



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

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

(11)



1-0053984

(51)^{2020.01} H04N 19/70

(13) B

(21) 1-2022-02087

(22) 07/09/2020

(86) PCT/RU2020/050214 07/09/2020

(87) WO2021/045653 A1 11/03/2021

(30) PCT/RU2019/000625 06/09/2019 RU

(45) 25/11/2025 452

(43) 27/06/2022 411A

(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)

Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, P. R. China

(72) FILIPPOV, Alexey Konstantinovich (RU); RUFITSKIY, Vasily Alexeevich (RU);
ALSHINA, Elena Alexandrovna (RU).

(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP MÃ HÓA, PHƯƠNG PHÁP GIẢI MÃ, BỘ MÃ HÓA, BỘ GIẢI
MÃ, THIẾT BỊ GIẢI MÃ, PHƯƠNG TIỆN BẮT BIẾN ĐỌC ĐƯỢC BỞI MÁY
TÍNH

(21) 1-2022-02087

(57) Sáng chế đề cập đến phương pháp mã hóa, phương pháp giải mã, bộ mã hóa, bộ giải mã, thiết bị giải mã và phương tiện bất biến đọc được bởi máy tính, trong đó phương pháp mã hóa này bao gồm: xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS); mã hóa ít nhất một tham số dự đoán theo trọng số HLS; và mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS. Các phần tử cú pháp được sắp xếp lại sao cho việc mã hóa cấu trúc danh sách ảnh tham chiếu có thể được dựa trên các giá trị của ít nhất một tham số dự đoán theo trọng số HLS.

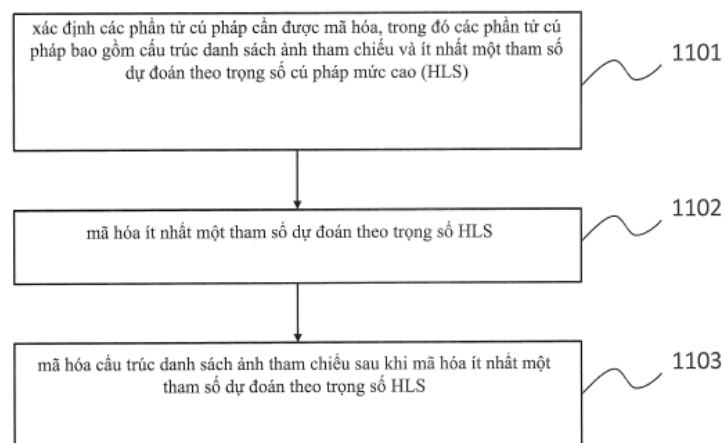


FIG. 11

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lĩnh vực xử lý hình ảnh và cụ thể hơn, đề cập đến việc tái lấy mẫu thích nghi theo hình dạng của khối dư để mã hóa hình ảnh tĩnh và video.

Tình trạng kỹ thuật của sáng chế

Việc mã hóa video (mã hóa và giải mã video) được sử dụng trong phạm vi rộng của các ứng dụng video số, chẳng hạn như TV quảng bá số, truyền video qua internet và các mạng di động, các ứng dụng đàm thoại thời gian thực như trò chuyện video, hội nghị truyền hình, DVD và các Blu-ray, hệ thống thu thập và chỉnh sửa nội dung video, và các máy quay của các ứng dụng bảo mật.

Lượng dữ liệu video cần thiết để thể hiện ngay cả video tương đối ngắn cũng có thể là đáng kể, điều này có thể dẫn đến khó khăn khi dữ liệu được truyền trực tuyến hoặc được truyền thông qua mạng truyền thông có dung lượng băng thông bị giới hạn. Do đó, dữ liệu video thường được nén trước khi được truyền thông qua các mạng viễn thông hiện đại. Kích cỡ của video cũng có thể là một vấn đề khi video được lưu trữ trên thiết bị lưu trữ vì tài nguyên bộ nhớ có thể bị hạn chế. Các thiết bị nén video thường sử dụng phần mềm và/hoặc phần cứng tại nguồn để mã hóa dữ liệu video trước khi truyền hoặc lưu trữ, do đó làm giảm lượng dữ liệu cần thiết để biểu diễn các ảnh video số. Dữ liệu được nén sau đó sẽ được thu nhận tại đích bởi một thiết bị giải nén video mà giải mã dữ liệu video. Với các tài nguyên mạng bị giới hạn và các nhu cầu ngày càng cao về chất lượng video cao hơn, các kỹ thuật nén và giải nén được cải tiến mà cải thiện tỷ lệ nén với ít hoặc không hy sinh chất lượng hình ảnh được mong muốn.

Dự đoán theo trọng số (WP-việc dự đoán theo trọng số) là công cụ, mà đặc

biệt hữu ích cho việc mã hóa các sự chuyển tiếp. Dự đoán theo trọng số có thể bù lại những thay đổi về ánh sáng, chẳng hạn như tăng dần, giảm dần hoặc chuyển tiếp chéo. Công cụ dự đoán theo trọng số (WP) đã được áp dụng trong tiêu chuẩn mã hóa video H.264 trong cấu hình chính và mở rộng để cải thiện hiệu quả mã hóa bằng cách áp dụng hệ số trọng số nhân và độ dịch cộng vào dự đoán được bù chuyển động để tạo ra việc dự đoán theo trọng số. Trong chế độ rõ ràng, hệ số trọng số và độ dịch có thể được mã hóa trong phần tiêu đề lát đối với mỗi chỉ số ảnh tham chiếu có thể cho phép. Trong chế độ ẩn, các hệ số trọng số không được mã hóa mà được thu nhận dựa trên khoảng cách số đếm thứ tự ảnh (POC) tương đối của hai ảnh tham chiếu.

Quan hệ của các ảnh về mặt thứ tự và khoảng cách nếu được sử dụng để dự đoán được biểu diễn bởi POC. Giá trị POC là số chỉ số mà xác định vị trí đầu ra của ảnh hiện tại trong chuỗi video được mã hóa. Giá trị POC được sử dụng để nhận dạng ảnh trong bộ đệm ảnh được giải mã. Để nhận dạng, giá trị của POC tăng lên theo thứ tự đầu ra của các ảnh được mã hóa.

Bản chất kỹ thuật của sáng chế

Theo khía cạnh thứ nhất của sáng chế, sáng chế đề xuất phương pháp mã hóa, phương pháp bao gồm: xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS); mã hóa ít nhất một tham số dự đoán theo trọng số HLS; và mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS. Các phần tử cú pháp được sắp xếp lại sao cho việc mã hóa cấu trúc danh sách ảnh tham chiếu có thể được dựa trên các giá trị của ít nhất một tham số dự đoán theo trọng số HLS.

Trong cách thức thực hiện thứ nhất của khía cạnh thứ nhất, danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số đếm thứ tự ảnh (POC-picture order count).

Trong cách thức thực hiện thứ hai của bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất, ít nhất một tham số dự đoán theo

trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số.

Trong cách thức thực hiện thứ ba của bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất, ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

Trong cách thức thực hiện thứ tư của phương pháp theo bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất, việc mã hóa của cấu trúc danh sách ảnh tham chiếu bao gồm giới hạn việc nhị phân hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu.

Trong cách thức thực hiện thứ năm của cách thức thực hiện thứ tư của khía cạnh thứ nhất, việc giới hạn nhị phân hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu bao gồm: khi cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

Trong cách thức thực hiện thứ của cách thức thực hiện thứ tư của khía cạnh thứ nhất, việc giới hạn nhị phân hóa bao gồm: (i) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số, và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); hoặc (ii) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự

đoán một chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); (iii) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cả tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cả tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và cả cả tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cả tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

Trong cách thức thực hiện thứ bảy của cách thức thực hiện thứ năm hoặc thứ sáu của khía cạnh thứ nhất, giá trị đen-ta POC được cải biến nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa bởi 1 đơn vị.

Theo khía cạnh thứ hai của sáng chế, sáng chế đề xuất phương pháp mã hóa, bao gồm: xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số số đếm thứ tự ảnh (POC); và mã hóa các phần tử cú pháp được xác định trong thứ tự mã hóa với giới hạn về nhị phân hóa của phần tử cú pháp mà có vị trí phía sau trong thứ tự mã hóa; trong đó khi ít nhất một tham số dự đoán theo trọng số HLS được mã hóa sau cấu trúc danh sách ảnh tham chiếu trong thứ tự mã hóa, giới hạn về nhị phân hóa của các phần tử cú pháp bao gồm: mã hóa ít nhất một tham số dự đoán theo trọng số HLS chỉ khi danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0. Điều này có ưu điểm rằng số lượng các tham số dự đoán theo trọng số được mã hóa có thể được giảm bớt.

Theo khía cạnh thứ ba của sáng chế, sáng chế đề xuất phương pháp giải mã bởi bộ giải mã, bao gồm: thu dòng bit; giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó trong các phần tử cú pháp, ít nhất một tham số dự đoán theo trọng số HLS được giải mã entropy trước cấu trúc danh sách ảnh tham chiếu; thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán; khôi phục khối được khôi phục dựa trên khối dự đoán; và thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Trong cách thức thực hiện thứ nhất của khía cạnh thứ ba, ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

Theo khía cạnh thứ tư của sáng chế, sáng chế đề xuất phương pháp giải mã bởi bộ giải mã, bao gồm: thu dòng bit; giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và cờ được thiết lập trước, trong đó giá trị của cờ được thiết lập trước chỉ báo rằng các phần tử cú pháp có bao gồm ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS) hay không; thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán; khôi phục khối được khôi phục dựa trên khối dự đoán; và thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Trong cách thức thực hiện thứ nhất của khía cạnh thứ tư, giá trị của cờ được thiết lập trước tương ứng với việc danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0 hay không.

Trong cách thức thực hiện thứ hai của cách thức thực hiện thứ nhất của khía cạnh thứ tư, giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp

bao gồm ít nhất một tham số dự đoán theo trọng số HLS; hoặc khi giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu không có bất kỳ phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp không bao gồm ít nhất một tham số dự đoán theo trọng số HLS.

Trong cách thức thực hiện thứ ba của phương pháp theo bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ tư hoặc khía cạnh thứ tư, ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

Trong cách thức thực hiện thứ tư của phương pháp theo bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ tư hoặc khía cạnh thứ tư, cờ được thiết lập trước là RestrictWPFlag mà được thiết lập là đúng (true) trong xử lý mã hóa khi giá trị 0 của giá trị đen-ta POC (AbsDeltaPocSt) xuất hiện trong khi kiểm tra mỗi phần tử của danh sách ảnh tham chiếu.

Theo khía cạnh thứ năm của sáng chế, sáng chế đề xuất bộ mã hóa bao gồm mạch xử lý để thực hiện phương pháp theo khía cạnh thứ nhất, bất kỳ một trong số cách thức thực hiện thứ nhất đến thứ bảy của khía cạnh thứ nhất, hoặc khía cạnh thứ hai.

Theo khía cạnh thứ sáu của sáng chế, sáng chế đề xuất bộ giải mã bao gồm mạch xử lý để thực hiện phương pháp theo khía cạnh thứ ba, cách thức thực hiện thứ nhất của khía cạnh thứ ba, khía cạnh thứ tư, hoặc bất kỳ một trong số các cách thức thực hiện thứ nhất đến thứ tư của khía cạnh thứ tư.

Theo khía cạnh thứ bảy của sáng chế, sáng chế đề xuất sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo khía cạnh thứ nhất, bất kỳ một trong số các cách thức thực hiện thứ nhất đến thứ bảy của khía cạnh thứ nhất, khía cạnh thứ hai, khía cạnh thứ ba, cách thức thực hiện thứ nhất của khía cạnh thứ ba, khía cạnh thứ tư, hoặc bất kỳ một trong số các cách thức thực hiện thứ nhất đến thứ tư của khía cạnh thứ tư.

Theo khía cạnh thứ tám của sáng chế, sáng chế đề xuất bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ giải mã để thực hiện phương pháp theo khía cạnh thứ ba, cách thức thực hiện thứ nhất của khía cạnh thứ ba, khía cạnh thứ tư, hoặc bất kỳ một trong số các cách thức thực hiện thứ nhất đến thứ tư của khía cạnh thứ tư.

Theo khía cạnh thứ chín của sáng chế, sáng chế đề xuất bộ giải mã, bao gồm: phương tiện thu để thu dòng bit; phương tiện giải mã entropy để giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó trong các phần tử cú pháp, ít nhất một tham số dự đoán theo trọng số HLS được giải mã entropy trước cấu trúc danh sách ảnh tham chiếu; phương tiện dự đoán để thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán; phương tiện khôi phục để khôi phục khối được khôi phục dựa trên khối dự đoán; và phương tiện thu nhận để thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Trong cách thức thực hiện thứ nhất của khía cạnh thứ chín, giá trị của cờ được thiết lập trước tương ứng với việc danh sách ảnh tham chiếu thu được từ cấu trúc danh sách tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0 hay không.

Trong cách thức thực hiện thứ hai của cách thức thực hiện thứ nhất của khía cạnh thứ chín, giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp bao gồm ít nhất một tham số dự đoán theo trọng số HLS; hoặc khi giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu không có bất kỳ phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp không bao gồm ít nhất một tham số dự đoán theo trọng số HLS.

Trong cách thức thực hiện thứ ba của bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ chín hoặc khía cạnh thứ chín, ít nhất một tham số dự đoán theo

trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

Trong cách thức thực hiện thứ tư của bất kỳ cách thức thực hiện nêu trên của khía cạnh thứ chín hoặc khía cạnh thứ chín, cờ được thiết lập trước là RestrictWPFlag mà được thiết lập là đúng (true) trong xử lý mã hóa khi giá trị 0 của giá trị đen-ta POC (AbsDeltaPocSt) xuất hiện trong khi kiểm tra mỗi phần tử của danh sách ảnh tham chiếu.

Theo khía cạnh thứ mười của sáng chế, sáng chế đề xuất bộ mã hóa, bao gồm: một hoặc nhiều bộ xử lý; và phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ mã hóa để thực hiện phương pháp theo khía cạnh thứ nhất, bất kỳ một trong số cách thức thực hiện thứ nhất đến thứ bảy của khía cạnh thứ nhất, hoặc khía cạnh thứ hai.

Theo khía cạnh thứ mười một của sáng chế, sáng chế đề xuất bộ mã hóa bao gồm: phương tiện xác định để xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS); và phương tiện mã hóa để mã hóa ít nhất một tham số dự đoán theo trọng số HLS và để mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS.

Trong cách thức thực hiện thứ nhất của khía cạnh thứ mười một, danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số số đếm thứ tự ảnh (POC).

Trong cách thức thực hiện thứ hai của khía cạnh thứ mười một hoặc cách thức thực hiện thứ nhất của khía cạnh thứ mười một, ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số.

Trong cách thức thực hiện thứ ba của khía cạnh thứ mười một hoặc bất kỳ một trong số các cách thức thực hiện nêu trên của khía cạnh thứ mười một, ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

Trong cách thức thực hiện thứ tư của khía cạnh thứ mười một hoặc bất kỳ một trong số các cách thức thực hiện nêu trên của khía cạnh thứ mười một, việc mã hóa của cấu trúc danh sách ảnh tham chiếu bao gồm giới hạn về nhị phân hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu.

Trong cách thức thực hiện thứ năm của cách thức thực hiện thứ tư của khía cạnh thứ mười một, việc giới hạn nhị phân hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu bao gồm: khi cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập bằng 0, báo hiệu giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

Trong cách thức thực hiện thứ của cách thức thực hiện thứ tư của khía cạnh thứ mười một, việc giới hạn nhị phân hóa bao gồm: (i) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số, và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); hoặc (ii) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu

trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); (iii) khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và cả cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập bằng 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

Trong cách thức thực hiện thứ bảy của cách thức thực hiện thứ năm hoặc thứ sáu của khía cạnh thứ mười một, giá trị đen-ta POC được cải biến nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa bởi 1 đơn vị.

Theo khía cạnh thứ mười một của sáng chế, sáng chế đề xuất phương tiện bất biến đọc được bởi máy tính mà mang mã chương trình mà, khi được thực thi bởi thiết bị máy tính, làm cho thiết bị máy tính thực hiện phương pháp theo khía cạnh thứ nhất, bất kỳ một trong số cách thức thực hiện thứ nhất đến thứ bảy của khía cạnh thứ nhất, khía cạnh thứ hai, khía cạnh thứ ba, cách thức thực hiện thứ nhất của khía cạnh thứ ba, khía cạnh thứ tư, hoặc bất kỳ một trong số cách thức thực hiện thứ nhất đến thứ tư của khía cạnh thứ tư.

Các phương án đề xuất các phương pháp mã hóa và giải mã chuỗi video với báo hiệu liên kết của các tham số dự đoán theo trọng số cú pháp mức cao và danh sách ảnh tham chiếu.

Các phương án đề xuất việc mã hóa và/hoặc giải mã hiệu quả nhờ sử dụng thông tin liên quan đến tín hiệu trong các thông tin tiêu đề lát chỉ đối với các lát mà cho phép hoặc có thể dự đoán liên đới hai chiều, ví dụ trong các lát dự đoán hai chiều (B), cũng được gọi là các lát B.

Có thể thu được phần nêu trên và các đối tượng khác bởi đối tượng của các

điểm yêu cầu bảo hộ độc lập. Các cách thức thực hiện khác là rõ ràng từ các điểm yêu cầu bảo hộ phụ thuộc, phần mô tả và các hình vẽ.

Các chi tiết của một hoặc nhiều phương án được mô tả trong các hình vẽ kèm theo và phần mô tả sau đây. Các đặc điểm, mục đích, và ưu điểm khác là rõ ràng từ bản mô tả, hình vẽ, và yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Trong phần sau đây, các phương án của sáng chế được mô tả chi tiết hơn có tham chiếu đến các hình vẽ và hình vẽ đính kèm, trong đó:

FIG.1A là sơ đồ khối thể hiện ví dụ về hệ thống mã hóa video có cấu trúc để thực hiện các phương án của sáng chế;

FIG.1B là sơ đồ khối thể hiện ví dụ khác của hệ thống mã hóa video có cấu trúc để thực hiện các phương án của sáng chế;

FIG.2 là sơ đồ khối thể hiện ví dụ về bộ mã hóa video có cấu trúc 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 có cấu trúc để thực hiện các phương án của sáng chế;

FIG.4 là sơ đồ khối minh họa ví dụ về thiết bị mã hóa hoặc thiết bị giải mã;

FIG.5 là sơ đồ khối minh họa ví dụ khác về thiết bị mã hóa hoặc thiết bị giải mã;

FIG.6 là lưu đồ để đưa ra quyết định về phía bộ mã hóa dự đoán theo trọng số và ước lượng tham số;

FIG.7 là lưu đồ của phương pháp được đề xuất trong đó danh sách ảnh tham chiếu được báo hiệu sau các tham số dự đoán theo trọng số;

FIG.8 là lưu đồ của phương pháp được đề xuất trong đó các tham số dự đoán theo trọng số được báo hiệu có điều kiện sau danh sách ảnh tham chiếu;

FIG.9 là sơ đồ khối thể hiện cấu trúc ví dụ của hệ thống cung cấp nội dung 3100 mà thực hiện dịch vụ cung cấp nội dung M ;

FIG.10 là sơ đồ khối thể hiện cấu trúc của ví dụ về thiết bị đầu cuối;

FIG.11 là sơ đồ khối minh họa phương pháp mã hóa theo khía cạnh thứ nhất của sáng chế;

FIG.12 là sơ đồ khối minh họa phương pháp mã hóa theo khía cạnh thứ hai của sáng chế;

FIG.13 là sơ đồ khối minh họa phương pháp giải mã theo khía cạnh thứ ba của sáng chế;

FIG.14 là sơ đồ khối minh họa phương pháp giải mã bởi bộ giải mã theo khía cạnh thứ tư của sáng chế;

FIG.15 là sơ đồ khối minh họa bộ giải mã theo khía cạnh thứ tám của sáng chế;

FIG.16 là sơ đồ khối minh họa bộ giải mã theo khía cạnh thứ chín của sáng chế;

FIG.17 là sơ đồ khối minh họa bộ mã hóa theo khía cạnh thứ mười của sáng chế; và

FIG.18 là sơ đồ khối minh họa bộ mã hóa theo khía cạnh thứ mười một của sáng chế.

Trong phần sau đây, các dấu hiệu tham chiếu giống nhau liên quan đến các đặc tính giống nhau hoặc ít nhất là tương đương về mặt chức năng nếu không được quy định rõ ràng khác.

Mô tả chi tiết sáng chế

Trong phần mô tả sau đây, việc tham chiếu được thực hiện tới các hình vẽ kèm theo, mà tạo thành một phần của sáng chế và thể hiện, bằng cách minh họa, các khía cạnh cụ thể của các phương án của sáng chế hoặc các khía cạnh cụ thể mà 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 trong các khía cạnh khác và bao

gồm các thay đổi về cấu trúc hoặc logic không được thể hiện trong các hình vẽ. Do đó, phần mô tả chi tiết sau đây không được hiểu theo nghĩa giới hạn và phạm vi của sáng chế được xác định bởi bộ yêu cầu bảo hộ kèm theo.

Ví dụ, được hiểu rằng sáng chế liên quan đến phương pháp được mô tả cũng có thể đúng với thiết bị hoặc hệ thống tương ứng được tạo cấu hình để thực hiện phương pháp đó và ngược lại. Ví dụ, nếu một hoặc nhiều bước của 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 các bước của phương pháp được mô tả (ví dụ, một bộ phận mà thực hiện một hoặc nhiều bước, hoặc mỗi trong số nhiều bộ phận mà thực hiện một hoặc nhiều bước), ngay cả nếu một hoặc nhiều bộ phận này không được mô tả rõ ràng hoặc được minh họa trong các hình vẽ. Mặt khác, ví dụ, nếu thiết bị cụ thể được mô tả dựa trên 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 để thực hiện chức năng của một hoặc nhiều bộ phận, hoặc mỗi trong số nhiều bước được sử dụng để thực hiện chức năng của một hoặc nhiều bộ phận), ngay cả nếu một hoặc nhiều bước này không được mô tả rõ ràng hoặc được minh họa trong các hình vẽ. Ngoài ra, sẽ được hiểu rằng các đặc điểm của các phương án ví dụ khác nhau và/hoặc các khía cạnh được mô tả ở đây có thể được kết hợp với nhau, trừ khi được mô tả cụ thể khác.

Việc mã hóa video thường liên quan đến xử lý của chuỗi ảnh, mà tạo thành video hoặc chuỗi video. Thay vì thuật ngữ “ảnh”, thuật ngữ “khung” hoặc “hình ảnh” có thể được sử dụng là từ đồng nghĩa trong lĩnh vực mã hóa video. Việc mã hóa video (hoặc mã hóa nói chung) bao gồm hai phần, mã hóa video và giải mã video. Việc mã hóa video được thực hiện tại phía nguồn, và thường bao gồm xử lý (ví dụ, bằng cách nén) ảnh video gốc để làm giảm lượng dữ liệu được yêu cầu để biểu diễn các ảnh video (để lưu trữ và/hoặc truyền hiệu quả hơn). Việc giải mã video được thực hiện tại phía đích, và thường bao gồm xử lý ngược so với bộ mã hóa để khôi phục ảnh video. Các phương án đề cập đến việc “mã hóa” các ảnh video (hoặc các ảnh nói chung) sẽ được hiểu là liên quan đến việc “mã hóa” hoặc

“giải mã” các ảnh video hoặc các chuỗi video tương ứng. Sự kết hợp giữa phần mã hóa và phần giải mã cũng được gọi là mã hóa-giải mã (Codec-Coding and Decoding).

Trong trường hợp mã hóa video không tổn hao, các ảnh video gốc có thể được khôi phục, tức là các ảnh video được khôi phục có chất lượng tương tự như các ảnh video gốc (giả sử không bị mất khi truyền hoặc mất dữ liệu khác trong quá trình lưu trữ hoặc truyền). Trong trường hợp mã hóa video tổn hao, việc nén thêm, chẳng hạn như bằng cách lượng tử hóa, được thực hiện, để làm giảm lượng dữ liệu mà biểu diễn các ảnh video, mà không thể được khôi phục hoàn toàn tại bộ giải mã, tức là chất lượng của các ảnh video được khôi phục thấp hơn hoặc kém hơn so với chất lượng của các ảnh video gốc.

Một số tiêu chuẩn mã hóa video thuộc nhóm “mã hóa-giải mã video lai tổn hao” (tức là, kết hợp dự đoán thời gian và không gian trong miền mẫu và mã hóa biến đổi 2D để áp dụng lượng tử hóa trong miền biến đổi). Mỗi ảnh trong chuỗi video thường được phân chia thành tập hợp của các khối không chồng lấn, và việc mã hóa thường được thực hiện tại cấp độ khối. Nói cách khác, tại bộ mã hóa, video thường được xử lý, tức là được mã hóa, ở cấp độ khối (khối video), ví dụ, bằng cách sử dụng dự đoán không gian (nội ảnh) và/hoặc dự đoán thời gian (liên ảnh) để tạo ra khối dự đoán, trừ đi khối dự đoán từ khối hiện tại (khối hiện đang được xử lý/sẽ được xử lý) để thu được khối dư, biến đổi khối dư này và lượng tử hóa khối dư này trong miền biến đổi để giảm lượng dữ liệu cần được truyền (nén), trong khi bộ giải mã, xử lý ngược với bộ mã hóa được áp dụng tới khối được mã hóa hoặc nén để khôi phục khối hiện tại cho việc biểu diễn. Hơn nữa, bộ mã hóa sao chép vòng xử lý của bộ giải mã sao cho cả hai sẽ tạo ra các dự đoán giống đồng nhất (ví dụ, các dự đoán nội bộ và dự đoán liên đới) và/hoặc cấu trúc lại để xử lý, tức là mã hóa, các khối tiếp theo.

Trong phần sau đây, các phương án của hệ thống mã hóa video 10, bộ mã hóa video 20 và bộ giải mã video 30 được mô tả dựa trên các Fig.1 đến Fig.3.

Fig.1A là sơ đồ khối giản lược minh họa hệ thống mã hóa ví dụ 10, tức là

hệ thống mã hóa video 10 (hoặc một cách ngắn gọn là hệ thống mã hóa 10) mà có thể sử dụng các kỹ thuật của sáng chế. Bộ mã hóa video 20 (hoặc một cách ngắn gọn là bộ mã hóa 20) và bộ giải mã video 30 (hoặc một cách ngắn gọn là bộ giải mã 30) của hệ thống mã hóa video 10 đại diện cho các ví dụ của thiết bị có thể có cấu trúc để thực hiện các kỹ thuật theo các ví dụ khác nhau được mô tả trong sáng chế.

Như được thể hiện trong FIG.1A, hệ thống mã hóa 10 bao gồm thiết bị nguồn 12 có cấu trúc 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 ngoài ra, tức là, một cách tùy chọn, bao gồm nguồn ảnh 16, bộ tiền xử lý (hoặc bộ xử lý trước) 18, tức là bộ tiền xử lý ảnh 18, và giao diện truyền thông hoặc bộ truyền thông 22.

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

Để phân biệt với bộ tiền xử lý 18 và xử lý được thực hiện bởi bộ xử lý trước 18, ảnh hoặc dữ liệu ảnh 17 có thể được gọi là ảnh thô hoặc dữ liệu ảnh thô 17.

Bộ tiền xử lý 18 có cấu trúc để thu dữ liệu ảnh (thô) 17 và 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 cắt, chuyển đổi khuôn dạng màu (tức là từ RGB thành YCbCr), hiệu chỉnh màu, hoặc khử tạp âm. Có thể được hiểu rằng bộ tiền xử lý 18 có thể là bộ phận tùy chọn.

Bộ mã hóa video 20 có cấu trúc để thu dữ liệu ảnh được tiền xử lý 19 và cung cấp dữ liệu ảnh được mã hóa 21 (các chi tiết hơn nữa sẽ được mô tả dưới

đây, ví dụ, dựa trên Fig.2).

Giao diện truyền thông 22 của thiết bị nguồn 12 có thể có cấu trúc để thu dữ liệu ảnh được mã hóa 21 và để truyền dữ liệu ảnh được mã hóa 21 (hoặc bất kỳ phiên bản được xử lý tiếp) trên kênh truyền thông 13 tới thiết bị khác, tức là thiết bị đích 14 hoặc bất kỳ thiết bị khác, để lưu trữ hoặc khôi phục trực tiếp.

Thiết bị đích 14 bao gồm bộ giải mã 30 (tức là bộ giải mã video 30), và có thể, tức là một cách tùy chọn, bao gồm thêm giao diện truyền thông hoặc bộ 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 có cấu trúc để thu dữ liệu ảnh được mã hóa 21 (hoặc bất kỳ phiên bản được xử lý tiếp), tức là một cách trực tiếp từ thiết bị nguồn 12 hoặc từ bất kỳ nguồn khác, tức là thiết bị lưu trữ, tức là thiết bị lưu trữ dữ liệu ảnh được mã hóa, 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 có cấu trúc để truyền hoặc thu dữ liệu ảnh được mã hóa 21 hoặc dữ liệu được mã hóa 13 thông qua liên kết truyền thông trực tiếp giữa thiết bị nguồn 12 và thiết bị đích 14, tức là kết nối có dây hoặc không dây trực tiếp, hoặc thông qua loại mạng bất kỳ, tức là mạng có dây hoặc không dây hoặc bất kỳ kết hợp của chúng, hoặc bất kỳ loại mạng công cộng hoặc riêng tư, hoặc bất kỳ loại kết hợp của chúng.

Ví dụ, giao diện truyền thông 22 có thể có cấu trúc để đóng gói dữ liệu ảnh được mã hóa 21 thành khuôn dạng thích hợp, ví dụ như các gói tin và/hoặc xử lý dữ liệu ảnh được mã hóa bằng cách sử dụng bất kỳ loại mã hóa truyền hoặc xử lý truyền để truyền qua liên kết truyền thông hoặc mạng truyền thông.

Ví dụ, giao diện truyền thông 28, mà tạo thành phần đối xứng của giao diện truyền thông 22, có thể có cấu trúc để nhận dữ liệu được truyền và xử lý dữ liệu truyền bằng cách sử dụng bất kỳ loại giải mã hoặc xử lý truyền tương ứng và/hoặc giải gói để thu được dữ liệu ảnh được mã hóa 21.

Cả hai giao diện truyền thông 22 và giao diện truyền thông 28 có thể có cấu

trúc như là giao diện truyền thông một chiều như được chỉ báo bởi mũi tên đối với kênh truyền thông 13 trong Fig.1A mà trở từ thiết bị nguồn 12 đến thiết bị đích 14, hoặc các giao diện truyền thông hai hướng, và có thể có cấu trúc, ví dụ để gửi và nhận các bản tin, ví dụ để thiết lập kết nối, xác nhận và trao đổi bất kỳ thông tin nào khác liên quan đến liên kết truyền thông và/hoặc truyền dữ liệu, ví dụ như truyền dữ liệu ảnh được mã hóa.

Bộ giải mã 30 có cấu trúc 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 (các chi tiết khác sẽ được mô tả dưới đây, ví dụ, dựa trên Fig.3 hoặc Fig.5).

Bộ hậu xử lý 32 của thiết bị đích 14 có cấu trúc để hậu xử lý dữ liệu ảnh được giải mã 31 (còn 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 hậu xử lý 32 có thể bao gồm, ví dụ, việc chuyển đổi khuôn dạng màu (ví dụ, từ YCbCr thành RGB), hiệu chỉnh màu, biên tập, lấy mẫu lại, hoặc bất kỳ xử lý 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 có cấu trúc để thu dữ liệu ảnh được hậu xử lý 33 để hiển thị ảnh, ví dụ, tới người dùng, người xem. Thiết bị hiển thị 34 có thể là hoặc có thể bao gồm loại bất kỳ của màn hình để trình diễn ảnh được khôi phục, ví dụ, màn hình hoặc máy hiển thị được tích hợp hoặc phía ngoài. Các màn hình có thể, ví dụ bao gồm màn hình tinh thể lỏng (LCD-liquid crystal display), màn hình điốt phát quang hữu cơ (OLED-organic light emitting diode), màn hình plasma, máy chiếu, màn hình micro LED, tinh thể lỏng trên silicon (LCoS), bộ xử lý ánh sáng số (DLP-digital light processor) hoặc bất kỳ loại màn hình khác.

Mặc dù Fig.1A thể hiện thiết bị nguồn 12 và thiết bị đích 14 là các thiết bị riêng biệt, nhưng các phương án của các thiết bị cũng có thể 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 này, thiết bị nguồn 12 hoặc chức

năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng có thể được thực hiện bằng cách sử dụng cùng phần cứng và/hoặc phần mềm hoặc bởi phần cứng và/hoặc phần mềm riêng biệt hoặc bất kỳ sự kết hợp của chúng.

Sẽ là rõ ràng đối với người có hiểu biết trung bình dựa trên phần mô tả, sự tồn tại và sự phân chia (chính xác) các chức năng của các bộ phận hoặc chức năng khác nhau trong thiết bị nguồn 12 và/hoặc thiết bị đích 14 như được thể hiện trong Fig.1A có thể khác nhau 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 thông qua mạch xử lý như trong Fig.1B, chẳng hạn như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (DSP-digital signal processor), mạch tích hợp ứng dụng riêng (ASIC-application-specific integrated circuit), mảng cổng lập trình dạng trường (FPGA-field-programmable gate array), logic rời rạc, phần cứng, mã hóa video chuyên dụng hoặc bất kỳ kết hợp của chúng. Bộ mã hóa 20 có thể được thực hiện thông qua mạch xử lý 46 để thể hiện các mô đun khác nhau như được mô tả liên quan đến bộ mã hóa 20 của FIG.2 và/hoặc bất kỳ hệ thống mã hóa hoặc hệ thống con nào khác được mô tả ở đây. Bộ giải mã 30 có thể được thực hiện thông qua mạch xử lý 46 để thể hiện các mô đun khác nhau như được mô tả liên quan đến bộ giải mã 30 của FIG.3 và/hoặc bất kỳ hệ thống giải mã hoặc hệ thống con khác được mô tả ở đây. Mạch xử lý có thể có cấu trúc để thực hiện các hoạt động khác nhau như được mô tả 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 đối với phần mềm trong phương tiện lưu trữ bất biến đọc được bởi máy tính thích hợp, và có thể thực thi các lệnh trong phần cứng bằng cách sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật trong sáng chế. Bộ mã hóa video **20** hoặc bộ giải mã video **30** có thể được tích hợp như là một phần của bộ mã hóa/giải mã kết hợp (CODEC) trong một thiết bị, ví dụ, như thể hiện trong Fig.1B.

Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm bất kỳ một trong số các thiết bị khác nhau, bao gồm loại bất kỳ của thiết bị cầm tay hoặc cố định, ví

dụ, máy tính xách tay cỡ nhỏ hoặc máy tính laptop, điện thoại di động, điện thoại thông minh, thiết bị dạng bảng hoặc máy tính bảng, camera, máy tính để bàn, hộp giải mã, tivi, thiết bị hiển thị, máy chơi nội dung đa phương tiện số, máy chơi trò chơi video, thiết bị truyền dòng video (như máy chủ cung cấp nội dung hoặc máy chủ phân phát nội dung), thiết bị máy thu quảng bá, hoặc thiết bị máy truyền quảng bá, hoặc loại tương tự và có thể sử dụng hoặc có thể không sử dụng loại hệ điều hành bất kỳ. Trong một vài 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 vài trường hợp, hệ thống mã hóa video 10 được thể hiện trên FIG.1A chỉ là ví dụ, và các kỹ thuật trong sáng chế có thể được áp dụng tới thiết lập mã hóa video (ví dụ, mã hóa video hoặc giải mã video) mà không cần bao gồm bất kỳ truyền thông dữ liệu giữa các thiết bị mã hóa và thiết bị giải mã. Trong các ví dụ khác, dữ liệu được truy xuất từ bộ nhớ cục bộ, được truyền trực tuyến qua mạng hoặc loại tương tự. Thiết bị mã hóa video có thể mã hóa dữ liệu và lưu trữ dữ liệu tới bộ nhớ, và/hoặc thiết bị giải mã video có thể gọi ra và giải mã dữ liệu từ bộ nhớ. Trong một vài ví dụ, việc mã hóa và giải mã được thực hiện bởi các thiết bị mà không truyền thông với nhau mà chỉ đơn giản mã hóa dữ liệu tới bộ nhớ và/hoặc gọi ra và giải mã dữ liệu từ bộ nhớ.

Để thuận tiện cho việc mô tả, các phương án của sáng chế được mô tả ở đây, ví dụ, bằng cách tham chiếu đến Mã hóa video hiệu quả cao (HEVC-High-Efficiency Video Coding) hoặc tham chiếu đến phần mềm tham chiếu của mã hóa video đa năng (VVC-Versatile Video coding), tiêu chuẩn mã hóa video thế hệ tiếp theo được phát triển bởi Nhóm cộng tác về mã hóa video (JCT-VC) của Nhóm chuyên gia mã hóa video ITU-T (VCEG) và Nhóm chuyên gia hình ảnh ISO/IEC (MPEG). Người có hiểu biết trung bình trong lĩnh vực kỹ thuật sẽ hiểu rằng các phương án của sáng chế không bị 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 có cấu

trúc để thực hiện các kỹ thuật của sáng chế. Trong ví dụ của Fig.2, bộ mã hóa video 20 bao gồm bộ đầu vào 201 (hoặc giao diện đầu vào 201), bộ tính toán phân dư 204, bộ xử lý biến đổi 206, bộ lượng tử hóa 208, bộ lượng tử hóa ngược 210, và bộ xử lý biến đổi ngược 212, bộ khôi phục 214, bộ lọc vòng 220, bộ đệm ảnh được giải mã (DPB-decoded picture buffer) 230, bộ lựa chọn chế độ 260, bộ mã hóa entropy 270 và bộ đầu ra 272 (hoặc giao diện đầu ra 272). Bộ lựa chọn chế độ 260 có thể bao gồm bộ dự đoán liên đới 244, bộ dự đoán nội bộ 254 và bộ phân chia 262. Bộ dự đoán liên đới 244 có thể bao gồm bộ ước lượng chuyển động và bộ bù chuyển động (không được hiển thị). Bộ mã hóa video 20 như được thể hiện trên FIG.2 có thể cũng được gọi là bộ mã hóa video lai hoặc bộ mã hóa video theo bộ mã hóa-giải mã video lai.

Bộ tính toán phân dư 204, bộ xử lý biến đổi 206, bộ lượng tử hóa 208, bộ lựa chọn chế độ 260 có thể tạo thành đường truyền tín hiệu tiền của bộ mã hóa 20, trong khi đó bộ lượng tử hóa ngược 210, bộ xử lý biến đổi ngược 212, bộ khôi phục 214, bộ đệm 216, bộ lọc vòng 220, bộ đệm ảnh được giải mã (DPB) 230, bộ dự đoán liên đới 244 và bộ dự đoán nội bộ 254 có thể tạo thành đường truyền tín hiệu lùi của bộ mã hóa video 20, trong đó đường truyền tín hiệu lùi của bộ mã hóa video 20 tương ứng với đường truyền tín hiệu của bộ giải mã (xem bộ giải mã video 30 trong Fig.3). Bộ lượng tử hóa ngược 210, bộ xử lý biến đổi ngược 212, bộ khôi phục 214, bộ lọc vòng 220, bộ đệm ảnh được giải mã (DPB) 230, bộ dự đoán liên đới 244 và bộ dự đoán nội bộ 254 cũng tạo thành “bộ giải mã lắp trong” của bộ mã hóa video 20.

Ảnh & Phân chia ảnh (Các Ảnh & Khối)

Bộ mã hóa 20 có thể có cấu trúc để thu, ví dụ, thông qua bộ đầu vào 201, ảnh 17 (hoặc dữ liệu ảnh 17), tức là ảnh của các chuỗi ảnh tạo thành video hoặc chuỗi video. Ảnh hoặc dữ liệu ảnh thu được cũng có thể là ảnh được tiền xử lý 19 (hoặc dữ liệu ảnh được tiền xử lý 19). Để đơn giản, phần mô tả sau đây đề cập đến Fig.17. Ảnh 17 cũng có thể được gọi là ảnh hiện tại hoặc ảnh cần được mã hóa (cụ thể trong mã hóa video để phân biệt ảnh hiện tại với các hình ảnh khác,

ví dụ như các ảnh được mã hóa và/hoặc giải mã trước đó của cùng chuỗi video, tức là chuỗi video cũng bao gồm ảnh hiện tại).

Ảnh (kỹ thuật số) là hoặc có thể được xem là mảng hoặc ma trận hai chiều của các mẫu với các giá trị cường độ. Mẫu trong mảng cũng có thể được gọi là điểm ảnh (dạng ngắn của phần tử ảnh) hoặc viết tắt là “pel”. Số lượng mẫu trong các chiều (hoặc các trục) ngang và dọc của mảng hoặc ảnh xác định kích cỡ và/hoặc độ phân giải của ảnh. Đối với trình diễn màu, ba thành phần màu thường được sử dụng, tức là, ảnh có thể được biểu diễn hoặc bao gồm ba mảng mẫu. Trong khuôn dạng RGB hoặc không gian màu, ảnh bao gồm các mảng mẫu đỏ, xanh lá cây, và xanh nước biển tương ứng. Tuy nhiên, trong mã hóa video, mỗi điểm ảnh thường được biểu diễn trong khuôn dạng sắc độ và độ chói hoặc không gian màu, ví dụ, YCbCr mà bao gồm thành phần độ chói được chỉ báo bởi Y (đôi lúc được chỉ báo bởi L) và hai thành phần sắc độ được chỉ báo bởi Cb và Cr. Thành phần độ chói (hoặc viết tắt là luma) Y biểu diễn độ sáng hoặc cường độ mức xám (ví dụ tương tự trong ảnh thang màu xám), trong khi hai thành phần sắc độ (hoặc viết tắt là chroma) Cb và Cr biểu diễn sắc độ hoặc các thành phần thông tin màu. Do đó, ảnh trong khuôn dạng YCbCr bao gồm mảng mẫu độ chói của các giá trị mẫu độ chói (Y), và hai mảng mẫu sắc độ của các giá trị sắc độ (Cb và Cr). Ảnh trong khuôn dạng RGB có thể được chuyển đổi hoặc biến đổi thành khuôn dạng YCbCr và ngược lại, xử lý này còn được gọi là chuyển đổi hoặc biến đổi màu. Nếu ảnh là đơn sắc, ảnh có thể bao gồm chỉ mảng mẫu độ chói. Do đó, ảnh có thể, ví dụ, là mảng của các mẫu độ chói trong khuôn dạng đơn sắc hoặc mảng của các mẫu độ chói và hai mảng của các mẫu sắc độ tương ứng trong khuôn dạng màu 4:2:0, 4:2:2, và 4:4:4.

Các phương án của bộ mã hóa 20 có thể bao gồm bộ phân chia ảnh (không được thể hiện trên FIG.2), có cấu trúc để phân chia ảnh 17 thành các khối ảnh (thường không chồng lấn) 203. Các khối này cũng có thể được gọi là các khối gốc, khối macrô (H.264/AVC) hoặc khối cây mã hóa (CTB) hoặc đơn vị cây mã hóa (CTU) (H.265/HEVC và VVC). Bộ phân chia có thể có cấu trúc để sử dụng kích cỡ khối giống nhau đối với tất cả ảnh của chuỗi video và lưới tương ứng mà xác

định kích cỡ khối, hoặc để thay đổi kích cỡ khối giữa các ảnh, hoặc các tập con, hoặc các nhóm ảnh, và phân chia mỗi ảnh thành các khối tương ứng.

Trong các phương án khác, bộ mã hóa video có thể có cấu trúc để thu một cách trực tiếp khối 203 của ảnh 17, ví dụ, một, một vài hoặc tất cả khối tạo thành ảnh 17. Khối ảnh 203 cũng có thể được gọi là khối ảnh hiện tại hoặc khối ảnh cần được mã hóa.

Tương tự ảnh 17, khối ảnh 203 một lần nữa hoặc có thể được xem là mảng hoặc ma trận hai chiều của các mẫu với các giá trị cường độ (các giá trị mẫu), mặc dù có kích thước nhỏ hơn so với ảnh 17. Nói cách khác, khối 203 có thể bao gồm, ví dụ, một mảng mẫu (ví dụ một mảng độ chói trong trường hợp của ảnh đơn sắc 17, hoặc mảng độ chói hoặc mảng sắc độ trong trường hợp của ảnh màu) hoặc ba mảng mẫu (ví dụ một mảng độ chói và hai mảng sắc độ trong trường hợp của ảnh màu 17) hoặc bất kỳ số lượng và/hoặc loại mảng khác phụ thuộc vào khuôn dạng màu được áp dụng. Số lượng mẫu theo chiều (hoặc trục) ngang và dọc của khối 203 xác định kích cỡ của khối 203. Do đó, khối có thể, ví dụ, là mảng $M \times N$ (M -cột x hàng N) 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 có cấu trúc để mã hóa ảnh 17 theo từng khối, tức là việc 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 có cấu trúc để phân chia và/hoặc mã hóa ảnh bằng cách sử dụng các lát (cũng được gọi là các lát video), trong đó ảnh có thể được phân chia hoặc được mã hóa nhờ sử dụng một hoặc nhiều lát (thường không chồng lấn), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ các CTU).

Các phương án của bộ mã hóa video 20 như được thể hiện trên FIG.2 có thể còn có cấu trúc để phân chia và/hoặc mã hóa ảnh bằng cách sử dụng các nhóm ô (cũng được gọi là các nhóm ô video) và/hoặc các ô (cũng được gọi là các ô video), trong đó ảnh có thể được phân chia hoặc được mã hóa nhờ sử dụng một hoặc nhiều nhóm ô (thường không xếp chồng), và mỗi nhóm ô có thể bao gồm, ví

dự một hoặc nhiều khối (ví dụ các CTU) hoặc một hoặc nhiều ô, trong đó mỗi ô, ví dụ, có thể có dạng chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ các khối hoàn chỉnh hoặc riêng phần.

Tính toán phần dư

Bộ tính toán phần dư 204 có thể có cấu trúc để tính toán khối dư 205 (cũng được gọi là phần dư 205) dựa trên khối ảnh 203 và khối dự đoán 265 (chi tiết hơn về khối dự đoán 265 được mô tả sau đây), ví dụ bằng cách trừ các giá trị mẫu của khối dự đoán 265 từ các giá trị mẫu của khối ảnh 203, theo từng mẫu (theo từng điểm ảnh) để thu nhận khối dư 205 trong miền mẫu.

Biến đổi

Bộ xử lý biến đổi 206 có thể có cấu trúc để áp dụng biến đổi, tức là biến đổi cosin rời rạc (DCT-discrete cosine transform) hoặc biến đổi sin rời rạc (DST-discrete sine transform), trên các giá trị mẫu của khối dư 205 để thu nhận các hệ số biến đổi 207 trong miền biến đổi. Các hệ số biến đổi 207 cũng có thể được gọi là các hệ số dư biến đổi và biểu diễn khối dư 205 trong miền biến đổi.

Bộ xử lý biến đổi 206 có thể có cấu trúc để áp dụng phép xấp xỉ nguyên của DCT/DST, như các biến đổi được chỉ rõ đối với HEVC/H.265. So với phép biến đổi DCT trực giao, các phép xấp xỉ nguyên này thường được chia tỷ lệ theo hệ số nhất định. Để duy trì định chuẩn của khối dư mà được xử lý bởi các biến đổi tiến và biến đổi ngược, hệ số tỷ lệ bổ sung được áp dụng như là một phần của xử lý biến đổi. Các hệ số biến đổi tỷ lệ thường được lựa chọn dựa trên các điều kiện định trước như các hệ số biến đổi tỷ lệ là lũy thừa của hai đối với các toán tử dịch, độ sâu bit của các hệ số biến đổi, sự cân bằng giữa độ chính xác và chi phí thực hiện, v.v. Các hệ số biến đổi tỷ lệ cụ thể, ví dụ, được chỉ rõ đối với việc biến đổi ngược, ví dụ bởi bộ xử lý biến đổi ngược 212 (và biến đổi ngược tương ứng, ví dụ bởi bộ xử lý biến đổi ngược 312 tại bộ giải mã video 30) và các hệ số biến đổi tỷ lệ tương ứng đối với biến đổi tiến, ví dụ bởi bộ xử lý biến đổi 206, tại bộ mã hóa 20 có thể được chỉ rõ tương ứng.

Các phương án của bộ mã hóa video 20 (tương ứng với bộ xử lý biến đổi

206) có thể có cấu trúc để xuất ra các tham số biến đổi, tức là loại biến đổi hoặc các biến đổi, ví dụ một cách trực tiếp hoặc được mã hóa hoặc được nén thông qua bộ mã hóa entropy 270, sao cho, ví dụ, bộ giải mã video 30 có thể thu và sử dụng các tham số biến đổi để giải mã.

Lượng tử hóa

Bộ lượng tử hóa 208 có thể có cấu trúc 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 tỷ lệ hoặc lượng tử hóa vectơ. Các hệ số được lượng tử hóa 209 cũng có thể được gọi là hệ số biến đổi được lượng tử hóa 209 hoặc hệ số dư được lượng tử hóa 209.

Xử lý lượng tử hóa có thể làm giảm độ sâu bit được kết hợp với một số hoặc tất cả các hệ số biến đổi 207. Ví dụ, hệ số biến đổi n -bit có thể được làm tròn xuống thành hệ số biến đổi m -bit trong khi lượng tử hóa, trong đó n lớn hơn m . Độ lượng tử hóa có thể được cải biến bằng cách điều chỉnh tham số lượng tử hóa (quantization parameter, QP). Ví dụ, đối với lượng tử hóa tỷ lệ, các tỷ lệ khác nhau có thể được áp dụng để thu được việc lượng tử hóa tinh hơn hoặc thô hơn. Các kích cỡ bước lượng tử hóa nhỏ hơn tương ứng với việc lượng tử hóa tinh hơn, trong khi các kích cỡ bước lượng tử hóa lớn hơn tương ứng với việc lượng tử hóa thô hơn. Kích cỡ bước lượng tử hóa có thể được áp dụng có thể được chỉ báo bởi tham số lượng tử hóa (QP). Tham số lượng tử hóa có thể, ví dụ là chỉ số đối với tập hợp được xác định trước của các kích cỡ bước lượng tử hóa có thể được áp dụng. Ví dụ, các tham số lượng tử hóa nhỏ có thể tương ứng với lượng tử hóa tinh (kích cỡ bước lượng tử hóa nhỏ) và các tham số lượng tử hóa lớn có thể tương ứng với lượng tử hóa thô (bước lượng tử hóa lớn), hoặc ngược lại. Việc lượng tử hóa có thể bao gồm phân chia bởi kích cỡ bước lượng tử hóa và giải lượng tử ngược và/hoặc tương ứng, tức là bởi bộ lượng tử hóa ngược 210, có thể bao gồm việc nhân bởi kích cỡ bước lượng tử hóa. Các phương án theo một vài tiêu chuẩn, tức là HEVC, có thể có cấu trúc để sử dụng tham số lượng tử hóa để xác định kích cỡ bước lượng tử hóa. Nói chung, kích cỡ bước lượng tử hóa có thể được tính toán

dựa trên tham số lượng tử hóa nhờ sử dụng phép xấp xỉ điểm cố định của phương trình bao gồm phép chia. Các hệ số tỷ lệ bổ sung có thể được đưa vào cho việc lượng tử hóa và giải lượng tử để khôi phục dạng chuẩn của khối dư, mà có thể được cải biến do việc chia tỷ lệ được sử dụng trong phép xấp xỉ điểm cố định của phương trình được sử dụng cho kích cỡ bước lượng tử hóa và tham số lượng tử hóa. Trong một cách thức thực hiện ví dụ, tỷ lệ của phép biến đổi ngược và giải lượng tử có thể được kết hợp. Ngoài ra, các bảng lượng tử hóa tùy chỉnh có thể được sử dụng và được báo hiệu từ bộ mã hóa đến bộ giải mã, ví dụ như trong dòng bit. Lượng tử hóa là hoạt động tổn hao, trong đó tổn hao tăng lên khi kích cỡ bước lượng tử hóa tăng lên.

Các phương án của bộ mã hóa video 20 (tương ứng với bộ lượng tử hóa 208) có thể có cấu trúc để xuất ra các tham số lượng tử hóa (QP), tức là một cách trực tiếp hoặc được mã hóa thông qua bộ mã hóa entropy 270, sao cho, ví dụ, bộ giải mã video 30 có thể thu và áp dụng các tham số lượng tử hóa để giải mã.

Lượng tử hóa ngược

Bộ lượng tử hóa ngược 210 có cấu trúc để áp dụng lượng tử hóa ngược của bộ lượng tử hóa 208 trên các hệ số được lượng tử hóa để thu nhận các hệ số được giải lượng tử 211, ví dụ bằng cách áp dụng nghịch đảo của phương pháp lượng tử hóa mà bộ lượng tử hóa 208 áp dụng dựa trên hoặc sử dụng cùng kích cỡ bước lượng tử hóa như bộ lượng tử hóa 208. Các hệ số được giải lượng tử 211 cũng có thể được gọi là hệ số dư được giải lượng tử 211 và tương ứng - mặc dù thường không đồng nhất với các hệ số biến đổi do sự tổn hao khi lượng tử hóa - với các hệ số biến đổi 207.

Biến đổi ngược

Bộ xử lý biến đổi ngược 212 có cấu trúc để áp dụng việc biến đổi ngược của biến đổi được áp dụng bởi bộ xử lý biến đổi 206, tức là biến đổi cosin rời rạc (DCT-discrete cosine transform) ngược hoặc biến đổi sin rời rạc (DST-discrete sine transform) ngược hoặc các biến đổi ngược khác, để thu nhận khối dư được khôi phục 213 (hoặc các hệ số được giải lượng tử tương ứng 213) trong miền mẫu.

Khối dư được khôi phục 213 cũng có thể được gọi là khối biến đổi 213.

Khôi phục

Bộ khôi phục 214 (tức là bộ cộng hoặc bộ cộng tổng 214) có cấu trúc để cộng khối biến đổi 213 (tức là khối dư được khôi phục 213) tới khối dự đoán 265 để thu nhận khối được khôi phục 215 trong miền mẫu, tức là bằng cách cộng – theo từng mẫu - các giá trị mẫu của khối dư được khôi phục 213 và các giá trị mẫu của khối dự đoán 265.

Lọc

Bộ phận lọc vòng 220 (hoặc một cách ngắn gọn là “bộ lọc vòng” 220), có cấu trúc để lọc khối được khôi phục 215 để thu nhận khối được lọc 221, hoặc nói chung, lọc các mẫu được khôi phục để thu nhận các mẫu được lọc. Ví dụ, bộ lọc vòng lặp có cấu trúc để làm phẳng các chuyển tiếp điểm ảnh hoặc cải thiện chất lượng video. Bộ lọc vòng 220 có thể bao gồm một hoặc nhiều bộ lọc vòng như bộ lọc giải khối, bộ lọc bù thích ứng mẫu (SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc song phương, bộ lọc vòng thích ứng (ALF), các bộ lọc san bằng, làm nhay hoặc bộ lọc cộng tác, hoặc bất kỳ sự kết hợp của chúng. Mặc dù bộ lọc vòng 220 được hiển thị trong Fig.2 là bộ lọc vòng nội bộ, trong các cấu trúc khác, bộ lọc vòng 220 có thể được thực hiện như bộ lọc vòng sau. Khối được lọc 221 cũng có thể được gọi là khối được khôi phục được lọc 221.

Các phương án của bộ mã hóa video 20 (một cách tương ứng là bộ lọc vòng 220) có thể có cấu trúc để xuất ra các tham số lọc vòng (như thông tin độ dịch thích nghi mẫu), tức là một cách trực tiếp hoặc được mã hóa thông qua bộ mã hóa entropy 270, sao cho, ví dụ, bộ giải mã 30 có thể thu và áp dụng cùng tham số lọc vòng hoặc bộ lọc vòng tương ứng để 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 nói chung là dữ liệu ảnh tham chiếu, để mã hóa dữ liệu video bởi bộ mã hóa video 20. DPB 230 có thể được tạo thành bởi bất kỳ thiết bị bộ nhớ,

chẳng hạn như bộ nhớ truy nhập ngẫu nhiên động (DRAM-dynamic random access memory), bao gồm DRAM đồng bộ (SDRAM), RAM từ tính (MRAM), RAM điện trở (RRAM) hoặc các loại thiết bị bộ nhớ khác. Bộ đệm ảnh được giải mã (DPB) 230 có thể được có cấu trúc để lưu trữ một hoặc nhiều khối được lọc 221. Bộ đệm ảnh được giải mã 230 có thể còn có cấu trúc để lưu trữ các khối được lọc trước đó, ví dụ như các khối được khôi phục và được lọc trước đó 221, của cùng ảnh hiện tại hoặc của các ảnh khác nhau, ví dụ như các ảnh được khôi phục trước đó và có thể cung cấp các ảnh được khôi phục hoàn chỉnh trước đó, tức là các ảnh được giải mã (và các khối và mẫu tham chiếu tương ứng) và/hoặc ảnh hiện tại được khôi phục một phần (và các khối và mẫu tham chiếu tương ứng), ví dụ cho việc dự đoán liên đới. Bộ đệm ảnh được giải mã (DPB) 230 cũng có thể có cấu trúc để lưu trữ một hoặc nhiều khối được khôi phục không được lọc 215, hoặc nói chung là các mẫu được khôi phục không được lọc, ví dụ, nếu khối được khôi phục 215 không được lọc bởi bộ lọc vòng 220 hoặc bất kỳ phiên bản khác được xử lý thêm của các khối hoặc mẫu được khôi phục.

Lựa chọn chế độ (Phân chia & Dự đoán)

Bộ lựa chọn chế độ 260 bao gồm bộ phân chia 262, bộ dự đoán liên đới 244 và bộ dự đoán nội bộ 254, và có cấu trúc để thu hoặc thu nhận dữ liệu ảnh gốc, tức là khối gốc 203 (khối hiện tại 203 của ảnh hiện tại 17), và dữ liệu ảnh được khôi phục, tức là các mẫu hoặc các khối được khôi phục được lọc và/hoặc không được lọc của cùng ảnh (hiện tại) và/hoặc từ một hoặc nhiều ảnh được giải mã trước đó, tức là từ bộ đệm ảnh được giải mã 230 hoặc các bộ đệm khác (tức là bộ đệm dòng, không được thể hiện).. Dữ liệu ảnh được khôi phục được sử dụng như là dữ liệu ảnh tham chiếu cho việc dự đoán, tức là dự đoán liên đới hoặc dự đoán nội bộ, để thu nhận khối dự đoán 265 hoặc biến dự đoán 265.

Bộ lựa chọn chế độ 260 có thể có cấu trúc để xác định hoặc lựa chọn phân vùng đối với chế độ dự đoán khối hiện tại (bao gồm không phân chia) và chế độ dự đoán (tức là chế độ dự đoán liên đới hoặc nội bộ) và tạo ra khối dự đoán tương ứng 265, mà được sử dụng cho việc tính toán của khối dự 205 và cho việc khôi

phục của khối được khôi phục 215.

Các phương án của bộ lựa chọn chế độ 260 có thể có cấu trúc để lựa chọn chế độ phân chia và dự đoán (ví dụ, từ những phương án được hỗ trợ hoặc khả dụng đối với bộ lựa chọn chế độ 260), cung cấp kết quả phù hợp nhất hoặc nói cách khác là phần dư nhỏ nhất (phần dư nhỏ nhất có nghĩa là việc nén tốt hơn để truyền hoặc lưu trữ), hoặc thông tin tiêu đề báo hiệu nhỏ nhất (thông tin tiêu đề báo hiệu nhỏ nhất có nghĩa là việc nén tốt hơn để truyền hoặc lưu trữ), mà xem xét hoặc cân bằng cả hai. Bộ lựa chọn chế độ 260 có thể có cấu trúc để xác định chế độ phân chia và dự đoán dựa trên tối ưu hóa méo tỷ lệ (RDO-rate distortion optimization), tức là lựa chọn chế độ dự đoán mà cung cấp méo tỷ lệ nhỏ nhất. Các thuật ngữ như “tốt nhất”, “nhỏ nhất”, “tối ưu”, v.v. trong ngữ cảnh này không nhất thiết đề cập đến “tốt nhất”, “nhỏ nhất”, “tối ưu” chung v.v. nhưng cũng có thể đề cập đến việc hoàn thành tiêu chuẩn chấm dứt hoặc lựa chọn như giá trị vượt quá hoặc nằm dưới ngưỡng hoặc các điều kiện khác có khả năng dẫn đến “lựa chọn dưới mức tối ưu” nhưng giảm độ phức tạp và thời gian xử lý.

Nói cách khác, bộ phân chia 262 có thể có cấu trúc để phân chia khối 203 thành các phân vùng khối nhỏ hơn hoặc các khối con (mà tạo thành khối lân nữa), tức là sử dụng lặp lại việc phân chia cây tứ phân (QT-quad-tree), phân chia cây nhị phân (Bt-binary tree) hoặc phân chia cây tam phân (TT-triple-tree) hoặc bất kỳ kết hợp của chúng, và để thực hiện, ví dụ, việc dự đoán đối với mỗi phân vùng khối hoặc các khối con, trong đó việc lựa chọn chế độ bao gồm việc lựa chọn của cấu trúc cây của khối được phân chia 203 và các chế độ dự đoán được áp dụng tới mỗi phân vùng khối hoặc các khối con.

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

Phân chia

Bộ phân chia 262 có thể phân chia (hoặc tách) khối hiện tại 203 thành các phân vùng nhỏ hơn, ví dụ các khối nhỏ hơn có kích cỡ hình vuông hoặc hình chữ

nhật. Các khối nhỏ hơn này (mà cũng có thể được gọi là các khối con) có thể được phân chia tiếp thành các phân vùng nhỏ hơn nữa. Điều này cũng được gọi là phân chia dạng cây hoặc phân chia dạng cây phân cấp, trong đó khối gốc, tức là tại mức cây gốc 0 (mức phân cấp 0, độ sâu 0), có thể được phân chia đệ quy, tức là được phân chia thành hai khối hoặc nhiều hơn của mức cây thấp hơn tiếp theo, tức là các nút tại mức cây 1 (mức phân cấp 1, độ sâu 1), trong đó các khối này có thể được phân chia tiếp thành hai khối hoặc nhiều hơn của mức thấp hơn tiếp theo, tức là mức cây 2 (mức phân cấp 2, độ sâu 2), v.v. cho đến khi việc phân chia được kết thúc, ví dụ do tiêu chuẩn kết thúc được thỏa mãn, ví dụ, đạt tới độ sâu cây lớn nhất hoặc kích cỡ khối nhỏ nhất. Các khối mà không được phân chia tiếp cũng được gọi là các khối nhánh hoặc nút nhánh của cây. Cây mà sử dụng phân chia thành hai phân vùng được gọi là cây nhị phân (BT-binary-tree), cây mà sử dụng phân chia thành ba phân vùng được gọi là cây tam phân (TT-ternary-tree) và cây mà sử dụng phân chia thành bốn phân vùng được gọi là cây tứ phân (QT-quad-tree).

Như đã đề cập trước đó, thuật ngữ “khối” như được sử dụng ở đây có thể là một phân, cụ thể là phần hình vuông hoặc hình chữ nhật, của ảnh. Ví dụ, tham chiếu tới HEVC và VVC, khối có thể là hoặc tương ứng với đơn vị cây mã hóa (CTU-coding tree unit), đơn vị mã hóa (CU-coding unit), đơn vị dự đoán (PU-prediction unit) và đơn vị biến đổi (TU-transform unit) và/hoặc các khối tương ứng, ví dụ khối cây mã hóa (CTB-coding tree block), khối mã hóa (CB-coding block), khối biến đổi (TB-transform block) hoặc khối dự đoán (PB-prediction block).

Ví dụ, đơn vị cây mã hóa (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 mã hóa sử dụng ba mặt phẳng màu riêng biệt và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu. Một cách tương ứng, khối cây mã hóa (CTB) có thể là khối NxN của các mẫu đối với một vài giá trị của N sao cho việc phân chia của thành phần thành các CTB là sự phân chia. Đơn vị mã hóa (CU-coding unit) có thể là hoặc bao gồm khối mã

hóa của các mẫu độ chói, hai khối mã hóa tương ứng của các mẫu sắc độ của ảnh mà có ba mảng mẫu, hoặc khối mã hóa của các mẫu của ảnh đơn sắc hoặc ảnh mà được mã hóa nhờ sử dụng ba mặt phẳng màu riêng biệt và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu. Một cách tương ứng, khối mã hóa (CB) có thể là khối $M \times N$ của các mẫu đối với một vài giá trị của M và N sao cho việc phân chia CTB thành các khối mã hóa là sự phân chia.

Trong các phương án, ví dụ, theo HEVC, đơn vị cây mã hóa (CTU-coding tree unit) có thể được phân chia thành các CU bằng cách sử dụng cấu trúc cây tứ phân được ký hiệu là cây mã hóa. Quyết định về mã vùng ảnh nhờ sử dụng dự đoán ảnh liên đới (theo thời gian) hay dự đoán nội ảnh (theo không gian) có thể được thực hiện tại mức CU. Mỗi CU có thể còn được phân chia thành một, hai, hoặc bốn PU theo loại phân chia PU. Bên trong một PU, xử lý dự đoán giống nhau được áp dụng và thông tin liên quan có thể được truyền tới bộ giải mã trên cơ sở PU. Sau khi thu nhận khối dư bằng cách áp dụng xử lý dự đoán dựa trên loại phân chia PU, CU có thể được phân chia thành các đơn vị biến đổi (TU) theo cấu trúc cây tứ phân khác tương tự cây mã hóa đối với CU.

Trong các phương án, ví dụ, theo tiêu chuẩn mã hóa video mới nhất hiện đang phát triển, mà được gọi là mã hóa video đa năng (VVC-Versatile Video Coding), việc phân chia cây nhị phân và cây tứ phân (QTBT) kết hợp là ví dụ được sử dụng để phân chia khối mã hóa. Trong cấu trúc khối QTBT, CU có thể có dạng hình vuông hoặc hình chữ nhật. Ví dụ, đơn vị cây mã hóa (CTU-coding tree unit) được phân chia đầu tiên bởi cấu trúc cây tứ phân. Các nút nhánh cây tứ phân được phân chia tiếp bởi cấu trúc cây nhị phân hoặc cây tam phân (hoặc bậc ba). Các nút nhánh cây phân chia được gọi là các đơn vị mã hóa (CU-coding unit), và phân đoạn này được sử dụng để xử lý dự đoán và biến đổi mà không cần bất kỳ phân chia tiếp. Điều này có nghĩa rằng CU, PU và TU có cùng kích cỡ khối trong cấu trúc khối mã hóa QTBT. Một cách song song, nhiều phân vùng, ví dụ, phân vùng cây tam phân có thể được sử dụng cùng với cấu trúc khối QTBT.

Trong một ví dụ, bộ lựa chọn chế độ 260 của bộ mã hóa video 20 có thể có

cấu trúc để thực hiện bất kỳ kết hợp của các kỹ thuật phân chia được mô tả ở đây.

Như được mô tả nêu trên, bộ mã hóa video 20 có cấu trúc để xác định hoặc lựa chọn chế độ dự đoán tốt nhất hoặc tối ưu từ tập hợp chế độ dự đoán (ví dụ, được xác định trước). Tập hợp các chế độ dự đoán có thể bao gồm, ví dụ, các chế độ dự đoán nội bộ và/hoặc các chế độ dự đoán liên đới.

Dự đoán nội bộ

Tập hợp chế độ dự đoán nội bộ có thể bao gồm 35 chế độ dự đoán nội bộ khác nhau, ví dụ, các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ có hướng, ví dụ như được định nghĩa trong HEVC, hoặc có thể bao gồm 67 chế độ dự đoán nội bộ khác nhau, ví dụ, các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ có hướng như được định nghĩa trong VVC.

Bộ dự đoán nội bộ 254 có cấu trúc để sử dụng các mẫu được khôi phục của các khối lân cận của cùng ảnh hiện tại để tạo ra khối dự đoán nội bộ 265 theo chế độ dự đoán nội bộ của tập hợp các chế độ dự đoán nội bộ.

Bộ dự đoán nội bộ 254 (hoặc nói chung là bộ lựa chọn chế độ 260) còn có cấu trúc để xuất ra các tham số dự đoán nội bộ (hoặc nói chung là thông tin chỉ báo chế độ dự đoán nội bộ được lựa chọn cho khối) tới bộ mã hóa entropy 270 dưới dạng các phần tử cú pháp 266 để đưa vào dữ liệu ảnh được mã hóa 21, sao cho, ví dụ, bộ giải mã video 30 có thể thu và sử dụng các tham số dự đoán để giải mã.

Dự đoán liên đới

Tập hợp của các chế độ dự đoán liên đới phụ thuộc vào các ảnh tham chiếu khả dụng (tức là các ảnh trước đó ít nhất đã được giải mã một phần, ví dụ được lưu trữ trong DBP 230) và các tham số dự đoán liên đới khác, ví dụ như toàn bộ ảnh tham chiếu hay chỉ một phần, ví dụ, vùng cửa sổ tìm kiếm xung quanh vùng của khối hiện tại, của ảnh tham chiếu được sử dụng để tìm kiếm khối tham chiếu phù hợp nhất và/hoặc ví dụ, nội suy điểm ảnh có được áp dụng hay không, ví dụ,

nửa/nửa điểm ảnh và/hoặc nội suy một phần tư điểm hay, hay không.

Ngoài các chế độ dự đoán nêu trên, chế độ bỏ qua và/hoặc chế độ trực tiếp có thể được áp dụng.

Bộ dự đoán liên đới 244 có thể bao gồm bộ ước lượng chuyển động (ME-motion estimation) và bộ bù chuyển động (MC-motion compensation) (cả hai không được thể hiện trên FIG.2). Bộ ước lượng chuyển động có thể có cấu trúc 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, tức là các khối được khôi phục của một hoặc nhiều các ảnh được giải mã trước đó khác/khác nhau 231, để ước lượng chuyển động. Ví dụ, chuỗi video có thể bao gồm ảnh hiện tại và các ảnh được giải mã trước đó 231, hay nói cách khác, ảnh hiện tại và các ảnh được giải mã trước đó 231 có thể là một phần của hoặc tạo thành chuỗi ảnh mà tạo thành chuỗi video.

Ví dụ, bộ mã hóa 20 có thể có cấu trúc để: lựa chọn khối tham chiếu từ các khối tham chiếu của cùng ảnh hoặc các ảnh khác nhau trong các ảnh khác, và cung cấp ảnh tham chiếu (hoặc chỉ số ảnh tham chiếu) và/hoặc độ dịch (độ dịch không gian) giữa vị trí (các tọa độ x-y) của khối tham chiếu và vị trí của khối hiện tại như là các tham số dự đoán liên đới tới bộ ước lượng chuyển động. Độ dịch này cũng được gọi là vector chuyển động (MV-motion vector).

Bộ bù chuyển động có cấu trúc để: thu nhận, ví dụ, thu tham số dự đoán liên đới, và thực hiện việc dự đoán liên đới dựa trên hoặc bằng cách sử dụng tham số dự đoán liên đới để thu nhận khối dự đoán liên đới 265. Việc bù chuyển động được thực hiện bởi bộ bù chuyển động có thể bao gồm truy xuất hoặc tạo ra khối dự đoán dựa trên vector chuyển động/khối được xác định bằng cách ước lượng chuyển động, có thể thực hiện các nội suy trên 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, nhờ đó tăng khả năng số lượng khối dự đoán ứng viên mà có thể được sử dụng để mã hóa khối ảnh. Sau khi thu vector chuyển động đối với PU của khối ảnh hiện tại, bộ bù chuyển động có thể xác định vị trí khối dự đoán mà các vector chuyển động

trở tới trong một trong số các danh sách ảnh tham chiếu.

Bộ bù chuyển động cũng có thể tạo ra các phần tử cú pháp được kết hợp với các khối và các lát video để bộ giải mã video 30 sử dụng trong việc giải mã các khối ảnh của lát video. Ngoài các lát và các phần tử cú pháp tương ứng, các nhóm ô và /hoặc các ô và các phần tử cú pháp tương ứng có thể được tạo ra hoặc được sử dụng.

Mã hóa Entropy

Bộ mã hóa entropy 270 có cấu trúc để áp dụng, ví dụ, phương pháp hoặc thuật toán mã hóa entropy (tức là phương pháp mã hóa độ dài biến thiên (VLC-variable length coding), phương pháp VLC thích nghi ngữ cảnh (CAVLC), phương pháp mã hóa số học, nhị phân hóa, mã hóa số học nhị phân thích nghi ngữ cảnh (CABAC), mã hóa số học nhị phân thích nghi ngữ cảnh dựa trên cú pháp (SBAC), mã hóa entropy phân chia khoảng cách xác suất (PIPE-probability interval partitioning entropy) hoặc phương pháp hoặc kỹ thuật mã hóa entropy khác) hoặc kỹ thuật đường vòng (không nén) trên các hệ số được lượng tử hóa 209, các tham số dự đoán liên đới, các tham số nội dự đoán, tham số lọc vòng và/hoặc các phần tử cú pháp khác để thu nhận dữ liệu ảnh được mã hóa 21 mà có thể được xuất ra thông qua bộ đầu ra 272, ví dụ dưới dạng của dòng bit được mã hóa 21, sao cho, ví dụ bộ giải mã video 30 có thể thu và sử dụng các tham số để giải mã. Dòng bit được mã hóa 21 có thể được truyền tới bộ giải mã video 30, hoặc có thể được lưu trữ trong bộ nhớ cho việc truyền tiếp theo hoặc được gọi ra 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 dựa trên không biến đổi 20 có thể lượng tử hóa trực tiếp tín hiệu dư mà không có bộ xử lý biến đổi 206 đối với một vài khối hoặc khung. Trong cách thức thực hiện khác, bộ mã hóa 20 có thể có bộ lượng tử hóa 208 và bộ lượng tử hóa ngược 210 mà được kết hợp thành một bộ phận.

Bộ giải mã và phương pháp giải mã

Fig.3 thể hiện ví dụ về bộ giải mã video 30 mà có cấu trúc để thực hiện các

kỹ thuật của sáng chế. Bộ giải mã video 30 có cấu trúc để thu dữ liệu ảnh được mã hóa 21 (tức là dòng bit được mã hóa 21), tức là 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 để 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 ô hoặc các ô) và các phần tử cú pháp được kết hợp.

Trong ví dụ của Fig.3, bộ giải mã 30 bao gồm bộ giải mã entropy 304, bộ lượng tử hóa ngược 310, bộ xử lý biến đổi ngược 312, bộ khôi phục 314 (ví dụ bộ cộng tổng 314), bộ lọc vòng 320, bộ đệm ảnh được giải mã (DBP) 330, bộ áp dụng chế độ 360, bộ dự đoán liên đới 344 và bộ dự đoán nội bộ 354. Bộ dự đoán liên đới 344 có thể là hoặc bao gồm bộ bù chuyển động. Trong một số ví dụ, bộ giải mã video 30 có thể thực hiện đường giải mã thường tương hỗ với đường mã hóa được mô tả liên quan đến bộ mã hóa video 100 từ Fig.2.

Như được giải thích đối với bộ mã hóa 20, bộ lượng tử hóa ngược 210, bộ xử lý biến đổi ngược 212, bộ khôi phục 214, bộ lọc vòng 220, bộ đệm ảnh được giải mã (DPB) 230, bộ dự đoán liên đới 344 và bộ dự đoán nội bộ 354 cũng tạo thành “bộ giải mã lắp trong” của bộ mã hóa video 20. Do đó, bộ lượng tử hóa ngược 310 có thể đồng nhất về chức năng với bộ lượng tử hóa ngược 110, bộ xử lý biến đổi ngược 312 có thể đồng nhất về chức năng với bộ xử lý biến đổi ngược 212, bộ khôi phục 314 có thể đồng nhất về chức năng với bộ khôi phục 214, bộ lọc vòng 320 có thể đồng nhất về chức năng với bộ lọc vòng 220, và bộ đệm ảnh được giải mã 330 có thể đồng nhất về chức năng với bộ đệm ảnh được giải mã 230. Do đó, các giải thích được mô tả cho các bộ phận và chức năng tương ứng của bộ mã hóa video 20 áp dụng tương ứng cho các bộ phận và chức năng tương ứng của bộ giải mã video 30.

Giải mã Entropy

Bộ giải mã entropy 304 có cấu trúc để phân tích dòng bit 21 (hoặc nói chung là dữ liệu ảnh được mã hóa 21) và thực hiện, ví dụ, việc giải mã entropy đối với dữ liệu ảnh được mã hóa 21 để thu nhận, ví dụ, các hệ số được lượng tử hóa 309

và/hoặc các tham số mã hóa được giải mã (không được thể hiện trên FIG.3), tức là bất kỳ hoặc tất cả tham số dự đoán liên đới (tức là chỉ số ảnh tham chiếu và vector chuyển động), tham chiếu dự đoán nội bộ (tức là chế độ hoặc chỉ số dự đoán nội bộ), các tham số biến đổi, các tham số lượng tử hóa, các tham số lọc vòng, và/hoặc các phần tử cú pháp khác. Bộ giải mã entropy 304 có thể có cấu trúc để áp dụng các thuật toán hoặc sơ đồ giải mã tương ứng với các sơ đồ mã hóa như được mô tả liên quan đến bộ mã hóa entropy 270 của bộ mã hóa 20. Bộ giải mã entropy 304 có thể còn có cấu trúc để cung cấp các tham số dự đoán liên đới, tham số dự đoán nội bộ và/hoặc các phần tử cú pháp khác tới bộ áp dụng chế độ 360 và các tham số khác tới các bộ phận khác của bộ giải mã 30. Bộ giải mã video 30 có thể thu các phần tử cú pháp tại cấp lát video và/hoặc cấp khối video. Ngoài các lát và các phần tử cú pháp tương ứng, các nhóm ô và/hoặc các ô và các phần tử cú pháp tương ứng có thể được thu và/hoặc được sử dụng.

Lượng tử hóa ngược

Bộ lượng tử hóa ngược 310 có thể có cấu trúc để thu các tham 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 từ dữ liệu ảnh được mã hóa 21 (ví dụ bằng cách phân tích và/hoặc giải lượng tử, ví dụ bởi bộ giải mã entropy 304) và để áp dụng, dựa trên các tham số lượng tử hóa, việc lượng tử hóa ngược trên các hệ số được lượng tử hóa được giải mã 309 để thu nhận các hệ số được lượng tử hóa 311, mà có thể cũng được gọi là các hệ số biến đổi 311. Xử lý lượng tử hóa ngược có thể bao gồm việc sử dụng tham số lượng tử hóa được xác định bởi bộ mã hóa video 20 đối với mỗi khối video trong lát video (hoặc ô hoặc nhóm ô) để xác định mức độ lượng tử hóa và loại tương tự, mức độ lượng tử hóa ngược mà sẽ được áp dụng.

Biến đổi ngược

Bộ xử lý biến đổi ngược 312 có thể có cấu trúc để thu các hệ số được giải lượng tử 311, cũng được gọi là các hệ số biến đổi 311, và để áp dụng biến đổi tới các hệ số được giải lượng tử 311 để thu nhận các khối dư được khôi phục 213 trong miền mẫu. Các khối dư được khôi phục 213 cũng có thể 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 nguyên ngược, hoặc xử lý biến đổi nghịch đảo tương tự về mặt khái niệm. Bộ xử lý biến đổi ngược 312 có thể còn có cấu trúc để thu các tham số biến đổi hoặc thông tin tương ứng từ dữ liệu ảnh được mã hóa 21 (ví dụ bằng cách phân tích và/hoặc giải mã, ví dụ bởi bộ giải mã entropy 304) để xác định biến đổi cần được áp dụng tới các hệ số được giải lượng tử 311.

Khôi phục

Bộ khôi phục 314 (tức là bộ cộng hoặc bộ cộng tổng 314) có thể có cấu trúc để cộng khối dư được khôi phục 313, tới khối dự đoán 365 để thu nhận khối được khôi phục 315 trong miền mẫu, tức là bằng cách cộng các giá trị mẫu của khối dư được khôi phục 313 và các giá trị mẫu của khối dự đoán 365.

Lọc

Bộ lọc vòng 320 (hoặc trong vòng lặp mã hóa hoặc sau vòng lặp mã hóa) có cấu trúc để lọc khối được khôi phục 315 để thu nhận khối được lọc 321, ví dụ để làm mượt sự chuyển tiếp điểm ảnh, hoặc cải thiện chất lượng video. Bộ lọc vòng 320 có thể bao gồm một hoặc nhiều bộ lọc vòng như bộ lọc giải khối, bộ lọc bù thích ứng mẫu (SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc song phương, bộ lọc vòng thích ứng (ALF), các bộ lọc san bằng, làm nhay hoặc bộ lọc cộng tác, hoặc bất kỳ sự kết hợp của chúng. Mặc dù bộ lọc vòng 320 được hiển thị trong Fig.3 là bộ lọc vòng nội bộ, trong các cấu trúc khác, bộ lọc vòng 320 có thể được thực hiện như bộ lọc vòng sau.

Bộ đệm ảnh được giải mã

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

Bộ giải mã 30 có cấu trúc để xuất ra ảnh được giải mã 311, tức là thông qua bộ đầu ra 312, để trình diễn hoặc quan sát tới người dùng.

Dự đoán

Bộ dự đoán liên đới 344 có thể đồng nhất với bộ dự đoán liên đới 244 (cụ thể, đồng nhất với bộ bù chuyển động) và bộ dự đoán nội bộ 354 có thể đồng nhất với bộ dự đoán liên đới 254 về chức năng, và thực hiện các quyết định phân chia hoặc phân vùng và dự đoán dựa trên các tham số phân chia và/hoặc dự đoán hoặc thông tin tương ứng thu được từ dữ liệu ảnh được mã hóa 21 (ví dụ bằng cách phân tích và/hoặc giải mã, ví dụ bởi bộ giải mã entropy 304). Bộ áp dụng chế độ 360 có thể có cấu trúc để thực hiện việc dự đoán (dự đoán nội bộ hoặc liên đới) theo khối dựa trên các ảnh được khôi phục, các khối hoặc các mẫu tương ứng (được lọc hoặc không được lọc) để thu nhận khối dự đoán 365.

Khi lát video được mã hóa như là lát được mã hóa nội bộ (I), bộ dự đoán nội bộ 354 của bộ áp dụng chế độ 360 có cấu trúc để tạo khối dự đoán 365 đối với khối ảnh của lát video hiện tại dựa trên chế độ dự đoán nội bộ được báo hiệu và dữ liệu từ các khối đã giải mã trước đó của ảnh hiện tại. Khi ảnh video được mã hóa như là lát được mã hóa liên đới (tức là, B, hoặc P), bộ dự đoán liên đới 344 (tức là bộ bù chuyển động) của bộ áp dụng chế độ 360 có cấu trúc để tạo ra các khối dự đoán 365 đối với khối video của lát video hiện tại dựa trên các vector chuyển động và các phần tử cú pháp khác thu được từ bộ giải mã entropy 304. Đối với việc dự đoán liên đới, các khối dự đoán có thể được tạo ra từ một trong số các ảnh tham chiếu trong một trong số các danh sách ảnh tham chiếu. Bộ giải mã video 30 có thể xây dựng các danh sách khung tham chiếu: danh sách 0 và danh sách 1 bằng cách sử dụng kỹ thuật xây dựng mặc định dựa trên các ảnh tham chiếu được lưu trữ trong DPB 330. Điều giống hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án mà sử dụng các nhóm ô (ví dụ, các nhóm ô video) và/hoặc các ô (ví dụ, các ô video) ngoài các lát (ví dụ, các lát video), ví dụ video có thể được mã hóa nhờ sử dụng các nhóm ô và/hoặc các ô I, P hoặc B.

Bộ áp dụng chế độ 360 có cấu trúc để xác định thông tin dự đoán đối với khối video của lát video hiện tại bằng cách phân tích vector chuyển động hoặc thông tin liên quan và các phần tử cú pháp khác, và sử dụng thông tin dự đoán để

tạo ra các khối dự đoán đối với khối video hiện tại đang được giải mã. Ví dụ, bộ áp dụng chế độ 360 sử dụng một vài phần tử cú pháp thu được, để xác định chế độ dự đoán (ví dụ, dự đoán nội bộ hoặc liên đới) được sử dụng để mã hóa các khối video của lát video, loại lớp dự đoán liên đới (ví dụ, lát B, lát P, hoặc lát GPB), thông tin xây dựng đối với một hoặc nhiều danh sách ảnh tham chiếu đối với lát, các vectơ chuyển động đối với mỗi khối video được mã hóa liên đới của lát, trạng thái dự đoán liên đới đối với mỗi khối video được mã hóa liên đới của lát, và thông tin khác, để giải mã các khối video trong lát video hiện tại. Điều giống hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án mà sử dụng các nhóm ô (ví dụ các nhóm ô video) và/hoặc các ô (ví dụ các ô video) ngoài các lát (ví dụ các lát video), ví dụ video có thể được mã hóa nhờ sử dụng các nhóm ô và/hoặc các ô I, P hoặc B.

Các phương án của bộ giải mã video 30 như được thể hiện trên FIG.3 có thể có cấu trúc để phân chia và/hoặc giải mã ảnh bằng cách sử dụng các lát (cũng được gọi là các lát video), trong đó ảnh có thể được phân chia hoặc được giải mã nhờ sử dụng một hoặc nhiều lát (thường không chồng lấn), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ các CTU).

Các phương án của bộ giải mã video 30 như được thể hiện trên FIG.3 có thể có cấu trúc để phân chia và/hoặc giải mã ảnh bằng cách sử dụng các nhóm ô (cũng được gọi là các nhóm ô video) và/hoặc các ô (cũng được gọi là các ô video), trong đó ảnh có thể được phân chia hoặc được giải mã nhờ sử dụng một hoặc nhiều nhóm ô (thường không xếp chồng), và mỗi nhóm ô có thể bao gồm, ví dụ một hoặc nhiều khối (tức là các CTU) hoặc một hoặc nhiều ô, trong đó mỗi ô, ví dụ, có thể có dạng chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ các khối hoàn chỉnh hoặc riêng 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 cần bộ lọc vòng 320. Ví dụ, bộ giải mã dựa trên không biến đổi 30 có thể lượng tử hóa ngược trực tiếp tín hiệu dư mà không có bộ xử lý biến đổi ngược

312 đối với một vài khối hoặc khung. Trong cách thức thực hiện khác, bộ giải mã video 30 có thể có bộ lượng tử hóa ngược 310 và bộ xử lý biến đổi ngược 312 được kết hợp thành một bộ phận.

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 được xuất tới bước tiếp theo. Ví dụ, sau khi lọc nội suy, việc thu nhận vector chuyển động hoặc lọc vòng, hoạt động khác, chẳng hạn như cắt hoặc dịch chuyển, có thể được thực hiện trên kết quả xử lý của việc lọc nội suy, thu nhận vector chuyển động hoặc lọc vòng.

Cần lưu ý rằng các hoạt động khác có thể được áp dụng tới các vector chuyển động thu được của khối hiện tại (bao gồm nhưng không giới hạn ở vector chuyển động điểm điều khiển trong chế độ affine, vector chuyển động khối con trong chế độ affine, phẳng, ATMVP, các vector chuyển động thời gian, v.v). Ví dụ, giá trị của vector chuyển động được giới hạn trong một phạm vi được xác định trước theo bit đại diện của nó. Nếu bit đại diện của vector chuyển động là bitDepth, thì phạm vi là $-2^{(bitDepth-1)} \sim 2^{(bitDepth-1)} - 1$, trong đó “^” có nghĩa là lũy thừa. Ví dụ, nếu bitDepth được thiết lập bằng 16, phạm vi là $-32768 \sim 32767$; nếu bitDepth được thiết lập bằng 18, phạm vi là $-131072 \sim 131071$. Ví dụ, giá trị của vector chuyển động thu được (ví dụ, các MV của bốn khối con 4x4 trong một khối 8x8) bị giới hạn sao cho độ chênh lệch lớn nhất giữa các phân nguyên của bốn MV khối con 4x4 không lớn hơn N điểm ảnh, như không quá 1 điểm ảnh. Ở đây cung cấp hai phương pháp để giới hạn vector chuyển động theo bitDepth.

Phương pháp 1: Loại bỏ MSB (bit quan trọng nhất-most significant bit) tràn dòng bằng các toán tử

$$ux = (mvx + 2^{bitDepth}) \% 2^{bitDepth} \quad (1)$$

$$mvx = (ux \geq 2^{bitDepth-1}) ? (ux - 2^{bitDepth}) : ux \quad (2)$$

$$uy = (mvy + 2^{bitDepth}) \% 2^{bitDepth} \quad (3)$$

$$mvy = (uy \geq 2^{bitDepth-1}) ? (uy - 2^{bitDepth}) : uy \quad (4)$$

trong đó mvx là thành phần chiều ngang của vector chuyển động của khối ảnh hoặc

khối con, mvy là thành phần chiều dọc của vector chuyển động của khối ảnh hoặc khối con và ux và uy chỉ báo giá trị trung gian ;

Ví dụ, nếu giá trị của mvx là -32769, sau khi áp dụng công thức (1) và (2), giá trị kết quả là 32767. Trong hệ thống máy tính, số thập phân được lưu trữ như là dạng phân bù của hai. Phân bù của hai -32769 là 1,0111,1111,1111,1111 (17 bit), sau đó MSB bị loại bỏ, do đó phân bù của hai thu được là 0111,1111,1111,1111 (số thập phân là 32767), mà tương tự như đầu ra bằng cách áp dụng công thức (1) và (2).

$$ux = (mvpx + mvdx + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (5)$$

$$mvx = (ux \geq 2^{\text{bitDepth}-1}) ? (ux - 2^{\text{bitDepth}}) : ux \quad (6)$$

$$uy = (mvpv + mvdy + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (7)$$

$$mvy = (uy \geq 2^{\text{bitDepth}-1}) ? (uy - 2^{\text{bitDepth}}) : uy \quad (8)$$

Các phép toán có thể được áp dụng trong tổng.mvp và mvd, như được thể hiện trong công thức (5) đến (8).

Phương pháp 2: Loại bỏ MSB tràn dòng bằng cách cắt giá trị

$$vx = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1} - 1, vx)$$

$$vy = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1} - 1, vy)$$

trong đó vx là thành phần chiều ngang của vector chuyển động của khối ảnh hoặc khối con, vy là thành phần chiều dọc của vector chuyển động của khối ảnh hoặc khối con; x, y và z tương ứng với ba giá trị đầu vào của xử lý cắt MV và định nghĩa của hàm Clip3 là như sau:

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

FIG.4 là sơ đồ giản lược của thiết bị mã hóa video 400 theo phương án của sáng chế. Thiết bị mã hóa video 400 là thích hợp để thực hiện các phương án được bộc lộ như được mô tả ở đây. Theo phương án của sáng chế, thiết bị mã hóa video 400 có thể là bộ giải mã như bộ giải mã video 30 của FIG.1A hoặc bộ mã hóa như

bộ mã hóa video 20 của FIG.1A.

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

Bộ xử lý 430 được thực hiện bởi phần cứng và phần mềm. Bộ xử lý 430 có thể được thực hiện dưới dạng một hoặc nhiều chip CPU, lõi (ví dụ, như bộ xử lý đa lõi), các FPGA, ASIC và DSP. Bộ xử lý 430 truyền thông với các cổng vào 410, bộ thu 420, bộ phát 440, cổng ra 450 và bộ nhớ 460. Bộ xử lý 430 bao gồm môđun mã hóa 470. Môđun mã hóa 470 thực hiện các phương án được bộc lộ như được mô tả nêu trên. Ví dụ, môđun mã hóa 470 thực hiện, xử lý, chuẩn bị hoặc cung cấp các hoạt động mã hóa khác nhau. Do đó, việc bao gồm môđun mã hóa 470 cung cấp cải tiến đáng kể đối với chức năng của thiết bị mã hóa video 400 và tạo hiệu ứng biến đổi thiết bị mã hóa video 400 thành trạng thái khác. Ngoài ra, môđun mã hóa 470 được thực hiện như các lệnh được lưu trữ trong bộ nhớ 460 và được thực thi bởi bộ xử lý 430.

Bộ nhớ 460 có thể bao gồm một hoặc nhiều đĩa, ổ băng và ổ bán dẫn và có thể được sử dụng như một thiết bị lưu trữ dữ liệu tràn dòng, để lưu trữ các chương trình khi các chương trình này được lựa chọn để thực thi và để lưu trữ các lệnh và dữ liệu được đọc trong quá trình thực thi chương trình. Ví dụ, bộ nhớ 460 có thể là bộ nhớ bất biến và/hoặc không bất biến và có thể là bộ nhớ chỉ đọc (ROM-read-only memory), bộ nhớ truy cập ngẫu nhiên (RAM-random access memory), bộ nhớ định địa chỉ nội dung bậc ba (TCAM-ternary content-addressable memory) và/hoặc bộ nhớ truy nhập ngẫu nhiên tĩnh (SRAM-static random-access memory).

Fig.5 là sơ đồ khối giản lược của thiết bị **500** mà có thể được sử dụng như

là thiết bị nguồn 12 hoặc thiết bị đích 14 hoặc cả hai thiết bị từ Fig.1 theo phương án ví dụ.

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

Bộ nhớ 504 trong thiết bị 500 có thể là thiết bị bộ nhớ chỉ đọc (ROM) hoặc thiết bị bộ nhớ truy cập ngẫu nhiên (RAM) trong quá trình thực hiện. Bất kỳ loại thiết bị lưu trữ thích hợp khác đều có thể được sử dụng làm bộ nhớ 504. Bộ nhớ 504 có thể bao gồm mã và dữ liệu 506 mà được truy nhập bởi bộ xử lý 502 bằng cách sử dụng kênh truyền 512. Bộ nhớ 504 có thể bao gồm thêm 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, các chương trình này bao gồm thêm ứng dụng mã hóa video mà thực hiện các phương pháp được mô tả ở đây.

Thiết bị 500 cũng có thể bao gồm một hoặc nhiều thiết bị đầu ra, chẳng hạn như màn hình 518. Trong một ví dụ, màn hình 518 có thể là màn hình cảm ứng chạm mà kết hợp màn hình với phần tử cảm ứng chạm mà có thể hoạt động để cảm nhận đầu vào chạm. Màn hình 518 có thể được ghép nối với bộ xử lý 502 thông qua kênh truyền 512.

Mặc dù được mô tả ở đây là một kênh truyền, kênh truyền 512 của thiết bị 500 có thể bao gồm nhiều kênh truyền. Ngoài ra, bộ lưu trữ thứ cấp 514 có thể được ghép nối trực tiếp với các thành phần khác của thiết bị 500 hoặc có thể được truy nhập qua mạng và có thể bao gồm một bộ tích hợp như thẻ nhớ hoặc nhiều bộ phận như nhiều thẻ nhớ. Do đó, thiết bị 500 có thể được thực hiện với nhiều cấu hình khác nhau.

Như được đề cập trong tài liệu của J.M. Boyce “Dự đoán theo trọng số trong tiêu chuẩn mã hóa video H.264/MPEG AVC”, hội nghị chuyên đề quốc tế IEEE về các mạch và hệ thống, tháng 5 năm 2004, Canada, trang 789-792, Dự đoán theo trọng số (WP) là công cụ, đặc biệt hữu ích để mã hóa các sự chuyển tiếp. Công cụ dự đoán theo trọng số (WP) đã được áp dụng trong tiêu chuẩn mã hóa video H.264 trong cấu hình chính và mở rộng để cải thiện hiệu quả mã hóa bằng cách áp dụng hệ số trọng số nhân và độ dịch cộng vào dự đoán được bù chuyển động để tạo ra việc dự đoán theo trọng số. Trong chế độ rõ ràng, hệ số trọng số và độ dịch có thể được mã hóa trong phần tiêu đề lát đối với mỗi chỉ số ảnh tham chiếu có thể cho phép. Trong chế độ ẩn, các hệ số trọng số không được mã hóa mà được thu nhận dựa trên khoảng cách số đếm thứ tự ảnh (POC) tương đối của hai ảnh tham chiếu. Các kết quả thử nghiệm được cung cấp mà đo lường sự cải thiện hiệu quả mã hóa bằng cách sử dụng WP. Khi mã hóa các chuỗi từ chuyển tiếp sang màu đen, mức giảm tốc độ bit lên đến 67% có thể đạt được.

Khi được áp dụng cho một dự đoán, như trong các ảnh P, WP là tương tự như dự đoán rõ, đã được đề xuất trước đó để phục hồi lỗi. Dự đoán rõ trở thành một trường hợp đặc biệt của WP, với hệ số biến đổi tỷ lệ được giới hạn trong phạm vi $0 \leq \alpha \leq 1$. WP H.264 cho phép hệ số biến đổi tỷ lệ âm và hệ số biến đổi tỷ lệ lớn hơn một. Hệ số trọng số được áp dụng từng điểm ảnh bằng cách sử dụng trường nhân được mã hóa, để nén hiệu quả các vùng được phủ và không được phủ. Khác biệt chính của công cụ WP của H.264 so với các đề xuất trước đó liên quan đến dự đoán theo trọng số đối với hiệu quả nén là sự kết hợp của chỉ số ảnh tham chiếu với các tham số hệ số trọng số, mà cho phép báo hiệu hiệu quả các tham số này trong môi trường đa ảnh tham chiếu. Như được mô tả trong bài viết của R. Zhang và G. Cote “ước tính tham số chính xác và dò tìm sự chuyển tiếp hiệu quả đối với việc dự đoán theo trọng số trong nén video H.264”, hội nghị quốc tế IEEE lần thứ 15 về xử lý ảnh, tháng 10 năm 2008, San Diego, California, US, trang 2836-2839, thủ tục áp dụng WP trong hệ thống mã hóa thời gian thực có thể được chính thức hóa như là chuỗi các bước được thể hiện trong Fig 6. Đầu tiên, một vài số liệu thống kê 611 được tạo ra thông qua phân tích video 610. Số liệu

thống kê 611 trong cửa sổ nhỏ, từ một số ảnh trước đó cho đến hình ảnh hiện tại, sau đó được sử dụng để phát hiện sự chuyển tiếp. Mỗi ảnh được gán một giá trị trạng thái 631 mà chỉ báo ảnh đang ở trạng thái thường hay ở trạng thái chuyển tiếp. Các giá trị trạng thái này được lưu cho mỗi ảnh. Khi mã hóa ảnh, nếu có trạng thái chuyển tiếp (FADE) trong ảnh hiện tại hoặc một trong các ảnh tham chiếu của nó, WP sẽ được sử dụng cho cặp tham chiếu hiện tại này và số liệu thống kê về ảnh hiện tại và ảnh tham chiếu tương ứng được xử lý ở bước 650 để ước tính các thông số WP. Các tham số này sau đó được chuyển đến cơ cấu mã hóa 660. Nếu không, quá trình mã hóa bình thường được thực hiện.

Như đã đề cập trong bài báo của A. Leontaris và AM Tourapis “Các phương pháp dự đoán theo trọng số để cải thiện khả năng bù chuyển động”, Hội nghị quốc tế IEEE lần thứ 16 về xử lý hình ảnh (ICIP), tháng 11 năm 2009, Cairo, Ai Cập, trang 1029-1032, khối macro trong H.264 được chia thành các phân vùng khối macro. Đối với mỗi phân vùng khối macro, việc tham chiếu được chọn từ mỗi danh sách tham chiếu khả dụng (thường được ký hiệu trong tiêu chuẩn kỹ thuật là RefPicList), danh sách 0 cho các lát được mã hóa P hoặc B hoặc danh sách tham chiếu 1 đối với các lát được mã hóa B. Các tham chiếu được sử dụng có thể khác nhau đối với mỗi phân vùng. Nhờ sử dụng các tham chiếu này, khối dự đoán được tạo cho mỗi danh sách, tức là P đối với dự đoán danh sách đơn và P_0 và P_1 đối với dự đoán kép, nhờ sử dụng thông tin chuyển động với, một cách tùy chọn, độ chính xác điểm ảnh con. Các khối dự đoán có thể được xử lý thêm tùy thuộc vào tính khả dụng của dự đoán theo trọng số đối với lát hiện tại. Đối với các lát P, các tham số WP được truyền tại thông tin tiêu đề lát. Đối với lát B, có hai lựa chọn. Trong WP rõ ràng, các tham số được truyền trong thông tin tiêu đề lát và trong WP ẩn, các tham số được thu nhận dựa trên số đếm thứ tự ảnh (POC-Picture Order Count) được báo hiệu trong thông tin tiêu đề lát. Trong bài này, chúng tôi sẽ chỉ tập trung vào WP rõ ràng và phương pháp này có thể được sử dụng như thế nào để cải thiện khả năng bù chuyển động. Lưu ý rằng trong HEVC và VVC, PB được sử dụng tương tự như phân vùng khối macro trong AVC.

Đối với các lát P hoặc WP rõ ràng của danh sách đơn trong các lát B, khối dự đoán được rút ra từ tham chiếu đơn. Gọi p là một giá trị mẫu trong khối dự đoán P . Nếu dự đoán theo trọng số không được sử dụng, thì mẫu dự đoán liên đới cuối cùng là $f=p$. Nếu không phải, mẫu được dự đoán là:

$$f = \begin{cases} ((p \times w_x + 2^{\log WD-1}) \gg \log WD) + o_x, & \log WD \geq 1 \\ (p \times w_x + o_x), & \text{khác} \end{cases}$$

Các thuật ngữ w_x và o_x là các tham số độ dịch và độ khuếch đại WP đối với danh sách tham chiếu x . Thuật ngữ $\log WD$ được truyền trong dòng bit và điều khiển độ chính xác toán học của xử lý dự đoán theo trọng số. Đối với $\log WD \geq 1$, biểu thức nêu trên làm tròn khác 0. Tương tự, đối với dự đoán kép, hai khối dự đoán, một khối cho mỗi danh sách tham chiếu, được xem xét. p_0 và p_1 ký hiệu các mẫu trong mỗi khối trong số hai khối dự đoán P_0 và P_1 . Nếu dự đoán theo trọng số không được sử dụng, việc dự đoán được thực hiện như sau:

$$f = (p_0 + p_1 + 1) \gg 1.$$

Đối với dự đoán kép theo trọng số, việc dự đoán được thực hiện như sau:

$$f = \left(\left(p_0 \times w_0 + p_1 \times w_1 + 2^{\log WD} \right) \gg (\log WD + 1) \right) + ((o_0 + o_1 + 1) \gg 1)$$

Cần lưu ý rằng dự đoán theo trọng số có thể bù cho những thay đổi về ánh sáng, chẳng hạn như tăng dần, giảm dần hoặc chuyển tiếp chéo.

Ở mức cao trong VVC, việc dự đoán theo trọng số được báo hiệu trong SPS, PPS và thông tin tiêu đề lát. Trong SPS, các phần tử cú pháp sau được sử dụng cho điều này:

- `sps_weighted_pred_flag` bằng 1 chỉ rõ rằng việc dự đoán theo trọng số có thể được áp dụng tới các lát P có viện dẫn tới SPS. `sps_weighted_pred_flag` bằng 0 chỉ rõ rằng việc dự đoán theo trọng số không được áp dụng tới các lát P có viện dẫn tới SPS;
- `sps_weighted_bipred_flag` bằng 1 chỉ rõ rằng việc dự đoán theo trọng số rõ

ràng có thể được áp dụng tới các lát B có viện dẫn tới SPS. `sps_weighted_bipred_flag` bằng 0 chỉ rõ rằng việc dự đoán theo trọng số rõ ràng không được áp dụng tới các lát B có viện dẫn tới SPS.

Trong PPS, các phần tử cú pháp sau được sử dụng cho điều này:

- `pps_weighted_pred_flag` bằng 0 chỉ rõ rằng việc dự đoán theo trọng số không được áp dụng tới các lát P có viện dẫn đến PPS. `pps_weighted_pred_flag` bằng 1 chỉ rõ rằng việc dự đoán theo trọng số được áp dụng tới các lát P có viện dẫn đến PPS. Khi `sps_weighted_pred_flag` bằng 0, giá trị của `pps_weighted_pred_flag` sẽ bằng 0;
- `pps_weighted_bipred_flag` bằng 0 chỉ rõ rằng việc dự đoán theo trọng số rõ ràng không được áp dụng tới các lát B có viện dẫn đến PPS. `pps_weighted_bipred_flag` bằng 1 chỉ rõ rằng việc dự đoán theo trọng số rõ ràng được áp dụng tới các lát B có viện dẫn đến PPS. Khi `sps_weighted_bipred_flag` bằng 0, giá trị của `pps_weighted_bipred_flag` sẽ bằng 0.

Trong thông tin tiêu đề lát, các tham số dự đoán theo trọng số được báo hiệu là `pred_weight_table( )` có cấu trúc như trong Bảng 1 và chứa các phần tử sau đây:

`luma_log2_weight_denom` là logarit cơ số 2 của mẫu số cho tất cả các hệ số trọng số độ chói. Giá trị của `luma_log2_weight_denom` sẽ nằm trong đoạn từ 0 đến 7.

`delta_chroma_log2_weight_denom` là độ chênh lệch của logarit cơ số 2 của mẫu số đối với tất cả các hệ số trọng số sắc độ. Khi `delta_chroma_log2_weight_denom` không được hiện diện, tham số này được xem là bằng 0.

variable `ChromaLog2WeightDenom` được thu nhận bằng `luma_log2_weight_denom + delta_chroma_log2_weight_denom` và giá trị này sẽ nằm trong đoạn từ 0 đến 7.

$\text{luma_weight_10_flag}[i]$ bằng 1 chỉ rõ rằng các hệ số trọng số đối với thành phần độ chói của dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$ được hiện diện. $\text{luma_weight_10_flag}[i]$ bằng 0 chỉ rõ rằng các hệ số trọng số này không được hiện diện.

$\text{chroma_weight_10_flag}[i]$ bằng 1 chỉ rõ rằng các hệ số trọng số đối với các giá trị dự đoán sắc độ của dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$ được hiện diện. $\text{chroma_weight_10_flag}[i]$ bằng 0 chỉ rõ rằng các hệ số trọng số này không được hiện diện. Khi $\text{chroma_weight_10_flag}[i]$ không được hiện diện, tham số này được xem là bằng 0.

$\text{delta_luma_weight_10}[i]$ là độ chênh lệch của hệ số trọng số được áp dụng tới giá trị dự đoán độ chói đối với dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$.

Biến $\text{LumaWeightL0}[i]$ được thu nhận bằng $(1 \ll \text{luma_log2_weight_denom}) + \text{delta_luma_weight_10}[i]$. Khi $\text{luma_weight_10_flag}[i]$ bằng 1, giá trị của $\text{delta_luma_weight_10}[i]$ sẽ nằm trong đoạn từ -128 đến 127. Khi $\text{luma_weight_10_flag}[i]$ bằng 0, $\text{LumaWeightL0}[i]$ được xem là bằng $2^{\text{luma_log2_weight_denom}}$.

$\text{luma_offset_10}[i]$ là độ dịch cộng được áp dụng tới giá trị dự đoán độ chói đối với dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$. Giá trị của $\text{luma_offset_10}[i]$ sẽ nằm trong đoạn từ -128 đến 127. Khi $\text{luma_weight_10_flag}[i]$ bằng 0, $\text{luma_offset_10}[i]$ được xem là bằng 0.

$\text{delta_chroma_weight_10}[i][j]$ là độ chênh lệch của hệ số trọng số được áp dụng tới các giá trị dự đoán sắc độ đối với dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$ với j bằng 0 đối với Cb và j bằng 1 đối với Cr.

Biến $\text{ChromaWeightL0}[i][j]$ được thu nhận bằng $(1 \ll \text{ChromaLog2WeightDenom}) + \text{delta_chroma_weight_10}[i][j]$. Khi $\text{chroma_weight_10_flag}[i]$ bằng 1, giá trị của $\text{delta_chroma_weight_10}[i][j]$ sẽ nằm trong đoạn từ -128 đến 127. Khi $\text{chroma_weight_10_flag}[i]$ bằng 0, $\text{ChromaWeightL0}[i][j]$ được xem là bằng $2^{\text{ChromaLog2WeightDenom}}$.

$\text{delta_chroma_offset_l0}[i][j]$ là độ chênh lệch của độ dịch cộng được áp dụng tới các giá trị dự đoán sắc độ đối với dự đoán danh sách 0 nhờ sử dụng $\text{RefPicList}[0][i]$ với j bằng 0 đối với Cb và j bằng 1 đối với Cr.

Biến $\text{ChromaOffsetL0}[i][j]$ được thu nhận như sau:

$$\begin{aligned} \text{ChromaOffsetL0}[i][j] = & \text{Clip3}(-128, 127, \\ & (128 + \text{delta_chroma_offset_l0}[i][j] - \\ & ((128 * \text{ChromaWeightL0}[i][j]) \\ & \text{ChromaLog2WeightDenom}))) \end{aligned} \quad \gg$$

Giá trị của $\text{delta_chroma_offset_l0}[i][j]$ sẽ nằm trong đoạn từ $-4 * 128$ đến $4 * 127$. Khi $\text{chroma_weight_l0_flag}[i]$ bằng 0, $\text{ChromaOffsetL0}[i][j]$ được xem là bằng 0.

$\text{luma_weight_l1_flag}[i]$, $\text{chroma_weight_l1_flag}[i]$, $\text{delta_luma_weight_l1}[i]$, $\text{luma_offset_l1}[i]$, $\text{delta_chroma_weight_l1}[i][j]$ và $\text{delta_chroma_offset_l1}[i][j]$ có cùng ngữ nghĩa học như $\text{luma_weight_l0_flag}[i]$, $\text{chroma_weight_l0_flag}[i]$, $\text{delta_luma_weight_l0}[i]$, $\text{luma_offset_l0}[i]$, $\text{delta_chroma_weight_l0}[i][j]$ và $\text{delta_chroma_offset_l0}[i][j]$, một cách lần lượt, với l0, L0, danh sách 0 và List0 được thay thế bởi l1, L1, danh sách 1 và List1, một cách lần lượt.

Biến sumWeightL0Flags được thu nhận bằng tổng của $\text{luma_weight_l0_flag}[i] + 2 * \text{chroma_weight_l0_flag}[i]$, với $i = 0.. \text{NumRefIdxActive}[0] - 1$.

Khi slice_type bằng B, biến sumWeightL1Flags được thu nhận bằng tổng của $\text{luma_weight_l1_flag}[i] + 2 * \text{chroma_weight_l1_flag}[i]$, với $i = 0.. \text{NumRefIdxActive}[1] - 1$.

Yêu cầu về sự tương thích dòng bit là khi slice_type bằng P, sumWeightL0Flags sẽ nhỏ hơn hoặc bằng 24 và khi slice_type bằng B, tổng của sumWeightL0Flags và sumWeightL1Flags sẽ nhỏ hơn hoặc bằng 24.

Bảng 1. Cú pháp các tham số dự đoán theo trọng số

pred_weight_table() {	Mô tả
luma_log2_weight_denom	ue(v)
if(ChromaArrayType != 0)	
delta_chroma_log2_weight_denom	se(v)
for(i = 0; i < NumRefIdxActive[0]; i++)	
luma_weight_l0_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumRefIdxActive[0]; i++)	
chroma_weight_l0_flag[i]	u(1)
for(i = 0; i < NumRefIdxActive[0]; i++) {	
if(luma_weight_l0_flag[i]) {	
delta_luma_weight_l0[i]	se(v)
luma_offset_l0[i]	se(v)
}	
if(chroma_weight_l0_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l0[i][j]	se(v)
delta_chroma_offset_l0[i][j]	se(v)
}	
}	
if(slice_type == B) {	
for(i = 0; i < NumRefIdxActive[1]; i++)	
luma_weight_l1_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumRefIdxActive[1]; i++)	

chroma_weight_l1_flag[i]	u(1)
for(i = 0; i < NumRefIdxActive[1]; i++) {	
if(luma_weight_l1_flag[i]) {	
delta_luma_weight_l1[i]	se(v)
luma_offset_l1[i]	se(v)
}	
if(chroma_weight_l1_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l1[i][j]	se(v)
delta_chroma_offset_l1[i][j]	se(v)
}	
}	
}	
}	

Trong phần JVET-O0244 (V. Seregin và cộng sự “AHG17: đen-ta POC 0 trong cấu trúc ảnh tham chiếu,” hội thảo JVET thứ 15 tại, Gothenburg, Thụy Điển), được chỉ ra rằng trong dự thảo tiêu chuẩn kỹ thuật VVC hiện tại, các ảnh tham chiếu được báo hiệu trong cấu trúc ảnh tham chiếu (RPS), trong đó `abs_delta_poc_st` biểu diễn giá trị đen-ta POC, mà có thể bằng 0. RPS có thể được báo hiệu trong SPS và thông tin tiêu đề lát. Chức năng này là cần thiết để báo hiệu các trọng số khác nhau đối với cùng ảnh tham chiếu và có khả năng cần thiết nếu khả năng mở rộng phân lớp được hỗ trợ với cùng các giá trị POC được sử dụng trên các lớp trong đơn vị truy nhập. Trong đó, có quy định rằng các ảnh tham chiếu lặp lại là không cần thiết khi dự đoán theo trọng số không được cho phép. Trong số đó, trong phần này, có đề xuất không cho phép các giá trị đen-ta POC 0 khi việc dự đoán theo trọng số không được cho phép.

Bảng 2. Cú pháp RBSP (Raw Byte Sequence Payload-Tải trọng chuỗi byte thô) Tập hợp tham số chuỗi

<code>seq_parameter_set_rbsp() {</code>	Mô tả
<code>sps_decoding_parameter_set_id</code>	u(4)
<code>sps_video_parameter_set_id</code>	u(4)
<code>sps_max_sub_layers_minus1</code>	u(3)
<code>sps_reserved_zero_5bits</code>	u(5)
<code>profile_tier_level(sps_max_sub_layers_minus1)</code>	
<code>gdr_enabled_flag</code>	u(1)
<code>sps_seq_parameter_set_id</code>	ue(v)
<code>chroma_format_idc</code>	ue(v)
<code>if(chroma_format_idc == 3)</code>	
<code>separate_colour_plane_flag</code>	u(1)
<code>pic_width_max_in_luma_samples</code>	ue(v)
<code>pic_height_max_in_luma_samples</code>	ue(v)
<code>subpics_present_flag</code>	u(1)
<code>if(subpics_present_flag) {</code>	
<code>max_subpics_minus1</code>	u(8)
<code>subpic_grid_col_width_minus1</code>	u(v)
<code>subpic_grid_row_height_minus1</code>	u(v)
<code>for(i = 0; i < NumSubPicGridRows; i++)</code>	
<code>for(j = 0; j < NumSubPicGridCols; j++)</code>	
<code>subpic_grid_idx[i][j]</code>	u(v)

for(i = 0; i <= NumSubPics; i++) {	
subpic_treated_as_pic_flag[i]	u(1)
loop_filter_across_subpic_enabled_flag[i]	u(1)
}	
}	
bit_depth_luma_minus8	ue(v)
bit_depth_chroma_minus8	ue(v)
min_qp_prime_ts_minus4	ue(v)
log2_max_pic_order_cnt_lsb_minus4	ue(v)
if(sps_max_sub_layers_minus1 > 0)	
sps_sub_layer_ordering_info_present_flag	u(1)
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 :sps_max_sub_layers_minus1); i <= sps_max_sub_layers_minus1; i++) {	
sps_max_dec_pic_buffering_minus1[i]	ue(v)
sps_max_num_reorder_pics[i]	ue(v)
sps_max_latency_increase_plus1[i]	ue(v)
}	
long_term_ref_pics_flag	u(1)
inter_layer_ref_pics_present_flag	u(1)
sps_idr_rpl_present_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)	
}	
if(ChromaArrayType != 0)	
qtbtt_dual_tree_intra_flag	u(1)
log2_ctu_size_minus5	u(2)
log2_min_luma_coding_block_size_minus2	ue(v)
partition_constraints_override_enabled_flag	u(1)
sps_log2_diff_min_qt_min_cb_intra_slice_luma	ue(v)
sps_log2_diff_min_qt_min_cb_inter_slice	ue(v)
sps_max_mtt_hierarchy_depth_inter_slice	ue(v)
sps_max_mtt_hierarchy_depth_intra_slice_luma	ue(v)
if(sps_max_mtt_hierarchy_depth_intra_slice_luma != 0) {	
sps_log2_diff_max_bt_min_qt_intra_slice_luma	ue(v)
sps_log2_diff_max_tt_min_qt_intra_slice_luma	ue(v)
}	
if(sps_max_mtt_hierarchy_depth_inter_slices != 0) {	
sps_log2_diff_max_bt_min_qt_inter_slice	ue(v)
sps_log2_diff_max_tt_min_qt_inter_slice	ue(v)
}	
if(qtbtt_dual_tree_intra_flag) {	
sps_log2_diff_min_qt_min_cb_intra_slice_chroma	ue(v)
sps_max_mtt_hierarchy_depth_intra_slice_chroma	ue(v)

if (sps_max_mtt_hierarchy_depth_intra_slice_chroma != 0) {	
sps_log2_diff_max_bt_min_qt_intra_slice_chroma	ue(v)
sps_log2_diff_max_tt_min_qt_intra_slice_chroma	ue(v)
}	
}	
sps_max_luma_transform_size_64_flag	u(1)
if(ChromaArrayType != 0) {	
same_qp_table_for_chroma	u(1)
for(i = 0; i < same_qp_table_for_chroma ? 1 : 3; i++) {	
num_points_in_qp_table_minus1[i]	ue(v)
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {	
delta_qp_in_val_minus1[i][j]	ue(v)
delta_qp_out_val[i][j]	ue(v)
}	
}	
}	
sps_weighted_pred_flag	u(1)
sps_weighted_bipred_flag	u(1)
sps_sao_enabled_flag	u(1)
sps_alf_enabled_flag	u(1)
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)

sps_joint_cbr_enabled_flag	u(1)
sps_ref_wraparound_enabled_flag	u(1)
if(sps_ref_wraparound_enabled_flag)	
sps_ref_wraparound_offset_minus1	ue(v)
sps_temporal_mvp_enabled_flag	u(1)
if(sps_temporal_mvp_enabled_flag)	
sps_sbtmvp_enabled_flag	u(1)
sps_amvr_enabled_flag	u(1)
sps_bdof_enabled_flag	u(1)
sps_smvd_enabled_flag	u(1)
sps_dmvr_enabled_flag	u(1)
if(sps_bdof_enabled_flag sps_dmvr_enabled_flag)	
sps_bdof_dmvr_slice_present_flag	u(1)
sps_mmvd_enabled_flag	u(1)
sps_isp_enabled_flag	u(1)
sps_mrl_enabled_flag	u(1)
sps_mip_enabled_flag	u(1)
if(ChromaArrayType != 0)	
sps_cclm_enabled_flag	u(1)
if(sps_cclm_enabled_flag && chroma_format_idc == 1)	
sps_cclm_colocated_chroma_flag	u(1)
sps_mts_enabled_flag	u(1)
if(sps_mts_enabled_flag) {	

sps_explicit_mts_intra_enabled_flag	u(1)
sps_explicit_mts_inter_enabled_flag	u(1)
}	
sps_sbt_enabled_flag	u(1)
if(sps_sbt_enabled_flag)	
sps_sbt_max_size_64_flag	u(1)
sps_affine_enabled_flag	u(1)
if(sps_affine_enabled_flag) {	
sps_affine_type_flag	u(1)
sps_affine_amvr_enabled_flag	u(1)
sps_affine_prof_enabled_flag	u(1)
}	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
sps_bcw_enabled_flag	u(1)
sps_ibc_enabled_flag	u(1)
sps_ciip_enabled_flag	u(1)
if(sps_mmvd_enabled_flag)	
sps_fpel_mmvd_enabled_flag	u(1)
sps_triangle_enabled_flag	u(1)
sps_lmcs_enabled_flag	u(1)
sps_lfnst_enabled_flag	u(1)
sps_ladf_enabled_flag	u(1)

if (sps_ladf_enabled_flag) {	
sps_num_ladf_intervals_minus2	u(2)
sps_ladf_lowest_interval_qp_offset	se(v)
for(i = 0; i < sps_num_ladf_intervals_minus2 + 1; i++) {	
sps_ladf_qp_offset[i]	se(v)
sps_ladf_delta_threshold_minus1[i]	ue(v)
}	
}	
sps_scaling_list_enabled_flag	u(1)
hrd_parameters_present_flag	u(1)
if(general_hrd_parameters_present_flag) {	
num_units_in_tick	u(32)
time_scale	u(32)
sub_layer_cpb_parameters_present_flag	u(1)
if(sub_layer_cpb_parameters_present_flag)	
general_hrd_parameters(0, sps_max_sub_layers_minus1)	
else	
general_hrd_parameters(sps_max_sub_layers_minus1, sps_max_sub_layers_minus1)	
}	
vui_parameters_present_flag	u(1)
if(vui_parameters_present_flag)	
vui_parameters()	

sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Bảng 3. Cú pháp cấu trúc danh sách ảnh tham chiếu

ref_pic_list_struct(listIdx, rplsIdx) {	Mô tả
num_ref_entries [listIdx][rplsIdx]	ue(v)
if(long_term_ref_pics_flag)	
ltp_in_slice_header_flag [listIdx][rplsIdx]	u(1)
for(i = 0, j = 0; i < num_ref_entries[listIdx][rplsIdx]; i++) {	
if(inter_layer_ref_pics_present_flag)	
inter_layer_ref_pic_flag [listIdx][rplsIdx][i]	u(1)
if(!inter_layer_ref_pics_flag[listIdx][rplsIdx][i]) {	
if(long_term_ref_pics_flag)	
st_ref_pic_flag [listIdx][rplsIdx][i]	u(1)
if(st_ref_pic_flag[listIdx][rplsIdx][i]) {	
abs_delta_poc_st [listIdx][rplsIdx][i]	ue(v)
if(AbsDeltaPocSt[listIdx][rplsIdx][i] > 0)	
strp_entry_sign_flag [listIdx][rplsIdx][i]	u(1)

<code>} else if(!ltp_in_slice_header_flag[listIdx][rplsIdx])</code>	
<code> rpls_poc_lsb_lt[listIdx][rplsIdx][j++]</code>	<code>u(v)</code>
<code>} else</code>	
<code> ilrp_idc[listIdx][rplsIdx][i]</code>	<code>ue(v)</code>
<code>}</code>	
<code>}</code>	

Cấu trúc cú pháp `ref_pic_list_struct ( listIdx , rplsIdx )` có thể nằm trong SPS hoặc trong thông tin tiêu đề lát. Tùy thuộc vào việc cấu trúc cú pháp được bao gồm trong thông tin tiêu đề lát hay SPS, những điều sau sẽ áp dụng:

- Nếu có trong thông tin tiêu đề lát, cấu trúc cú pháp `ref_pic_list_struct ( listIdx , rplsIdx )` chỉ rõ danh sách ảnh tham chiếu `listIdx` của ảnh hiện tại (ảnh mà chứa lát).
- Nếu không phải (nằm trong SPS), cấu trúc cú pháp `ref_pic_list_struct ( listIdx , rplsIdx )` chỉ rõ ứng viên cho danh sách ảnh tham chiếu `listIdx` và thuật ngữ "ảnh hiện tại" trong ngữ nghĩa được chỉ rpx trong phần còn lại của mệnh đề này liên quan đến mỗi ảnh mà: (1) có một hoặc nhiều lát mà chứa `ref_pic_list_idx [ listIdx ]` bằng chỉ số vào danh sách của cấu trúc cú pháp `ref_pic_list_struct ( listIdx , rplsIdx )` được bao gồm trong SPS và (2) nằm trong CVS mà liên quan đến SPS.

`num_ref_entries[ listIdx ][ rplsIdx ]` chỉ rõ số lượng mục trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`. Giá trị của `num_ref_entries[ listIdx ][ rplsIdx ]` sẽ nằm trong đoạn từ 0 đến `sps_max_dec_pic_buffering_minus1 + 14`.

`ltp_in_slice_header_flag[ listIdx ][ rplsIdx ]` bằng 0 chỉ rõ rằng các POC LSB của các mục LTRP trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`

được hiện diện trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`. `ltrp_in_slice_header_flag[ listIdx ][ rplsIdx ]` bằng 1 chỉ rõ rằng các POC LSB của các mục LTRP trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` không được hiện diện trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`. `inter_layer_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` bằng 1 chỉ rõ rằng mục thứ `i` trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` là mục ILRP. `inter_layer_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` bằng 0 chỉ rõ rằng mục thứ `i` trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` không phải mục ILRP. Khi không được hiện diện, giá trị của `inter_layer_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` được xem là bằng 0.

`st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` bằng 1 chỉ rõ rằng mục thứ `i` trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` là mục STRP. `st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` bằng 0 chỉ rõ rằng mục thứ `i` trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` là mục LTRP. Khi `inter_layer_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` bằng 0 và `st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` không được hiện diện, giá trị của `st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` được xem là bằng 1.

Biến `NumLtrpEntries[ listIdx ][ rplsIdx ]` được thu nhận như sau:

```
for( i = 0, NumLtrpEntries[ listIdx ][ rplsIdx ] = 0; i <
    num_ref_entries[ listIdx ][ rplsIdx ]; i++ )
    if(!inter_layer_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]
        && !st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ] )
        NumLtrpEntries[ listIdx ][ rplsIdx ]++
```

`abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ]` chỉ rõ giá trị của biến `AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ]` như sau:

```
if( sps_weighted_pred_flag || sps_weighted_bipred_flag )
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] =
    abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ]
else
```

$$\text{AbsDeltaPocSt}[\text{listIdx}][\text{rplsIdx}][i] = \\ \text{abs_delta_poc_st}[\text{listIdx}][\text{rplsIdx}][i] + 1$$

Giá trị của $\text{abs_delta_poc_st}[\text{listIdx}][\text{rplsIdx}][i]$ sẽ nằm trong đoạn từ 0 đến $215 - 1$, inclusive.

$\text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 1 chỉ rõ rằng mục thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$ có giá trị lớn hơn hoặc bằng 0. $\text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]$ bằng 0 chỉ rõ rằng mục thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$ có giá trị nhỏ hơn 0. Khi không được hiện diện, giá trị của $\text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]$ được xem là bằng 1.

Danh sách $\text{DeltaPocValSt}[\text{listIdx}][\text{rplsIdx}]$ được thu nhận như sau:

$$\begin{aligned} & \text{for}(i = 0; i < \text{num_ref_entries}[\text{listIdx}][\text{rplsIdx}]; i++) \\ & \quad \text{if}(\quad \quad \quad !\text{inter_layer_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i] \quad \quad \quad \&\& \\ & \quad \quad \text{st_ref_pic_flag}[\text{listIdx}][\text{rplsIdx}][i]) \\ & \quad \quad \quad \text{DeltaPocValSt}[\text{listIdx}][\text{rplsIdx}][i] \quad \quad \quad = \quad \quad \quad (\\ & \quad \quad \quad \text{strp_entry_sign_flag}[\text{listIdx}][\text{rplsIdx}][i]) ? \\ & \quad \quad \quad \text{AbsDeltaPocSt}[\text{listIdx}][\text{rplsIdx}][i] \quad \quad \quad : 0 \quad \quad \quad - \\ & \quad \quad \quad \text{AbsDeltaPocSt}[\text{listIdx}][\text{rplsIdx}][i] \end{aligned}$$

$\text{rpls_poc_lsb_lt}[\text{listIdx}][\text{rplsIdx}][i]$ chỉ rõ giá trị của môđun số đếm thứ tự ảnh MaxPicOrderCntLsb của ảnh được viện dẫn bởi mục thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$. Độ dài của phần tử cú pháp $\text{rpls_poc_lsb_lt}[\text{listIdx}][\text{rplsIdx}][i]$ là $\log_2 \text{max_pic_order_cnt_lsb_minus4} + 4$ bit.

$\text{ilrp_idc}[\text{listIdx}][\text{rplsIdx}][i]$ chỉ rõ chỉ số, tới danh sách của các lớp phụ thuộc trực tiếp, của ILRP của mục thứ i trong cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$ tới danh sách của các lớp phụ thuộc trực tiếp. Giá trị của $\text{ilrp_idc}[\text{listIdx}][\text{rplsIdx}][i]$ sẽ nằm trong đoạn từ 0 đến $\text{GeneralLayerIdx}[\text{nuh_layer_id}] - 1$.

Sáng chế là phương pháp báo hiệu liên kết của các tham số dự đoán theo trọng số cú pháp mức cao (HLS-high-level syntax) và danh sách ảnh tham chiếu, trong đó danh sách ảnh tham chiếu có thể bao gồm các ảnh tham chiếu với cùng giá trị số đếm thứ tự ảnh (POC-picture order count). Các ảnh tham chiếu này tương ứng với cùng ảnh gốc được mã hóa, nhưng được mã hóa bằng cách sử dụng các tham số khác nhau, ví dụ khi các tham số khác nhau của dự đoán trọng số được sử dụng. Việc báo hiệu các cờ dự đoán theo trọng số có thể phụ thuộc vào việc danh sách tham chiếu có chứa các mục này hay không.

Theo phương án của sáng chế, danh sách ảnh tham chiếu được giới hạn ở các giá trị khác 0 khi các cờ dự đoán theo trọng số bằng 1. Tuy nhiên, trong mã hóa video hiện tại, các tham số dự đoán theo trọng số được báo hiệu sau khi báo hiệu danh sách ảnh tham chiếu. Trong bảng sau đây, có đề xuất sắp xếp lại thứ tự các phần tử cú pháp này và giới hạn mã hóa nhị phân của phần tử cú pháp đen-ta POC dựa trên các giá trị của các cờ dự đoán theo trọng số.

Bảng 4. Cú pháp RBSP tập tham số chuỗi (phương án thứ nhất)

seq_parameter_set_rbsp() {	Mô tả
sps_decoding_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)
sps_max_sub_layers_minus1	u(3)
sps_reserved_zero_5bits	u(5)
profile_tier_level(sps_max_sub_layers_minus1)	
gdr_enabled_flag	u(1)
sps_seq_parameter_set_id	ue(v)
chroma_format_idc	ue(v)

if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
pic_width_max_in_luma_samples	ue(v)
pic_height_max_in_luma_samples	ue(v)
subpics_present_flag	u(1)
if(subpics_present_flag) {	
max_subpics_minus1	u(8)
subpic_grid_col_width_minus1	u(v)
subpic_grid_row_height_minus1	u(v)
for(i = 0; i < NumSubPicGridRows; i++)	
for(j = 0; j < NumSubPicGridCols; j++)	
subpic_grid_idx[i][j]	u(v)
for(i = 0; i <= NumSubPics; i++) {	
subpic_treated_as_pic_flag[i]	u(1)
loop_filter_across_subpic_enabled_flag[i]	u(1)
}	
}	
bit_depth_luma_minus8	ue(v)
bit_depth_chroma_minus8	ue(v)
min_qp_prime_ts_minus4	ue(v)
sps_weighted_pred_flag	u(1)

sps_weighted_bipred_flag	u(1)
log2_max_pic_order_cnt_lsb_minus4	ue(v)
if(sps_max_sub_layers_minus1 > 0)	
sps_sub_layer_ordering_info_present_flag	u(1)
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 :sps_max_sub_layers_minus1); i <= sps_max_sub_layers_minus1; i++) {	
sps_max_dec_pic_buffering_minus1[i]	ue(v)
sps_max_num_reorder_pics[i]	ue(v)
sps_max_latency_increase_plus1[i]	ue(v)
}	
long_term_ref_pics_flag	u(1)
inter_layer_ref_pics_present_flag	u(1)
sps_idr_rpl_present_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)	
}	
...	

Và giá trị của đen-ta POC (biến AbsDeltaPocSt) được khôi phục có điều kiện tại phía bộ giải mã như sau:

abs_delta_poc_st[listIdx][rplsIdx][i] chỉ rõ giá trị của biến AbsDeltaPocSt[listIdx][rplsIdx][i] như sau:

```

if( sps_weighted_pred_flag || sps_weighted_bipred_flag )
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] =
    abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ]
else
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] =
    abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ] + 1

```

Lưu đồ trong Fig.7 minh họa phương pháp nêu trên. Trong bước 701, các tham số dự đoán theo trọng số (cụ thể là, sps_weighted_pred_flag và sps_weighted_bipred_flag) được báo hiệu. Tùy thuộc vào giá trị của chúng, báo hiệu của danh sách ảnh tham chiếu 702 được thực hiện khác nhau. Cụ thể, khi sps_weighted_pred_flag hoặc sps_weighted_bipred_flag là đúng (true), thì AbsDeltaPocSt được phép có giá trị bằng 0. Nếu không, AbsDeltaPocSt được khôi phục từ dòng bit bằng cách sử dụng giá trị tăng dần của abs_delta_poc_st, mà không cho phép các giá trị bằng 0 của AbsDeltaPocSt.

Trong phương án khác được bộc lộ tiếp, các cờ dự đoán theo trọng số sps_weighted_pred_flag và sps_weighted_bipred_flag được báo hiệu chỉ trong trường hợp khi ít nhất một danh sách ảnh tham chiếu ref_pic_list_struct có ít nhất một giá trị AbsDeltaPocSt bằng 0.

Bảng 5. Cú pháp RBSP tập tham số chuỗi (phương án thứ hai)

seq_parameter_set_rbsp() {	Mô tả
sps_decoding_parameter_set_id	u(4)
sps_video_parameter_set_id	u(4)

sps_max_sub_layers_minus1	u(3)
sps_reserved_zero_5bits	u(5)
profile_tier_level(sps_max_sub_layers_minus1)	
gdr_enabled_flag	u(1)
sps_seq_parameter_set_id	ue(v)
chroma_format_idc	ue(v)
if(chroma_format_idc == 3)	
separate_colour_plane_flag	u(1)
pic_width_max_in_luma_samples	ue(v)
pic_height_max_in_luma_samples	ue(v)
subpics_present_flag	u(1)
if(subpics_present_flag) {	
max_subpics_minus1	u(8)
subpic_grid_col_width_minus1	u(v)
subpic_grid_row_height_minus1	u(v)
for(i = 0; i < NumSubPicGridRows; i++)	
for(j = 0; j < NumSubPicGridCols; j++)	
subpic_grid_idx[i][j]	u(v)
for(i = 0; i <= NumSubPics; i++) {	
subpic_treated_as_pic_flag[i]	u(1)
loop_filter_across_subpic_enabled_flag[i]	u(1)
}	
}	
bit_depth_luma_minus8	ue(v)

bit_depth_chroma_minus8	ue(v)
min_qp_prime_ts_minus4	ue(v)
log2_max_pic_order_cnt_lsb_minus4	ue(v)
if(sps_max_sub_layers_minus1 > 0)	
sps_sub_layer_ordering_info_present_flag	u(1)
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 :sps_max_sub_layers_minus1); i <= sps_max_sub_layers_minus1; i++) {	
sps_max_dec_pic_buffering_minus1[i]	ue(v)
sps_max_num_reorder_pics[i]	ue(v)
sps_max_latency_increase_plus1[i]	ue(v)
}	
long_term_ref_pics_flag	u(1)
inter_layer_ref_pics_present_flag	u(1)
sps_idr_rpl_present_flag	u(1)
RestrictWPFlag = false	
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)	
}	
...	
if (!RestrictWPFlag) {	
sps_weighted_pred_flag	u(1)

sps_weighted_bipred_flag	u(1)
}	
...	

Bảng 6. Cú pháp ref_pic_list_struct (phương án thứ hai)

Ref_pic_list_struct(listIdx, rplsIdx) {	Mô tả
num_ref_entries[listIdx][rplsIdx]	ue(v)
if(long_term_ref_pics_flag)	
ltrp_in_slice_header_flag[listIdx][rplsIdx]	u(1)
for(i = 0, j = 0; i < num_ref_entries[listIdx][rplsIdx]; i++) {	
if(inter_layer_ref_pics_present_flag)	
inter_layer_ref_pic_flag[listIdx][rplsIdx][i]	u(1)
if(!inter_layer_ref_pics_flag[listIdx][rplsIdx][i]) {	
if(long_term_ref_pics_flag)	
st_ref_pic_flag[listIdx][rplsIdx][i]	u(1)
if(st_ref_pic_flag[listIdx][rplsIdx][i]) {	
abs_delta_poc_st[listIdx][rplsIdx][i]	ue(v)
if(AbsDeltaPocSt[listIdx][rplsIdx][i] > 0)	
strp_entry_sign_flag[listIdx][rplsIdx][i]	u(1)
else	
RestrictWPFlag = true	
} else if(!ltrp_in_slice_header_flag[listIdx][rplsIdx])	

rpls_poc_lsb_lt [listIdx][rplsIdx][j++]	u(v)
} else	
ilrp_idc [listIdx][rplsIdx][i]	ue(v)
}	
}	

Fig.8 minh họa phương pháp được bộc lộ trong phương án này. Theo thứ tự mã hóa được xác định trong Bảng 5, danh sách ảnh tham chiếu 801 được báo hiệu trước khi báo hiệu các tham số dự đoán theo trọng số 803. Các tham số dự đoán theo trọng số 803 chỉ được báo hiệu khi danh sách ảnh tham chiếu chứa ít nhất một phần tử có AbsDeltaPocSt bằng 0. Việc kiểm tra này được thực hiện trong bước 802 bởi biến RestrictWPFlag được khởi tạo thành sai (false) và được thiết lập thành đúng (true) khi giá trị 0 của AbsDeltaPocSt xuất hiện trong quá trình kiểm tra mỗi phần tử của danh sách ảnh tham chiếu.

Phần sau đây là giải thích của các ứng dụng của phương pháp mã hóa cũng như phương pháp giải mã như được thể hiện trong các phương án nêu trên, và hệ thống sử dụng các phương pháp này.

Hình 9 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 thu 3102, thiết bị đầu cuối 3106, và một cách tùy chọn bao gồm màn hình 3126. Thiết bị thu 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 mô tả nêu trên. Liên kết truyền thông 3104 bao gồm nhưng không giới hạn ở WIFI, Ethernet, Cáp, không dây (3G / 4G / 5G), USB hoặc bất kỳ loại kết hợp của chúng hoặc tương tự.

Thiết bị thu 3102 tạo ra dữ liệu và có thể mã hóa dữ liệu bằng phương pháp mã hóa như được thể hiện trong các phương án nêu trên. Ngoài ra, thiết bị thu 3102 có thể phân phối dữ liệu tới máy chủ tạo dòng (không được thể hiện trên

hình vẽ), và máy chủ mã hóa dữ liệu và truyền dữ liệu được mã hóa tới thiết bị đầu cuối 3106. Thiết bị thu 3102 bao gồm nhưng không giới hạn ở camera, điện thoại thông minh hoặc máy tính bảng, máy tính hoặc máy tính xách tay, hệ thống hội nghị truyền hình, PDA, thiết bị gắn trên xe, hoặc sự kết hợp của bất kỳ thiết bị trong số chúng hoặc loại tương tự. Ví dụ, thiết bị thu 3102 có thể bao gồm thiết bị nguồn 12 như được mô tả nêu trên. Khi dữ liệu bao gồm video, bộ mã hóa video 20 được chứa trong thiết bị thu 3102 có thể một cách thực tế thực hiện xử lý mã hóa video. Khi dữ liệu bao gồm audio (tức là, thoại), bộ mã hóa audio được chứa trong thiết bị thu 3102 có thể một cách thực tế thực hiện xử lý mã hóa audio. Đối với một số tình huống thực tế, thiết bị thu 3102 phân phối dữ liệu audio thành và video được mã hóa bằng cách ghép kênh chúng với nhau. Đối với các tình huống thực tế khác, chẳng hạn như trong hệ thống hội nghị truyền hình, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được ghép kênh. Thiết bị thu 3102 phân phối dữ liệu audio được mã hóa và dữ liệu video được mã hóa đến 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, như điện thoại thông minh hoặc máy tính bảng 3108, máy tính hoặc laptop 3110, bộ ghi video mạng (NVR)/ bộ ghi video số (DVR) 3112, TV 3114, hộp giải mã (STB) 3116, hệ thống hội nghị truyền hình 3118, hệ thống giám sát video 3120, thiết bị hỗ trợ cá nhân số (PDA) 3122, thiết bị lắp trên xe 3124, hoặc kết hợp bất kỳ của chúng, hoặc loại 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 mô tả nêu trên. Khi dữ liệu được mã hóa bao gồm video, bộ giải mã video 30 được chứa trong thiết bị đầu cuối được ưu tiên để thực hiện việc giải mã video. Khi dữ liệu được mã hóa bao gồm audio, bộ giải mã audio được chứa trong thiết bị đầu cuối được ưu tiên để thực hiện xử lý giải mã audio.

Đối với thiết bị đầu cuối có màn hình, ví dụ, điện thoại thông minh hoặc máy tính bảng 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 cá nhân số (PDA)

3122 hoặc thiết bị được gắn trên xe 3124, thiết bị đầu cuối có thể cung cấp dữ liệu được giải mã tới màn hình của thiết bị này. Đối với thiết bị đầu cuối không được trang bị màn hình, chẳng hạn như STB 3116, hệ thống hội nghị truyền hình 3118 hoặc hệ thống giám sát video 3120, màn hình ngoài 3126 được kết nối trong đó để thu và hiển thị dữ liệu được giải mã.

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

FIG.10 là sơ đồ 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 thu 3102, bộ 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 (RTSP), Giao thức truyền siêu văn bản (HTTP), Giao thức tạo dòng trực tiếp HTTP (HLS), MPEG-DASH, Giao thức truyền tải thời gian thực (RTP), Giao thức nhắn tin thời gian thực (RTMP)), hoặc bất kỳ loại kết hợp của chúng, hoặc loại tương tự.

Sau khi bộ tiến hành giao thức 3202 xử lý dòng, tệp dòng được tạo. Tệp được xuất ra bộ giải ghép kênh 3204. Bộ giải ghép kênh 3204 có thể tách biệt dữ liệu được ghép kênh thành dữ liệu audio được mã hóa và dữ liệu video được mã hóa. Như được mô tả nêu trên, đối với các tình huống thực tế khác, chẳng hạn như trong hệ thống hội nghị truyền hình, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được ghép kênh. Trong trường hợp này, dữ liệu được mã hóa được truyền tới bộ giải mã video 3206 và bộ giải mã audio 3208 mà không được truyền thông qua bộ giải ghép kênh 3204.

Thông qua xử lý giải ghép kênh, dòng sơ cấp (ES-elementary stream) video, audio ES và phụ đề tùy chọn được tạo ra. Bộ giải mã video 3206 mà bao gồm bộ giải mã video 30 như được giải thích trong các phương án nêu trên, giải mã dòng sơ cấp (ES) video bởi phương pháp giải mã như được thể hiện trong các phương án nêu trên để tạo ra khung video, và cấp dữ liệu này tới bộ đồng bộ 3212. Bộ giải mã audio 3208 giải mã dòng sơ cấp audio để tạo ra khung audio, và cấp dữ

liệu này tới bộ đồng bộ 3212. Ngoài ra, khung video có thể được lưu trữ trong bộ đệm (không được thể hiện trên FIG.10) trước khi cấp khung video tới bộ đồng bộ 3212. Tương tự, khung audio có thể được lưu trữ trong bộ đệm (không được thể hiện trên FIG.10) trước khi cấp khung audio tới bộ đồng bộ 3212.

Bộ đồng bộ 3212 đồng bộ khung video và khung audio, và cấp video/audio tới màn hình video/audio 3214. Ví dụ, bộ đồng bộ 3212 đồng bộ việc trình diễn thông tin video và thông tin audio. Thông tin có thể mã hóa trong cú pháp nhờ sử dụng các dấu thời gian liên quan đến việc trình diễn của audio được mã hóa và dữ liệu trực quan và các dấu thời gian liên quan đến việc phân phát dòng dữ liệu.

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

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

Fig.11 minh họa phương pháp mã hóa theo khía cạnh thứ nhất của sáng chế. Phương pháp mã hóa theo khía cạnh thứ nhất bao gồm các bước:

1101. xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS);

1102. mã hóa ít nhất một tham số dự đoán theo trọng số HLS; và

1103. mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS.

Fig.12 minh họa phương pháp mã hóa theo khía cạnh thứ hai của sáng chế. Phương pháp mã hóa theo khía cạnh thứ hai bao gồm các bước:

1201. xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó danh sách ảnh tham chiếu

thu được từ cấu trúc danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số số đếm thứ tự ảnh (POC); và 1202. mã hóa các phần tử cú pháp được xác định trong thứ tự mã hóa với giới hạn về nhị phân hóa của phần tử cú pháp mà có vị trí phía sau trong thứ tự mã hóa; trong đó khi ít nhất một tham số dự đoán theo trọng số HLS được mã hóa sau cấu trúc danh sách ảnh tham chiếu trong thứ tự mã hóa, giới hạn về nhị phân hóa của các phần tử cú pháp bao gồm: mã hóa ít nhất một tham số dự đoán theo trọng số HLS chỉ khi danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0.

Fig.13 minh họa phương pháp giải mã theo khía cạnh thứ ba của sáng chế. Phương pháp giải mã theo khía cạnh thứ ba bao gồm:

1301. Thu dòng bit;

1302. giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó trong các phần tử cú pháp, ít nhất một tham số dự đoán theo trọng số HLS được giải mã entropy trước cấu trúc danh sách ảnh tham chiếu;

1303. thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán;

1304. khôi phục khối được khôi phục dựa trên khối dự đoán; và

1305. thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Fig.14 minh họa phương pháp giải mã bởi bộ giải mã theo khía cạnh thứ tư của sáng chế. Phương pháp giải mã theo khía cạnh thứ tư bao gồm:

1401. Thu dòng bit;

1402. giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và cờ được thiết lập trước, trong đó giá trị của cờ được thiết lập trước chỉ báo rằng các phần tử cú pháp có bao gồm ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS) hay không;

1403. thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán;

1404. khôi phục khối được khôi phục dựa trên khối dự đoán; và

1405. thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Fig.15 minh họa bộ giải mã theo khía cạnh thứ tám của sáng chế. Bộ giải mã 1500, bao gồm một hoặc nhiều bộ xử lý 1501 và phương tiện lưu trữ bất biến đọc được bởi máy tính 1502 được ghép nối tới một hoặc nhiều bộ xử lý 1502 và lưu trữ chương trình để thực thi bởi các bộ xử lý 1501, trong đó chương trình, khi được thực thi bởi các bộ xử lý 1501, cấu hình bộ giải mã 1500 để thực hiện phương pháp theo khía cạnh thứ ba, cách thức thực hiện thứ nhất của khía cạnh thứ ba, khía cạnh thứ tư, hoặc bất kỳ một trong số cách thức thực hiện thứ nhất đến thứ tư của khía cạnh thứ tư.

Fig.16 minh họa bộ giải mã theo khía cạnh thứ chín của sáng chế. Bộ giải mã 1600 bao gồm phương tiện thu 1601 để thu dòng bit; phương tiện giải mã entropy 1602 để giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó trong các phần tử cú pháp, ít nhất một tham số dự đoán theo trọng số HLS được giải mã entropy trước cấu trúc danh sách ảnh tham chiếu; phương tiện dự đoán 1603 để thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán; phương tiện khôi phục 1604 để khôi phục khối được khôi phục dựa trên khối dự đoán; và phương tiện thu nhận 1605 để thu nhận ảnh được giải mã dựa trên khối được khôi phục.

Fig.17 minh họa bộ mã hóa theo khía cạnh thứ mười của sáng chế. Bộ mã hóa 1700 bao gồm: một hoặc nhiều bộ xử lý 1701; và phương tiện lưu trữ bất biến đọc được bởi máy tính 1702 được ghép nối với các bộ xử lý 1701 và lưu trữ chương trình để thực thi bởi các bộ xử lý 1701, trong đó chương trình, khi được thực thi bởi các bộ xử lý 1701, cấu hình bộ mã hóa 1700 để thực hiện phương pháp theo khía cạnh thứ nhất, bất kỳ một trong số cách thức thực hiện thứ nhất

đến thứ bảy của khía cạnh thứ nhất, hoặc khía cạnh thứ hai.

Fig.18 minh họa bộ mã hóa theo khía cạnh thứ mười một của sáng chế. Bộ mã hóa 1800 bao gồm: phương tiện xác định 1801 để xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS); và phương tiện mã hóa 1802 để mã hóa ít nhất một tham số dự đoán theo trọng số HLS và mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS.

Sáng chế đưa ra các phương án được lấy làm ví dụ khác sau đây:

1. Phương án ví dụ: Phương pháp báo hiệu liên kết các tham số dự đoán theo trọng số cú pháp mức cao (HLS) và danh sách ảnh tham chiếu, trong đó danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số số đếm thứ tự ảnh (POC), phương pháp này bao gồm:

xác định các phần tử cú pháp cần được báo hiệu, trong đó các phần tử cú pháp bao gồm danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số HLS; và

báo hiệu các phần tử cú pháp được xác định trong thứ tự mã hóa với giới hạn về nhị phân hóa của phần tử cú pháp mà có vị trí phía sau trong thứ tự mã hóa.

2. Phương án ví dụ: Phương pháp theo phương án ví dụ 1, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số.

3. Phương án ví dụ: Phương pháp theo phương án ví dụ 1 hoặc 2, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

4. Phương án ví dụ: Phương pháp theo phương án ví dụ 2, trong đó khi ít nhất một tham số dự đoán theo trọng số HLS được báo hiệu trước danh sách ảnh tham chiếu trong thứ tự mã hóa, giới hạn về nhị phân hóa của các phần tử cú pháp bao gồm :

khi cò tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập thành 0, báo hiệu giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

5. Phương án ví dụ: Phương pháp theo phương án ví dụ 3, trong đó khi ít nhất một tham số dự đoán theo trọng số HLS được báo hiệu trước danh sách ảnh tham chiếu trong thứ tự mã hóa, giới hạn về nhị phân hóa của các phần tử cú pháp bao gồm :

khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số, và cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số được thiết lập thành 0, báo hiệu giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); hoặc

khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cò tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và ít nhất một trong số cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cò tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập thành 0, báo hiệu giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt); hoặc

khi ít nhất một tham số dự đoán theo trọng số HLS bao gồm cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cò tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số, và cả cò tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số và cò tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập thành 0, báo hiệu giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-

ta POC được cải biến (abs_delta_poc_st) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (AbsDeltaPocSt).

6. Phương án ví dụ: Phương pháp theo phương án ví dụ 4 hoặc 5, trong đó giá trị đen-ta POC được cải biến nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa bởi 1 đơn vị.

7. Phương án ví dụ: Phương pháp theo bất kỳ một trong số các phương án ví dụ 1 đến 3, trong đó khi ít nhất một tham số dự đoán theo trọng số HLS được báo hiệu sau danh sách ảnh tham chiếu trong thứ tự mã hóa, giới hạn về nhị phân hóa của các phần tử cú pháp bao gồm:

báo hiệu ít nhất một tham số dự đoán theo trọng số HLS chỉ khi danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0.

8. Phương án ví dụ: phương pháp giải mã bởi bộ giải mã, bao gồm:

thu dòng bit;

giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số HLS, trong đó trong các phần tử này, ít nhất một tham số dự đoán theo trọng số HLS được hiện diện trước danh sách ảnh tham chiếu;

thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán;

khôi phục khối được khôi phục dựa trên khối dự đoán; và

thu nhận ảnh được giải mã dựa trên khối được khôi phục.

9. Phương án ví dụ: Phương pháp theo phương án ví dụ 8, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

10. Phương án ví dụ: phương pháp giải mã bởi bộ giải mã, bao gồm:

thu dòng bit;

giải mã entropy dòng bit để thu nhận các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm danh sách ảnh tham chiếu và cờ được thiết lập trước, trong đó giá trị của cờ được thiết lập trước chỉ báo rằng các phần tử cú pháp có bao gồm ít nhất một tham số dự đoán theo trọng số HLS hay không;

thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán;

khôi phục khối được khôi phục dựa trên khối dự đoán; và

thu nhận ảnh được giải mã dựa trên khối được khôi phục.

11. Phương án ví dụ: Phương pháp theo phương án ví dụ 10, trong đó giá trị của cờ được thiết lập trước tương ứng với việc danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0 hay không.

12. Phương án ví dụ: Phương pháp theo phương án ví dụ 11, trong đó khi giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu có ít nhất một phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp bao gồm ít nhất một tham số dự đoán theo trọng số HLS; hoặc

khi giá trị của cờ được thiết lập trước tương ứng với danh sách ảnh tham chiếu không có bất kỳ phần tử với giá trị đen-ta POC bằng 0, các phần tử cú pháp không bao gồm ít nhất một tham số dự đoán theo trọng số HLS.

13. Phương án ví dụ: Phương pháp theo bất kỳ một trong số các phương án ví dụ 10 đến 12, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

14. Phương án ví dụ: Phương pháp theo bất kỳ một trong số các phương án ví dụ 10 đến 13, trong đó cờ được thiết lập trước là RestrictWPFlag như được định nghĩa trong tiêu chuẩn kỹ thuật.

15. Phương án ví dụ: Bộ mã hóa (20) bao gồm mạch xử lý để thực hiện phương pháp theo bất kỳ một trong số các phương án ví dụ 1 đến 7.

16. Phương án ví dụ: Bộ giải mã (30) bao gồm mạch xử lý để thực hiện

phương pháp theo bất kỳ một trong số các phương án ví dụ 8 đến 14.

17. Phương án ví dụ: Sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo bất kỳ một trong số các phương án ví dụ 1 đến 14.

18. Phương án ví dụ: Bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ giải mã để thực hiện phương pháp theo bất kỳ một trong số các phương án ví dụ 8 đến 14.

19. Phương án ví dụ: Bộ mã hóa, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ mã hóa để thực hiện phương pháp theo bất kỳ một trong số các phương án ví dụ 1 đến 7.

20. Phương án ví dụ: Phương tiện bất biến đọc được bởi máy tính mà mang mã chương trình mà, khi được thực thi bởi thiết bị máy tính, làm cho thiết bị máy tính thực hiện phương pháp theo bất kỳ một trong số các phương án ví dụ 1 đến 14.

Các toán tử toán học

Các toán tử toán học được sử dụng trong sáng chế là tương tự như các toán tử được sử dụng trong ngôn ngữ lập trình C. Tuy nhiên, các kết quả của phép chia nguyên và các phép dịch số học được định nghĩa chính xác hơn, và các phép toán khác được xác định, như sự mũ hóa và phép chia có giá trị thực. Các quy ước đánh số và đếm thường bắt đầu từ 0, ví dụ, "thứ nhất" 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 nghĩa như sau:

$+$ Phép cộng

$-$ Phép trừ (dưới dạng toán tử hai đối số) hoặc phủ định (dưới dạng toán tử tiền tố một ngôi)

$*$ Phép nhân, bao gồm phép nhân ma trận

x^y Lũy thừa. Chỉ rõ x nâng lên lũy thừa của y . Trong các ngữ cảnh khác, ký hiệu này được sử dụng để ghi chỉ số trên không nhằm mục đích để hiểu như là lũy thừa.

$/$ Phép chia số nguyên với việc cắt kết quả về phía 0. Ví dụ, $7 / 4$ và $-7 / -4$ được cắt thành 1 và $-7 / 4$ và $7 / -4$ được cắt về -1 .

$\div$ Được sử dụng để ký hiệu phép chia trong các phương trình toán học mà không có ý định cắt hoặc làm tròn.

$\frac{x}{y}$ Được sử dụng để ký hiệu phép chia trong các phương trình toán học mà không có ý định cắt hoặc làm tròn.

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

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

Các toán tử logic

Các toán tử logic sau được định nghĩa như sau:

$x \&\& y$ logic Boolean "và" của x và y

$x || y$ logic Boolean "hoặc" của x và y

! Logic Boolean "không phải (not)"

$x ? y : z$ Nếu x là đúng (TRUE) hoặc không bằng 0, đánh giá là giá trị của y ; nếu không phải, đánh giá là giá trị của z .

Các toán tử quan hệ

Các toán tử quan hệ sau đây được định nghĩa 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 phần tử cú pháp hoặc biến đã được gán giá trị "na" (không áp dụng), giá trị "na" được coi là giá trị khác biệt đối với phần tử cú pháp hoặc biến. Giá trị "na" được xem là không bằng bất kỳ giá trị khác.

Toán tử thao tác bit (bit-wise)

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

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

| Thao tác bit "hoặc (or)". Khi thao tác trên các đối số nguyên, thao tác trên biểu diễn phần bù của hai của giá trị nguyên. Khi thao tác trên một đối số nhị phân mà chứa ít bit hơn đối số khác, đối số ngắn hơn được mở rộng bằng cách thêm nhiều 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 biểu diễn phần bù của hai của giá trị nguyên.

Khi thao tác trên một đối số nhị phân mà chứa ít bit hơn đối số khác, đối số ngắn hơn được mở rộng bằng cách thêm nhiều bit quan trọng hơn bằng 0.

$x \gg y$ Dịch phải số học của biểu diễn nguyên phần bù của hai của x bởi y ký tự nhị phân. Hàm này chỉ được xác định đối với các giá trị nguyên không âm của y . Các bit được dịch chuyển thành các bit quan trọng nhất (MSB) do kết quả của việc dịch phải có giá trị bằng MSB của x trước phép dịch.

$x \ll y$ Dịch trái số học của biểu diễn nguyên phần bù của hai của x bởi y ký tự nhị phân. Hàm này chỉ được xác định đối với các giá trị nguyên không âm của y . Các bit được dịch chuyển thành các bit ít quan trọng nhất (LSB) do kết quả của việc dịch trái có giá trị bằng 0.

Các toán tử gán

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

= Toán tử gán

++ tăng, tức là, $x++$ tương đương với $x = x + 1$; khi được sử dụng trong chỉ số mảng, đánh giá là giá trị của biến trước phép toán tăng.

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

+= Tăng theo số lượng được chỉ định, tức là $x += 3$ tương đương với $x = x + 3$, và $x += (-3)$ tương đương với $x = x + (-3)$.

- = Giảm theo số lượng đã chỉ định, tức là $x -= 3$ tương đương với $x = x - 3$, và $x -= (-3)$ tương đương với $x = x - (-3)$.

Ký hiệu phạm vi

Ký hiệu sau được sử dụng để chỉ rõ phạm vi của các giá trị:

$x = y..z$ x nhận các giá trị nguyên bắt đầu từ đoạn 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 được định nghĩa:

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

$\text{Asin}(x)$ hàm sin nghịch đảo lượng giác, hoạt động trên đối số x nằm trong đoạn từ $-1,0$ đến $1,0$, với giá trị đầu ra trong đoạn từ $-\pi \div 2$ đến $\pi \div 2$, tính bằng đơn vị radian

$\text{Atan}(x)$ hàm tang nghịch đảo lượng giác, hoạt động trên đối số x , với giá trị đầu ra trong đoạn từ $-\pi \div 2$ đến $\pi \div 2$, tính bằng đơn vị 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 hoạt động trên đối số x tính bằng đơn vị radian.

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

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

$\text{Ln}(x)$ logarit tự nhiên của x (logarit cơ số e , trong đó e là hằng số logarit tự nhiên 2,718 281 828 ...).

$\text{Log } 2(x)$ logarit cơ số 2 của x .

$\text{Log}_{10}(x)$ logarit 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 hoạt động trên đối số x tính bằng đơn vị radian

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

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

$\text{Tan}(x)$ hàm tang lượng giác hoạt động trên đối số x tính bằng đơn vị radian

Thứ tự ưu tiên phép toán

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

- Các phép toán có mức độ ưu tiên cao hơn được đánh giá trước bất kỳ phép toán có mức độ ưu tiên thấp hơn.
- Các phép toán có cùng thứ tự ưu tiên được đánh giá tuần tự từ trái sang phải.

Bảng dưới đây chỉ rõ mức độ ưu tiên của các phép toán từ cao nhất đến thấp nhất; vị trí cao hơn trong bảng chỉ báo mức độ ưu tiên cao hơn.

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

Bảng : Mức độ ưu tiên phép toán từ cao nhất (ở đầu bảng) đến thấp nhất (ở cuối bảng)

các phép toán (với các toán hạng x, y và z)
"x++", "x--"
"!x", "-x" (dưới dạng toán tử tiền tố một ngôi)
x^y
"x * y", "x / y", "x ÷ y", " $\frac{x}{y}$ ", "x % y"
"x + y", "x - y" (dưới dạng toán tử 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"

Phần mô tả văn bản của các phép toán logic

Trong văn bản này, câu lệnh của các phép toán logic sẽ được mô tả về mặt toán học ở dạng sau đây:

if(điều kiện 0)

câu lệnh 0

```

else if( điều kiện 1 )
    câu lệnh 1
...
else /* ghi chú thông tin điều kiện còn lại */
    câu lệnh n

```

có thể được mô tả theo cách thức sau:

... như sau / ... những điều sau áp dụng:

- If điều kiện 0, câu lệnh 0
- Nếu không phải, if điều kiện 1, câu lệnh 1
- ...
- Nếu không phải (ghi chú thông tin về điều kiện còn lại), câu lệnh n.

Mỗi "If ... Nếu không phải, if ... Nếu không phải, ..." câu lệnh trong văn bản được đưa ra với "... như sau" hoặc "... Những điều sau đây được áp dụng" ngay sau "If ...". Điều kiện cuối cùng của "If ... Nếu không phải, if ... Nếu không phải, ..." luôn là "Nếu không phải, ...". Ghép xen "If ... Nếu không phải, if ... Nếu không phải, ..." các câu lệnh có thể được nhận dạng bằng cách ghép "... Như sau" hoặc "... Những điều sau đây được áp dụng" với kết thúc "Nếu không phải, ...".

Trong văn bản này, câu lệnh của các phép toán logic sẽ được mô tả về mặt toán học ở dạng sau đây:

```

if( điều kiện 0a && điều kiện 0b )
    câu lệnh 0
else if( điều kiện 1a || điều kiện 1b )
    câu lệnh 1
...
else
    câu lệnh n

```

có thể được mô tả theo cách thức sau:

... như sau / ... những điều sau được áp dụng:

- Nếu tất cả điều kiện sau đây là đúng, câu lệnh 0:
 - điều kiện 0a
 - điều kiện 0b
- Nếu không phải, nếu một hoặc nhiều điều kiện sau đây là đúng, câu lệnh 1:
 - điều kiện 1a
 - điều kiện 1b
- ...
- Nếu không phải, câu lệnh n

Trong văn bản này, câu lệnh của các phép toán logic sẽ được mô tả về mặt toán học ở dạng sau đây:

```
if( điều kiện 0 )
    câu lệnh 0
if( điều kiện 1 )
    câu lệnh 1
```

có thể được mô tả theo cách thức sau:

```
khi điều kiện 0, câu lệnh 0
khi điều kiện 1, câu lệnh 1.
```

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 có viện dẫn tới bộ mã hóa 20 và bộ giải mã 30, có thể được thực hiện trong phần cứng, phần mềm, vi chương trình, hoặc bất kỳ kết hợp của chúng. Nếu được thực hiện trong phần mềm, các chức năng có thể được lưu trữ trong phương tiện đọc được bởi máy tính hoặc được truyền qua phương tiện truyền thông như là một hoặc nhiều lệnh hoặc mã và được thực thi bởi bộ xử lý dựa trên phần cứng. Phương tiện đọc được bởi máy tính có thể bao gồm phương tiện lưu trữ đọc được bởi máy tính, mà tương ứng với phương tiện hữu hình như phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông bao gồm bất kỳ phương tiện mà hỗ trợ truyền chương trình máy tính từ nơi này tới nơi khác, ví

dụ, theo giao thức truyền thông. Theo cách này, phương tiện đọc được bởi máy tính, nói chung có thể tương ứng với (1) phương tiện lưu trữ hữu hình đọc được bởi máy tính mà là bất biến hoặc (2) phương tiện truyền thông như tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ liệu có thể là bất kỳ phương tiện khả dụng mà có thể được truy nhập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để gọi ra các lệnh, mã, và/hoặc các cấu trúc dữ liệu để thực hiện các kỹ thuật được mô tả trong sáng chế. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được bởi máy tính.

Bằng ví dụ không bị giới hạn, phương tiện lưu trữ đọc được bởi máy tính này có thể bao gồm RAM, ROM, EEPROM, CD-ROM hoặc bộ lưu trữ đĩa quang khác, bộ lưu trữ đĩa từ hoặc thiết bị lưu trữ từ khác, bộ nhớ chớp, hoặc bất kỳ phương tiện khác mà có thể được sử dụng để lưu trữ mã chương trình mong muốn dưới dạng của lệnh hoặc cấu trúc dữ liệu và có thể được truy nhập bởi máy tính. Ngoài ra, bất kỳ kết nối cũng được gọi một cách thích hợp là phương tiện đọc được bởi máy tính. Ví dụ, nếu các lệnh được truyền từ trang mạng, máy chủ hoặc nguồn từ xa khác bằng cách sử dụng cáp đồng trục, cáp sợi quang, cặp dây xoắn, đường dây thuê bao số (Digital Subscriber Line-DSL) hoặc các kỹ thuật vô tuyến như tia hồng ngoại, vô tuyến và viba, thì cáp đồng trục, cáp sợi quang, cặp dây xoắn, DSL hoặc kỹ thuật không dây như tia hồng ngoại, vô tuyến và viba nằm trong định nghĩa của phương tiện. Tuy nhiên, sẽ được hiểu rằng phương tiện lưu trữ đọc được bởi máy tính và phương tiện lưu trữ dữ liệu không bao gồm các kết nối, sóng mang, tín hiệu, hoặc phương tiện chuyển tiếp khác, mà thay vì đó liên quan đến phương tiện lưu trữ hữu hình bất biến. Đĩa như được sử dụng ở đây, bao gồm đĩa nén (CD), đĩa laze, đĩa quang, đĩa đa năng số (DVD), đĩa mềm và đĩa Blu-ray, trong đó đĩa thường tái tạo dữ liệu bằng từ tính, trong khi đĩa tái tạo dữ liệu quang học bằng tia laze. Các kết hợp của bất kỳ phương tiện nêu trên cũng sẽ được nằm trong phạm vi của phương tiện đọc được bởi máy tính.

Các lệnh có thể được thực thi bởi một hoặc nhiều bộ xử lý, chẳng hạn như một hoặc nhiều bộ xử lý tín số (DSP), bộ vi xử lý đa năng, mạch tích hợp ứng dụng riêng (ASIC), mảng logic có thể lập trình dạng trường (FPGA) hoặc mạch

logic tích hợp hoặc rời rạc tương đương khác . Do đó, thuật ngữ “bộ xử lý”, như được sử dụng ở đây có thể đề cập đến bất kỳ cấu trúc nêu trên hoặc bất kỳ cấu trúc khác phù hợp để thực hiện các kỹ thuật được mô tả ở đây. Ngoài ra, trong một vài khía cạnh, chức năng được mô tả ở đây có thể được đề xuất trong phần cứng dành riêng và/hoặc các môđun phần mềm được cấu hình để mã hóa và giải mã, hoặc được tích hợp trong bộ mã hóa-giải mã được kết hợp. Ngoài ra, các kỹ thuật có thể đều được thực hiện trong một hoặc nhiều mạch hoặc các phần tử logic.

Các kỹ thuật của sáng chế có thể được thực hiện trong nhiều loại thiết bị hoặc cơ cấu, bao gồm bộ cầm tay không dây, mạch tích hợp (integrated circuit, IC) hoặc tập hợp IC (ví dụ, bộ chip). Các thành phần, môđun, hoặc bộ phận khác nhau được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị có cấu trúc để thực hiện các kỹ thuật được bộc lộ, nhưng không cần thiết được thực hiện bởi các bộ phận cứng khác nhau. Ngoài ra, như được mô tả nêu trên, các bộ phận khác nhau có thể được kết hợp trong bộ phần cứng mã hóa-giải mã hoặc được cung cấp bởi một tập hợp các bộ phận cứng tương tác, bao gồm một hoặc nhiều bộ xử lý như được mô tả nêu trên, kết hợp với phần mềm và/hoặc vi chương trình phù hợp.

YÊU CẦU BẢO HỘ

1. Phương pháp mã hóa, phương pháp này bao gồm:

- xác định các phần tử cú pháp cần được mã hóa, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS - high-level syntax);
- mã hóa ít nhất một tham số dự đoán theo trọng số HLS vào cú pháp tập hợp tham số chuỗi (SPS - sequence parameter set); và
- mã hóa cấu trúc danh sách ảnh tham chiếu sau khi mã hóa ít nhất một tham số dự đoán theo trọng số HLS vào cú pháp tập hợp tham số chuỗi (SPS).

2. Phương pháp theo điểm 1, trong đó danh sách ảnh tham chiếu thu được từ cấu trúc danh sách ảnh tham chiếu bao gồm các ảnh tham chiếu mà có cùng tham số số đếm thứ tự ảnh (POC).

3. Phương pháp theo điểm 1, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

4. Phương pháp theo điểm bất kỳ trong số các điểm 1 đến 3, trong đó việc mã hóa cấu trúc danh sách ảnh tham chiếu bao gồm giới hạn về mã hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu.

5. Phương pháp theo điểm 3, trong đó giới hạn về mã hóa của ít nhất một phần của cấu trúc danh sách ảnh tham chiếu bao gồm:

khi cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số được thiết lập thành 0, mã hóa giá trị đen-ta POC được cải biến đối với phần tử của danh sách ảnh tham chiếu, trong đó giá trị đen-ta POC được cải biến (`abs_delta_poc_st`) nhỏ hơn giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa (`AbsDeltaPocSt`).

6. Phương pháp theo điểm 5, trong đó giá trị đen-ta POC được cải biến nhỏ hơn

giá trị đen-ta POC đang được sử dụng trong xử lý mã hóa bởi 1.

7. Phương pháp giải mã bởi bộ giải mã, bao gồm:

thu dòng bit bao gồm cú pháp tập hợp tham số chuỗi (SPS - sequence parameter set);

giải mã dòng bit để thu nhận các phần tử cú pháp của cú pháp SPS, trong đó các phần tử cú pháp bao gồm cấu trúc danh sách ảnh tham chiếu và ít nhất một tham số dự đoán theo trọng số cú pháp mức cao (HLS), trong đó trong các phần tử cú pháp, ít nhất một tham số dự đoán theo trọng số HLS được giải mã trước cấu trúc danh sách ảnh tham chiếu;

thực hiện việc dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán;

khôi phục khối được khôi phục dựa trên khối dự đoán; và

thu nhận ảnh được giải mã dựa trên khối được khôi phục.

8. Phương pháp theo điểm 7, trong đó ít nhất một tham số dự đoán theo trọng số HLS bao gồm ít nhất một trong số cờ tập hợp tham số chuỗi cho việc dự đoán một chiều theo trọng số và cờ tập hợp tham số chuỗi cho việc dự đoán hai chiều theo trọng số.

9. Phương pháp theo điểm 7, trong đó việc giải mã của dòng bit để thu nhận các phần tử cú pháp được thực hiện bằng cách giải mã entropy.

10. Phương pháp theo điểm bất kỳ trong số các điểm 7 đến 9, trong đó việc thực hiện dự đoán dựa trên các phần tử cú pháp thu được để thu nhận khối dự đoán bao gồm:

thu nhận giá trị của đen-ta số đếm thứ tự ảnh (POC) dựa trên ít nhất một tham số dự đoán theo trọng số HLS và phần tử cú pháp trong cấu trúc danh sách ảnh tham chiếu;

thực hiện việc dự đoán dựa trên giá trị của đen-ta POC.

11. Phương pháp theo điểm 10, trong đó việc thu nhận giá trị của đen-ta POC dựa

trên ít nhất một tham số dự đoán theo trọng số HLS bao gồm:

xác định, dựa trên giá trị của ít nhất một tham số dự đoán theo trọng số HLS, rằng giá trị của đen-ta POC có được cho phép để có giá trị bằng 0 hay không;

khi được xác định rằng giá trị của đen-ta POC không được cho phép để có giá trị bằng 0, khôi phục giá trị của đen-ta POC nhờ sử dụng giá trị tăng của phần tử cú pháp trong cấu trúc danh sách ảnh tham chiếu.

12. Phương pháp theo điểm 10, trong đó phần tử cú pháp trong cấu trúc danh sách ảnh tham chiếu là `abs_delta_poc_st`; trong đó giá trị của đen-ta POC thu được như sau:

```
if( sps_weighted_pred_flag || sps_weighted_bipred_flag )
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] =
    abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ]
else
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] =
    abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ] + 1
```

trong đó `AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ]` là giá trị tuyệt đối của đen-ta POC, `abs_delta_poc_st[ listIdx ][ rplsIdx ][ i ]` là phần tử cú pháp trong cấu trúc danh sách ảnh tham chiếu.

13. Bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ giải mã để thực hiện phương pháp theo điểm bất kỳ trong số các điểm 7 đến 9.

14. Bộ mã hóa, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ bất biến đọc được bởi máy tính được ghép nối tới các bộ xử

lý và lưu trữ chương trình để thực thi bởi các bộ xử lý, trong đó chương trình, khi được thực thi bởi các bộ xử lý, cấu hình bộ mã hóa để thực hiện phương pháp theo điểm bất kỳ trong số các điểm 1 đến 3.

15. Phương tiện bất biến đọc được bởi máy tính mà mang mã chương trình mà, khi được thực thi bởi thiết bị máy tính, làm cho thiết bị máy tính thực hiện phương pháp theo điểm bất kỳ trong số các điểm 1 đến 3.

16. Phương tiện bất biến đọc được bởi máy tính mà mang mã chương trình mà, khi được thực thi bởi thiết bị máy tính, làm cho thiết bị máy tính thực hiện phương pháp theo điểm bất kỳ trong số các điểm 7 đến 9.

17. Thiết bị giải mã, bao gồm:

bộ lưu trữ nhớ bất biến, được tạo cấu hình để lưu trữ dữ liệu video dưới dạng của dòng bit; và

bộ giải mã được tạo cấu hình để thực hiện bất kỳ các phương pháp theo điểm bất kỳ trong số các điểm từ 7 đến 9.

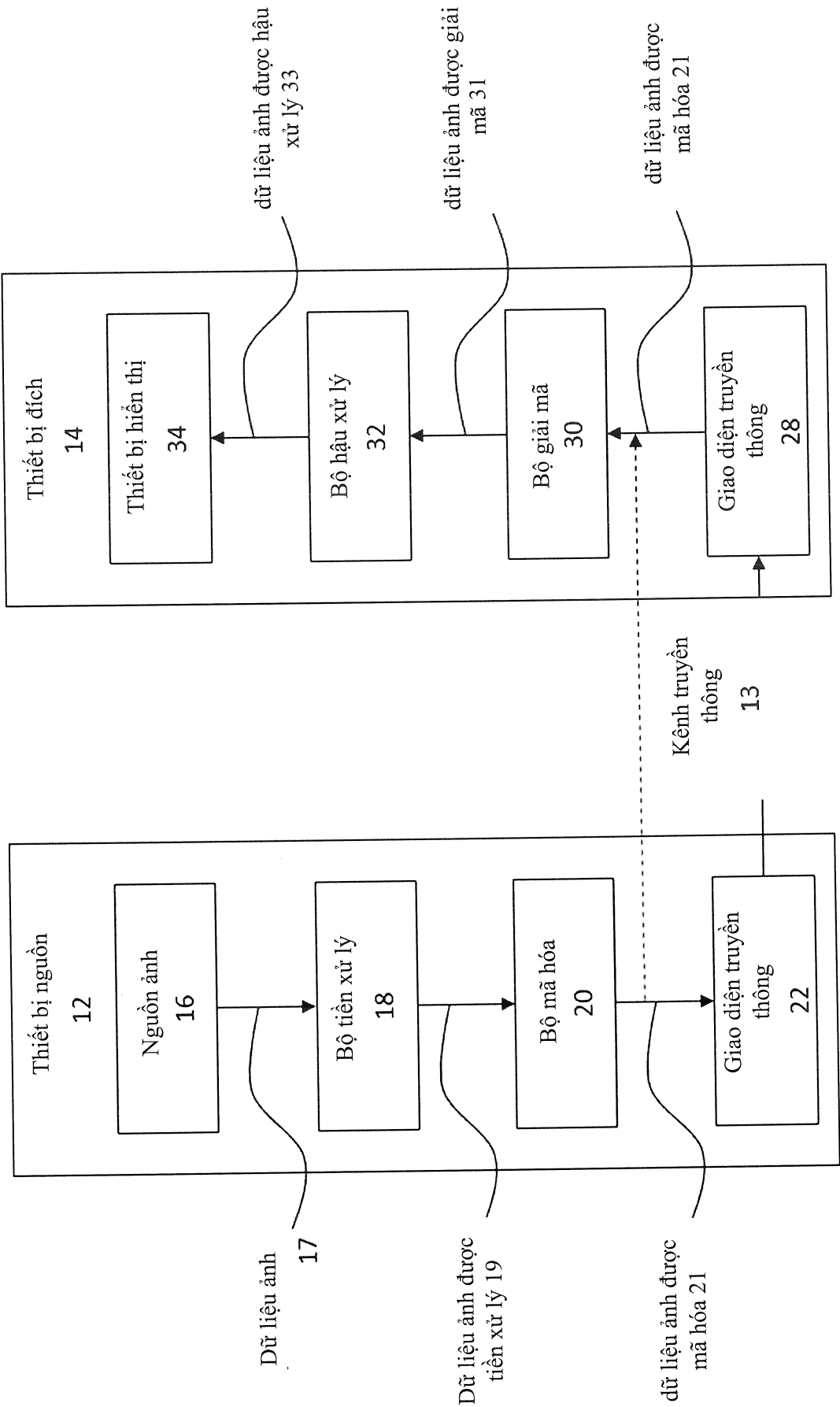


FIG. 1A

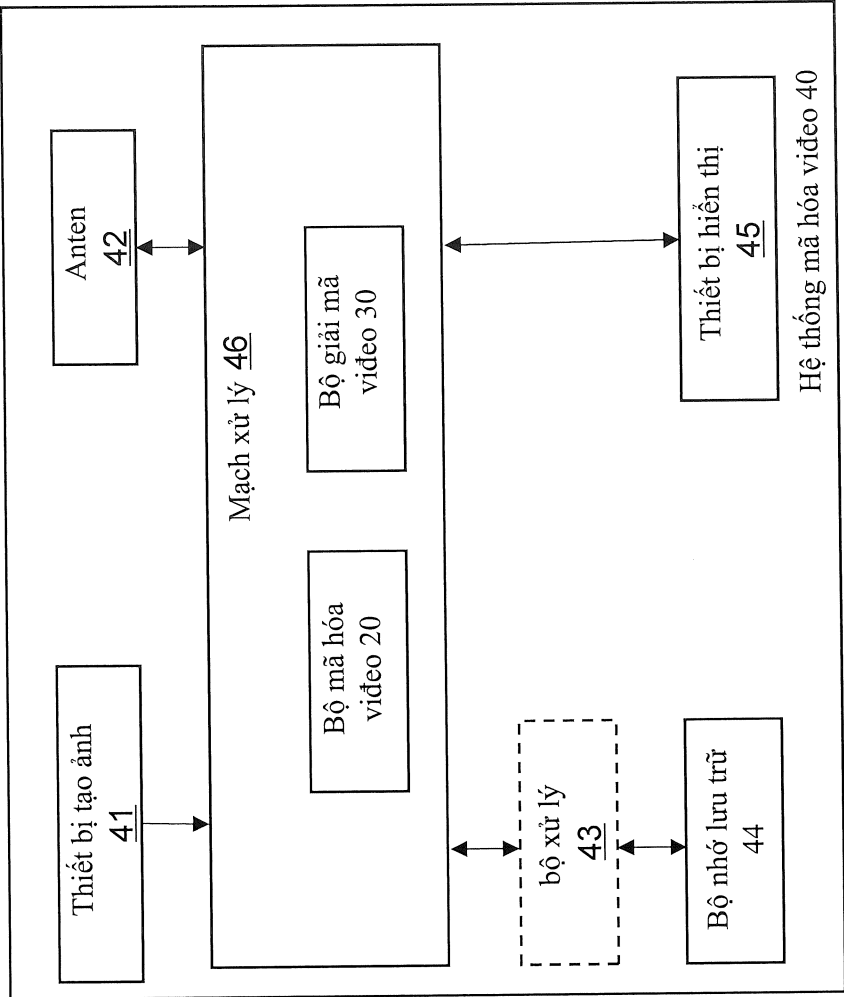


FIG. 1B

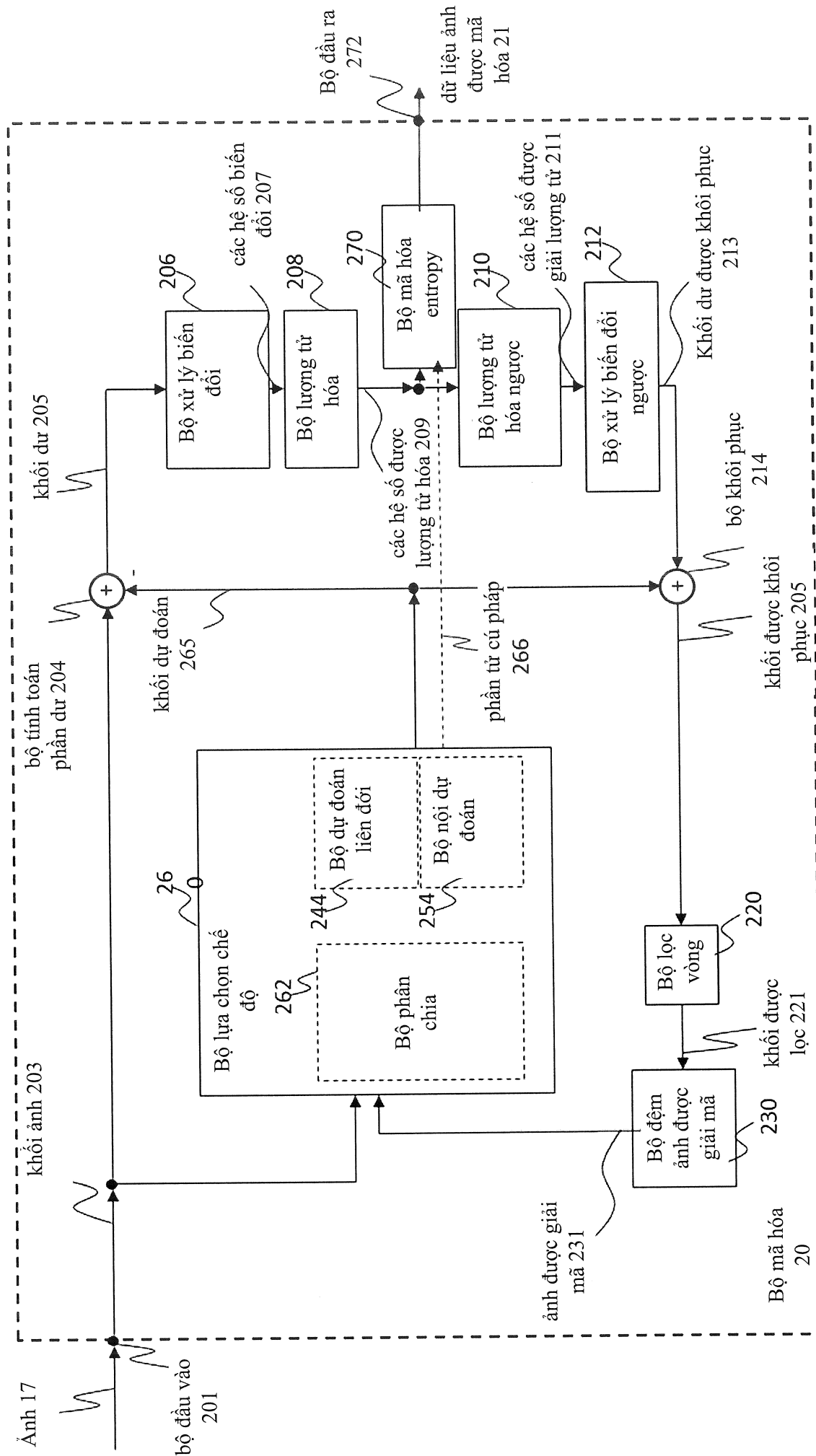
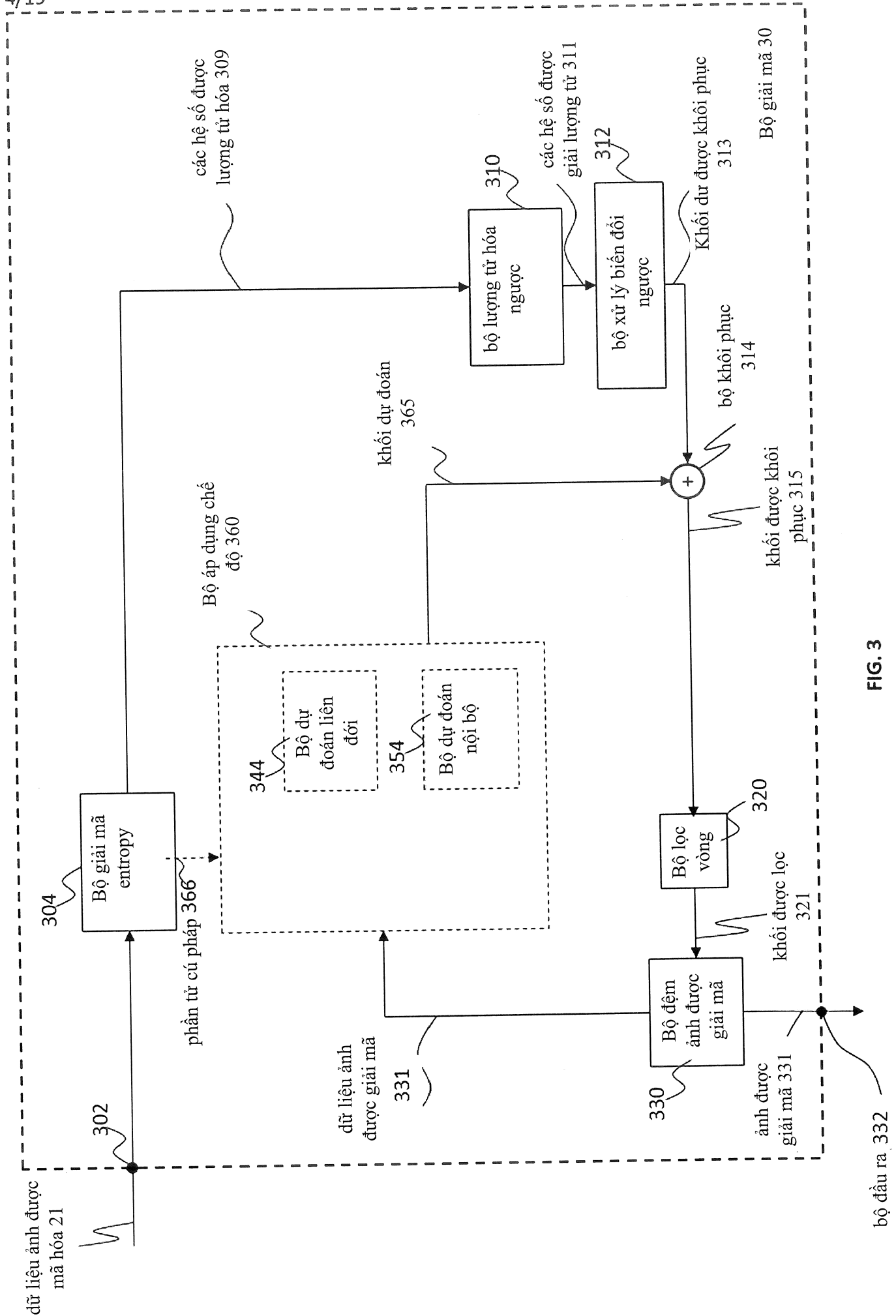


FIG. 2



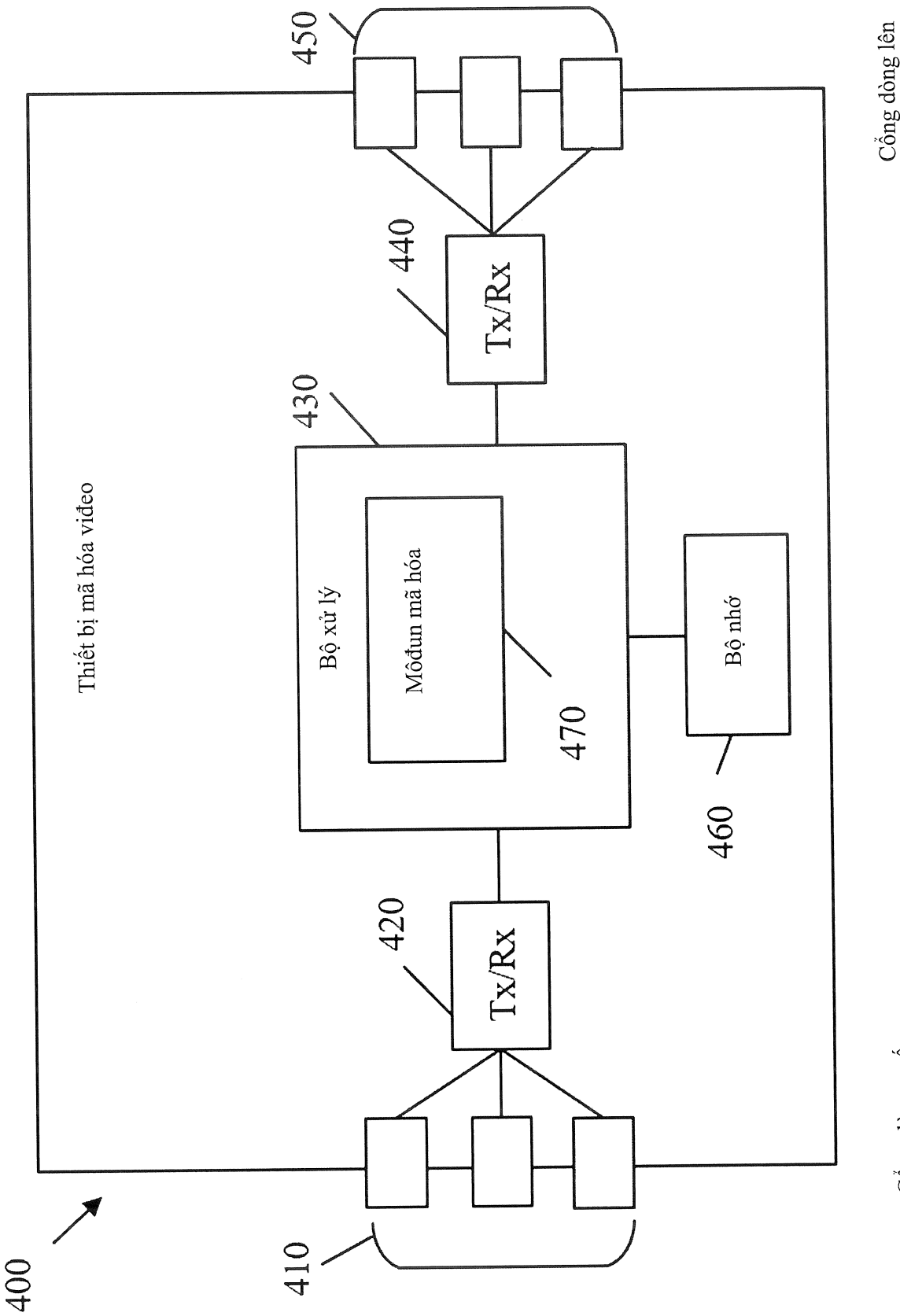


FIG. 4

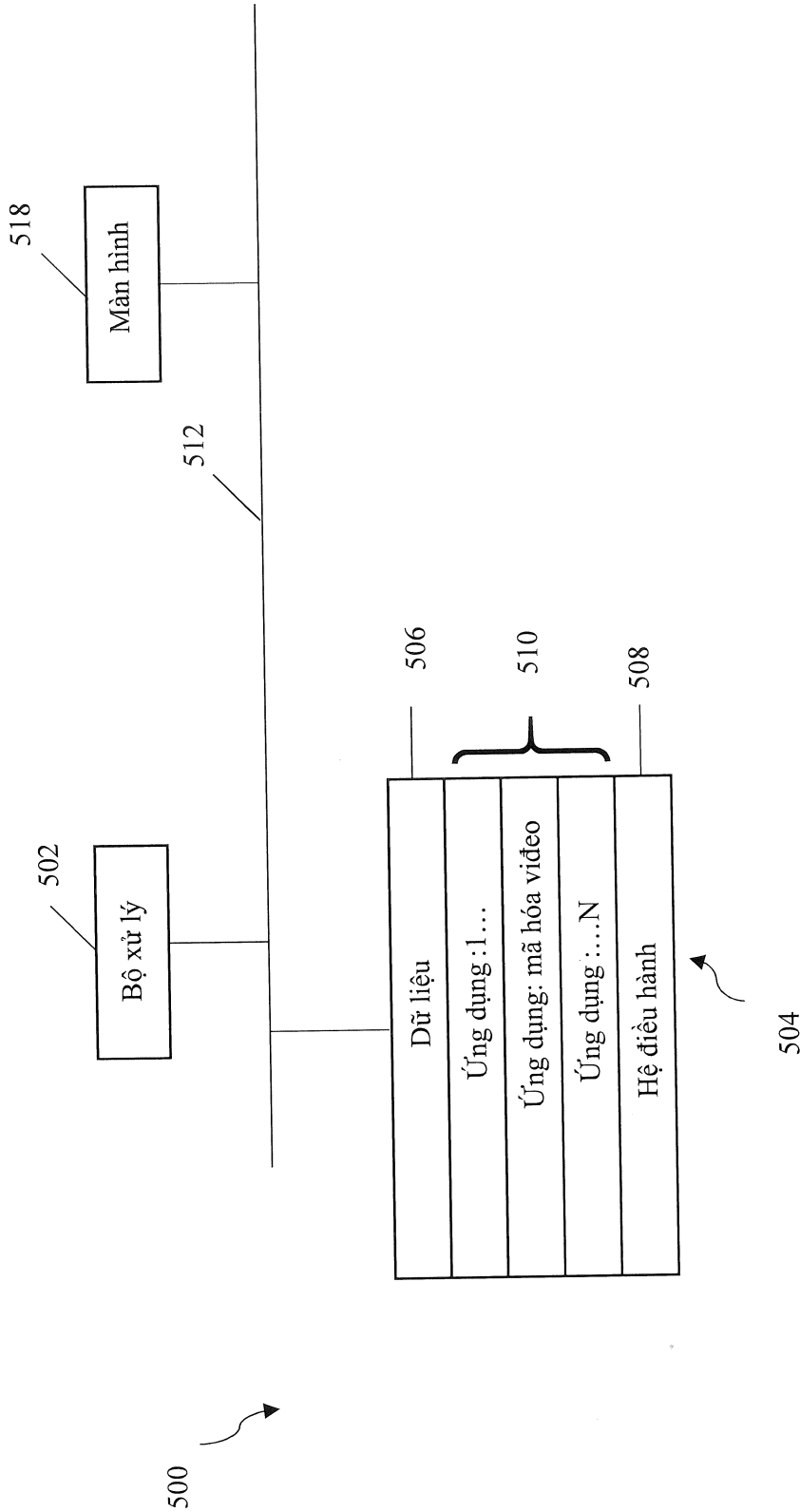


FIG. 5

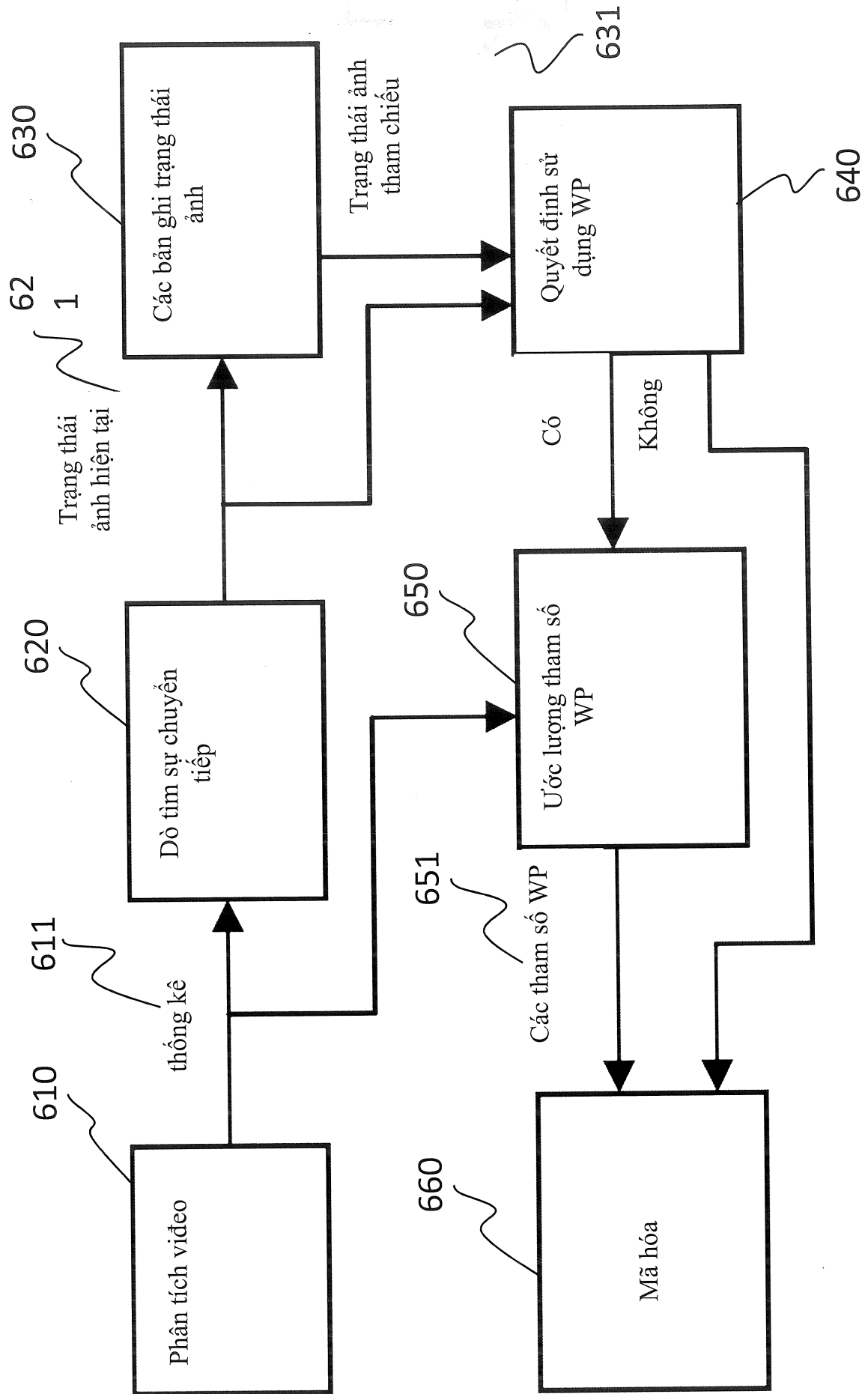


FIG. 6

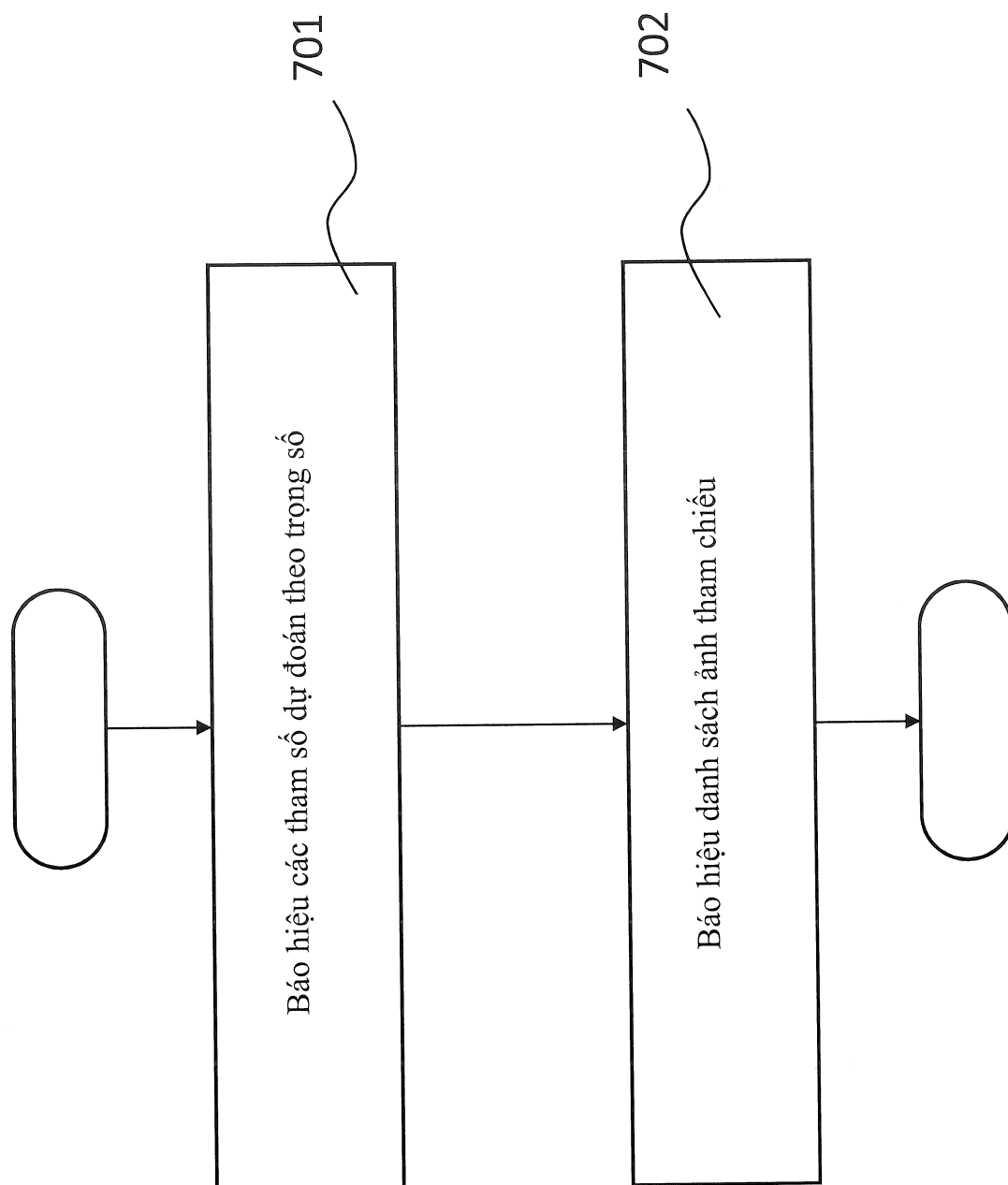


FIG. 7

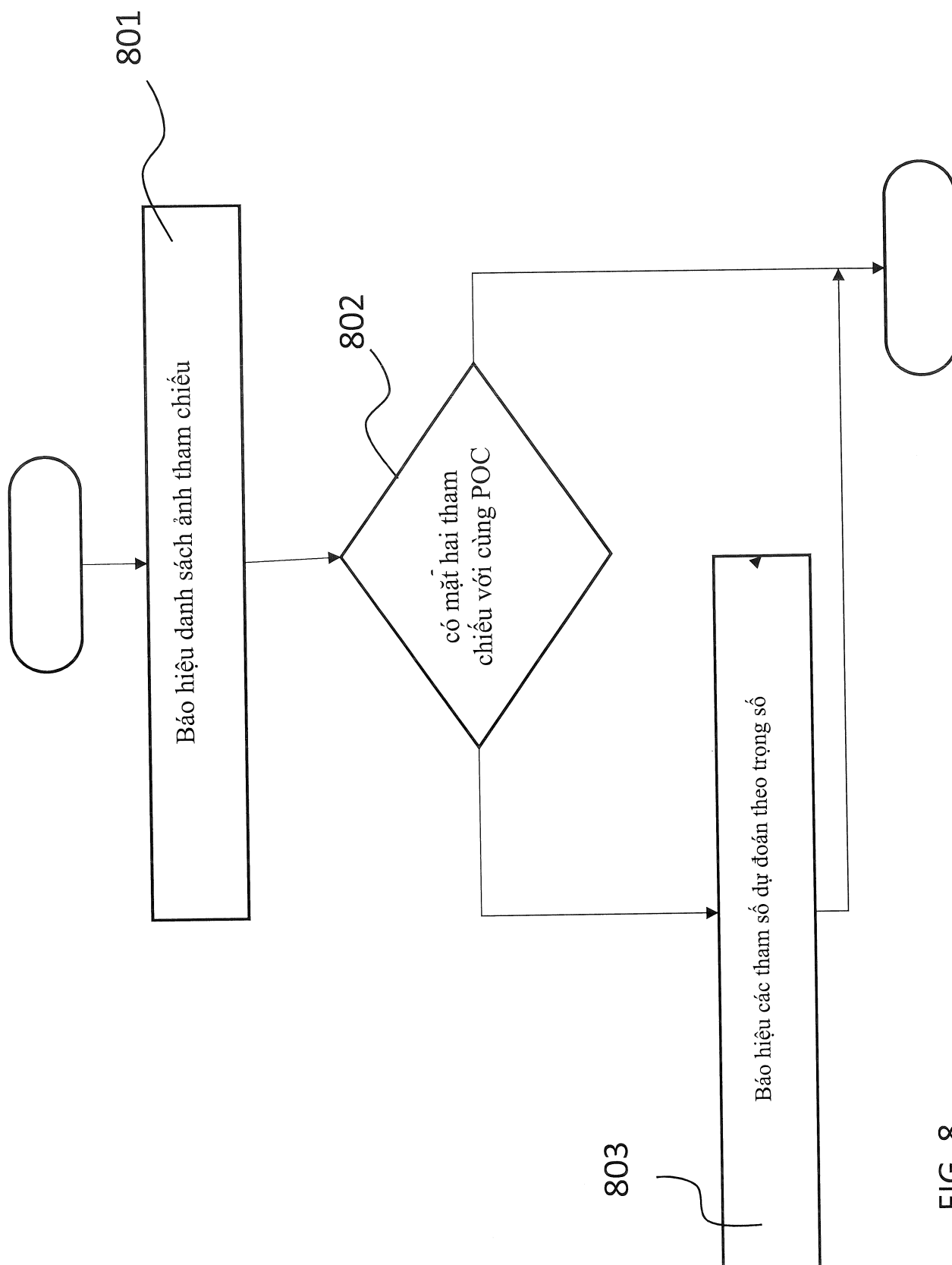


FIG. 8

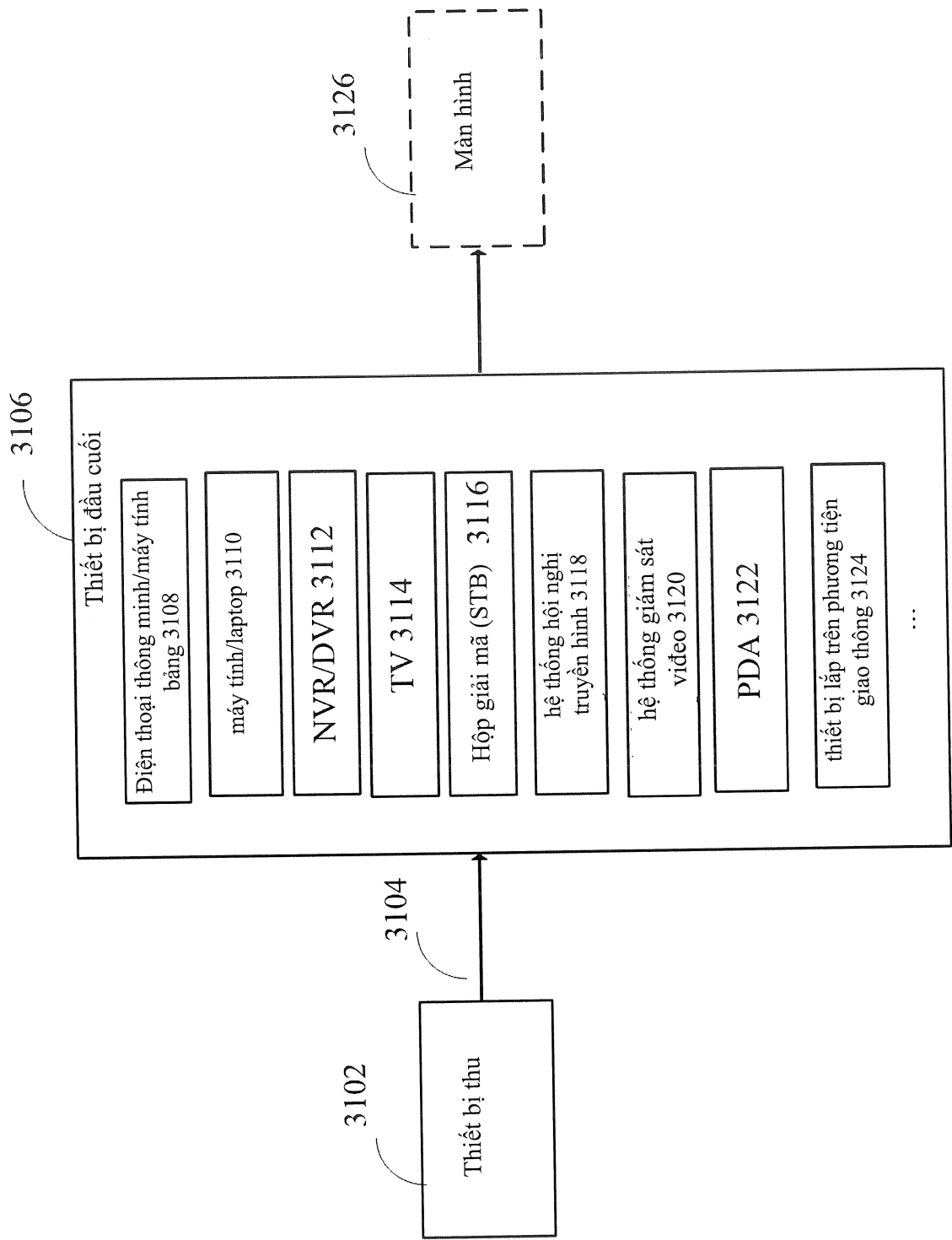


FIG. 9

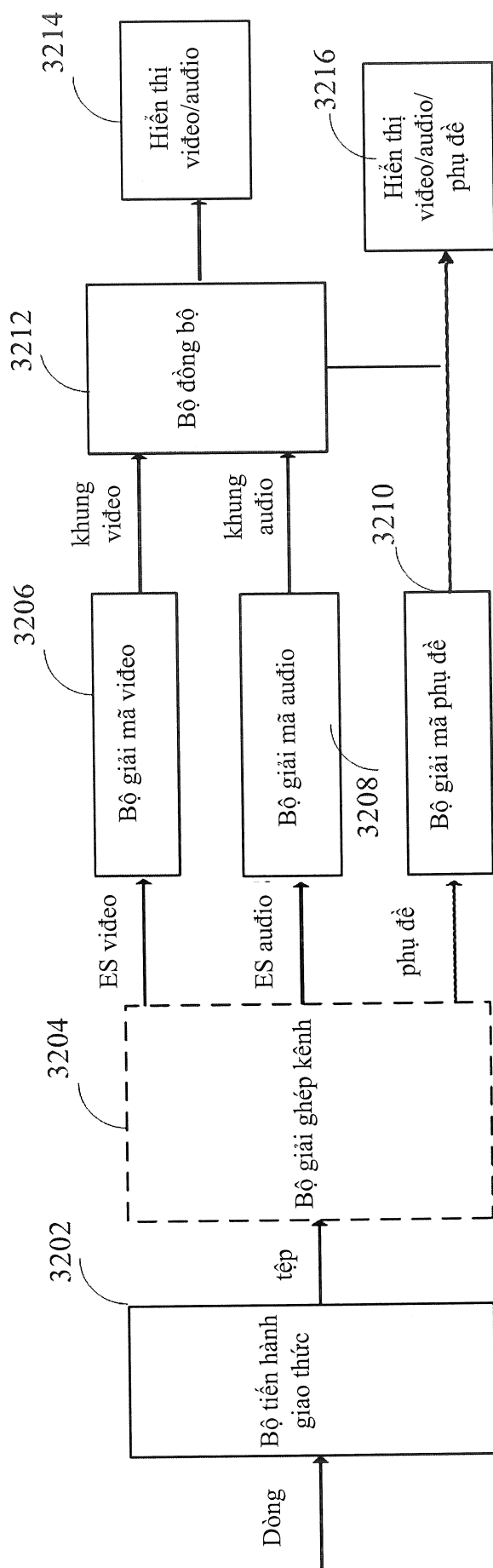


FIG. 10

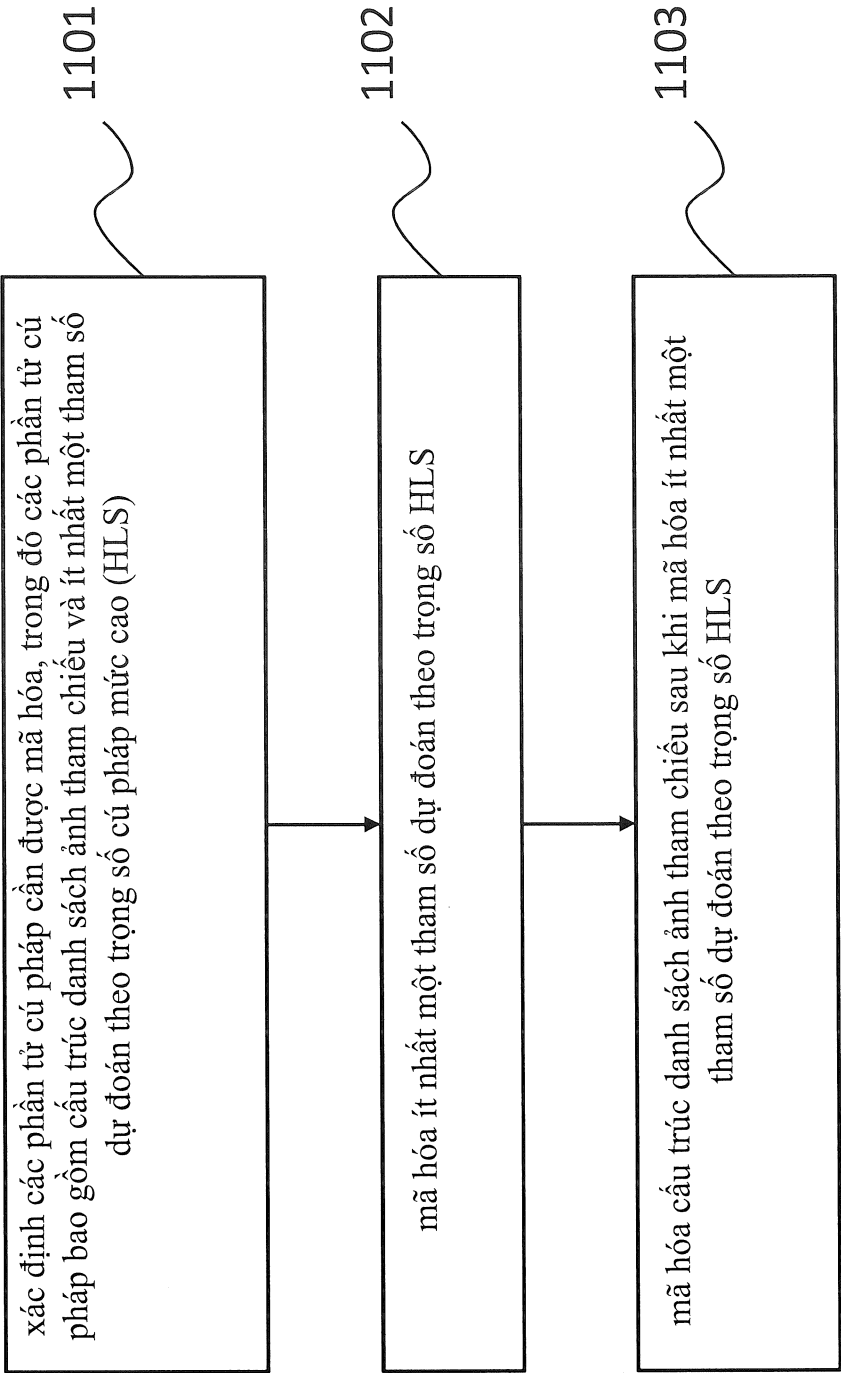


FIG. 11

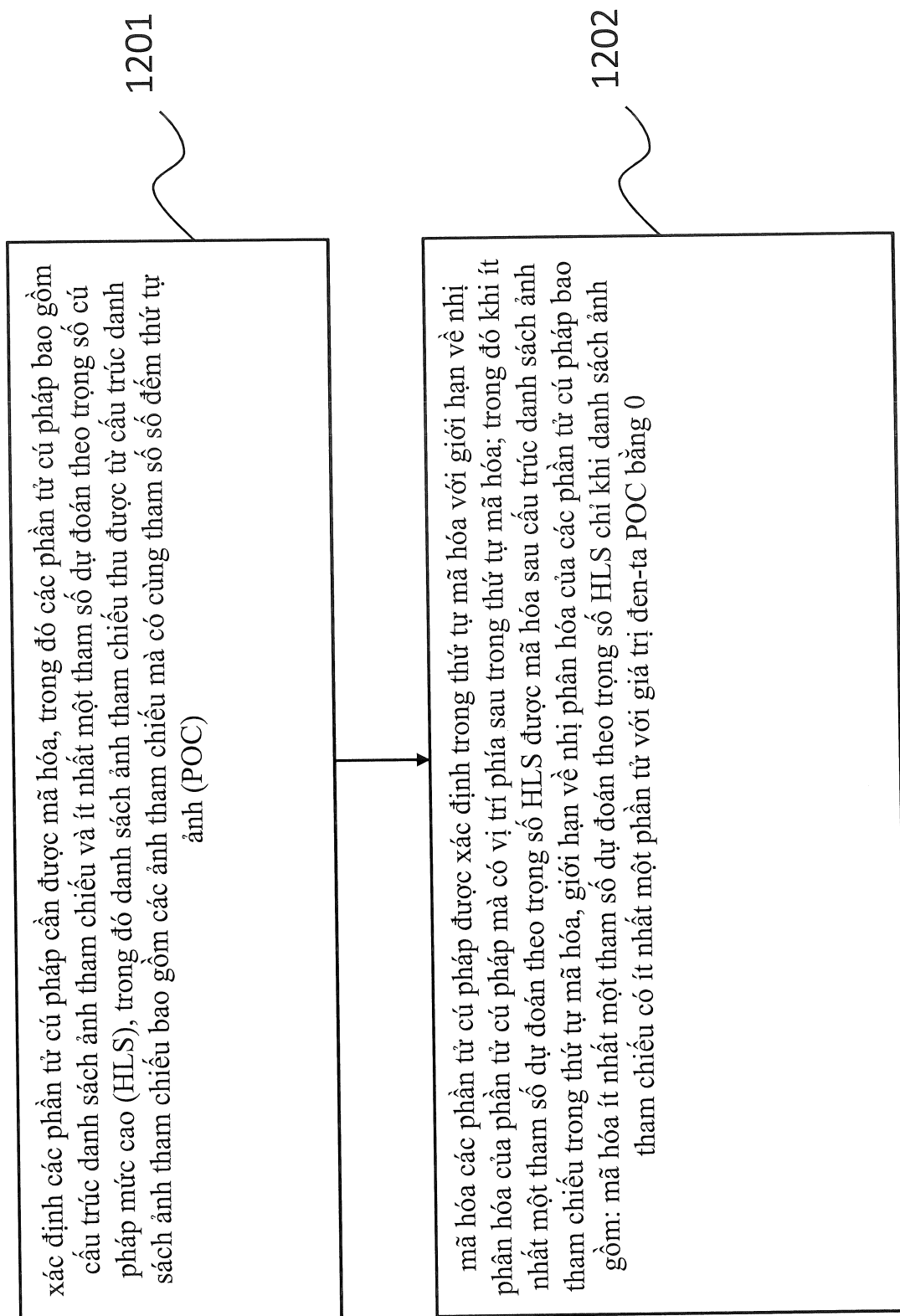


FIG. 12

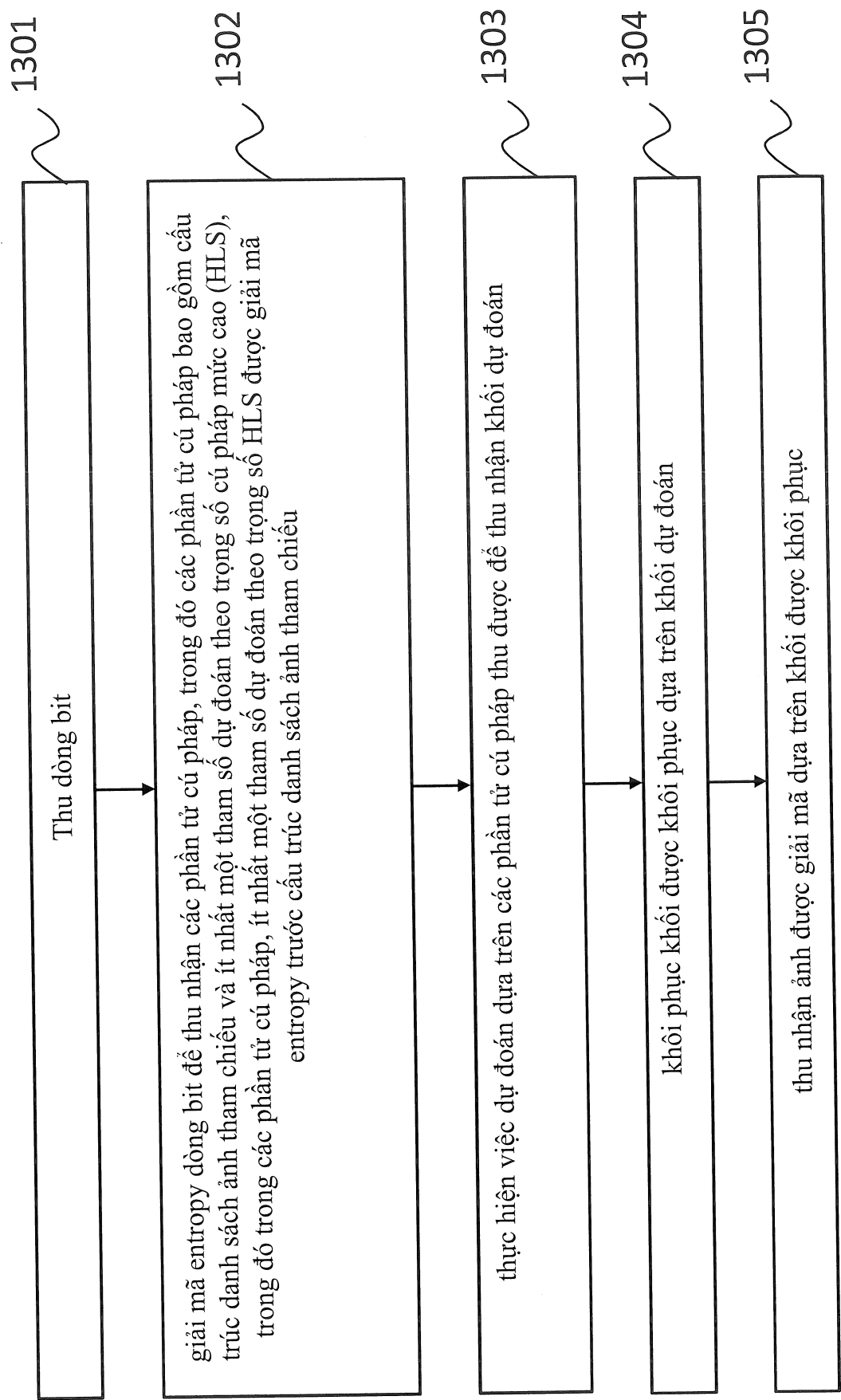


FIG. 13

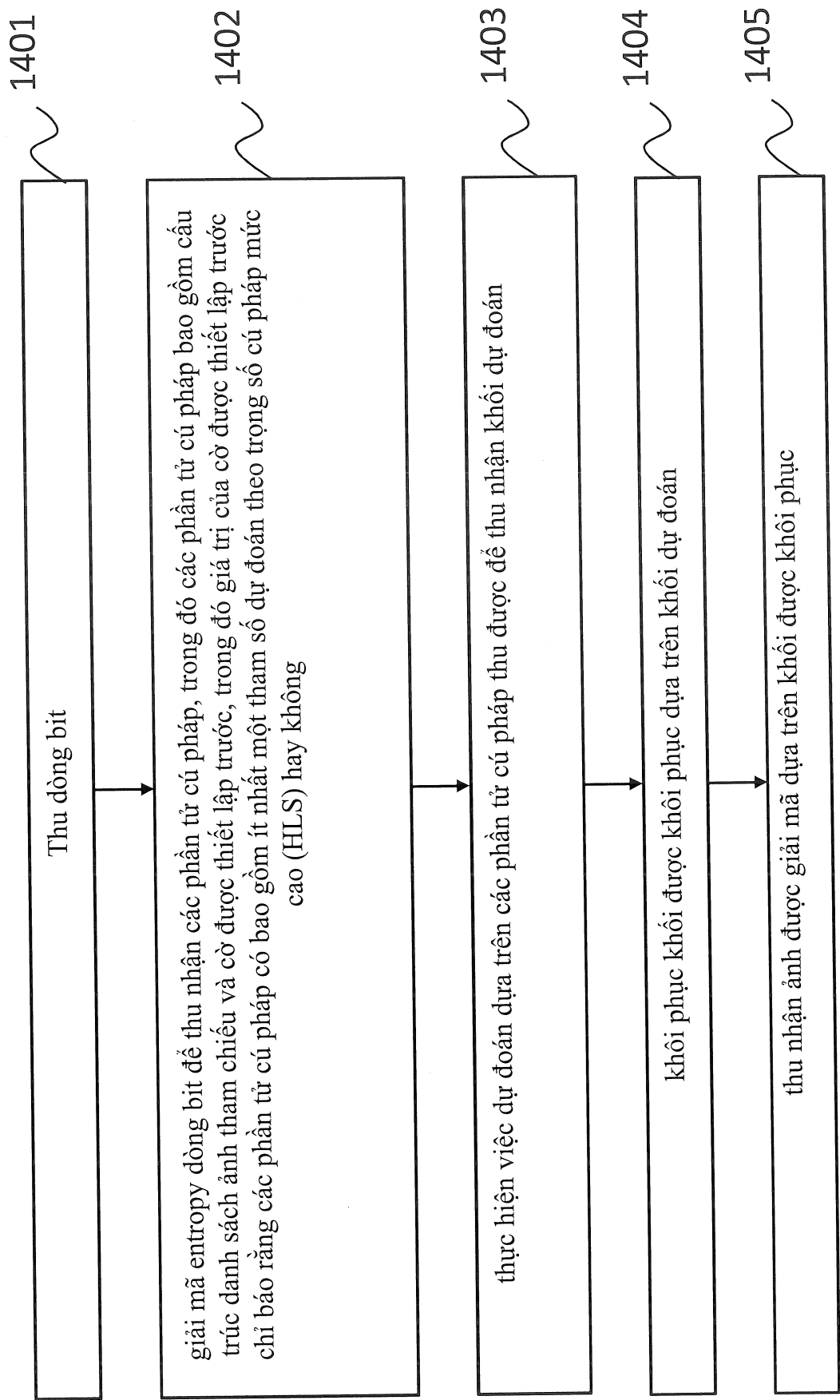


FIG. 14

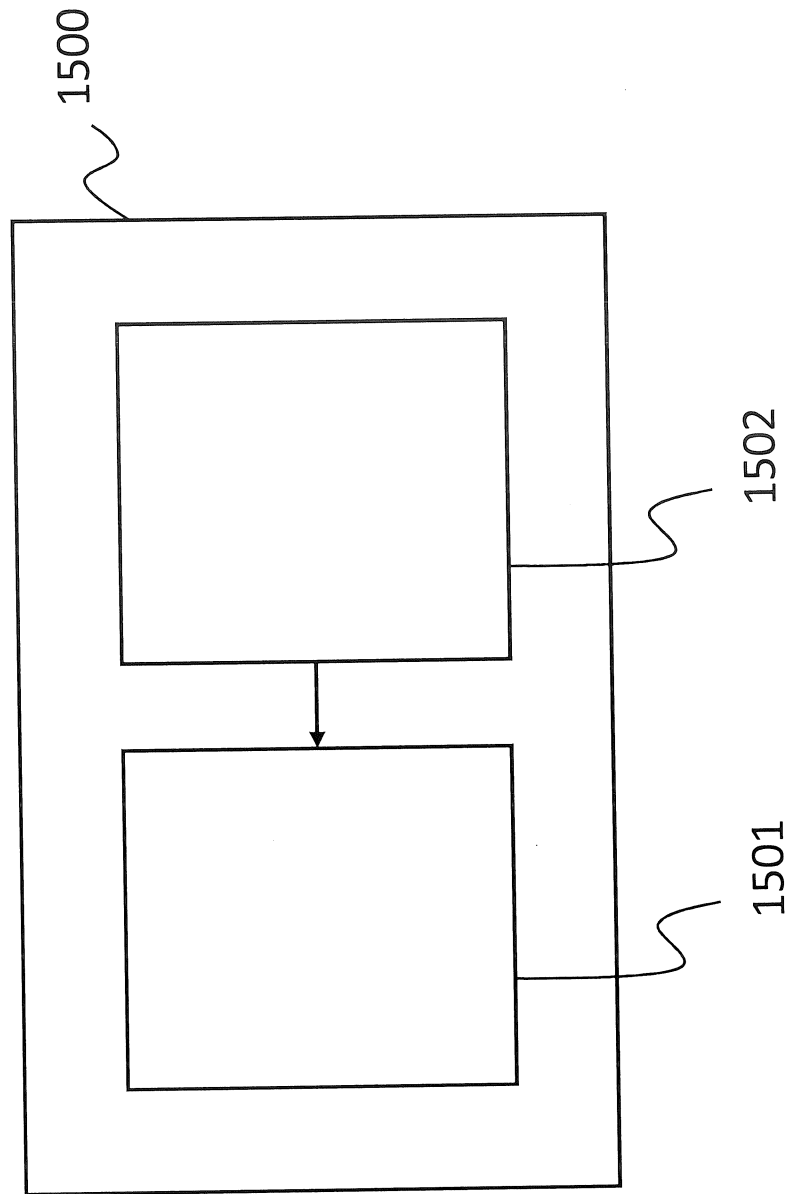


FIG. 15

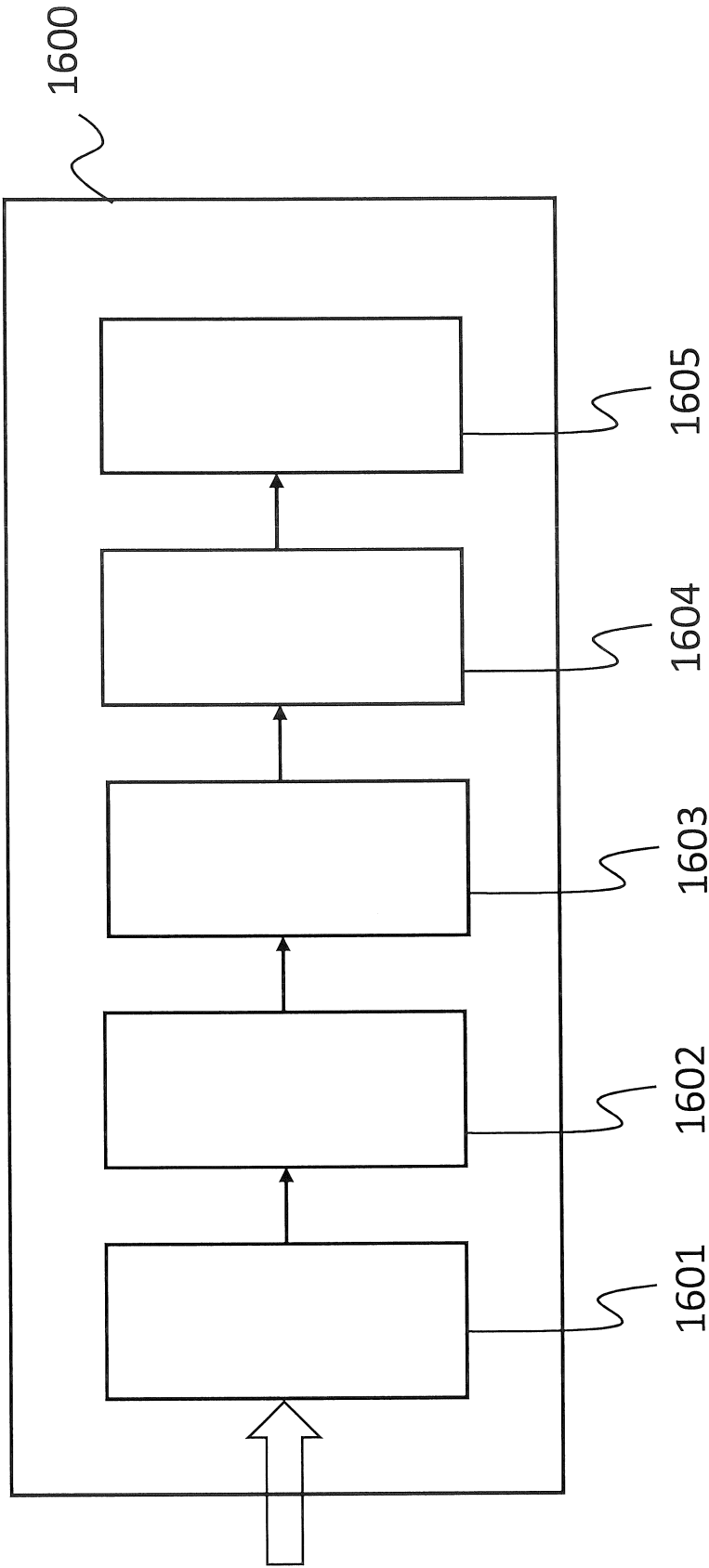


FIG. 16

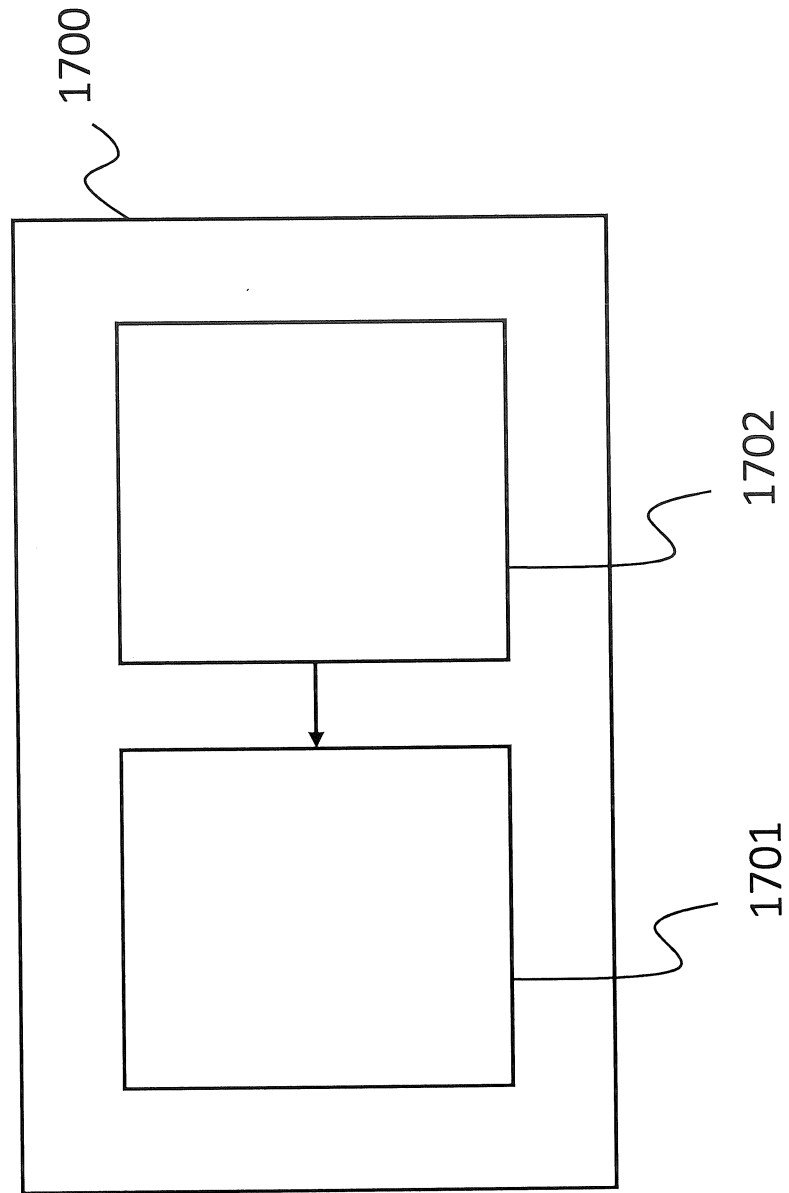


FIG. 17

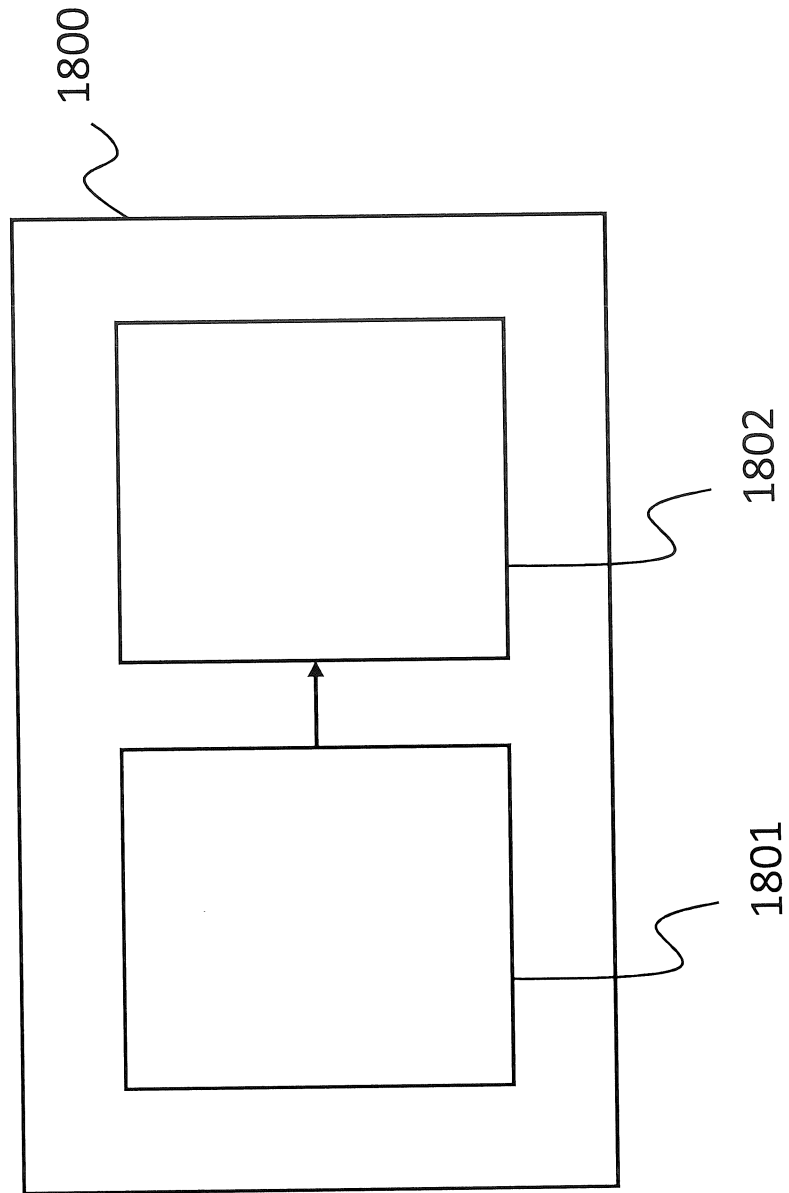


FIG. 18