



(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)



1-0052106

(51)^{2020.01} H04N 19/70; H04N 19/50

(13) B

(21) 1-2022-02757

(22) 30/09/2020

(86) PCT/CN2020/119696 30/09/2020

(87) WO2021/068854 15/04/2021

(30) 62/912,046 07/10/2019 US

(45) 27/10/2025 451

(43) 25/07/2022 412A

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

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

(72) MA, Xiang (CN); YANG, Haitao (CN).

(74) Công ty Luật TNHH T&G (TGVN)

(54) BỘ MÃ HOÁ, BỘ GIẢI MÃ, PHƯƠNG PHÁP GIẢI MÃ LUỒNG BIT VIDEO
ĐƯỢC LẬP MÃ, PHƯƠNG PHÁP MÃ HOÁ LUỒNG BIT VIDEO ĐƯỢC LẬP
MÃ, VÀ PHƯƠNG TIỆN PHI NHẤT THỜI ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(21)1-2022-02757

(57) Sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước: thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp; thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu. Phương pháp mã hoá luồng bit video được lập mã, bộ mã hoá, bộ giải mã, và phương tiện phi nhất thời đọc được bằng máy tính cũng được đề xuất.

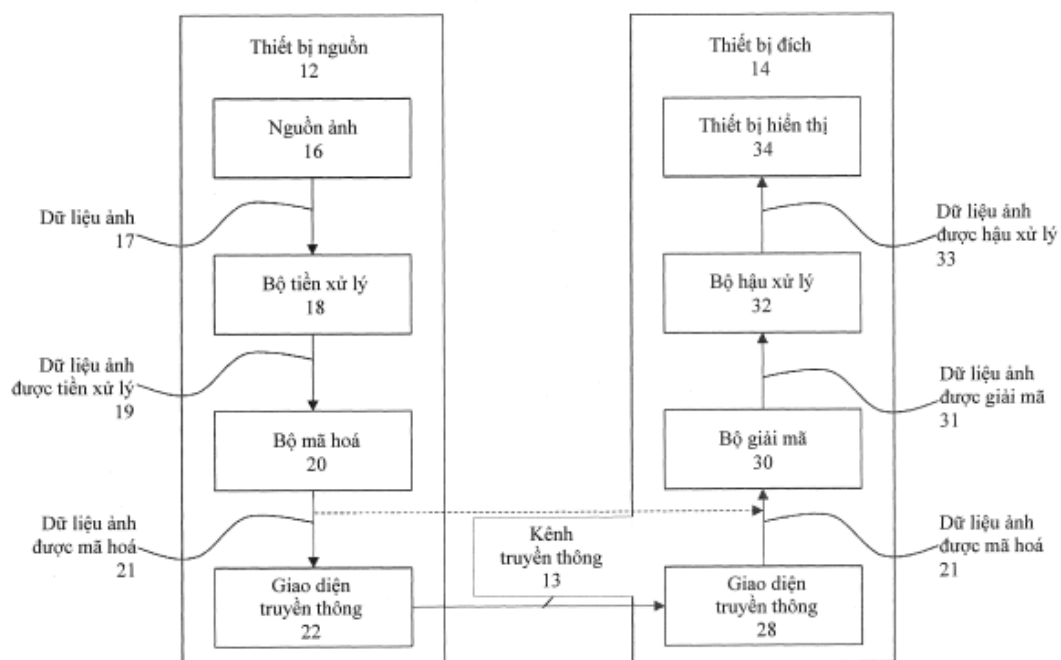


Fig.1A

Lĩnh vực kỹ thuật được đề cập

Các phương án của sáng chế (sự bộc lộ này) đề cập chung đến lĩnh vực xử lý ảnh, và cụ thể hơn là đến việc dự đoán liên lớp.

Tình trạng kỹ thuật của sáng chế

Việc lập mã video (mã hoá và giải mã video) được sử dụng rộng rãi trong nhiều loại ứng dụng video kỹ thuật số, ví dụ, truyền hình kỹ thuật số (broadcast digital TV), truyền video qua mạng Internet và các mạng di động, các ứng dụng đàm thoại thời gian thực, chẳng hạn chat video, hội nghị video, đĩa DVD (Digital Video Disc - đĩa video kỹ thuật số) và đĩa Blu-ray, các hệ thống thu thập và biên tập nội dung video, và các máy ghi hình cầm tay của các ứng dụng an ninh.

Lượng dữ liệu video cần thiết để thể hiện ngay cả một video tương đối ngắn cũng có thể là lớn, điều này có thể dẫn đến những khó khăn khi dữ liệu này cần được tạo luồng hoặc được truyền thông qua mạng truyền thông có dung lượng băng thông hạn chế. Do đó, dữ liệu video thường được nén trước khi được truyền thông qua các mạng viễn thông thời hiện đại. Kích thước của video cũng có thể là một vấn đề khi video được lưu giữ trên thiết bị lưu trữ vì các 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 lập mã dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó làm giảm lượng dữ liệu cần thiết để biểu thị các hình ảnh video số. Sau đó, dữ liệu được nén này được nhận tại đích đến bởi thiết bị giải nén video mà giải mã dữ liệu video đó. Với các tài nguyên mạng bị hạn chế và các nhu cầu ngày càng tăng về chất lượng video cao hơn, thì mong muốn là có các kỹ thuật nén và giải nén được cải thiện để cải thiện tỷ số nén mà chỉ phải hy sinh ít hoặc không phải hy sinh chất lượng ảnh.

Bản chất kỹ thuật của sáng chế

Theo các phương án, sáng chế đề xuất các thiết bị và các phương pháp để mã hoá và giải mã theo các điểm yêu cầu bảo hộ độc lập.

Mục đích nêu trên và các mục đích khác là đạt được nhờ đối tượng của các điểm yêu cầu bảo hộ độc lập. Các dạng thực hiện nữa được thể hiện rõ ở các điểm yêu cầu bảo hộ phụ thuộc, phần mô tả và các hình vẽ.

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

Theo khía cạnh thứ nhất, sáng chế liên quan đến phương pháp giải mã luồng bit video được lập mã, phương pháp này được thực hiện bởi thiết bị giải mã. Phương pháp này bao gồm các bước: thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp; thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu.

Theo cách khác, phần tử cú pháp thứ nhất quy định xem một hoặc nhiều phần tử cú pháp thứ hai, mà liên quan đến một hoặc nhiều lớp thứ hai, có mặt trong luồng bit video được lập mã hay không. Ngoài ra, trong đó việc phần tử cú pháp thứ nhất bằng 1 quy định rằng một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai là không có mặt trong luồng bit video được lập mã; hoặc việc phần tử cú pháp thứ nhất bằng 0 quy định rằng một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai là có mặt trong luồng bit video được lập mã.

Trong đó lớp bao gồm chuỗi ảnh được lập mã có cùng chỉ số lớp.

Trong đó các chỉ số lớp của một hoặc nhiều lớp thứ hai là nhỏ hơn chỉ số lớp của lớp thứ nhất.

Trong đó các lớp thứ hai liên quan đến phần tử cú pháp thứ hai khác là có chỉ số lớp khác.

Trong đó một hoặc nhiều phần tử cú pháp thứ hai là có sự tương ứng một-một với một hoặc nhiều lớp thứ hai.

Trong đó luồng bit là chuỗi bit tạo thành một hoặc nhiều chuỗi video được lập mã (Coded Video Sequence - CVS).

Trong đó chuỗi video được lập mã (Coded Video Sequence - CVS) là chuỗi AU (Access Unit - đơn vị truy cập).

Trong đó chuỗi video lớp được lập mã (Coded Layer Video Sequence - CLVS) là chuỗi PU (Picture Unit - đơn vị ảnh) có cùng giá trị của `nuh_layer_id`.

Trong đó đơn vị truy cập (Access Unit - AU) là tập hợp PU mà thuộc về các lớp khác nhau và chứa các ảnh được lập mã được liên kết với cùng thời điểm để được xuất ra từ DPB (Decoded Picture Buffer - bộ đệm ảnh được giải mã).

Trong đó đơn vị ảnh (Picture Unit - PU) là tập hợp đơn vị NAL (Network Abstraction Layer - lớp trừu tượng mạng) mà được liên kết với nhau theo quy tắc phân loại được quy định, liền nhau theo thứ tự giải mã, và chứa chính xác một ảnh được lập mã.

Trong đó ảnh tham chiếu liên lớp (Inter-Layer Reference Picture - ILRP) là ảnh trong cùng AU với ảnh hiện tại, có `nuh_layer_id` nhỏ hơn `nuh_layer_id` của ảnh hiện tại.

Trong đó SPS (Sequence Parameters Set - tập hợp thông số chuỗi) là cấu trúc cú pháp chứa các phần tử cú pháp mà áp dụng cho không hoặc nhiều hơn không CLVS toàn bộ.

Trong đó nếu lớp A sử dụng lớp B làm lớp tham chiếu, thì lớp B là lớp tham chiếu trực tiếp cho lớp A; nếu lớp A sử dụng lớp B làm lớp tham chiếu, lớp B sử dụng lớp C làm lớp tham chiếu, nhưng lớp A không sử dụng lớp C làm lớp tham chiếu, thì lớp C không phải là lớp tham chiếu trực tiếp cho lớp A.

Theo một dạng thực hiện khả thi của phương pháp theo khía cạnh thứ nhất như vậy, trong đó việc phân tử cú pháp thứ nhất bằng 1 quy định rằng lớp thứ nhất không sử dụng việc dự đoán liên lớp; hoặc việc phân tử cú pháp thứ nhất bằng 0 quy định rằng lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

Theo một dạng thực hiện khả thi của phương pháp theo cách thức thực hiện bất kỳ nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất như vậy, trong đó việc phân tử cú pháp thứ hai bằng 0 quy định rằng lớp thứ hai, mà liên quan đến phần tử cú pháp thứ hai, không phải là lớp tham chiếu trực tiếp cho lớp thứ nhất; hoặc việc phân tử cú pháp thứ hai bằng 1 quy định rằng lớp thứ hai, mà liên quan đến phần tử cú pháp thứ hai, là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Theo một dạng thực hiện khả thi của phương pháp theo cách thức thực hiện bất kỳ nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất như vậy, trong đó bước thu thập một hoặc nhiều phần tử cú pháp thứ hai là được thực hiện trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

Theo một dạng thực hiện khả thi của phương pháp theo cách thức thực hiện bất kỳ nêu trên của khía cạnh thứ nhất hoặc khía cạnh thứ nhất như vậy, trong đó phương pháp này còn bao gồm bước: thực hiện việc dự đoán đối với ảnh của lớp thứ nhất mà không sử dụng ảnh của lớp liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp.

Theo khía cạnh thứ hai, sáng chế liên quan đến phương pháp mã hoá luồng bit video được lập mã. Phương pháp này được thực hiện bởi bộ mã hoá. Phương pháp này bao gồm các bước: xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; mã hoá phần tử cú pháp vào luồng bit video được lập mã, trong đó phần tử cú pháp này quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; trong đó giá trị của phần tử cú pháp này quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp, trong trường hợp mà không có lớp nào trong số ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Trong đó bước xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không là bao gồm các bước: xác định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất dựa trên việc xác định rằng chi phí tốc độ - độ méo thứ nhất là nhỏ hơn hoặc bằng chi phí tốc độ - độ méo thứ hai; xác định lớp thứ hai không phải là lớp tham chiếu trực tiếp cho lớp thứ nhất dựa trên việc xác định rằng chi phí tốc độ - độ méo thứ nhất là lớn hơn hoặc bằng chi phí tốc độ - độ méo thứ hai; trong đó chi phí tốc độ - độ méo thứ nhất là chi phí bởi việc sử dụng lớp thứ hai làm lớp tham chiếu trực tiếp cho lớp thứ nhất, chi phí tốc độ - độ méo thứ hai là chi phí không sử dụng lớp thứ hai làm lớp tham chiếu trực tiếp cho lớp thứ nhất.

Theo một dạng thực hiện khả thi của phương pháp theo khía cạnh thứ hai như vậy, trong đó giá trị của phần tử cú pháp quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp, trong trường hợp mà ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Theo khía cạnh thứ ba, sáng chế liên quan đến thiết bị để giải mã luồng bit video được lập mã. Thiết bị này bao gồm: đơn vị thu thập, được tạo cấu hình để thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; đơn vị thu thập còn được tạo cấu hình để thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp; và đơn vị dự đoán được tạo cấu hình để thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu.

Theo một dạng thực hiện khả thi của phương pháp theo khía cạnh thứ ba như vậy, trong đó việc phần tử cú pháp thứ nhất bằng 1 quy định rằng lớp thứ nhất không sử dụng việc dự đoán liên lớp; hoặc việc phần tử cú pháp thứ nhất bằng 0 quy định rằng lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

Theo một dạng thực hiện khả thi của phương pháp theo cách thức thực hiện bất kỳ nêu trên của khía cạnh thứ ba hoặc khía cạnh thứ ba như vậy, trong đó việc phần tử cú pháp thứ hai bằng 0 quy định rằng lớp thứ hai, mà liên quan đến phần tử cú pháp thứ hai, không phải là lớp tham chiếu trực tiếp cho lớp thứ nhất; hoặc việc phần tử cú pháp thứ hai bằng 1 quy định rằng lớp thứ hai, mà liên quan đến phần tử cú pháp thứ hai, là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Theo một dạng thực hiện khả thi của phương pháp theo cách thức thực hiện bất kỳ nêu trên của khía cạnh thứ ba hoặc khía cạnh thứ ba như vậy, thì đơn vị dự đoán mà được tạo cấu hình để thu thập một hoặc nhiều phần tử cú pháp thứ hai là được thực hiện trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

Theo khía cạnh thứ tư, sáng chế liên quan đến thiết bị để mã hoá luồng bit video được lập mã. Thiết bị này bao gồm: đơn vị xác định được tạo cấu hình để xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; đơn vị mã hoá được tạo cấu hình để mã hoá phần tử cú pháp vào luồng bit video được

lập mã, trong đó phần tử cú pháp này quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; trong đó giá trị của phần tử cú pháp này quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp, trong trường hợp mà không có lớp nào trong số ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Theo một dạng thực hiện khả thi của phương pháp theo khía cạnh thứ tư như vậy, trong đó giá trị của phần tử cú pháp quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp, trong trường hợp mà ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Phương pháp theo khía cạnh thứ nhất của sáng chế là có thể được thực hiện bởi thiết bị theo khía cạnh thứ ba của sáng chế. Các dấu hiệu và các dạng thực hiện nữa của phương pháp theo khía cạnh thứ ba của sáng chế là tương ứng với các dấu hiệu và các dạng thực hiện của thiết bị theo khía cạnh thứ nhất của sáng chế.

Phương pháp theo khía cạnh thứ hai của sáng chế là có thể được thực hiện bởi thiết bị theo khía cạnh thứ tư của sáng chế. Các dấu hiệu và các dạng thực hiện nữa của phương pháp theo khía cạnh thứ tư của sáng chế là tương ứng với các dấu hiệu và các dạng thực hiện của thiết bị theo khía cạnh thứ hai của sáng chế.

Phương pháp theo khía cạnh thứ hai có thể được mở rộng thành các dạng thực hiện tương ứng với các dạng thực hiện của thiết bị thứ nhất theo khía cạnh thứ nhất. Do đó, dạng thực hiện của phương pháp này là bao gồm (các) dấu hiệu của dạng thực hiện tương ứng của thiết bị thứ nhất.

Các ưu điểm của các phương pháp theo khía cạnh thứ hai là giống như các ưu điểm đối với các dạng thực hiện tương ứng của thiết bị thứ nhất theo khía cạnh thứ nhất.

Theo khía cạnh thứ năm, sáng chế liên quan đến thiết bị để giải mã luồng video, thiết bị này bao gồm bộ xử lý và bộ nhớ. Bộ nhớ lưu giữ các chỉ dẫn mà khiến bộ xử lý thực hiện phương pháp theo khía cạnh thứ nhất.

Theo khía cạnh thứ sáu, sáng chế đề cập đến thiết bị để mã hoá luồng video, bao gồm bộ xử lý và bộ nhớ. Bộ nhớ lưu giữ các chỉ dẫn mà khiến bộ xử lý thực hiện phương pháp theo khía cạnh thứ hai.

Theo khía cạnh thứ bảy, sáng chế đề xuất phương tiện lưu trữ đọc được bằng máy tính có lưu giữ trên đó các chỉ dẫn mà khi được thực thi thì khiến một hoặc nhiều bộ xử lý được tạo cấu hình để lập mã dữ liệu video. Các chỉ dẫn này khiến một hoặc

nhiều bộ xử lý này thực hiện phương pháp theo khía cạnh thứ nhất hoặc khía cạnh thứ hai hoặc phương án khả thi bất kỳ của khía cạnh thứ nhất hoặc khía cạnh thứ hai.

Theo khía cạnh thứ tám, sáng chế liên quan đến chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo khía cạnh thứ nhất hoặc khía cạnh thứ hai hoặc phương án khả thi bất kỳ của khía cạnh thứ nhất hoặc khía cạnh thứ hai khi được thực thi trên máy tính.

Theo khía cạnh thứ chín, sáng chế liên quan đến phương tiện lưu trữ phi nhất thời mà bao gồm luồng bit được lập mã để được giải mã bởi thiết bị, luồng bit này được tạo ra bằng cách chia khung của tín hiệu video hoặc tín hiệu hình ảnh thành nhiều khối, và bao gồm nhiều phần tử cú pháp, trong đó các phần tử cú pháp này bao gồm phần tử cú pháp thứ nhất mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không, và một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp.

Các chi tiết về một hoặc nhiều phương án được nêu trên các hình vẽ kèm theo và phần mô tả dưới đây. Các dấu hiệu, các mục đích, và các ưu điểm khác sẽ được làm rõ từ phần mô tả, các hình vẽ, và các điểm yêu cầu bảo hộ.

Ngoài ra, các phương án sau đây được cung cấp.

Theo một phương án, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước:

phân tích phần tử cú pháp thứ nhất mà quy định xem lớp có chỉ số i có sử dụng việc dự đoán liên lớp hay không, i là số nguyên và i lớn hơn 0;

khi điều kiện thứ nhất được thoả mãn, thì phân tích phần tử cú pháp thứ hai mà quy định xem lớp có chỉ số j có phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i hay không, trong đó j là số nguyên, và j nhỏ hơn i và lớn hơn hoặc bằng 0, trong đó điều kiện thứ nhất này bao gồm việc phần tử cú pháp thứ nhất quy định rằng lớp có chỉ số i có thể sử dụng việc dự đoán liên lớp, j bằng $i-1$, và lớp bất kỳ trong số các lớp có chỉ số nhỏ hơn j đều không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i ;

dự đoán ảnh của lớp có chỉ số i dựa trên giá trị của phần tử cú pháp thứ hai.

Theo một phương án, trong đó khi phần tử cú pháp thứ nhất quy định rằng lớp có chỉ số i có thể sử dụng việc dự đoán liên lớp, thì tổng của $vps_direct_dependency_flag[i][k]$ là lớn hơn 0 với k đối với tất cả trong số các số nguyên trong khoảng từ 0 đến $i-1$, trong đó việc $vps_direct_dependency_flag$ bằng 1 quy định lớp có chỉ số k là lớp tham chiếu trực tiếp cho lớp có chỉ số i , $vps_direct_dependency_flag$ bằng 0 quy định lớp có chỉ số k không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i .

Theo một phương án, trong đó khi phần tử cú pháp thứ nhất quy định rằng lớp có chỉ số i có thể sử dụng việc dự đoán liên lớp, thì ít nhất một giá trị của $vps_direct_dependency_flag[i][k]$ là bằng 1, trong đó k là số nguyên và k nằm trong khoảng từ 0 đến $i-1$, trong đó việc $vps_direct_dependency_flag$ bằng 1 quy định lớp có chỉ số k là lớp tham chiếu trực tiếp cho lớp có chỉ số i , $vps_direct_dependency_flag$ bằng 0 quy định lớp có chỉ số k không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i .

Theo một phương án, trong đó ảnh của lớp có chỉ số i là bao gồm ảnh ở lớp có chỉ số i hoặc ảnh liên quan đến lớp có chỉ số i .

Theo một phương án, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước:

phân tích phần tử cú pháp mà quy định xem lớp có chỉ số i có sử dụng việc dự đoán liên lớp hay không, i là số nguyên và i lớn hơn 0;

khi điều kiện được thoả mãn, thì dự đoán ảnh của lớp có chỉ số i nhờ sử dụng lớp có chỉ số j làm lớp tham chiếu trực tiếp cho lớp có chỉ số i , trong đó j là số nguyên, và j bằng $i-1$, trong đó điều kiện này bao gồm việc phần tử cú pháp quy định rằng lớp có chỉ số i có thể sử dụng việc dự đoán liên lớp.

Theo một phương án, trong đó ảnh của lớp có chỉ số i là bao gồm ảnh ở lớp có chỉ số i hoặc ảnh liên quan đến lớp có chỉ số i .

Theo một phương án, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước:

phân tích phần tử cú pháp mà quy định xem ít nhất một ảnh tham chiếu dài hạn (Long-Term Reference Picture - LTRP) có được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong chuỗi video được lập mã (Coded Video Sequence - CVS) hay không, trong đó mỗi ảnh của ít nhất một LTRP này được đánh dấu là "used for long-

term reference" (được sử dụng cho việc tham chiếu dài hạn), nhưng không phải là ảnh tham chiếu liên lớp (Inter-Layer Reference Picture - ILRP);

dự đoán một hoặc nhiều ảnh được lập mã trong CVS dựa trên giá trị của phần tử cú pháp này.

Theo một phương án, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước:

Xác định xem điều kiện có được thoả mãn hay không, trong đó điều kiện này bao gồm việc chỉ số lớp của lớp hiện tại là lớn hơn giá trị được thiết đặt trước;

Khi điều kiện này được thoả mãn, thì phân tích phần tử cú pháp thứ nhất mà quy định xem ít nhất một ảnh tham chiếu liên lớp (Inter-Layer Reference Picture - ILRP) có được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong chuỗi video được lập mã (Coded Video Sequence - CVS) hay không;

dự đoán một hoặc nhiều ảnh được lập mã trong CVS dựa trên giá trị của phần tử cú pháp thứ nhất này.

Theo một phương án, trong đó giá trị được thiết đặt trước là 0.

Theo một phương án, trong đó điều kiện này còn bao gồm việc phần tử cú pháp thứ hai (ví dụ, `sps_video_parameter_set_id`) là lớn hơn 0.

Theo một phương án, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã, phương pháp này bao gồm các bước:

Xác định xem điều kiện có được thoả mãn hay không, trong đó điều kiện này bao gồm việc chỉ số lớp của lớp hiện tại là lớn hơn giá trị được thiết đặt trước và mục nhập hiện tại trong cấu trúc danh sách ảnh tham chiếu là mục nhập ILRP;

Khi điều kiện này được thoả mãn, thì phân tích phần tử cú pháp mà quy định chỉ số đến danh sách các lớp phụ thuộc trực tiếp của lớp hiện tại;

dự đoán một hoặc nhiều ảnh được lập mã trong CVS dựa trên cấu trúc danh sách ảnh tham chiếu mà mục nhập hiện tại của nó là ILRP được thu thập nhờ sử dụng chỉ số đến danh sách lớp phụ thuộc trực tiếp.

Theo một phương án, trong đó giá trị được thiết đặt trước là 1.

Theo một phương án, sáng chế đề xuất bộ mã hoá (20) bao gồm hệ mạch xử lý để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 12.

Theo một phương án, sáng chế đề xuất bộ giải mã (30) bao gồm hệ mạch xử lý để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 12.

Theo một phương án, sáng chế đề xuất sản phẩm chương trình máy tính, bao gồm mã chương trình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm nêu trên khi được thực thi trên máy tính hoặc bộ xử lý.

Theo một phương án, sáng chế đề xuất bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ phi nhất thời đọc được bằng máy tính được ghép nối đến các bộ xử lý và lưu giữ chương trình để thực thi bởi các bộ xử lý này, trong đó chương trình này, khi được thực thi bởi các bộ xử lý này, thì tạo cấu hình cho bộ giải mã này để thực hiện phương pháp theo điểm bất kỳ trong số các điểm nêu trên.

Theo một phương án, sáng chế đề xuất bộ mã hoá, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ phi nhất thời đọc được bằng máy tính được ghép nối đến các bộ xử lý và lưu giữ chương trình để thực thi bởi các bộ xử lý này, trong đó chương trình này, khi được thực thi bởi các bộ xử lý này, thì tạo cấu hình cho bộ mã hoá này để thực hiện phương pháp theo điểm bất kỳ trong số các điểm nêu trên.

Theo một phương án, sáng chế đề xuất phương tiện phi nhất thời đọc được bằng máy tính, có mang mã chương trình mà khi được thực thi bởi thiết bị máy tính thì khiến thiết bị máy tính đó thực hiện phương pháp theo điểm bất kỳ trong số các điểm nêu trên.

Mô tả vắn tắt các hình vẽ

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

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

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

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

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

Fig.4 là hình vẽ thể hiện sơ đồ khối theo một ví dụ của thiết bị mã hoá hoặc thiết bị giải mã;

Fig.5 là hình vẽ thể hiện sơ đồ khối theo ví dụ khác của thiết bị mã hoá hoặc thiết bị giải mã;

Fig.6 là hình vẽ thể hiện sơ đồ khối của việc lập mã khả biến với 2 lớp;

Fig.7 là hình vẽ thể hiện sơ đồ khối cấu trúc ví dụ của hệ thống cung cấp nội dung 3100 mà thực hiện dịch vụ phân phối nội dung.

Fig.8 là hình vẽ thể hiện sơ đồ khối cấu trúc theo một ví dụ của thiết bị đầu cuối.

Fig.9 là hình vẽ thể hiện lưu đồ của phương pháp giải mã theo một phương án.

Fig.10 là hình vẽ thể hiện lưu đồ của phương pháp mã hoá theo một phương án.

Fig.11 là hình vẽ thể hiện sơ đồ của bộ mã hoá theo một phương án.

Fig.12 là hình vẽ thể hiện sơ đồ của bộ giải mã theo một phương án.

Trong phần sau đây thì các ký hiệu chỉ dẫn giống nhau được dùng để chỉ các dấu hiệu giống nhau hoặc ít nhất là tương đương về mặt chức năng, nếu không được quy định khác đi một cách rõ ràng.

Mô tả chi tiết các phương án thực hiện sáng chế

Trong phần mô tả sau đây thì các hình vẽ kèm theo sẽ được mô tả, vốn tạo thành một phần của phần bộc lộ này, và thể hiện, bằng cách minh hoạ, 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. Cần hiểu rằng các phương án của sáng chế có thể được sử dụng theo các khía cạnh khác và bao gồm những sự thay đổi về cấu trúc hoặc logic mà không được thể hiện trên các hình vẽ. Do đó, phần mô tả chi tiết sau đây là không nhằm mục đích giới hạn, và phạm vi của sáng chế là được xác định bởi các điểm yêu cầu bảo hộ kèm theo.

Ví dụ, cần hiểu rằng sự bộc lộ liên quan đến phương pháp được mô tả cũng có thể là đúng đối 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 phương pháp cụ thể được mô tả, thì thiết bị tương ứng có thể bao gồm một hoặc nhiều đơn vị, ví dụ, các đơn vị chức năng, để thực hiện một hoặc nhiều trong số các bước phương pháp được mô tả đó (ví dụ, một đơn vị thực hiện một hoặc nhiều bước đó, hoặc mỗi trong số các đơn vị thực hiện một hoặc nhiều trong số các bước đó), ngay cả khi một hoặc nhiều đơn vị đó không được mô tả hoặc thể hiện một cách tường minh trên các hình vẽ. Ngược lại, ví dụ, nếu thiết bị cụ thể được mô tả dựa trên một hoặc nhiều đơn vị, ví dụ, các đơn vị

chức năng, thì phương pháp tương ứng có thể bao gồm một bước để thực hiện chức năng của một hoặc nhiều trong số các đơn vị đó (ví dụ, một bước thực hiện chức năng của một hoặc nhiều trong số các đơn vị, hoặc mỗi trong số các bước thực hiện chức năng của một hoặc nhiều trong số các đơn vị đó), ngay cả khi một hoặc nhiều trong số các bước đó không được mô tả hoặc thể hiện một cách tường minh trên các hình vẽ. Ngoài ra, cần hiểu rằng các dấu hiệu của các phương án ví dụ và/hoặc các khía cạnh khác nhau được mô tả ở đây là có thể được kết hợp với nhau, trừ khi được nói khác đi một cách cụ thể.

Việc lập mã video thường là việc xử lý chuỗi ảnh, mà tạo thành video hoặc chuỗi video. Thay vì thuật ngữ “ảnh”, thì thuật ngữ “khung” hoặc “hình ảnh” có thể được sử dụng như các từ đồng nghĩa trong lĩnh vực lập mã video. Việc lập mã video (hay lập mã nói chung) bao gồm hai phần, là mã hoá video và giải mã video. Việc mã hoá video được thực hiện ở phía nguồn, thường bao gồm việc xử lý (ví dụ, bằng cách nén) các ảnh video gốc để giảm lượng dữ liệu cần thiết để biểu diễn các ảnh video này (để lưu giữ và/hoặc truyền một cách hiệu quả hơn). Việc giải mã video được thực hiện ở phía đích và thường bao gồm việc xử lý ngược so với bộ mã hoá để tái tạo các ảnh video. Các phương án đề cập đến việc “lập mã” các ảnh video (hoặc các ảnh nói chung) sẽ được hiểu là liên quan đến việc “mã hoá” hoặc “giải mã” các ảnh video hoặc các chuỗi video tương ứng. Sự kết hợp của phần mã hoá và phần giải mã cũng được gọi là CODEC (COding and DECoдинг - lập mã và giải mã).

Trong trường hợp lập mã video không tổn hao, thì các ảnh video gốc có thể được tái tạo, tức là các ảnh video được tái tạo có chất lượng giống như các ảnh video gốc (giả sử không có sự tổn hao truyền hoặc sự tổn hao dữ liệu khác trong quá trình lưu giữ hoặc truyền). Trong trường hợp lập mã video có tổn hao, thì việc nén thêm, ví dụ, bằng cách lượng tử hoá, là được thực hiện, để giảm lượng dữ liệu biểu diễn các ảnh video, vốn không thể được tái tạo hoàn toàn tại bộ giải mã, tức là, chất lượng của các ảnh video được tái tạo là thấp hơn hoặc xấu hơn so với chất lượng của các ảnh video gốc.

Một số tiêu chuẩn lập mã video là thuộc về nhóm “các codec video lai có tổn hao” (tức là kết hợp dự đoán về mặt không gian và dự đoán về mặt thời gian trong miền mẫu và lập mã biến đổi 2D để áp dụng việc lượng tử hoá trong miền biến đổi). Mỗi ảnh của chuỗi video thường được phân vùng thành tập hợp các khối không chồng nhau và việc lập mã thường được thực hiện trên mức độ khối. Nói cách khác, tại bộ mã hoá thì

video thường được xử lý, tức là được mã hoá, trên mức độ khối (khối video), ví dụ, nhờ sử dụng việc 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 này từ khối hiện tại (khối được xử lý hiện tại/cần được xử lý) để thu được khối dư, biến đổi khối dư và lượng tử hoá khối dư này trong miền biến đổi để giảm lượng dữ liệu cần được truyền (nén), còn tại bộ giải mã thì tiến trình xử lý ngược so với bộ mã hoá là được áp dụng cho khối đã được mã hoá hoặc được nén để tái tạo khối hiện tại để biểu diễn. Ngoài ra, bộ mã hoá nhân bản vòng lặp xử lý của bộ giải mã sao cho cả hai sẽ tạo ra các kết quả dự đoán giống nhau (ví dụ, kết quả nội dự đoán và kết quả liên dự đoán) và/hoặc các kết quả tái tạo giống nhau để xử lý, tức là lập mã, các khối sau đó.

Ở các phương án sau đây về hệ thống lập mã video 10, thì bộ mã hoá video 20 và bộ giải mã video 30 được mô tả dựa trên các hình vẽ từ Fig.1 đến Fig.3.

Fig.1A là hình vẽ thể hiện sơ đồ khối của hệ thống lập mã 10 ví dụ, ví dụ, hệ thống lập mã video 10 (hoặc ngắn gọn là hệ thống lập mã 10) mà có thể sử dụng các kỹ thuật của sáng chế. Bộ mã hoá video 20 (hoặc ngắn gọn là bộ mã hoá 20) và bộ giải mã video 30 (hoặc ngắn gọn là bộ giải mã 30) của hệ thống lập mã video 10 biểu thị các ví dụ về các thiết bị mà có thể được tạo cấu hình để thực hiện các kỹ thuật theo các ví dụ khác nhau được mô tả theo sáng chế.

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

Thiết bị nguồn 12 bao gồm bộ mã hoá 20, và có thể còn, tức là tùy ý, bao gồm nguồn ảnh 16, bộ tiền xử lý (hoặc đơn vị tiền xử lý) 18, ví dụ, bộ tiền xử lý ảnh 18, và giao diện truyền thông hoặc đơn vị truyền thông 22.

Nguồn ảnh 16 có thể bao gồm hoặc là loại thiết bị chụp ảnh bất kỳ, ví dụ, camera để chụp ảnh thế giới thực, và/hoặc là loại thiết bị tạo ảnh bất kỳ, ví dụ, bộ xử lý đồ hoạ máy tính để tạo ra ảnh hoạt hình máy tính, hoặc bất kỳ loại thiết bị nào khác để thu thập và/hoặc cung cấp ảnh thế giới thực, ảnh được tạo ra bằng máy tính (ví dụ, nội dung trên màn hình, ảnh thực tế ảo (Virtual Reality - VR)) và/hoặc tổ hợp bất kỳ của chúng (ví dụ, ảnh thực tế tăng cường (Augmented Reality - AR)). Nguồn ảnh có thể là loại bộ nhớ hoặc bộ lưu trữ bất kỳ mà lưu giữ ảnh bất kỳ trong số các ảnh nêu trên.

Khi phân biệt bộ tiền xử lý 18 và tiến trình xử lý được thực hiện bởi đơn vị tiền xử lý 18, thì ảnh hoặc dữ liệu ảnh 17 cũng có thể được gọi là ảnh thô hoặc dữ liệu ảnh thô 17.

Bộ tiền xử lý 18 được tạo cấu hình để nhận dữ liệu ảnh (thô) 17 và thực hiện việc tiền xử lý trên dữ liệu ảnh 17 này để thu được 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 việc cắt xén, chuyển đổi định dạng màu (ví dụ, từ RGB sang YCbCr), hiệu chỉnh màu, hoặc khử nhiễu. Có thể hiểu rằng đơn vị tiền xử lý 18 có thể là thành phần tùy ý.

Bộ mã hoá video 20 được tạo cấu hình để nhận dữ liệu ảnh được tiền xử lý 19 và cung cấp dữ liệu ảnh được mã hoá 21 (các chi tiết 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 được tạo cấu hình để nhận dữ liệu ảnh được mã hoá 21 và truyền dữ liệu ảnh được mã hoá 21 (hoặc bất kỳ phiên bản được xử lý tiếp nào của nó) qua kênh truyền thông 13 đến thiết bị khác, ví dụ, thiết bị đích 14 hoặc thiết bị bất kỳ khác, để lưu trữ hoặc tái tạo trực tiếp.

Thiết bị đích 14 bao gồm bộ giải mã 30 (ví dụ, bộ giải mã video 30), và có thể còn, tức là tùy ý, bao gồm giao diện truyền thông hoặc đơn vị truyền thông 28, bộ hậu xử lý 32 (hoặc đơn vị hậu xử lý 32) và thiết bị hiển thị 34.

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

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

Giao diện truyền thông 22 có thể, ví dụ, được tạo cấu hình để gói dữ liệu ảnh được mã hoá 21 thành định dạng thích hợp, ví dụ, các gói, và/hoặc xử lý dữ liệu ảnh

được mã hoá nhờ sử dụng loại mã hoá hoặc xử lý truyền bất kỳ để truyền trên liên kết truyền thông hoặc mạng truyền thông.

Giao diện truyền thông 28, mà tạo thành đôi tác của giao diện truyền thông 22, có thể, ví dụ, được tạo cấu hình để nhận dữ liệu được truyền và xử lý dữ liệu truyền này nhờ sử dụng loại giải mã hoặc xử lý và/hoặc mở gói truyền tương ứng bất kỳ để thu được dữ liệu ảnh được mã hoá 21.

Cả giao diện truyền thông 22 và giao diện truyền thông 28 đều có thể được tạo cấu hình dưới dạng các giao diện truyền thông một chiều như được chỉ thị bằng hình mũi tên đối với kênh truyền thông 13 trên Fig.1A mà trở từ thiết bị nguồn 12 đến thiết bị đích 14, hoặc các giao diện truyền thông hai chiều, và có thể được tạo cấu hình, ví dụ, để gửi và nhận các thông điệp, ví dụ, để cài đặt kết nối, để báo nhận và trao đổi thông tin bất kỳ khác liên quan đến liên kết truyền thông và/hoặc việc truyền dữ liệu, ví dụ, truyền dữ liệu ảnh được mã hoá.

Bộ giải mã 30 được tạo cấu hình để nhận dữ liệu ảnh được mã hoá 21 và cung cấp dữ liệu ảnh được giải mã 31 hoặc ảnh được giải mã 31 (các chi tiết nữa 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 được tạo cấu hình để hậu xử lý dữ liệu ảnh được giải mã 31 (còn được gọi là dữ liệu ảnh được tái tạo), ví dụ, ảnh được giải mã 31, để thu được dữ liệu ảnh được hậu xử lý 33, ví dụ, ảnh được hậu xử lý 33. Việc hậu xử lý mà được thực hiện bởi đơn vị hậu xử lý 32 là có thể bao gồm, ví dụ, việc chuyển đổi định dạng màu (ví dụ, từ YCbCr sang RGB), hiệu chỉnh màu, cắt xén, hoặc lấy mẫu lại, hoặc việc xử lý bất kỳ khác, ví dụ, để chuẩn bị dữ liệu ảnh được giải mã 31 để hiển thị, ví dụ, bởi thiết bị hiển thị 34.

Thiết bị hiển thị 34 của thiết bị đích 14 được tạo cấu hình để nhận dữ liệu ảnh được hậu xử lý 33 để hiển thị ảnh, ví dụ, cho người dùng hoặc người xem. Thiết bị hiển thị 34 có thể là hoặc bao gồm loại thiết bị hiển thị bất kỳ để biểu diễn ảnh được tái tạo, ví dụ, thiết bị hiển thị hoặc màn hình tích hợp hoặc thiết bị hiển thị hoặc màn hình ngoài. Các thiết bị hiển thị có thể, ví dụ, bao gồm thiết bị hiển thị tinh thể lỏng (Liquid Crystal Display - LCD), thiết bị hiển thị đèn điốt phát sáng hữu cơ (Organic Light Emitting Diode - OLED), thiết bị hiển thị plasma, máy chiếu, thiết bị hiển thị vi LED, tinh thể lỏng trên silic (Liquid Crystal on Silicon - LCoS), bộ xử lý ánh sáng số (Digital Light Processor - DLP) hoặc bất kỳ loại thiết bị hiển thị nào khác.

Tuy Fig.1A thể hiện thiết bị nguồn 12 và thiết bị đích 14 dưới dạng các thiết bị riêng biệt, nhưng các phương án của các thiết bị cũng có thể bao gồm cả hai thiết bị đó 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 đó thì 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 cùng phần cứng và/hoặc phần mềm hoặc bằng phần cứng và/hoặc phần mềm riêng biệt hoặc tổ hợp bất kỳ của chúng.

Như người có hiểu biết trung bình sẽ thấy rõ dựa trên phần mô tả này, thì sự tồn tại và việc chia (chính xác) đối với các chức năng của các đơn vị khác nhau hoặc các chức năng trong thiết bị nguồn 12 và/hoặc thiết bị đích 14 như được thể hiện trên Fig.1A là có thể biến thiên tùy theo thiết bị và ứng dụng thực tế.

Bộ mã hoá 20 (ví dụ, bộ mã hoá video 20) hoặc bộ giải mã 30 (ví dụ, bộ giải mã video 30) hoặc cả bộ mã hoá 20 và bộ giải mã 30 có thể được thực hiện thông qua hệ mạch xử lý như được thể hiện trên Fig.1B, chẳng hạn một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (Digital Signal Processor - DSP), mạch tích hợp chuyên dụng (Application-Specific Integrated Circuit - ASIC), mảng cổng lập trình được dạng trường (Field-Programmable Gate Array - FPGA), logic rời rạc, phần cứng, lập mã video dành riêng hoặc các tổ hợp bất kỳ của chúng. Bộ mã hoá 20 có thể được thực hiện thông qua hệ mạch xử lý 46 để thực hiện các môđun khác nhau như được mô tả đối với bộ mã hoá 20 trên Fig.2 và/hoặc hệ thống mã hoá hoặc hệ thống mã hoá con bất kỳ khác được mô tả ở đây. Bộ giải mã 30 có thể được thực hiện thông qua hệ mạch xử lý 46 để thực hiện các môđun khác nhau như được mô tả đối với bộ giải mã 30 trên Fig.3 và/hoặc hệ thống giải mã hoặc hệ thống giải mã con bất kỳ khác được mô tả ở đây. Hệ mạch xử lý có thể được tạo cấu hình 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 này được thực hiện một phần bằng phần mềm, thì thiết bị có thể lưu giữ các chỉ dẫn dành cho phần mềm đó trong, phương tiện lưu trữ phi nhất thời phù hợp được bằng máy tính và có thể thực thi các chỉ dẫn này bằng phần cứng nhờ sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật theo sáng chế. Bất kỳ trong số bộ mã hoá video 20 và bộ giải mã video 30 có thể được tích hợp như một phần của bộ mã hoá/bộ giải mã (CODEC) được kết hợp trong thiết bị đơn lẻ, ví dụ, như được thể hiện trên Fig.1B.

Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm thiết bị bất kỳ trong số nhiều loại thiết bị, bao gồm loại thiết bị cầm tay hoặc thiết bị cố định bất kỳ, ví dụ, máy ghi chép hoặc máy tính xách tay, điện thoại di động, điện thoại thông minh, máy tính bảng, camera, máy tính để bàn, hộp thu tín hiệu truyền hình, ti vi, thiết bị hiển thị, máy chơi phương tiện số, máy chơi video game, thiết bị tạo luồng video (chẳng hạn máy chủ dịch vụ nội dung hoặc máy chủ phân phối nội dung), thiết bị thu quảng bá, thiết bị phát quảng bá, v.v., và có thể không sử dụng loại hệ điều hành nào hoặc sử dụng loại hệ điều hành bất kỳ. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị cho việc truyền thông không dây. Do đó, thiết bị nguồn 12 và thiết bị đích 14 có thể là các thiết bị truyền thông không dây.

Trong một số trường hợp, thì hệ thống lập mã video 10 được thể hiện trên Fig.1A chỉ là ví dụ, và các kỹ thuật theo sáng chế có thể áp dụng cho các thiết đặt lập mã video (ví dụ, mã hoá video hoặc giải mã video) mà không nhất thiết phải bao gồm việc truyền thông dữ liệu nào giữa thiết bị mã hoá và thiết bị giải mã. Theo các ví dụ khác, thì dữ liệu được truy hồi từ bộ nhớ cục bộ, được tạo luồng qua mạng, v.v.. Thiết bị mã hoá video có thể mã hoá và lưu giữ dữ liệu vào bộ nhớ, và/hoặc thiết bị giải mã video có thể truy hồi và giải mã dữ liệu từ bộ nhớ. Theo một số ví dụ, thì việc mã hoá và giải mã là được thực hiện bởi các thiết bị không truyền thông với nhau, mà chỉ đơn giản là mã hoá dữ liệu vào bộ nhớ và/hoặc truy hồi và giải mã dữ liệu từ bộ nhớ.

Để thuận tiện cho việc mô tả, thì các phương án theo sáng chế được mô tả ở đây, ví dụ, dựa vào tiêu chuẩn lập mã video hiệu quả cao (High-Efficiency Video Coding - HEVC) hoặc phần mềm tham chiếu của tiêu chuẩn lập mã video vạn năng (Versatile Video Coding - VVC), là tiêu chuẩn lập mã video thế hệ tiếp theo được phát triển bởi nhóm hợp tác chung về lập mã video (Joint Collaboration Team on Video Coding - JCT-VC) của nhóm chuyên gia lập mã video (Video Coding Experts Group - VCEG) của ITU-T (International Telecommunication Union - hiệp hội viễn thông quốc tế) và nhóm chuyên gia ảnh động (Motion Picture Experts Group - MPEG) của ISO (International Organization for Standardization - tổ chức tiêu chuẩn hoá quốc tế)/IEC (International Electrotechnical Commission - ủy ban kỹ thuật điện quốc tế). Người có hiểu biết trung bình trong lĩnh vực sẽ hiểu rằng các phương án của sáng chế là không chỉ giới hạn ở HEVC hoặc VVC.

Bộ mã hoá và phương pháp mã hoá

Fig.2 là hình vẽ thể hiện sơ đồ khối của bộ mã hoá video 20 ví dụ mà được tạo cấu hình để thực hiện các kỹ thuật theo sáng chế. Theo ví dụ trên Fig.2, thì bộ mã hoá video 20 bao gồm đầu vào 201 (hoặc giao diện đầu vào 201), đơn vị tính phần dư 204, đơn vị xử lý biến đổi 206, đơn vị lượng tử hoá 208, đơn vị lượng tử hoá ngược 210, và đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, đơn vị bộ lọc vòng 220, bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230, đơn vị chọn chế độ 260, đơn vị mã hoá entropy 270 và đầu ra 272 (hoặc giao diện đầu ra 272). Đơn vị chọn chế độ 260 có thể bao gồm đơn vị liên dự đoán 244, đơn vị nội dự đoán 254 và đơn vị phân vùng 262. Đơn vị liên dự đoán 244 có thể bao gồm đơn vị ước lượng chuyển động và đơn vị bù chuyển động (không được thể hiện trên hình vẽ). Bộ mã hoá video 20 như được thể hiện trên Fig.2 cũng có thể được gọi là bộ mã hoá video lai hoặc bộ mã hoá video theo codec video lai.

Đơn vị tính phần dư 204, đơn vị xử lý biến đổi 206, đơn vị lượng tử hoá 208, đơn vị chọn chế độ 260 có thể được coi là tạo thành đường tín hiệu thuận của bộ mã hoá 20, còn đơn vị lượng tử hoá ngược 210, đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, bộ đệm 216, bộ lọc vòng 220, bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230, đơn vị liên dự đoán 244 và đơn vị nội dự đoán 254 có thể được coi là tạo thành đường tín hiệu ngược của bộ mã hoá video 20, trong đó đường tín hiệu ngược của bộ mã hoá video 20 là tương ứng với đường tín hiệu của bộ giải mã (xem bộ giải mã video 30 trên Fig.3). Đơn vị lượng tử hoá ngược 210, đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, bộ lọc vòng 220, bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230, đơn vị liên dự đoán 244 và đơn vị nội dự đoán 254 cũng được coi là tạo thành “bộ giải mã tích hợp” của bộ mã hoá video 20.

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

Bộ mã hoá 20 có thể được tạo cấu hình để nhận, ví dụ, qua đầu vào 201, ảnh 17 (hoặc dữ liệu ảnh 17), ví dụ, ảnh của chuỗi ảnh mà tạo thành video hoặc chuỗi video. Ảnh hoặc dữ liệu ảnh nhận được cũng có thể là ảnh được tiền xử lý 19 (hoặc dữ liệu ảnh được tiền xử lý 19). Để cho đơn giản thì phần mô tả sau đây đề cập đến ảnh 17. Ảnh 17 cũng có thể được gọi là ảnh hiện tại hoặc ảnh cần được lập mã (cụ thể là trong việc lập mã video để phân biệt ảnh hiện tại với các ảnh khác, ví dụ, các ảnh được mã hoá và/hoặc được giải mã trước đó của cùng chuỗi video, tức là chuỗi video mà cũng bao gồm ảnh hiện tại này).

Ảnh (kỹ thuật số) là hoặc có thể được coi là mảng hoặc ma trận hai chiều của các mẫu với các giá trị cường độ. Mẫu trong mảng này cũng có thể được gọi là điểm ảnh (dạng ngắn của phân tử ảnh) hoặc pel (điểm ảnh). Số lượng mẫu theo chiều (hoặc trục) ngang và dọc của mảng hoặc ảnh xác định kích thước và/hoặc độ phân giải của ảnh. Để biểu diễn màu, thì thông thường ba thành phần màu được sử dụng, tức là ảnh có thể được biểu diễn hoặc bao gồm ba mảng mẫu. Trong định dạng hoặc không gian màu RGB, thì ảnh bao gồm mảng mẫu đỏ, xanh lục và xanh lam tương ứng. Tuy nhiên, trong việc lập mã video thì mỗi điểm ảnh thường được biểu diễn trong định dạng hoặc không gian màu độ chói và độ màu, ví dụ, YCbCr, mà bao gồm thành phần độ chói được chỉ thị bởi Y (đôi khi còn có L được sử dụng thay vào đó) và hai thành phần độ màu được chỉ thị bởi Cb và Cr. Thành phần độ chói (hay viết ngắn gọn là luma) Y biểu thị độ sáng hoặc cường độ mức xám (ví dụ, như trong ảnh thang xám), còn hai thành phần độ màu (hay viết ngắn gọn là chroma) Cb và Cr thì biểu thị các thành phần thông tin về sắc hoặc màu. Theo đó, ảnh trong định dạng YCbCr thì bao gồm mảng mẫu độ chói của các giá trị mẫu độ chói (Y), và hai mảng mẫu độ màu của các giá trị độ màu (Cb và Cr). Các ảnh trong định dạng RGB có thể được chuyển đổi hoặc được biến đổi thành định dạng YCbCr và ngược lại, tiến trình này còn được gọi là biến đổi hoặc chuyển đổi màu. Nếu một ảnh là đơn sắc, thì ảnh đó có thể chỉ bao gồm mảng mẫu độ chói. Theo đó, ảnh có thể là, ví dụ, mảng mẫu độ chói trong định dạng đơn sắc hoặc mảng mẫu độ chói và hai mảng mẫu sắc độ tương ứng trong định dạng màu 4:2:0, 4:2:2, và 4:4:4.

Theo các phương án thì bộ mã hoá video 20 có thể bao gồm đơn vị phân vùng ảnh (không được thể hiện trên Fig.2) được tạo cấu hình để phân vùng ảnh 17 thành nhiều khối ảnh 203 (thường là không chồng nhau). Các khối này cũng có thể được gọi là các khối gốc, các khối macro (H.264/AVC) hoặc các khối cây lập mã (Coding Tree Block - CTB) hoặc các đơn vị cây lập mã (Coding Tree Unit - CTU) (H.265/HEVC và VVC). Đơn vị phân vùng ảnh có thể được tạo cấu hình để sử dụng kích thước khối bằng nhau cho tất cả các ảnh của chuỗi video và lưới tương ứng mà xác định kích thước khối, hoặc để thay đổi kích thước khối giữa các ảnh hoặc các tập hợp con hoặc các nhóm ảnh, và phân vùng mỗi ảnh thành các khối tương ứng.

Theo các phương án nữa, thì bộ mã hoá video có thể được tạo cấu hình để nhận trực tiếp khối 203 của ảnh 17, ví dụ, một, một số hoặc tất cả các khối mà tạo thành ảnh

17. Khối ảnh 203 cũng có thể được gọi là khối ảnh hiện tại hoặc khối ảnh cần được lập mã.

Cũng như ảnh 17, thì khối ảnh 203 là hoặc có thể được coi 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, mảng độ chói trong trường hợp ảnh đơn sắc 17, hoặc mảng độ chói hoặc mảng sắc độ trong trường hợp ảnh màu) hoặc ba mảng mẫu (ví dụ, mảng độ chói và hai mảng sắc độ trong trường hợp ảnh màu 17) hoặc số lượng và/hoặc loại mảng bất kỳ khác tùy theo định dạng màu được áp dụng. Số lượng mẫu theo chiều (hoặc trục) ngang và dọc của khối 203 xác định kích thước của khối 203. Theo đó, thì khối có thể là, ví dụ, mảng mẫu $M \times N$ (M cột N hàng), hoặc mảng hệ số biến đổi $M \times N$.

Theo các phương án thì bộ mã hoá video 20 như được thể hiện trên Fig.2 có thể được tạo cấu hình để mã hoá ảnh 17 từng khối một, ví dụ, việc mã hoá và dự đoán là được thực hiện trên mỗi khối 203.

Theo các phương án thì bộ mã hoá video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân vùng và/hoặc mã hoá ảnh nhờ sử dụng các lát cắt (còn được gọi là các lát cắt video), trong đó ảnh có thể được phân vùng thành hoặc được mã hoá nhờ sử dụng một hoặc nhiều lát cắt (thường là không chồng nhau), và mỗi lát cắt có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều nhóm khối (ví dụ, các ô (H.265/HEVC và VVC) hoặc các viên gạch (VVC)).

Theo các phương án thì bộ mã hoá video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân vùng và/hoặc mã hoá ảnh nhờ sử dụng các lát cắt/các nhóm ô (còn được gọi là các nhóm ô video) và/hoặc các ô (còn được gọi là các ô video), trong đó ảnh có thể được phân vùng thành hoặc được mã hoá nhờ sử dụng một hoặc nhiều lát cắt/nhóm ô (thường là không chồng nhau), và mỗi lát cắt/nhóm ô có thể bao gồm, ví dụ, một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều ô, trong đó mỗi ô, ví dụ, có thể có hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ, các khối hoàn chỉnh hoặc các phân đoạn khối.

Tính phần dư

Đơn vị tính phần dư 204 có thể được tạo cấu hình để tính khối dư 205 (còn được gọi là phần dư 205) dựa trên khối ảnh 203 và khối dự đoán 265 (các chi tiết nữa về khối dự đoán 265 sẽ được cung cấp sau), ví dụ, bằng cách trừ đi các giá trị mẫu của khối dự

đoán 265 từ các giá trị mẫu của khối ảnh 203, theo từng mẫu một (từng điểm ảnh một) để thu được khối dư 205 trong miền mẫu.

Biến đổi

Đơn vị xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng phép biến đổi, ví dụ, phép biến đổi cosin rời rạc (Discrete Cosine Transform - DCT) hoặc phép biến đổi sin rời rạc (Discrete Sine Transform - DST), trên các giá trị mẫu của khối dư 205 để thu được 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 thị khối dư 205 trong miền biến đổi.

Đơn vị xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng các phép xấp xỉ nguyên của DCT/DST, chẳng hạn các phép biến đổi được quy định cho H.265/HEVC. So với phép biến đổi DCT trực giao, thì các phép xấp xỉ nguyên đó thường được thay đổi tỷ lệ bằng hệ số nhất định. Để bảo tồn chuẩn đích của khối dư mà được xử lý bởi các phép biến đổi thuận và ngược, thì các hệ số thay đổi tỷ lệ bổ sung được áp dụng như một phần của tiến trình biến đổi. Các hệ số thay đổi tỷ lệ thường được chọn dựa trên các ràng buộc nhất định, chẳng hạn như các hệ số thay đổi tỷ lệ là lũy thừa của hai đối với các phép toán dịch, độ sâu bit của các hệ số biến đổi, sự đánh đổi giữa độ chính xác và các chi phí thực hiện, v.v.. Các hệ số thay đổi tỷ lệ cụ thể là, ví dụ, được quy định cho phép biến đổi ngược, ví dụ, bởi đơn vị xử lý biến đổi ngược 212 (và phép biến đổi ngược tương ứng, ví dụ, bởi đơn vị xử lý biến đổi ngược 312 tại bộ giải mã video 30) và các hệ số thay đổi tỷ lệ tương ứng cho phép biến đổi thuận, ví dụ, bởi đơn vị xử lý biến đổi 206, tại bộ mã hoá 20 là có thể được quy định theo đó.

Theo các phương án thì bộ mã hoá video 20 (tương ứng là đơn vị xử lý biến đổi 206) có thể được tạo cấu hình để xuất ra các thông số biến đổi, ví dụ, kiểu biến đổi hoặc các kiểu biến đổi, ví dụ, trực tiếp hoặc được mã hoá hoặc được nén qua đơn vị mã hoá entropy 270, để, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số biến đổi này để giải mã.

Lượng tử hoá

Đơn vị lượng tử hoá 208 có thể được tạo cấu hình để lượng tử hoá các hệ số biến đổi 207 để thu được các hệ số được lượng tử hoá 209, ví dụ, bằng cách áp dụng phép lượng tử hoá vô hướng hoặc lượng tử hoá vector. Các hệ số được lượng tử hoá 209 cũng có thể được gọi là các hệ số biến đổi được lượng tử hoá 209 hoặc các hệ số dư được lượng tử hoá 209.

Tiến trình lượng tử hoá có thể giảm chiều sâu bit được liên kết với một số hoặc tất cả trong số các hệ số biến đổi 207. Ví dụ, hệ số biến đổi n bit có thể được làm tròn xuống thành hệ số biến đổi m bit trong quá trình lượng tử hoá, trong đó n lớn hơn m . Mức độ lượng tử hoá có thể được cải biến bằng cách điều chỉnh thông số lượng tử hoá (Quantization Parameter - QP). Ví dụ, đối với việc lượng tử hoá vô hướng, thì việc thay đổi tỷ lệ khác nhau có thể được áp dụng để đạt được việc lượng tử hoá mịn hơn hoặc thô hơn. Các kích thước bước lượng tử nhỏ hơn thì tương ứng với việc lượng tử hoá mịn hơn, còn các kích thước bước lượng tử lớn hơn thì tương ứng với việc lượng tử hoá thô hơn. Kích thước bước lượng tử áp dụng được có thể được chỉ thị bởi thông số lượng tử hoá (Quantization Parameter - QP). Thông số lượng tử hoá có thể là, ví dụ, chỉ số đến tập hợp được định trước của các kích thước bước lượng tử áp dụng được. Ví dụ, các thông số lượng tử hoá nhỏ có thể tương ứng với việc lượng tử hoá mịn (các kích thước bước lượng tử nhỏ) và các thông số lượng tử hoá lớn có thể tương ứng với việc lượng tử hoá thô (các kích thước bước lượng tử lớn), hoặc ngược lại. Việc lượng tử hoá có thể bao gồm việc chia cho kích thước bước lượng tử, và việc giải lượng tử và/hoặc giải lượng tử ngược tương ứng, ví dụ, bởi đơn vị lượng tử hoá ngược 210, có thể bao gồm việc nhân với kích thước bước lượng tử. Các phương án theo một số tiêu chuẩn, ví dụ, HEVC có thể được tạo cấu hình để sử dụng thông số lượng tử hoá để xác định kích thước bước lượng tử. Nói chung, kích thước bước lượng tử có thể được tính toán dựa trên thông số lượng tử hoá nhờ sử dụng phép xấp xỉ điểm cố định của phương trình bao gồm phép chia. Các hệ số thay đổi tỷ lệ bổ sung có thể được đưa vào cho việc lượng tử hoá và giải lượng tử để khôi phục chuẩn đích của khôi dư, mà có thể được cải biến vì sự thay đổi tỷ lệ được sử dụng trong phép xấp xỉ điểm cố định của phương trình dành cho kích thước bước lượng tử và thông số lượng tử hoá. Theo một cách thức thực hiện ví dụ, thì việc thay đổi tỷ lệ của phép biến đổi ngược và phép giải lượng tử là có thể được kết hợp. Theo cách khác, các bảng lượng tử hoá được tùy biến có thể được sử dụng và được báo hiệu từ bộ mã hoá đến bộ giải mã, ví dụ, trong luồng bit. Việc lượng tử hoá là thao tác có tổn hao, trong đó sự tổn hao tăng lên khi tăng kích thước bước lượng tử.

Theo các phương án thì bộ mã hoá video 20 (tương ứng là đơn vị lượng tử hoá 208) có thể được tạo cấu hình để xuất ra các thông số lượng tử hoá (Quantization Parameter - QP), ví dụ, trực tiếp hoặc được mã hoá qua đơn vị mã hoá entropy 270, để,

ví dụ, bộ giải mã video 30 có thể nhận và áp dụng các thông số lượng tử hoá này để giải mã.

Lượng tử hoá ngược

Đơn vị lượng tử hoá ngược 210 được tạo cấu hình để áp dụng phép lượng tử hoá ngược của đơn vị lượng tử hoá 208 trên các hệ số được lượng tử hoá để thu được các hệ số được giải lượng tử 211, ví dụ, bằng cách áp dụng nghịch đảo của sơ đồ lượng tử hoá mà được áp dụng bởi đơn vị lượng tử hoá 208 dựa trên hoặc nhờ sử dụng kích thước bước lượng tử giống như đơn vị lượng tử hoá 208. Các hệ số được giải lượng tử 211 cũng có thể được gọi là các hệ số dư được giải lượng tử 211 và tương ứng với các hệ số biến đổi 207 - mặc dù thường không giống với các hệ số biến đổi do sự tổn hao bởi việc lượng tử hoá.

Biến đổi ngược

Đơn vị xử lý biến đổi ngược 212 được tạo cấu hình để áp dụng phép biến đổi ngược của phép biến đổi mà được áp dụng bởi đơn vị xử lý biến đổi 206, ví dụ, phép biến đổi cosin rời rạc (Discrete Cosine Transform - DCT) ngược hoặc phép biến đổi sin rời rạc (Discrete Sine Transform - DST) ngược hoặc các phép biến đổi ngược khác, để thu được khối dư được tái tạo 213 (hoặc các hệ số được giải lượng tử tương ứng 213) trong miền mẫu. Khối dư được tái tạo 213 cũng có thể được gọi là khối biến đổi 213.

Tái tạo

Đơn vị tái tạo 214 (ví dụ, bộ cộng hoặc bộ lấy tổng 214) được tạo cấu hình để cộng khối biến đổi 213 (tức là khối dư được tái tạo 213) vào khối dự đoán 265 để thu được khối được tái tạo 215 trong miền mẫu, ví dụ, bằng cách cộng – theo từng mẫu một - các giá trị mẫu của khối dư được tái tạo 213 và các giá trị mẫu của khối dự đoán 265.

Lọc

Đơn vị bộ lọc vòng 220 (hoặc ngắn gọn là “bộ lọc vòng” 220) được tạo cấu hình để lọc khối được tái tạo 215 để thu được khối được lọc 221, hoặc nói chung, là để lọc các mẫu được tái tạo để thu được các giá trị mẫu được lọc. Ví dụ, đơn vị bộ lọc vòng này được tạo cấu hình để làm nhẵn những sự chuyển tiếp điểm ảnh, hoặc theo cách khác là cải thiện chất lượng video. Đơn vị bộ lọc vòng 220 có thể bao gồm một hoặc nhiều bộ lọc vòng chẳng hạn như bộ lọc khử khối, bộ lọc dịch thích ứng mẫu (Sample-Adaptive Offset - SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc vòng thích ứng (Adaptive Loop Filter - ALF), bộ lọc khử nhiễu (Noise Suppression Filter - NSF), hoặc

tổ hợp bất kỳ của chúng. Theo một ví dụ, thì đơn vị bộ lọc vòng 220 có thể bao gồm bộ lọc khử khối, bộ lọc SAO và bộ lọc ALF. Thứ tự của tiến trình lọc có thể là bộ lọc khử khối, SAO và ALF. Theo ví dụ khác, tiến trình được gọi là ánh xạ độ chói có thay đổi tỷ lệ sắc độ (Luma Mapping with Chroma Scaling - LMCS) (tức là, bộ tạo hình lại trong vòng lặp thích ứng) là được bổ sung. Tiến trình này được thực hiện trước khi khử khối. Theo ví dụ khác, tiến trình lọc khử khối cũng có thể được áp dụng cho các mép khối con bên trong, ví dụ, các mép khối con afin, các mép khối con ATMVP, các mép biến đổi khối con (Sub-Block Transform - SBT) và các mép nội phân vùng con (Intra Sub-Partition - ISP). Tuy đơn vị bộ lọc vòng 220 được thể hiện trên Fig.2 là bộ lọc trong vòng lặp, nhưng theo các cấu hình khác, thì đơn vị bộ lọc vòng 220 có thể được thực hiện dưới dạng bộ lọc sau vòng lặp. Khối được lọc 221 cũng có thể được gọi là khối được tái tạo được lọc 221.

Theo các phương án thì bộ mã hoá video 20 (tương ứng là đơn vị bộ lọc vòng 220) có thể được tạo cấu hình để xuất ra các thông số bộ lọc vòng (chẳng hạn các thông số bộ lọc SAO hoặc các thông số bộ lọc ALF hoặc các thông số LMCS), ví dụ, trực tiếp hoặc được mã hoá qua đơn vị mã hoá entropy 270, để, ví dụ, bộ giải mã 30 có thể nhận và áp dụng các thông số bộ lọc vòng này hoặc các bộ lọc vòng tương ứng để giải mã.

Bộ đệm ảnh được giải mã

Bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230 có thể là bộ nhớ mà lưu giữ các ảnh tham chiếu, hoặc nói chung là dữ liệu ảnh tham chiếu, để mã hoá dữ liệu video bởi bộ mã hoá video 20. DPB 230 có thể được tạo ra bởi bất kỳ trong số các thiết bị nhớ khác nhau, chẳng hạn bộ nhớ truy cập ngẫu nhiên động (Dynamic Random Access Memory - DRAM), bao gồm DRAM đồng bộ (Synchronous DRAM - SDRAM), RAM từ điện trở (Magnetoresistive RAM - MRAM), RAM điện trở (Resistive RAM - RRAM), hoặc các loại thiết bị nhớ khác. Bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230 có thể được tạo cấu hình để lưu giữ một hoặc nhiều khối được lọc 221. Bộ đệm ảnh được giải mã 230 có thể còn được tạo cấu hình để lưu giữ các khối được lọc khác trước đó, ví dụ, các khối được tái tạo và được lọc 221 trước đó, của cùng ảnh hiện tại hoặc của các ảnh khác nhau, ví dụ, các ảnh được tái tạo trước đó, và có thể cung cấp các ảnh được tái tạo, tức là được giải mã, hoàn chỉnh trước đó (và các khối tham chiếu và các mẫu tương ứng) và/hoặc ảnh hiện tại được tái tạo một phần (và các khối tham chiếu và các mẫu tương ứng), ví dụ, để liên dự đoán. Bộ đệm

ảnh được giải mã (Decoded Picture Buffer - DPB) 230 có thể còn được tạo cấu hình để lưu giữ một hoặc nhiều khối được tái tạo chưa được lọc 215, hoặc nói chung là các mẫu được tái tạo chưa được lọc, ví dụ, nếu khối được tái tạo 215 không được lọc bởi đơn vị bộ lọc vòng 220, hoặc phiên bản được xử lý tiếp bất kỳ khác của các khối hoặc các mẫu được tái tạo.

Chọn chế độ (phân vùng & dự đoán)

Đơn vị chọn chế độ 260 bao gồm đơn vị phân vùng 262, đơn vị liên dự đoán 244 và đơn vị nội dự đoán 254, và được tạo cấu hình để nhận hoặc thu thập dữ liệu ảnh gốc, ví dụ, khối gốc 203 (khối hiện tại 203 của ảnh hiện tại 17), và dữ liệu ảnh được tái tạo, ví dụ, các mẫu hoặc các khối được tái tạo được lọc và/hoặc chưa được lọc của cùng ảnh (ảnh hiện tại) và/hoặc từ một hoặc nhiều ảnh được giải mã trước đó, ví dụ, từ bộ đệm ảnh được giải mã 230 hoặc các bộ đệm khác (ví dụ, bộ đệm hàng (không được thể hiện trên hình vẽ)). Dữ liệu ảnh được tái tạo này được sử dụng làm dữ liệu ảnh tham chiếu cho việc dự đoán, ví dụ, liên dự đoán hoặc nội dự đoán, để thu được khối dự đoán 265 hoặc bộ dự đoán 265.

Đơn vị chọn chế độ 260 có thể được tạo cấu hình để xác định hoặc chọn việc phân vùng cho chế độ dự đoán khối hiện tại (bao gồm việc không phân vùng) và chế độ dự đoán (ví dụ, chế độ nội dự đoán hoặc chế độ liên dự đoán) và tạo ra khối dự đoán 265 tương ứng, mà được sử dụng cho việc tính toán khối dư 205 và tái tạo khối được tái tạo 215.

Theo các phương án thì đơn vị chọn chế độ 260 có thể được tạo cấu hình để chọn việc phân vùng và chế độ dự đoán (ví dụ, từ các chế độ được hỗ trợ bởi hoặc khả dụng cho đơn vị chọn chế độ 260), mà cung cấp phần dư khớp nhất hay nói cách khác là phần dư nhỏ nhất (phần dư nhỏ nhất có nghĩa là nén tốt hơn để truyền hoặc lưu giữ), hoặc phí tổn báo hiệu nhỏ nhất (phí tổn báo hiệu nhỏ nhất có nghĩa là nén tốt hơn để truyền hoặc lưu giữ), hoặc chế độ mà tính đến hoặc cân bằng cả hai yếu tố này. Đơn vị chọn chế độ 260 có thể được tạo cấu hình để xác định việc phân vùng và chế độ dự đoán dựa trên sự tối ưu hoá tốc độ và độ méo (Rate Distortion Optimization - RDO), tức là chọn chế độ dự đoán mà có sự méo tốc độ nhỏ nhất. Các từ ngữ như “tốt nhất”, “nhỏ nhất”, “tối ưu”, v.v., trong ngữ cảnh này không nhất thiết phải là “tốt nhất”, “nhỏ nhất”, “tối ưu” về mặt tổng thể, v.v., mà cũng có thể là việc thoả mãn tiêu chuẩn chấm dứt hoặc tiêu chuẩn chọn như việc một giá trị lớn hơn hoặc nằm dưới ngưỡng hoặc các ràng buộc khác mà

có thể dẫn đến việc “chọn dưới mức tối ưu” nhưng giảm được sự phức tạp và thời gian xử lý.

Nói cách khác, đơn vị phân vùng 262 có thể được tạo cấu hình để phân vùng ảnh từ chuỗi video thành chuỗi đơn vị cây lập mã (Coding Tree Unit - CTU), và CTU 203 có thể còn được phân vùng thành các phân vùng khối nhỏ hơn hoặc các khối con (mà lại tạo thành các khối), ví dụ, nhờ sử dụng lặp đi lặp lại việc phân vùng cây tứ phân (Quad-Tree-partitioning - QT), phân vùng nhị phân (Binary-Tree partitioning - BT) hoặc phân vùng cây tam phân (Triple-Tree-partitioning - TT) hoặc tổ hợp bất kỳ của chúng, và để thực hiện, ví dụ, việc dự đoán đối với mỗi trong số các phân vùng khối hoặc các khối con này, trong đó việc chọn chế độ bao gồm việc chọn cấu trúc cây của khối 203 được phân vùng và các chế độ dự đoán là được áp dụng cho mỗi trong số các phân vùng khối hoặc các khối con này.

Trong phần sau đây, việc phân vùng (ví dụ, bởi đơn vị phân vùng 260) và tiến trình dự đoán (bởi đơn vị liên dự đoán 244 và đơn vị nội dự đoán 254), mà được thực hiện bởi bộ mã hoá video 20 ví dụ, sẽ được giải thích chi tiết hơn.

Phân vùng

Đơn vị phân vùng 262 có thể được tạo cấu hình để phân vùng ảnh từ chuỗi video thành chuỗi đơn vị cây lập mã (Coding Tree Unit - CTU), và đơn vị phân vùng 262 có thể phân vùng (hoặc chia) đơn vị cây lập mã (Coding Tree Unit - CTU) 203 thành các phân vùng nhỏ hơn, ví dụ, các khối nhỏ hơn có hình vuông hoặc hình chữ nhật. Đối với ảnh mà có ba mảng mẫu, thì CTU bao gồm khối $N \times N$ các mẫu độ chói cùng với hai khối tương ứng của các mẫu sắc độ. Kích thước lớn nhất được phép của khối độ chói trong CTU được quy định là 128×128 theo tiêu chuẩn lập mã video vạn năng (Versatile Video Coding - VVC) đang phát triển, nhưng nó có thể được quy định là giá trị không phải là 128×128 trong tương lai, ví dụ, 256×256 . Các CTU của ảnh có thể được tạo nhóm/được xếp nhóm dưới dạng các lát cắt/các nhóm ô, các ô hoặc các viên gạch. Ô thì bao phủ vùng hình chữ nhật của ảnh, và ô có thể được chia thành một hoặc nhiều viên gạch. Một viên gạch bao gồm một số lượng hàng CTU trong ô. Ô mà không được phân vùng thành các viên gạch thì có thể được gọi là viên gạch. Tuy nhiên, viên gạch là tập hợp con thật của ô và không được gọi là ô. Có hai chế độ của các nhóm ô được hỗ trợ trong VVC, tức là chế độ lát cắt/nhóm ô quét mảnh và chế độ lát cắt hình chữ nhật. Trong chế độ nhóm ô quét mảnh, thì lát cắt/nhóm ô chứa chuỗi ô trong quá trình quét

mảnh ô của ảnh. Trong chế độ lát cắt hình chữ nhật, thì lát cắt chứa một số lượng viên gạch của ảnh mà cùng tạo thành vùng hình chữ nhật của ảnh. Các viên gạch trong lát cắt hình chữ nhật là theo thứ tự quét mảnh viên gạch của lát cắt. Các khối nhỏ hơn này (mà cũng có thể được gọi là các khối con) có thể còn được phân vùng thành các phân vùng thậm chí nhỏ hơn nữa. Việc này cũng được gọi là phân vùng cây hoặc phân vùng cây phân cấp, trong đó khối gốc, ví dụ, tại mức cây gốc 0 (mức phân cấp 0, độ sâu 0), có thể được phân vùng theo cách đệ quy, ví dụ, được phân vùng thành hai hoặc nhiều khối có mức cây thấp hơn tiếp theo, ví dụ, các nút tại mức cây 1 (mức phân cấp 1, độ sâu 1), trong đó các khối này có thể lại được phân vùng thành hai hoặc nhiều khối có mức thấp hơn tiếp theo, ví dụ, mức cây 2 (mức phân cấp 2, độ sâu 2), v.v., cho đến khi việc phân vùng được chấm dứt, ví dụ, bởi vì tiêu chí chấm dứt được thoả mãn, ví dụ, đã đạt tới độ sâu cây lớn nhất hoặc kích thước khối nhỏ nhất. Các khối mà không được phân vùng tiếp thì còn được gọi là các khối lá hoặc các nút lá của cây. Cây mà sử dụng việc phân vùng thành hai phân vùng thì được gọi là cây nhị phân (Binary-Tree - BT), cây mà sử dụng việc phân vùng thành ba phân vùng thì được gọi là cây tam phân (Ternary-Tree - TT), và cây mà sử dụng việc phân vùng thành bốn phân vùng thì được gọi là cây tứ phân (Quad-Tree - QT).

Ví dụ, đơn vị cây lập mã (Coding Tree Unit - CTU) có thể là hoặc bao gồm CTB của các mẫu độ chói, hai CTB tương ứng của các mẫu sắc độ của ảnh mà có ba mảng mẫu, hoặc CTB của các mẫu của ảnh đơn sắc hoặc ảnh mà được lập mã nhờ sử dụng ba mặt phẳng màu riêng biệt và các cấu trúc cú pháp được dùng để lập mã các mẫu này. Tương ứng theo đó, khối cây lập mã (Coding Tree Block - CTB) có thể là khối $N \times N$ mẫu đối với một số giá trị của N sao cho việc chia thành phần thành các CTB là việc phân vùng. Đơn vị lập mã (Coding Unit - CU) có thể là hoặc bao gồm khối lập mã của các mẫu độ chói, hai khối lập mã tương ứng của các mẫu sắc độ của ảnh mà có ba mảng mẫu, hoặc khối lập mã của các mẫu của ảnh đơn sắc hoặc ảnh mà được lập mã nhờ sử dụng ba mặt phẳng màu riêng biệt và các cấu trúc cú pháp được dùng để lập mã các mẫu này. Tương ứng theo đó, khối lập mã (Coding Block - CB) có thể là khối $M \times N$ mẫu đối với một số giá trị của M và N sao cho việc chia CTB thành các khối lập mã là việc phân vùng.

Theo các phương án, ví dụ, theo HEVC, thì đơn vị cây lập mã (Coding Tree Unit - CTU) có thể được chia thành các CU nhờ sử dụng cấu trúc cây tứ phân được biểu thị

là cây lập mã. Việc quyết định xem lập mã vùng ảnh nhờ sử dụng việc dự đoán liên ảnh (thời gian) hay dự đoán nội ảnh (không gian) là được thực hiện tại mức độ CU lá. Mỗi CU lá có thể còn được chia thành một, hai hoặc bốn PU theo kiểu chia PU. Bên trong một PU, thì cùng một tiến trình dự đoán được áp dụng, và thông tin liên quan được truyền đến bộ giải mã trên cơ sở PU. Sau khi thu được khối dư bằng cách áp dụng tiến trình dự đoán dựa trên kiểu chia PU, thì CU lá có thể được phân vùng thành các đơn vị biến đổi (Transform Unit - TU) theo cấu trúc cây tứ phân khác tương tự như cây lập mã đối với CU.

Theo các phương án, ví dụ, theo tiêu chuẩn lập mã video mới nhất hiện đang được phát triển, mà được gọi là lập mã video vạn năng (Versatile Video Coding - VVC), thì cây đa kiểu được lồng cây tứ phân kết hợp sử dụng nhị phân và tam phân chia cấu trúc phân đoạn, ví dụ, được dùng để phân vùng đơn vị cây lập mã. Theo cấu trúc cây lập mã trong đơn vị cây lập mã, thì CU có thể có hình vuông hoặc hình chữ nhật. Ví dụ, đơn vị cây lập mã (Coding Tree Unit - CTU) được phân vùng trước hết bởi cây tứ phân. Sau đó, các nút lá của cây tứ phân này có thể được phân vùng tiếp bởi cấu trúc cây đa kiểu. Có bốn kiểu chia trong cấu trúc cây đa kiểu, là chia nhị phân dọc (SPLIT_BT_VER), chia nhị phân ngang (SPLIT_BT_HOR), chia tam phân dọc (SPLIT_TT_VER), và chia tam phân ngang (SPLIT_TT_HOR). Các nút lá của cây đa kiểu này được gọi là các đơn vị lập mã (Coding Unit - CU), và trừ khi CU là quá lớn đối với chiều dài biến đổi lớn nhất, thì sự phân đoạn này là được sử dụng cho việc xử lý dự đoán và biến đổi mà không cần phân vùng thêm. Điều này có nghĩa là, trong phần lớn các trường hợp, thì CU, PU và TU là có cùng kích thước khối ở cây tứ phân với cấu trúc khối lập mã cây đa kiểu được lồng. Sự ngoại lệ xuất hiện khi chiều dài biến đổi lớn nhất được hỗ trợ là nhỏ hơn chiều rộng hoặc chiều cao của thành phần màu của CU. VVC phát triển cơ chế báo hiệu độ nhất của thông tin chia phân vùng trong cây tứ phân với cấu trúc cây lập mã cây đa kiểu được lồng. Theo cơ chế báo hiệu này, thì đơn vị cây lập mã (Coding Tree Unit - CTU) là được coi như gốc của cây tứ phân và được phân vùng đầu tiên bởi cấu trúc cây tứ phân. Sau đó, mỗi nút lá của cây tứ phân (khi đủ lớn để cho phép nó) tiếp tục được phân vùng bởi cấu trúc cây đa kiểu. Trong cấu trúc cây đa kiểu, thì cờ thứ nhất (mtt_split_cu_flag) được báo hiệu để chỉ thị xem nút có được phân vùng tiếp hay không; khi nút được phân vùng tiếp, thì cờ thứ hai (mtt_split_cu_vertical_flag) được báo hiệu để chỉ thị chiều chia, và sau đó cờ thứ ba

(`mtt_split_cu_binary_flag`) được báo hiệu để chỉ thị xem việc chia là chia nhị phân hay chia tam phân. Dựa trên các giá trị của `mtt_split_cu_vertical_flag` và `mtt_split_cu_binary_flag`, thì chế độ chia cây đa kiểu (`MttSplitMode`) của CU có thể được trích xuất bởi bộ giải mã dựa trên quy tắc định trước hoặc bảng. Cần lưu ý, đối với thiết kế nhất định, ví dụ, thiết kế khối độ chói 64×64 và tạo đường ống sắc độ 32×32 ở các bộ giải mã phần cứng VVC, thì việc chia TT là bị cấm khi chiều rộng hoặc chiều cao của khối lập mã độ chói lớn hơn 64, như được thể hiện trên Fig.6. Việc chia TT cũng bị cấm khi chiều rộng hoặc chiều cao của khối lập mã sắc độ lớn hơn 32. Thiết kế tạo đường ống này sẽ chia ảnh thành các đơn vị dữ liệu đường ống ảo (Virtual Pipeline Data Unit - VPDU) mà được xác định dưới dạng các đơn vị không chồng nhau trong ảnh. Ở các bộ giải mã phần cứng, thì các VPDU liên tiếp được xử lý bởi nhiều tầng đường ống đồng thời. Kích thước VPDU là gần như tỷ lệ thuận với kích thước bộ đệm trong phần lớn các tầng đường ống, nên việc giữ cho kích thước VPDU nhỏ là điều quan trọng. Trong phần lớn các bộ giải mã phần cứng, thì kích thước VPDU có thể được thiết đặt bằng kích thước khối biến đổi (Transform Block - TB) lớn nhất. Tuy nhiên, trong VVC, thì việc phân vùng cây tam phân (Ternary Tree - TT) và cây nhị phân (Binary Tree - BT) có thể dẫn đến việc làm tăng kích thước VPDU.

Ngoài ra, cần lưu ý rằng, khi một phần của khối nút cây vượt quá đường biên ảnh đáy hoặc bên phải, thì khối nút cây đó bị bắt buộc phải được chia cho đến khi tất cả các mẫu của mỗi CU được lập mã được đặt bên trong các đường biên ảnh.

Theo một ví dụ, công cụ nội phân vùng con (Intra Sub-Partition - ISP) có thể chia các khối được nội dự đoán độ chói theo chiều dọc hoặc theo chiều ngang thành 2 hoặc 4 phân vùng con tùy theo kích thước khối.

Theo một ví dụ, thì đơn vị chọn chế độ 260 của bộ mã hoá video 20 có thể được tạo cấu hình để thực hiện tổ hợp bất kỳ của các kỹ thuật phân vùng được mô tả ở đây.

Như đã mô tả trên đây, bộ mã hoá video 20 được tạo cấu hình để xác định hoặc chọn chế độ dự đoán tốt nhất hoặc chế độ dự đoán tối ưu từ tập hợp các chế độ dự đoán (ví dụ, được định trước). Tập hợp chế độ dự đoán này có thể bao gồm, ví dụ, các chế độ nội dự đoán và/hoặc các chế độ liên dự đoán.

Nội dự đoán

Tập hợp chế độ nội dự đoán có thể bao gồm 35 chế độ nội dự đoán khác nhau, ví dụ, các chế độ không có hướng như chế độ DC (Direct Current - dòng điện một chiều)

(hay trung bình) và chế độ phẳng, hoặc các chế độ có hướng, ví dụ, như được xác định trong HEVC, hoặc có thể bao gồm 67 chế độ nội dự đoán khác nhau, ví dụ, các chế độ không có hướng như chế độ DC (hay trung bình) và chế độ phẳng, hoặc các chế độ có hướng, ví dụ, như được xác định cho VVC. Theo một ví dụ, thì một số chế độ nội dự đoán có góc thông thường là được thay thế theo cách thích ứng bằng các chế độ nội dự đoán góc rộng đối với các khối không vuông, ví dụ, như được xác định ở VVC. Theo ví dụ khác, để tránh các phép toán chia đối với việc dự đoán DC, thì chỉ có cạnh dài là được dùng để tính toán trung bình cho các khối không vuông. Và, các kết quả của việc nội dự đoán của chế độ phẳng có thể còn được cải biến bởi phương pháp kết hợp nội dự đoán phụ thuộc vị trí (Position Dependent intra Prediction Combination - PDPC).

Đơn vị nội dự đoán 254 được tạo cấu hình để sử dụng các mẫu được tái tạo của các khối lân cận của cùng ảnh hiện tại để tạo ra khối nội dự đoán 265 theo chế độ nội dự đoán của tập hợp các chế độ nội dự đoán.

Đơn vị nội dự đoán 254 (hoặc nói chung là đơn vị chọn chế độ 260) còn được tạo cấu hình để xuất các thông số nội dự đoán (hoặc nói chung là thông tin chỉ thị chế độ nội dự đoán được chọn cho khối) đến đơn vị mã hoá entropy 270 dưới dạng các phần tử cú pháp 266 để bao gồm vào dữ liệu ảnh được mã hoá 21, để, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số dự đoán này để giải mã.

Liên dự đoán (bao gồm dự đoán liên lớp)

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

Ngoài các chế độ dự đoán nêu trên, thì chế độ bỏ cóc, chế độ trực tiếp và/hoặc chế độ liên dự đoán khác cũng có thể được áp dụng.

Ví dụ, dự đoán hợp nhất mở rộng, danh sách ứng viên hợp nhất của chế độ đó là được xây dựng bằng cách bao gồm năm loại ứng viên sau đây theo thứ tự: MVP không gian từ các CU lân cận về không gian, MVP thời gian từ các CU cùng vị trí, MVP dựa trên lịch sử từ bảng FIFO, MVP trung bình từng đôi và các Zero MV. Và việc tinh chỉnh

vector chuyển động phía bộ giải mã (Decoder side Motion Vector Refinement - DMVR) dựa trên việc so khớp hai chiều có thể được áp dụng để tăng độ chính xác của các MV của chế độ hợp nhất. Chế độ hợp nhất có MVD (Merge mode with MVD - MMVD), mà đến từ chế độ hợp nhất với các hiệu vector chuyển động. Cờ MMVD được báo hiệu ngay sau khi gửi cờ bỏ cóc và cờ hợp nhất để quy định xem chế độ MMVD có được sử dụng cho CU hay không. Và sơ đồ phân giải vector chuyển động thích ứng (Adaptive Motion Vector Resolution - AMVR) mức độ CU có thể được áp dụng. AMVR cho phép MVD của CU được lập mã với độ chính xác khác nhau. Tùy theo chế độ dự đoán đối với CU hiện tại mà các MVD của CU hiện tại có thể được chọn thích ứng. Khi CU được lập mã trong chế độ hợp nhất, thì chế độ liên dự đoán/nội dự đoán kết hợp (Combined Inter/Intra Prediction - CIIP) có thể được áp dụng cho CU hiện tại. Việc tính trung bình có trọng số của các tín hiệu liên dự đoán và nội dự đoán được thực hiện để thu được sự dự đoán CIIP. Dự đoán có bù chuyển động affine, trường chuyển động affine của khối là được mô tả bởi thông tin chuyển động của các vector chuyển động hai điểm kiểm soát (4 thông số) hoặc ba điểm kiểm soát (6 thông số). Dự đoán vector chuyển động theo thời gian dựa trên khối con (Subblock-based Temporal Motion Vector Prediction - SbTMVP), mà tương tự như việc dự đoán vector chuyển động theo thời gian (Temporal Motion Vector Prediction - TMVP) trong HEVC, nhưng dự đoán các vector chuyển động của các CU con trong CU hiện tại. Luồng quang học hai chiều (Bi-Directional Optical Flow - BDOF), mà trước đó được gọi là BIO, là phiên bản đơn giản hơn mà yêu cầu việc tính toán ít hơn nhiều, đặc biệt là về số lượng phép nhân và kích thước của bộ nhân. Chế độ phân vùng hình tam giác, trong đó chế độ này, thì CU được chia đều thành hai phân vùng hình tam giác, nhờ sử dụng việc chia theo đường chéo hoặc chia phản đường chéo. Bên cạnh đó, chế độ lưỡng dự đoán được mở rộng vượt qua việc lấy trung bình đơn giản, để cho phép việc lấy trung bình có trọng số của hai tín hiệu dự đoán.

Đơn vị liên dự đoán 244 có thể bao gồm đơn vị ước lượng chuyển động (Motion Estimation - ME) và đơn vị bù chuyển động (Motion Compensation - MC) (cả hai đơn vị này đều không được thể hiện trên Fig.2). Đơn vị ước lượng chuyển động có thể được tạo cấu hình để nhận hoặc thu thập 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 tái tạo trước đó, ví dụ, các khối được tái tạo của một hoặc nhiều ảnh được giải mã trước đó 231 khác/khác nhau, 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, hoặc nói cách khác, là 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.

Bộ mã hoá 20 có thể, ví dụ, được tạo cấu hình để chọn khối tham chiếu từ các khối tham chiếu của cùng ảnh hoặc các ảnh khác nhau trong số các ảnh khác đó 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 dưới dạng các thông số liên dự đoán cho đơn vị ước lượng chuyển động. Độ dịch này còn được gọi vector chuyển động (Motion Vector - MV).

Đơn vị bù chuyển động được tạo cấu hình để thu thập, ví dụ, nhận, thông số liên dự đoán và thực hiện việc liên dự đoán dựa trên hoặc nhờ sử dụng thông số liên dự đoán này để thu được khối liên dự đoán 265. Việc bù chuyển động, mà được thực hiện bởi đơn vị bù chuyển động, là có thể bao gồm việc tìm nạp hoặc tạo ra khối dự đoán dựa trên vector chuyển động/vector khối được xác định bởi việc ước lượng chuyển động, có thể là thực hiện các phép nội suy đến độ chính xác là điểm ảnh con. Việc lọc nội suy có thể tạo ra các mẫu điểm ảnh bổ sung từ các mẫu điểm ảnh đã biết, do đó có thể tăng số lượng khối dự đoán ứng viên mà có thể được dùng để lập mã khối ảnh. Khi nhận được vector chuyển động dành cho PU của khối ảnh hiện tại, thì đơn vị bù chuyển động có thể định vị khối dự đoán mà vector chuyển động này trỏ đến đó trong một trong số các danh sách ảnh tham chiếu.

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

Lập mã entropy

Đơn vị mã hoá entropy 270 được tạo cấu hình để áp dụng, ví dụ, thuật toán hoặc sơ đồ mã hoá entropy (ví dụ, sơ đồ lập mã chiều dài biến thiên (Variable Length Coding - VLC), sơ đồ VLC thích ứng ngữ cảnh (Context Adaptive VLC - CAVLC), sơ đồ lập mã số học, nhị phân hoá, lập mã số học nhị phân thích ứng ngữ cảnh (Context Adaptive Binary Arithmetic Coding - CABAC), lập mã số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (Syntax-based context-adaptive Binary Arithmetic Coding - SBAC), lập

mã entropy phân vùng khoảng xác suất (Probability Interval Partitioning Entropy - PIPE) hoặc phương pháp hoặc kỹ thuật mã hoá entropy khác) hoặc vòng qua (không nén) trên các hệ số được lượng tử hoá 209, các thông số liên dự đoán, các thông số nội dự đoán, các thông số bộ lọc vòng và/hoặc các phần tử cú pháp khác để thu được dữ liệu ảnh được mã hoá 21 mà có thể được xuất ra qua đầu ra 272, ví dụ, dưới dạng luồng bit được mã hoá 21, để, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số này để giải mã. Luồng bit được mã hoá 21 có thể được truyền đến bộ giải mã video 30, hoặc được lưu giữ trong bộ nhớ để về sau truyền hoặc truy hồi bởi bộ giải mã video 30.

Các biến thể khác về cấu trúc của bộ mã hoá video 20 có thể được dùng để mã hoá luồng video. Ví dụ, bộ mã hoá dựa trên việc không biến đổi 20 có thể lượng tử hoá tín hiệu dư một cách trực tiếp mà không cần đơn vị xử lý biến đổi 206 đối với các khối hoặc các khung nhất định. Theo cách thức thực hiện khác, thì bộ mã hoá 20 có thể có đơn vị lượng tử hoá 208 và đơn vị lượng tử hoá ngược 210 được kết hợp thành một đơn vị.

Bộ giải mã và phương pháp giải mã

Fig.3 là hình vẽ thể hiện một ví dụ của bộ giải mã video 30 mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Bộ giải mã video 30 được tạo cấu hình để nhận dữ liệu ảnh được mã hoá 21 (ví dụ, luồng bit được mã hoá 21), ví dụ, được mã hoá bởi bộ mã hoá 20, để thu được ảnh được giải mã 331. Dữ liệu ảnh hoặc luồng bit được mã hoá này bao gồm thông tin để giải mã dữ liệu ảnh được mã hoá này, ví dụ, dữ liệu mà biểu thị các khối ảnh của lát cắt video được mã hoá (và/hoặc các nhóm ô hoặc các ô) và các phần tử cú pháp được liên kết.

Theo ví dụ trên Fig.3, thì bộ giải mã 30 bao gồm đơn vị giải mã entropy 304, đơn vị lượng tử hoá ngược 310, đơn vị xử lý biến đổi ngược 312, đơn vị tái tạo 314 (ví dụ, bộ lấy tổng 314), bộ lọc vòng 320, bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 330, đơn vị áp dụng chế độ 360, đơn vị liên dự đoán 344 và đơn vị nội dự đoán 354. Đơn vị liên dự đoán 344 có thể là hoặc bao gồm đơn vị bù chuyển động. Theo một số ví dụ, thì bộ giải mã video 30 có thể thực hiện chằng giải mã mà nói chung là nghịch đảo với chằng mã hoá được mô tả đối với bộ mã hoá video 100 từ Fig.2.

Như đã được mô tả đối với bộ mã hoá 20, thì đơn vị lượng tử hoá ngược 210, đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, bộ lọc vòng 220, bộ đệm ảnh được giải mã (Decoded Picture Buffer - DPB) 230, đơn vị liên dự đoán 344 và đơn vị nội dự

đoán 354 cũng được coi là tạo thành “bộ giải mã tích hợp” của bộ mã hoá video 20. Theo đó, đơn vị lượng tử hoá ngược 310 có thể giống về chức năng với đơn vị lượng tử hoá ngược 110, đơn vị xử lý biến đổi ngược 312 có thể giống về chức năng với đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 314 có thể giống về chức năng với đơn vị tái tạo 214, bộ lọc vòng 320 có thể giống về chức năng với bộ lọc vòng 220, và bộ đệm ảnh được giải mã 330 có thể giống về chức năng với bộ đệm ảnh được giải mã 230. Do đó, những phần giải thích được cung cấp cho các đơn vị và các chức năng tương ứng của bộ mã hoá video 20 cũng áp dụng theo cách tương ứng cho các đơn vị và các chức năng tương ứng của bộ giải mã video 30.

Giải mã entropy

Đơn vị giải mã entropy 304 được tạo cấu hình để phân tích luồng bit 21 (hoặc nói chung là dữ liệu ảnh được mã hoá 21) và thực hiện, ví dụ, việc giải mã entropy đối với dữ liệu ảnh được mã hoá 21 để thu được, ví dụ, các hệ số được lượng tử hoá 309 và/hoặc các thông số lập mã được giải mã (không được thể hiện trên Fig.3), ví dụ, bất kỳ hoặc tất cả trong số các thông số liên dự đoán (ví dụ, chỉ số ảnh tham chiếu và vector chuyển động), thông số nội dự đoán (ví dụ, chế độ nội dự đoán hoặc chỉ số), các thông số biến đổi, các thông số lượng tử hoá, các thông số bộ lọc vòng, và/hoặc các phần tử cú pháp khác. Đơn vị giải mã entropy 304 có thể được tạo cấu hình để áp dụng các thuật toán hoặc các sơ đồ giải mã tương ứng với các sơ đồ mã hoá như được mô tả đối với đơn vị mã hoá entropy 270 của bộ mã hoá 20. Đơn vị giải mã entropy 304 có thể còn được tạo cấu hình để cung cấp các thông số liên dự đoán, thông số nội dự đoán và/hoặc các phần tử cú pháp khác cho đơn vị áp dụng chế độ 360 và các thông số khác cho các đơn vị khác của bộ giải mã 30. Bộ giải mã video 30 có thể nhận các phần tử cú pháp này tại mức độ lát cắt video và/hoặc mức độ khối video. Ngoài ra hoặc thay vì các lát cắt và các phần tử cú pháp tương ứng, thì các nhóm ô và/hoặc các ô và các phần tử cú pháp tương ứng cũng có thể được nhận và/hoặc được sử dụng.

Lượng tử hoá ngược

Đơn vị lượng tử hoá ngược 310 có thể được tạo cấu hình để nhận các thông số lượng tử hoá (Quantization Parameter - QP) (hoặc nói chung là thông tin liên quan đến việc lượng tử hoá ngược) và các hệ số được lượng tử hoá từ dữ liệu ảnh được mã hoá 21 (ví dụ, bằng cách phân tích và/hoặc giải mã, ví dụ, bởi đơn vị giải mã entropy 304) và để áp dụng, dựa trên các thông số lượng tử hoá, việc lượng tử hoá ngược lên các hệ

số được lượng tử hoá 309 được giải mã, để thu được các hệ số được giải lượng tử 311, mà cũng có thể được gọi là các hệ số biến đổi 311. Tiến trình lượng tử hoá ngược có thể bao gồm việc sử dụng thông số lượng tử hoá xác định được bởi bộ mã hoá video 20 cho mỗi khối video trong lát cắt video (hoặc ô hoặc nhóm ô) để xác định độ lượng tử hoá và, tương tự, độ lượng tử hoá ngược mà cần được áp dụng.

Biến đổi ngược

Đơn vị xử lý biến đổi ngược 312 có thể được tạo cấu hình để nhận các hệ số được giải lượng tử 311, còn được gọi là các hệ số biến đổi 311, và để áp dụng phép biến đổi cho các hệ số được giải lượng tử 311 để thu được các khối dư được tái tạo 213 trong miền mẫu. Các khối dư được tái tạo 213 cũng có thể được gọi là các khối biến đổi 313. Phép biến đổi có thể là phép biến đổi ngược, ví dụ, DCT ngược, DST ngược, biến đổi nguyên ngược, hoặc tiến trình biến đổi ngược tương tự về mặt khái niệm. Đơn vị xử lý biến đổi ngược 312 có thể còn được tạo cấu hình để nhận các thông số biến đổi hoặc thông tin tương ứng từ dữ liệu ảnh được mã hoá 21 (ví dụ, bằng cách phân tích và/hoặc giải mã, ví dụ, bởi đơn vị giải mã entropy 304) để xác định phép biến đổi cần được áp dụng cho các hệ số được giải lượng tử 311.

Tái tạo

Đơn vị tái tạo 314 (ví dụ, bộ cộng hoặc bộ lấy tổng 314) có thể được tạo cấu hình để cộng khối dư được tái tạo 313 vào khối dự đoán 365 để thu được khối được tái tạo 315 trong miền mẫu, ví dụ, bằng cách cộng các giá trị mẫu của khối dư được tái tạo 313 và các giá trị mẫu của khối dự đoán 365.

Lọc

Đơn vị bộ lọc vòng 320 (trong vòng lặp mã hoặc sau vòng lặp mã) được tạo cấu hình để lọc khối được tái tạo 315 để thu được khối được lọc 321, ví dụ, để làm nhẵn sự chuyển tiếp điểm ảnh, hoặc theo cách khác là cải thiện chất lượng video. Đơn vị bộ lọc vòng 320 có thể bao gồm một hoặc nhiều bộ lọc vòng chẳng hạn như bộ lọc khử khối, bộ lọc dịch thích ứng mẫu (Sample-Adaptive Offset - SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ, bộ lọc vòng thích ứng (Adaptive Loop Filter - ALF), bộ lọc khử nhiễu (Noise Suppression Filter - NSF), hoặc tổ hợp bất kỳ của chúng. Theo một ví dụ, thì đơn vị bộ lọc vòng 220 có thể bao gồm bộ lọc khử khối, bộ lọc SAO và bộ lọc ALF. Thứ tự của tiến trình lọc có thể là bộ lọc khử khối, SAO và ALF. Theo ví dụ khác, tiến trình được gọi là ánh xạ độ chói có thay đổi tỷ lệ sắc độ (Luma Mapping with

Chroma Scaling - LMCS) (tức là, bộ tạo hình lại trong vòng lặp thích ứng) là được bổ sung. Tiến trình này được thực hiện trước khi khử khối. Theo ví dụ khác, tiến trình lọc khử khối cũng có thể được áp dụng cho các mép khối con bên trong, ví dụ, các mép khối con afin, các mép khối con ATMVP, các mép biến đổi khối con (Sub-Block Transform - SBT) và các mép nội phân vùng con (Intra Sub-Partition - ISP). Tuy đơn vị bộ lọc vòng 320 được thể hiện trên Fig.3 là bộ lọc trong vòng lặp, nhưng theo các cấu hình khác, thì đơn vị bộ lọc vòng 320 có thể được thực hiện dưới dạng bộ lọc sau vòng lặp.

Bộ đệm ảnh được giải mã

Sau đó, các khối video được giải mã 321 của ảnh được lưu giữ trong bộ đệm ảnh được giải mã 330, vốn lưu giữ các ảnh được giải mã 331 dưới dạng các ảnh tham chiếu để bù chuyển động sau đó cho các ảnh khác và/hoặc để xuất ra hoặc hiển thị.

Bộ giải mã 30 được tạo cấu hình để xuất ra ảnh được giải mã 311, ví dụ, thông qua đầu ra 312, để hiển thị hoặc để người dùng xem.

Dự đoán

Đơn vị liên dự đoán 344 có thể giống với đơn vị liên dự đoán 244 (cụ thể là giống với đơn vị bù chuyển động) và đơn vị nội dự đoán 354 có thể giống với đơn vị liên dự đoán 254 về mặt chức năng, và thực hiện quyết định chia hoặc quyết định phân vùng và việc dự đoán dựa trên các thông số phân vùng và/hoặc các thông số dự đoán hoặc thông tin tương ứng nhận được từ dữ liệu ảnh được mã hoá 21 (ví dụ, bằng cách phân tích và/hoặc giải mã, ví dụ, bởi đơn vị giải mã entropy 304). Đơn vị áp dụng chế độ 360 có thể được tạo cấu hình để thực hiện việc dự đoán (nội dự đoán hoặc liên dự đoán) trên mỗi khối dựa trên các ảnh được tái tạo, các khối hoặc các mẫu tương ứng (được lọc hoặc chưa được lọc) để thu được khối dự đoán 365.

Khi lát cắt video được lập mã dưới dạng lát cắt được nội lập mã (I), thì đơn vị nội dự đoán 354 của đơn vị áp dụng chế độ 360 được tạo cấu hình để tạo ra khối dự đoán 365 cho khối ảnh của lát cắt video hiện tại dựa trên chế độ nội dự đoán được báo hiệu và dữ liệu từ các khối được giải mã trước đó của ảnh hiện tại. Khi ảnh video được lập mã dưới dạng lát cắt được liên lập mã (tức là, B, hoặc P), thì đơn vị liên dự đoán 344 (ví dụ, đơn vị bù chuyển động) của đơn vị áp dụng chế độ 360 được tạo cấu hình để tạo ra các khối dự đoán 365 cho khối video của lát cắt video hiện tại dựa trên các vector chuyển động và các phần tử cú pháp khác nhận được từ đơn vị giải mã entropy

304. Đối với việc liên dự đoán, thì 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, là Danh sách 0 và Danh sách 1, nhờ sử dụng các kỹ thuật xây dựng mặc định dựa trên các ảnh tham chiếu được lưu giữ trong DPB 330. Nguyên lý giống hoặc tương tự như vậy có thể được áp dụng cho hoặc bởi các phương án sử dụng các nhóm ô (ví dụ, các nhóm ô video) và/hoặc các ô (ví dụ, các ô video) ngoài hoặc thay vì các lát cắt (ví dụ, các lát cắt video), ví dụ, video có thể được lập mã nhờ sử dụng các nhóm ô và/hoặc các ô I, P hoặc B.

Đơn vị áp dụng chế độ 360 được tạo cấu hình để xác định thông tin dự đoán cho khối video của lát cắt video hiện tại bằng cách phân tích các vectơ chuyển động hoặc thông tin liên quan và các phần tử cú pháp khác, và sử dụng thông tin dự đoán này để tạo ra các khối dự đoán cho khối video hiện tại mà đang được giải mã. Ví dụ, đơn vị áp dụng chế độ 360 sử dụng một số trong số các phần tử cú pháp nhận được để xác định chế độ dự đoán (ví dụ, nội dự đoán hoặc liên dự đoán) được dùng để lập mã các khối video của lát cắt video, loại lát cắt liên dự đoán (ví dụ, lát cắt B, lát cắt P, hoặc lát cắt GPB), thông tin xây dựng dành cho một hoặc nhiều trong số các danh sách ảnh tham chiếu dành cho lát cắt, các vectơ chuyển động dành cho mỗi khối video được liên mã hoá của lát cắt, trạng thái liên dự đoán dành cho mỗi khối video được liên lập mã của lát cắt, và thông tin khác để giải mã các khối video trong lát cắt video hiện tại. Nguyên lý giống hoặc tương tự như vậy có thể được áp dụng cho hoặc bởi các phương án sử dụng các nhóm ô (ví dụ, các nhóm ô video) và/hoặc các ô (ví dụ, các ô video) ngoài hoặc thay vì các lát cắt (ví dụ, các lát cắt video), ví dụ, video có thể được lập mã nhờ sử dụng các nhóm ô và/hoặc các ô I, P hoặc B.

Theo các phương án thì bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân vùng và/hoặc giải mã ảnh nhờ sử dụng các lát cắt (còn được gọi là các lát cắt video), trong đó ảnh có thể được phân vùng thành hoặc được giải mã nhờ sử dụng một hoặc nhiều lát cắt (thường là không chồng nhau), và mỗi lát cắt có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều nhóm khối (ví dụ, các ô (H.265/HEVC và VVC) hoặc các viên gạch (VVC)).

Theo các phương án thì bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân vùng và/hoặc giải mã ảnh nhờ sử dụng các lát cắt/các nhóm ô (còn được gọi là các nhóm ô video) và/hoặc các ô (còn được gọi là các ô video), trong

đó ảnh có thể được phân vùng thành hoặc được giải mã nhờ sử dụng một hoặc nhiều lát cắt/nhóm ô (thường là không chồng nhau), và mỗi lát cắt/nhóm ô có thể bao gồm, ví dụ, một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều ô, trong đó mỗi ô, ví dụ, có thể có hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ, các khối hoàn chỉnh hoặc các phân đoạn khối.

Các biến thể khác của bộ giải mã video 30 có thể được dùng để giải mã dữ liệu ảnh được mã hoá 21. Ví dụ, bộ giải mã 30 có thể tạo ra luồng video đầu ra mà không cần đơn vị lọc vòng 320. Ví dụ, bộ giải mã dựa trên việc không biến đổi 30 có thể lượng tử hoá ngược tín hiệu dư một cách trực tiếp mà không cần đơn vị xử lý biến đổi ngược 312 đối với các khối hoặc các khung nhất định. Theo cách thức thực hiện khác, thì bộ giải mã video 30 có thể có đơn vị lượng tử hoá ngược 310 và đơn vị xử lý biến đổi ngược 312 được kết hợp thành một đơn vị.

Cần hiểu rằng, ở bộ mã hoá 20 và bộ giải mã 30, thì kết quả xử lý của bước hiện tại có thể được xử lý tiếp và sau đó được xuất đến bước tiếp theo. Ví dụ, sau khi lọc nội suy, trích xuất vector chuyển động hoặc lọc vòng, thì hoạt động nữa, chẳng hạn xén hoặc dịch, có thể được thực hiện trên kết quả xử lý của việc lọc nội suy, trích xuất vector chuyển động hoặc lọc vòng này.

Cần lưu ý rằng các thao tác nữa có thể được áp dụng cho các vector chuyển động trích xuất được của khối hiện tại (bao gồm, nhưng không chỉ giới hạn ở các vector chuyển động của điểm kiểm soát của chế độ afin, các vector chuyển động của khối con trong chế độ afin, chế độ phẳng, chế độ ATMVP (Advanced Temporal Motion Vector Prediction - dự đoán vector chuyển động về thời gian nâng cao), các vector chuyển động về thời gian, v.v.). Ví dụ, giá trị của vector chuyển động được ràng buộc ở khoảng được định trước theo bit biểu diễn của nó. Nếu bit biểu diễn của vector chuyển động là bitDepth, thì khoảng là $-2^{(\text{bitDepth}-1)} \sim 2^{(\text{bitDepth}-1)}-1$, trong đó “^” có nghĩa là số mũ. Ví dụ, nếu bitDepth được đặt bằng 16, thì khoảng là -32768 ~ 32767; nếu bitDepth được đặt bằng 18, thì khoảng là -131072 ~ 131071. Ví dụ, giá trị của vector chuyển động trích xuất được (ví dụ, các MV có bốn khối con 4x4 trong một khối 8x8) là được ràng buộc sao cho hiệu số lớn nhất giữa các phần nguyên của các MV có bốn khối con 4x4 này không quá N điểm ảnh, chẳng hạn không quá 1 điểm ảnh. Sáng chế đề xuất hai phương pháp để ràng buộc vector chuyển động theo bitDepth.

Fig.4 là hình vẽ thể hiện sơ đồ của thiết bị lập mã video 400 theo một phương án của sáng chế. Thiết bị lập mã video 400 này phù hợp để thực hiện các phương án được bộc lộ như được mô tả ở đây. Theo một phương án, thì thiết bị lập mã video 400 có thể là bộ giải mã, chẳng hạn bộ giải mã video 30 trên Fig.1A, hoặc là bộ mã hoá, chẳng hạn bộ mã hoá video 20 trên Fig.1A.

Thiết bị lập mã video 400 bao gồm các cổng đầu vào 410 (hay các cổng vào 410) và các đơn vị bộ thu (Receiver - Rx) 420 để nhận dữ liệu; bộ xử lý, đơn vị logic, hoặc đơn vị xử lý trung tâm (Central Processing Unit - CPU) 430 để xử lý dữ liệu; các đơn vị bộ phát (Transmitter - Tx) 440 và các cổng đầu ra 450 (hay các cổng ra 450) để truyền dữ liệu; và bộ nhớ 460 để lưu giữ dữ liệu. Thiết bị lập mã video 400 có thể còn bao gồm các thành phần quang sang điện (Optical-to-Electrical - OE) và các thành phần điện sang quang (Electrical-to-Optical - EO) được ghép nối đến các cổng đầu vào 410, các đơn vị bộ thu 420, các đơn vị bộ phát 440, và các cổng đầu ra 450 để xuất ra hoặc nhập vào tín hiệu quang hoặc tín hiệu điện.

Bộ xử lý 430 được thực hiện bằng phần cứng và phần mềm. Bộ xử lý 430 có thể được thực hiện dưới dạng một hoặc nhiều chip CPU, lõi (ví dụ, dưới dạng bộ xử lý đa lõi), FPGA, ASIC, và DSP. Bộ xử lý 430 có giao tiếp với các cổng đầu vào 410, các đơn vị bộ thu 420, các đơn vị bộ phát 440, các cổng đầu ra 450, và bộ nhớ 460. Bộ xử lý 430 bao gồm môđun lập mã 470. Môđun lập mã 470 này thực hiện các phương án được bộc lộ đã mô tả trên đây. Ví dụ, môđun lập mã 470 thực hiện, xử lý, chuẩn bị, hoặc cung cấp các hoạt động lập mã khác nhau. Do đó, việc bao gồm môđun lập mã 470 đem lại sự cải thiện lớn đối với chức năng của thiết bị lập mã video 400 và thực hiện việc biến đổi thiết bị lập mã video 400 sang trạng thái khác. Theo cách khác, môđun lập mã 470 được thực hiện dưới dạng các chỉ dẫn được lưu giữ trong bộ nhớ 460 và được thực thi bởi bộ xử lý 430.

Bộ nhớ 460 có thể bao gồm một hoặc nhiều cơ cấu đĩa, băng, và ổ cứng thể rắn, và có thể được sử dụng làm thiết bị lưu trữ dữ liệu tràn, để lưu giữ các chương trình khi các chương trình đó được chọn để thực thi, và để lưu giữ các chỉ dẫn và dữ liệu mà được đọc trong quá trình thực thi chương trình. Bộ nhớ 460 có thể là, ví dụ, khả biến và/hoặc bất biến và có thể là bộ nhớ chỉ đọc (Read-Only Memory - ROM), bộ nhớ truy cập ngẫu nhiên (Random Access Memory - RAM), bộ nhớ đánh địa chỉ được nội dung bậc ba

(Ternary Content-Addressable Memory - TCAM), và/hoặc bộ nhớ truy cập ngẫu nhiên tĩnh (Static Random-Access Memory - SRAM).

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

Bộ xử lý 502 ở thiết bị 500 có thể là đơn vị xử lý trung tâm. Theo cách khác, bộ xử lý 502 có thể là loại thiết bị bất kỳ khác, hoặc nhiều thiết bị, có khả năng vận dụng hoặc xử lý thông tin hiện có hoặc được phát triển sau này. Tuy những cách thức thực hiện được bộc lộ có thể được thực hiện với một bộ xử lý đơn như được thể hiện, ví dụ, bộ xử lý 502, nhưng có thể đạt được các ưu điểm về tốc độ và hiệu quả nhờ sử dụng nhiều hơn một bộ xử lý.

Bộ nhớ 504 ở thiết bị 500 có thể là thiết bị nhớ chỉ đọc (Read Only Memory - ROM) hoặc thiết bị nhớ truy cập ngẫu nhiên (Random Access Memory - RAM) theo một cách thức thực hiện. Loại thiết bị lưu trữ phù hợp bất kỳ khác có thể được sử dụng làm bộ nhớ 504. Bộ nhớ 504 có thể bao gồm mã và dữ liệu 506 mà được truy cập bởi bộ xử lý 502 nhờ sử dụng buýt 512. Bộ nhớ 504 có thể còn bao gồm hệ điều hành 508 và các chương trình ứng dụng 510, các chương trình ứng dụng 510 này 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 1 đến N, mà còn bao gồm ứng dụng lập mã 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 thiết bị hiển thị 518. Theo một ví dụ, thì thiết bị hiển thị 518 có thể là thiết bị hiển thị nhạy tiếp xúc mà kết hợp thiết bị hiển thị với phần tử nhạy tiếp xúc mà hoạt động được để cảm biến những sự chạm đầu vào. Thiết bị hiển thị 518 có thể được ghép nối đến bộ xử lý 502 qua buýt 512.

Tuy được thể hiện ở đây dưới dạng buýt đơn, nhưng buýt 512 của thiết bị 500 có thể bao gồm nhiều buýt. Ngoài ra, bộ lưu trữ thứ cấp 514 có thể được ghép nối trực tiếp đến các thành phần còn lại của thiết bị 500 hoặc có thể được truy cập qua mạng và có thể bao gồm một đơn vị tích hợp, chẳng hạn thẻ nhớ, hoặc nhiều đơn vị, chẳng hạn nhiều thẻ nhớ. Do đó, thiết bị 500 có thể được thực hiện theo nhiều cấu hình khác nhau.

Lập mã khả biến

Lập mã khả biến bao gồm sự khả biến chất lượng (khả biến PSNR), khả biến

không gian, v.v.. Ví dụ, như được thể hiện trên Fig.6, chuỗi có thể được lấy mẫu giảm thành phiên bản có độ phân giải không gian thấp. Cả phiên bản có độ phân giải không gian thấp và phiên bản có độ phân giải không gian gốc (độ phân giải không gian cao) sẽ được mã hoá. Và nói chung, độ phân giải không gian thấp sẽ được lập mã trước, và nó sẽ được sử dụng làm tham chiếu cho độ phân giải không gian cao được lập mã sau.

Để mô tả thông tin của các lớp (số, sự phụ thuộc, sự xuất ra), thì có VPS (Video Parameter Set - tập hợp thông số video) được xác định như sau:

video_parameter_set_rbsp() {	Bộ mô tả
...	
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i])	
for(j = 0; j < i; j++)	
vps_direct_dependency_flag[i][j]	u(1)
}	
}	
...	
}	

vps_max_layers_minus1 cộng 1 quy định số lượng lớp lớn nhất được phép trong mỗi CVS mà tham chiếu đến VPS.

vps_all_independent_layers_flag bằng 1 quy định rằng tất cả các lớp trong CVS là được lập mã độc lập mà không sử dụng việc dự đoán liên lớp.

vps_all_independent_layers_flag bằng 0 quy định rằng một hoặc nhiều trong số các lớp trong CVS có thể sử dụng việc dự đoán liên lớp. Khi không có mặt, thì giá trị của vps_all_independent_layers_flag được suy ra là bằng 1. Khi vps_all_independent_layers_flag bằng 1, thì giá trị của vps_independent_layer_flag[i] được suy ra là bằng 1. Khi vps_all_independent_layers_flag bằng 0, thì giá trị của vps_independent_layer_flag[0] được suy ra là bằng 1.

vps_layer_id[i] quy định giá trị vps_layer_id của lớp thứ i. Đối với hai giá trị

nguyên không âm m và n bất kỳ, khi m nhỏ hơn n , thì giá trị của $vps_layer_id[m]$ sẽ nhỏ hơn $vps_layer_id[n]$.

$vps_independent_layer_flag[i]$ bằng 1 quy định rằng lớp có chỉ số i không sử dụng việc dự đoán liên lớp. $vps_independent_layer_flag[i]$ bằng 0 quy định rằng lớp có chỉ số i có thể sử dụng việc dự đoán liên lớp và $vps_layer_dependency_flag[i]$ là có mặt trong VPS.

$vps_direct_dependency_flag[i][j]$ bằng 0 quy định rằng lớp có chỉ số j không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i . $vps_direct_dependency_flag[i][j]$ bằng 1 quy định rằng lớp có chỉ số j là lớp tham chiếu trực tiếp cho lớp có chỉ số i . Khi $vps_direct_dependency_flag[i][j]$ không có mặt đối với i và j nằm trong khoảng từ 0 đến $vps_max_layers_minus1$, không loại trừ, thì nó được suy ra là bằng 0.

Biến số $DirectDependentLayerIdx[i][j]$, mà quy định lớp phụ thuộc trực tiếp thứ j của lớp thứ i , là được trích xuất như sau:

```
for( i = 1; i < vps_max_layers_minus1; i-- )
    if( !vps_independent_layer_flag[i] )
        for( j = i, k = 0; j >= 0; j-- )
            if( vps_direct_dependency_flag[i][j] )
                DirectDependentLayerIdx[i][k++] = j
```

(7-2)

Biến số $GeneralLayerIdx[i]$, mà quy định chỉ số lớp của lớp có nuh_layer_id bằng $vps_layer_id[i]$, là được trích xuất như sau:

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

(7-3)

Sự mô tả đơn giản là như sau:

$vps_max_layers_minus1$ cộng 1 có nghĩa là số lượng lớp.

$vps_all_independent_layers_flag$ chỉ thị xem tất cả các lớp có được lập mã độc lập hay không.

$vps_layer_id[i]$ chỉ thị ID lớp của lớp thứ i .

$vps_independent_layer_flag[i]$ chỉ thị xem lớp thứ i có được lập mã độc lập hay không.

$vps_direct_dependency_flag[i][j]$ chỉ thị xem lớp thứ j có được sử dụng làm

tham chiếu cho lớp thứ i hay không.

Quản lý DPB và đánh dấu ảnh tham chiếu.

Để quản lý các ảnh tham chiếu đó trong tiến trình giải mã, thì các ảnh được giải mã cần được giữ trong bộ đệm ảnh giải mã (Decoding Picture Buffer - DPB), để sử dụng làm tham chiếu cho việc giải mã ảnh sau đó. Để chỉ thị các ảnh đó, thì thông tin số đếm thứ tự ảnh (Picture Order Count - POC) của chúng cần được báo hiệu trong mào đầu lát cắt một cách trực tiếp hoặc gián tiếp. Nói chung, có hai danh sách ảnh tham chiếu, list0 và list1. Và, chỉ số ảnh tham chiếu cũng cần được bao gồm để báo hiệu ảnh trong danh sách. Đối với việc đơn dự đoán, thì các ảnh tham chiếu được tìm nạp từ một danh sách ảnh tham chiếu, đối với việc lưỡng dự đoán, thì các ảnh tham chiếu được tìm nạp từ hai danh sách ảnh tham chiếu.

Tất cả các ảnh tham chiếu được lưu giữ trong DPB. Tất cả các ảnh trong DPB được đánh dấu là “được sử dụng làm tham chiếu dài hạn”, “được sử dụng làm tham chiếu ngắn hạn”, hoặc “không được sử dụng làm tham chiếu”, và chỉ một cho ba trạng thái. Khi ảnh được đánh dấu là “không được sử dụng làm tham chiếu”, thì nó sẽ không được sử dụng làm tham chiếu nữa. Nếu nó cũng không cần được lưu giữ để xuất ra, thì nó có thể được loại bỏ khỏi DPB. Trạng thái của các ảnh tham chiếu có thể được báo hiệu trong mào đầu lát cắt, hoặc có thể được trích xuất từ thông tin mào đầu lát cắt.

Phương pháp quản lý ảnh tham chiếu mới đã được đề xuất, được gọi là phương pháp RPL (Reference Picture List - danh sách ảnh tham chiếu). RPL sẽ đề xuất toàn bộ tập hợp hoặc các tập hợp ảnh tham chiếu cho ảnh lập mã hiện tại, ảnh tham chiếu trong tập hợp ảnh tham chiếu được sử dụng cho việc giải mã ảnh hiện tại hoặc việc giải mã ảnh trong tương lai (về sau, hoặc sau đó). Do đó, RPL phản ánh thông tin các ảnh trong DPB, ngay cả khi ảnh tham chiếu không được sử dụng làm tham chiếu cho ảnh hiện tại, nếu nó sẽ được sử dụng làm tham chiếu cho ảnh sau đó, thì nó cần được lưu giữ trong RPL.

Sau khi ảnh được tái tạo, thì nó sẽ được lưu giữ trong DPB, và được đánh dấu là “được sử dụng làm tham chiếu ngắn hạn” theo mặc định. Hoạt động quản lý DPB sẽ bắt đầu sau khi phân tích thông tin RPL trong mào đầu lát cắt.

Xây dựng danh sách ảnh tham chiếu.

Thông tin ảnh tham chiếu có thể được báo hiệu qua mào đầu lát cắt. Ngoài ra, có thể có một số ứng viên RPL trong tập hợp thông số chuỗi (Sequence Parameters Set - SPS),

trong trường hợp này, mào đầu lát cắt có thể bao gồm chỉ số RPL để lấy thông tin RPL được cần thiết, mà không báo hiệu toàn bộ cấu trúc cú pháp RPL. Hoặc, toàn bộ cấu trúc cú pháp RPL có thể được báo hiệu trong mào đầu lát cắt.

Giới thiệu phương pháp RPL.

Để tiết kiệm chi phí các bit của báo hiệu RPL, thì có thể có một số ứng viên RPL trong SPS. Ảnh có thể sử dụng chỉ số RPL (`ref_pic_list_idx[ i ]`) để lấy thông tin RPL của nó từ SPS. Các ứng viên RPL được báo hiệu như sau:

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

Ngữ nghĩa là như sau:

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

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

`num_ref_pic_lists_in_sps[ i ]` quy định số lượng cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )` với `listIdx` bằng `i` được bao gồm trong SPS. Giá trị của `num_ref_pic_lists_in_sps[ i ]` sẽ nằm trong khoảng từ 0 đến 64, không loại trừ.

Bên cạnh việc lấy thông tin RPL dựa trên chỉ số RPL từ SPS, thì thông tin RPL có thể được báo hiệu trong mào đầu lát cắt.

slice_header() {	Bộ mô tả
...	
if((nal_unit_type != IDR_W_RADL && nal_unit_type != IDR_N_LP) sps_idr_rpl_present_flag) {	
for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && !pps_ref_pic_list_sps_idc[i] && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_sps_flag[i]	u(1)
if(ref_pic_list_sps_flag[i]) {	
if(num_ref_pic_lists_in_sps[i] > 1 && (i == 0 (i == 1 && rpl1_idx_present_flag)))	
ref_pic_list_idx[i]	u(v)
} else	
ref_pic_list_struct(i, num_ref_pic_lists_in_sps[i])	
for(j = 0; j < NumLtrpEntries[i][RplIdx[i]]; j++) {	
if(ltrp_in_slice_header_flag[i][RplIdx[i]])	
slice_poc_lsb_lt[i][j]	u(v)
delta_poc_msb_present_flag[i][j]	u(1)
if(delta_poc_msb_present_flag[i][j])	
delta_poc_msb_cycle_lt[i][j]	ue(v)
}	
}	
...	

ref_pic_list_sps_flag[i] bằng 1 quy định rằng danh sách ảnh tham chiếu i của lát cắt hiện tại là được trích xuất dựa trên một trong số các cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx) với listIdx bằng i trong SPS. ref_pic_list_sps_flag[i] bằng 0 quy định rằng danh sách ảnh tham chiếu i của lát cắt hiện tại là được trích xuất dựa trên cấu trúc cú pháp ref_pic_list_struct(listIdx, rplIdx) với listIdx bằng i mà được bao gồm trực tiếp trong các mào đầu lát cắt của ảnh hiện tại. Khi ref_pic_list_sps_flag[i] không có mặt, thì áp dụng như sau:

- Nếu num_ref_pic_lists_in_sps[i] bằng 0, thì giá trị của ref_pic_list_sps_flag[i] được suy ra là bằng 0.

- Nếu không ($\text{num_ref_pic_lists_in_sps}[i]$ lớn hơn 0), nếu $\text{rpl1_idx_present_flag}$ bằng 0, thì giá trị của $\text{ref_pic_list_sps_flag}[1]$ được suy ra là bằng $\text{ref_pic_list_sps_flag}[0]$.
- Còn nếu khác, thì giá trị của $\text{ref_pic_list_sps_flag}[i]$ được suy ra là bằng $\text{pps_ref_pic_list_sps_idx}[i] - 1$.

$\text{ref_pic_list_idx}[i]$ quy định chỉ số, vào danh sách các cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$ với listIdx bằng i được bao gồm trong SPS, của cấu trúc cú pháp $\text{ref_pic_list_struct}(\text{listIdx}, \text{rplsIdx})$ với listIdx bằng i mà được sử dụng cho việc trích xuất danh sách ảnh tham chiếu i của ảnh hiện tại. Phần tử cú pháp $\text{ref_pic_list_idx}[i]$ là được biểu diễn bởi $\text{Ceil}(\text{Log2}(\text{num_ref_pic_lists_in_sps}[i]))$ bit. Khi không có mặt, thì giá trị của $\text{ref_pic_list_idx}[i]$ được suy ra là bằng 0. Giá trị của $\text{ref_pic_list_idx}[i]$ sẽ nằm trong khoảng từ 0 đến $\text{num_ref_pic_lists_in_sps}[i] - 1$, không loại trừ. Khi $\text{ref_pic_list_sps_flag}[i]$ bằng 1 và $\text{num_ref_pic_lists_in_sps}[i]$ bằng 1, thì giá trị của $\text{ref_pic_list_idx}[i]$ được suy ra là bằng 0. Khi $\text{ref_pic_list_sps_flag}[i]$ bằng 1 và $\text{rpl1_idx_present_flag}$ bằng 0, thì giá trị của $\text{ref_pic_list_idx}[1]$ được suy ra là bằng $\text{ref_pic_list_idx}[0]$.

Biến số $\text{RplsIdx}[i]$ được trích xuất như sau:

$$\begin{aligned} \text{RplsIdx}[i] &= \text{ref_pic_list_sps_flag}[i] ? \text{ref_pic_list_idx}[i] : \\ &\quad \text{num_ref_pic_lists_in_sps}[i] \end{aligned} \quad (7-95)$$

$\text{slice_poc_lsb_lt}[i][j]$ quy định giá trị của môđulo số đếm thứ tự ảnh MaxPicOrderCntLsb của mục nhập LTRP thứ j trong danh sách ảnh tham chiếu thứ i . Chiều dài của phần tử cú pháp $\text{slice_poc_lsb_lt}[i][j]$ là $\text{log2_max_pic_order_cnt_lsb_minus4} + 4$ bit.

Biến số $\text{PocLsbLt}[i][j]$ được trích xuất như sau:

$$\begin{aligned} \text{PocLsbLt}[i][j] &= \text{ltrp_in_slice_header_flag}[i][\text{RplsIdx}[i]] ? \\ &\quad \text{slice_poc_lsb_lt}[i][j] : \text{rpls_poc_lsb_lt}[\text{listIdx}][\text{RplsIdx}[i]][j] \end{aligned} \quad (7-96)$$

$\text{delta_poc_msb_present_flag}[i][j]$ bằng 1 quy định rằng $\text{delta_poc_msb_cycle_lt}[i][j]$ là có mặt. $\text{delta_poc_msb_present_flag}[i][j]$ bằng 0 quy định rằng $\text{delta_poc_msb_cycle_lt}[i][j]$ là không có mặt.

Gọi prevTid0Pic là ảnh trước đó theo thứ tự giải mã mà có nuh_layer_id giống với ảnh hiện tại, có TemporalId bằng 0, và không phải là ảnh RASL hoặc RADL. Gọi setOfPrevPocVals là tập hợp bao gồm các thành phần sau đây:

- PicOrderCntVal của prevTid0Pic,
- PicOrderCntVal của mỗi ảnh mà được tham chiếu đến bởi các mục nhập trong RefPicList[0] hoặc RefPicList[1] của prevTid0Pic và có nuh_layer_id giống với ảnh hiện tại,
- PicOrderCntVal của mỗi ảnh mà đi sau prevTid0Pic theo thứ tự giải mã, có nuh_layer_id giống với ảnh hiện tại, và đi trước ảnh hiện tại theo thứ tự giải mã.

Khi có nhiều hơn một giá trị trong setOfPrevPocVals mà đối với chúng giá trị môđulo MaxPicOrderCntLsb là bằng PocLsbLt[i][j], thì giá trị của delta_poc_msb_present_flag[i][j] sẽ bằng 1.

delta_poc_msb_cycle_lt[i][j] quy định giá trị của biến số FullPocLt[i][j] như sau:

if(j == 0)

DeltaPocMsbCycleLt[i][j] = delta_poc_msb_cycle_lt[i][j]

else

(7-97)

DeltaPocMsbCycleLt[i][j] = delta_poc_msb_cycle_lt[i][j] +

DeltaPocMsbCycleLt[i][j - 1]

FullPocLt[i][j] = PicOrderCntVal - DeltaPocMsbCycleLt[i][j] *

MaxPicOrderCntLsb -

(PicOrderCntVal & (MaxPicOrderCntLsb - 1)) + PocLsbLt[i][j]

Giá trị của delta_poc_msb_cycle_lt[i][j] sẽ nằm trong khoảng từ 0 đến $2^{(32 - \log_2 \text{max_pic_order_cnt_lsb_minus4} - 4)}$, không loại trừ. Khi không có mặt, thì giá trị của delta_poc_msb_cycle_lt[i][j] được suy ra là bằng 0.

Cấu trúc cú pháp của RPL là như sau:

ref_pic_list_struct(listIdx, rplsIdx) {	Bộ 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 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)
}	
}	

num_ref_entries[listIdx][rplsIdx] quy định số lượng mục nhập trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx). Giá trị của num_ref_entries[listIdx][rplsIdx] sẽ nằm trong khoảng từ 0 đến sps_max_dec_pic_buffering_minus1 + 14, không loại trừ.

ltrp_in_slice_header_flag[listIdx][rplsIdx] bằng 0 quy định rằng các POC LSB của các mục nhập LTRP trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là có mặt trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx).

ltrp_in_slice_header_flag[listIdx][rplsIdx] bằng 1 quy định rằng các POC LSB của các mục nhập LTRP trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là không có mặt trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx).

inter_layer_ref_pic_flag[listIdx][rplsIdx][i] bằng 1 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là mục nhập ILRP.

inter_layer_ref_pic_flag[listIdx][rplsIdx][i] bằng 0 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) không phải là mục nhập ILRP. Khi không có mặt, thì giá trị của inter_layer_ref_pic_flag[listIdx][rplsIdx][i] được suy ra là bằng 0.

st_ref_pic_flag[listIdx][rplsIdx][i] bằng 1 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là mục nhập STRP. st_ref_pic_flag[listIdx][rplsIdx][i] bằng 0 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là mục nhập LTRP. Khi inter_layer_ref_pic_flag[listIdx][rplsIdx][i] bằng 0 và st_ref_pic_flag[listIdx][rplsIdx][i] không có mặt, thì giá trị của st_ref_pic_flag[listIdx][rplsIdx][i] được suy ra là bằng 1.

Biến số NumLtrpEntries[listIdx][rplsIdx] được trích xuất 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 ]++
```

(7-120)

abs_delta_poc_st[listIdx][rplsIdx][i] quy định giá trị của biến số 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
```

(7-121)

Giá trị của abs_delta_poc_st[listIdx][rplsIdx][i] sẽ nằm trong khoảng từ 0 đến $2^{15} - 1$, không loại trừ.

strp_entry_sign_flag[listIdx][rplsIdx][i] bằng 1 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là có giá trị lớn hơn hoặc bằng 0. strp_entry_sign_flag[listIdx][rplsIdx][i] bằng 0 quy định rằng mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) là có giá trị nhỏ hơn 0. Khi không có mặt, thì giá trị của strp_entry_sign_flag[listIdx][rplsIdx][i] được suy ra là bằng 1.

Danh sách DeltaPocValSt[listIdx][rplsIdx] được trích xuất như sau:

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

(7-122)

```

st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ] = (
    strp_entry_sign_flag[ listIdx ][ rplsIdx ][ i ] ) ?
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ] : 0 -
    AbsDeltaPocSt[ listIdx ][ rplsIdx ][ i ]

```

`rpls_poc_lsb_lt[ listIdx ][ rplsIdx ][ i ]` quy định giá trị của môđulo số đếm thứ tự ảnh `MaxPicOrderCntLsb` của ảnh được tham chiếu đến bởi mục nhập thứ `i` trong cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`. Chiều dài của phần tử cú pháp `rpls_poc_lsb_lt[ listIdx ][ rplsIdx ][ i ]` là `log2_max_pic_order_cnt_lsb_minus4 + 4` bit.

Một số phần mô tả tổng quát của cấu trúc RPL.

Đối với mỗi danh sách, thì có cấu trúc RPL. Đầu tiên, `num_ref_entries[ listIdx ][ rplsIdx ]` được báo hiệu để chỉ thị số lượng ảnh tham chiếu trong danh sách. `ltrp_in_slice_header_flag[ listIdx ][ rplsIdx ]` được dùng để chỉ thị xem thông tin LSB (Least Significant Bit - bit ít có nghĩa nhất) có được báo hiệu trong mào đầu lát cắt hay không. Nếu ảnh tham chiếu hiện tại không phải là ảnh tham chiếu liên lớp, thì `st_ref_pic_flag[ listIdx ][ rplsIdx ][ i ]` để chỉ thị xem nó có phải là ảnh tham chiếu dài hạn hay không. Nếu nó là ảnh tham chiếu ngắn hạn, thì thông tin POC (`abs_delta_poc_st` và `strp_entry_sign_flag`) được báo hiệu. Nếu `ltrp_in_slice_header_flag[ listIdx ][ rplsIdx ]` là không, thì `rpls_poc_lsb_lt[ listIdx ][ rplsIdx ][ j++ ]` được dùng để trích xuất thông tin LSB của ảnh tham chiếu hiện tại. MSB (Most Significant Bit - bit có nghĩa nhất) có thể được trích xuất trực tiếp, hoặc được trích xuất dựa trên thông tin (`delta_poc_msb_present_flag[ i ][ j ]` và `delta_poc_msb_cycle_lt[ i ][ j ]`) trong mào đầu lát cắt.

Tiến trình giải mã để xây dựng danh sách ảnh tham chiếu

Tiến trình này được gọi ra lúc bắt đầu tiến trình giải mã đối với mỗi lát cắt của ảnh không phải IDR.

Các ảnh tham chiếu được đánh địa chỉ thông qua các chỉ số tham chiếu. Chỉ số tham chiếu là chỉ số vào danh sách ảnh tham chiếu. Khi giải mã lát cắt I, thì không có danh sách ảnh tham chiếu nào được sử dụng trong quá trình giải mã dữ liệu lát cắt.

Khi giải mã lát cắt P, thì chỉ có danh sách ảnh tham chiếu 0 (tức là, RefPicList[0]), là được sử dụng trong quá trình giải mã dữ liệu lát cắt. Khi giải mã lát cắt B, thì cả danh sách ảnh tham chiếu 0 và danh sách ảnh tham chiếu 1 (tức là, RefPicList[1]) là được sử dụng trong quá trình giải mã dữ liệu lát cắt.

Lúc bắt đầu tiến trình giải mã đối với mỗi lát cắt của ảnh không phải IDR, thì các danh sách ảnh tham chiếu RefPicList[0] và RefPicList[1] là được trích xuất. Các danh sách ảnh tham chiếu này được sử dụng trong việc đánh dấu các ảnh tham chiếu như được quy định ở khoản 8.3.3 hoặc trong quá trình giải mã dữ liệu lát cắt.

GHI CHÚ 1 – Đối với lát cắt I của ảnh không phải IDR mà không phải là lát cắt đầu tiên của ảnh, thì RefPicList[0] và RefPicList[1] có thể được trích xuất cho mục đích kiểm tra sự hợp cách luồng bit, nhưng việc trích xuất của chúng là không cần thiết cho việc giải mã ảnh hiện tại hoặc các ảnh đi sau ảnh hiện tại theo thứ tự giải mã. Đối với lát cắt P mà không phải là lát cắt đầu tiên của ảnh, thì RefPicList[1] có thể được trích xuất cho mục đích kiểm tra sự hợp cách luồng bit, nhưng việc trích xuất của nó là không cần thiết cho việc giải mã ảnh hiện tại hoặc các ảnh đi sau ảnh hiện tại theo thứ tự giải mã.

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

```

for( i = 0; i < 2; i++ ) {
    for( j = 0, k = 0, pocBase = PicOrderCntVal; j < num_ref_entries[ i ][ RplsIdx[ i ] ];
        j++ ) {
        if( !inter_layer_ref_pic_flag[ i ][ RplsIdx[ i ] ][ j ] ) {
            if( st_ref_pic_flag[ i ][ RplsIdx[ i ] ][ j ] ) {
                RefPicPocList[ i ][ j ] = pocBase - DeltaPocValSt[ i ][ RplsIdx[ i ] ][ j ]
                if( there is a reference picture picA in the DPB with the same nuh_layer_id
                    as the current picture
                    and PicOrderCntVal equal to RefPicPocList[ i ][ j ] )
                    RefPicList[ i ][ j ] = picA
            }
            else
                RefPicList[ i ][ j ] = "no reference
picture"
            (8-5)
            pocBase = RefPicPocList[ i ][ j ]
        } else {
            if( !delta_poc_msb_cycle_lt[ i ][ k ] ) {
                if( there is a reference picA in the DPB with the same nuh_layer_id as
                    the current picture and
                    PicOrderCntVal & ( MaxPicOrderCntLsb - 1 ) equal to
                    PocLsbLt[ i ][ k ] )
                    RefPicList[ i ][ j ] = picA
            }
        }
    }
}

```

```

else
    RefPicList[ i ][ j ] = "no reference picture"
    RefPicLtPocList[ i ][ j ] = PocLsbLt[ i ][ k ]
} else {
    if( there is a reference picA in the DPB with the same nuh_layer_id as
the current picture and
        PicOrderCntVal equal to FullPocLt[ i ][ k ] )
        RefPicList[ i ][ j ] = picA
    else
        RefPicList[ i ][ j ] = "no reference picture"
        RefPicLtPocList[ i ][ j ] = FullPocLt[ i ][ k ]
    }
    k++
}
} else {
    layerIdx =
DirectDependentLayerIdx[ GeneralLayerIdx[ nuh_layer_id ] ][ ilrp_idc[ i ][ RplIdx ][ j
]]
    refPicLayerId = vps_layer_id[ layerIdx ]
    if( there is a reference picture picA in the DPB with nuh_layer_id equal to
refPicLayerId and
        the same PicOrderCntVal as the current picture )
        RefPicList[ i ][ j ] = picA
    else
        RefPicList[ i ][ j ] = "no reference picture"
    }
}
}
}

```

Sau khi các RPL được xây dựng, trong đó refPicLayerId là ID lớp của ILRP, PicOrderCntVal là giá trị POC của picA, thì tiến trình đánh dấu là như sau:

Tiến trình giải mã dành cho việc đánh dấu ảnh tham chiếu

Tiến trình này được gọi ra một lần trên mỗi ảnh, sau khi giải mã mào đầu lát cắt và tiến trình giải mã để xây dựng danh sách ảnh tham chiếu cho lát cắt như được quy định ở khoản 8.3.2, nhưng trước khi giải mã dữ liệu lát cắt. Tiến trình này có thể dẫn đến việc một hoặc nhiều ảnh tham chiếu trong DPB được đánh dấu là "không được sử dụng làm tham chiếu" hoặc "được sử dụng làm tham chiếu dài hạn".

Ảnh được giải mã trong DPB có thể được đánh dấu là "không được sử dụng làm tham chiếu", "được sử dụng làm tham chiếu ngắn hạn" hoặc "được sử dụng làm tham chiếu dài hạn", nhưng chỉ một trong ba trạng thái này tại thời điểm cụ thể bất kỳ trong quá trình hoạt động của tiến trình giải mã. Việc gán một trong số các dấu này cho ảnh sẽ loại bỏ một cách không tường minh dấu khác trong số các dấu này khi có thể áp dụng

được. Khi ảnh được coi là được đánh dấu dưới dạng "được sử dụng làm tham chiếu", thì điều này được gọi chung là ảnh được đánh dấu dưới dạng "được sử dụng làm tham chiếu ngắn hạn" hoặc "được sử dụng làm tham chiếu dài hạn" (nhưng không phải cả hai).

Các STRP và các ILRP được nhận dạng bởi các giá trị `nuh_layer_id` và `PicOrderCntVal` của chúng. Các LTRP được nhận dạng bởi các giá trị `nuh_layer_id` của chúng và $\text{Log}_2(\text{MaxLtPicOrderCntLsb})$ LSBs của các giá trị `PicOrderCntVal` của chúng.

Nếu ảnh hiện tại là ảnh CLVSS, thì tất cả các ảnh tham chiếu hiện tại trong DPB (nếu có) với `nuh_layer_id` giống như ảnh hiện tại là được đánh dấu là "không được sử dụng làm tham chiếu".

Nếu khác thì áp dụng như sau:

- Đối với mỗi mục nhập LTRP trong `RefPicList[ 0 ]` hoặc `RefPicList[ 1 ]`, khi ảnh được tham chiếu là STRP có cùng `nuh_layer_id` với ảnh hiện tại, thì ảnh đó được đánh dấu là "được sử dụng làm tham chiếu dài hạn".
- Mỗi ảnh tham chiếu có cùng `nuh_layer_id` với ảnh hiện tại trong DPB mà không được tham chiếu đến bởi mục nhập nào trong `RefPicList[ 0 ]` hoặc `RefPicList[ 1 ]` thì được đánh dấu là "không được sử dụng làm tham chiếu".
- Đối với mỗi mục nhập ILRP trong `RefPicList[ 0 ]` hoặc `RefPicList[ 1 ]`, thì ảnh được tham chiếu được đánh dấu là "được sử dụng làm tham chiếu dài hạn".

Ở đây lưu ý rằng, ILRP (Inter-Layer Reference Picture - ảnh tham chiếu liên lớp) là được đánh dấu là "được sử dụng làm tham chiếu dài hạn".

Có hai cú pháp trong SPS mà có liên quan đến thông tin tham chiếu liên lớp.

<code>seq_parameter_set_rbsp() {</code>	Bộ mô tả
...	
<code>sps_video_parameter_set_id</code>	u(4)
...	
<code>long_term_ref_pics_flag</code>	u(1)
<code>inter_layer_ref_pics_present_flag</code>	u(1)
<code>sps_idr_rpl_present_flag</code>	u(1)

rpl1_same_as_rpl0_flag	u(1)
...	

sps_video_parameter_set_id, khi lớn hơn 0, thì quy định giá trị của vps_video_parameter_set_id cho VPS được tham chiếu đến bởi SPS. Khi sps_video_parameter_set_id bằng 0, thì SPS không tham chiếu đến VPS và không có VPS nào được tham chiếu đến khi giải mã mỗi CVS tham chiếu đến SPS.

long_term_ref_pics_flag bằng 0 quy định rằng không có LTRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. long_term_ref_pics_flag bằng 1 quy định rằng các LTRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS.

inter_layer_ref_pics_present_flag bằng 0 quy định rằng không có ILRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. inter_layer_ref_pics_flag bằng 1 quy định rằng các ILRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS. Khi sps_video_parameter_set_id bằng 0, thì giá trị của inter_layer_ref_pics_present_flag được suy ra là bằng 0.

Sự mô tả đơn giản là như dưới đây:

long_term_ref_pics_flag được dùng để chỉ thị xem LTRP có thể được sử dụng trong tiến trình giải mã hay không.

inter_layer_ref_pics_present_flag được dùng để chỉ thị xem ILRP có thể được sử dụng trong tiến trình giải mã hay không.

Do đó, khi inter_layer_ref_pics_present_flag bằng 1, thì có thể có ILRP mà được sử dụng trong tiến trình giải mã, và nó được đánh dấu là "được sử dụng làm tham chiếu dài hạn". Trong trường hợp này, thì có LTRP được sử dụng trong tiến trình giải mã, ngay cả khi long_term_ref_pics_flag bằng 0. Do đó, có sự không nhất quán với ngữ nghĩa của long_term_ref_pics_flag.

Theo phương pháp hiện có, thì một số phần tử cú pháp dành cho thông tin tham chiếu liên lớp là luôn được báo hiệu, mà không tính đến chỉ số của lớp hiện tại. Sáng

chế đề xuất việc bổ sung một số điều kiện vào các phần tử cú pháp để cải thiện hiệu quả báo hiệu.

Vì `long_term_ref_pics_flag` chỉ được dùng để kiểm soát việc phân tích `ltrp_in_slice_header_flag` và `st_ref_pic_flag`, nên ngữ nghĩa được cải biến để kiểm soát việc phân tích các cờ phân tích trong RPL.

Các phần tử cú pháp dành cho thông tin tham chiếu liên lớp là được báo hiệu có tính đến chỉ số của lớp hiện tại. Nếu thông tin này có thể được trích xuất bởi chỉ số của lớp hiện tại, thì thông tin này không cần phải được báo hiệu.

Vì `long_term_ref_pics_flag` chỉ được dùng để kiểm soát việc phân tích `ltrp_in_slice_header_flag` và `st_ref_pic_flag`, nên ngữ nghĩa được cải biến để kiểm soát việc phân tích các cờ phân tích trong RPL.

Các phần tử cú pháp dành cho thông tin tham chiếu liên lớp là được báo hiệu có tính đến chỉ số của lớp hiện tại. Nếu thông tin này có thể được trích xuất bởi chỉ số của lớp hiện tại, thì thông tin này không cần phải được báo hiệu.

Phương án thứ nhất của sáng chế [ngữ nghĩa] (cải biến ngữ nghĩa của `long_term_ref_pics_flag` để loại bỏ sự không nhất quán của LTRP và ILRP.)

Vì `long_term_ref_pics_flag` chỉ được dùng để kiểm soát việc phân tích `ltrp_in_slice_header_flag` và `st_ref_pic_flag`, nên ngữ nghĩa được cải biến như sau:

`long_term_ref_pics_flag` bằng 1, quy định rằng `ltrp_in_slice_header_flag`, và `st_ref_pic_flag` là có mặt trong các cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`. `long_term_ref_pics_flag` bằng 0 quy định rằng các phần tử cú pháp này là không có mặt trong các cấu trúc cú pháp `ref_pic_list_struct( listIdx, rplsIdx )`.

Ngoài ra, ngữ nghĩa có thể được cải biến để loại trừ ILRP như sau:

`long_term_ref_pics_flag` bằng 0 quy định rằng không có LTRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. `long_term_ref_pics_flag` bằng 1 quy định rằng các LTRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS. Ở đây LTRP không bao gồm ILRP (Inter-Layer Reference Picture - ảnh tham chiếu liên lớp).

Phương án thứ hai của sáng chế [VPS]

Đề xuất 1: báo hiệu có điều kiện đối với `vps_direct_dependency_flag[ i ][ j ]` (thông tin tham chiếu liên lớp được báo hiệu có tính đến chỉ số của lớp hiện tại, để loại bỏ sự báo hiệu thông tin dư thừa, để cải thiện hiệu quả lập mã.)

Tuỳ chọn 1.A:

Ở đây lưu ý rằng, khi `i` bằng 1, thì điều này có nghĩa là `layer1` (lớp 1) cần tham chiếu đến lớp khác. Trong khi chỉ `layer0` có thể là lớp tham chiếu, nên `vps_direct_dependency_flag[ i ][ j ]` không cần phải được báo hiệu. Chỉ khi `i` lớn hơn 1, thì `vps_direct_dependency_flag[ i ][ j ]` mới cần được báo hiệu.

<code>video_parameter_set_rbsp() {</code>	Bộ mô tả
<code> vps_video_parameter_set_id</code>	u(4)
<code> vps_max_layers_minus1</code>	u(6)
<code> if(vps_max_layers_minus1 > 0)</code>	
<code> vps_all_independent_layers_flag</code>	u(1)
<code> for(i = 0; i <= vps_max_layers_minus1; i++) {</code>	
<code> vps_layer_id[i]</code>	u(6)
<code> if(i > 0 && !vps_all_independent_layers_flag) {</code>	
<code> vps_independent_layer_flag[i]</code>	u(1)
<code> if(i > 1 && !vps_independent_layer_flag[i])</code>	
<code> for(j = 0; j < i; j++)</code>	
<code> vps_direct_dependency_flag[i][j]</code>	u(1)
<code> }</code>	
<code> }</code>	
<code> if(vps_max_layers_minus1 > 0) {</code>	
<code> ...</code>	

`vps_direct_dependency_flag[ i ][ j ]` bằng 0 quy định rằng lớp có chỉ số `j` không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số `i`. `vps_direct_dependency_flag[ i ][ j ]` bằng 1 quy định rằng lớp có chỉ số `j` là lớp tham chiếu trực tiếp cho lớp có chỉ số `i`. Khi `vps_direct_dependency_flag[ i ][ j ]` không có mặt đối với `i` và `j` nằm trong khoảng từ 0 đến `vps_max_layers_minus1`, không loại trừ, nếu `i` bằng 1 và `vps_independent_layer_flag[ i ]` bằng 0, thì `vps_direct_dependency_flag[ i ][ j ]` được suy ra là bằng 1, nếu khác thì nó được suy ra là bằng 0.

Tuỳ chọn 1.B:

Bên cạnh phương pháp thực hiện nêu trên (Tuỳ chọn 1.A), thì có phương pháp khác là

Tuỳ chọn 1.B., điều này có nghĩa là, đối với i và j nằm trong khoảng từ 0 đến $i-1$, không loại trừ, và khi $\text{vps_independent_layer_flag}[i]$ bằng 0, và tất cả giá trị của $\text{vps_direct_dependency_flag}[i][j]$ bằng 0 đối với j nằm trong khoảng từ 0 đến $i-2$, không loại trừ, thì giá trị của $\text{vps_direct_dependency_flag}[i][i-1]$ không cần được báo hiệu, và nó được suy ra là bằng 1.

video_parameter_set_rbsp() {	Bộ mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i])	
for(j = 0, SumDependencyFlag=0; j < i; j++) {	
if(!(j == i-1 && SumDependencyFlag == 0))	
vps_direct_dependency_flag[i][j]	u(1)
SumDependencyFlag += vps_direct_dependency_flag[i][j]	
}	
}	
}	
if(vps_max_layers_minus1 > 0) {	
...	

$\text{vps_direct_dependency_flag}[i][j]$ bằng 0 quy định rằng lớp có chỉ số j không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i . $\text{vps_direct_dependency_flag}[i][j]$ bằng 1 quy định rằng lớp có chỉ số j là lớp tham chiếu trực tiếp cho lớp có chỉ số i . Khi $\text{vps_direct_dependency_flag}[i][j]$ không có mặt đối với i và j nằm trong khoảng từ 0 đến $\text{vps_max_layers_minus1}$, không loại trừ, nếu $\text{vps_independent_layer_flag}[i]$ bằng 0, và j bằng $i-1$ và giá trị của SumDependencyFlag bằng 0, thì $\text{vps_direct_dependency_flag}[i][j]$ được suy ra là bằng 1, nếu khác thì nó được suy ra là bằng 0.

Đề xuất 2: các ràng buộc về ngữ nghĩa của $\text{vps_direct_dependency_flag}[i][j]$

Cũng có thể bổ sung sự ràng buộc vào ngữ nghĩa của $vps_direct_dependency_flag[i][j]$, mà không làm thay đổi phương pháp báo hiệu cú pháp, hoặc bảng cú pháp. Về cơ bản, đối với i , nếu lớp có chỉ số i là lớp phụ thuộc ($vps_independent_layer_flag[i]$ bằng 0), thì ít nhất một giá trị của $vps_direct_dependency_flag[i][j]$, j nằm trong khoảng từ 0 đến $i-1$, là bằng 1. Theo cách khác, tổng của $vps_direct_dependency_flag[i][j]$, j nằm trong khoảng từ 0 đến $i-1$, là không được bằng 0. Hoặc, nên lớn hơn hoặc bằng 1. (ví dụ ≥ 1). Hoặc, nên lớn hơn 0. (ví dụ, >0)

Tuỳ chọn 2.A:

$vps_direct_dependency_flag[i][j]$ bằng 0 quy định rằng lớp có chỉ số j không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i . $vps_direct_dependency_flag[i][j]$ bằng 1 quy định rằng lớp có chỉ số j là lớp tham chiếu trực tiếp cho lớp có chỉ số i . Khi $vps_direct_dependency_flag[i][j]$ không có mặt đối với i và j nằm trong khoảng từ 0 đến $vps_max_layers_minus1$, không loại trừ, thì nó được suy ra là bằng 0. Ở đây, đối với i và j nằm trong khoảng từ 0 đến $i-1$, không loại trừ, và khi $vps_independent_layer_flag[i]$ bằng 0, thì tổng của $vps_direct_dependency_flag[i][j]$ nên lớn hơn 0.

Tuỳ chọn 2.B:

$vps_direct_dependency_flag[i][j]$ bằng 0 quy định rằng lớp có chỉ số j không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i . $vps_direct_dependency_flag[i][j]$ bằng 1 quy định rằng lớp có chỉ số j là lớp tham chiếu trực tiếp cho lớp có chỉ số i . Khi $vps_direct_dependency_flag[i][j]$ không có mặt đối với i và j nằm trong khoảng từ 0 đến $vps_max_layers_minus1$, không loại trừ, thì nó được suy ra là bằng 0. Ở đây, đối với i và j nằm trong khoảng từ 0 đến $i-1$, không loại trừ, và khi $vps_independent_layer_flag[i]$ bằng 0, thì ít nhất một giá trị của $vps_direct_dependency_flag[i][j]$ nên bằng 1.

Đề xuất 3: Đề xuất 1 + Đề xuất 2

Tuỳ chọn 3

Trên thực tế, Tuỳ chọn 1 và Tuỳ chọn 2 có thể kết hợp để là phương pháp thực hiện

khác.

Như Tuỳ chọn 1.B + Tuỳ chọn 2.B.

video_parameter_set_rbsp() {	Bộ mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(6)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i])	
for(j = 0, SumDependencyFlag=0; j < i; j++) {	
if(!(j == i-1 && SumDependencyFlag == 0))	
vps_direct_dependency_flag[i][j]	u(1)
SumDependencyFlag += vps_direct_dependency_flag[i][j]	
}	
}	
}	
if(vps_max_layers_minus1 > 0) {	
...	

vps_direct_dependency_flag[i][j] bằng 0 quy định rằng lớp có chỉ số j không phải là lớp tham chiếu trực tiếp cho lớp có chỉ số i. vps_direct_dependency_flag[i][j] bằng 1 quy định rằng lớp có chỉ số j là lớp tham chiếu trực tiếp cho lớp có chỉ số i. Khi vps_direct_dependency_flag[i][j] không có mặt đối với i và j nằm trong khoảng từ 0 đến vps_max_layers_minus1, không loại trừ, nếu vps_independent_layer_flag[i] bằng 0, và j bằng i-1 và giá trị của SumDependencyFlag bằng 0, thì vps_direct_dependency_flag[i][j] được suy ra là bằng 1, nếu khác thì nó được suy ra là bằng 0. Ở đây, đối với i và j nằm trong khoảng từ 0 đến i-1, không loại trừ, và khi vps_independent_layer_flag[i] bằng 0, thì ít nhất một giá trị của vps_direct_dependency_flag[i][j] nên bằng 1.

Phương pháp kết hợp không bị giới hạn ở đây, nó cũng có thể là

Như Tuỳ chọn 1.A + Tuỳ chọn 2.B.

Như Tuỳ chọn 1.A + Tuỳ chọn 2.A.

Như Tuỳ chọn 1.B + Tuỳ chọn 2.A.

Phương án thứ ba của sáng chế [sps] (thông tin tham chiếu liên lớp được báo hiệu có tính đến chỉ số của lớp hiện tại, để loại bỏ sự báo hiệu thông tin dư thừa, để cải thiện hiệu quả lập mã.)

Ở đây, lưu ý rằng nếu `sps_video_parameter_set_id` bằng 0, thì nó có nghĩa là không có nhiều lớp, nên không cần báo hiệu `inter_layer_ref_pics_flag`, và cờ này là 0 theo mặc định.

<code>seq_parameter_set_rbsp() {</code>	Bộ mô tả
...	
<code>sps_video_parameter_set_id</code>	u(4)
...	
<code>long_term_ref_pics_flag</code>	u(1)
<code>if(sps_video_parameter_set_id > 0)</code>	
<code>inter_layer_ref_pics_present_flag</code>	u(1)
<code>sps_idr_rpl_present_flag</code>	u(1)
<code>rpl1_same_as_rpl0_flag</code>	u(1)
...	

`inter_layer_ref_pics_present_flag` bằng 0 quy định rằng không có ILRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. `inter_layer_ref_pics_flag` bằng 1 quy định rằng các ILRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS. Khi `inter_layer_ref_pics_flag` không có mặt, thì nó được suy ra là bằng 0.

Ở đây, lưu ý rằng, khi `GeneralLayerIdx[ nuh_layer_id ]` bằng 0, thì lớp hiện tại là lớp thứ 0, nó không thể tham chiếu đến lớp nào khác. Do đó, không cần báo hiệu `inter_layer_ref_pics_present_flag`, và giá trị là 0 theo mặc định.

<code>seq_parameter_set_rbsp() {</code>	Bộ mô tả
...	
<code>sps_video_parameter_set_id</code>	u(4)

...	
long_term_ref_pics_flag	u(1)
if(GeneralLayerIdx[nuh_layer_id] > 0)	
inter_layer_ref_pics_present_flag	u(1)
sps_idr_rpl_present_flag	u(1)
rpl1_same_as_rpl0_flag	u(1)
...	

inter_layer_ref_pics_present_flag bằng 0 quy định rằng không có ILRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. inter_layer_ref_pics_flag bằng 1 quy định rằng các ILRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS. Khi inter_layer_ref_pics_flag không có mặt, thì nó được suy ra là bằng 0.

Lập mã cả hai trường hợp nêu trên, thì ví dụ ứng dụng khác được thể hiện dưới đây:

seq_parameter_set_rbsp() {	Bộ mô tả
...	
sps_video_parameter_set_id	u(4)
...	
long_term_ref_pics_flag	u(1)
if(sps_video_parameter_set_id > 0 && GeneralLayerIdx[nuh_layer_id] > 0)	
inter_layer_ref_pics_present_flag	u(1)
sps_idr_rpl_present_flag	u(1)
rpl1_same_as_rpl0_flag	u(1)
...	

inter_layer_ref_pics_present_flag bằng 0 quy định rằng không có ILRP nào được sử dụng cho việc liên dự đoán ảnh được lập mã nào trong CVS. inter_layer_ref_pics_flag bằng 1 quy định rằng các ILRP có thể được sử dụng cho việc liên dự đoán một hoặc nhiều ảnh được lập mã trong CVS. Khi inter_layer_ref_pics_flag không có mặt, thì nó được suy ra là bằng 0.

Phương án thứ tư của sáng chế [RPL]

Ở đây, lưu ý rằng, khi GeneralLayerIdx[nuh_layer_id] bằng 1, thì lớp hiện tại là layer1, và nó chỉ có thể tham chiếu đến layer0, trong khi ilrp_idc của layer0 phải là 0. Do đó, không cần báo hiệu ilrp_idc trong trường hợp này.

ref_pic_list_struct(listIdx, rplsIdx) {	Bộ 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 if(!ltrp_in_slice_header_flag[listIdx][rplsIdx])	
rpls_poc_lsb_lt[listIdx][rplsIdx][j++]	u(v)
} else if(GeneralLayerIdx[nuh_layer_id] > 1)	
ilrp_idc[listIdx][rplsIdx][i]	ue(v)
}	
}	
}	

ilrp_idc[listIdx][rplsIdx][i] quy định chỉ số, đến danh sách các lớp phụ thuộc trực tiếp, của ILRP của mục nhập thứ i trong cấu trúc cú pháp ref_pic_list_struct(listIdx, rplsIdx) đến danh sách lớp phụ thuộc trực tiếp. Giá trị của ilrp_idc[listIdx][rplsIdx][i] sẽ nằm trong khoảng từ 0 đến GeneralLayerIdx[nuh_layer_id] - 1, không loại trừ. Khi GeneralLayerIdx[nuh_layer_id] bằng 1, thì giá trị của ilrp_idc[listIdx][rplsIdx][i] được suy ra là bằng 0

Phương án thứ năm của sáng chế [kết hợp]

Ở đây, lưu ý rằng một phần của hoặc toàn bộ các phương án từ phương án 1 đến

phương án 4 là có thể được kết hợp để tạo ra phương án mới.

Ví dụ, phương án 1 + phương án 2 + phương án 3 + phương án 4, hoặc phương án 2 + phương án 3 + phương án 4, hoặc những sự kết hợp khác.

Sau đây là phần giải thích về các ứng dụng của phương pháp mã hoá cũng như phương pháp giải mã như được thể hiện ở các phương án nêu trên, và hệ thống sử dụng chúng.

Fig.7 là hình vẽ thể hiện sơ đồ khối của 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 ghi nhận 3102, thiết bị đầu cuối 3106, và tùy ý bao gồm thiết bị hiển thị 3126. Thiết bị ghi nhận 3102 truyền thông với thiết bị đầu cuối 3106 qua liên kết truyền thông 3104. Liên kết truyền thông này có thể bao gồm kênh truyền thông 13 được mô tả trên đây. Liên kết truyền thông 3104 bao gồm, nhưng không chỉ giới hạn ở, WiFi, Ethernet, cáp, không dây (3G (3rd Generation - thế hệ thứ ba)/4G (4th Generation - thế hệ thứ tư)/5G (5th Generation - thế hệ thứ năm)), USB (Universal Serial Bus - buýt nối tiếp vạn năng), hoặc loại kết hợp bất kỳ của chúng, v.v..

Thiết bị ghi nhận 3102 tạo ra dữ liệu, và có thể mã hoá dữ liệu này bằng phương pháp mã hoá như được thể hiện ở các phương án nêu trên. Theo cách khác, thiết bị ghi nhận 3102 có thể phân bố dữ liệu này đến máy chủ tạo luồng (không được thể hiện trên các hình vẽ), và máy chủ này mã hoá dữ liệu này và truyền dữ liệu được mã hoá đến thiết bị đầu cuối 3106. Thiết bị ghi nhận 3102 bao gồm, nhưng không chỉ giới hạn ở, camera, điện thoại thông minh hoặc máy tính bảng, máy tính hoặc máy tính xách tay, hệ thống hội nghị video, máy PDA (Personal Digital Assistant - máy trợ lý cá nhân kỹ thuật số), thiết bị được gắn trên xe, hoặc tổ hợp của bất kỳ trong số chúng, v.v.. Ví dụ, thiết bị ghi nhận 3102 có thể bao gồm thiết bị nguồn 12 như đã mô tả trên đây. Khi dữ liệu này bao gồm video, thì bộ mã hoá video 20 mà được bao gồm trong thiết bị ghi nhận 3102 có thể thực sự thực hiện việc xử lý mã hoá video. Khi dữ liệu này bao gồm audio (tức là, giọng nói), thì bộ mã hoá audio được bao gồm trong thiết bị ghi nhận 3102 có thể thực sự thực hiện việc xử lý mã hoá audio. Đối với một số tình huống thực tiễn, thì thiết bị ghi nhận 3102 phân phối dữ liệu video và audio được mã hoá bằng cách ghép kênh chúng cùng nhau. Đối với các tình huống thực tiễn khác, ví dụ, trong hệ thống hội nghị video, thì dữ liệu audio được mã hoá và dữ liệu video được mã hoá không được ghép kênh. Thiết bị ghi nhận 3102 phân phối dữ liệu audio được mã hoá và dữ liệu

video được mã hoá đến thiết bị đầu cuối 3106 một cách riêng biệt.

Ở hệ thống cung cấp nội dung 3100, thì thiết bị đầu cuối 310 nhận và tái tạo dữ liệu được mã hoá này. Thiết bị đầu cuối 3106 có thể là thiết bị có khả năng nhận và khôi phục dữ liệu, chẳng hạn điện thoại thông minh hoặc máy tính bảng 3108, máy tính hoặc máy tính xách tay 3110, đầu ghi video mạng (Network Video Recorder - NVR)/đầu ghi video kỹ thuật số (Digital Video Recorder - DVR) 3112, TV (TeleVision - máy thu hình) 3114, hộp thu tín hiệu truyền hình (Set Top Box - STB) 3116, hệ thống hội nghị video 3118, hệ thống giám sát video 3120, máy trợ lý cá nhân kỹ thuật số (Personal Digital Assistant - PDA) 3122, thiết bị được gắn trên xe 3124, hoặc tổ hợp của bất kỳ trong số chúng, v.v., mà có khả năng giải mã dữ liệu được mã hoá nêu trên. Ví dụ, thiết bị đầu cuối 3106 có thể bao gồm thiết bị đích 14 như đã mô tả trên đây. Khi dữ liệu được mã hoá này bao gồm video, thì bộ giải mã video 30 được bao gồm trong thiết bị đầu cuối này được ưu tiên thực hiện việc giải mã video. Khi dữ liệu được mã hoá này bao gồm audio, thì bộ giải mã audio được bao gồm trong thiết bị đầu cuối này được ưu tiên thực hiện việc xử lý giải mã audio.

Đối với thiết bị đầu cuối có thiết bị hiển thị của nó, ví dụ, điện thoại thông minh hoặc máy tính bảng 3108, máy tính hoặc máy tính xách tay 3110, đầu ghi video mạng (Network Video Recorder - NVR)/đầu ghi video kỹ thuật số (Digital Video Recorder - DVR) 3112, TV 3114, máy trợ lý cá nhân kỹ thuật số (Personal Digital Assistant - PDA) 3122, hoặc thiết bị được gắn trên xe 3124, thì thiết bị đầu cuối này có thể cấp dữ liệu được giải mã đến thiết bị hiển thị của nó. Đối với thiết bị đầu cuối không được trang bị thiết bị hiển thị, ví dụ, STB 3116, hệ thống hội nghị video 3118, hoặc hệ thống giám sát video 3120, thì thiết bị hiển thị ngoài 3126 được cắm vào đó để nhận và thể hiện dữ liệu được giải mã.

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

Fig.8 là hình vẽ thể hiện sơ đồ cấu trúc theo một ví dụ của thiết bị đầu cuối 3106. Sau khi thiết bị đầu cuối 3106 nhận được luồng từ thiết bị ghi nhận 3102, thì đơn vị xử lý giao thức 3202 phân tích giao thức truyền của luồng này. Giao thức này bao gồm, nhưng không chỉ giới hạn ở, giao thức tạo luồng thời gian thực (Real Time Streaming Protocol - RTSP), giao thức truyền siêu văn bản (Hyper Text Transfer Protocol - HTTP),

giao thức tạo luồng trực tiếp HTTP (HTTP Live Streaming protocol - HLS), MPEG-DASH (Dynamic Adaptive Streaming over HTTP - tạo luồng thích ứng động qua HTTP), giao thức vận chuyển thời gian thực (Real-time Transport Protocol - RTP), giao thức gửi thông điệp thời gian thực (Real Time Messaging Protocol - RTMP), hoặc bất kỳ loại kết hợp nào của chúng, v.v..

Sau khi đơn vị xử lý giao thức 3202 xử lý luồng này, thì tệp tin luồng được tạo ra. Tệp tin này được xuất đến đơn vị giải ghép kênh 3204. Đơn vị giải ghép kênh 3204 có thể phân tách dữ liệu được ghép kênh thành dữ liệu audio được mã hoá và dữ liệu video được mã hoá. Như đã mô tả trên đây, đối với một số tình huống thực tiễn, ví dụ, trong hệ thống hội nghị video, thì dữ liệu audio được mã hoá và dữ liệu video được mã hoá không được ghép kênh. Trong trường hợp này, thì dữ liệu được mã hoá được truyền đến bộ giải mã video 3206 và bộ giải mã audio 3208 mà không qua đơn vị giải ghép kênh 3204.

Thông qua việc xử lý giải ghép kênh, thì luồng cơ bản (Elementary Stream - ES) video, ES audio, và tùy ý là phụ đề, được tạo ra. Bộ giải mã video 3206, mà bao gồm bộ giải mã video 30 như đã được mô tả ở các phương án nêu trên, sẽ giải mã ES video này bằng phương pháp giải mã như được thể hiện ở các phương án nêu trên để tạo ra khung video, và cấp dữ liệu này đến đơn vị đồng bộ 3212. Bộ giải mã audio 3208 giải mã ES audio để tạo ra khung audio, và cấp dữ liệu này đến đơn vị đồng bộ 3212. Theo cách khác, khung video có thể được lưu giữ trong bộ đệm (không được thể hiện trên Fig. Y) trước khi nó được cấp đến đơn vị đồng bộ 3212. Tương tự, khung audio có thể được lưu giữ trong bộ đệm (không được thể hiện trên Fig. Y) trước khi nó được cấp đến đơn vị đồng bộ 3212.

Đơn vị đồng bộ 3212 đồng bộ khung video và khung audio này, và cung cấp video/audio này đến thiết bị hiển thị video/audio 3214. Ví dụ, đơn vị đồng bộ 3212 đồng bộ sự thể hiện của thông tin video và thông tin audio này. Thông tin có thể được lập mã trong cú pháp nhờ sử dụng các mốc thời gian về sự thể hiện của dữ liệu audio và dữ liệu thị giác được lập mã và các mốc thời gian về việc phân phối bản thân luồng dữ liệu này.

Nếu phụ đề được bao gồm trong luồng này, thì bộ giải mã phụ đề 3210 giải mã phụ đề này, và đồng bộ nó với khung video và khung audio, và cung cấp video/audio/phụ đề này đến thiết bị hiển thị video/audio/phụ đề 3216.

Sáng chế không chỉ giới hạn ở hệ thống nêu trên, và thiết bị mã hoá ảnh hoặc

thiết bị giải mã ảnh ở các phương án nêu trên có thể được kết hợp vào hệ thống khác, ví dụ, hệ thống trên xe ô tô.

Các toán tử toán học

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

Các toán tử số học

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

+	Cộng
-	Trừ (dưới dạng toán tử hai đối số) hoặc phủ định (dưới dạng toán tử tiền tố một ngôi)
*	Nhân, bao gồm nhân ma trận
x^y	Số mũ. Quy định x lũy thừa y . Trong các ngữ cảnh khác thì ký hiệu đó được dùng để đánh chỉ số trên chứ không được hiểu là số mũ.
/	Phép chia nguyên với việc cắt số hạng của kết quả về phía 0. Ví dụ, $7 / 4$ và $-7 / -4$ được cắt số hạng thành 1 và $-7 / 4$ và $7 / -4$ được cắt số hạng thành -1.
$\div$	Được dùng để biểu thị phép chia trong các phương trình toán học mà trong đó không có việc cắt số hạng hay làm tròn nào.
$\frac{x}{y}$	Được dùng để biểu thị phép chia trong các phương trình toán học mà trong đó không có việc cắt số hạng hay làm tròn nào.
$\sum_{i=x}^y f(i)$	Việc lấy tổng của $f(i)$ với i lấy tất cả các giá trị nguyên từ x đến và bao gồm y .
$x \% y$	Môđun. Số dư của x chia y , được xác định chỉ đối với các số nguyên x và y với $x \geq 0$ và $y > 0$.

Các toán tử logic

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

$x \&\& y$	Logic Boolean "và" của x và y
$x y$	Logic Boolean "hoặc" của x và y

! Logic Boolean "không"

$x ? y : z$ Nếu x là ĐÚNG hoặc không bằng 0, thì lấy giá trị của y ; còn nếu ngược lại, thì lấy giá trị của z .

Các toán tử quan hệ

Các toán tử quan hệ sau đây được xác định 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 cho phần tử cú pháp hoặc biến số mà đã được gán giá trị "na" (not applicable - không áp dụng được), thì giá trị "na" này được coi như giá trị khác biệt cho phần tử cú pháp hoặc biến số này. Giá trị "na" này được coi là không được bằng bất kỳ giá trị nào khác.

Các toán tử thao tác bit

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

- & Thao tác bit "và". Khi hoạt động trên các đối số nguyên, thì hoạt động trên sự biểu diễn bù hai của giá trị nguyên. Khi hoạt động trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, thì đối số ngắn hơn được mở rộng bằng cách cộng thêm các bit có nghĩa bằng 0.
- | Thao tác bit "hoặc". Khi hoạt động trên các đối số nguyên, thì hoạt động trên sự biểu diễn bù hai của giá trị nguyên. Khi hoạt động trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, thì đối số ngắn hơn được mở rộng bằng cách cộng thêm các bit có nghĩa bằng 0.
- ^ Thao tác bit "hoặc loại trừ". Khi hoạt động trên các đối số nguyên, thì hoạt động trên sự biểu diễn bù hai của giá trị nguyên. Khi hoạt động trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, thì đối số ngắn hơn được mở rộng bằng cách cộng thêm các bit có nghĩa bằng 0.
- $x \gg y$ Dịch phải số học đối với sự biểu diễn số nguyên bù hai của x một lượng y chữ số nhị phân. Hàm này chỉ được xác định cho các giá trị nguyên không âm của y . Các bit mà được dịch thành các bit có nghĩa nhất (Most Significant Bit - MSB) do việc dịch phải này là có giá trị bằng MSB của x trước phép toán dịch.
- $x \ll y$ Dịch trái số học đối với sự biểu diễn số nguyên bù hai của x một lượng y chữ số nhị phân. Hàm này chỉ được xác định cho các giá trị nguyên không âm của y . Các bit mà được dịch thành các bit ít có nghĩa nhất (Least Significant Bit - LSB) do việc dịch trái này là có giá trị bằng 0.

Các toán tử gán

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

- = Toán tử gán
- ++ Tăng, tức là, $x++$ là tương đương với $x = x + 1$; khi được sử dụng trong chỉ số mảng, thì lấy giá trị của biến số trước phép toán tăng.
- Giảm, tức là, $x--$ là tương đương với $x = x - 1$; khi được sử dụng trong chỉ số mảng, thì lấy giá trị của biến số trước phép toán giảm.
- += Tăng lượng được quy định, tức là, $x += 3$ là tương đương với $x = x + 3$, và $x += (-3)$ là tương đương với $x = x + (-3)$.
- Giảm lượng được quy định, tức là, $x -= 3$ là tương đương với $x = x - 3$, và $x -= (-3)$ là tương đương với $x = x - (-3)$.

Ký hiệu khoảng

Ký hiệu sau đây được dùng để quy định khoảng giá trị:

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

Các hàm toán học

Các hàm toán học sau đây được xác định:

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

$\text{Asin}(x)$ hàm sin lượng giác ngược, hoạt động trên đối số x mà nằm trong khoảng $-1,0$ đến $1,0$, không loại trừ, với giá trị đầu ra nằm trong khoảng $-\pi/2$ đến $\pi/2$, không loại trừ, với đơn vị là radian

$\text{Atan}(x)$ hàm tang lượng giác ngược, hoạt động trên đối số x , với giá trị đầu ra nằm trong khoảng $-\pi/2$ đến $\pi/2$, không loại trừ, với đơn vị là radian

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

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

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

$$\text{Clip1c}(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_c) - 1, x)$$

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

$\text{Cos}(x)$ hàm cosin lượng giác hoạt động trên đối số x với đơn vị là radian.

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

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

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

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

$\text{Log10}(x)$ 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 với đơn vị là 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 với đơn vị là radian

Thứ tự độ ưu tiên của phép toán

Khi thứ tự độ ưu tiên trong biểu thức không được chỉ thị một cách tường minh nhờ sử dụng dấu ngoặc đơn, thì các quy tắc sau đây áp dụng:

- Các phép toán có độ ưu tiên cao hơn thì được thực hiện trước bất kỳ phép toán nào

có độ ưu tiên thấp hơn.

- Các phép toán có cùng độ ưu tiên thì được thực hiện tuần tự từ trái sang phải.

Bảng dưới đây quy định độ ưu tiên của các phép toán từ cao nhất đến thấp nhất; vị trí cao hơn trong bảng chỉ thị độ ưu tiên cao hơn.

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

Bảng: Độ ưu tiên của phép toán từ cao nhất (trên 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 bằng văn bản đối với các phép toán logic

Trong phần văn bản này, thì câu lệnh của các phép toán logic sẽ được mô tả về mặt toán

học dưới dạng như sau:

nếu (điều kiện 0)

câu lệnh 0

còn nếu (điều kiện 1)

câu lệnh 1

...

nếu khác thì /* thông tin lưu ý về điều kiện còn lại */

câu lệnh n

có thể được mô tả theo cách sau đây:

... như sau / ... áp dụng như sau:

- Nếu điều kiện 0, thì câu lệnh 0
- Còn nếu điều kiện 1, thì câu lệnh 1
- ...
- Còn nếu khác (thông tin lưu ý về điều kiện còn lại), thì câu lệnh n

Mỗi câu lệnh "Nếu ... Nếu khác thì, nếu ... Nếu khác thì, ..." trong phần văn bản này là được đưa vào với "... như sau" hoặc "... áp dụng như sau" ngay sau đó là "Nếu ...". Điều kiện cuối cùng của "Nếu ... Nếu khác thì, nếu ... Nếu khác thì, ..." luôn là "Nếu khác thì, ...". Các câu lệnh "Nếu ... Nếu khác thì, nếu ... Nếu khác thì, ..." đan xen có thể được nhận dạng bằng cách so khớp "... như sau" hoặc "... áp dụng như sau" với "Nếu khác thì, ..." kết thúc.

Trong phần văn bản này, thì câu lệnh của các phép toán logic sẽ được mô tả về mặt toán học dưới dạng như sau:

nếu (điều kiện 0a && điều kiện 0b)

câu lệnh 0

còn nếu (điều kiện 1a || điều kiện 1b)

câu lệnh 1

...

else

câu lệnh n

có thể được mô tả theo cách sau đây:

... như sau / ... áp dụng như sau:

- Nếu tất cả trong số các điều kiện sau đây là đúng, thì câu lệnh 0:
 - điều kiện 0a
 - điều kiện 0b
- Còn nếu một hoặc nhiều trong số các điều kiện sau đây là đúng, thì câu lệnh 1:
 - điều kiện 1a
 - điều kiện 1b
- ...
- Nếu khác thì câu lệnh n

Trong phần văn bản này, thì câu lệnh của các phép toán logic sẽ được mô tả về mặt toán học dưới dạng như sau:

nếu (điều kiện 0)

câu lệnh 0

nếu (điều kiện 1)

câu lệnh 1

có thể được mô tả theo cách sau đây:

Khi điều kiện 0, thì câu lệnh 0

Khi điều kiện 1, thì câu lệnh 1

Tuy các phương án của sáng chế đã chủ yếu được mô tả dựa trên việc lập mã video, nhưng cần lưu ý rằng các phương án của hệ thống lập mã 10, bộ mã hoá 20 và bộ giải mã 30 (và tương ứng theo đó là hệ thống 10), và các phương án còn lại được mô tả ở đây, là cũng có thể được tạo cấu hình cho việc xử lý hoặc lập mã ảnh tĩnh, tức là xử lý hoặc lập mã ảnh riêng lẻ mà không phụ thuộc vào ảnh đi trước hoặc theo sau nào

như trong lập mã video. Nói chung là chỉ có các đơn vị liên dự đoán 244 (bộ mã hoá) và 344 (bộ giải mã) là có thể không khả dụng trong trường hợp việc lập mã xử lý ảnh bị giới hạn ở ảnh đơn 17. Tất cả các chức năng khác (còn được gọi là các công cụ hoặc các công nghệ) của bộ mã hoá video 20 và bộ giải mã video 30 đều có thể được sử dụng cho việc xử lý ảnh tĩnh, ví dụ, việc tính toán phần dư 204/304, biến đổi 206, lượng tử hoá 208, lượng tử hoá ngược 210/310, biến đổi (ngược) 212/312, phân vùng 262/362, nội dự đoán 254/354, và/hoặc lọc vòng 220, 320, và lập mã entropy 270 và giải mã entropy 304.

Các phương án, ví dụ, của bộ mã hoá 20 và bộ giải mã 30, và các chức năng được mô tả ở đây, ví dụ, dựa vào bộ mã hoá 20 và bộ giải mã 30, là có thể được thực hiện bằng phần cứng, phần mềm, phần sụn, hoặc tổ hợp bất kỳ của chúng. Nếu được thực hiện bằng phần mềm, thì các chức năng này có thể được lưu giữ trên phương tiện đọc được bằng máy tính hoặc được truyền qua phương tiện truyền thông dưới dạng một hoặc nhiều chỉ dẫn hoặc mã và được thực thi bởi đơn vị xử lý dựa trên phần cứng. Phương tiện đọc được bằng máy tính có thể bao gồm các phương tiện lưu trữ đọc được bằng máy tính, mà tương ứng với phương tiện hữu hình, chẳng hạn các phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông bao gồm phương tiện bất kỳ mà tạo thuận lợi cho việc truyền chương trình máy tính từ nơi này đến nơi khác, ví dụ, theo giao thức truyền thông. Theo cách này, thì phương tiện đọc được bằng máy tính nói chung là có thể tương ứng với (1) các phương tiện lưu trữ hữu hình đọc được bằng máy tính mà là phi nhất thời hoặc (2) môi trường truyền thông như tín hiệu hoặc sóng mang. Các phương tiện lưu trữ dữ liệu có thể là các phương tiện khả dụng bất kỳ mà có thể được truy cập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để truy hồi các chỉ dẫn, mã và/hoặc các cấu trúc dữ liệu để thực hiện các kỹ thuật được mô tả trong bản mô tả này. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được bằng máy tính.

Cụ thể là, sáng chế đề xuất phương pháp giải mã luồng bit video được lập mã được thực hiện ở bộ giải mã như được thể hiện trên Fig.9, phương pháp này bao gồm các bước: S901, thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất (tức là, `vps_independent_layer_flag[ i ]`) mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không. S902, thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai (tức là, `vps_direct_dependency_flag[ i ][ j ]`) liên

quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp. và S903, thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu.

Tương tự, sáng chế đề xuất phương pháp mã hoá luồng bit video bao gồm dữ liệu được lập mã để được thực hiện ở bộ mã hoá như được thể hiện trên Fig.10, phương pháp này bao gồm các bước: S1001, xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; S1003, mã hoá phần tử cú pháp vào luồng bit video được lập mã, trong đó phần tử cú pháp này quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; trong đó giá trị của phần tử cú pháp này quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp, trong trường hợp mà không có lớp nào trong số ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Fig.11 là hình vẽ thể hiện bộ giải mã 1100 được tạo cấu hình để giải mã luồng bit video bao gồm dữ liệu được lập mã đối với các ảnh. Bộ giải mã 1100 theo ví dụ được thể hiện này bao gồm: đơn vị thu thập 1110 được tạo cấu hình để thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất mà quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; đơn vị thu thập 1110 còn được tạo cấu hình để thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai quy định xem lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; trong đó ít nhất một phần tử cú pháp thứ hai trong số một hoặc nhiều phần tử cú pháp thứ hai này có giá trị quy định lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất được phép sử dụng việc dự đoán liên lớp; và đơn vị dự đoán 1120 được tạo cấu hình để thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu.

Trong đó đơn vị này có thể là môđun phần mềm để thực thi bởi các bộ xử lý, hoặc hệ mạch xử lý.

Trong đó đơn vị thu thập 1110 có thể là đơn vị giải mã entropy 304. Đơn vị dự đoán 1120 có thể là đơn vị liên dự đoán 344. Bộ giải mã 1100 có thể là thiết bị đích 14, bộ giải mã 30, thiết bị 500, bộ giải mã video 3206 hoặc thiết bị đầu cuối 3106.

Tương tự, sáng chế đề xuất bộ mã hoá 1200 được tạo cấu hình để mã hoá luồng bit video bao gồm dữ liệu được lập mã đối với các ảnh, như được thể hiện trên Fig.12. Bộ mã hoá 1200 này bao gồm: đơn vị xác định 1210 được tạo cấu hình để xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không; đơn vị mã hoá 1220 được tạo cấu hình để mã hoá phần tử cú pháp vào luồng bit video được lập mã, trong đó phần tử cú pháp này quy định xem lớp thứ nhất có sử dụng việc dự đoán liên lớp hay không; trong đó giá trị của phần tử cú pháp này quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp, trong trường hợp mà không có lớp nào trong số ít nhất một lớp thứ hai là lớp tham chiếu trực tiếp cho lớp thứ nhất.

Trong đó đơn vị này có thể là mô đun phần mềm để thực thi bởi các bộ xử lý, hoặc hệ mạch xử lý.

Trong đó đơn vị mã hoá thứ nhất 1210 và đơn vị mã hoá thứ hai 1220 có thể là đơn vị mã hoá entropy 270.

Đơn vị xác định có thể là đơn vị chọn chế độ 260. Bộ mã hoá 1200 có thể là thiết bị nguồn 12, bộ mã hoá 20, hoặc thiết bị 500.

Theo một ví dụ không nhằm mục đích giới hạn, thì các phương tiện lưu trữ đọc được bằng máy tính đó có thể bao gồm RAM, ROM, EEPROM (Electrically Erasable Programmable ROM - ROM lập trình được và xoá được bằng điện), đĩa CD-ROM hoặc thiết bị lưu trữ đĩa quang khác, bộ lưu trữ bằng cơ cấu đĩa từ, hoặc các thiết bị lưu trữ từ tính khác, bộ nhớ flash (bộ nhớ chớp nhoáng), hoặc phương tiện bất kỳ khác mà có thể được dùng để lưu giữ mã chương trình mong muốn dưới dạng các chỉ dẫn hoặc các cấu trúc dữ liệu và có thể được truy cập bởi máy tính. Ngoài ra, kết nối bất kỳ cũng được gọi một cách phù hợp bằng thuật ngữ là phương tiện đọc được bằng máy tính. Ví dụ, nếu các chỉ dẫn được truyền từ một trang web, máy chủ, hoặc nguồn ở xa khác, bằng cáp đồng trục, cáp quang, cáp dây xoắn, đường dây thuê bao số (Digital Subscriber Line - DSL), hoặc các công nghệ không dây như hồng ngoại, sóng vô tuyến, và sóng vi ba, thì cáp đồng trục, cáp quang, cáp dây xoắn, DSL, hoặc các công nghệ không dây như hồng ngoại, sóng vô tuyến, và sóng vi ba đó cũng được bao gồm trong định nghĩa phương tiện. Tuy nhiên, cần hiểu rằng các phương tiện lưu trữ đọc được bằng máy tính

và các phương tiện lưu trữ dữ liệu là không bao gồm các kết nối, các sóng mang, các tín hiệu, hoặc các phương tiện nhất thời khác, mà thay vào đó, chúng là các phương tiện lưu trữ hữu hình phi nhất thời. Cơ cấu đĩa và đĩa, như được sử dụng ở đây, là bao gồm đĩa compac (Compact Disc - CD), đĩa laze, đĩa quang, đĩa đa năng kỹ thuật số (Digital Versatile Disc - DVD), cơ cấu đĩa mềm và đĩa Blu-ray, trong đó các cơ cấu đĩa thì thường tái tạo dữ liệu về mặt từ tính, còn đĩa thì tái tạo dữ liệu về mặt quang học bằng laze. Tổ hợp các thành phần nêu trên cũng được bao gồm trong phạm vi các phương tiện đọc được bằng máy tính.

Các chỉ dẫn có thể được thực thi bởi một hoặc nhiều bộ xử lý, chẳng hạn một hoặc nhiều bộ xử lý tín hiệu số (Digital Signal Processor - DSP), bộ vi xử lý thông dụng, mạch tích hợp chuyên dụng (Application Specific Integrated Circuit - ASIC), mảng logic lập trình được dạng trường (Field Programmable Logic Array - FPGA), hoặc hệ mạch logic tích hợp hoặc rời rạc tương đương khác. Theo đó, thuật ngữ “bộ xử lý”, như được sử dụng ở đây, có thể là cấu trúc bất kỳ trong số các cấu trúc nêu trên hoặc là cấu trúc bất kỳ khác mà phù hợp để thực hiện các kỹ thuật được mô tả ở đây. Ngoài ra, theo một số khía cạnh, thì chức năng được mô tả ở đây là có thể được cung cấp trong các môđun phần cứng và/hoặc phần mềm dành riêng mà được tạo cấu hình để mã hoá và giải mã, hoặc được kết hợp trong codec tổng hợp. Các kỹ thuật đó cũng có thể được thực hiện hoàn toàn trong một hoặc nhiều mạch hoặc phần tử logic.

Các kỹ thuật trong bản mô tả này có thể được thực hiện trong nhiều loại dụng cụ hoặc thiết bị khác nhau, bao gồm thiết bị cầm tay không dây, mạch tích hợp (Integrated Circuit - IC) hoặc tập hợp IC (ví dụ, tập hợp chip). Các thành phần, môđun, hoặc đơn vị khác nhau được mô tả trong bản mô tả này để nhấn mạnh các khía cạnh về mặt chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật được bộc lộ, chứ không nhất thiết yêu cầu việc thực hiện bằng các đơn vị phần cứng khác nhau. Thay vào đó, như đã mô tả trên đây, các đơn vị khác nhau có thể được kết hợp trong đơn vị phần cứng codec hoặc được cung cấp bởi tập hợp các đơn vị phần cứng phối hợp, bao gồm một hoặc nhiều bộ xử lý như đã mô tả trên đây, cùng với phần mềm và/hoặc phần sụn phù hợp.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã luồng bit video được lập mã, trong đó phương pháp này bao gồm các bước:

thu thập, từ luồng bit video được lập mã này, phần tử cú pháp thứ nhất với chỉ số i mà quy định xem lớp thứ nhất với chỉ số i này có sử dụng việc dự đoán liên lớp hay không;

thu thập, từ luồng bit video được lập mã này, một hoặc nhiều phần tử cú pháp thứ hai liên quan đến một hoặc nhiều lớp thứ hai, mỗi phần tử cú pháp thứ hai với chỉ số i và chỉ số j quy định xem lớp thứ hai với chỉ số j này có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất với chỉ số i hay không;

trong đó ít nhất một phần tử cú pháp thứ hai trong số các phần tử cú pháp thứ hai với chỉ số j trong khoảng từ 0 đến $i-1$, không loại trừ, là bằng 1, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất với chỉ số i là được phép sử dụng việc dự đoán liên lớp; trong đó việc ít nhất một phần tử cú pháp thứ hai bằng 1 quy định rằng lớp thứ hai với chỉ số j là lớp tham chiếu trực tiếp cho lớp thứ nhất với chỉ số i ;

thực hiện việc dự đoán liên lớp đối với ảnh của lớp thứ nhất nhờ sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu.

2. Phương pháp theo điểm 1, trong đó việc phần tử cú pháp thứ nhất bằng 1 quy định rằng lớp thứ nhất không sử dụng việc dự đoán liên lớp; hoặc việc phần tử cú pháp thứ nhất bằng 0 quy định rằng lớp thứ nhất được phép sử dụng việc dự đoán liên lớp.

3. Phương pháp theo điểm 1 hoặc 2, trong đó việc phần tử cú pháp thứ hai bằng 0 quy định rằng lớp thứ hai mà liên quan đến phần tử cú pháp thứ hai thì không phải là lớp tham chiếu trực tiếp cho lớp thứ nhất.

4. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 3, trong đó bước thu thập một hoặc nhiều phần tử cú pháp thứ hai là được thực hiện trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

5. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 4, trong đó phương pháp này còn bao gồm bước: thực hiện việc dự đoán đối với ảnh của lớp thứ nhất mà không sử dụng ảnh của lớp thứ hai liên quan đến ít nhất một phần tử cú pháp thứ hai làm ảnh tham chiếu, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất không sử dụng việc dự đoán liên lớp.

6. Phương pháp mã hoá luồng bit video được lập mã, trong đó phương pháp này bao gồm các bước:

xác định xem ít nhất một lớp thứ hai có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất hay không;

mã hoá phần tử cú pháp thứ nhất với chỉ số i vào luồng bit video được lập mã này, trong đó phần tử cú pháp thứ nhất này quy định xem lớp thứ nhất với chỉ số i này có sử dụng việc dự đoán liên lớp hay không;

mã hoá một hoặc nhiều phần tử cú pháp thứ hai liên quan đến ít nhất một lớp thứ hai vào luồng bit video được lập mã này, trong đó mỗi phần tử cú pháp thứ hai với chỉ số i và chỉ số j quy định xem lớp thứ hai với chỉ số j có phải là lớp tham chiếu trực tiếp cho lớp thứ nhất với chỉ số i hay không; trong đó việc phần tử cú pháp thứ hai bằng 1 quy định rằng lớp thứ hai với chỉ số j là lớp tham chiếu trực tiếp cho lớp thứ nhất với chỉ số i ;

trong đó ít nhất một phần tử cú pháp thứ hai trong số các phần tử cú pháp thứ hai với chỉ số j trong khoảng từ 0 đến $i-1$, không loại trừ, là bằng 1, trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất với chỉ số i là được phép sử dụng việc dự đoán liên lớp.

7. Phương pháp theo điểm 6, trong đó việc phần tử cú pháp thứ nhất bằng 1 quy định rằng lớp thứ nhất không sử dụng việc dự đoán liên lớp; hoặc việc phần tử cú pháp thứ nhất bằng 0 quy định rằng lớp thứ nhất được phép sử dụng việc dự đoán liên lớp.

8. Phương pháp theo điểm 6 hoặc 7, trong đó việc phần tử cú pháp thứ hai bằng 0 quy định rằng lớp thứ hai mà liên quan đến phần tử cú pháp thứ hai thì không phải là lớp tham chiếu trực tiếp của lớp thứ nhất.

9. Phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 8, trong đó bước mã hoá một hoặc nhiều phần tử cú pháp thứ hai liên quan đến ít nhất một lớp thứ hai vào luồng bit video được lập mã là được thực hiện trong trường hợp mà giá trị của phần tử cú pháp thứ nhất quy định lớp thứ nhất là được phép sử dụng việc dự đoán liên lớp.

10. Bộ giải mã, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ phi nhất thời đọc được bằng máy tính được ghép nối đến các bộ xử lý này và lưu giữ chương trình để thực thi bởi các bộ xử lý này, trong đó chương trình này, khi được thực thi bởi các bộ xử lý này, thì được tạo cấu hình để khiến bộ giải mã này thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5.

11. Bộ mã hoá, bao gồm:

một hoặc nhiều bộ xử lý; và

phương tiện lưu trữ phi nhất thời đọc được bằng máy tính được ghép nối đến các bộ xử lý này và lưu giữ chương trình để thực thi bởi các bộ xử lý này, trong đó chương trình này, khi được thực thi bởi các bộ xử lý này, thì được tạo cấu hình để khiến bộ mã hoá này thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 9.

12. Phương tiện phi nhất thời đọc được bằng máy tính có mang mã chương trình mà khi được thực thi bởi thiết bị máy tính thì khiến thiết bị máy tính đó thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5.

13. Phương tiện phi nhất thời đọc được bằng máy tính có mang mã chương trình mà khi được thực thi bởi thiết bị máy tính thì khiến thiết bị máy tính đó thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 6 đến 9.

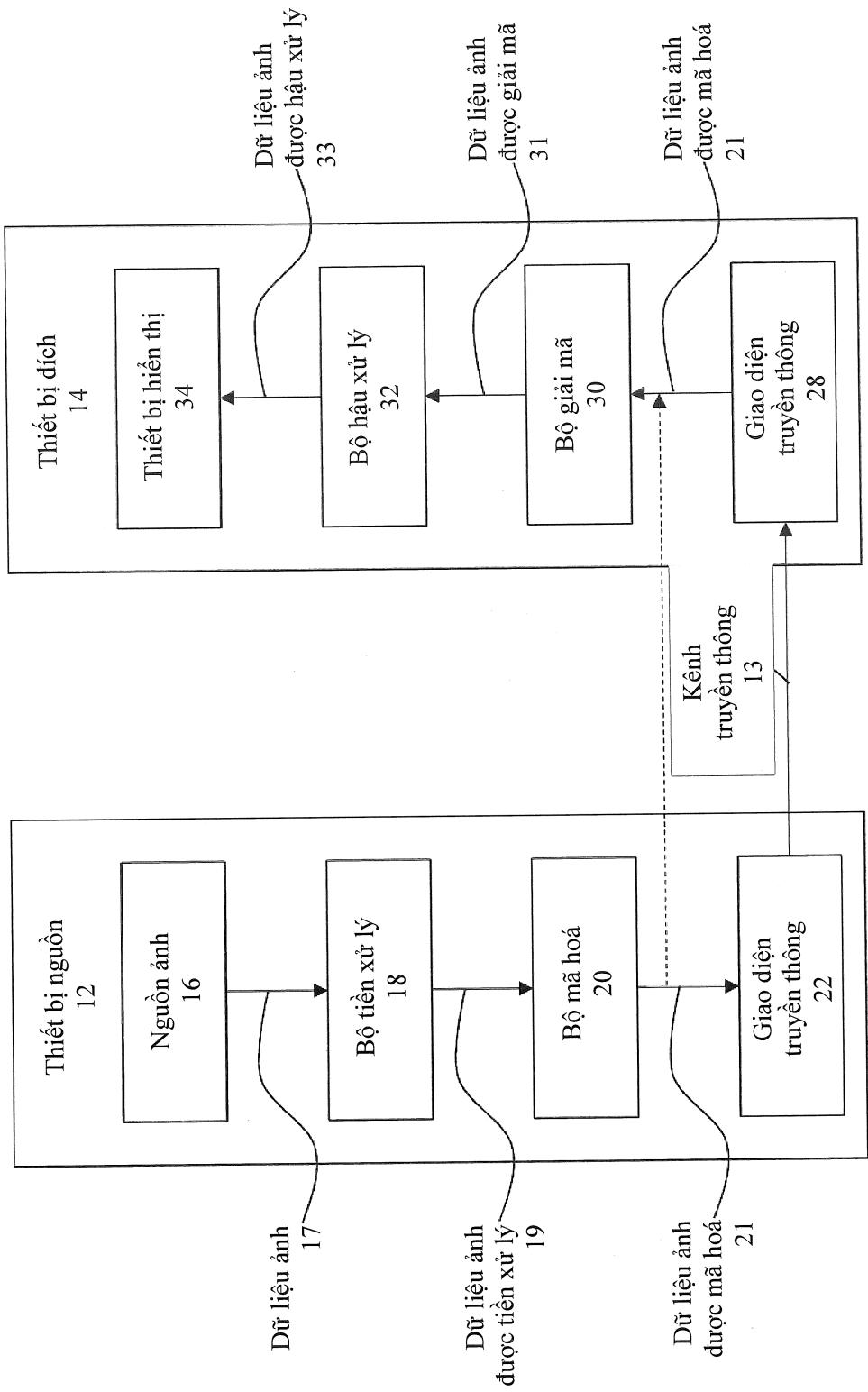
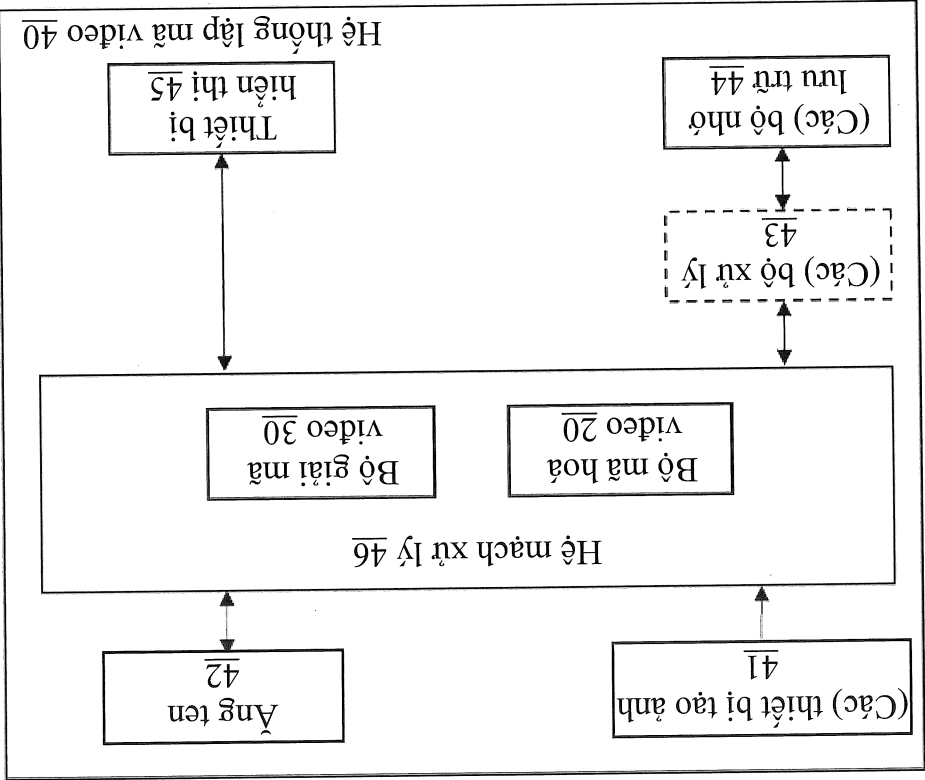


Fig.1A

Fig. 1B



3/13

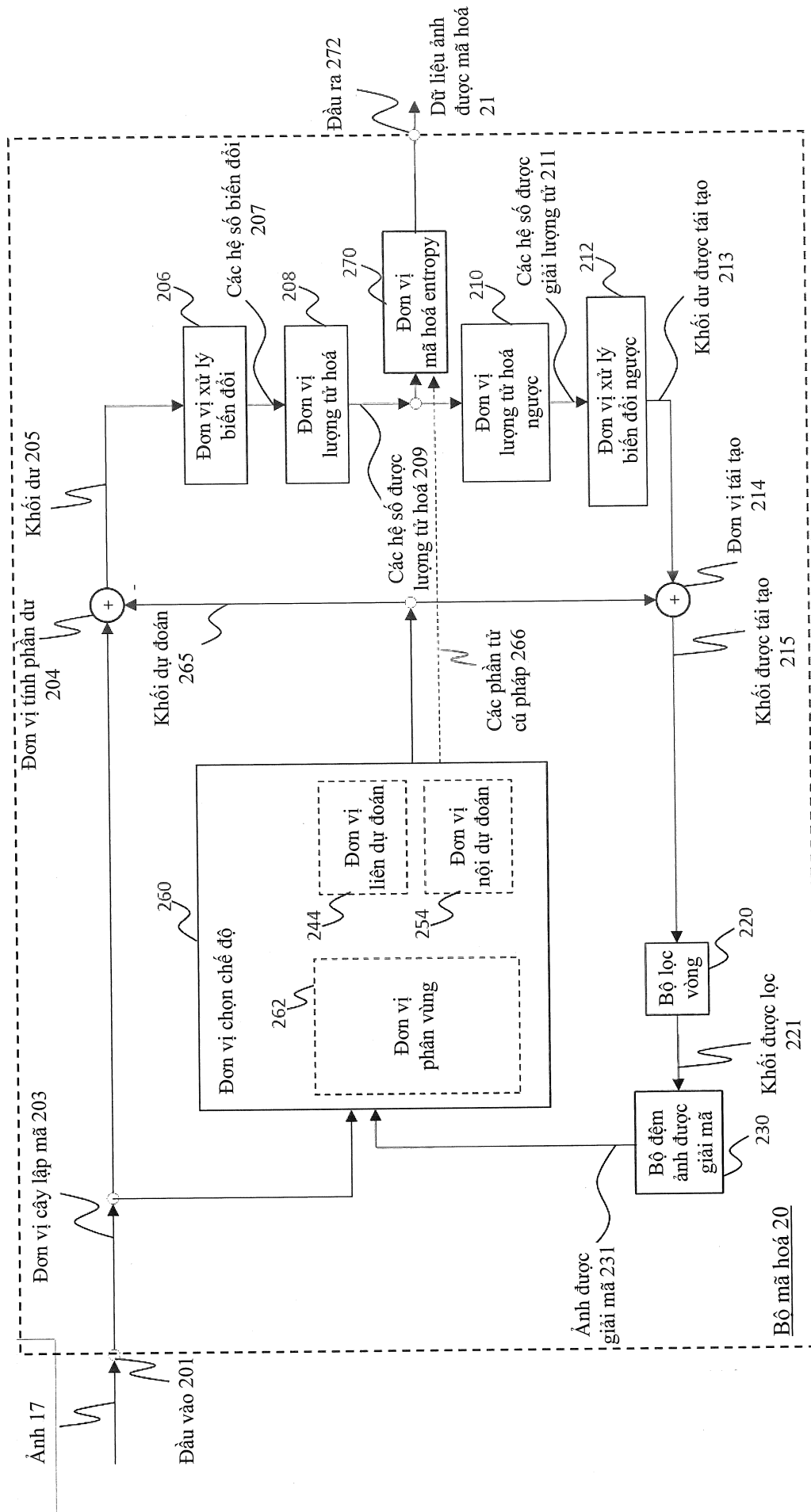


Fig.2

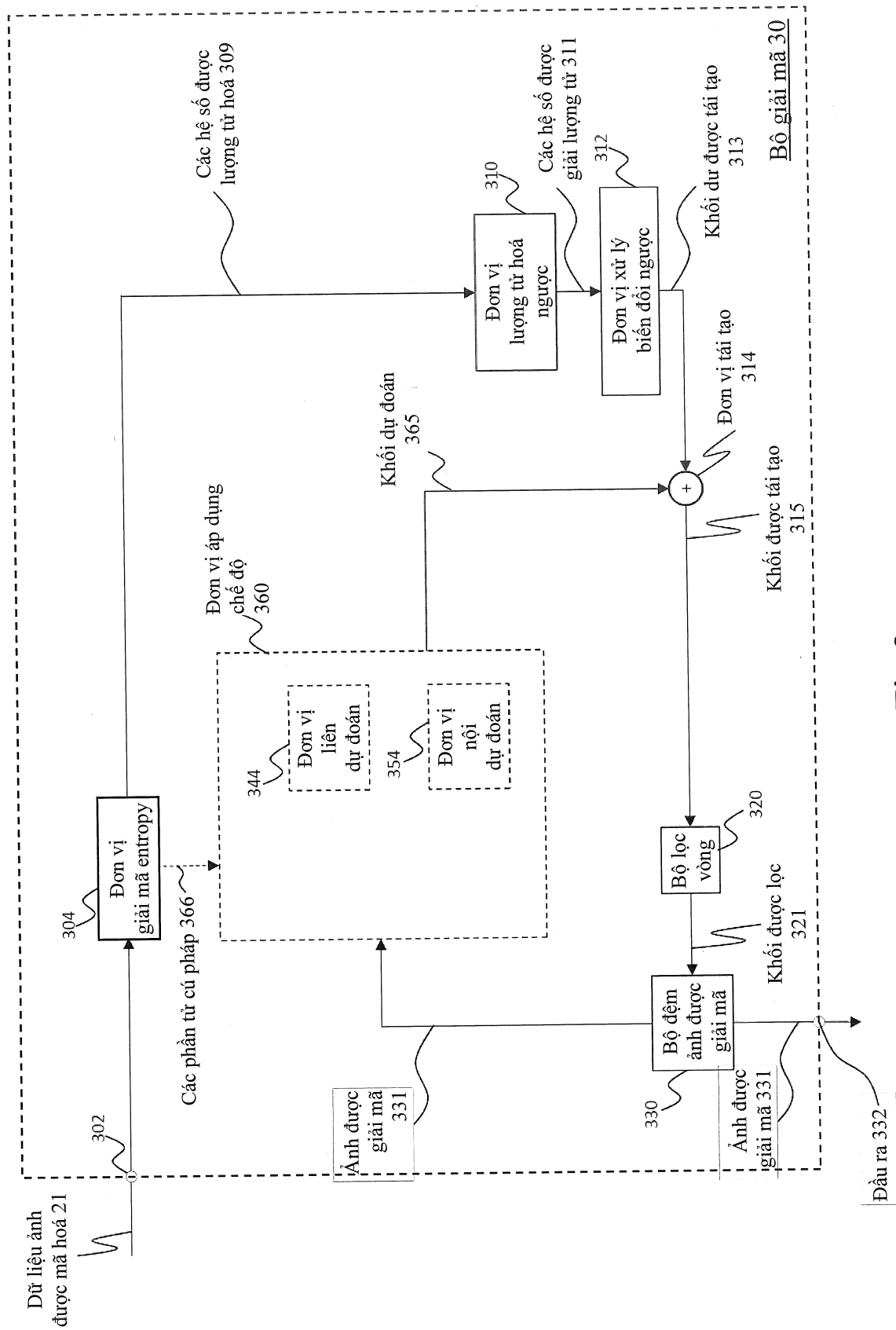
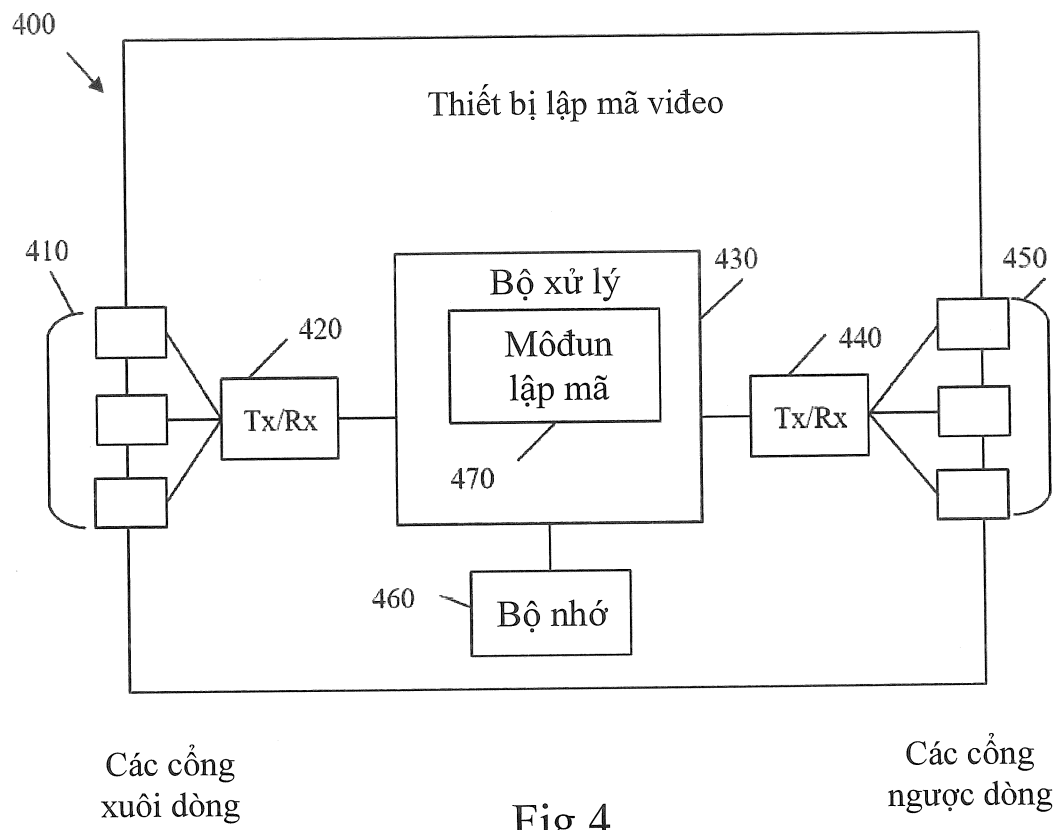


Fig.3



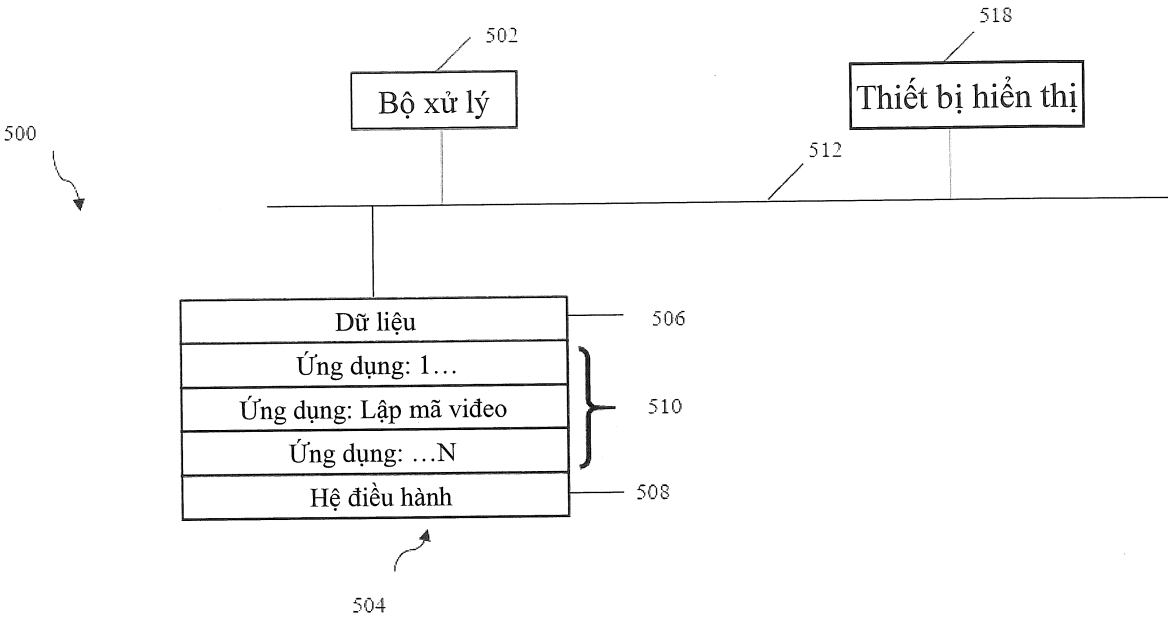


Fig.5

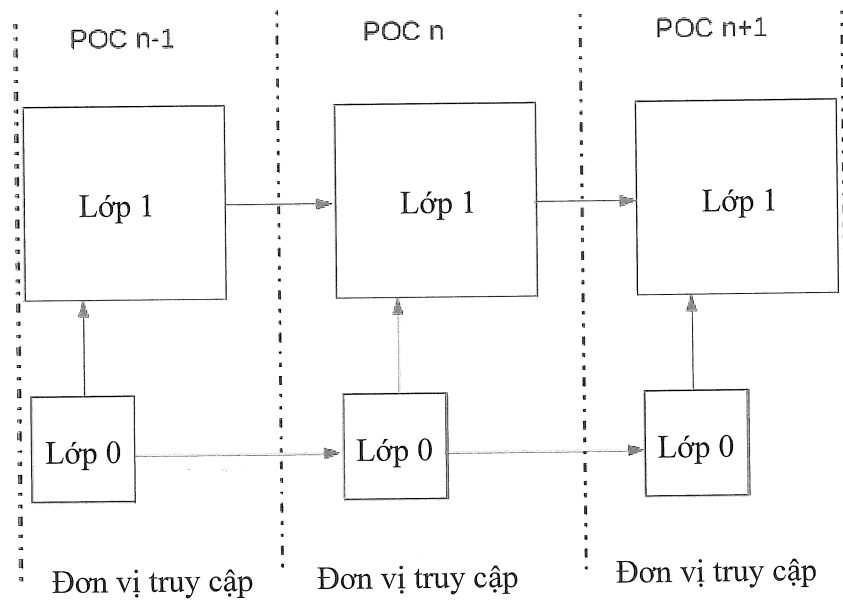


Fig.6

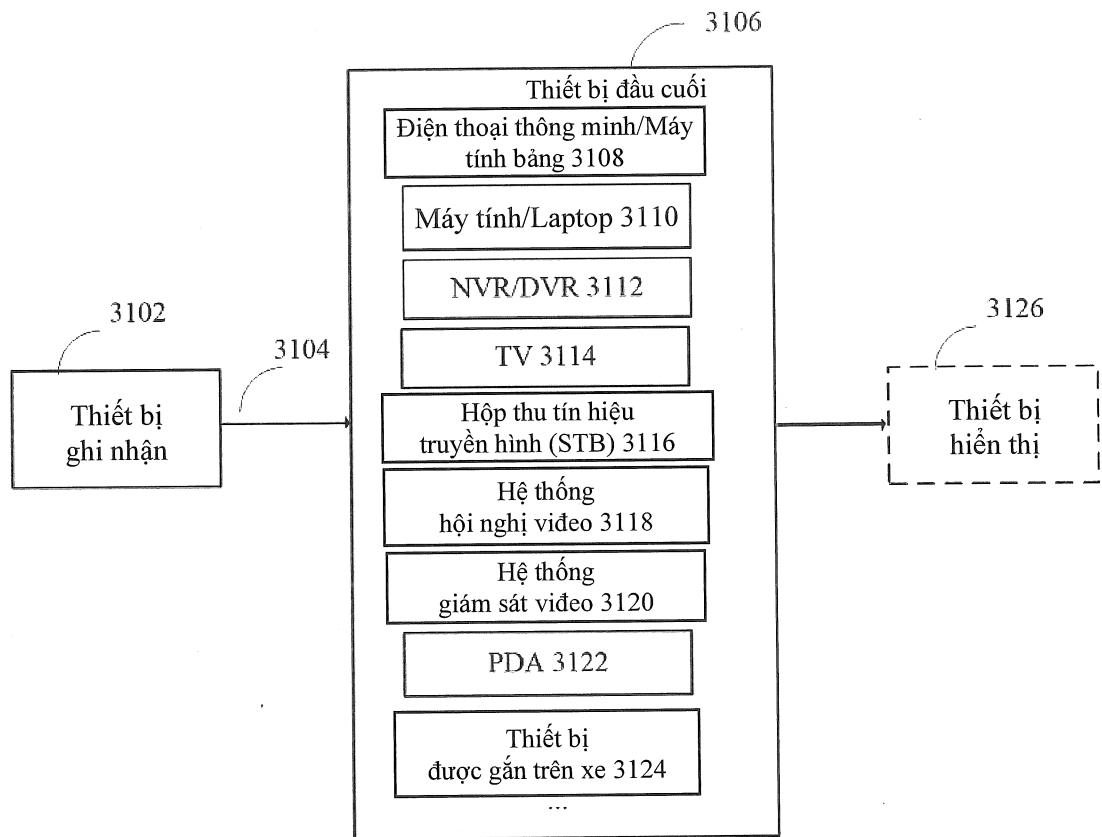


Fig.7

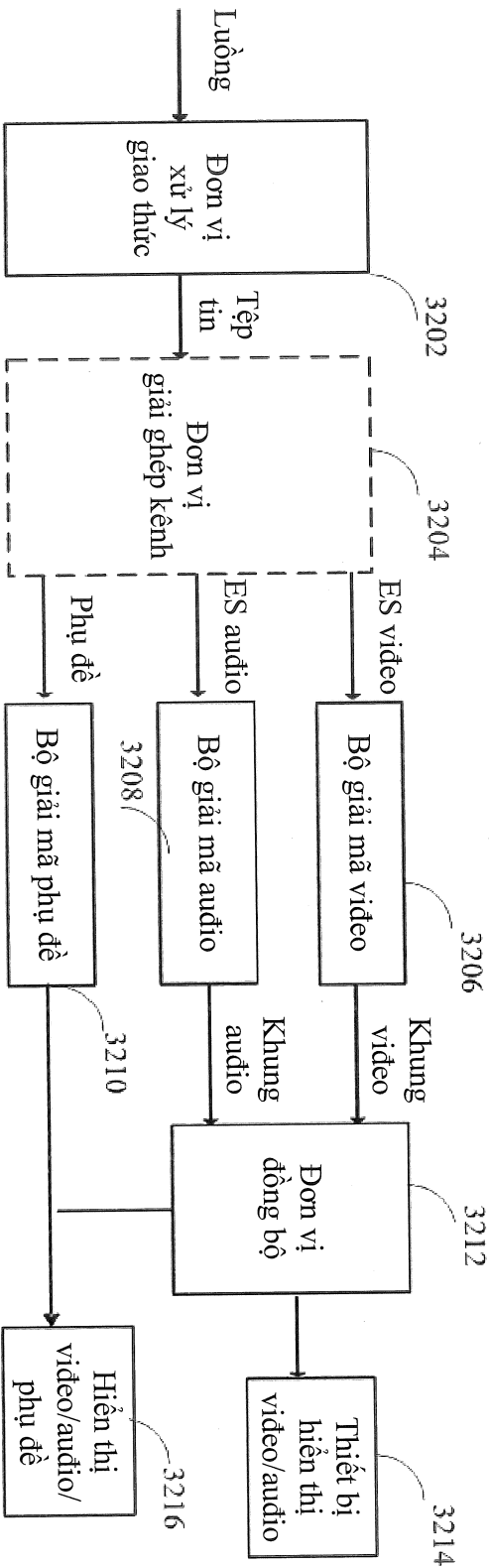


Fig.8

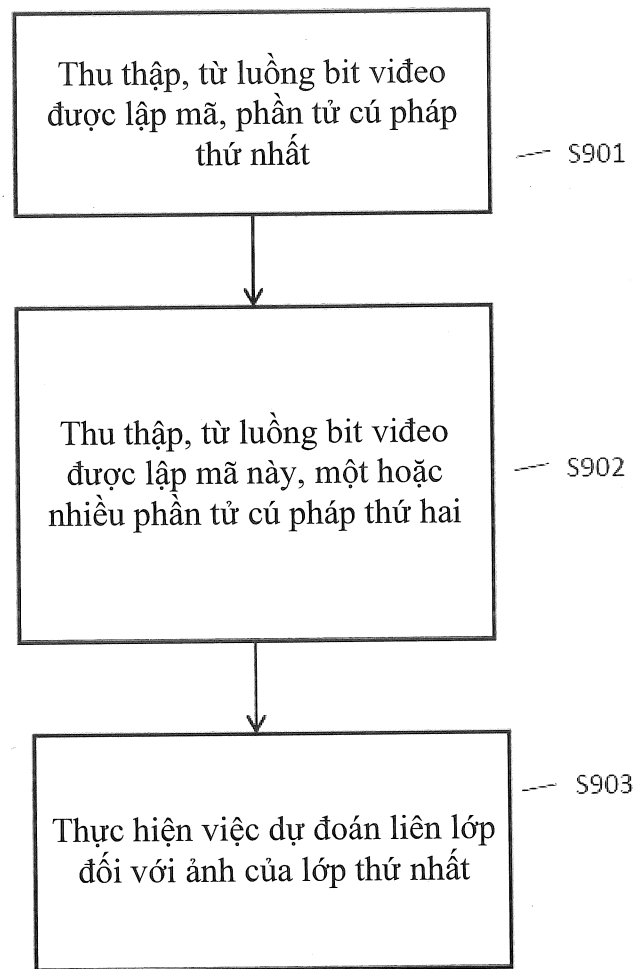


Fig.9

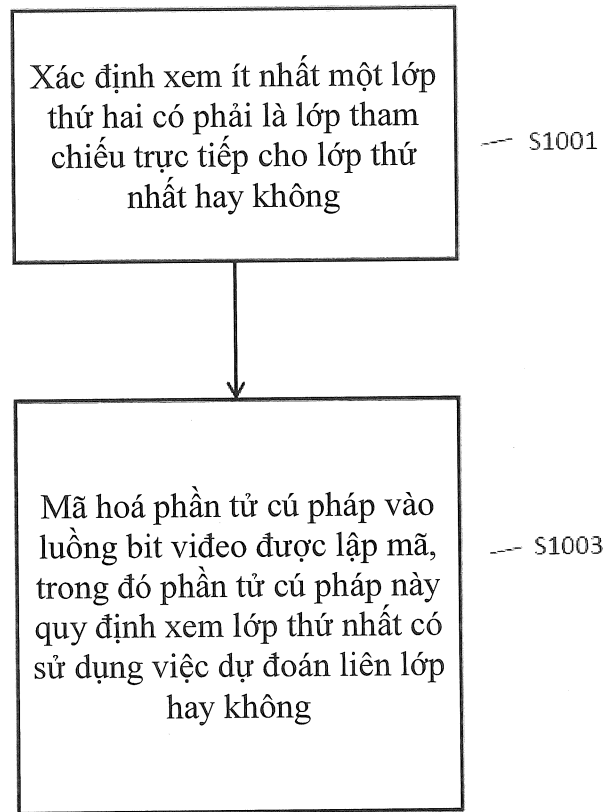


Fig.10

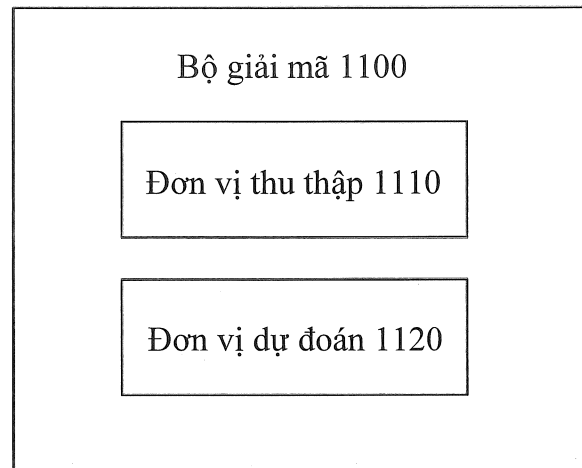


Fig.11

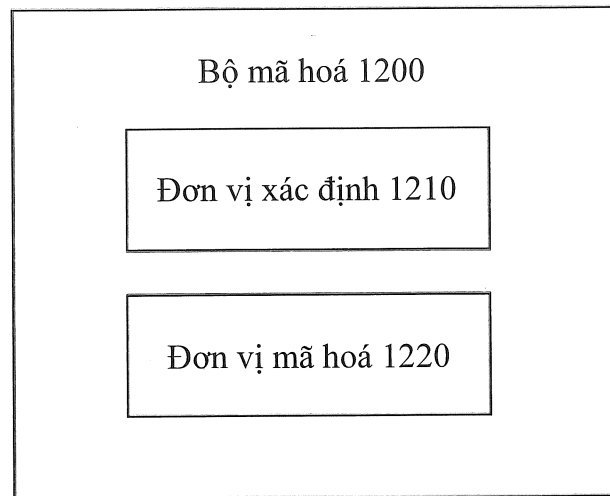


Fig.12