filter, viết tắt là NSF), hoặc sự kết hợp bất kỳ của nó. Theo ví dụ, bộ phận lọc vòng lặp 220 có thể bao gồm bộ lọc tách khối, bộ lọc SAO và bộ lọc ALF. Thứ tự của quy trình lọc có thể là lọc tách khối, SAO và ALF. Theo ví dụ khác, quy trình xử lý được gọi là ánh xạ độ chói với việc mở rộng sắc độ (luma mapping with chroma scaling, viết tắt là LMCS) (chẳng hạn như, bộ tạo lại hình dạng trong vòng lặp

thích ứng (adaptive in-loop reshaper)) được bổ sung. Quy trình xử lý được thực hiện trước khi tách khối. Theo ví dụ khác, quy trình lọc tách khối có thể cũng được áp dụng tới các mép khối con bên trong, ví dụ, các mép các khối con afin, các mép các khối con ATMVP, các mép biến đổi khối con (sub-block transform, viết tắt là SBT) và các mép phân chia con bên trong (intra sub-partition, viết tắt là ISP). Mặc dù bộ phận lọc vòng lặp 320 được thể hiện trên Fig.3 là bộ lọc trong vòng lặp, theo các cấu hình khác, bộ phận lọc vòng lặp 320 có thể được thực hiện như bộ lọc vòng lặp sau.

Bộ đệm ảnh được giải mã

Các khối video được giải mã 321 của ảnh sau đó được lưu trữ trong bộ đệm ảnh được giải mã 330, mà lưu trữ các ảnh được giải mã 331 như các ảnh tham chiếu cho việc bù chuyển động tiếp theo đối với các ảnh khác và/hoặc cho việc đưa ra màn hình tương ứng.

Bộ giải mã 30 được tạo cấu hình để đưa ra ảnh được giải mã 331, ví dụ, qua đầu ra 332, cho việc biểu diễn hoặc xem tới người dùng.

Dự đoán

Bộ phận dự đoán liên kết 344 có thể là giống hệt với bộ phận dự đoán liên kết 244 (cụ thể là tới bộ phận bù chuyển động) và bộ phận dự đoán bên trong 354 có thể là giống hệt với bộ phận dự đoán liên kết 254 về chức năng, và thực hiện các quyết định phân tách hoặc phân chia và dự đoán dựa vào các thông số dự đoán và/hoặc phân chia hoặc thông tin tương ứng được thu từ dữ liệu ảnh được mã hóa 21 (ví dụ, bằng cách phân tách và/hoặc giải mã, ví dụ, bởi bộ phận giải mã entropi 304). Bộ phận ứng dụng chế độ 360 có thể được tạo cấu hình để thực hiện việc dự đoán (dự đoán liên kết hoặc bên trong) trên mỗi khối dựa vào các ảnh được khôi phục, các khối hoặc các mẫu tương ứng (được lọc hoặc chưa được lọc) để thu nhận khối dự đoán 365.

Khi lát video được lập mã như lát được lập mã bên trong (I), bộ phận dự đoán bên trong 354 của bộ phận ứng dụng chế độ 360 được tạo cấu hình để tạo ra khối dự đoán 365 đối với khối ảnh của lát video hiện tại dựa vào dữ liệu và chế độ dự đoán bên trong được báo hiệu từ các khối được giải mã trước đó của ảnh hiện tại. Khi ảnh video được lập mã như lát được lập mã liên kết (nghĩa là, B, hoặc P), bộ phận dự đoán liên kết 344 (ví dụ, bộ phận bù chuyển động) của bộ phận ứng dụng chế độ 360 được tạo cấu hình

để tạo ra các khối dự đoán 365 đối với khối video của lát video hiện tại dựa vào các vector chuyển động và các thành phần cú pháp khác được thu từ bộ phận giải mã entropi 304. Đối với dự đoán liên kết, các khối dự đoán có thể được tạo ra từ một trong số các ảnh tham chiếu nằm trong một trong số các danh mục ảnh tham chiếu. Bộ giải mã video 30 có thể xây dựng các danh mục khung tham chiếu, danh mục 0 (List 0) và danh mục 1 (List 1), nhờ sử dụng các kỹ thuật xây dựng mặc định dựa vào các ảnh tham chiếu được lưu trữ trong DPB 330. Điều giống nhau hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án nhờ sử dụng các nhóm mảnh (ví dụ, các nhóm mảnh video) và/hoặc các mảnh (ví dụ, các mảnh video) bổ sung hoặc thay thế cho các lát (ví dụ, các lát video), ví dụ, video có thể được lập mã nhờ sử dụng các mảnh và/hoặc các nhóm mảnh I, P hoặc B.

Bộ phận ứng dụng chế độ 360 được tạo cấu hình để xác định thông tin dự đoán đối với khối video của lát video hiện tại bằng cách phân tách các vector chuyển động hoặc thông tin liên quan và các thành phần cú pháp khác, và sử dụng thông tin dự đoán để tạo ra các khối dự đoán đối với khối video hiện tại được giải mã. Ví dụ, bộ phận ứng dụng chế độ 360 sử dụng một số trong số các thành phần cú pháp được thu để xác định chế độ dự đoán (ví dụ, dự đoán liên kết hoặc bên trong) được sử dụng để lập mã các khối video của lát video, loại lát dự đoán liên kết (ví dụ, lát B, lát P, hoặc lát GPB), thông tin xây dựng đối với một hoặc nhiều danh mục ảnh tham chiếu cho lát, các vector chuyển động đối với mỗi khối video được mã hóa liên kết của lát, trạng thái dự đoán liên kết đối với mỗi khối video được lập mã liên kết của lát, và thông tin khác để giải mã các khối video trong lát video hiện tại. Điều giống nhau hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án nhờ sử dụng các nhóm mảnh (ví dụ, các nhóm mảnh video) và/hoặc các mảnh (ví dụ, các mảnh video) bổ sung hoặc thay thế cho các lát (ví dụ, các lát video), ví dụ, video có thể được lập mã nhờ sử dụng các mảnh và/hoặc các nhóm mảnh I, P hoặc B.

Các phương án về bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân chia và/hoặc giải mã ảnh nhờ sử dụng các lát (cũng được đề cập đến là các lát video), trong đó ảnh có thể được phân chia thành hoặc được giải mã nhờ sử dụng một hoặc nhiều lát (cụ thể là không chồng lấn), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều nhóm của các khối (ví dụ, các

mảnh (H.265/HEVC và VVC) hoặc các miếng (VVC)).

Các phương án về bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân chia và/hoặc giải mã ảnh nhờ sử dụng các lát/các nhóm mảnh (cũng được đề cập đến là các nhóm mảnh video) và/hoặc các mảnh (cũng được đề cập đến là các mảnh video), trong đó ảnh có thể được phân chia thành hoặc được giải mã nhờ sử dụng một hoặc nhiều lát/các nhóm mảnh (cụ thể là không chồng lấn), và mỗi nhóm mảnh/lát có thể bao gồm, ví dụ, một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều mảnh, trong đó mỗi mảnh, ví dụ, có thể có dạng hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ, các khối hoàn thiện hoặc một phần.

Các biến thể khác của bộ giải mã video 30 có thể được sử dụng để giải mã dữ liệu ảnh được mã hóa 21. Ví dụ, bộ giải mã 30 có thể tạo ra dòng video đầu ra mà không có bộ phận lọc vòng lặp 320. Ví dụ, bộ giải mã trên cơ sở không biến đổi 30 có thể lượng tử hóa ngược tín hiệu phần dư một cách trực tiếp mà không có bộ phận xử lý biến đổi ngược 312 đối với các khối hoặc các khung nhất định. Theo cách thực hiện khác, bộ giải mã video 30 có thể có bộ phận lượng tử hóa ngược 310 và bộ phận xử lý biến đổi ngược 312 được kết hợp thành bộ phận đơn.

Cần hiểu rằng, trong bộ mã hóa 20 và bộ giải mã 30, kết quả xử lý của bước hiện tại có thể được xử lý thêm và sau đó đưa ra tới bước tiếp theo. Ví dụ, sau khi lọc nội suy, dẫn ra vector chuyển động hoặc lọc vòng lặp, thao tác khác, chẳng hạn như ghim (Clip) hoặc dịch vị (shift), có thể được thực hiện trên kết quả xử lý của việc lọc nội suy, dẫn ra vector chuyển động hoặc lọc vòng lặp.

Cần lưu ý rằng các thao tác khác có thể được áp dụng tới các vector chuyển động được dẫn ra của khối hiện tại (bao gồm nhưng không giới hạn ở các vector chuyển động điểm điều khiển của chế độ afin, các vector chuyển động khối con trong các chế độ ATMVP, afin, mặt phẳng, các vector chuyển động theo thời gian, và v.v.). Ví dụ, trị số của vector chuyển động bị ràng buộc tới khoảng được định rõ trước theo bit biểu diễn của nó. Nếu bit biểu diễn của vector chuyển động là bitDepth, thì khoảng là $-2^{(bitDepth-1)} \sim 2^{(bitDepth-1)}-1$, trong đó “^” nghĩa là số mũ. Ví dụ, nếu bitDepth được thiết đặt bằng 16, khoảng là -32768 ~ 32767; nếu bitDepth được thiết đặt bằng 18, khoảng là -131072~131071. Ví dụ, trị số của vector chuyển động được dẫn ra (ví dụ, các MV của bốn khối con 4x4 nằm trong một khối 8x8) bị ràng buộc sao cho độ chênh lệch

lớn nhất giữa các phần số nguyên của bốn MV khối con 4x4 là không lớn hơn N điểm ảnh, chẳng hạn như không lớn hơn 1 điểm ảnh. Ở đây đưa ra hai phương pháp để ràng buộc vector chuyển động theo bitDepth.

Fig.4 là hình vẽ giản lược của thiết bị lập mã video 400 theo phương án của sáng chế. Thiết bị lập mã video 400 là thích hợp để thực hiện các phương án được bộc lộ như được mô tả ở đây. Trong phương án, thiết bị lập mã video 400 có thể là bộ giải mã chẳng hạn như bộ giải mã video 30 trên Fig.1A hoặc bộ mã hóa chẳng hạn như bộ mã hóa video 20 trên Fig.1A.

Thiết bị lập mã video 400 bao gồm các cổng vào 410 (hoặc các cổng đầu vào 410) và các bộ phận bộ thu (Rx) 420 để thu dữ liệu; bộ xử lý, bộ phận logic, hoặc bộ phận xử lý trung tâm (central processing unit, viết tắt là CPU) 430 để xử lý dữ liệu; các bộ phận bộ truyền (Tx) 440 và các cổng ra 450 (hoặc các cổng đầu ra 450) dùng để truyền dữ liệu; và bộ nhớ 460 để lưu trữ dữ liệu. Thiết bị lập mã video 400 có thể cũng bao gồm các thành phần quang thành điện (optical-to-electrical, viết tắt là OE) và các thành phần điện thành quang (electrical-to-optical, viết tắt là EO) được ghép nối với các cổng vào 410, các bộ phận bộ thu 420, các bộ phận bộ truyền 440, và các cổng ra 450 để đưa ra hoặc đưa vào các tín hiệu quang hoặc điện.

Bộ xử lý 430 được thực hiện bởi phần cứng và phần mềm. Bộ xử lý 430 có thể được thực hiện như một hoặc nhiều chip CPU, các lõi (ví dụ, như bộ xử lý đa lõi), các FPGA, các ASIC, và các DSP. Bộ xử lý 430 truyền thông với các cổng vào 410, các bộ phận bộ thu 420, các bộ phận bộ truyền 440, các cổng ra 450, và bộ nhớ 460. Bộ xử lý 430 bao gồm môđun lập mã 470. Môđun lập mã 470 thực hiện các phương án được bộc lộ nêu trên. Chẳng hạn, môđun lập mã 470 thực hiện việc xử lý, chuẩn bị, hoặc cung cấp các thao tác lập mã khác nhau. Sự bao gồm của môđun lập mã 470 do đó đưa ra sự cải tiến đáng kể đối với chức năng của thiết bị lập mã video 400 và đem lại sự biến đổi của thiết bị lập mã video 400 tới trạng thái khác nhau. Theo cách khác, môđun lập mã 470 được thực hiện như các lệnh được lưu trữ trong bộ nhớ 460 và được thực hiện bởi bộ xử lý 430.

Bộ nhớ 460 có thể bao gồm một hoặc nhiều đĩa, các ổ băng, và các ổ đĩa trạng thái rắn và có thể được sử dụng như thiết bị lưu trữ dữ liệu tràn, để lưu trữ các chương trình khi các chương trình như vậy được lựa chọn cho việc thực hiện, và lưu trữ các lệnh và

dữ liệu mà được đọc trong khi thực hiện chương trình. Bộ nhớ 460 có thể là, ví dụ, khả biến và/hoặc bất khả biến và có thể là bộ nhớ chỉ đọc (read-only memory, viết tắt là ROM), bộ nhớ truy cập ngẫu nhiên (random access memory, viết tắt là RAM), bộ nhớ nội dung địa chỉ bậc ba (ternary content-addressable memory, viết tắt là TCAM), và/hoặc bộ nhớ truy cập ngẫu nhiên tĩnh (static random-access memory, viết tắt là SRAM).

Fig.5 là sơ đồ khối được giản lược của máy 500 mà có thể được sử dụng như một hoặc cả trong số thiết bị nguồn 12 và thiết bị đích 14 từ Fig.1 theo phương án lấy làm ví dụ.

Bộ xử lý 502 trong máy 500 có thể là bộ phận xử lý trung tâm. Theo cách khác, bộ xử lý 502 có thể là loại bất kỳ khác của thiết bị, hoặc nhiều thiết bị, có khả năng thao tác hoặc xử lý thông tin hiện có hoặc được phát triển về sau. Mặc dù các cách thực hiện được bộc lộ có thể được thực hiện với bộ xử lý đơn như được thể hiện, ví dụ, bộ xử lý 502, các ưu điểm về tốc độ và hiệu quả có thể là đạt được nhờ sử dụng nhiều hơn một bộ xử lý.

Bộ nhớ 504 trong máy 500 có thể là thiết bị bộ nhớ chỉ đọc (read only memory, viết tắt là ROM) hoặc thiết bị bộ nhớ truy cập ngẫu nhiên (random access memory, viết tắt là RAM) theo cách thực hiện. Loại thích hợp bất kỳ khác của thiết bị lưu trữ có thể được sử dụng như bộ nhớ 504. Bộ nhớ 504 có thể bao gồm mã và dữ liệu 506 mà được truy cập bởi bộ xử lý 502 nhờ sử dụng bus 512. Bộ nhớ 504 có thể còn bao gồm hệ điều hành 508 và các chương trình ứng dụng 510, các chương trình ứng dụng 510 bao gồm ít nhất một chương trình mà cho phép bộ xử lý 502 thực hiện các phương pháp được mô tả ở đây. Ví dụ, các chương trình ứng dụng 510 có thể bao gồm các ứng dụng từ 1 đến N, mà còn bao gồm ứng dụng lập mã video mà thực hiện các phương pháp được mô tả ở đây.

Máy 500 có thể cũng bao gồm một hoặc nhiều thiết bị đầu ra, chẳng hạn như màn hình 518. Màn hình 518 có thể là, theo một ví dụ, màn hình cảm ứng chạm (touch sensitive display) mà kết hợp màn hình với thành phần cảm ứng chạm mà hoạt động được để cảm nhận các đầu vào chạm. Màn hình 518 có thể được ghép nối với bộ xử lý 502 qua bus 512.

Mặc dù được mô tả ở đây là bus đơn, bus 512 của máy 500 có thể được bao gồm nhiều bus. Hơn nữa, bộ phận lưu trữ thứ cấp 514 có thể được ghép nối trực tiếp với các thành phần khác của máy 500 hoặc có thể được truy cập qua mạng và có thể bao gồm bộ phận tích hợp đơn chẳng hạn như thẻ nhớ hoặc nhiều bộ phận chẳng hạn như nhiều thẻ nhớ. Máy 500 do đó có thể được thực hiện theo nhiều loại cấu hình.

Các phương án của sáng chế cho phép thực hiện hai hoặc nhiều hơn hai sơ đồ quản lý ảnh tham chiếu (ví dụ, phương pháp cửa sổ trượt ngầm (ví dụ, như được sử dụng trong AVC) và phương pháp quản lý danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) (ví dụ, VVC) bên trong một bộ mã hóa/giải mã.

Theo phương án thứ nhất của sáng chế, phương pháp cửa sổ trượt ngầm và các phương pháp các danh mục ảnh tham chiếu (RPL) được kết hợp trong một bộ mã hóa/giải mã và phương pháp bất kỳ trong số hai phương pháp này có thể được lựa chọn, ví dụ, cho việc sử dụng ở mức chuỗi (SPS).

Bảng dưới đây đưa ra đoạn trích của bảng cú pháp SPS định rõ việc chuyển đổi phương pháp quản lý ảnh tham chiếu.

seq_parameter_set_rbsp() {	Ký hiệu mô tả
...	
sps_rpl_flag	u(1)
...	
}	

sps_rpl_flag bằng 1 định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt và phương pháp RPL được sử dụng; sps_rpl_flag bằng 0 định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt và AVC phương pháp được sử dụng.

Bảng dưới đây đưa ra ví dụ về việc báo hiệu cú pháp các ảnh tham chiếu liên quan đến các phương pháp AVC hoặc RPL.

seq_parameter_set_rbsp() {	Ký hiệu mô tả
-----------------------------	---------------

...	
if(!sps_rpl_flag)	
max_num_tid0_ref_pics	ue(v)
else {	
long_term_ref_pics_flag	u(1)
rpl1_same_as_rpl0_flag	u(1)
for(i = 0; i < !rpl1_same_as_rpl0_flag ? 2 : 1; i++) {	
num_ref_pic_lists_in_sps[i]	ue(v)
for(j = 0; j < num_ref_pic_lists_in_sps[i]; j++)	
ref_pic_list_struct(i, j, long_term_ref_pics_flag)	
}	
...	
}	
picture_cropping_flag	u(1)
if(picture_cropping_flag) {	

Thông số cú pháp (hoặc cờ) `sps_rpl_flag`, ví dụ, được gửi ở mức SPS (trong phương án khác nó có thể được gửi ở PPS hoặc đoạn đầu lát) và chỉ báo xem RPL sẽ được sử dụng hay không cho việc quản lý ảnh tham chiếu (trị số cờ “1” chỉ báo RPL được sử dụng, trị số cờ “0” chỉ báo rằng RPL không được sử dụng mà là cơ chế cửa sổ trượt ngầm).

Nếu phương pháp RPL không được phép (`sps_rpl_flag` bằng 0), chỉ `max_num_tid0_ref_pics` được báo hiệu đối với phương pháp quản lý ảnh tham chiếu cửa sổ trượt ngầm.

`max_num_tid0_ref_pics` định rõ số lớn nhất của các ảnh tham chiếu trong lớp thời gian 0 mà có thể được sử dụng bởi quy trình giải mã cho việc dự đoán liên kết (cửa sổ trượt ngầm việc quản lý ảnh tham chiếu phương pháp) của ảnh bất kỳ trong chuỗi video lập mã (Coding Video sequence, viết tắt là CVS). Trị số của `max_num_tid0_ref_pics` cũng được sử dụng để xác định kích thước của bộ đệm ảnh được giải mã. Trị số của

`max_num_tid0_ref_pics` cụ thể là trong khoảng từ 0 đến 5. Khi không có mặt, trị số của `max_num_tid0_ref_pics` được suy ra, ví dụ, là bằng 0.

`long_term_ref_pics_flag` bằng 0 định rõ rằng không có LTRP được sử dụng cho việc dự đoán liên kết của ảnh được lập mã bất kỳ trong CVS. `Long_term_ref_pics_flag` bằng 1 định rõ rằng LTRPs có thể được sử dụng cho việc dự đoán liên kết của một hoặc nhiều ảnh được lập mã trong CVS. Khi không có mặt, trị số của `long_term_ref_pics_flag` được suy ra là bằng 0.

`rpl1_same_as_rpl0_flag` bằng 1 định rõ rằng các cấu trúc cú pháp `num_ref_pic_lists_in_sps[ 1 ]` và `ref_pic_list_struct( 1, rplsIdx, ltrpFlag )` là không có mặt và sau đây áp dụng:

— Trị số của `num_ref_pic_lists_in_sps[ 1 ]` được suy ra là bằng trị số của `num_ref_pic_lists_in_sps[ 0 ]`.

— Trị số của mỗi của các thành phần cú pháp trong `ref_pic_list_struct( 1, rplsIdx, ltrpFlag )` được suy ra là bằng trị số của thành phần cú pháp tương ứng trong `ref_pic_list_struct( 0, rplsIdx, ltrpFlag )` đối với `rplsIdx` nằm trong khoảng từ 0 đến `num_ref_pic_lists_in_sps[ 0 ] - 1`.

`num_ref_pic_lists_in_sps[ i ]` định rõ số của các cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx, ltrpFlag )` với `listIdx` bằng `i` được bao gồm trong SPS. Trị số của `num_ref_pic_lists_in_sps[ i ]` sẽ nằm trong khoảng từ 0 đến 64. Khi không có mặt, trị số của `num_ref_pic_lists_in_sps[ i ]` được suy ra là bằng 0.

Lưu ý: Đối với mỗi trị số của `listIdx` (bằng 0 hoặc 1), bộ giải mã phải cấp phát bộ nhớ cho tổng số của các cấu trúc cú pháp `num_ref_pic_lists_in_sps[ i ] + 1` `ref_pic_list_struct( listIdx, rplsIdx, ltrpFlag )` vì có thể có một cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx, ltrpFlag )` được báo hiệu trực tiếp trong các đoạn đầu lát của ảnh hiện tại.

Nếu phương pháp RPL được phép (sps_rpl_flag bằng 1), cú pháp RPL sau đây được báo hiệu ở mức lát.

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 &&	
rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	
...	
}	

Theo bảng cú pháp đoạn đầu lát, có thành phần cú pháp (ví dụ, ref_pic_list_sps_flag[i]) định rõ xem danh mục ảnh tham chiếu được báo hiệu trong đoạn đầu lát hoặc một số của các danh mục ảnh tham chiếu mà đã được báo hiệu ở mức SPS hay không, được sử dụng trong lát hiện tại.

Theo phương án hiện tại, ref_pic_list_sps_flag[i] luôn được báo hiệu trong đoạn đầu lát đối với list0 và cũng được báo hiệu đối với list1 nếu chỉ rpl1_idx_present_flag bằng 1.

Phương án khác tương tự với phương án thứ nhất, nếu phương pháp RPL được phép (sps_rpl_flag bằng 1), cú pháp RPL sau đây được báo hiệu ở mức lát.

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

Theo phương án này, việc báo hiệu của thành phần ref_pic_list_sps_flag[i] không được thực hiện nếu số của các danh mục RPL bên trong SPS là bằng 0.

ref_pic_list_sps_flag[i] bằng 1 định rõ rằng danh mục ảnh tham chiếu i của ảnh hiện tại được dẫn ra dựa vào một trong số các cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx, ltrpFlag) với listIdx bằng i trong SPS hoạt động. ref_pic_list_sps_flag[i] bằng 0 định rõ rằng danh mục ảnh tham chiếu i của ảnh hiện tại được dẫn ra dựa vào cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx, ltrpFlag)

với listIdx bằng i mà được bao gồm trực tiếp trong các đoạn đầu lát của ảnh hiện tại. Khi num_ref_pic_lists_in_sps[i] bằng 0, trị số của ref_pic_list_sps_flag[i] sẽ bằng 0. Khi rpl1_idx_present_flag bằng 0 và ref_pic_list_sps_flag[0] là có mặt, trị số của ref_pic_list_sps_flag[1] được suy ra là bằng ref_pic_list_sps_flag[0]. Khi không có mặt, trị số của rep_pic_list_sps_flag[i] được suy ra là bằng 0.

ref_pic_list_idx[i] định rõ chỉ số, trong danh mục của các cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx, ltrpFlag) với listIdx bằng i được bao gồm trong SPS hoạt động, của cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx, ltrpFlag) với listIdx bằng i mà được sử dụng cho việc dẫn ra của danh mục ảnh tham chiếu i của ảnh hiện tại. Thành phần cú pháp ref_pic_list_idx[i] được biểu diễn bởi $\text{Ceil}(\text{Log2}(\text{num_ref_pic_lists_in_sps}[i]))$ các bit. Khi không có mặt, trị số của ref_pic_list_idx[i] được suy ra là bằng 0. Trị số của ref_pic_list_idx[i] sẽ nằm trong khoảng từ 0 đến num_ref_pic_lists_in_sps[i] - 1. Khi rpl1_idx_present_flag bằng 0 và ref_pic_list_sps_flag[0] là có mặt, trị số của ref_pic_list_idx[1] được suy ra là bằng ref_pic_list_idx[0].

Phần dưới đây là giải thích về các ứng dụng của phương pháp mã hóa cũng như phương pháp giải mã như được thể hiện theo các phương án nêu trên, và hệ thống sử dụng chúng.

Fig.6 là sơ đồ khối thể hiện hệ thống cung cấp nội dung 3100 để thực hiện dịch vụ phân phối nội dung. Hệ thống cung cấp nội dung 3100 này bao gồm thiết bị chụp 3102, thiết bị đầu cuối 3106, và một cách tùy chọn bao gồm màn hình 3126. Thiết bị chụp 3102 truyền thông với thiết bị đầu cuối 3106 qua liên kết truyền thông 3104. Liên kết truyền thông có thể bao gồm kênh truyền thông 13 được nêu trên. Liên kết truyền thông 3104 bao gồm nhưng không giới hạn ở WIFI, Ethernet, cáp, (3G/4G/5G) không dây, USB, hoặc loại kết hợp bất kỳ của nó, hoặc tương tự.

Thiết bị chụp 3102 tạo dữ liệu, và có thể mã hóa dữ liệu bởi phương pháp mã hóa như được thể hiện theo các phương án nêu trên. Theo cách khác, thiết bị chụp 3102 có thể phân phối dữ liệu tới máy chủ tạo dòng (không được thể hiện trên các hình vẽ), và máy chủ mã hóa dữ liệu và truyền dữ liệu được mã hóa tới thiết bị đầu cuối 3106. Thiết bị chụp 3102 bao gồm nhưng không giới hạn ở camera, điện thoại thông minh hoặc máy tính bảng (Pad), máy tính hoặc máy tính xách tay, hệ thống hội nghị video, PDA, thiết

bị lắp trên xe, hoặc sự kết hợp bất kỳ của chúng, hoặc tương tự. Ví dụ, thiết bị chụp 3102 có thể bao gồm thiết bị nguồn 12 như được nêu trên. Khi dữ liệu bao gồm video, bộ mã hóa video 20 được bao gồm trong thiết bị chụp 3102 có thể thực hiện một cách thực tế việc xử lý mã hóa video. Khi dữ liệu bao gồm audio (nghĩa là, âm thanh), bộ mã hóa audio được bao gồm trong thiết bị chụp 3102 có thể thực hiện một cách thực tế việc xử lý mã hóa audio. Đối với một số kịch bản thực tế, thiết bị chụp 3102 phân phối di audio và video được mã hóa bằng cách đa hợp chúng cùng với nhau. Đối với kịch bản thực tế khác, ví dụ trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được đa hợp. Thiết bị chụp 3102 phân phối dữ liệu audio được mã hóa và dữ liệu video được mã hóa tới thiết bị đầu cuối 3106 một cách riêng biệt.

Trong hệ thống cung cấp nội dung 3100, thiết bị đầu cuối 310 thu và tái tạo dữ liệu được mã hóa. Thiết bị đầu cuối 3106 có thể là thiết bị có khả năng thu và khôi phục dữ liệu chẳng hạn như điện thoại thông minh hoặc máy tính bảng (Pad) 3108, máy tính hoặc máy tính xách tay 3110, bộ ghi video mạng (network video recorder, viết tắt là NVR)/ bộ ghi video số (digital video recorder, viết tắt là DVR) 3112, TV 3114, thiết bị giải mã tín hiệu truyền hình (set top box, viết tắt là STB) 3116, hệ thống hội nghị video 3118, hệ thống giám sát video 3120, thiết bị hỗ trợ số cá nhân (personal digital assistant, viết tắt là PDA) 3122, thiết bị lắp trên xe 3124, hoặc sự kết hợp bất kỳ của chúng, hoặc tương tự có khả năng giải mã dữ liệu được mã hóa nêu trên. Ví dụ, thiết bị đầu cuối 3106 có thể bao gồm thiết bị đích 14 như được nêu trên. Khi dữ liệu được mã hóa bao gồm video, bộ giải mã video 30 được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện giải mã video. Khi dữ liệu được mã hóa bao gồm audio, bộ giải mã audio được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện việc xử lý giải mã audio.

Đối với thiết bị đầu cuối có màn hình của nó, ví dụ, điện thoại thông minh hoặc máy tính bảng (Pad) 3108, máy tính hoặc máy tính xách tay 3110, bộ ghi video mạng (NVR)/ bộ ghi video số (DVR) 3112, TV 3114, thiết bị hỗ trợ số cá nhân (PDA) 3122, hoặc thiết bị lắp trên xe 3124, thiết bị đầu cuối có thể cấp dữ liệu được giải mã tới màn hình của nó. Đối với thiết bị đầu cuối được trang bị không có màn hình, ví dụ, STB 3116, hệ thống hội nghị video 3118, hoặc hệ thống giám sát video 3120, màn hình bên ngoài 3126 được lắp trong đó để thu và thể hiện dữ liệu được giải mã.

Khi mỗi thiết bị trong hệ thống này thực hiện việc mã hóa hoặc giải mã, thiết bị mã hóa ảnh hoặc thiết bị giải mã ảnh, như được thể hiện theo các phương án nêu trên, có thể được sử dụng.

Fig.7 là hình vẽ thể hiện cấu trúc của ví dụ về thiết bị đầu cuối 3106. Sau khi thiết bị đầu cuối 3106 thu dòng từ thiết bị chụp 3102, bộ phận tiến hành giao thức 3202 phân tích giao thức truyền của dòng. Giao thức bao gồm nhưng không giới hạn ở giao thức tạo dòng thời gian thực (Real Time Streaming Protocol, viết tắt là RTSP), giao thức truyền siêu văn bản (Hyper Text Transfer Protocol, viết tắt là HTTP), giao thức tạo dòng trực tiếp HTTP (HTTP Live streaming protocol, viết tắt là HLS), MPEG-DASH, giao thức vận chuyển thời gian thực (Real-time Transport protocol, viết tắt là RTP), giao thức thông báo thời gian thực (Real Time Messaging Protocol, viết tắt là RTMP), hoặc loại kết hợp bất kỳ của nó, hoặc tương tự.

Sau khi bộ phận tiến hành giao thức 3202 xử lý dòng, tệp dòng được tạo ra. Tệp được đưa ra tới bộ phận giải đa hợp 3204. Bộ phận giải đa hợp 3204 có thể tách dữ liệu được đa hợp thành dữ liệu audio được mã hóa và dữ liệu video được mã hóa. Như được nêu trên, đối với một số kịch bản thực tế, ví dụ trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được đa hợp. Trong tình huống này, dữ liệu được mã hóa được truyền tới bộ giải mã video 3206 và bộ giải mã audio 3208 mà không qua bộ phận giải đa hợp 3204.

Qua việc xử lý giải đa hợp, dòng sơ cấp (elementary stream, viết tắt là ES) video, ES audio, và phụ đề được tạo ra một cách tùy chọn. Bộ giải mã video 3206, mà bao gồm bộ giải mã video 30 như được giải thích theo các phương án nêu trên, giải mã ES video bởi phương pháp giải mã như được thể hiện theo các phương án nêu trên để tạo ra khung video, và cấp dữ liệu này tới bộ phận đồng bộ hóa 3212. Bộ giải mã audio 3208, giải mã ES audio để tạo ra khung audio, và cấp dữ liệu này tới bộ phận đồng bộ hóa 3212. Theo cách khác, khung video có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.28) trước khi cấp nó tới bộ phận đồng bộ hóa 3212. Tương tự, khung audio có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.28) trước khi cấp nó tới bộ phận đồng bộ hóa 3212.

Bộ phận đồng bộ hóa 3212 đồng bộ hóa khung video và khung audio, và cấp video/audio tới màn hình video/audio 3214. Ví dụ, bộ phận đồng bộ hóa 3212 đồng bộ

hóa sự biểu diễn của thông tin video và audio. Thông tin có thể lập mã theo cú pháp nhờ sử dụng các dấu thời gian về việc biểu diễn của dữ liệu audio và hình ảnh được lập mã và các dấu thời gian về việc phân phối của chính dòng dữ liệu.

Nếu phụ đề được bao gồm trong dòng, bộ giải mã phụ đề 3210 giải mã phụ đề, và đồng bộ hóa nó với khung video và khung audio, và cấp video/audio/phụ đề tới màn hình video/audio/phụ đề 3216.

Phù hợp với phần mô tả nêu trên, cụ thể, các phương án sau đây được đề xuất. Fig.8 minh họa phương pháp giải mã chuỗi video được mã hóa theo phương án. Phương pháp được minh họa trên Fig.8 bao gồm bước thu nhận 810 (ví dụ, bằng cách phân tách) trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) (ví dụ, `sps_rpl_flag`) từ dòng bit của chuỗi video. Thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không. Phương pháp, hơn nữa, bao gồm, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận 820, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu. Ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được xây dựng 830, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit.

Phương pháp có thể còn bao gồm, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit. Trong trường hợp này, sự xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit có thể bao gồm thực hiện phương pháp cửa sổ trượt để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

Nhờ đó, việc quản lý danh mục ảnh tham chiếu mà được dựa vào và không được dựa vào danh mục ảnh tham chiếu có thể được thực hiện. Nhờ thành phần cú pháp RPL (ví dụ, “`sps_rpl_flag`”), xem phương pháp các danh mục ảnh tham chiếu có thể được sử

dụng cho quy trình dự đoán liên kết hay không, và, nhờ đó, việc quản lý của quy trình dự đoán liên kết, cụ thể là, việc quản lý ảnh tham chiếu, có thể được điều khiển một cách hữu hiệu.

Các bước xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu và ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit, một cách tương ứng, có thể được thực hiện phù hợp với chuẩn EVC đã được biết đối với người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng. Dựa vào chuẩn đó, khi ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được xây dựng sau đây áp dụng cho quy trình xây dựng:

Các ảnh tham chiếu được định địa chỉ qua các chỉ số tham chiếu. Chỉ số tham chiếu là chỉ số trong danh mục ảnh tham chiếu. Khi giải mã lát I, không có danh mục ảnh tham chiếu được sử dụng trong việc giải mã của dữ liệu lát. Khi giải mã lát P, chỉ danh mục ảnh tham chiếu 0 (nghĩa là, `RefPicList[ 0 ]`), được sử dụng trong việc giải mã của dữ liệu lát. Khi giải mã lát B, cả danh mục ảnh tham chiếu 0 và danh mục ảnh tham chiếu 1 (nghĩa là, `RefPicList[ 1 ]`) được sử dụng trong việc giải mã của dữ liệu lát.

Ở đầu của quy trình giải mã đối với mỗi lát của ảnh không IDR, các danh mục ảnh tham chiếu `RefPicList[ 0 ]` và `RefPicList[ 1 ]` được dẫn ra. Các danh mục ảnh tham chiếu được sử dụng trong việc đánh dấu của các ảnh tham chiếu như được định rõ trong khoản 8.3.3 hoặc trong việc giải mã của dữ liệu lát.

Lưu ý 1 – Đối với lát I của ảnh không IDR mà không phải lát thứ nhất của ảnh, `RefPicList[ 0 ]` và `RefPicList[ 1 ]` có thể được dẫn ra đối với mục đích kiểm tra sự phù hợp của dòng bit, nhưng sự dẫn ra của chúng là không cần thiết đối với việc giải mã của ảnh hiện tại hoặc các ảnh theo sau ảnh hiện tại theo thứ tự giải mã. Đối với lát P mà không phải lát thứ nhất của ảnh, `RefPicList[ 1 ]` có thể được dẫn ra đối với mục đích kiểm tra sự phù hợp của dòng bit, nhưng sự dẫn ra của nó là không cần thiết đối với việc giải mã của ảnh hiện tại hoặc các ảnh theo sau ảnh hiện tại theo thứ tự giải mã.

Các danh mục ảnh tham chiếu RefPicList[0] và RefPicList[1] được xây dựng như sau:

```

for( i = 0; i < 2; i++ )
    for( j = 0, pocBase = PicOrderCntVal; j < NumEntriesInList[ i ][
SliceRplsIdx[ i ] ]; j++ ) {
        if( !lt_ref_pic_flag[ i ][ SliceRplsIdx[ i ] ][ j ] ) {
            RefPicPocList[ i ][ j ] = pocBase - DeltaPocSt[ i ][ SliceRplsIdx[ i ] ][ j ]
            if( there is reference picture picA in DPB with PicOrderCntVal equal to
                RefPicPocList[ i ][ j ] )
                RefPicList[ i ][ j ] = picA
            else
                RefPicList[ i ][ j ] = “no reference picture”
            pocBase =
RefPicPocList[ i ][ j ]          (166)
        } else {
            if( there is reference picA in DPB with
PicOrderCntVal & ( MaxLtPicOrderCntLsb - 1 )
                equal to FullPocLsbLt[ i ][ SliceRplsIdx[ i ] ][ j ] )
                RefPicList[ i ][ j ] = picA
            else
                RefPicList[ i ][ j ] = “no reference picture”
        }
    }
}

```

Đối với mỗi i bằng 0 hoặc 1, sau đây áp dụng:

— Các mục nhập NumRefIdxActive[i] thứ nhất trong RefPicList[i] được tham chiếu tới như các mục nhập hoạt động trong RefPicList[i], và các mục nhập khác trong RefPicList[i] được tham chiếu tới như các mục nhập không hoạt động trong RefPicList[i].

— Mỗi mục nhập trong RefPicList[i][j] đối với j trong khoảng từ 0 đến NumEntriesInList[i][SliceRplsIdx[i]] - 1, được tham chiếu tới như mục nhập STRP

nếu `lt_ref_pic_flag[ i ][ SliceRplIdx[ i ] ][ j ]` bằng 0, và như mục nhập LTRP theo cách khác.”

Với việc tiếp tục tham chiếu tới chuẩn đó, khi ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết không sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được xây dựng sau đây áp dụng cho quy trình xây dựng:

“

1) Quy trình giải mã để điền danh mục ảnh tham chiếu với `PicOrderCntVal` thấp hơn trong khoản 8.3.2.2.2 được dẫn ra với `i` được thiết đặt bằng 0 và `startIdx` set bằng 0, và đầu ra là biến `nextIdx`.

2) Khi `nextIdx` là nhỏ hơn `NumRefIdxActive[ 0 ]`, quy trình giải mã để điền danh mục ảnh tham chiếu với `PicOrderCntVal` cao hơn trong khoản 8.3.2.2.3 được dẫn ra với `i` được thiết đặt bằng 0 và `startIdx` thiết đặt bằng `nextIdx`, và đầu ra là biến `nextIdx`.

3) Khi `nextIdx` là nhỏ hơn `NumRefIdxActive[ 0 ]`, `NumRefIdxActive[ 0 ]` được thiết đặt bằng `nextIdx`.

Đối với các lát B, danh mục ảnh tham chiếu `RefPicList[ 1 ]` được xây dựng như sau:

1) Quy trình giải mã để điền danh mục ảnh tham chiếu với `PicOrderCntVal` cao hơn trong khoản 8.3.2.2.3 được dẫn ra với `i` được thiết đặt bằng 1 và `startIdx` set bằng 0, và đầu ra là biến `nextIdx`.

2) Khi `nextIdx` là nhỏ hơn `NumRefIdxActive[ 1 ]`, quy trình giải mã để điền danh mục ảnh tham chiếu với `PicOrderCntVal` thấp hơn trong khoản 8.3.2.2.2 được dẫn ra với `i` được thiết đặt bằng 1 và `startIdx` được thiết đặt bằng `nextIdx`, và đầu ra là biến `nextIdx`.

3) Khi `nextIdx` là nhỏ hơn `NumRefIdxActive[ 1 ]`, `NumRefIdxActive[ 1 ]` được thiết đặt bằng `nextIdx`.”

8.3.2.2.2 Quy trình giải mã để điền danh mục ảnh tham chiếu với các ảnh `PicOrderCntVal` thấp hơn

Các đầu vào tới quy trình này là:

- ký hiệu nhận dạng danh mục ảnh tham chiếu i , và
- vị trí chỉ số bắt đầu $startIdx$.

Đầu ra của quy trình này là biến $nextIdx$ biểu diễn số của các vị trí được điền trong danh mục ảnh tham chiếu.

biến $nextIdx$ được thiết đặt bằng $startIdx$.

biến $nextTemporalId$ được thiết đặt bằng $\text{Max}(\text{TemporalId} - 1, 0)$.

Để $minPoc$ được thiết đặt bằng trị số thấp nhất của $\text{PictureOrderCountVal}$ của tất cả các ảnh tham chiếu trong DPB.

Danh mục ảnh tham chiếu $\text{RefPicList}[i]$ được điền với các ảnh PicOrderCntVal thấp hơn như sau:

```

for( j = PicOrderCntVal; j >= minPoc && nextIdx < NumRefIdxActive[ i ]; j-- )
{
    if( there is reference picture picA in DPB with PicOrderCntVal equal to j and
        with TemporalId <= nextTemporalId ) {
        RefPicList[ i ][ nextIdx++ ] =
picA                                (167)
        nextTemporalId = Max( TemporalId of picA - 1, 0 )
    }
}

```

8.3.2.2.3 Quy trình giải mã để điền danh mục ảnh tham chiếu với PicOrderCntVal cao hơn

Các đầu vào tới quy trình này là:

- ký hiệu nhận dạng danh mục ảnh tham chiếu i , và
- vị trí chỉ số bắt đầu $startIdx$.

Đầu ra của quy trình này là biến $nextIdx$ biểu diễn số của các vị trí được điền trong danh mục ảnh tham chiếu.

biến $nextIdx$ được thiết đặt bằng $startIdx$.

biến $nextTemporalId$ được thiết đặt bằng $\text{Max}(\text{TemporalId} - 1, 0)$.

Để maxPoc được thiết đặt bằng trị số cao nhất của PictureOrderCountVal của tất cả các ảnh tham chiếu trong DPB.

Danh mục ảnh tham chiếu RefPicList[i] được điền với các ảnh PicOrderCntVal cao hơn như sau:

```
for( j = PicOrderCntVal; j <= maxPoc && nextIdx < NumRefIdxActive[ i ]; j++ )
{
    if( there is reference picture picA in DPB with PicOrderCntVal equal to j and
        with TemporalId <= nextTemporalId ) {
        RefPicList[ i ][ nextIdx++ ] =
picA                                (168)
        nextTemporalId = Max( TemporalId of picA - 1, 0 )
    }
}
```

Theo phương án khác, có đề xuất phương pháp mã hóa chuỗi video tương ứng với phương pháp giải mã nêu trên và như được minh họa trên Fig.9. Phương pháp mã hóa chuỗi video được minh họa trên Fig.9 bao gồm bước xác định 910 xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng (bởi thiết bị giải mã) của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết hay không. Hơn nữa, phương pháp này bao gồm tạo 920 dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit trong trường hợp mà nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng (bởi thiết bị giải mã) cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Các phương pháp nêu trên có thể được thực hiện lần lượt trong thiết bị giải mã hoặc thiết bị mã hóa, như được mô tả trong phần dưới đây.

Như được thể hiện trên Fig.10 có đề xuất, theo phương án, thiết bị giải mã 1000 để giải mã chuỗi video được mã hóa. Thiết bị giải mã bao gồm bộ phận thu nhận 1010

được tạo cấu hình để thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, viết tắt là RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không và, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu. Hơn nữa, thiết bị giải mã 1000 bao gồm bộ phận xây dựng RPL 1020 để, khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được cung cấp bởi bộ phận thu nhận 1010.

Tương tự, có đề xuất thiết bị mã hóa để mã hóa chuỗi video như được thể hiện trên Fig.11. Thiết bị mã hóa được thể hiện trên Fig.11 bao gồm bộ phận xác định 1110 để xác định xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết hay không. Hơn nữa, thiết bị mã hóa 1100 bao gồm bộ phận tạo dòng bit 1120 được tạo cấu hình để tạo dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit trong trường hợp mà nó được xác định bởi bộ phận xác định 1110 rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu đối với việc dự đoán liên kết.

Thiết bị giải mã 1000 được thể hiện trên Fig.10 có thể là hoặc có thể được bao gồm bởi bộ giải mã 30 được thể hiện trên Fig.1A, Fig.1B, Fig.3 và bộ giải mã video 3206 được thể hiện trên Fig.7. Hơn nữa, thiết bị giải mã 1000 có thể được bao gồm bởi thiết bị lập mã video 400 được thể hiện trên Fig.4, máy 500 được thể hiện trên Fig.5 và thiết bị đầu cuối 3106 được thể hiện trên Fig.6. Thiết bị mã hóa 1100 được thể hiện trên Fig.11 có thể là hoặc có thể được bao gồm bởi bộ mã hóa 20 được thể hiện trên Fig.1A, Fig.1B, Fig.3. Hơn nữa, thiết bị mã hóa 1100 có thể được bao gồm bởi thiết bị lập mã

video 400 được thể hiện trên Fig.4, máy 500 được thể hiện trên Fig.5 và thiết bị chụp 3102 được thể hiện trên Fig.6.

Hơn nữa, các phương án sau đây được đề xuất ở đây.

1. Phương pháp để giải mã (ví dụ, video), ví dụ, được thực hiện bởi thiết bị giải mã, bao gồm các bước:

thu nhận cờ (ví dụ, `sps_rpl_flag` hoặc một số thông số khác) định rõ xem cú pháp (ví dụ, thông tin cú pháp hoặc các thông số cú pháp) liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có mặt trong dòng bit hay không bằng cách phân tách dòng bit (ví dụ, của video được mã hóa);

- nếu cờ (ví dụ, `sps_rpl_flag`) định rõ xem cú pháp (ví dụ, thông tin cú pháp hoặc các thông số cú pháp) liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có mặt trong dòng bit hay không, bằng 1 (hoặc nói chung, nếu trị số cờ được thiết đặt tới trị số thứ nhất, ví dụ, 1 hoặc 0, chỉ báo rằng RPL sẽ được sử dụng, trong đó trị số đối/ngịch đảo thứ hai, ví dụ, 0 hoặc 1, chỉ báo rằng RPL sẽ không được sử dụng)

- thu nhận các thành phần cú pháp biểu diễn (hoặc định rõ) các danh mục ảnh tham chiếu bằng cách phân tách dòng bit; và

- thực hiện việc quản lý danh mục ảnh tham chiếu dựa vào các danh mục ảnh tham chiếu được thu nhận;

- mặt khác,

- thực hiện việc quản lý danh mục ảnh tham chiếu khác nhau (ví dụ, việc quản lý ảnh tham chiếu nhờ sử dụng phương pháp cửa sổ trượt ngầm, ví dụ, sử dụng thành phần cú pháp `max_num_tid0_ref_pics`).

2. Phương pháp theo phương án 1, trong đó cờ (ví dụ, `sps_rpl_flag`) định rõ xem cú pháp liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có mặt trong dòng bit được báo hiệu ở mức SPS hay không.

3. Phương pháp theo phương án 1 hoặc 2, trong đó cú pháp (thông tin cú pháp hoặc các thông số cú pháp) biểu diễn các danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi (SPS) (ở mức SPS ví dụ, `rpl1_same_as_rpl0_flag` và `num_ref_pic_lists_in_sps[i]`, trong đó “i” là số của danh mục ảnh tham chiếu, ví dụ, trị số “0” tương ứng với list0 và trị số “1” tương ứng với list1) và/hoặc mức đoạn đầu

lát (SH) (ở mức SH, ví dụ, `ref_pic_list_sps_flag[ i ]` và `ref_pic_list_idx[ i ]`, trong đó “i” là số của danh mục ảnh tham chiếu, ví dụ, trị số “0” tương ứng với list0 và trị số “1” tương ứng với list1).

4. Phương pháp theo phương án bất kỳ trong số các phương án nêu trên, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), định rõ xem các danh mục ảnh tham chiếu (xem ví dụ, bảng cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag)`) mà được báo hiệu ở mức SPS, được sử dụng trong hoặc cho lát hiện tại (trong đó đoạn đầu lát thuộc về) hay không.

5. Phương pháp theo phương án 4, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát, định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong lát hiện tại hay không, luôn được báo hiệu đối với danh mục tham chiếu thứ nhất (ví dụ, list0) và cũng được báo hiệu đối với danh mục tham chiếu thứ hai (ví dụ, list1) nếu chỉ `rpl1_idx_present_flag` bằng 1 (hoặc nói chung nếu chỉ thành phần cú pháp, ví dụ cờ thứ hai, chỉ báo xem danh mục tham chiếu thứ hai được báo hiệu trong dòng bit có trị số được xác định trước, ví dụ, 1) hay không.

6. Phương pháp theo phương án 5, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS, được sử dụng trong lát hiện tại (trong đó đoạn đầu lát thuộc về), được báo hiệu ở đoạn đầu lát hay không theo đoạn trích bảng sau đây.

<code>slice_header() {</code>	Ký hiệu mô tả
<code>...</code>	
<code>if(sps_rpl_flag) {</code>	
<code>for(i = 0; i < 2; i++) {</code>	
<code>if(i == 0 (i == 1 && rpl1_idx_present_flag))</code>	
<code>ref_pic_list_sps_flag[i]</code>	u(1)

if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 &&	
rpl1_idx_present_flag))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	
...	
}	

7. Phương pháp theo phương án 4, trong đó thành phần cú pháp (ví dụ, ref_pic_list_sps_flag[i]), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), mà định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong hoặc cho lát hiện tại, không được báo hiệu nếu không có các danh mục ảnh tham chiếu được báo hiệu ở mức SPS.

8. Phương pháp theo phương án 7, trong đó thành phần cú pháp (ví dụ, ref_pic_list_sps_flag[i]), được báo hiệu ở đoạn đầu mức lát, mà định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong lát hiện tại hay không, được báo hiệu cho list0 và được báo hiệu cho list1 nếu chỉ rpl1_idx_present_flag bằng 1 (hoặc nói chung nếu chỉ thành phần cú pháp, ví dụ, cờ thứ hai, chỉ báo xem danh mục tham chiếu thứ hai được báo hiệu trong dòng bit có trị số được xác định trước, ví dụ, 1) hay không.

9. Phương pháp theo phương án 7 hoặc phương án 8, trong đó thành phần cú pháp định rõ danh mục ảnh tham chiếu nào được sử dụng được báo hiệu ở đoạn đầu lát theo đoạn trích bảng sau đây.

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i], long_term_ref_pics_flag)	
...	
}	
...	
}	

10. Phương pháp mã hóa (ví dụ, video), ví dụ, được thực hiện bởi thiết bị mã hóa, bao gồm các bước:

tạo dòng bit (ví dụ, biểu diễn phiên bản được mã hóa của video);

bổ sung (tới dòng bit) cờ (ví dụ, sps_rpl_flag hoặc một số thông số khác) định rõ xem cú pháp (ví dụ, thông tin cú pháp hoặc các thông số cú pháp) liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có mặt (hoặc được bổ sung) trong dòng bit hay không;

- nếu cờ (ví dụ, sps_rpl_flag) định rõ xem cú pháp (ví dụ, thông tin cú pháp hoặc các thông số cú pháp) liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có

mặt trong dòng bit hay không, bằng 1 (hoặc nói chung, nếu trị số cờ được thiết đặt tới trị số thứ nhất, ví dụ, 1 hoặc 0, chỉ báo rằng RPL sẽ được sử dụng, trong đó trị số đối/ngược đảo thứ hai, ví dụ, 0 hoặc 1, chỉ báo rằng RPL sẽ không được sử dụng)

- bổ sung các thành phần cú pháp biểu diễn (hoặc định rõ) các danh mục ảnh tham chiếu tới dòng bit;

- otherwise,

- không bổ sung cú pháp hoặc cú pháp khác để cho phép bộ giải mã thực hiện việc quản lý danh mục ảnh tham chiếu khác nhau (ví dụ, việc quản lý ảnh tham chiếu sử dụng phương pháp của số trượt ngầm, ví dụ, sử dụng thành phần cú pháp `max_num_tid0_ref_pics`).

11. Phương pháp theo phương án 10, trong đó cờ (ví dụ, `sps_rpl_flag`) định rõ xem cú pháp liên quan đến các danh mục ảnh tham chiếu (ví dụ, RPL) là có mặt trong dòng bit được báo hiệu ở mức SPS hay không.

12. Phương pháp theo phương án 10 hoặc 11, trong đó cú pháp (thông tin cú pháp hoặc các thông số cú pháp) biểu diễn các danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi (SPS) (ví dụ, ở mức SPS `rpl1_same_as_rpl0_flag` và `num_ref_pic_lists_in_sps[ i ]`, trong đó “i” là số của danh mục ảnh tham chiếu, ví dụ, trị số “0” tương ứng với list0 và trị số “1” tương ứng với list1) và/hoặc mức đoạn đầu lát (SH) (ở mức SH, ví dụ, `ref_pic_list_sps_flag[ i ]` và `ref_pic_list_idx[ i ]`, trong đó “i” là số của danh mục ảnh tham chiếu, ví dụ, trị số “0” tương ứng với list0 và trị số “1” tương ứng với list1).

13. Phương pháp theo phương án bất kỳ trong số các phương án 10 đến 12, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), định rõ xem các danh mục ảnh tham chiếu (xem ví dụ, bảng cú pháp `ref_pic_list_struct( listIdx, rplIdx, ltrpFlag)`) mà được báo hiệu ở mức SPS, được sử dụng trong hoặc cho lát hiện tại (trong đó đoạn đầu lát thuộc về) hay không.

14. Phương pháp theo phương án 13, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát, định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong lát hiện tại hay không,

luôn được báo hiệu đối với danh mục tham chiếu thứ nhất (ví dụ, list0) và cũng được báo hiệu đối với danh mục tham chiếu thứ hai (ví dụ, list1) nếu chỉ rpl1_idx_present_flag bằng 1 (hoặc nói chung nếu chỉ thành phần cú pháp, ví dụ cờ thứ hai, chỉ báo xem danh mục tham chiếu thứ hai được báo hiệu trong dòng bit có trị số được xác định trước, ví dụ, 1) hay không.

15. Phương pháp theo phương án 14, trong đó thành phần cú pháp (ví dụ, ref_pic_list_sps_flag[i]), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong lát hiện tại hay không (trong đó đoạn đầu lát thuộc về), được báo hiệu ở đoạn đầu lát theo đoạn trích bảng sau đây.

slice_header() {	Ký hiệu mô tả
...	
if(sps_rpl_flag) {	
for(i = 0; i < 2; i++) {	
if(i == 0 (i == 1 && rpl1_idx_present_flag))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 &&	
(i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)	
...	
}	
...	
}	

16. Phương pháp theo phương án 15, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát (hoặc trong đoạn đầu lát), mà định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS, được sử dụng trong hoặc cho lát hiện tại hay không, không được báo hiệu nếu không có các danh mục ảnh tham chiếu được báo hiệu ở mức SPS .

17. Phương pháp theo phương án 16, trong đó thành phần cú pháp (ví dụ, `ref_pic_list_sps_flag[ i ]`), được báo hiệu ở đoạn đầu mức lát, mà định rõ xem các danh mục ảnh tham chiếu mà được báo hiệu ở mức SPS được sử dụng trong lát hiện tại hay không, được báo hiệu cho list0 và được báo hiệu cho list1 nếu chỉ `rpl1_idx_present_flag` bằng 1 (hoặc nói chung nếu chỉ thành phần cú pháp, ví dụ, cờ thứ hai, chỉ báo xem danh mục tham chiếu thứ hai được báo hiệu trong dòng bit có trị số được xác định trước, ví dụ, 1) hay không.

18. Phương pháp theo phương án 16 hoặc phương án 17, trong đó thành phần cú pháp định rõ danh mục ảnh tham chiếu nào được sử dụng được báo hiệu ở đoạn đầu lát theo đoạn trích bảng sau đây.

<code>slice_header() {</code>	Ký hiệu mô tả
<code>...</code>	
<code>if(sps_rpl_flag) {</code>	
<code>for(i = 0; i < 2; i++) {</code>	
<code>if(num_ref_pic_lists_in_sps[i] > 0 && (i == 0 (i == 1 && rpl1_idx_present_flag)))</code>	
<code>ref_pic_list_sps_flag[i]</code>	u(1)
<code>if(ref_pic_list_sps_flag[i]) {</code>	
<code>if(num_ref_pic_lists_in_sps[i] > 1 &&</code>	
<code>(i == 0 (i == 1 && rpl1_idx_present_flag)))</code>	
<code>ref_pic_list_idx[i]</code>	u(v)
<code>} else</code>	

ref_pic_list_struct(i,	num_ref_pic_lists_in_sps[i],	
long_term_ref_pics_flag)		
...		
}		
...		
}		

19. Bộ mã hóa (20) bao gồm hệ mạch xử lý để thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 10 đến 18.

20. Bộ giải mã (30) bao gồm hệ mạch xử lý để thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 1 đến 9.

21. Sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 1 đến 18 khi được thực hiện trên máy tính hoặc bộ xử lý .

22. Bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ đọc được bởi máy tính không tạm thời được ghép nối với các bộ xử lý và lưu trữ sự lập trình để thực hiện bởi các bộ xử lý, trong đó sự lập trình, khi được thực hiện bởi các bộ xử lý, tạo cấu hình bộ giải mã để thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 1 đến 9.

23. Bộ mã hóa, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ đọc được bởi máy tính không tạm thời được ghép nối với các bộ xử lý và lưu trữ sự lập trình để thực hiện bởi các bộ xử lý, trong đó sự lập trình, khi được thực hiện bởi các bộ xử lý, tạo cấu hình bộ mã hóa để thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 10 đến 18.

24. Phương tiện đọc được bởi máy tính không tạm thời mang mã chương trình mã, khi được thực hiện bởi thiết bị máy tính, khiến thiết bị máy tính thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 1 đến 18.

Hơn nữa, có đề xuất:

1. Phương pháp lập mã được thực hiện bởi thiết bị giải mã, bao gồm các bước:

thu nhận cờ định rõ xem cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không bằng cách phân tách dòng bit;

trong điều kiện mà cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt, thu nhận cú pháp liên quan đến các danh mục ảnh tham chiếu bằng cách phân tách dòng bit;

dự đoán các ảnh liên quan đến cờ, dựa vào cú pháp liên quan đến các danh mục ảnh tham chiếu.

2. Phương pháp lập mã được thực hiện bởi thiết bị mã hóa, bao gồm các bước:

xác định xem cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không bằng cách phân tách dòng bit;

trong điều kiện mà cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt, mã hóa cú pháp liên quan đến các danh mục ảnh tham chiếu và cờ định rõ cú pháp liên quan đến các danh mục ảnh tham chiếu có mặt trong dòng bit thành dòng bit.

Sáng chế không giới hạn ở hệ thống nêu trên, và thiết bị mã hóa ảnh hoặc thiết bị giải mã ảnh theo các phương án nêu trên có thể được kết hợp trong hệ thống khác, ví dụ, hệ thống xe hơi.

Các toán tử toán học

Các toán tử toán học được sử dụng trong đơn này là tương tự với các toán tử toán học được sử dụng trong ngôn ngữ lập trình C. Tuy nhiên, các kết quả của phép chia số nguyên và các thao tác dịch vị số học được định rõ một cách chính xác hơn, và các thao tác bổ sung được định rõ, chẳng hạn như lũy thừa và phép chia trị số thực. Các quy ước đếm và đánh số thường bắt đầu từ 0, ví dụ, "thứ nhất" là tương đương với thứ 0, "thứ hai" tương đương với thứ 1, v.v..

Các toán tử số học

Các toán tử số học sau đây được định rõ như sau:

$+$ Cộng

$-$ Trừ (như toán tử hai đối số) hoặc phủ định (như toán tử tiền số một ngôi)

$*$ Nhân, bao gồm nhân ma trận

x^y Lũy thừa. Định rõ x có lũy thừa là y . Trong các ngữ cảnh khác, ký hiệu như vậy được sử dụng để viết lên trên không nhằm diễn giải là lũy thừa.

$/$ Chia số nguyên với sự cắt bớt của kết quả về phía không. Ví dụ, $7 / 4$ và $-7 / -4$ được cắt bớt tới 1 và $-7 / 4$ và $7 / -4$ được cắt bớt tới -1.

$\div$ Được sử dụng để biểu thị phép chia trong các phương trình toán học trong đó việc cắt bớt hoặc làm tròn không được dự định.

$\frac{x}{y}$ Được sử dụng để biểu thị phép chia trong các phương trình toán học trong đó việc cắt bớt hoặc làm tròn không được dự định.

$\sum_{i=x}^y f(i)$ Tổng của $f(i)$ với i lấy tất cả các trị số nguyên từ x lên đến và bao gồm y .

$x \% y$ Môđun. Phần còn lại của x được chia cho y , được định rõ chỉ cho các số nguyên x và y với $x \geq 0$ và $y > 0$.

Các toán tử logic

Các toán tử logic sau đây được định rõ như sau:

$x \&\& y$ Logic Boolean "và (and)" của x và y

$x || y$ Logic Boolean "hoặc (or)" của x và y

$!$ Logic Boolean "không (not)"

$x ? y : z$ Nếu x là Đúng (TRUE) hoặc không bằng 0, đánh giá tới trị số của y ; mặt khác, đánh giá tới trị số của z .

Các toán tử quan hệ

Các toán tử quan hệ sau đây được định rõ như sau:

>	Lớn hơn
>=	Lớn hơn hoặc bằng
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
=	Bằng
!=	Không bằng

Khi toán tử quan hệ được áp dụng tới thành phần cú pháp hoặc biến mà đã được gán trị số "na" (không áp dụng được), trị số "na" được xử lý như trị số khác biệt đối với thành phần cú pháp hoặc biến. Trị số "na" được coi là không bằng trị số bất kỳ khác.

Các toán tử thao tác bit (Bit-wise operator)

Các toán tử thao tác bit sau đây được định rõ như sau:

& Thao tác bit "và (and)". Khi thao tác trên các đối số nguyên, thao tác trên sự biểu diễn phân bù của hai của trị số nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit quan trọng hơn bằng 0.

| Thao tác bit "hoặc (or)". Khi thao tác trên các đối số nguyên, thao tác trên sự biểu diễn phân bù của hai của trị số nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit quan trọng hơn bằng 0.

^ Thao tác bit "hoặc loại trừ (exclusive or)". Khi thao tác trên các đối số nguyên, thao tác trên sự biểu diễn phân bù của hai của trị số nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit quan trọng hơn bằng 0.

x >> y Dịch vị phải số học của sự biểu diễn số nguyên phân bù của hai của x bởi y các chữ số nhị phân. Hàm này được định rõ chỉ đối với các trị số nguyên không âm của y. Các bit được dịch vị thành các bit quan trọng nhất (most significant bit, viết tắt là MSB) như kết quả của dịch vị phải có trị số bằng MSB của x trước thao tác dịch vị.

$x \ll y$ Dịch vị trái số học của sự biểu diễn số nguyên phần bù của hai của x bởi y các chữ số nhị phân. Hàm này được định rõ chỉ đối với các trị số nguyên không âm của y . Các bit được dịch vị thành các bit ít quan trọng nhất (least significant bit, viết tắt là LSB) như kết quả của dịch vị trái có trị số bằng 0.

Các toán tử gán

Các toán tử số học sau đây được định rõ như sau:

= Toán tử gán

++ Tăng, nghĩa là, $x++$ là tương đương với $x = x + 1$; khi được sử dụng trong chỉ số mảng, đánh giá tới trị số của biến trước thao tác tăng.

-- Giảm, nghĩa là, $x--$ là tương đương với $x = x - 1$; khi được sử dụng trong chỉ số mảng, đánh giá tới trị số của biến trước thao tác giảm.

+= Sự tăng bởi lượng được định rõ, nghĩa là, $x += 3$ là tương đương với $x = x + 3$, và $x += (-3)$ là tương đương với $x = x + (-3)$.

-- Sự giảm bởi lượng được định rõ, nghĩa là, $x -= 3$ là tương đương với $x = x - 3$, và $x -= (-3)$ là tương đương với $x = x - (-3)$.

Ký hiệu khoảng

Ký hiệu dưới đây được sử dụng để định rõ khoảng của các trị số:

$x = y..z$ x lấy các trị số nguyên bắt đầu từ y đến z , với x , y , và z là các số nguyên và z lớn hơn y .

Các hàm toán học

Các hàm toán học sau đây được định rõ:

$$\text{Abs}(x) = \begin{cases} x & ; \quad x \geq 0 \\ -x & ; \quad x < 0 \end{cases}$$

$\text{Asin}(x)$ hàm sin nghịch đảo lượng giác, thao tác trên đối số x mà nằm trong khoảng từ $-1,0$ đến $1,0$, với trị số đầu ra trong khoảng từ $-\pi/2$ đến $\pi/2$, theo các đơn vị của các radian

$\text{Atan}(x)$ hàm tang nghịch đảo lượng giác, thao tác trên đối số x , với trị số đầu ra nằm trong khoảng từ $-\pi/2$ đến $\pi/2$, theo các đơn vị của các radian

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; \quad x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; \quad x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; \quad x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; \quad x = 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \quad \text{khác} \end{cases}$$

$\text{Ceil}(x)$ số nguyên nhỏ nhất lớn hơn hoặc bằng x .

$$\text{Clip1}_Y(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_Y) - 1, x)$$

$$\text{Clip1}_C(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_C) - 1, x)$$

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{khác} \end{cases}$$

$\text{Cos}(x)$ hàm cosin lượng giác thao tác trên đối số x theo các đơn vị của các radian.

$\text{Floor}(x)$ số nguyên lớn nhất nhỏ hơn hoặc bằng x .

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; \quad b - a \geq d/2 \\ c - d & ; \quad a - b > d/2 \\ c & ; \quad \text{khác} \end{cases}$$

$\text{Ln}(x)$ lôgarit tự nhiên của x (lôgarit cơ số e , trong đó e là hằng số cơ số lôgarit tự nhiên 2.718 281 828...).

$\text{Log2}(x)$ lôgarit cơ số 2 của x .

$\text{Log10}(x)$ lôgarit cơ số 10 của x .

$$\text{Min}(x, y) = \begin{cases} x & ; \quad x \leq y \\ y & ; \quad x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0,5)$$

$$\text{Sign}(x) = \begin{cases} 1 & ; \quad x > 0 \\ 0 & ; \quad x = 0 \\ -1 & ; \quad x < 0 \end{cases}$$

$\text{Sin}(x)$ hàm sin lượng giác thao tác trên đối số x theo các đơn vị của các radian

$$\text{Sqrt}(x) = \sqrt{x}$$

$$\text{Swap}(x, y) = (y, x)$$

$\text{Tan}(x)$ hàm tang lượng giác thao tác trên đối số x theo các đơn vị của các radian

Thứ tự ưu tiên thao tác

Khi thứ tự ưu tiên trong biểu thức không được chỉ báo rõ ràng nhờ sử dụng dấu ngoặc đơn, các quy tắc sau đây áp dụng:

- Các thao tác của mức ưu tiên cao hơn được đánh giá trước thao tác bất kỳ mức ưu tiên thấp hơn.
- Các thao tác của cùng mức ưu tiên được đánh giá một cách liên tục từ trái sang phải.

Bảng dưới đây định rõ mức ưu tiên của các thao tác từ cao nhất tới thấp nhất; vị trí cao hơn trong bảng chỉ báo mức ưu tiên cao hơn.

Đối với các toán tử này mà cũng được sử dụng trong ngôn ngữ lập trình C, thứ tự của mức ưu tiên được sử dụng trong bản mô tả này là giống như được sử dụng trong ngôn ngữ lập trình C.

Bảng: Mức ưu tiên thao tác từ cao nhất (ở trên cùng của bảng) tới thấp nhất (ở dưới cùng của bảng)

Các thao tác (với toán hạng x, y, và z)
"x++", "x--"
"!x", "-x" (với toán hạng tiền tố một ngôi)
x^y
"x * y", "x / y", "x ÷ y", " $\frac{x}{y}$ ", "x % y"
"x + y", "x - y" (với toán hạng hai đối số), " $\sum_{i=x}^y f(i)$ "
"x << y", "x >> y"
"x < y", "x <= y", "x > y", "x >= y"
"x == y", "x != y"
"x & y"
"x y"
"x && y"
"x y"
"x ? y : z"
"x.y"
"x = y", "x += y", "x -= y"

Mô tả bằng văn bản của các thao tác logic

Trong văn bản, câu lệnh của các thao tác logic như sẽ được mô tả dưới dạng toán học theo dạng sau đây:

```

if( condition 0 )
    statement 0
else if( condition 1 )
    statement 1
...
else /* informative remark on remaining condition */
    statement n

```

có thể được mô tả theo cách sau đây:

... as follows / ... following applies:

- If condition 0, statement 0
- Otherwise, if condition 1, statement 1
- ...
- Otherwise (informative remark on remaining condition), statement n

Mỗi câu lệnh "If ... Otherwise, if ... Otherwise, ..." trong văn bản được đưa ra với "... as follows" hoặc "... following applies" được theo ngay sau bởi "If ... ". Điều kiện cuối cùng của "If ... Otherwise, if ... Otherwise, ..." luôn là "Otherwise, ...". Các câu lệnh "If ... Otherwise, if ... Otherwise, ..." được đan xen có thể được nhận dạng bằng cách so khớp "... as follows" hoặc "... following applies" với kết thúc "Otherwise, ...".

Trong văn bản, câu lệnh của các thao tác logic như sẽ được mô tả dưới dạng toán học theo dạng sau đây:

```

if( condition 0a  &&  condition 0b )
    statement 0
else if( condition 1a  ||  condition 1b )
    statement 1
...
else
    statement n
  
```

có thể được mô tả theo cách sau đây:

... as follows / ... following applies:

- If all of following conditions are true, statement 0:
- condition 0a
- condition 0b
- Otherwise, if one or more of following conditions are true, statement 1:
- condition 1a
- condition 1b

– ...

– Otherwise, statement n

Trong văn bản, câu lệnh của các thao tác logic như sẽ được mô tả dưới dạng toán học theo dạng sau đây:

if(condition 0)

statement 0

if(condition 1)

statement 1

có thể được mô tả theo cách sau đây:

When condition 0, statement 0

When condition 1, statement 1

Mặc dù các phương án của sáng chế đã được mô tả chủ yếu dựa vào lập mã video, cần lưu ý rằng các phương án của hệ thống lập mã 10, bộ mã hóa 20 và bộ giải mã 30 (và tương ứng là hệ thống 10) và các phương án được mô tả khác ở đây có thể cũng được tạo cấu hình cho việc lập mã hoặc xử lý ảnh tĩnh, nghĩa là xử lý hoặc lập mã của ảnh riêng độc lập với bất kỳ ảnh đứng trước hoặc tiếp theo như trong lập mã video. Nói chung chỉ các bộ phận dự đoán liên kết 244 (bộ mã hóa) và 344 (bộ giải mã) có thể là không khả dụng trong trường hợp lập mã xử lý ảnh bị giới hạn ở ảnh đơn 17. Tất cả các chức năng khác (cũng được đề cập đến là các công cụ hoặc các công nghệ) của bộ mã hóa video 20 và bộ giải mã video 30 có thể đều được sử dụng cho việc xử lý ảnh tĩnh, ví dụ, tính phần dư 204/304, biến đổi 206, lượng tử hóa 208, lượng tử hóa ngược 210/310, biến đổi (ngược) 212/312, phân chia 262/362, dự đoán bên trong 254/354, và/hoặc lọc vòng lặp 220, 320, và lập mã entropi 270 và giải mã entropi 304.

Các phương án, ví dụ, của bộ mã hóa 20 và bộ giải mã 30, và các chức năng được mô tả ở đây, ví dụ, dựa vào bộ mã hóa 20 và bộ giải mã 30, có thể được thực hiện trong phần cứng, phần mềm, phần sụn, hoặc sự kết hợp bất kỳ của nó. Nếu được thực hiện trong phần mềm, các chức năng có thể được lưu trữ trên phương tiện đọc được bởi máy tính hoặc được truyền qua các phương tiện truyền thông dưới dạng một hoặc nhiều lệnh hoặc mã và được thực hiện bởi bộ xử lý dựa trên phần cứng. Các phương tiện đọc được bởi máy tính có thể bao gồm các phương tiện lưu trữ đọc được bởi máy tính, mà tương

ứng với phương tiện hữu hình chẳng hạn như các phương tiện lưu trữ dữ liệu, hoặc các phương tiện truyền thông bao gồm phương tiện bất kỳ mà tạo thuận lợi cho việc truyền của chương trình máy tính từ một vị trí này đến vị trí khác, ví dụ, theo giao thức truyền thông. Theo cách này, các phương tiện đọc được bởi máy tính nói chung có thể tương ứng với (1) các phương tiện lưu trữ đọc được bởi máy tính hữu hình mà không phải tạm thời hoặc (2) phương tiện truyền thông chẳng hạn như tín hiệu hoặc sóng mang. Các phương tiện lưu trữ dữ liệu có thể là các phương tiện khả dụng bất kỳ mà có thể được truy cập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để truy xuất các lệnh, các cấu trúc dữ liệu và/hoặc mã cho việc thực hiện của các kỹ thuật được mô tả trong sáng chế này. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được bởi máy tính.

Bằng ví dụ, và không giới hạn, các phương tiện lưu trữ đọc được bởi máy tính như vậy có thể bao gồm RAM, ROM, EEPROM, CD-ROM hoặc phương tiện lưu trữ dạng đĩa quang khác, phương tiện lưu trữ dạng đĩa từ, hoặc các thiết bị lưu trữ từ khác, bộ nhớ tia chớp, hoặc phương tiện bất kỳ khác mà có thể được sử dụng để lưu trữ mã chương trình mong muốn dưới dạng các cấu trúc dữ liệu hoặc các lệnh và mà có thể được truy cập bởi máy tính. Ngoài ra, sự kết nối bất kỳ được gọi một cách thích hợp là phương tiện đọc được bởi máy tính. Ví dụ, nếu các lệnh được truyền từ trang mạng (website), máy chủ, hoặc nguồn từ xa khác sử dụng cáp đồng trục, cáp sợi quang, cáp cặp xoắn, đường dây thuê bao số (digital subscriber line, viết tắt là DSL), hoặc các công nghệ không dây chẳng hạn như hồng ngoại, radio, và vi sóng, thì cáp đồng trục, cáp sợi quang, cáp cặp xoắn, DSL, hoặc các công nghệ không dây chẳng hạn như hồng ngoại, radio, và vi sóng được bao gồm theo định nghĩa về phương tiện. Tuy nhiên, cần hiểu rằng các phương tiện lưu trữ đọc được bởi máy tính và các phương tiện lưu trữ dữ liệu không bao gồm các sự kết nối, các sóng mang, các tín hiệu, hoặc các phương tiện tạm thời khác, nhưng thay vào đó được hướng đến các phương tiện lưu trữ hữu hình, không tạm thời. Đĩa (disk) và đĩa (disc), như được sử dụng ở đây, bao gồm đĩa compact (compact disc, viết tắt là CD), đĩa laser, đĩa quang, đĩa đa năng số (digital versatile disc, viết tắt là DVD), đĩa mềm và đĩa Blu-ray, trong đó các đĩa (disk) thường tái tạo dữ liệu dưới dạng từ, trong khi đĩa (disc) tái tạo dữ liệu dưới dạng quang học bằng các tia laser. Các sự kết hợp của chúng cũng sẽ được bao gồm trong phạm vi của các phương tiện

đọc được bởi máy tính.

Các lệnh có thể được thực hiện bởi một hoặc nhiều bộ xử lý, chẳng hạn như một hoặc nhiều bộ xử lý tín hiệu số (DSP), các bộ vi xử lý đa năng, các mạch tích hợp chuyên dụng (application specific integrated circuit, viết tắt là ASIC), các mảng logic lập trình được dạng trường (field programmable logic array, viết tắt là FPGA), hoặc hệ mạch logic rời rạc hoặc được tích hợp tương đương khác. Theo đó, thuật ngữ “bộ xử lý,” như được sử dụng ở đây có thể đề cập đến bất kỳ của cấu trúc nêu trên hoặc cấu trúc bất kỳ khác thích hợp cho việc thực hiện của các kỹ thuật được mô tả ở đây. Ngoài ra, theo một số khía cạnh, chức năng được mô tả ở đây có thể được đưa ra trong các mô đun phần cứng và/hoặc phần mềm chuyên dụng được tạo cấu hình để mã hóa và giải mã, hoặc được kết hợp trong bộ mã hóa-giải mã được kết hợp. Ngoài ra, các kỹ thuật sẽ được thực hiện đầy đủ trong một hoặc nhiều mạch hoặc các phần tử logic.

Các kỹ thuật của sáng chế có thể được thực hiện trong phạm vi rộng của các thiết bị hoặc các máy, bao gồm thiết bị cầm tay không dây, mạch tích hợp (integrated circuit, viết tắt là IC) hoặc tập hợp của các IC (ví dụ, bộ chip). Các thành phần, các mô đun, hoặc các bộ phận khác nhau được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật được bộc lộ, mà không nhất thiết phải thực hiện bởi các bộ phận phần cứng khác nhau. Đúng hơn, như được nêu trên, các bộ phận khác nhau có thể được kết hợp trong bộ phận phần cứng bộ mã hóa-giải mã hoặc được cung cấp bởi sự tập hợp của các bộ phận phần cứng tương tác, bao gồm một hoặc nhiều bộ xử lý như được nêu trên, cùng với phần mềm và/hoặc phần sụn thích hợp.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã chuỗi video được mã hóa, bao gồm các bước:

thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, RPL) từ dòng bit của chuỗi video được mã hóa, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không;

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit: a) thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu và b) xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu; và

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit.

2. Phương pháp theo điểm 1, trong đó việc xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit bao gồm thực hiện phương pháp của sở trượt để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

3. Phương pháp theo điểm 1, còn bao gồm bước:

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, thu nhận, từ dòng bit, trị số của thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu, trong đó thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu cung cấp thông tin về số lớn nhất của các ảnh tham chiếu sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video được mã hóa.

4. Phương pháp theo điểm 1, trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng

của các danh mục ảnh tham chiếu;

trong đó các thành phần cú pháp thứ nhất liên quan đến việc xây dựng của các danh mục ảnh tham chiếu được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một danh mục ảnh tham chiếu.

5. Phương pháp theo điểm 1, trong đó thành phần cú pháp RPL là cờ, và trị số thứ nhất của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit; trong đó trị số thứ nhất là 1 và trị số thứ hai là 0.

6. Phương pháp theo điểm 1, trong đó trị số của thành phần cú pháp RPL được thu nhận từ một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh hoặc đoạn đầu lát; hoặc

trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được thu nhận từ tập hợp thông số chuỗi hoặc đoạn đầu lát.

7. Phương pháp theo điểm 1, còn bao gồm bước:

khi thành phần cú pháp RPL định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit, thu nhận trị số của ít nhất một thành phần cú pháp thứ ba từ đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không;

trong đó ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó trị số của một trong số hai thành phần cú pháp thứ ba được thu nhận đối với danh mục ảnh tham chiếu thứ nhất và khi thành phần cú pháp thứ tư được thu nhận từ dòng bit chỉ báo rằng thành phần cú pháp thứ ba khác trong số hai thành phần cú pháp thứ ba là có mặt trong dòng bit, trị số của thành phần cú pháp thứ ba khác được thu nhận đối với danh mục tham chiếu thứ hai;

trong đó thành phần cú pháp thứ tư là cờ, trị số là 1 của cờ chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit; và

trong đó trị số của ít nhất một thành phần cú pháp thứ ba được thu nhận khi có ít

nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một danh mục ảnh tham chiếu.

8. Phương pháp mã hóa chuỗi video, bao gồm các bước:

xác định xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết hay không; và

tạo dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó

thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết; và

thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết để cho phép bộ giải mã để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit.

9. Phương pháp theo điểm 8, trong đó dòng bit bao gồm thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu, khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết;

trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là

các thành phần cú pháp thứ nhất liên quan đến việc xây dựng của các danh mục ảnh tham chiếu;

trong đó các thành phần cú pháp thứ nhất liên quan đến việc xây dựng của các danh mục ảnh tham chiếu được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu; và

trong đó thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu cung cấp thông tin về số lớn nhất của các ảnh tham chiếu sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video.

10. Phương pháp theo điểm 8, trong đó thành phần cú pháp RPL là cờ, và trị số thứ nhất của cờ định rõ ràng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai của cờ định rõ ràng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit; và

trong đó trị số thứ nhất là 1 và trị số thứ hai là 0.

11. Phương pháp theo điểm 8, trong đó thành phần cú pháp RPL được báo hiệu trong một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh hoặc đoạn đầu lát; hoặc

trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được bao gồm trong tập hợp thông số chuỗi hoặc đoạn đầu lát; hoặc

trong đó ít nhất một thành phần cú pháp thứ ba được bao gồm trong đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không, khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

12. Phương pháp theo điểm 11, trong đó dòng bit bao gồm thành phần cú pháp thứ tư và ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó một trong số hai thành phần cú pháp thứ ba là cho danh mục ảnh tham chiếu thứ nhất và thành phần cú pháp khác trong số hai thành phần cú pháp thứ ba là cho danh mục tham chiếu thứ hai khi trị số của thành phần cú pháp thứ tư chỉ báo rằng thành phần cú

pháp thứ ba khác là có mặt trong dòng bit; và

trong đó thành phần cú pháp thứ tư là cờ, trị số là 1 của cờ chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit.

13. Phương pháp theo điểm 11, trong đó ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng cho việc xây dựng của một danh mục ảnh tham chiếu.

14. Bộ giải mã bao gồm:

ít nhất một bộ xử lý; và

một hoặc nhiều bộ nhớ được ghép nối với ít nhất một bộ xử lý và lưu trữ các lệnh lập trình, mà khi được thực hiện bởi ít nhất một bộ xử lý, khiến bộ giải mã:

thu nhận trị số của thành phần cú pháp danh mục ảnh tham chiếu (reference picture list, RPL) từ dòng bit của chuỗi video, thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không;

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit:

thu nhận, từ dòng bit, các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu; và

xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết nhờ sử dụng các trị số được thu nhận của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu; và

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit.

15. Bộ giải mã theo điểm 14, trong đó để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit, các lệnh lập trình, mà khi được thực hiện bởi ít nhất một bộ xử lý, khiến bộ

giải mã: thực hiện phương pháp cửa sổ trượt để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết.

16. Bộ giải mã theo điểm 14, trong đó các lệnh lập trình, mà khi được thực hiện bởi ít nhất một bộ xử lý, còn khiến bộ giải mã:

khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit, thu nhận, từ dòng bit, trị số của thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu, trong đó thành phần cú pháp thứ hai liên quan đến việc đánh dấu ảnh tham chiếu cung cấp thông tin về số lớn nhất của các ảnh tham chiếu sử dụng được cho việc dự đoán liên kết của các ảnh của chuỗi video.

17. Bộ giải mã theo điểm 14, trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là các thành phần cú pháp thứ nhất liên quan đến việc xây dựng của các danh mục ảnh tham chiếu; và

trong đó các thành phần cú pháp thứ nhất liên quan đến việc xây dựng của các danh mục ảnh tham chiếu được bao gồm trong ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một danh mục ảnh tham chiếu.

18. Bộ giải mã theo điểm 14, trong đó thành phần cú pháp RPL là cờ và trị số thứ nhất của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit và trị số thứ hai của cờ định rõ rằng cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit; trong đó trị số thứ nhất là 1 và trị số thứ hai là 0.

19. Bộ giải mã theo điểm 14, trong đó trị số của thành phần cú pháp RPL được thu nhận từ một trong số tập hợp thông số chuỗi, tập hợp thông số ảnh hoặc đoạn đầu lát; hoặc

trong đó các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu được thu nhận từ tập hợp thông số chuỗi hoặc đoạn đầu lát.

20. Bộ giải mã theo điểm 14, trong đó các lệnh lập trình, mà khi được thực hiện bởi ít nhất một bộ xử lý, còn khiến bộ giải mã:

khi thành phần cú pháp RPL định rõ rằng cú pháp liên quan đến các danh mục

ảnh tham chiếu là có mặt trong dòng bit, thu nhận trị số của ít nhất một thành phần cú pháp thứ ba từ đoạn đầu lát định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu có thể thu nhận được từ tập hợp thông số chuỗi hoặc đoạn đầu lát hay không;

trong đó ít nhất một thành phần cú pháp thứ ba bao gồm hai thành phần cú pháp thứ ba, trong đó trị số của một trong số hai thành phần cú pháp thứ ba được thu nhận đối với danh mục ảnh tham chiếu thứ nhất, và khi thành phần cú pháp thứ tư được thu nhận từ dòng bit chỉ báo rằng thành phần cú pháp thứ ba khác trong số hai thành phần cú pháp thứ ba là có mặt trong dòng bit, trị số của thành phần cú pháp thứ ba khác được thu nhận đối với danh mục tham chiếu thứ hai;

trong đó thành phần cú pháp thứ tư là cờ, trị số là 1 của cờ chỉ báo rằng thành phần cú pháp thứ ba khác là có mặt trong dòng bit; và

trong đó trị số của ít nhất một thành phần cú pháp thứ ba được thu nhận khi có ít nhất một cấu trúc cú pháp danh mục ảnh tham chiếu được báo hiệu ở mức tập hợp thông số chuỗi trong dòng bit, trong đó mỗi cấu trúc cú pháp danh mục ảnh tham chiếu được sử dụng để xây dựng một danh mục ảnh tham chiếu.

21. Bộ mã hóa bao gồm:

ít nhất một bộ xử lý; và

một hoặc nhiều bộ nhớ được ghép nối với ít nhất một bộ xử lý và lưu trữ các lệnh lập trình, mà khi được thực hiện bởi ít nhất một bộ xử lý, khiến bộ mã hóa:

xác định xem các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết hay không;

tạo dòng bit bao gồm thành phần cú pháp RPL định rõ xem các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit hay không, trong đó

thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là có mặt trong dòng bit khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham

chiều là để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết; và

thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit khi nó được xác định rằng các trị số của các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không để được sử dụng cho việc xây dựng của ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết để cho phép bộ giải mã để xây dựng ít nhất một danh mục ảnh tham chiếu cho việc dự đoán liên kết mà không thu nhận thêm các trị số của các thành phần cú pháp từ dòng bit khi thành phần cú pháp RPL định rõ rằng các thành phần cú pháp liên quan đến các danh mục ảnh tham chiếu là không có mặt trong dòng bit.

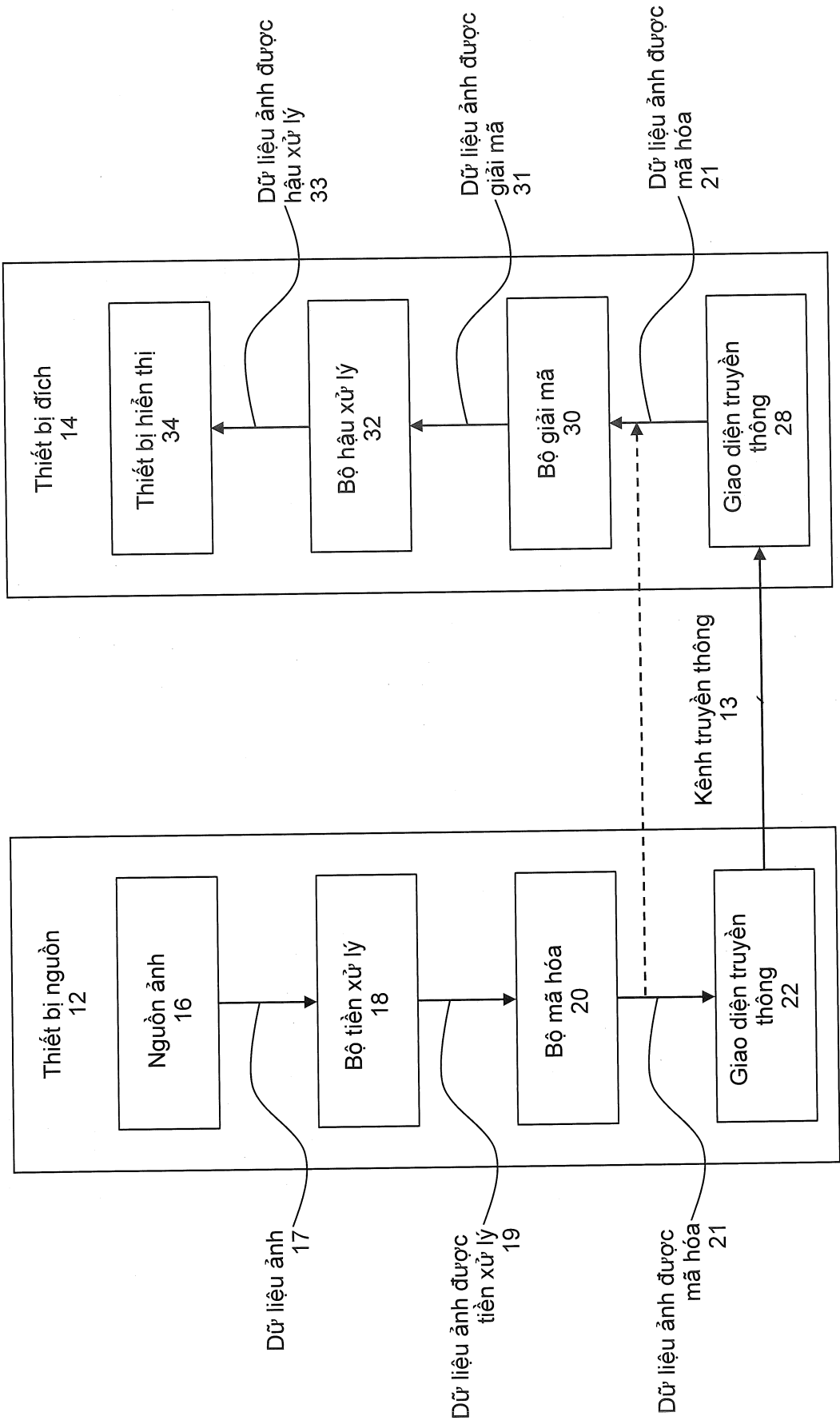


Fig. 1A

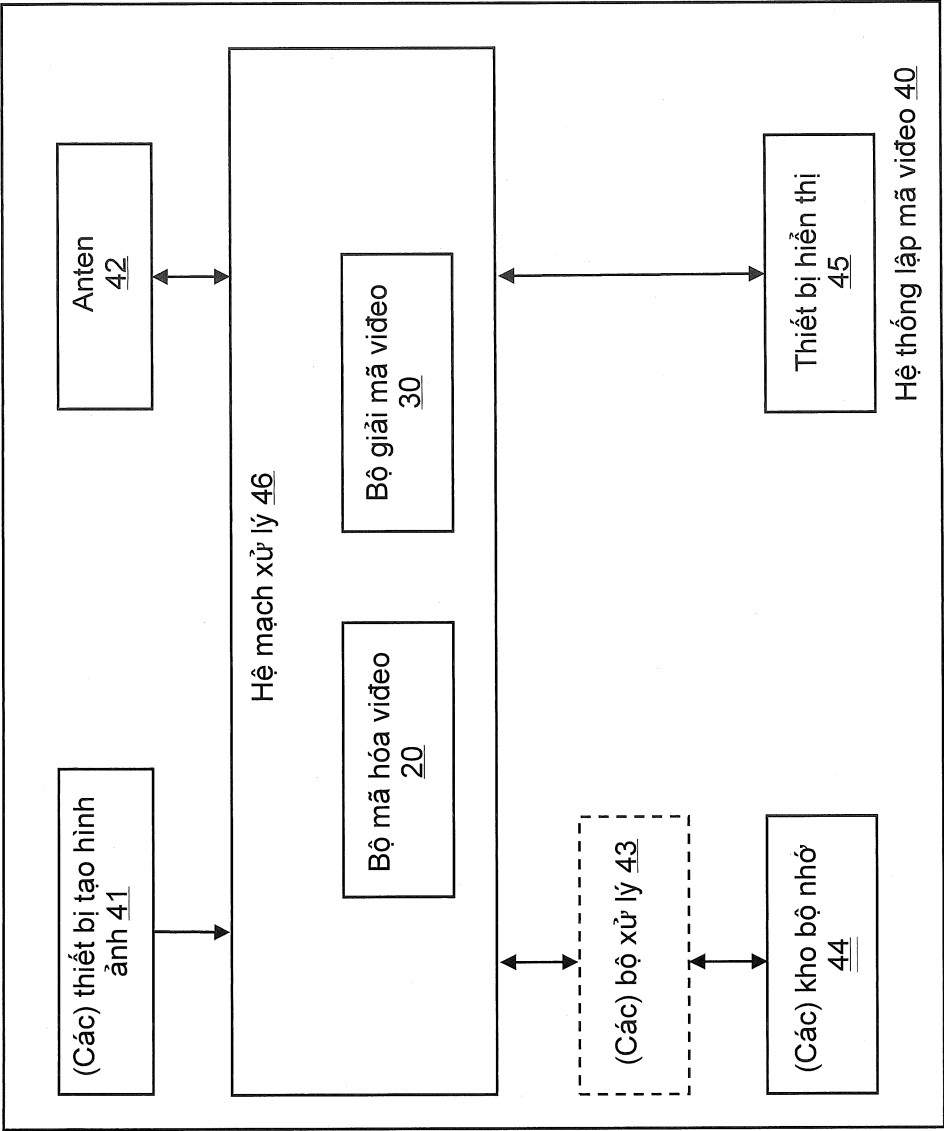


Fig. 1B

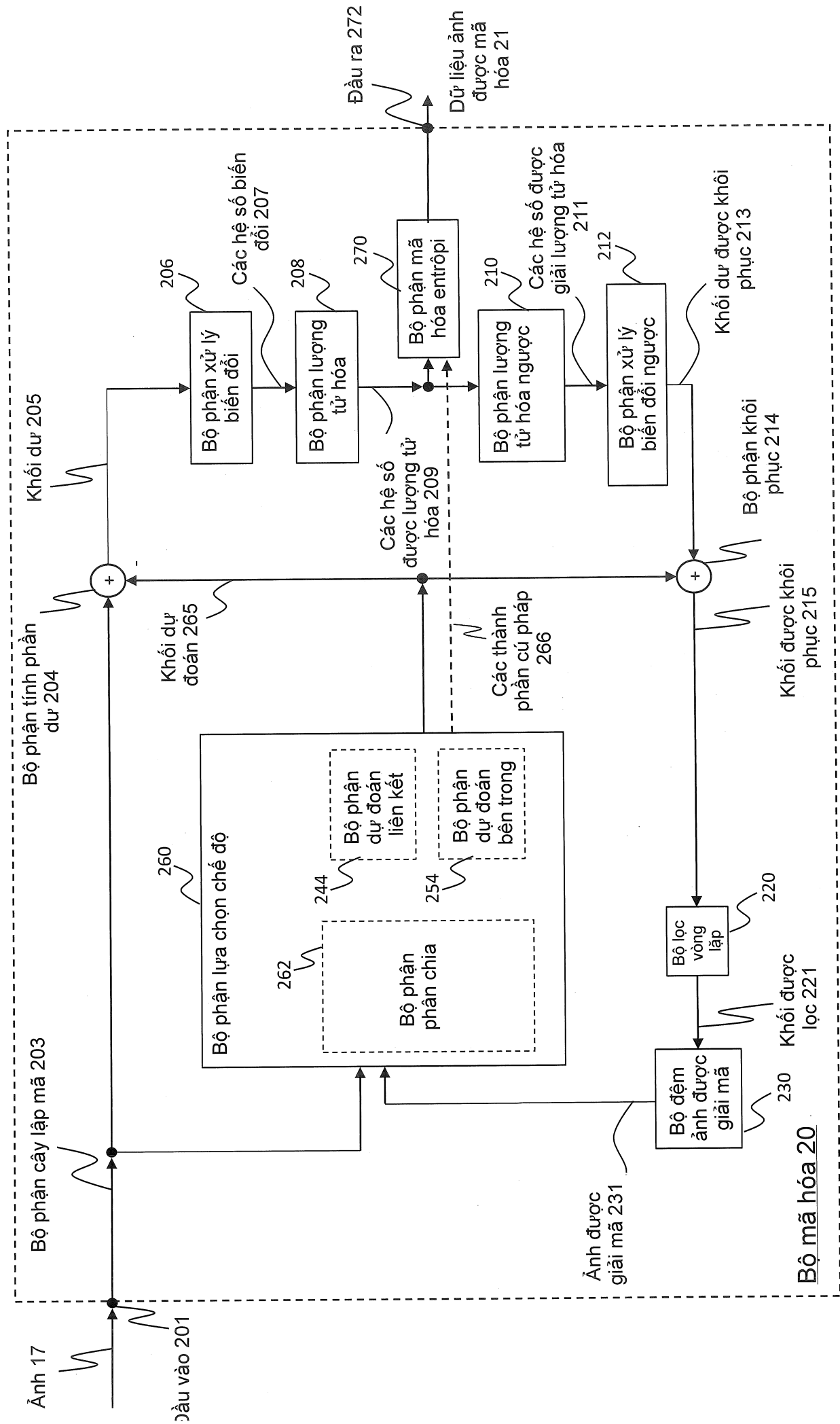


Fig. 2

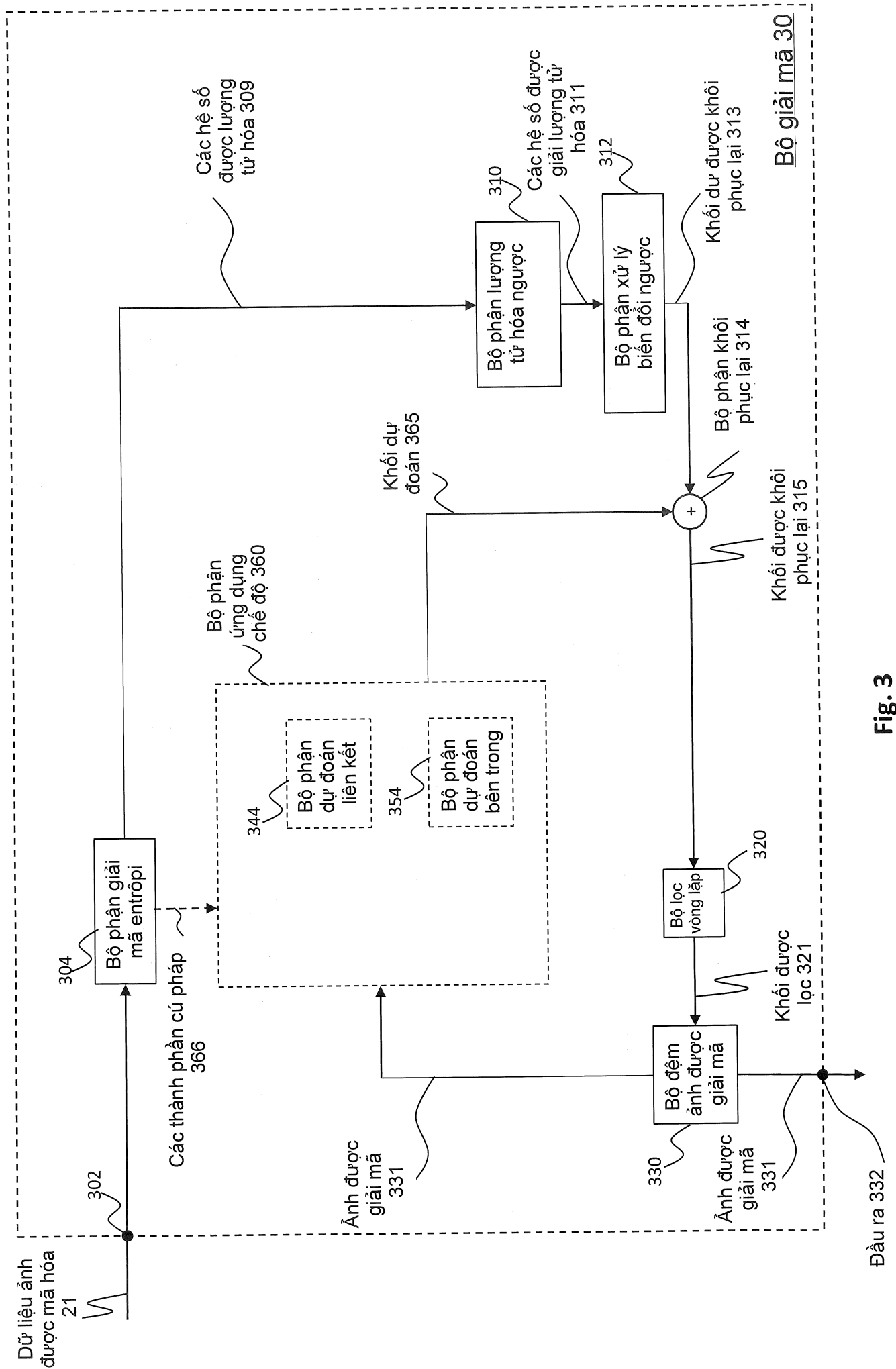
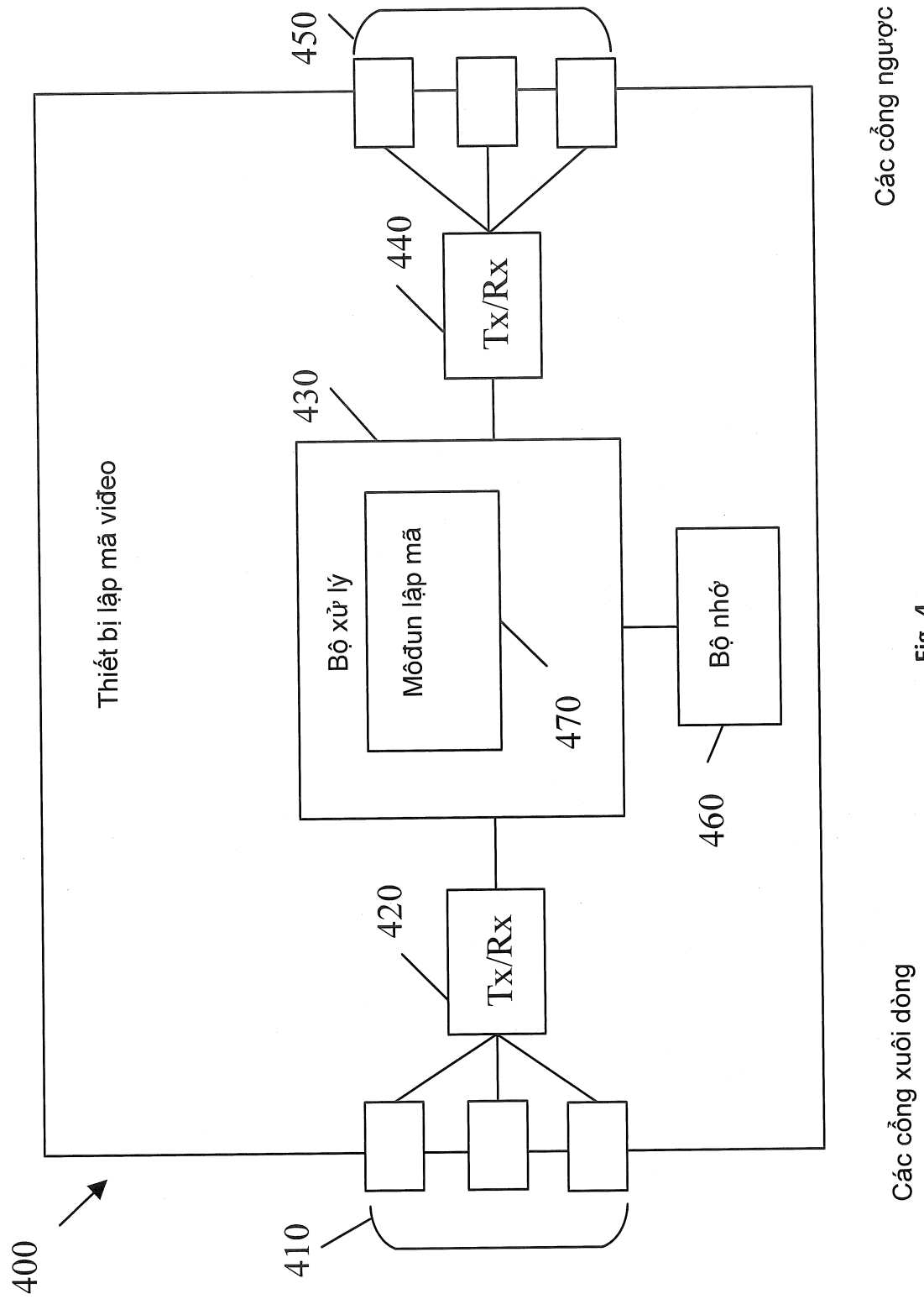


Fig. 3



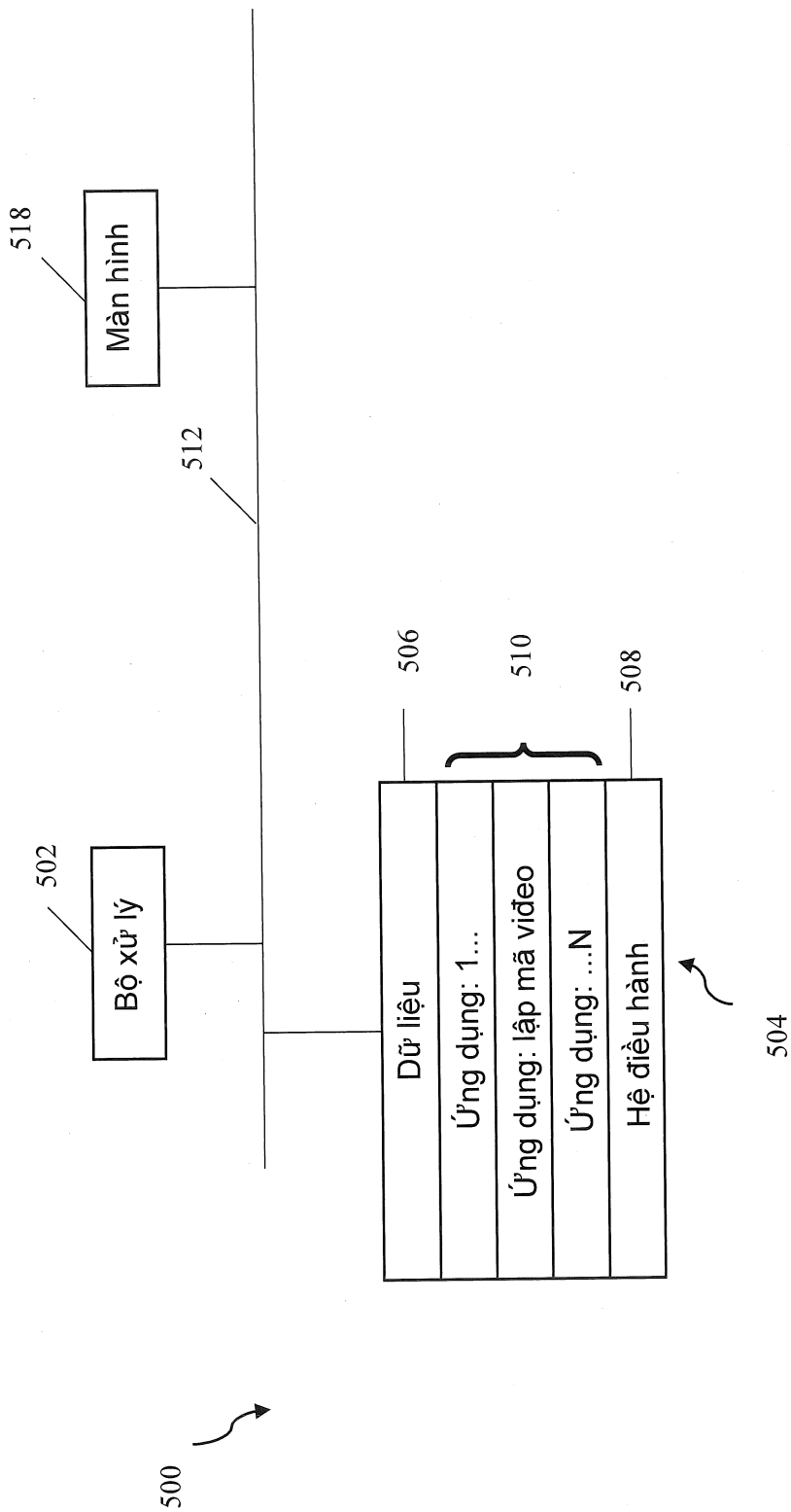


Fig. 5

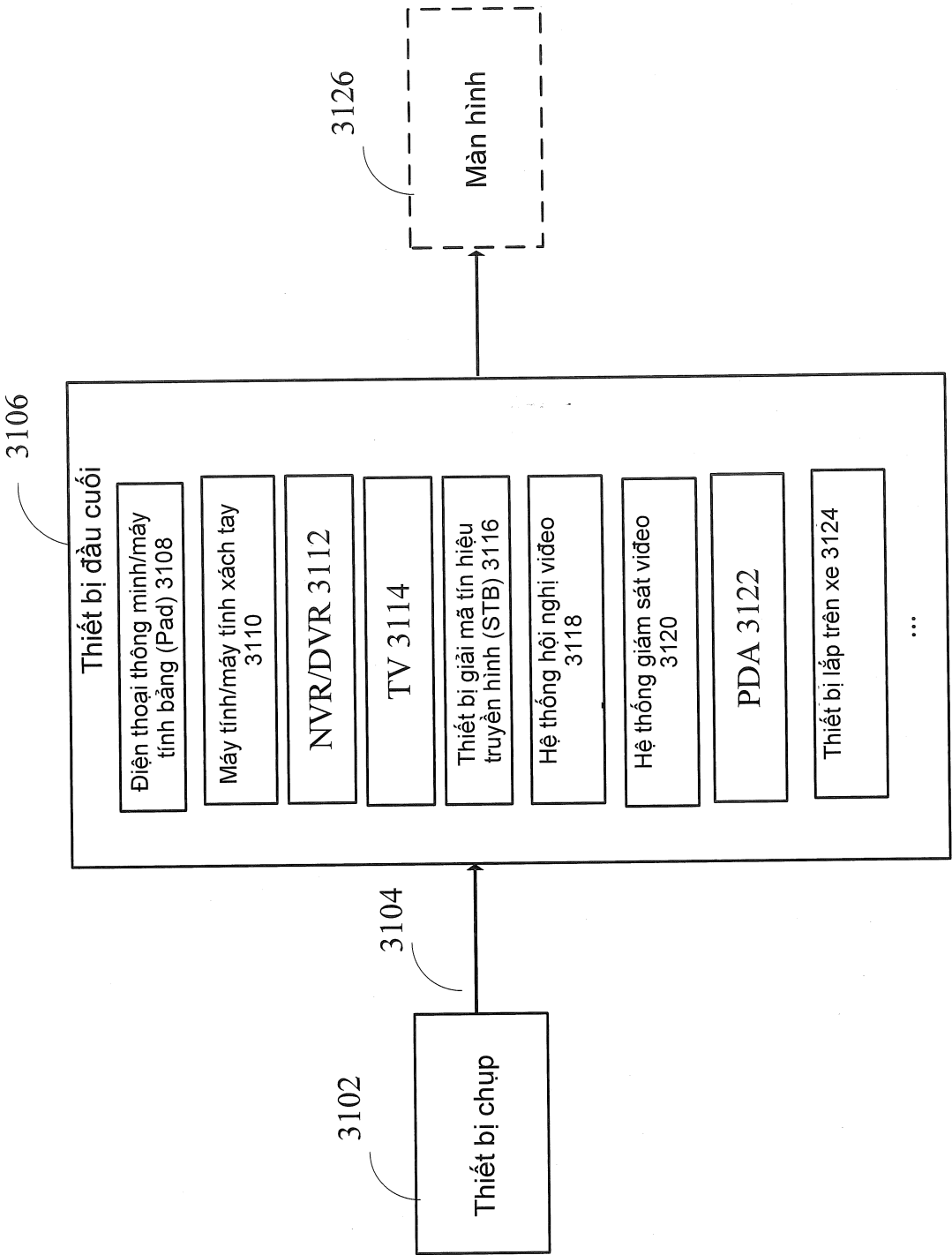


Fig. 6

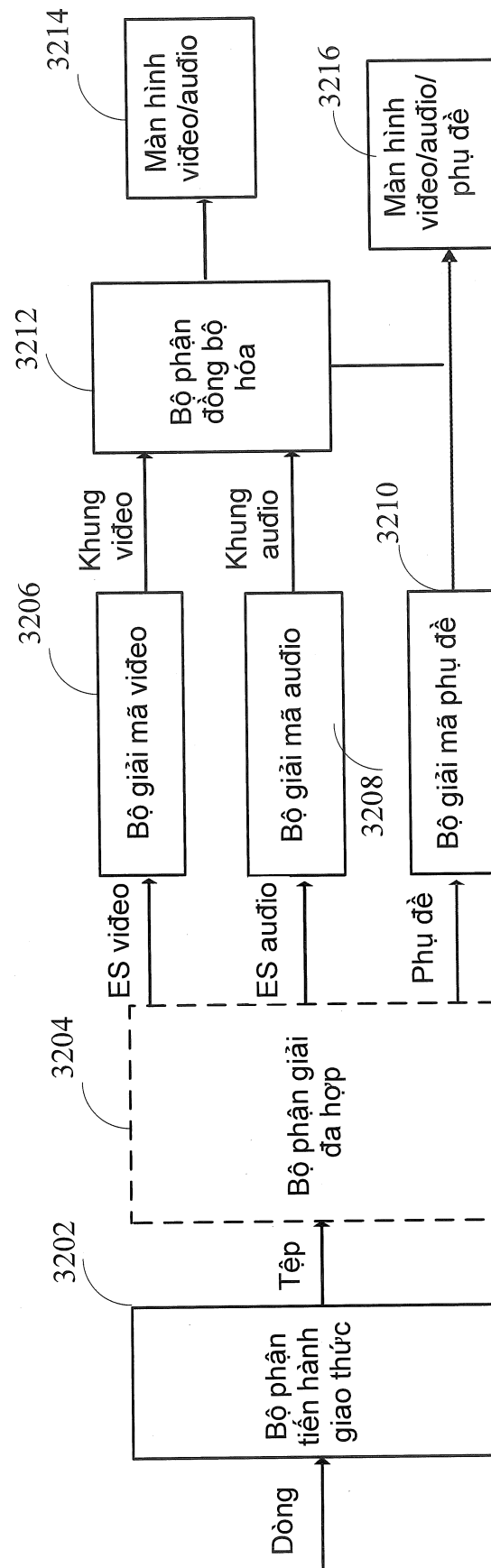


Fig. 7

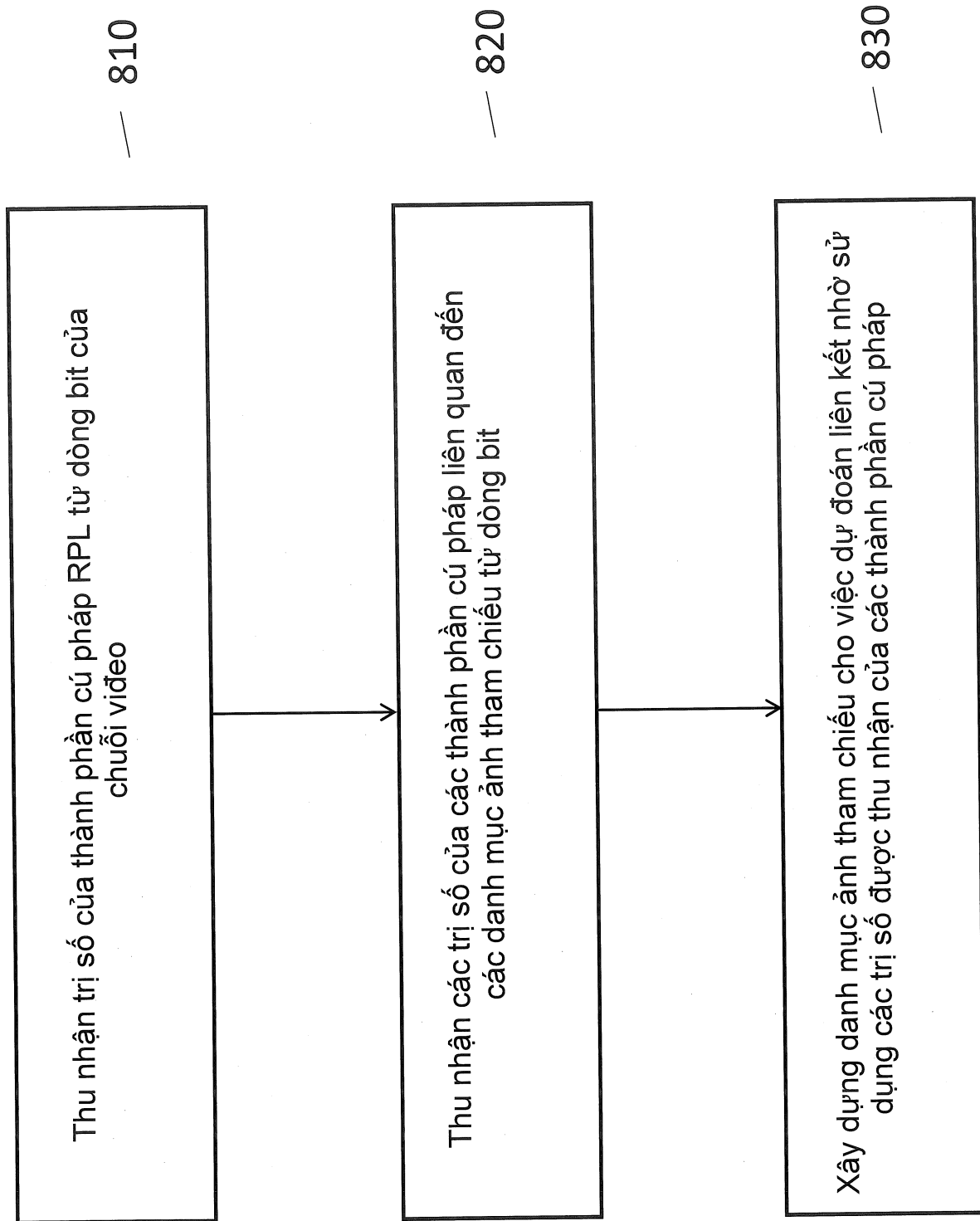


Fig. 8

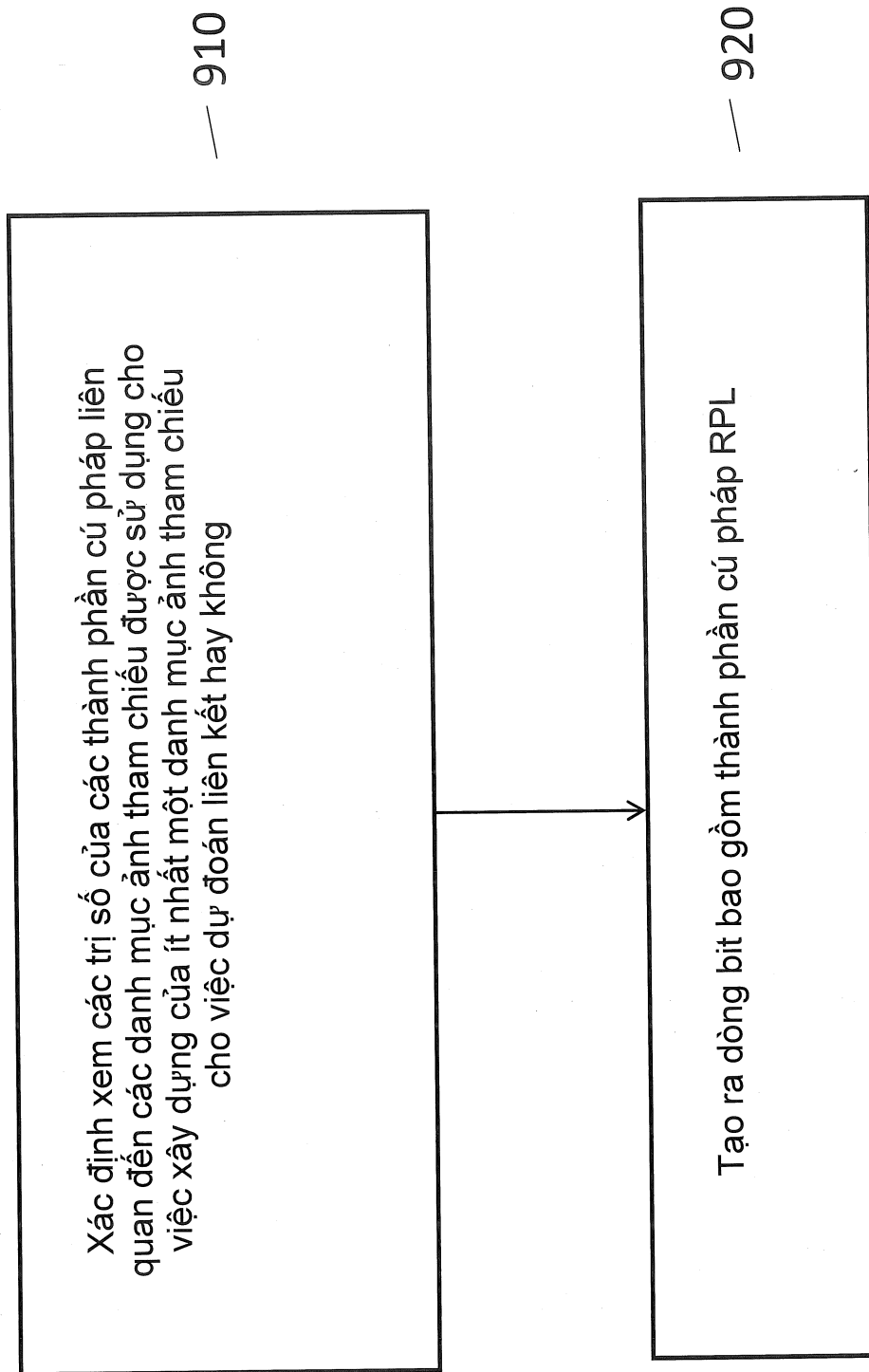
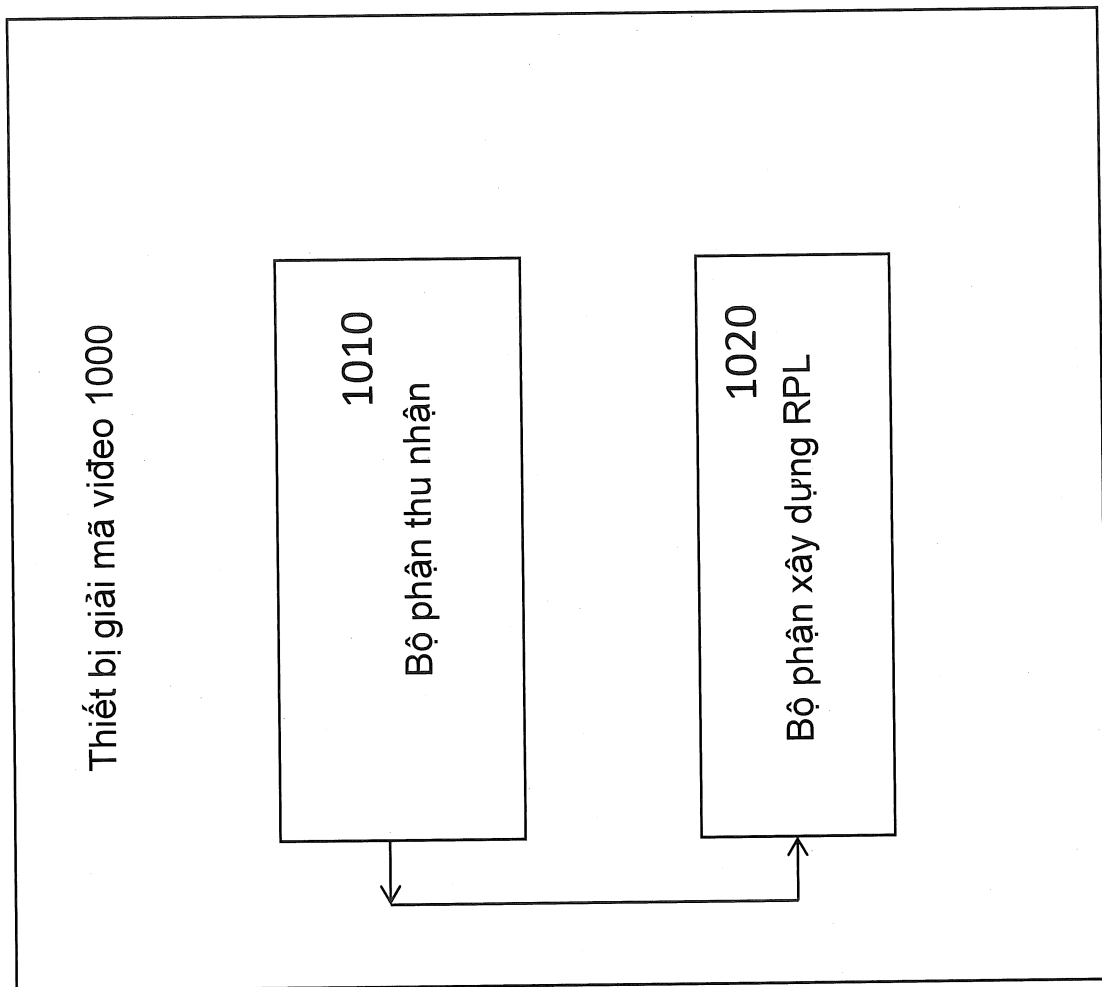
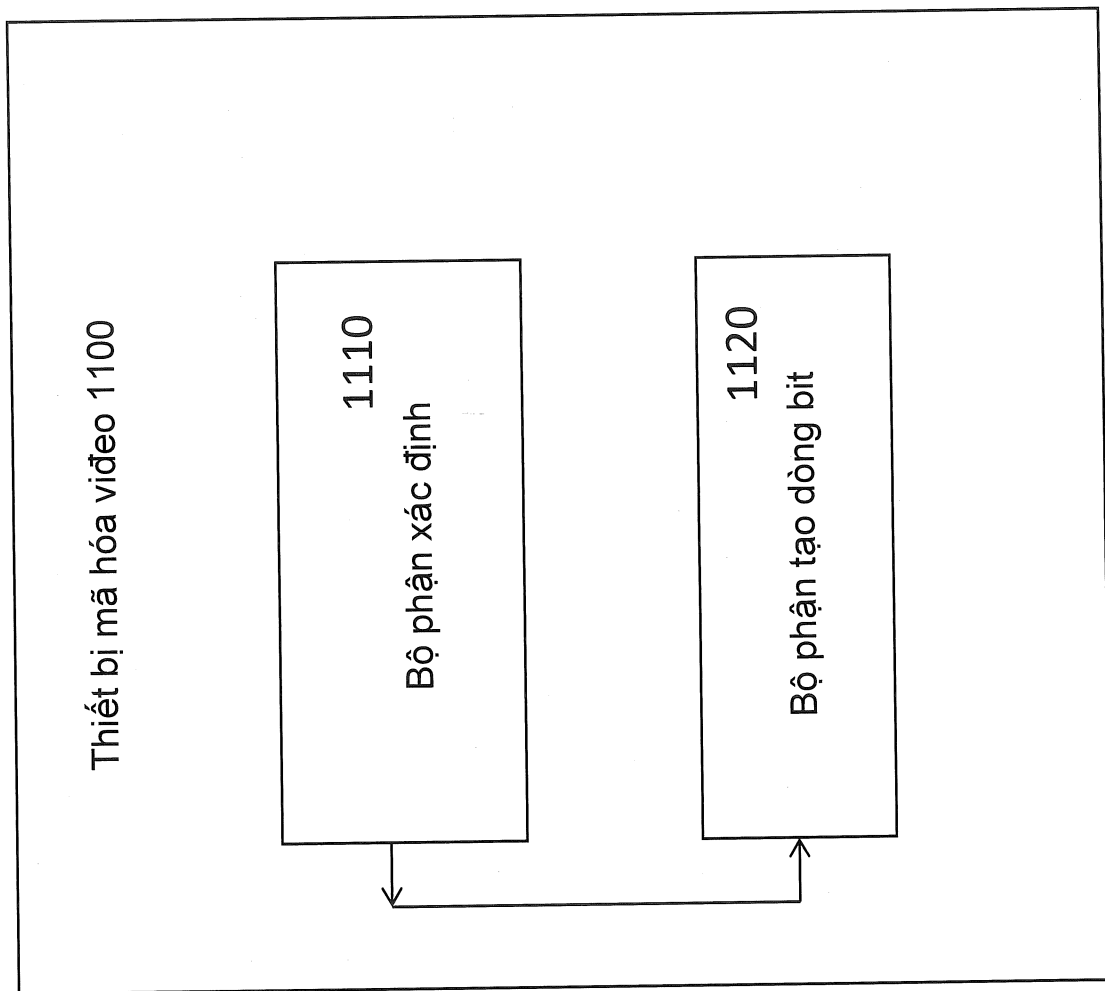


Fig. 9

**Fig. 10**

**Fig. 11**