



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

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11)
CỤC SỞ HỮU TRÍ TUỆ



1-0045041

(51)^{2020.01} H04N 19/70; H04N 19/174; H04N 19/46 (13) B

(21) 1-2021-00209

(22) 12/06/2020

(86) PCT/SE2020/050611 12/06/2020

(87) WO2020/263158 30/12/2020

(30) 62/865,464 24/06/2019 US

(45) 25/04/2025 445

(43) 25/03/2022 408A

(71) TELEFONAKTIEBOLAGET LM ERICSSON (PUBL) (SE)

SE-164 83 Stockholm, Sweden

(72) SJÖBERG, Rickard (SE); DAMGHANIAN, Mitra (IR); WENNERSTEN, Per (SE);
PETTERSSON, Martin (SE).

(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP ĐỂ GIẢI MÃ VÀ MÃ HÓA, BỘ GIẢI MÃ, BỘ MÃ HÓA VÀ
VẬT GHI ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(21) 1-2021-00209

(57) Sáng chế đề cập đến phương pháp (400) được thực hiện bởi bộ giải mã (260, 600, 701) để giải mã luồng bit bao gồm tập tham số ảnh (PPS, picture parameter set) và tập các lát thứ nhất. Phương pháp gồm bước thu (s402) tập tham số ảnh. Phương pháp còn gồm bước giải mã (s404) phần tử cú pháp được gồm trong tập tham số ảnh để thu giá trị chỉ báo. Bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt thành giá trị thứ nhất sau đó bộ giải mã xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, nếu không thì bộ giải mã xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể. Nếu tiêu đề ảnh bao gồm giá trị tham số tương ứng với tham số cụ thể, sau đó giá trị tham số này được sử dụng để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất. Sáng chế còn đề cập đến phương pháp mã hóa, bộ giải mã, bộ mã hóa và vật ghi đọc được bằng máy tính.

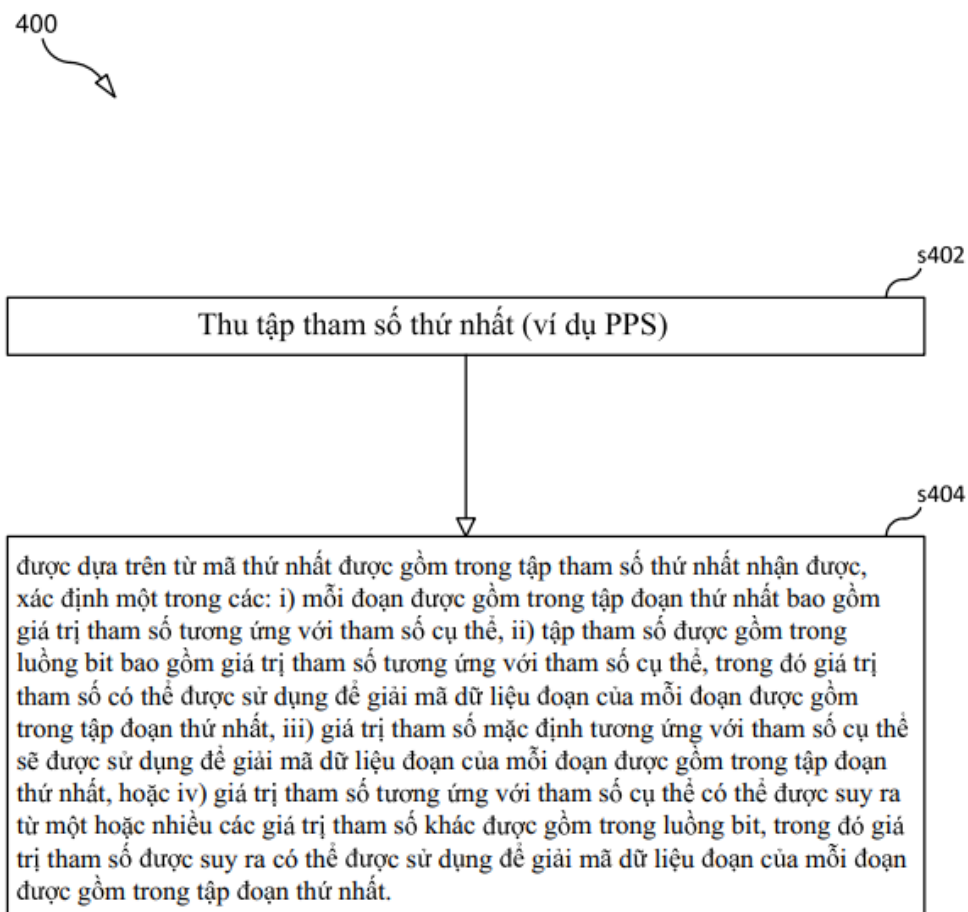


FIG. 4

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp mã hóa và giải mã video.

Tình trạng kỹ thuật của sáng chế

1.1 HEVC và VVC

Mã hóa video hiệu quả cao (High Efficiency Video Coding, HEVC) là bộ mã hóa-giải mã video được dựa vào khối được chuẩn hóa bởi ITU-T và MPEG. Bộ mã hóa-giải mã video HEVC sử dụng cả dự đoán theo thời gian và không gian. Dự đoán không gian được thực hiện bằng cách sử dụng nội dự đoán (I) từ trong ảnh hiện tại. Dự đoán thời gian được thực hiện bằng cách sử dụng liên dự đoán đơn hướng (P) hoặc hai hướng (B) trên mức khối từ các ảnh tham chiếu đã được giải mã trước đó. Trong bộ mã hóa, sự khác biệt giữa dữ liệu điểm ảnh gốc và dữ liệu điểm ảnh dự đoán, được gọi là phần dư, được chuyển đổi thành miền tần số, được lượng tử hóa và sau đó được mã hóa entropy trước khi được truyền cùng với các tham số dự đoán cần thiết, như là, ví dụ, chế độ dự đoán và các vectơ chuyển động, mà cũng được mã hóa entropy. Bộ giải mã thực hiện giải mã entropy, lượng tử hóa nghịch đảo và biến đổi nghịch đảo để thu phần dư, và sau đó bổ sung phần dư cho nội dự đoán hoặc liên dự đoán để tái tạo ảnh.

MPEG và ITU-T đang nghiên cứu sản phẩm kế thừa HEVC trong nhóm chuyên gia video chung (JVET, Joint Video Experts Team). Tên của bộ mã hóa-giải mã video này đang được phát triển là mã hóa video đa năng (Versatile Video Coding, VVC). Phần mô tả dự thảo VVC hiện tại được tìm thấy trong JVET tài liệu JVET-N1001-v7.

1.2 Các thành phần

Chuỗi video gồm có chuỗi các ảnh trong đó mỗi ảnh gồm có một hoặc nhiều các thành phần. Mỗi thành phần có thể được mô tả như là mảng hình chữ nhật hai chiều của các giá trị mẫu. Thường là ảnh trong chuỗi video gồm có ba thành phần: một thành phần độ chói Y, trong đó các giá trị mẫu là các giá trị độ chói, và hai

thành phần sắc độ Cb và Cr, trong đó các giá trị mẫu là các giá trị sắc độ. Thường kích thước của các thành phần sắc độ nhỏ hơn các thành phần độ chói hai lần ở mỗi chiều để tiết kiệm bit trong bước nén. Ví dụ, kích thước của thành phần độ chói của ảnh HD là 1920x1080 và các thành phần sắc độ của mỗi chiều là 960x540. Các thành phần được gọi là các thành phần màu sắc.

1.3 Các khối và các đơn vị

Khối là một mảng hai chiều các giá trị mẫu (ngắn gọn là “các mẫu”). Trong mã hóa video, mỗi thành phần được chia vào các khối và luồng bit video được mã hóa gồm có chuỗi các khối được mã hóa. Thông thường trong mã hóa video mà ảnh được chia vào trong các đơn vị mà bao phủ vùng cụ thể của ảnh. Mỗi đơn vị gồm có tất cả các khối từ tất cả các thành phần mà tạo nên vùng cụ thể và mỗi khối hoàn toàn thuộc về một đơn vị. Khối macro trong H.264 và đơn vị mã hóa (CU, coding unit) trong HEVC là các ví dụ về các đơn vị A.

Khối có thể được định nghĩa theo cách khác là mảng hai chiều mà phép biến đổi được sử dụng trong mã hóa được áp dụng. Những khối này được gọi là “các khối biến đổi”. Ngoài ra, khối có thể được định nghĩa là mảng hai chiều được áp dụng một chế độ dự đoán duy nhất. Các khối này được gọi là “các khối dự đoán”. Theo sáng chế này, khối từ không bị ràng buộc với một trong các định nghĩa này nhưng các mô tả ở đây có thể áp dụng cho một trong hai định nghĩa.1

1.4 Các đơn vị NAL

Cả HEVC và VVC đều xác định Lớp trừu tượng mạng (NAL, Network Abstraction Layer). Tất cả dữ liệu (tức là cả dữ liệu lớp mã hóa video (VCL, Video Coding Layer) hoặc dữ liệu không phải VCL) trong HEVC và VVC được đóng gói trong các đơn vị NAL. Đơn vị VCL NAL chứa dữ liệu đại diện cho các giá trị mẫu - tức là đơn vị VCL NAL chứa “dữ liệu mẫu”. Một đơn vị NAL không phải VCL chứa dữ liệu liên quan như các tập tham số và thông báo thông tin nâng cao bổ sung (SEI, supplemental enhancement information). Đơn vị NAL trong HEVC và VVC bắt đầu bằng tiêu đề đơn vị NAL mà mô tả loại đơn vị NAL của đơn vị NAL, ID lớp của lớp mà đơn vị NAL thuộc về và ID tạm thời của lớp con mà NAL đơn vị thuộc về. Loại đơn vị NAL xác định loại dữ liệu được mang trong đơn vị NAL.

Trong HEVC, `forbidden_zero_bit` được báo hiệu để tránh mô phỏng mã hóa bắt đầu luồng sơ cấp được đóng gói theo lớp luồng (PES, packetized elementary stream) MPEG-2 và loại đơn vị NAL được truyền trong phần từ cú pháp `nal_unit_type` trong tiêu đề đơn vị NAL. Các phần từ cú pháp cho tiêu đề đơn vị NAL trong HEVC như được thể hiện trong Bảng 1, và các phần từ cú pháp cho tiêu đề đơn vị NAL trong phiên bản hiện tại của đặc tả dự thảo VVC, JVET-N1001-v7, như được thể hiện trong Bảng 2.

BẢNG 1 - Cú pháp tiêu đề đơn vị HEVC NAL

<code>nal_unit_header() {</code>	Phần mô tả
<code>forbidden_zero_bit</code>	<code>f(1)</code>
<code>nal_unit_type</code>	<code>u(6)</code>
<code>nuh_layer_id</code>	<code>u(6)</code>
<code>nuh_temporal_id_plus1</code>	<code>u(3)</code>
<code>}</code>	

BẢNG 2 - Cú pháp tiêu đề đơn vị NAL cho phiên bản hiện tại của đặc tả dự thảo VVC

<code>nal_unit_header() {</code>	Phần mô tả
<code>zero_tid_required_flag</code>	u(1)
<code>nuh_temporal_id_plus1</code>	u(3)
<code>nal_unit_type_lsb</code>	u(4)
<code>nuh_layer_id</code>	u(7)
<code>nuh_reserved_zero_bit</code>	u(1)
<code>}</code>	

Trong phiên bản hiện tại của VVC, `zero_tid_required_flag` được báo hiệu là bit thứ nhất của tiêu đề đơn vị NAL. Khi cờ này bằng 1, `nuh_temporal_id_plus1` phải bằng 1. `zero_tid_required_flag` sau đó có thể được sử dụng cùng với `nal_unit_type_lsb` để trích loại đơn vị NAL theo: $\text{NalUnitType} = (\text{zero_tid_required_flag} \ll 4) + \text{nal_unit_type_lsb}$.

Bộ giải mã hoặc bộ phân tích cú pháp luồng bit có thể xác định các xử lý đơn vị NAL, ví dụ được phân tích cú pháp và được giải mã, sau khi xem tiêu đề đơn vị NAL. Phần còn lại của các byte của đơn vị NAL là tải tin của loại được chỉ báo bởi loại đơn vị NAL. Luồng bit VVC (cũng như luồng bit HEVC) gồm có chuỗi các đơn vị NAL. Luồng bit có thể được truyền qua mạng bởi bộ phát hoặc được lưu trữ trong đơn vị lưu trữ. Theo đó, bộ giải mã có thể thu luồng bit bằng cách nhận từ bộ phát hoặc lấy nó từ đơn vị lưu trữ.

Loại đơn vị NAL chỉ báo và định nghĩa cách đơn vị NAL được phân tích cú pháp và được giải mã. Đơn vị VCL NAL cung cấp thông tin về loại ảnh của ảnh hiện tại. Các loại đơn vị NAL của phiên bản hiện tại của đặc tả dự thảo VVC được minh họa trong Bảng 3.

Thứ tự giải mã là thứ tự mà trong đó các đơn vị NAL sẽ được giải mã, mà giống như thứ tự của các đơn vị NAL trong luồng bit. Thứ tự giải mã có thể khác

với thứ tự đầu ra, mà là thứ tự mà trong đó các ảnh được giải mã được đưa ra, như là để hiển thị, bởi bộ giải mã.

BẢNG 3 - các loại đơn vị NAL trong phiên bản hiện tại của đặc tả dự thảo VVC

NalUnitType	Tên của NalUnitType	Nội dung của đơn vị NAL và cấu trúc cú pháp Rbsp	loại lớp đơn vị NAL
0	PPS_NUT	Tập tham số ảnh pic_parameter_set_rbsp()	không phải VCL
1	AUD_NUT	Bộ phân định đơn vị truy cập access_unit_delimiter_rbsp()	không phải VCL
2 3	PREFIX_SEI_NUT SUFFIX_SEI_NUT	Thông tin nâng cao bổ sung sei_rbsp()	không phải VCL
4	APS_NUT	Tập tham số thích ứng adaptation_parameter_set_rbsp()	không phải VCL

5..7	RSV_NVCL5. RSV_NVCL7	Được dành riêng	không phải VCL
8	TRAIL_NUT	Lát được mã hóa của ảnh theo sau không phải STSA slice_layer_rbsp()	VCL
9	STSA_NUT	Lát được mã hóa của ảnh STSA slice_layer_rbsp()	VCL
10	RADI_NUT	Lát được mã hóa của ảnh RADI slice_layer_rbsp()	VCL
11	RASL_NUT	Lát được mã hóa của ảnh RASL slice_layer_rbsp()	VCL
12..15	RSV_VCL_12.. RSV_VCL_15	Các loại đơn vị VCL NAL không phải IRAP được dành riêng	VCL
16	DPS_NUT	Giải mã tập tham số decoding_parameter_set_rbsp()	không phải VCL
17	SPS_NUT	Tập tham số chuỗi seq_parameter_set_rbsp()	không phải VCL
18	EOS_NUT	Đầu cuối của chuỗi end_of_seq_rbsp()	không phải VCL
19	EOB_NUT	Đầu cuối của luồng bit end_of_bitstream_rbsp()	không phải

			VCL
20	VPS_NUT	Tập tham số video video_parameter_set_rbsp()	không phải VCL
21.23	RSV_NVCL21.. RSV_NVCL23	Reserved	không phải VCL
24 25	IDR_W_RADL IDR_N_LP	Lát được mã hóa của ảnh IDR slice_layer_rbsp()	VCL
26	CRA_NUT	Lát được mã hóa của ảnh CRA slice_layer_rbsp()	VCL
27	GRA_NUT	Lát được mã hóa của ảnh truy cập ngẫu nhiên dần dần slice_layer_rbsp()	VCL
28..31	UNSPEC28.. UNSPEC31	Không xác định	không phải VCL

1.5 Ảnh điểm truy cập ngẫu nhiên nội bộ (IRAP, intra random access point) và chuỗi video được mã hóa (CVS, coded video sequence).

Để mã hóa lớp đơn trong HEVC và đặc tả dự thảo hiện tại VVC, đơn vị truy cập (AU, access unit) là biểu diễn mã của ảnh đơn. AU có thể gồm có vài lớp mã hóa video (VCL, video coding layer) cũng như các đơn vị NAL không phải VCL. AU có thể bắt đầu một cách tùy ý với đơn vị truy cập bộ phận định (AUD) mà chỉ báo việc bắt đầu của AU và loại của các lát được cho phép trong ảnh, nghĩa là I, I-P hoặc I-P-B.

Ảnh điểm truy cập ngẫu nhiên nội bộ (IRAP) in HEVC là ảnh không tham chiếu đến các ảnh bất kỳ khác nào ngoài chính nó để dự đoán trong thủ tục giải mã của nó. Ảnh thứ nhất trong luồng bit trong thủ tục giải mã trong HEVC phải là ảnh

IRAP nhưng ảnh IRAP cũng có thể xuất hiện sau đó trong luồng bit. HEVC định rõ ba loại ảnh IRAP, ảnh truy cập liên kết hỏng (broken link access, BLA), ảnh làm mới bộ giải mã tức thời (instantaneous decoder refresh, IDR) và ảnh truy cập ngẫu nhiên sạch (clean random access, CRA).

Chuỗi video được mã hóa (CVS, coded video sequence) trong HEVC là một loạt các AU bắt đầu từ IRAP AU trở lên, nhưng không gồm IRAP AU tiếp theo trong thứ tự giải mã.

Các ảnh IDR luôn bắt đầu CVS mới. Ảnh IDR có thể có các ảnh dẫn đầu có thể giải mã truy cập ngẫu nhiên được liên kết (RADL, random access decodable leading). Ảnh IDR không có các ảnh RASL được liên kết.

Các ảnh BLA trong HEVC cũng bắt đầu CVS mới và có hiệu quả tương tự trên thủ tục giải mã như ảnh IDR. Tuy nhiên, ảnh BLA trong HEVC có thể chứa các phần tử cú pháp mà chỉ rõ tập ảnh tham chiếu không trống. Ảnh BLA có thể có các ảnh RASL được liên kết, mà không được đưa ra bởi bộ giải mã và có thể không giải mã được, vì chúng có thể chứa các tham chiếu đến ảnh mà có thể không được biểu diễn trong luồng bit. Ảnh BLA cũng có thể có các ảnh RADL được liên kết, mà được giải mã. Các ảnh BLA không được định nghĩa trong phiên bản hiện tại của VVC.

Ảnh CRA có thể có các ảnh RADL hoặc RASL được kết nối. Vì với ảnh BLA, ảnh CRA có thể chứa các phần tử cú pháp mà chỉ rõ tập ảnh tham chiếu không trống. Đối với các ảnh CRA, có thể được đặt để chỉ rõ các ảnh RASL được liên kết không được đưa ra bởi bộ giải mã, do chúng có thể không giải mã được, vì chúng có thể chứa các tham chiếu đến ảnh mà không được biểu diễn trong luồng bit. CRA có thể bắt đầu CVS.

Trong phiên bản hiện tại của đặc tả dự thảo VVC, CVS được bắt đầu tại đơn vị truy cập bắt đầu CVS (CVSS, CVS start), mà có thể chứa ảnh IRAP, nghĩa là, ảnh IDR hoặc CRA, hoặc ảnh truy cập ngẫu nhiên dần dần (gradual random access, GRA).

Các ảnh GRA cơ bản được sử dụng để truy cập ngẫu nhiên trong các luồng bit

được mã hóa để mã hóa độ trễ thấp trong đó ảnh IRAP đầy đủ sẽ gây ra độ trễ lớn. Ảnh GRA có thể sử dụng làm mới nội dần dần mà cập nhật ảnh video bởi ảnh trong đó mỗi ảnh chỉ được mã hóa nội một phần. Nó được báo hiệu bằng ảnh GRA khi video được làm mới hoàn toàn và sẵn sàng để đưa ra, với điều kiện là luồng bit được điều chỉnh trong ảnh GRA. GRA có thể bắt đầu CVS.

1.6 Các tập tham số

HEVC và VVC chỉ rõ ba loại tập tham số: 1) tập tham số ảnh (PPS, picture parameter set), 2) tập tham số chuỗi (SPS, sequence parameter set), và 3) tập tham số video (VPS). PPS chứa dữ liệu mà chung cho một hoặc nhiều ảnh, SPS chứa dữ liệu mà chung cho chuỗi video được mã hóa (CVS, coded video sequence), và VPS chứa dữ liệu mà chung cho nhiều CVS.

Phiên bản hiện tại của VVC cũng định rõ hai tập tham số bổ sung: tập tham số thích ứng (APS, adaptation parameter set) và tập tham số bộ giải mã (DPS, decoder parameter set). APS mang các tham số cần thiết cho công cụ bộ lặp vòng lặp thích ứng (adaptive loop filter, ALF) và công cụ ánh xạ độ chói và chia tỷ lệ sắc độ (LMCS, luma mapping and chroma scaling). DPS định rõ thông tin mà có thể không thay đổi trong suốt phiên giải mã và có thể tốt cho bộ giải mã để biết về, ví dụ số lượng lớp con được phép tối đa. Thông tin trong DPS không cần thiết cho hoạt động của thủ tục giải mã.

1.7 Các ô và các ô đơn vị

Tiêu chuẩn mã hóa video VVC dự thảo gồm công cụ gọi là ô mà chia hình ảnh thành các vùng độc lập về không gian hình chữ nhật. Ô trong tiêu chuẩn mã hóa video VVC dự thảo tương tự với ô được sử dụng trong HEVC, nhưng với cơ chế phân vùng hai bước. Sử dụng ô, ảnh trong VVC có thể được phân chia thành các hàng và các cột của các mẫu trong đó ô là giao điểm của hàng và cột. Ví dụ, ảnh có thể được chia thành 4 hàng ô và 5 cột ô dẫn đến là 20 ô cho cả ảnh.

Cấu trúc ô được báo hiệu trong PPS bằng cách chỉ định độ dày (tức là chiều cao) của các hàng và chiều rộng của cột. Các hàng và cột riêng lẻ có thể có các kích thước khác nhau, nhưng sự phân vùng luôn kéo dài trên toàn bộ ảnh, lần lượt

từ trái sang phải và từ trên xuống dưới.

Không có sự phụ thuộc giải mã giữa các ô của cùng một ảnh. Điều này bao gồm nội dự đoán, lựa chọn ngữ cảnh để mã hóa entropy và dự đoán vectơ chuyển động. Một ngoại lệ là các phụ thuộc lọc trong vòng thường được cho phép giữa các ô.

Phân vùng ô hai bước trong VVC bắt đầu bằng cách phân vùng ảnh thành các ô như trong HEVC. Sau đó, mỗi ô có thể được tùy ý phân vùng thành các khối theo đường biên ngang. Trong dự thảo đặc điểm kỹ thuật VVC hiện tại, ô đơn vị từ cũng được sử dụng cho các loại ô không được phân chia thêm thành các ô đơn vị.

1.8 Các lát

Khái niệm cắt lát trong HEVC chia ảnh thành một hoặc nhiều lát được mã hóa độc lập, trong đó việc giải mã một lát trong ảnh là độc lập với các lát khác trong cùng một ảnh. Các kiểu mã hóa khác nhau có thể được sử dụng cho các lát của cùng một ảnh - - tức là, một lát có thể là lát I, lát P hoặc lát B. Mục đích chính của các lát là cho phép đồng bộ hóa lại trong trường hợp mất dữ liệu.

Trong phiên bản hiện tại của VVC, một lát bao gồm một số ô hoàn chỉnh hoặc chỉ một chuỗi liên tiếp các ô đơn vị hoàn chỉnh của một ô. Mỗi lát có tiêu đề lát bao gồm các tham số có thể được đặt cho từng lát và dữ liệu lát. Một số tham số bị hạn chế là giống nhau cho tất cả các lát trong ảnh. Mỗi lát trong CVS được mang trong một đơn vị VCL NAL riêng biệt. Trong phiên bản trước của đặc điểm kỹ thuật dự thảo VVC, các lát được gọi là nhóm ô.

Bản chất kỹ thuật của sáng chế

Có một số thách thức nhất định. Ví dụ, trong phiên bản hiện tại của đặc tả dự thảo VVC, một hoặc nhiều tham số cụ thể được báo hiệu trong tiêu đề lát có thể không đổi trong toàn bộ luồng bit. Theo đó, việc báo hiệu cùng một giá trị cho một tham số như vậy trong tất cả các tiêu đề lát cắt gây ra một hình phạt không cần thiết trong hiệu quả nén.

Sự kém hiệu quả này có thể được giảm bớt bằng cách, ví dụ, 1) xác định ít

nhất một tham số giá trị của nó thỉnh thoảng sẽ không đối đối với mỗi đoạn được gồm trong tập các đoạn (ví dụ, đối với mỗi lát được gồm trong CVS cụ thể) và thỉnh thoảng sẽ biến đổi và 2) gồm thông tin trong, ví dụ, tập tham số cụ thể (ví dụ, tập tham số được gồm trong CVS cụ thể) trong đó thông tin chỉ báo rằng: i) mỗi đoạn được gồm trong tập các đoạn bao gồm giá trị tham số tương ứng với tham số (ví dụ, mỗi đoạn chứa từ mã (ví dụ, phần từ cú pháp) từ đó giá trị tham số có thể được suy ra), ii) tập tham số bao gồm giá trị tham số tương ứng với tham số và giá trị tham số này có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn, iii) giá trị tham số mặc định tương ứng với tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn, hoặc iv) giá trị tham số tương ứng với tham số có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit và giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn. Bằng cách này, các bit có thể được tiết kiệm bằng cách báo hiệu trong tập tham số giá trị tham số mà có thể được sử dụng cho tất cả các đoạn trong tập đoạn cụ thể, hơn là báo hiệu giá trị tham số trong mỗi đoạn được gồm trong tập các đoạn.

Theo đó, trong một khía cạnh của sáng chế đề xuất phương pháp giải mã được thực hiện bởi bộ giải mã để giải mã luồng bit, luồng bit bao gồm tập tham số ảnh, PPS, và tập các lát thứ nhất. Phương pháp gồm bộ giải mã thu tập tham số ảnh. Phương pháp cũng gồm bộ giải mã giải mã phần từ cú pháp thứ nhất được gồm trong tập tham số ảnh để thu giá trị chỉ báo. Bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt thành giá trị thứ nhất thì sau đó nó được xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, nếu không nó được xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể. Nếu tiêu đề ảnh bao gồm giá trị tham số tương ứng với tham số cụ thể, sau đó giá trị tham số được sử dụng để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất.

Trong một khía cạnh khác đề xuất phương pháp mã hóa được thực hiện bởi bộ mã hóa. Phương pháp gồm bộ mã hóa tạo tập các lát thứ nhất. Phương pháp cũng gồm bộ mã hóa tạo tập tham số ảnh, trong đó tập tham số ảnh gồm từ mã thứ nhất mà được đặt làm giá trị thứ nhất hoặc giá trị thứ hai. Khi từ mã thứ nhất được đặt là

giá trị thứ nhất, sau đó từ mã thứ nhất chỉ báo rằng tiêu đề ảnh được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất, và khi từ mã thứ nhất được đặt là giá trị thứ hai, sau đó từ mã thứ nhất chỉ báo rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

Trong một khía cạnh khác đề xuất chương trình máy tính bao gồm các lệnh mà khi được thực thi bằng mạch xử lý làn cho mạch xử lý để thực hiện phương pháp bất kỳ được bộc lộ ở đây. Trong một khía cạnh khác đề xuất sóng mang có chứa chương trình máy tính, trong đó sóng mang là một trong những tín hiệu điện từ, tín hiệu quang học, tín hiệu vô tuyến, và vật ghi đọc được bằng máy tính.

Trong một khía cạnh khác đề xuất bộ giải mã để thực hiện phương pháp giải mã được mô tả ở trên. Theo một phương án, bộ giải mã bao gồm vật ghi đọc được bằng máy tính và mạch xử lý kết hợp với phương tiện lưu trữ, trong đó bộ giải mã được tạo cấu hình để thực hiện phương pháp giải mã.

Trong một khía cạnh khác đề xuất bộ mã hóa để thực hiện phương pháp mã hóa được mô tả ở trên. Theo một phương án, bộ mã hóa bao gồm vật ghi đọc được bằng máy tính và mạch xử lý kết hợp với phương tiện lưu trữ, trong đó bộ mã hóa được tạo cấu hình để thực hiện phương pháp mã hóa.

Ưu điểm của các phương án được bộc lộ ở đây là, đối với mỗi tham số mà không đòi hỏi ít nhất một phần của luồng bit (ví dụ trong CVS), không cần phải báo hiệu giá trị tham số cho tham số ở cấp độ đoạn (ví dụ lát), mà giúp tiết kiệm các bit và cải thiện hiệu quả nén tổng thể.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối giản lược của bộ mã hóa video theo một phương án.

Fig.2 là sơ đồ khối giản lược của bộ giải mã video theo một phương án.

Fig.3 minh họa luồng bit video được mã hóa theo phương án của sáng chế.

Fig.4 là sơ đồ minh họa thủ tục giải mã theo phương án của sáng chế.

Fig.5 là sơ đồ minh họa quy trình mã hóa theo phương án của sáng chế.

Fig.6 là sơ đồ khối thiết bị theo phương án của sáng chế.

Fig.7A là sơ đồ khối thiết bị giải mã video theo phương án của sáng chế.

Fig.7B là sơ đồ khối thiết bị mã hóa video theo phương án của sáng chế

Mô tả chi tiết sáng chế

Các phương án được mô tả ở đây có thể được sử dụng trong bộ mã hóa video hoặc bộ giải mã video.

Fig.1 là sơ đồ khối giản lược của bộ mã hóa video 140 theo một phương án. Khối hiện tại của các điểm ảnh được dự đoán bằng cách thực hiện ước tính chuyển động sử dụng bộ ước tính chuyển động 150 từ khối điểm ảnh được cung cấp sẵn trong cùng khung hoặc trong khung trước đó. Kết quả của việc ước tính chuyển động là vector chuyển động hoặc dịch chuyển được liên kết với khối tham chiếu, trong trường hợp liên dự đoán. Vector chuyển động có thể được sử dụng bởi bộ bù chuyển động 150 để đưa ra liên dự đoán của khối của các điểm ảnh. Bộ nội dự đoán 149 tính nội dự đoán của khối hiện tại của các điểm ảnh. Các kết quả đầu ra từ bộ ước tính chuyển động/bộ bù chuyển động 150 và bộ nội dự đoán 149 được đưa vào trong bộ chọn 151 mà chọn nội dự đoán hoặc liên dự đoán cho khối hiện tại của các điểm ảnh. Đầu ra từ bộ chọn 151 được đưa vào bộ tính lỗi theo dạng của bộ cộng 141 mà cũng nhận các giá trị điểm ảnh của khối hiện tại của các điểm ảnh. Bộ cộng 141 tính toán và đưa ra lỗi phần dư như là sự khác biệt trong các giá trị điểm ảnh giữa khối của các điểm ảnh và sự dự đoán của nó. Lỗi được biến đổi trong bộ biến đổi 142, như là bởi biến đổi cosin rời rạc, và được lượng tử hóa bởi bộ lượng tử hóa 143 được cho phép bằng cách mã hóa trong bộ mã hóa 144, như là bởi bộ mã hóa entropy. Trong việc liên mã hóa, vector chuyển động được ước lượng cũng được đưa tới bộ mã hóa 144 để tạo biểu diễn mã của khối hiện tại của các điểm ảnh. Lỗi phần dư được biến đổi và được lượng tử hóa cho khối hiện tại của các điểm ảnh cũng được cấp cho bộ lượng tử hóa nghịch đảo 145 và bộ biến đổi nghịch đảo 146 để lấy lỗi phần dư gốc. Lỗi này được bổ sung bởi bộ cộng 147 cho đầu ra dự đoán khối từ bộ bù chuyển động 150 hoặc bộ nội dự đoán 149 để tạo khối tham chiếu của các điểm ảnh mà có thể được sử dụng trong việc dự đoán và mã hóa của khối tiếp theo của các điểm ảnh. Khối tham chiếu mới được xử lý đầu tiên bởi bộ lọc giải khối

100. Khối tham chiếu mới được xử lý sau đó được lưu trữ tạm thời trong bộ đệm khung 148, trong đó nơi có sẵn cho bộ nội dự đoán 149 và bộ ước tính chuyển động/bù chuyển động 150.

Fig.2 là sơ đồ khối của bộ giải mã video 260 theo một số phương án. Bộ giải mã 260 gồm bộ giải mã 261, như là bộ giải mã, để giải mã biểu diễn được mã hóa của khối của các điểm ảnh để có được tập các phần dư được lượng tử hóa và được biến đổi lỗi. Những lỗi phần dư này được giải lượng tử hóa bởi bộ lượng tử hóa nghịch đảo 262 và được biến đổi nghịch đảo bởi bộ biến đổi nghịch đảo 263 để cấp cho tập các lỗi phần dư. Các lỗi phần dư này được bổ sung bởi bộ cộng 264 cho các giá trị điểm ảnh của khối tham chiếu của các điểm ảnh. Khối tham chiếu được xác định bởi bộ ước tính chuyển động/bù chuyển động 267 hoặc bộ nội dự đoán 266, phụ thuộc vào xem liệu có hay không bộ liên dự đoán hoặc nội dự đoán được thực hiện. Bộ chọn 268 do đó được kết nối với nhau vào bộ cộng 264 và bộ ước tính chuyển động/bù chuyển động 267 và bộ nội dự đoán 266. Khối giải mã kết quả của các điểm ảnh đưa ra từ bộ cộng 264 được đưa vào bộ lọc giải khối 200. Khối được lọc của các điểm ảnh được đưa ra từ bộ giải mã 260 và có thể còn được cấp tạm thời cho bộ đệm khung 265 được sử dụng như khối tham chiếu của các điểm ảnh cho khối tiếp theo của các điểm ảnh để được giải mã. Bộ đệm khung 265 do đó được kết nối với bộ ước tính chuyển động/bù chuyển động 267 để tạo các khối được lưu trữ của các điểm ảnh sẵn có cho bộ ước tính chuyển động/bù chuyển động 267. Đầu ra từ bộ cộng 264 cũng có thể đưa vào bộ nội dự đoán 266 được sử dụng như khối tham chiếu không được lọc của các điểm ảnh.

Như được lưu ý ở trên, trong phiên bản hiện tại của đặc tả dự thảo VVC, một hoặc nhiều tham số cụ thể được báo hiệu trong lát có thể không đổi cho toàn bộ luồng bit (hoặc cho một số vị trí của luồng bit, như là CVS). Theo đó, báo hiệu cùng một giá trị cho tham số như vậy trong tất cả các lát gây ra hình phạt không cần thiết trong hiệu quả nén.

Trong quá trình phát triển VVC, phần mềm tham chiếu được gọi là VTM và tập các điều kiện kiểm tra chung (CTC, common test condition) đã được xác định để dễ dàng xác minh các cải tiến đối với bộ mã hóa-giải mã và để đảm bảo rằng bộ

mã hóa-giải mã được phát triển với phạm vi thích hợp. VVC CTC bao gồm bốn tập điều kiện mã hóa khác nhau: 1) tất cả nội bộ (AI, all intra) trong đó tất cả ảnh chỉ được mã hóa trong nội bộ, 2) truy cập ngẫu nhiên (RA, random access) với ảnh IRAP được mã hóa trong một khoảng thời gian định kỳ, 3) độ trễ thấp với các ảnh B (LDB, low-delay with B-picture) và 4) độ trễ thấp với các ảnh P (LDP, low-delay with P-picture). Mỗi tập điều kiện mã hóa bao gồm cài đặt cấu hình để đầu vào bộ mã hóa VVC và một tập các trình tự kiểm tra để sử dụng để kiểm tra bộ mã hóa-giải mã VVC.

Trong bảng 4 dưới đây, thể hiện tiêu đề các tham số lát (nghĩa là, tiêu đề lát các phần tử cú pháp) không thay đổi trên tất cả các chuỗi kiểm tra cho mỗi bốn tập điều kiện kiểm tra AI, RA, LDB và LDP trong phiên bản hiện tại của CTC sử dụng VTM 5.0.

BẢNG 4

Tên (tham số lát)	Không đổi trong			
	AI	RA	LDB	LDP
no_output_of_prior_pics_flag	x			
slice_sao_luma_flag	x			
slice_sao_chroma_flag	x			
tile_group_alf_enabled_flag	x			
dep_quant_enabled_flag	x	x	x	x
partition_constraint_override_flag	x			
slice_loop_filter_across_slices_enabled_flag	x	x	x	x
short_term_ref_pic_set_sps_flag	x		x	x
inter_ref_pic_set_prediction_flag	x			

slice_temporal_mvp_enabled_flag			x	x
mvd_l1_zero_flag			x	
collocated_from_l0_flag			x	
num_negative_pics	x			
num_positive_pics	x			
reshaper_model_delta_max_bin_idx	x			
slice_reshaper_ChromaAdj			x	x
six_minus_max_num_merge_cand	x	x	x	x
five_minus_max_num_affine_merge_cand		x	x	x
max_num_merge_cand_minus_max_num_triangle_cand		x	x	
collocated_ref_idx		x	x	x
slice_type	x			

Sự kém hiệu quả được mô tả ở trên có thể được giảm bớt bằng cách, ví dụ, 1) xác định ít nhất một tham số giá trị của nó thỉnh thoảng sẽ không đổi đối với mỗi đoạn được gồm trong tập các đoạn (ví dụ, cho mỗi lát được gồm trong CVS cụ thể) và thỉnh thoảng sẽ biến đổi và 2) gồm thông tin trong, ví dụ, tập tham số cụ thể (ví dụ, tập tham số được gồm trong CVS cụ thể) trong đó thông tin chỉ báo rằng: i) mỗi đoạn được gồm trong tập các đoạn bao gồm giá trị tham số tương ứng với tham số (ví dụ, mỗi đoạn được gồm trong tập các đoạn bao gồm từ mã từ đó giá trị tham số có thể được suy ra), ii) tập tham số bao gồm giá trị tham số tương ứng với tham số và mã giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn, iii) giá trị tham số mặc định tương ứng với tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn, hoặc iv) giá trị tham số tương ứng với tham số có thể được suy ra từ một hoặc

nhiều các giá trị tham số khác được gồm trong luồng bit và giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn. Bằng cách này, các bit có thể được tiết kiệm bằng cách báo hiệu trong tập tham số giá trị tham số mà có thể được sử dụng cho tất cả các đoạn trong tập đoạn cụ thể, hơn là báo hiệu giá trị tham số trong mỗi đoạn được gồm trong tập các đoạn.

Theo đó, theo một phương án, giá trị tham số mà luôn không đổi trong CVS là tốt nhất là được báo hiệu chỉ một lần trong CVS, thường trong tập tham số chuỗi (SPS, sequence parameter set). Các ví dụ của các tham số mà luôn không đổi trong CVS là chiều rộng và chiều cao của video - ví dụ, `pic_width_in_luma_samples` và `pic_height_in_luma_samples` trong phiên bản hiện tại của VVC. Giá trị tham số mà luôn thay đổi trong CVS với nhiều đoạn cần được báo hiệu đối với mỗi đoạn. Tham số ví dụ mà luôn thay đổi picture-by-picture trong CVS là các bit ít quan trọng nhất của giá trị đếm thứ tự ảnh, `slice_pic_order_cnt_lsb`, trong phiên bản hiện tại của VVC (khi có nhiều đoạn trong một ảnh, `slice_pic_order_cnt_lsb` là giống nhau đối với tất cả các đoạn của ảnh). Ngoài các giá trị tham số luôn không đổi và luôn thay đổi, có một số giá trị tham số thỉnh thoảng thay đổi giữa các đoạn và thỉnh thoảng không thay đổi giữa các đoạn cho toàn bộ CVS.

Theo đó, theo một phương án, cho tập các tham số mà có thể không đổi cho tất cả các đoạn trong CVS, nó được chỉ báo trong tập tham số, ví dụ SPS, xem liệu có hay không giá trị của mỗi tham số trong tập được báo hiệu trong tập tham số hoặc giá trị của tham số được báo hiệu trong tiêu đề mỗi đoạn, ví dụ, tiêu đề lát. Nếu giá trị của tham số cụ thể trong tập không đổi trong CVS, các bit có thể được tiết kiệm bằng cách báo hiệu giá trị tham số trong tập tham số. Nếu giá trị tham số không thay đổi cho tất cả các đoạn trong CVS, sau đó giá trị được báo hiệu trong tiêu đề mỗi đoạn của CVS.

Fig.3 minh họa luồng bit video ví dụ 300. Luồng bit 300 gồm CVS 301, mà bao gồm tập tham số (PS) 310 (ví dụ, đơn vị NAL không phải VCL mà chứa tập tham số) và số lượng các đoạn (ví dụ, số lượng của đơn vị VCL NAL mà chứa lát). Các đoạn 312a và 312b được thể hiện. Đoạn là đơn vị dữ liệu mà bao gồm dữ liệu đoạn (SD), mà bao gồm dữ liệu mẫu. Đoạn có thể có tiêu đề đoạn (SH) ngoài dữ

liệu đoạn (SD). Lát VVC là ví dụ về đoạn. Đoạn cũng có thể là ảnh, nhóm ô hoặc một vài thực thể khác mà bao gồm ảnh đầy đủ hoặc một phần của ảnh. Trong ví dụ này, mỗi đoạn gồm tiêu đề đoạn ngoài dữ liệu đoạn.

Giá trị tham số trong tập tham số hoặc đoạn

Theo một phương án, giá trị của tham số cụ thể (được biểu thị là “P”) có thể không đổi cho tất cả các đoạn trong CVS hoặc có thể thay đổi từ đoạn sang đoạn phụ thuộc vào tình huống. Ví dụ, tham số P có thể không đổi đối với các cấu hình hiện tại (ví dụ trong ứng dụng hội nghị video độ trễ thấp), nhưng không thay đổi trong tình huống phát quảng bá. Bộ mã hóa có thể thường được tạo cấu hình hoặc được biết trước xem liệu có hay không tham số P sẽ giữ nguyên hoặc không đổi đối với CVS.

Trong phương án này giá trị chỉ báo được báo hiệu trong tập tham số, trong đó giá trị chỉ báo định rõ xem liệu có hay không giá trị cho tham số P được báo hiệu trong tập tham số hoặc được báo hiệu đối với mỗi đoạn trong CVS. Theo một phương án, giá trị chỉ báo có thể được suy ra từ từ mã (nghĩa là, tập bit cụ thể) được gồm trong tập tham số.

Nếu giá trị chỉ báo định rõ mà giá trị tham số được báo hiệu trong tập tham số, sau đó giá trị tham số chỉ được báo hiệu trong tập tham số và mà giá trị có thể sau đó được sử dụng cho tất cả các đoạn trong CVS.

Nếu giá trị chỉ báo định rõ mà giá trị tham số được báo hiệu đối với mỗi đoạn, sau đó đối với một phiên bản cho phương án này giá trị tham số để sử dụng đối với mỗi đoạn được báo hiệu trong mỗi đoạn và không trong tập tham số.

Theo phiên bản khác cho phương án này, giá trị của tham số là tập thứ nhất trong tập tham số và sau đó nó được xác định đối với mỗi đoạn xem liệu có hay không giá trị tham số sẽ được ghi đè hoặc không.

Theo một phiên bản của phương án này, việc chỉ báo xem liệu có hay không giá trị của tham số được báo hiệu trong tập tham số hoặc trong tiêu đề mỗi đoạn, được báo hiệu tách biệt với giá trị tham số trong tập tham số (ví dụ, giá trị chỉ báo và giá trị tham số được báo hiệu với hai từ mã khác nhau, mỗi từ mã tương ứng với

phần tử cú pháp khác nhau). Điều này được minh họa trong tập tham số chuỗi (SPS, sequence parameter set) và bảng cú pháp tiêu đề đoạn và mô tả nghĩa, trong đó param_X, param_Y và param_Z là các tham số cấp độ đoạn mà có thể được báo hiệu trong SPS:

BẢNG 5 - Cú pháp SPS mẫu

seq_parameter_set_rbsp() {	Phần mô tả
...	
sps_signal_common_slice_params_in_sps_flag	u(1)
if (sps_signal_common_slice_params_in_sps_flag) {	
sps_param_X_flag	u(1)
sps_param_Y_flag	u(1)
sps_param_Z_flag	u(1)
...	
if (sps_param_X_flag)	
param_X	u(1)
if (sps_param_Y_flag)	
param_Y	u(v)
if (sps_param_Z_flag)	
param_Z	ue(v)
...	
}	

...	
}	

BẢNG 6 - Cú pháp tiêu đề đoạn mẫu

segment_header() {	Phần mô tả
...	...
if(!sps_param_X_flag)	
param_X	u(l)
if(!sps_param_Y_flag)	
param_Y	u(v)
if(!sps_param_Z_flag)	
param_Z	ue(v)
...	...
}	

BẢNG 7 - Ý nghĩa

sps_signal_common_slice_params_in_sps_flag:

Khi bằng 1 định rõ rằng param_X, param_Y, và param_Z có thể được định rõ một trong hai SPS hoặc trong các tiêu đề lát.

Khi bằng 0 định rõ rằng param_X, param_Y và param_Z được định rõ trong các tiêu đề lát và sps_param_X_flag, sps_param_Y_flag, sps_param_Z_flag được suy ra bằng 0.

sps_param_X_flag:

Khi bằng 1 định rõ rằng param_X được biểu diễn trong SPS.

Khi bằng 0 định rõ rằng param_X được biểu diễn trong các tiêu đề lát.

sps_param_Y_flag:

Khi bằng 1 định rõ rằng param_Y được biểu diễn trong SPS.

Khi bằng 0 định rõ rằng param_Y được biểu diễn trong các tiêu đề lát.

sps_param_Z_flag:

Khi bằng 1 định rõ rằng param_Z được biểu diễn trong SPS.

Khi bằng 0 định rõ rằng param_Z được biểu diễn trong các tiêu đề lát.

Theo một phiên bản của phương án này giá trị chỉ báo mà chỉ báo xem liệu có hay không giá trị nguyên không âm của tham số được báo hiệu trong tập tham số hoặc trong tiêu đề mỗi đoạn, được báo hiệu trong cùng từ mã (ký hiệu cw1) như là giá trị tham số nguyên không âm trong tập tham số, nếu giá trị chỉ báo định rõ rằng giá trị tham số nguyên không âm được báo hiệu trong tập tham số. Ví dụ, từ mã cw1 có thể nhận giá trị nguyên không âm bất kỳ với diễn giải sau:

Nếu cw1 = 0, sau đó giá trị tham số được báo hiệu trong các tiêu đề lát; nếu không, giá trị tham số bằng cw1 - 1.

Theo một cách thực hiện của phương án này, giá trị chỉ báo indicator_flag và giá trị tham số nguyên không âm param_p_value được suy ra từ từ mã thứ nhất cw1 trong tập tham số theo mà giả sau đây:

```
Indicator_flag = cw1 > 0
if ( indicator_flag ) {
    param_p_value = cw1 - 1
}
```

Mã giả sau đây thể hiện cách thức giá trị tham số được suy ra từ từ mã thứ hai cw2 trong tiêu đề đoạn nếu giá trị chỉ báo định rõ rằng giá trị tham số được báo hiệu đối với mỗi đoạn:

```

if ( !indicator_flag ) {

    param_p_value = cw2

}

```

Theo một cách thực hiện rút gọn của phiên bản của phương án này, từ mã thứ nhất `cw1` được báo hiệu trong tập tham số và giá trị tham số `param_p_value` được suy ra từ `cw1` theo mã giả sau đây, trong đó `cw1` bằng 0 chỉ báo rằng `param_p_value` được báo hiệu trong mỗi đoạn với từ mã `cw2`, và `cw1` lớn hơn 0 chỉ báo rằng `param_p_value` được báo hiệu trong tập tham số và có giá trị `cw1 - 1`:

```

if ( !cw1 ) {

    param_p_value = cw2

}

else {

    param_p_value = cw1 - 1

}

```

Lưu ý rằng trạng thái “`!cw1`” ở trên có thể được biểu thị tương đương là “`cw1>0`”.

Theo phiên bản khác cho phương án này, giá trị chỉ báo được báo hiệu trong tập tham số thứ nhất, ví dụ trong SPS. Nếu giá trị chỉ báo định rõ rằng giá trị tham số không được báo hiệu đối với mỗi đoạn, sau đó giá trị tham số được báo hiệu trong tập tham số thứ hai, ví dụ in PPS.

Tham số P có thể là tham số đoạn bất kỳ mà có thể không đối với CVS, ví dụ: bất kỳ tham số nào sau đây của phiên bản VTM 5.0 hiện tại:

```

no_output_of_prior_pics_flag,

slice_sao_luma_flag,

slice_sao_chroma_flag,

tile_group_alf_enabled_flag,

```

dep_quant_enabled_flag,
 partition_constraint_override_flag,
 slice_loop_filter_across_slices_enabled_flag,
 short_term_ref_pic_set_sps_flag,
 inter_ref_pic_set_prediction_flag,
 slice_temporal_mvp_enabled_flag,
 mvd_l1_zero_flag,
 collocated_from_l0_flag,
 num_negative_pics,
 num_positive_pics,
 reshaper_model_delta_max_bin_idx,
 slice_reshaper_ChromaAdj,
 six_minus_max_num_merge_cand,
 five_minus_max_num_affine_merge_cand,
 max_num_merge_cand_minus_max_num_triangle_cand,
 collocated_ref_idx
 slice_type,
 log2_diff_min_qt_min_cb,
 max_mtt_hierarchy_depth,
 log2_diff_max_bt_min_qt, or
 log2_diff_max_tt_min_qt.

Khi tham số P đã được thảo luận cho đến nay, có thể có nhiều hơn một tham số tại một thời điểm mà thỉnh thoảng không thay đổi trong CVS. Chúng ta sẽ gọi các tham số này là các tham số p1, p2, ... pN trong cú pháp mẫu dưới đây.

BẢNG 8 - Cú pháp SPS mẫu

seq_parameter_set_rbsp() {	Phần mô tả
...	...
sps_param_p1	ue(v)
sps_param_p2	ue(v)
...	...
sps_param_pN	ue(v)
...	...
}	

sps_param_p1 bằng 0 định rõ rằng param_p1 được biểu diễn trong các tiêu đề lát. sps_param_p1 lớn hơn 0 định rõ rằng giá trị của param_p1 được đặt bằng sps_param_p1 trừ đi 1.

sps_param_p2 bằng 0 định rõ rằng param_p1 được biểu diễn trong các tiêu đề lát. sps_param_p2 lớn hơn 0 định rõ rằng giá trị của param_p2 được đặt bằng sps_param_p2 trừ đi 1.

sps_param_pN bằng 0 định rõ rằng param_pN được biểu diễn trong các tiêu đề lát. sps_param_pN lớn hơn 0 định rõ rằng giá trị của param_pN được đặt bằng sps_param_pN trừ đi 1.

BẢNG 9 - Cú pháp tiêu đề lát mẫu

slice_header() {	Phần mô tả
...	...

if (!sps_param_p1)	
param_p1	ue(v)
if (!sps_param_p2)	
param_p2	ue(v)
...	...
if (!sps_param_pN)	
param_pN	ue(v)
...	...
}	

Theo một phương án cờ nhóm được sử dụng sao cho nếu cờ nhóm được đặt là 1, tất cả các tham số trong nhóm tham số có thể được báo hiệu trong tập tham số hoặc trong tiêu đề đoạn. Nếu cờ nhóm được đặt là 0, các tham số trong nhóm tham số luôn được báo hiệu trong tiêu đề lát. Điều này được minh họa trong cú pháp và ngữ nghĩa dưới đây.

BẢNG 10 - Cú pháp SPS mẫu minh họa cờ nhóm

seq_parameter_set_rbsp() {	Phần mô tả
...	...
grouping_flag	u(1)
if (grouping_flag) {	
sps_param_p1	ue(v)
sps_param_p2	ue(v)

...	...
sps_param_pN	uc(v)
}	
...	...
}	

grouping_flag bằng 1 định rõ rằng param_p1, param_p2, ..., param_pN có thể được định rõ một trong hai SPS hoặc trong các tiêu đề lát. **grouping_flag** bằng 0 định rõ rằng param_p1, param_p2, ..., param_pN được định rõ trong các tiêu đề lát.

sps_param_p1 bằng 0 định rõ rằng param_p1 được biểu diễn trong các tiêu đề lát. **sps_param_p1** lớn hơn 0 định rõ rằng giá trị của param_p1 được đặt bằng **sps_param_p1** trừ đi 1.

sps_param_p2 bằng 0 định rõ rằng param_p2 được biểu diễn trong các tiêu đề lát. **sps_param_p2** lớn hơn 0 định rõ rằng giá trị của param_p2 được đặt bằng **sps_param_p2** trừ đi 1.

sps_param_pN bằng 0 định rõ rằng param_pN được biểu diễn trong các tiêu đề lát. **sps_param_pN** lớn hơn 0 định rõ rằng giá trị của param_pN được đặt bằng **sps_param_pN** trừ đi 1.

BẢNG 11 - Tiêu đề đoạn mẫu minh họa cờ nhóm

segment_header() {	Phần mô tả
...	...
if(grouping_flag) {	
if(!sps_param_p1)	

param_p1	uc(v)
if (!sps_param_p2)	
param_p2	uc(v)
...	...
if (!sps_param_pN)	
param_pN	uc(v)
} else {	
param_p1	uc(v)
param_p2	uc(v)
...	
param_pN	uc(v)
}	
...	
}	

Theo một phương án cờ nhóm bằng 1 định rõ rằng các tham số trong nhóm tham số sử dụng các giá trị mặc định của chúng và cờ nhóm bằng 0 định rõ rằng các tham số trong nhóm tham số được báo hiệu trong tiêu đề đoạn. Do đó, trong phiên bản này của phương án các tham số không được báo hiệu trong tập tham số, do các giá trị mặc định được sử dụng thay thế.

Theo một phương án cờ nhóm được suy ra từ giá trị tham số trong luồng bit. Ví dụ, lựa chọn báo hiệu các giá trị tham số trong tập tham số hoặc tiêu đề đoạn có thể chỉ có sẵn cho hình ảnh không phải IRAP.

Theo một phương án, bộ mã hóa có thể thực hiện tập con hoặc tất cả các bước sau để mã hóa một hoặc nhiều ảnh vào trong luồng video được mã hóa (CVS, coded video sequence), trong đó CVS sẽ bao gồm ít nhất tập tham số thứ nhất và tập hai hoặc nhiều đoạn. Tập tham số thứ nhất ví dụ có thể là SPS, PPS, DPS, VPS hoặc tiêu đề ảnh tập tham số (PHPS).

Bước 1: Xác định tham số P nếu giá trị của tham số P không thay đổi cho tất cả các đoạn trong CVS.

Bước 2: Xây dựng tập tham số thứ nhất như là tập tham số cung cấp thông tin mà định rõ xem liệu có hay không giá trị của tham số P được báo hiệu trong tập tham số (ví dụ, tập tham số thứ nhất hoặc tập tham số khác). Ví dụ, nếu nó được xác định rằng giá trị của tham số P không thay đổi cho tất cả các đoạn trong CVS, sau đó thông tin sẽ chỉ rõ rằng giá trị của tham số P được báo hiệu trong tập tham số, nếu không thì thông tin sẽ chỉ rõ rằng giá trị của tham số P được báo hiệu đối với mỗi đoạn trong CVS. Nếu giá trị của tham số P được báo hiệu trong tập tham số, giá trị của tham số P có thể được mã hóa trong từ mã cũng cung cấp thông tin về việc xem liệu có hay không giá trị của tham số P được báo hiệu trong tập tham số.

Bước 3: Đối với mỗi đoạn trong CVS, nếu nó được xác định rằng giá trị của tham số P sẽ không thay đổi cho tất cả các đoạn trong CVS, sau đó gồm giá trị của tham số P trong đoạn. Giá trị tốt nhất là được gồm trong tiêu đề đoạn (ví dụ, được mã hóa trong từ mã trong tiêu đề đoạn).

Bước 4: Đối với mỗi đoạn, nếu giá trị tham số cho tham số P được gồm trong đoạn, sau đó dữ liệu đoạn mã hóa của đoạn sử dụng giá trị của tham số P mà được gồm trong đoạn, nếu không thì dữ liệu đoạn mã hóa của đoạn sử dụng giá trị của tham số P mà được báo hiệu bởi tập tham số.

Theo một phương án, bộ giải mã có thể thực hiện tập con hoặc tất cả các bước sau để giải mã một hoặc nhiều ảnh từ luồng video được mã hóa (CVS, coded video sequence), trong đó CVS bao gồm ít nhất tập tham số thứ nhất và hai hoặc nhiều đoạn.

Bước 1: Giải mã giá trị chỉ báo từ từ mã thứ nhất trong tập tham số thứ nhất trong CVS. Giá trị chỉ báo có thể là giá trị 1-bit (ví dụ, từ mã có thể gồm có bit đơn).

Bước 2: Xác định từ giá trị chỉ báo xem liệu có hay không giá trị của tham số P được chỉ rõ trong tập tham số hoặc được chỉ rõ bởi từ mã thứ hai trong mỗi đoạn. Sự hiện diện của từ mã thứ hai trong đoạn có thể được phụ thuộc vào giá trị chỉ báo.

Bước 3: Nếu nó được xác định rằng giá trị của tham số P được chỉ rõ trong tập tham số, sau đó giải mã giá trị của tham số P từ tập tham số. Giá trị của tham số P ví dụ có thể là được suy ra từ từ mã thứ nhất hoặc được giải mã từ từ mã thứ ba trong tập tham số.

Bước 4: Đối với mỗi đoạn, thực hiện các bước 4A và 4B.

Bước 4A: Nếu nó được xác định rằng giá trị của tham số P được chỉ rõ trong mỗi đoạn, sau đó giải mã giá trị của tham số P từ từ mã thứ hai trong đoạn. Từ mã thứ hai tốt nhất là được giải mã từ tiêu đề đoạn.

Bước 4B: Giải mã dữ liệu đoạn của đoạn sử dụng giá trị của tham số P.

Trong một biến thể của phương án này giá trị chỉ báo định rõ rằng giá trị của tham số P được đặt là giá trị mặc định đã biết bộ giải mã hoặc giá trị của tham số P được báo hiệu đối với mỗi đoạn trong CVS.

Các trường hợp trong đó giá trị tham số là giá trị 1-bit

Trong một số trường hợp sử dụng giá trị tham số cho tham số P là giá trị boolean, thay vì giá trị chung hoặc giá trị nguyên không âm. Đối với các trường hợp sử dụng như vậy tham số P được gọi là “cờ F”.

Theo một phương án, việc chỉ báo xem liệu có hay không cờ F được báo hiệu trong tập tham số hoặc trong tiêu đề mỗi đoạn được báo hiệu riêng từ giá trị của cờ F trong tập tham số, ví dụ, giá trị chỉ báo và giá trị tham số được báo hiệu với hai từ mã khác nhau.

Theo phương án thay thế, nếu giá trị chỉ báo định rõ rằng giá trị của cờ F được

báo hiệu trong tập tham số, sau đó giá trị chỉ báo được báo hiệu trong cùng từ mã như giá trị của cờ F.

Ví dụ, từ mã có thể là từ mã 2-bit (ký hiệu *cw*) mà có thể đảm nhận ít nhất ba giá trị khác nhau, 0, 1 và 2 với diễn giải sau:

cw = 0: Giá trị của cờ F được báo hiệu trong các tiêu đề đoạn;

cw = 1: Giá trị của cờ F là 0; và

cw > 1 (ví dụ, *cw*=2): Giá trị của cờ F là 1.

Theo phương án thay thế, các giá trị của *cw* có cách giải thích sau:

cw = 0: Giá trị của cờ F là 0;

cw = 1: Giá trị của cờ F là 1; và

cw > 1 (ví dụ, *cw*=2): Giá trị của cờ F được báo hiệu trong các tiêu đề đoạn.

Theo một phương án, từ mã thứ nhất *cw1* (ví dụ, từ mã dài 2-bit) được báo hiệu trong tập tham số và giá trị của cờ F (ký hiệu “*flag_f_value*”) được suy ra từ *cw1*, trong đó giá trị của *cw1* bằng 0 định rõ rằng *flag_f_value* được báo hiệu trong từ mã (ký hiệu *cw2*) trong mỗi đoạn, và giá trị của *cw1* bằng 1 hoặc 2 định rõ rằng *flag_f_value* = giá trị của (*cw1*-1). Điều này được minh họa trong mã giả sau đây:

```
if ( cw1 == 0 ) {
    flag_f_value = cw2
} else {
    flag_f_value = cw1-1
}
```

Theo phương án khác, giá trị chỉ báo, *indicator_flag*, và giá trị của cờ F, *flag_f_value*, được suy ra từ từ mã thứ nhất *cw1* trong tập tham số theo mã giả sau đây:

```
Nếu cw1 == 0, sau đó indicator_flag = 0, nếu không thì indicator_flag = 1;
if ( indicator_flag ) {
```

```

    flag_f_value = cw1-1
}

```

Mã giả sau đây thể hiện cách giá trị của cờ F được suy ra từ từ mã thứ hai (hoặc cờ) cw2 trong tiêu đề đoạn nếu giá trị chỉ báo định rõ rằng giá trị của cờ F được báo hiệu đối với mỗi đoạn:

```

if ( !indicator_flag ) {
    flag_f_value = cw2
}

```

Theo một phương án, giá trị chỉ báo cờ chỉ báo và giá trị của cờ F flag_f_value có thể được suy ra từ ba giá trị của từ mã thứ nhất cw1, theo ví dụ:

BẢNG 12 - Cờ chỉ báo và flag_f_value được suy ra từ ba giá trị của từ mã cw1

cw1	indicator_flag	flag_f_value
0	0	Giá trị được chỉ rõ trong tiêu đề mỗi đoạn
10	1	flag_f_value = 0
11	1	flag_f_value = 1

Từ mã cw1 được báo hiệu với phần mô tả mã hóa cố định như là u(2), nghĩa là hai bit luôn được báo hiệu. Từ mã cw1 có thể cũng được báo hiệu với phần mô tả nhị phân bậc một rút gọn với giá trị lớn nhất (maxVal) được đặt là 2, mà chỉ có thể báo hiệu ba giá trị, ví dụ (0, 10 hoặc 11). Phần mô tả nhị phân bậc một rút gọn có thể, như trong đặc tả VVC dự thảo, được chỉ báo bởi tu(v) trong bảng cú pháp:

BẢNG 13 - Ví dụ về phần mô tả mã hóa bậc ba

seq_parameter_set_rbsp() {	Phần mô tả
...	...
sps_or_slice_f1_flag	tu(v)
...	...
}	

Một lợi ích của việc mã hóa cwl với 0, 10 hoặc 11 so với việc sử dụng hai bit cố định là bất cứ khi nào flag_f_value được chỉ định trong tiêu đề đoạn, chỉ cần một bit được báo hiệu.

Các giá trị từ mã khác hoặc sự kết hợp của ba giá trị (0, 10, 11), có thể cũng được xem xét, ví dụ (1, 01, 00) hoặc (10, 11, 0).

Cú pháp và ngữ nghĩa mẫu dưới đây minh họa cách các cờ f1, f2,... fN có thể được định rõ.

BẢNG 14

seq_parameter_set_rbsp() {	Phần mô tả
...	...
sps_flag_f1	tu(v)
sps_flag_f2	tu(v)
...	...
sps_flag_fN	tu(v)
...	...

}	
---	--

Theo một phương án, bộ mã hóa có thể thực hiện tập con hoặc tất cả các bước sau để mã hóa một hoặc nhiều ảnh trong luồng video được mã hóa (CVS, coded video sequence), trong đó CVS sẽ bao gồm ít nhất tập tham số và hai hoặc nhiều đoạn.

Bước 1: Xác định đề cờ F nếu giá trị của cờ F sẽ không thay đổi cho tất cả các đoạn trong CVS.

Bước 2: Báo hiệu giá trị chỉ báo trong từ mã thứ nhất trong tập tham số, trong đó, nếu nó được xác định rằng giá trị của cờ F sẽ không thay đổi cho tất cả các đoạn trong CVS, sau đó giá trị chỉ báo định rõ rằng giá trị của cờ F được báo hiệu trong tập tham số (hoặc tập tham số khác), nếu không thì giá trị chỉ báo định rõ rằng giá trị của cờ F được báo hiệu đối với mỗi đoạn trong CVS.

Nếu giá trị chỉ báo định rõ rằng giá trị của cờ F được báo hiệu trong tập tham số, sau đó bước 3A được thực hiện hoặc bước 3B được thực hiện.

Bước 3A: a) Báo hiệu giá trị của cờ F trong tập tham số (giá trị của cờ F ví dụ có thể là được báo hiệu trong từ mã thứ nhất cùng với giá trị chỉ báo hoặc trong từ mã thứ ba trong tập tham số) và b) đối với mỗi đoạn, dữ liệu đoạn mã hóa của đoạn sử dụng giá trị của cờ F.

Bước 3B: Đối với mỗi đoạn, a) báo hiệu giá trị của cờ F trong từ mã thứ hai trong đoạn (từ mã thứ hai tốt nhất là được báo hiệu trong tiêu đề đoạn) và b) dữ liệu đoạn mã hóa của đoạn sử dụng giá trị của cờ F cho đoạn.

Bộ giải mã có thể thực hiện tập con hoặc tất cả các bước sau cho phương án này để giải mã một hoặc nhiều ảnh từ luồng video được mã hóa (CVS, coded video sequence), trong đó CVS bao gồm ít nhất một tập tham số và hai hoặc nhiều đoạn.

Bước 1: Giải mã giá trị chỉ báo từ từ mã thứ nhất trong tập tham số trong CVS. Tập tham số ví dụ có thể là SPS, PPS, DPS, VPS hoặc PHPS. Giá trị chỉ báo có thể là giá trị 1-bit.

Bước 2: Xác định từ giá trị chỉ báo xem liệu có hay không giá trị của cờ F được chỉ rõ trong tập tham số hoặc được chỉ rõ bởi từ mã thứ hai trong mỗi đoạn mã hóa.

Bước 3: Nếu được xác định rằng giá trị của cờ F được chỉ rõ trong tập tham số, sau đó giải mã giá trị của cờ F từ tập tham số. Giá trị của cờ F ví dụ có thể là được suy ra từ từ mã thứ nhất hoặc được giải mã từ từ mã thứ ba riêng biệt trong tập tham số.

Bước 4: Đối với mỗi đoạn thực hiện các bước 4A và 4B.

Bước 4A: Nếu được xác định rằng giá trị của cờ F được chỉ rõ trong mỗi đoạn, sau đó giải mã giá trị của cờ F từ từ mã thứ hai trong đoạn (từ mã thứ hai có thể là một phần của tiêu đề đoạn).

Bước 4B: Giải mã dữ liệu đoạn của đoạn sử dụng giá trị của cờ F, mà hoặc được xác định từ từ mã thứ hai hoặc từ tập tham số.

Được suy ra giá trị tham số

Theo một phương án, giá trị của tham số P đối với mỗi đoạn trong CVS có thể được suy ra từ một hoặc nhiều các giá trị tham số khác trong tiêu đề và/hoặc trong tập tham số trong CVS và giá trị chỉ báo được báo hiệu (ví dụ trong tập tham số trong CVS) mà định rõ xem liệu có hay không giá trị của tham số P mà được suy ra từ một hoặc nhiều giá trị tham số khác.

Theo phương án khác, giá trị của tham số P đối với mỗi đoạn trong CVS có thể là: i) được suy ra từ một hoặc nhiều giá trị tham số khác trong tiêu đề và/hoặc trong tập tham số trong CVS hoặc ii) được báo hiệu đối với mỗi đoạn, và giá trị chỉ báo được báo hiệu (ví dụ trong tập tham số trong CVS) mà định rõ một trong: i) là giá trị của tham số P mà được suy ra hoặc ii) là giá trị của tham số P được báo hiệu in mỗi đoạn.

Theo phương án khác, giá trị của tham số P đối với mỗi đoạn trong CVS may be: i) được suy ra từ một hoặc nhiều giá trị tham số khác trong tiêu đề và/hoặc trong tập tham số trong CVS, ii) được báo hiệu đối với mỗi đoạn, hoặc iii) được gồm trong tập tham số trong CVS, và giá trị chỉ báo được báo hiệu (ví dụ trong tập tham

số trong CVS) mà định rõ một trong: i) giá trị của tham số P nên được suy ra, ii) giá trị của tham số P được báo hiệu trong mỗi đoạn, hoặc iii) giá trị của tham số P được gồm trong tập tham số trong CVS.

Theo phương án khác, giá trị của tham số P đối với mỗi đoạn trong CVS có thể: i) được suy ra từ một hoặc nhiều giá trị tham số khác trong tiêu đề và/hoặc trong tập tham số trong CVS hoặc ii) được gồm trong tập tham số trong CVS, và giá trị chỉ báo được báo hiệu (ví dụ trong tập tham số trong CVS) mà định rõ một trong các: i) giá trị của tham số P được suy ra hoặc ii) giá trị của tham số P được gồm trong tập tham số trong CVS.

Theo phương án khác, tập tham số (ví dụ, SPS) cung cấp thông tin chỉ báo một trong các: i) giá trị của tham số P đối với mỗi đoạn được suy ra từ tham số khác đối với mỗi đoạn, ii) giá trị của tham số P được cố định cho tất cả các đoạn (ví dụ được cố định là giá trị mặc định hoặc giá trị được báo hiệu), hoặc iii) mà giá trị của tham số P được báo hiệu đối với mỗi đoạn.

Ví dụ, trong kịch bản mã hóa toàn bộ hoặc trong một kịch bản hội nghị truyền hình có độ trễ thấp, tham số `slice_type` có thể được suy ra từ loại ảnh. Theo đó, giá trị cho tham số `slice_type` cần không được báo hiệu cho mỗi lát (nghĩa là, mỗi lát không cần phải chứa giá trị cho tham số `slice_type` khi bộ giải mã có thể, đối với mỗi lát, suy ra giá trị tham số `slice_type` đúng).

Theo một phương án, bộ mã hóa có thể thực hiện tập con hoặc tất cả các bước dưới đây để mã hóa một hoặc nhiều ảnh trong luồng video được mã hóa (CVS, coded video sequence), trong đó CVS sẽ bao gồm ít nhất một tập tham số và hai hoặc nhiều đoạn.

Bước 1: Đối với tham số P, xác định xem liệu có hay không: i) cho tất cả các đoạn trong CVS giá trị của tham số P có thể được suy ra dựa trên tham số Q khác (ví dụ, được suy ra từ Q và 0 hoặc nhiều các tham số khác), ii) giá trị của tham số P sẽ không thay đổi cho tất cả các đoạn trong CVS, hoặc iii) giá trị của tham số P sẽ không thay đổi cho tất cả các đoạn trong CVS và, cho tất cả các đoạn trong CVS, không thể được suy ra từ một hoặc nhiều các tham số khác.

Bước 2: Gồm từ mã thứ nhất (cw1) trong tập tham số, trong đó giá trị của cw1 (còn được biết đến như “giá trị chỉ báo”) phụ thuộc vào giá trị nào được xác định trong bước 1. Nghĩa là, i) nếu được xác định rằng cho tất cả các đoạn trong CVS giá trị của tham số P có thể được suy ra dựa trên tham số Q khác, sau đó cw1 định rõ rằng, cho tất cả các đoạn trong CVS, giá trị của tham số P có thể được suy ra dựa trên tham số Q; ii) nếu không thì, nếu được xác định rằng giá trị của tham số P sẽ không thay đổi cho tất cả các đoạn trong CVS, sau đó cw1 định rõ rằng giá trị của tham số P được báo hiệu trong tập tham số; iii) nếu không thì, cw1 định rõ rằng giá trị của tham số P được báo hiệu đối với mỗi đoạn trong CVS.

Bước 3: Nếu cw1 định rõ rằng giá trị của tham số P được báo hiệu trong tập tham số, sau đó báo hiệu giá trị của tham số P trong tập tham số. Giá trị của tham số P ví dụ có thể là được báo hiệu trong từ mã thứ nhất cùng với giá trị chỉ báo hoặc trong từ mã thứ ba trong tập tham số.

Bước 4: Đối với mỗi đoạn thực hiện các bước 4A và 4B.

Bước 4A: Nếu cw1 định rõ rằng giá trị của tham số P được báo hiệu trong mỗi đoạn, sau đó báo hiệu giá trị của tham số P trong từ mã thứ hai trong đoạn. Từ mã thứ hai là tốt nhất là được báo hiệu trong tiêu đề đoạn.

Bước 4B: Dữ liệu đoạn mã hóa của đoạn sử dụng giá trị của tham số P.

Bộ giải mã có thể thực hiện tập con hoặc tất cả các bước sau để giải mã một hoặc nhiều ảnh từ luồng video được mã hóa (CVS, coded video sequence), trong đó CVS bao gồm ít nhất một tập tham số và hai hoặc nhiều đoạn.

Bước 1: Giải mã giá trị chỉ báo từ từ mã thứ nhất trong tập tham số trong CVS. Tập tham số ví dụ có thể là SPS, PPS, DPS, VPS hoặc PHPS.

Bước 2: Xác định từ giá trị chỉ báo xem liệu có hay không: i) giá trị của tham số P nên được suy ra được dựa trên tham số Q, ii) giá trị của tham số P được chỉ rõ trong tập tham số, hoặc iii) giá trị của tham số P được chỉ rõ bởi từ mã thứ hai trong mỗi đoạn của CVS.

Bước 3: Nếu được xác định rằng giá trị của tham số P được chỉ rõ trong tập tham số, sau đó giải mã giá trị của tham số P từ tập tham số. Giá trị của tham số P

ví dụ có thể là được suy ra từ từ mã thứ nhất hoặc được giải mã từ từ mã thứ ba trong tập tham số.

Bước 4: Nếu được xác định rằng giá trị của tham số P nên được suy ra được dựa trên tham số Q, sau đó xác định giá trị tham số cho tham số Q và sau đó suy ra giá trị của tham số P được dựa trên giá trị tham số đối với tham số Q. Ví dụ, $P_v = f(Q_v, \dots)$, trong đó P_v là giá trị tham số cho tham số P, $f()$ là hàm, và Q_v là giá trị tham số cho tham số Q. Nghĩa là, P_v có thể là hàm của ít nhất Q_v .

Bước 5: Đối với mỗi đoạn thực hiện các bước 5A và 5B.

Bước 5A: Nếu được xác định rằng giá trị của tham số P được chỉ rõ trong mỗi đoạn, sau đó giải mã giá trị của tham số P từ từ mã thứ hai trong đoạn.

Bước 5B: Giải mã dữ liệu đoạn của đoạn sử dụng giá trị của tham số P.

Các ví dụ

Dưới đây là một ví dụ cụ thể về cú pháp và ngữ nghĩa cho SPS và tiêu đề lát trên đầu đặc tả dự thảo hiện tại VVC. Lưu ý rằng không phải tất cả các tham số được tìm thấy là tính trong phiên bản hiện tại của phần mềm VTM (5.0) được biểu diễn trong phiên bản hiện tại của đặc tả dự thảo VVC (JVET-N1001-v7). Một số tham số cũng có tên khác nhau trong phần mềm và đặc tả mặc dù cách diễn giải của chúng là tương đương. Ví dụ: `five_minus_max_num_affine_merge_cand` trong phần mềm VTM 5.0 tương đương với `five_minus_max_num_subblock_merge_cand` trong đặc tả dự thảo VVC hiện tại. Điều này dự kiến sẽ được căn chỉnh khi thông số kỹ thuật VVC đã được hoàn thiện.

BẢNG 16

<code>seq_parameter_set_rbsp() {</code>	Phần mô tả
<code>sps_decoding_parameter_set_id</code>	u(4)
...	

sps_or_slice_flag	u(1)
if (sps_or_slice_flag) {	
sps_or_slice_dep_quant_enabled_flag	tu(v)
For (i=0, i<2;i++)	
sps_or_slice_ref_pic_list_sps_flag[i]	tu(v)
sps_or_slice_temporal_mvp_enabled_flag	tu(v)
sps_or_slice_mvd_l1_zero_flag	tu(v)
sps_or_slice_collocated_from_l0_flag	tu(v)
sps_or_slice_chroma_residual_scale_flag	tu(v)
sps_or_slice_six_minus_max_num_merge_cand	ue(v)
sps_or_slice_five_minus_max_num_subblock_merge_cand	ue(v)
sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand	ue(v)
}	
...	
}	

Các ngữ nghĩa SPS:

sps_or_slice_flag bằng 0 định rõ rằng các giá trị của
sps_or_slice_dep_quant_enabled_flag, sps_or_slice_ref_pic_list_sps_flag,
sps_or_slice_temporal_mvp_enabled_flag, sps_or_slice_mvd_l1_zero_flag,
sps_or_slice_collocated_from_l0_flag,

`sps_or_slice_six_minus_max_num_merge_cand`,
`sps_or_slice_five_minus_max_num_subblock_merge_cand` và
`sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` được suy ra bằng 0. `sps_or_slice_flag` bằng 1 định rõ rằng các phần tử cú pháp `sps_or_slice_dep_quant_enabled_flag`, `sps_or_slice_ref_pic_list_sps_flag`, `sps_or_slice_temporal_mvp_enabled_flag`, `sps_or_slice_mvd_l1_zero_flag`, `sps_or_slice_collocated_from_l0_flag`, `sps_or_slice_six_minus_max_num_merge_cand`, `sps_or_slice_five_minus_max_num_subblock_merge_cand` và `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` là có mặt trong SPS.

`sps_or_slice_dep_quant_enabled_flag` bằng 0 định rõ rằng `dep_quant_enabled_flag` được biểu diễn trong các tiêu đề lát. `sps_or_slice_dep_quant_enabled_flag` lớn hơn 0 định rõ rằng `dep_quant_enabled_flag` được suy ra bằng `sps_or_slice_dep_quant_enabled_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của tu(v) nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_ref_pic_list_sps_flag[i]` bằng 0 định rõ rằng `ref_pic_list_sps[i]` được biểu diễn trong các tiêu đề lát. `sps_or_slice_ref_pic_list_sps_flag[i]` lớn hơn 0 định rõ rằng `ref_pic_list_sps[i]` được suy ra bằng `sps_or_slice_ref_pic_list_sps_flag[i]` trừ đi 1. Giá trị lớn nhất `maxVal` của tu(v) nhị phân bậc một rút gọn được đặt bằng 2. [lưu ý: tham số chung cho tất cả `ref_pic_list_sps[i]` là điều có thể.]

`sps_or_slice_temporal_mvp_enabled_flag` bằng 0 định rõ rằng `slice_temporal_mvp_enabled_flag` được biểu diễn trong các tiêu đề lát của các lát với `slice_type` không bằng I trong CVS. `sps_or_slice_temporal_mvp_enabled_flag` lớn hơn 0 định rõ rằng `slice_temporal_mvp_enabled_flag` được suy ra bằng `sps_or_slice_temporal_mvp_enabled_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của tu(v) nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_mvd_l1_zero_flag` bằng 0 định rõ rằng `mvd_l1_zero_flag` được

biểu diễn trong các tiêu đề lát. `sps_or_slice_mvd_11_zero_flag` lớn hơn 0 định rõ rằng `mvd_11_zero_flag` được suy ra bằng `sps_or_slice_mvd_11_zero_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_collocated_from_10_flag` bằng 0 định rõ rằng `collocated_from_10_flag` được biểu diễn trong các tiêu đề lát. `sps_or_slice_collocated_from_10_flag` lớn hơn 0 định rõ rằng `collocated_from_10_flag` được suy ra bằng `sps_or_slice_collocated_from_10_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_chroma_residual_scale_flag` bằng 0 định rõ rằng `slice_chroma_residual_scale_flag` được biểu diễn trong các tiêu đề lát. `sps_or_slice_chroma_residual_scale_flag` lớn hơn 0 định rõ rằng `slice_chroma_residual_scale_flag` được suy ra bằng `sps_or_slice_chroma_residual_scale_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_six_minus_max_num_merge_cand` bằng 0 định rõ rằng `six_minus_max_num_merge_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_six_minus_max_num_merge_cand` lớn hơn 0 định rõ rằng `six_minus_max_num_merge_cand` được suy ra bằng `sps_or_slice_six_minus_max_num_merge_cand` trừ đi 1.

`sps_or_slice_five_minus_max_num_subblock_merge_cand` bằng 0 định rõ rằng `five_minus_max_num_subblock_merge_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_five_minus_max_num_subblock_merge_cand` lớn hơn 0 định rõ rằng `five_minus_max_num_subblock_merge_cand` được suy ra bằng `sps_or_slice_five_minus_max_num_subblock_merge_cand` trừ đi 1.

`sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` bằng 0 định rõ rằng `max_num_merge_cand_minus_max_num_triangle_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` lớn hơn 0 định rõ rằng `max_num_merge_cand_minus_max_num_triangle_cand` được suy ra

bằng `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand_minus`.

BẢNG 17

<code>slice_header() {</code>	Phần mô tả
<code>slice_pic_parameter_set_id</code>	<code>ue(v)</code>
<code>if(rect_slice_flag NumBricksInPic > 1)</code>	
<code>slice_address</code>	<code>u(v)</code>
<code>if(!rect_slice_flag && !single_brick_per_slice_flag)</code>	
<code>num_bricks_in_slice_minus1</code>	<code>ue(v)</code>
<code>slice_type</code>	<code>ue(v)</code>
<code>if(NalUnitType == GRA_NUT)</code>	
<code>recovery_poc_cnt</code>	<code>sc(v)</code>
<code>slice_pic_order_cnt_lsb</code>	<code>u(v)</code>
<code>if(NalUnitType == IDR_W_RADL NalUnitType == IDR_N_LP NalUnitType == CRA_NUT)</code>	
<code>no_output_of_prior_pics_flag</code>	<code>u(1)</code>
<code>if(output_flag_present_flag)</code>	
<code>pic_output_flag</code>	<code>u(1)</code>
<code>if((NalUnitType != IDR_W_RADL && NalUnitType != IDR_N_LP) sps_idr_rpl_present_flag) {</code>	

for(i = 0; i < 2; i++) {	
if(num_ref_pic_lists_in_sps[i] > 0 && !sps_or_slice_ref_pic_list_sps_flag[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)
}	
}	

<pre> if((slice_type != I && num_ref_entries[0][RplIdx[0]] > 1) (slice_type == B && num_ref_entries[1][RplIdx[1]] > 1)) { </pre>	
num_ref_idx_active_override_flag	u(i)
<pre>if(num_ref_idx_active_override_flag)</pre>	
<pre>for(i = 0; i < (slice_type == B ? 2 : 1); i++)</pre>	
<pre>if(num_ref_entries[i][RplIdx[i]] > 1)</pre>	
num_ref_idx_active_minus1[i]	ue(v)
<pre> } </pre>	
<pre> } </pre>	
<pre>if(partition_constraints_override_enabled_flag) {</pre>	
partition_constraints_override_flag	ue(v)
<pre>if(partition_constraints_override_flag) {</pre>	
slice_log2_diff_min_qt_min_cb_luma	ue(v)
slice_max_mtt_hierarchy_depth_luma	ue(v)
<pre>if(slice_max_mtt_hierarchy_depth_luma != 0)</pre>	
slice_log2_diff_max_bt_min_qt_luma	ue(v)
slice_log2_diff_max_tt_min_qt_luma	ue(v)
<pre> } </pre>	
<pre>if(slice_type == I && qtbtt_dual_tree_intra_flag) {</pre>	

slice_log2_diff_min_qt_min_cb_chroma	ue(v)
slice_max_mtt_hierarchy_depth_chroma	ue(v)
if(slice_max_mtt_hierarchy_depth_chroma != 0)	
slice_log2_diff_max_bt_min_qt_chroma	ue(v)
slice_log2_diff_max_tt_min_qt_chroma	ue(v)
}	
}	
}	
}	
if(slice_type != I) {	
if(!sps_or_slice_temporal_mvp_enabled_flag)	
slice_temporal_mvp_enabled_flag	u(1)
if(slice_type == B && !sps_or_slice_mvd_l1_zero_flag)	
mvd_l1_zero_flag	u(1)
if(cabac_init_present_flag)	
cabac_init_flag	u(1)
if(slice_temporal_mvp_enabled_flag) {	
if(slice_type == B && !sps_or_slice_collocated_from_l0_flag)	
collocated_from_l0_flag	u(1)
}	

if((weighted_pred_flag && slice_type == P) (weighted_bipred_flag && slice_type == B))	
pred_weight_table()	
if(!sps_or_slice_six_minus_max_num_merge_cand)	
six_minus_max_num_merge_cand	ue(v)
if(sps_affine_enabled_flag && !sps_or_slice_five_minus_max_num_subblock_merge_cand)	
five_minus_max_num_subblock_merge_cand	ue(v)
if(sps_fpel_mmvd_enabled_flag)	
slice_fpel_mmvd_enabled_flag	u(1)
if(sps_triangle_enabled_flag && MaxNumMergeCand >= 2 && !sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand)	
max_num_merge_cand_minus_max_num_triangle_cand	ue(v)
else if (sps_ibc_enabled_flag && !sps_or_slice_six_minus_max_num_merge_cand)	
six_minus_max_num_merge_cand	ue(v)
slice_qp_delta	se(v)
if(pps_slice_chroma_qp_offsets_present_flag) {	
slice_cb_qp_offset	se(v)
slice_cr_qp_offset	se(v)
slice_joint_cbr_qp_offset	se(v)

}	
if(sps_sao_enabled_flag) {	
slice_sao_luma_flag	u(1)
if(ChromaArrayType != 0)	
slice_sao_chroma_flag	u(1)
}	
if(sps_alf_enabled_flag) {	
slice_alf_enabled_flag	u(1)
if(slice_alf_enabled_flag) {	
slice_num_alf_aps_ids_luma	tb(v)
for(i = 0; i < slice_num_alf_aps_ids_luma; i++)	
slice_alf_aps_id_luma[i]	u(5)
slice_alf_chroma_idc	tu(v)
if(slice_alf_chroma_idc && (slice_type != I slice_num_alf_aps_ids_luma != 1))	
slice_alf_aps_id_chroma	u(5)
}	
}	
if(!sps_or_slice_dep_quant_enabled_flag)	
dep_quant_enabled_flag	u(1)

if(!dep_quant_enabled_flag)	
sign_data_hiding_enabled_flag	u(1)
if(deblocking_filter_override_enabled_flag)	
deblocking_filter_override_flag	u(1)
if(deblocking_filter_override_flag) {	
slice_deblocking_filter_disabled_flag	u(1)
if(!slice_deblocking_filter_disabled_flag) {	
slice_beta_offset_div2	se(v)
slice_tc_offset_div2	se(v)
}	
}	
if(sps_lmcs_enabled_flag) {	
slice_lmcs_enabled_flag	u(1)
if(slice_lmcs_enabled_flag) {	
slice_lmcs_aps_id	u(5)
if(!(qtbt_dual_tree_intra_flag && slice_type == I) && (!sps_or_slice_chroma_residual_scale_flag))	
slice_chroma_residual_scale_flag	u(1)
}	
}	
if(entropy_coding_sync_enabled_flag)	

num_entry_point_offsets	uc(v)
if(NumEntryPoints > 0) {	
offset_len_minus1	uc(v)
for(i = 0; i < NumEntryPoints; i++)	
entry_point_offset_minus1[i]	u(v)
}	
byte_alignment()	
}	

Fig.4 là sơ đồ minh họa quy trình 400 mà bộ giải mã theo một phương án có thể thực hiện. Quy trình 400 có thể bắt đầu từ bước s402.

Bước s402 bao gồm bộ giải mã thu (ví dụ, nhận từ bộ mã hóa hoặc nút mạng khác hoặc lấy từ bộ nhớ) tập tham số thứ nhất của luồng bit video được mã hóa.

Bước s404 bao gồm bộ giải mã, được dựa trên từ mã thứ nhất được gồm trong tập tham số thứ nhất nhận được, xác định một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể (ví dụ, tập tham số bao gồm từ mã từ đó tham số có thể được suy ra), trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong đó giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất. Nghĩa là, bộ giải mã, được dựa trên từ mã thứ nhất, xác định xem liệu có hay không mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

Theo một số phương án, ở bước s404, bộ giải mã, được dựa trên từ mã thứ nhất xác định hoặc: 1) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể hoặc 2) tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể.

Theo phương án khác, ở bước s404, bộ giải mã, được dựa trên từ mã thứ nhất xác định hoặc: 1) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể hoặc 2) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

Theo một phương án khác, ở bước s404, bộ giải mã, được dựa trên từ mã thứ nhất xác định hoặc: 1) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể hoặc 2) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit.

Theo một số phương án, bước xác định (bước s404) bao gồm bộ giải mã thu giá trị chỉ báo từ từ mã thứ nhất được gồm trong tập tham số và xác định xem liệu có hay không giá trị chỉ báo thu được bằng giá trị nhất định.

Theo một số phương án, do kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc do kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể. Theo một số phương án, do kết quả của bước xác định là tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ giải mã thu giá trị tham số từ tập tham số mà bao gồm giá trị tham số. Theo một số phương án, tập tham số mà bao gồm giá trị tham số là tập tham số thứ nhất, và giá trị tham số được mã hóa trong từ mã thứ nhất từ đó giá trị chỉ báo được thu và thu giá trị tham số bao gồm bước suy ra giá trị tham số từ từ mã thứ nhất. Theo một số phương án, giá trị tham số được mã hóa trong từ mã thứ hai đó là khác biệt từ từ mã thứ nhất, và thu giá trị tham số bao gồm bước suy ra giá trị tham số từ

từ mã thứ hai. Theo một số phương án, bước xử lý 400 còn gồm, đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số thu được để giải mã dữ liệu đoạn của đoạn.

Theo một số phương án, do kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc do kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể. Theo một số phương án, do kết quả của bước xác định là mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ giải mã thực hiện xử lý mà gồm: đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, thu từ từ mã thứ hai được gồm trong đoạn giá trị tham số tương ứng với tham số cụ thể và sử dụng giá trị tham số thu được để giải mã dữ liệu đoạn của đoạn.

Theo một số phương án, mỗi đoạn được gồm trong tập đoạn thứ nhất được gồm trong lớp mã hóa video khác, VCL, lớp trừu tượng mạng, NAL, đơn vị. Theo một số phương án, mỗi đoạn đã nêu bao gồm tiêu đề đoạn và đoạn tải tin. Theo một số phương án, đoạn là lát, tiêu đề đoạn là tiêu đề lát, và đoạn tải tin chứa dữ liệu lát.

Theo một số phương án, tập tham số thứ nhất được gồm trong phần tải tin của đơn vị NAL không phải VCL được gồm trong luồng bit. Theo một số phương án, tập tham số thứ nhất là tập tham số chuỗi (SPS, sequence parameter set), tập tham số ảnh (PPS, picture parameter set), bộ giải mã tập tham số (DPS, decoder parameter set), tập tham số video (VPS), hoặc tiêu đề ảnh tập tham số, PHPS. PHPS (đôi khi cũng được gọi là “tiêu đề ảnh”), chứa dữ liệu mà được sử dụng để giải mã ảnh đơn. Trường hợp ảnh được chia thành nhiều các lát, PHPS có thể được sử dụng để báo hiệu thông tin cho tất cả các lát của ảnh. Điều này tiết kiệm bit so với giải pháp thay thế trong đó thông tin được lặp lại trong mỗi lát của ảnh. PHPS có thể được gồm trong tiêu đề lát, nhưng một tiêu đề lát có giá trị cho tất cả các lát trong ảnh. Do đó, PHPS có thể là một phần của đơn vị NAL mà chứa lát thứ nhất của ảnh (trong kịch bản này PHPS được gọi là “tiêu đề ảnh”). PHPS có thể cũng được báo

hiệu thay thế, hoặc ngoài ra, tới tiêu đề lát của lát thứ nhất trong ảnh, Do đó, PHPS có thể được báo hiệu như đơn vị NAL riêng và chứ không phải trong bất kỳ đơn vị NAL lát nào.

Theo một số phương án, quy trình 400 còn gồm, trước khi thực hiện bước xác định, bộ giải mã thu thông tin chỉ báo rằng tập tham số thứ nhất bao gồm từ mã thứ nhất. Theo một số phương án, thu thông tin chỉ báo rằng tập tham số thứ nhất bao gồm từ mã thứ nhất bao gồm nhận thông tin từ từ mã thứ hai được gồm trong tập tham số thứ nhất.

Theo một số phương án, tham số cụ thể là 1-bit tham số.

Theo một số phương án, do kết quả của bước xác định là giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, bộ giải mã thực hiện xử lý mà gồm: thu giá trị tham số mặc định; và, đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số mặc định thu được để giải mã dữ liệu đoạn của đoạn.

Theo một số phương án, do kết quả của bước xác định là giá trị tham số tương ứng với tham số cụ thể phải được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, bộ giải mã thực hiện xử lý mà gồm: suy ra giá trị tham số tương ứng với tham số cụ thể từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit; và đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số được suy ra để giải mã dữ liệu đoạn của đoạn. Theo một số phương án, giá trị tham số là giá trị tham số loại lát, và suy ra giá trị tham số loại lát bao gồm thu loại ảnh giá trị tham số mà nhận dạng loại ảnh và suy ra giá trị tham số loại lát được dựa trên loại ảnh giá trị tham số thu được.

Theo một số phương án, luồng bit bao gồm CVS thứ nhất và CVS thứ hai, tập tham số thứ nhất và tập đoạn thứ nhất đều được gồm trong CVS thứ nhất, và tập tham số thứ hai và tập các đoạn thứ hai đều được gồm trong CVS thứ hai. Theo một số phương án, quy trình 400 còn gồm bộ giải mã nhận tập tham số thứ hai được gồm trong CVS thứ hai; và được dựa trên từ mã được gồm trong tập tham số thứ hai nhận được, bộ giải mã xác định một trong số: i) mỗi đoạn được gồm trong tập các đoạn thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số

được gồm trong CVS thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể và giá trị tham số sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai, iii) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai, hoặc iv) giá trị tham số tương ứng với tham số cụ thể phải được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong CVS và giá trị tham số được suy ra sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai.

Fig.5 là sơ đồ minh họa quy trình 500 mà bộ mã hóa theo một phương án có thể thực hiện. Quy trình 500 có thể bắt đầu từ bước s502. Bước s502 bao gồm bộ mã hóa tạo tập đoạn thứ nhất. Bước s504 bao gồm bộ mã hóa tạo tập tham số thứ nhất. Tập tham số thứ nhất gồm từ mã thứ nhất mà chỉ báo một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số thứ nhất hoặc tập tham số thứ hai được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác và giá trị tham số được suy ra được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất.

Theo một số phương án, quy trình 500 còn gồm, ưu tiên tạo tập tham số thứ nhất, xác định rằng giá trị tham số của tham số cụ thể sẽ không thay đổi đối với mỗi đoạn được gồm trong tập đoạn thứ nhất. Theo một số phương án, do kết quả của bước xác định là giá trị tham số của tham số cụ thể sẽ không thay đổi đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, bộ mã hóa tạo tập tham số thứ nhất như là từ mã thứ nhất chỉ báo rằng: i) tập tham số bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) giá trị tham số mặc định tương ứng với tham số cụ thể có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iii) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong đó giá trị

tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn.

Fig.6 là sơ đồ khối của thiết bị 600, theo một số phương án, để thực hiện bộ mã hóa video 140 hoặc bộ giải mã video 260. Như được thể hiện trên FIG. 6, thiết bị 600 có thể bao gồm: mạch xử lý (PC) 602, mà có thể gồm một hoặc nhiều bộ xử lý (P) 655 (ví dụ, bộ vi xử lý mục đích chung và/hoặc một hoặc nhiều bộ xử lý khác, như là mạch tích hợp ứng dụng cụ thể (ASIC, application specific integrated circuit), mảng cổng lập trình trường (FPGAs, field-programmable gate arrays), và tương tự), mà các bộ xử lý có thể đặt cùng vị trí trong một vỏ hoặc trong một trung tâm dữ liệu hoặc có thể được phân bố theo địa lý (nghĩa là, thiết bị 600 có thể là thiết bị tính toán phân tán); giao diện mạng 648 bao gồm bộ phát (Tx) 645 và bộ thu (receiver, Rx) 647 để cho phép thiết bị 600 để truyền dữ liệu đến và nhận dữ liệu từ các nút khác được kết nối tới mạng 110 (ví dụ, mạng giao thức Internet (Internet Protocol, IP)) tới giao diện mạng 648 được kết nối (trực tiếp hoặc gián tiếp) (ví dụ, giao diện mạng 648 có thể được kết nối không dây tới mạng 110, trong trường hợp giao diện mạng 648 được kết nối với thiết bị anten); và đơn vị lưu trữ nội bộ (còn được biết đến là “hệ thống lưu trữ dữ liệu”) 608, mà có thể gồm một hoặc nhiều thiết bị lưu trữ bất khả biến và/hoặc một hoặc nhiều thiết bị lưu trữ khả biến. Theo một phương án trong đó PC 602 gồm bộ xử lý lập trình được, sản phẩm chương trình máy tính (CPP, computer program product) 641 có thể được đề xuất. CPP 641 gồm phương tiện đọc được bằng máy tính (CRM, computer readable medium) 642 lưu trữ chương trình máy tính (CP, computer program) 643 bao gồm các lệnh đọc được bằng máy tính (CRI, computer readable instructions) 644. CRM 642 có thể là phương tiện đọc được bằng máy tính bất khả biến, như là, phương tiện từ tính (ví dụ, đĩa cứng), phương tiện quang học, các thiết bị nhớ (ví dụ, bộ nhớ truy cập ngẫu nhiên, bộ nhớ chớp), và tương tự. Theo một số phương án, CRI 644 của chương trình máy tính 643 được tạo cấu hình như là khi được thực thi bởi PC 602, CRI do thiết bị 600 để thực hiện các bước được mô tả ở đây (ví dụ, các bước được mô tả ở đây có tham chiếu tới các lưu đồ). Theo các phương án khác, thiết bị 600 có thể được tạo cấu hình để thực hiện các bước được mô tả ở đây mà không cần mã hóa. Nghĩa là, ví dụ, PC 602 có thể chỉ gồm có một hoặc nhiều ASIC. Do đó, các tính

năng của các phương án được mô tả ở đây có thể được triển khai trong phần cứng và/hoặc phần mềm.

Fig.7A minh họa các đơn vị chức năng của thiết bị giải mã video 701 theo phương án của sáng chế.

Fig.7B minh họa các đơn vị chức năng của thiết bị mã hóa video 721 theo phương án của sáng chế.

Tóm tắt các phương án khác

A1. Phương pháp 400 được thực hiện bởi bộ giải mã video 260, 600, 701 để giải mã luồng bit, luồng bit bao gồm tập tham số thứ nhất và tập đoạn thứ nhất, phương pháp bao gồm: thu tập tham số thứ nhất; được dựa trên từ mã thứ nhất được gồm trong tập tham số thứ nhất, bộ giải mã xác định một trong số các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong đó giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

A2. Phương pháp của phương án A1, trong đó xác định bao gồm bộ giải mã thu giá trị chỉ báo từ từ mã thứ nhất được gồm trong tập tham số và xác định xem liệu có hay không giá trị chỉ báo thu được bằng giá trị nhất định.

A3. Phương pháp của phương án A2, trong đó do kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc do kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể.

A4. Phương pháp của phương án A3, trong đó, do kết quả của bước xác định là tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ giải mã thu giá trị tham số từ tập tham số mà bao gồm giá trị tham số.

A5. Phương pháp của phương án A4, trong đó tập tham số mà bao gồm giá trị tham số là tập tham số thứ nhất.

A6a1. Phương pháp theo phương án bất kỳ A4 hoặc A5, trong đó giá trị tham số được mã hóa trong từ mã thứ nhất từ đó giá trị chỉ báo được thu và thu giá trị tham số bao gồm bước suy ra giá trị tham số từ từ mã thứ nhất.

A6a2. Phương pháp của phương án A6a1, trong đó suy ra giá trị tham số (PV) từ từ mã thứ nhất bao gồm bước tính toán: $PV = cw - 1$, trong đó cw là giá trị của từ mã thứ nhất.

A6b. Phương pháp của phương án A4 hoặc A5, trong đó giá trị tham số được mã hóa trong từ mã thứ hai đó là khác biệt từ từ mã thứ nhất, và thu giá trị tham số bao gồm bước suy ra giá trị tham số từ từ mã thứ hai.

A7. Phương pháp theo một trong các phương án bất kỳ A4, A5, A6a1, A6a2, và A6b, còn bao gồm: đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số thu được to giải mã dữ liệu đoạn của đoạn.

A8. Phương pháp của phương án A2, trong đó do kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc do kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

A9. Phương pháp của phương án A8, trong đó, do kết quả của bước xác định là mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ giải mã thực hiện xử lý mà gồm: đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, thu từ từ mã thứ hai được gồm trong đoạn giá trị tham số tương ứng với tham số cụ thể và sử dụng giá trị tham số thu được để giải mã dữ liệu

đoạn của đoạn.

A10. Phương pháp theo một trong các phương án bất kỳ trước đó, trong đó mỗi đoạn được gồm trong tập đoạn thứ nhất được gồm trong lớp mã hóa video khác, VCI., lớp trừu tượng mạng, NAL., đơn vị.

A11. Phương pháp của phương án A10, trong đó mỗi đoạn đã nêu bao gồm tiêu đề đoạn và đoạn tải tin.

A12. Phương pháp của phương án A11, trong đó đoạn là lát, tiêu đề đoạn là tiêu đề lát, và đoạn tải tin chứa dữ liệu lát.

A13. Phương pháp theo một trong các phương án bất kỳ trước đó, trong đó tập tham số thứ nhất được gồm trong phần tải tin của đơn vị NAL không phải VCI được gồm trong luồng bit.

A13b. Phương pháp của phương án A13, trong đó tập tham số thứ nhất tập tham số chuỗi, SPS, tập tham số ảnh, PPS, bộ giải mã tập tham số, DPS, tập tham số video, VPS, hoặc tiêu đề ảnh tập tham số, PHPS.

A14. Phương pháp theo một trong các phương án bất kỳ trước đó, còn bao gồm: trước khi thực hiện bước xác định, bộ giải mã thu thông tin chỉ báo rằng tập tham số thứ nhất bao gồm từ mã thứ nhất.

A15a. Phương pháp của phương án A14, trong đó thu thông tin chỉ báo rằng tập tham số thứ nhất bao gồm từ mã thứ nhất bao gồm nhận thông tin từ từ mã thứ hai được gồm trong tập tham số thứ nhất.

A15b. Phương pháp của phương án A14, trong đó từ mã thứ hai gồm có bit đơn.

A15c. Phương pháp theo một trong các phương án bất kỳ A14, A15a, và A15b, trong đó thông tin chỉ báo rằng tập tham số thứ nhất bao gồm từ mã thứ nhất còn chỉ báo rằng tập tham số thứ nhất còn bao gồm từ mã thứ ba tương ứng với tham số cụ thể thứ hai, và phương pháp còn bao gồm bộ giải mã, được dựa trên từ mã thứ ba, xác định xem liệu có hay không mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể thứ hai.

A16. Phương pháp theo một trong các phương án bất kỳ trước đó, trong đó tham số cụ thể là tham số 1-bit.

A17. Phương pháp theo một trong các phương án bất kỳ A1 hoặc A2, trong đó, do kết quả của bước xác định là giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, bộ giải mã thực hiện xử lý mà gồm: thu giá trị tham số mặc định; và đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số mặc định thu được để giải mã dữ liệu đoạn của đoạn.

A18. Phương pháp theo một trong các phương án bất kỳ A1 hoặc A2, trong đó, do kết quả của bước xác định là giá trị tham số tương ứng với tham số cụ thể phải được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, bộ giải mã thực hiện xử lý mà gồm: suy ra giá trị tham số tương ứng với tham số cụ thể từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit; và đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, sử dụng giá trị tham số được suy ra để giải mã dữ liệu đoạn của đoạn.

A19. Phương pháp của phương án A18, trong đó giá trị tham số là giá trị tham số loại lát, và suy ra giá trị tham số loại lát bao gồm thu loại ảnh giá trị tham số mà nhận dạng loại ảnh và suy ra giá trị tham số loại lát được dựa trên loại ảnh giá trị tham số thu được.

A20. Phương pháp của bất kỳ phương án nào trong số các phương án trước đó, trong đó luồng bit bao gồm chuỗi video được mã hóa thứ nhất, CVS, và CVS thứ hai; tập tham số thứ nhất và tập đoạn thứ nhất cả hai được gồm trong CVS thứ nhất, và tập tham số thứ hai và tập các đoạn thứ hai cả hai được gồm trong CVS thứ hai.

A21. Phương pháp của phương án A20, còn bao gồm bộ giải mã nhận tập tham số thứ hai được gồm trong CVS thứ hai; và được dựa trên từ mã được gồm trong tập tham số thứ hai nhận được, bộ giải mã xác định một trong các: i) mỗi đoạn được gồm trong tập các đoạn thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số được gồm trong CVS thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể và giá trị tham số sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai, iii) giá trị tham số mặc định

tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai, hoặc iv) giá trị tham số tương ứng với tham số cụ thể phải được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong CVS và giá trị tham số được suy ra sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn thứ hai.

A22. Phương pháp của phương án A1, trong đó xác định bước bao gồm hoặc:

i) xác định mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc ii) xác định tập tham số được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

A23. Phương pháp của phương án A1, trong đó xác định bước bao gồm hoặc:

i) xác định mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc ii) xác định giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

A24. Phương pháp của phương án A1, trong đó xác định bước bao gồm hoặc:

i) xác định mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc ii) xác định giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit.

B1. Phương pháp (500) được thực hiện bởi bộ mã hóa video (140, 600, 721), phương pháp bao gồm tạo tập đoạn thứ nhất; và tạo tập tham số thứ nhất, trong đó tập tham số thứ nhất gồm từ mã thứ nhất mà chỉ báo một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số thứ nhất hoặc tập tham số thứ hai được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham

số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác và giá trị tham số được suy ra được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất.

B2. Phương pháp của phương án B1, còn bao gồm: ưu tiên tạo tập tham số thứ nhất, xác định giá trị tham số của tham số cụ thể sẽ không thay đổi đối với mỗi đoạn được gồm trong tập đoạn thứ nhất.

B3. Phương pháp của phương án B2, trong đó, do kết quả của bước xác định là giá trị tham số của tham số cụ thể sẽ không thay đổi đối với mỗi đoạn được gồm trong tập đoạn thứ nhất, các bộ mã hóa tạo tập tham số thứ nhất như là từ mã thứ nhất chỉ báo rằng: i) tập tham số bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn, ii) giá trị tham số mặc định tương ứng với tham số cụ thể có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iii) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong đó giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập các đoạn.

C1. Phương pháp 400 được thực hiện bởi bộ giải mã 260, 600, 701 để giải mã luồng bit, luồng bit bao gồm tập tham số ảnh, PPS, và tập các lát thứ nhất, phương pháp bao gồm: thu s402 tập tham số ảnh; và giải mã phần tử cú pháp được gồm trong tập tham số ảnh để thu giá trị chỉ báo (ví dụ, cờ), trong đó bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt thành giá trị thứ nhất thì sau đó nó được xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất, nếu không nó được xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

C2. Phương pháp 500 được thực hiện bởi bộ mã hóa 140, 600, 721, phương pháp bao gồm tạo (s502) tập các lát thứ nhất; và tạo (s504) tập tham số ảnh, trong đó tập tham số ảnh gồm từ mã thứ nhất mà được đặt làm giá trị thứ nhất hoặc giá trị

thứ hai, khi từ mã thứ nhất được đặt là giá trị thứ nhất, sau đó từ mã thứ nhất chỉ báo rằng tiêu đề ảnh được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất, và khi từ mã thứ nhất được đặt là giá trị thứ hai, sau đó từ mã thứ nhất chỉ báo rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

D1. Bộ giải mã video (260, 600, 701), bộ giải mã video được làm thích ứng với: thu tập tham số thứ nhất được gồm trong luồng bit video được mã hóa, trong đó luồng bit cũng gồm tập đoạn thứ nhất; và được dựa trên từ mã thứ nhất được gồm trong tập tham số thứ nhất, xác định một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tham số thứ nhất hoặc tập tham số thứ hai được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong đó giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

E1. Bộ giải mã video (701), bộ giải mã video bao gồm: đơn vị thu (702) có thể hoạt động để thu tập tham số thứ nhất được gồm trong luồng bit video được mã hóa, trong đó luồng bit cũng gồm tập đoạn thứ nhất; và đơn vị xác định (704) có thể hoạt động để xác định, được dựa trên từ mã thứ nhất được gồm trong tập tham số thứ nhất, một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tham số thứ nhất hoặc tập tham số thứ hai được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, trong đó giá trị tham số có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể sẽ được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác được gồm trong luồng bit, trong

đó giá trị tham số được suy ra có thể được sử dụng để giải mã dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất.

F1. Bộ mã hóa video (140, 600, 721), bộ mã hóa video được làm thích ứng với: tạo tập đoạn thứ nhất; và tạo tập tham số thứ nhất, trong đó tập tham số thứ nhất gồm từ mã thứ nhất mà chỉ báo một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số thứ nhất hoặc tập tham số thứ hai được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác và giá trị tham số được suy ra được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất.

G1. Bộ mã hóa video (721), bộ mã hóa video bao gồm: đơn vị tạo đoạn (722) có thể hoạt động để tạo tập đoạn thứ nhất; và đơn vị tạo tập tham số (724) tạo tập tham số thứ nhất, trong đó tập tham số thứ nhất gồm từ mã thứ nhất mà chỉ báo một trong các: i) mỗi đoạn được gồm trong tập đoạn thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, ii) tập tham số thứ nhất hoặc tập tham số thứ hai được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu đoạn của mỗi đoạn được gồm trong tập đoạn thứ nhất, iii) giá trị tham số mặc định tương ứng với tham số cụ thể được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất, hoặc iv) giá trị tham số tương ứng với tham số cụ thể có thể được suy ra từ một hoặc nhiều các giá trị tham số khác và giá trị tham số được suy ra được sử dụng để dữ liệu đoạn mã hóa của mỗi đoạn được gồm trong tập đoạn thứ nhất.

H1. Chương trình máy tính (643) bao gồm các lệnh (644) mà khi được thực thi bằng mạch xử lý (602) lần cho mạch xử lý (602) thực hiện phương pháp của một trong các phương án ở trên A1-A24, B1-B3, C1-C2.

H2. Sóng mang có chứa chương trình máy tính của phương án H1, trong đó

sóng mang là một trong những tín hiệu điện tử, tín hiệu quang học, tín hiệu vô tuyến, và vật ghi đọc được bằng máy tính (642).

Ưu điểm

Như đã lưu ý trong phần Tóm tắt, ưu điểm của các phương án được mô tả ở trên là, đối với mỗi tham số mà không đổi trong ít nhất một phần của luồng bit (ví dụ trong CVS), không cần phải báo hiệu giá trị tham số cho tham số ở cấp độ đoạn (ví dụ lát), mà lưu trữ các bit và cải thiện hiệu quả nén tổng thể.

Theo ví dụ trước từ CTC, trung bình có thể tiết kiệm được 16.5, 7.9, 12.8 và 7.8 bit mỗi lát cho mỗi cấu hình AI, RA, LDB và LDP đối với VTM 5.0 như được thể hiện trên bảng dưới đây.

BẢNG 18 - Số lượng bit trung bình ước tính được tiết kiệm trên mỗi lát

tham số lát	Lượng bit trung bình được tiết kiệm trên mỗi lát			
	AI	RA	LDB	LDP
no_output_of_prior_pics_flag	1			
slice_sao_luma_flag	1			
slice_sao_chroma_flag	1			
tile_group_alf_enabled_flag	1			
dep_quant_enabled_flag	1	1	1	1
partition_constraint_override_flag	1			
slice_loop_filter_across_slices_enabled_flag ¹⁾	1	1	1	1
short_term_ref_pic_set_sps_flag ²⁾	0,99		1	1

inter_ref_pic_set_prediction_flag	0,99			
slice_temporal_mvp_enabled_flag			1	1
mvd_11_zero_flag			1	
collocated_from_l0_flag			1	
num_negative_pics	0,99			
num_positive_pics	0,99			
reshaper_model_delta_max_bin_idx	2,39			
slice_reshaper_ChromaAdj ³⁾			0,79	0,79
six_minus_max_num_merge_cand	0,16	0,98	1	1
five_minus_max_num_affine_merge_cand ⁴⁾		0,98	1	1
max_num_merge_cand_minus_max_num_triangle_cand		2,93	3	
collocated_ref_idx ⁵⁾		0,96	1	1
slice_type	3			
Tổng	16,5	7,9	12,8	7,8

Lưu ý: 1) slice_loop_filter_across_slices_enabled_flag không phải là một phần của thông số kỹ thuật VVC 5.0 v7 (cờ này được chuyển đến PPS); 2) short_term_ref_pic_set_sps_flag không phải là một phần của thông số kỹ thuật VVC 5.0 v7 (các cờ ref_pic_list_sps_flag[i] là tương đương; 3) slice_reshaper_ChromaAdj không phải là một phần của thông số kỹ thuật VVC 5.0 v7 (slice_chroma_residual_scale_flag là tương đương); 4) five_minus_max_num_affine_merge_cand không phải là một phần của thông số kỹ thuật VVC 5.0 v7 (tham số này tương đương với five_minus_max_num_subblock_merge_cand); và 5) collocated_ref_idx không

phải là một phần của thông số kỹ thuật VVC 5.0 v7 (dường như không có tham số tương đương).

Viết tắt

ALF	Bộ lọc lặp thích ứng
APS	Tập tham số thích ứng
AUD	Đơn vị truy cập bộ phân định
BLA	Truy cập liên kết hồng
CRA	Truy cập ngẫu nhiên sạch
CVS	Chuỗi video được mã hóa
CVSS	Bắt đầu CVS
CU	Đơn vị mã hóa
DPS	Tập tham số bộ giải mã
GRA	Truy cập ngẫu nhiên dần dần
HEVC	Mã hóa video hiệu quả cao
IDR	Làm mới bộ giải mã tức thời
IRAP	Điểm truy cập ngẫu nhiên nội bộ
JVET	Nhóm chuyên gia video chung
LMCS	Ảnh xạ độ chói và chia tỷ lệ sắc độ
MPEG	Motion Picture Experts Group (Nhóm các chuyên gia hình ảnh động)
NAI	Lớp trừu tượng mạng
PES	Luồng sơ cấp được đóng gói
PPS	Tập tham số ảnh
RADL	Dẫn đầu có thể giải mã truy cập ngẫu nhiên
RASL	Dẫn đầu loại bỏ truy cập ngẫu nhiên
SPS	Tập tham số chuỗi

STSA Truy cập lớp con theo thời gian từng bước

TSA Truy cập lớp con theo thời gian

VCL Lớp mã hóa video

VPS Tập tham số video

VVC Mã hóa video đa năng

SEI Lớp nâng cao bổ sung

Phần đóng góp (phần này của bản mô tả có chứa những thành phần phù hợp của phần đóng góp mà có thể được đệ trình để xem xét)

Đóng góp dự thảo bắt đầu

Tiêu đề: Các tham số trong SPS hoặc các tiêu đề lát

Tình trạng: Nhập tài liệu vào JVET

Mục đích: đề xuất

Tóm tắt

Một số các tham số lát trong VVC không thay đổi cho tất cả các lát trong luồng bit cho một số cấu hình CTC. Báo cáo này đề xuất cơ chế cho phép báo hiệu một số tham số lát trong VVC hoặc trong SPS hoặc cho mỗi lát.

Cụ thể hơn, những thay đổi sau đây đối với VVC được đề xuất trong đóng góp này

(1) Báo hiệu `sps_or_slice_flag` in SPS mà định rõ xem liệu có hay không các giá trị các phần tử cú pháp được liệt kê dưới đây có thể được báo hiệu có điều kiện trong SPS hoặc các tiêu đề lát, hoặc nếu chúng luôn được báo hiệu trong các tiêu đề lát.

- a. `dep_quant_enabled_flag`,
- b. `ref_pic_list_sps_flag`,
- c. `slice_temporal_mvp_enabled_flag`,
- d. `mvd_l1_zero_flag`,
- e. `collocated_from_l0_flag`,

- f. `six_minus_max_num_merge_cand`
- g. `five_minus_max_num_subblock_merge_cand`
- h. `max_num_merge_cand_minus_max_num_triangle_cand`

(2) Nếu `sps_or_slice_flag` bằng 1, sau đó đối với mỗi các phần tử cú pháp trên, báo hiệu tham số tương ứng trong SPS. Nếu giá trị tham số tương ứng bằng 0, sau đó phần tử cú pháp tương ứng được báo hiệu trong tiêu đề lát. Nếu không thì, giá trị cho tiêu đề lát phần tử cú pháp tương ứng được suy ra là giá trị tham số SPS trừ đi 1.

Số tỷ lệ BD độ chói cho CTC được báo cáo là 0.0%/0.0%/-0.1%/0.0% đối với AI/RA/LDB/LDP tương ứng và và những người đề xuất yêu cầu tiết kiệm cao hơn khi sử dụng nhiều lát cắt trên mỗi ảnh.

Giới thiệu

Xét thấy rằng các tham số trong tiêu đề lát không thay đổi cho tất cả các tiêu đề lát trong tất cả các chuỗi kiểm tra của CTC. Bảng thể hiện trong đó cấu hình CTC sử dụng các tham số không thay đổi hiện tại VTM 5.0 cho tất cả các tiêu đề lát trong tất cả các chuỗi kiểm tra.

BẢNG 19 - Các tham số lát không thay đổi cho mỗi cấu hình CTC sử dụng VTM 5.0

Tên tham số lát	Không đổi trong			
	AI	RA	LDB	LDP
<code>dep_quant_enabled_flag</code>	x	x	x	x
<code>short_term_ref_pic_set_sps_flag</code>	x		x	x
<code>slice_temporal_mvp_enabled_flag</code>			x	x
<code>mvd_11_zero_flag</code>			x	

collocated_from_l0_flag			x	
six_minus_max_num_merge_cand	x	x	x	x
five_minus_max_num_affine_merge_cand		x	x	x
max_num_merge_cand_minus_max_num_triangle_cand		x	x	

Cần lưu ý rằng tên cặp lát tham số khác nhau giữa VTM 5.0 và đặc tả VVC dự thảo, JVET-N1001-v8. Do đó, chúng tôi đề nghị thay đổi tên của short_term_ref_pic_set_sps_flag và five_minus_max_num_affine_merge_cand thành ref_pic_list_sps[0], ref_pic_list_sps[1] và five_minus_max_num_subblock_merge_cand trong phần mềm VTM.

Đề xuất

Phần dư gửi tới các tham số trong tiêu đề lát mà không thay đổi cho tất cả tiêu đề lát trong luồng bit. Do đó, cơ chế được đề xuất để cho phép báo hiệu một số các tham số lát hoặc trong SPS hoặc cho mỗi lát.

Cụ thể hơn, những thay đổi sau đây tới VVC được đề xuất trong báo cáo này:

(1) Báo hiệu sps_or_slice_flag trong SPS mà định rõ xem liệu có hay không các giá trị của các phần tử cú pháp được liệt kê dưới đây có thể được báo hiệu có điều kiện trong SPS hoặc các tiêu đề lát, hoặc nếu chúng luôn được báo hiệu trong các tiêu đề lát.

- i. dep_quant_enabled_flag,
- j. ref_pic_list_sps_flag,
- k. slice_temporal_mvp_enabled_flag,
- l. mvd_11_zero_flag,
- m. collocated_from_l0_flag,
- n. six_minus_max_num_merge_cand
- o. five_minus_max_num_subblock_merge_cand
- p. max_num_merge_cand_minus_max_num_triangle_cand

Nếu `sps_or_slice_flag` bằng 1, sau đó đối với mỗi trong các phần tử cú pháp nêu trên, báo hiệu tham số trong SPS tương ứng.

Nếu giá trị tham số bằng 0, sau khi phần tử cú pháp được báo hiệu cho mỗi tiêu đề lát. Nếu không thì, giá trị cho phần tử cú pháp được suy ra là giá trị tham số trừ đi 1.

Các thay đổi được đề xuất đối với đặc điểm kỹ thuật VVC

Các thay đổi được đề xuất ở đầu bản dự thảo VVC (JVET-N1001-v8) được hiển thị.

<code>seq_parameter_set_rbsp() {</code>	Phần mô tả
<code>...</code>	
<code><u>sps_or_slice_flag</u></code>	<code>u(1)</code>
<code>if (sps_or_slice_flag) {</code>	
<code><u>sps_or_slice_dep_quant_enabled_flag</u></code>	<code>tu(v)</code>
<code>for(i = 0; i < 2; i++)</code>	
<code><u>sps_or_slice_ref_pic_list_sps_flag[i]</u></code>	<code>tu(v)</code>
<code><u>sps_or_slice_temporal_mvp_enabled_flag</u></code>	<code>tu(1+v)</code>
<code><u>sps_or_slice_mvd_l1_zero_flag</u></code>	<code>tu(v)</code>
<code><u>sps_or_slice_collocated_from_l0_flag</u></code>	<code>tu(v)</code>
<code><u>sps_or_slice_six_minus_max_num_merge_cand</u></code>	<code>ue(v)</code>
<code><u>sps_or_slice_five_minus_max_num_subblock_merge_cand</u></code>	<code>ue(v)</code>

—	uc(v)
<u>sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand</u>	
<u>nd</u>	
—}	
if(sps_or_slice_temporal_mvp_enabled_flag != 1)	
<u>sps_sbtmvp_enabled_flag</u>	u(1)
...	
}	

sps_or_slice_flag bằng 0 định rõ rằng sps_or_slice_dep_quant_enabled_flag, sps_or_slice_ref_pic_list_sps_flag, sps_or_slice_temporal_mvp_enabled_flag, sps_or_slice_mvd_11_zero_flag, sps_or_slice_collocated_from_l0_flag, sps_or_slice_six_minus_max_num_merge_cand, sps_or_slice_five_minus_max_num_subblock_merge_cand và sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand được suy ra bằng 0. sps_or_slice_flag bằng 1 định rõ rằng các phần tử cú pháp này được biểu diễn trong SPS.

sps_or_slice_dep_quant_enabled_flag bằng 0 định rõ rằng dep_quant_enabled_flag được biểu diễn trong các tiêu đề lát. sps_or_slice_dep_quant_enabled_flag lớn hơn 0 định rõ rằng dep_quant_enabled_flag được suy ra bằng sps_or_slice_dep_quant_enabled_flag trừ đi 1. Giá trị lớn nhất maxVal của tu(v) nhị phân bậc một rút gọn được đặt bằng 2.

sps_or_slice_ref_pic_list_sps_flag[i] bằng 0 định rõ rằng ref_pic_list_sps[i] được biểu diễn trong các tiêu đề lát. sps_or_slice_ref_pic_list_sps_flag[i] lớn hơn 0 định rõ rằng ref_pic_list_sps[i] được suy ra bằng sps_or_slice_ref_pic_list_sps_flag[i] trừ đi 1. Giá trị lớn nhất maxVal của tu(v) nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_temporal_mvp_enabled_flag` bằng 0 định rõ rằng `slice_temporal_mvp_enabled_flag` được biểu diễn trong các tiêu đề lát của các lát với `slice_type` không bằng I trong CVS. `sps_or_slice_temporal_mvp_enabled_flag` lớn hơn 0 định rõ rằng `slice_temporal_mvp_enabled_flag` được suy ra bằng `sps_or_slice_temporal_mvp_enabled_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_mvd_11_zero_flag` bằng 0 định rõ rằng `mvd_11_zero_flag` được biểu diễn trong các tiêu đề lát. `sps_or_slice_mvd_11_zero_flag` lớn hơn 0 định rõ rằng `mvd_11_zero_flag` được suy ra bằng `sps_or_slice_mvd_11_zero_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_collocated_from_10_flag` bằng 0 định rõ rằng `collocated_from_10_flag` được biểu diễn trong các tiêu đề lát. `sps_or_slice_collocated_from_10_flag` lớn hơn 0 định rõ rằng `collocated_from_10_flag` được suy ra bằng `sps_or_slice_collocated_from_10_flag` trừ đi 1. Giá trị lớn nhất `maxVal` của `tu(v)` nhị phân bậc một rút gọn được đặt bằng 2.

`sps_or_slice_six_minus_max_num_merge_cand` bằng 0 định rõ rằng `six_minus_max_num_merge_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_six_minus_max_num_merge_cand` lớn hơn 0 định rõ rằng `six_minus_max_num_merge_cand` được suy ra bằng `sps_or_slice_six_minus_max_num_merge_cand` trừ đi 1. Giá trị của `sps_or_slice_six_minus_max_num_merge_cand` sẽ nằm trong khoảng từ 0 đến 6.

`sps_or_slice_five_minus_max_num_subblock_merge_cand` bằng 0 định rõ rằng `five_minus_max_num_subblock_merge_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_five_minus_max_num_subblock_merge_cand` lớn hơn 0 định rõ rằng `five_minus_max_num_subblock_merge_cand` được suy ra bằng `sps_or_slice_five_minus_max_num_subblock_merge_cand` trừ đi 1. Giá trị của `sps_or_slice_five_minus_max_num_subblock_merge_cand` sẽ nằm trong khoảng từ 0 đến 6.

`sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand`

bằng 0 định rõ rằng `max_num_merge_cand_minus_max_num_triangle_cand` được biểu diễn trong các tiêu đề lát. `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` lớn hơn 0 định rõ rằng `max_num_merge_cand_minus_max_num_triangle_cand` được suy ra bằng `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` trừ đi 1. Giá trị của `sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand` sẽ nằm trong khoảng từ 0 đến `MaxNumMergeCand - 1`.

<code>slice_header() {</code>	Phần mô tả
<code>...</code>	
<code>if((NalUnitType != IDR_W_RADL && NalUnitType != IDR_N_LP) sps_idr_rpl_present_flag) {</code>	
<code>for(i = 0; i < 2; i++) {</code>	
<code>if(num_ref_pic_lists_in_sps[i] > 0 && !sps_or_slice_ref_pic_list_sps_flag[i] && (i == 0 (i == 1 && rpl1_idx_present_flag)))</code>	
<code>ref_pic_list_sps_flag[i]</code>	<code>u(1)</code>
<code>...</code>	
<code>if(slice_type != I) {</code>	
<code>if(!sps_or_slice_temporal_mvp_enabled_flag)</code>	
<code>slice_temporal_mvp_enabled_flag</code>	<code>u(1)</code>
<code>if(slice_type == B && !sps_or_slice_mvd_11_zero_flag)</code>	

mvd_l1_zero_flag	u(1)
...	
if(slice_temporal_mvp_enabled_flag) {	
if(slice_type == B && !sps_or_slice_collocated_from_l0_flag)	
collocated_from_l0_flag	u(1)
}	
...	
if(!sps_or_slice_six_minus_max_num_merge_cand)	
six_minus_max_num_merge_cand	ue(v)
if(sps_affine_enabled_flag && !sps_or_slice_five_minus_max_num_subblock_merge_cand)	
five_minus_max_num_subblock_merge_cand	ue(v)
...	
if(sps_triangle_enabled_flag && MaxNumMergeCand >= 2 && !sps_or_slice_max_num_merge_cand_minus_max_num_triangle_cand)	
max_num_merge_cand_minus_max_num_triangle_cand	ue(v)
} else if (sps_ibc_enabled_flag && !sps_or_slice_six_minus_max_num_merge_cand)	
six_minus_max_num_merge_cand	ue(v)

...	
<u>if (!sps or slice dep quant enabled flag)</u>	
dep_quant_enabled_flag	u(1)
...	
byte_alignment()	
}	

Kết quả

Tiết kiệm tốc độ bit cho CTC sử dụng VTM-5.0 được hiển thị bên dưới. Đối với A1, sps or slice_flag được đặt thành 0 vì SPS được gửi cho mỗi ảnh. Lưu ý rằng không có nhiều trong các số tỷ lệ BD này, vì tăng ích không đến từ sự thay đổi chất lượng mà hoàn toàn là do tiết kiệm các bit bằng cách báo hiệu các tham số không đổi trong SPS thay vì trong tiêu đề lát. Đối với chuỗi Johnny trong LDB, tỷ lệ BD là -0,37%.

Tất cả cấu hình nội Main10					
qua VTM-5.0					
	Y	U	V	EncT	DecT
Lớp A1	0,00%	0,00%	0,00%	98%	98%
Lớp A2	0,00%	0,00%	0,00%	99%	100%
Lớp B	0,00%	0,00%	0,00%	98%	99%
Lớp C	0,00%	0,00%	0,00%	105%	102%
Lớp E	0,00%	0,00%	0,00%	94%	95%
Tổng thể	0,00%	0,00%	0,00%	99%	99%

Lớp D	0,00%	0,00%	0,00%	98%	100%
Lớp F	0,00%	0,00%	0,00%	100%	100%

Truy cập ngẫu nhiên Main 10					
Qua VTM-5.0					
	Y	U	V	EncT	DecT
Lớp A1	0,00%	0,00%	0,00%	100%	98%
Lớp A2	0,00%	0,00%	0,00%	100%	98%
Lớp B	-0,01%	-0,01%	-0,01%	99%	97%
Lớp C	-0,02%	-0,02%	-0,02%	99%	91%
Lớp E					
Tổng thể	-0,01%	-0,01%	-0,01%	101%	100%
Lớp D	-0,07%	-0,06%	-0,07%	98%	95%
Lớp F	-0,02%	-0,02%	-0,02%	105%	91%

B Main10 độ trễ thấp					
Qua VTM-5.0					
	Y	U	V	EncT	DecT
Lớp A1					

Lớp A2					
Lớp B	-0,02%	-0,01%	-0,02%	101%	98%
Lớp C	-0,04%	-0,04%	-0,04%	102%	100%
Lớp E	-0,25%	-0,20%	-0,20%	97%	99%
Tổng thể	-0,08%	-0,07%	-0,07%	100%	99%
Lớp D	-0,13%	-0,11%	-0,11%	103%	105%
Lớp F	-0,12%	-0,09%	-0,10%	100%	97%

P Main10 độ trễ thấp					
Qua VTM-5.0					
	Y	U	V	EncT	DecT
Lớp A1					
Lớp A2					
Lớp B	-0,01%	-0,01%	-0,01%	101%	101%
Lớp C	-0,02%	-0,02%	-0,02%	101%	102%
Lớp E	-0,08%	-0,07%	-0,07%	100%	101%
Tổng thể	-0,03%	-0,03	-0,03%	102%	102%
Lớp D	-0,05%	-0,04%	-0,04%	104%	105%
Lớp F	-0,05%	-0,05%	-0,05%	102%	101%

Kết quả ước tính cho các cấu hình không phải CTC

Trong cấu hình không phải CTC với nhiều lát lát, mức tiết kiệm bit dự kiến sẽ cao hơn. Một ước tính sơ bộ được thực hiện trong bảng dưới đây bằng cách nhân tiết kiệm bit CTC với số các lát:

Tiết kiệm bit ước tính cho nhiều lát trên mỗi ảnh

Các lát X trên mỗi ảnh	Tiết kiệm bit ước tính (Y)		
	RA	LDB	LDP
6	-0,06%	-0,48%	-0,18%
24 (Ví dụ OMAF)	-0,24%	-1,92%	-0,72%

Đóng góp cuối cùng

Trong khi các phương án khác nhau được mô tả ở đây (bao gồm cả đóng góp dự thảo đính kèm), cần hiểu rằng chúng chỉ được trình bày theo cách làm ví dụ, chứ không phải giới hạn sáng chế. Do đó, phạm vi của sáng chế này không nên bị giới hạn bởi bất kỳ phương án nào trong số các phương án được mô tả ở trên. Hơn nữa, bất kỳ sự kết hợp nào của các yếu tố được mô tả ở trên trong tất cả các biến thể có thể có của chúng đều được bao gồm trong sáng chế trừ khi có chỉ dẫn khác ở đây hoặc nếu không thì mâu thuẫn rõ ràng theo ngữ cảnh.

Ngoài ra, trong khi các quy trình được mô tả ở trên và được minh họa trong bản vẽ được thể hiện dưới dạng một chuỗi các bước, điều này được thực hiện chỉ nhằm mục đích minh họa. Theo đó, người ta dự tính rằng một số bước có thể được thêm vào, một số bước có thể được bỏ qua, thứ tự của các bước có thể được sắp xếp lại và một số bước có thể được thực hiện song song.

Yêu cầu bảo hộ

1. Phương pháp (400) được thực hiện bởi bộ giải mã (260, 600, 701) để giải mã luồng bit, luồng bit bao gồm tập tham số ảnh (PPS, picture parameter set) và tập các lát thứ nhất, phương pháp bao gồm các bước:

thu (s402) tập tham số ảnh; và

giải mã phân tử củ pháp thứ nhất được gồm trong tập tham số ảnh để thu giá trị chỉ báo, trong đó

bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt thành giá trị thứ nhất thì sau đó nó được xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, nếu không nó được xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, và

nếu tiêu đề ảnh bao gồm giá trị tham số tương ứng với tham số cụ thể, sau đó giá trị tham số được sử dụng để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất.

2. Phương pháp theo điểm 1, còn bao gồm bước xác định xem liệu có hay không giá trị chỉ báo thu được bằng giá trị nhất định.

3. Phương pháp theo điểm 2, trong đó:

theo kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc

theo kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể.

4. Phương pháp theo điểm 3, trong đó, do kết quả của bước xác định là tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ

giải mã thu giá trị tham số từ tiêu đề ảnh.

5. Phương pháp theo điểm 4, còn bao gồm:

đối với mỗi lát được gồm trong tập các lát thứ nhất, sử dụng giá trị tham số thu được để giải mã dữ liệu lát của lát.

6. Phương pháp theo điểm 2, trong đó:

theo kết quả của bước xác định là giá trị chỉ báo thu được bằng giá trị nhất định, bộ giải mã xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, hoặc

theo kết quả của bước xác định là giá trị chỉ báo thu được không bằng giá trị nhất định, bộ giải mã xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

7. Phương pháp theo điểm 6, trong đó, do kết quả của bước xác định là mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, bộ giải mã thực hiện xử lý mà gồm bước:

đối với mỗi lát được gồm trong tập các lát thứ nhất, thu từ phần tử cú pháp thứ hai được gồm trong lát giá trị tham số tương ứng với tham số cụ thể và sử dụng giá trị tham số thu được để giải mã dữ liệu lát của chuỗi.

8. Phương pháp theo điểm bất kỳ trong số các điểm nêu trên, trong đó mỗi lát được gồm trong tập các lát thứ nhất được gồm trong đơn vị lớp mã hóa video khác (VCL, video coding layer), lớp trừu tượng mạng (NAL, network abstraction layer).

9. Phương pháp theo điểm bất kỳ trong số các điểm nêu trên, trong đó tập tham số ảnh được gồm trong phần tải tin của đơn vị NAL không phải VCL được gồm trong luồng bit.

10. Phương pháp theo điểm bất kỳ trong số các điểm nêu trên, còn bao gồm:

trước khi thực hiện bước giải mã, bộ giải mã thu thông tin chỉ báo rằng tập tham số ảnh bao gồm phần tử cú pháp thứ nhất.

11. Phương pháp theo điểm 10, trong đó thu thông tin chỉ báo rằng tập tham số ảnh bao gồm phần tử cú pháp thứ nhất bao gồm nhận thông tin từ phần tử cú pháp thứ

hai được gồm trong tập tham số ảnh.

12. Phương pháp theo một điểm bất kỳ trong số các điểm từ 10-11, trong đó

thông tin chỉ báo rằng tập tham số ảnh bao gồm phần tử cú pháp thứ nhất còn chỉ báo rằng tập tham số ảnh còn bao gồm phần tử cú pháp thứ ba tương ứng với tham số cụ thể thứ hai, và

phương pháp còn bao gồm bộ giải mã, được dựa trên phần tử cú pháp thứ ba, xác định xem liệu có hay không mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể thứ hai.

13. Phương pháp theo điểm bất kỳ trong số các điểm nêu trên, trong đó

luồng bit bao gồm chuỗi video được mã hóa thứ nhất, CVS, và CVS thứ hai;

tập tham số ảnh và tập các lát thứ nhất đều được gồm trong CVS thứ nhất, và

tập tham số ảnh thứ hai và tập các lát thứ hai đều được gồm trong CVS thứ hai.

14. Phương pháp theo điểm 13, còn bao gồm:

giải mã phần tử cú pháp được gồm trong tập tham số ảnh thứ hai để thu giá trị chỉ báo, trong đó

bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt là giá trị thứ nhất thì sau đó nó được xác định rằng tiêu đề ảnh thứ hai được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, nếu không nó được xác định rằng mỗi lát được gồm trong tập các lát thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể, và

nếu tiêu đề ảnh thứ hai bao gồm giá trị tham số tương ứng với tham số cụ thể, sau đó giá trị tham số được sử dụng để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ hai.

15. Phương pháp mã hóa (500) được thực hiện bởi bộ mã hóa (140, 600, 721), phương pháp bao gồm các bước:

tạo (s502) tập các lát thứ nhất; và

tạo (s504) tập tham số ảnh, trong đó

tập tham số ảnh gồm phần tử cú pháp thứ nhất mà được đặt làm giá trị thứ nhất hoặc giá trị thứ hai,

khi phần tử cú pháp thứ nhất được đặt là giá trị thứ nhất, sau khi phần tử cú pháp thứ nhất chỉ báo rằng tiêu đề ảnh được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất, và

khi phần tử cú pháp thứ nhất được đặt là giá trị thứ hai, sau khi phần tử cú pháp thứ nhất chỉ báo rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể.

16. Vật ghi đọc được bằng máy tính (642) chứa chương trình máy tính (643) bao gồm các lệnh (644) mà khi được thực thi bằng mạch xử lý (602) lần cho mạch xử lý (602) thực hiện phương pháp theo một điểm bất kỳ trong số các điểm từ 1 đến 14.

17. Vật ghi đọc được bằng máy tính (642) chứa chương trình máy tính (643) bao gồm các lệnh (644) mà khi được thực thi bằng mạch xử lý (602) lần cho mạch xử lý (602) thực hiện phương pháp theo điểm 15.

18. Bộ giải mã (260, 600, 701) để giải mã luồng bit, luồng bit bao gồm tập tham số ảnh (PPS, picture parameter set) và tập các lát thứ nhất, bộ giải mã được làm thích ứng với:

thu (s402) tập tham số ảnh; và

giải mã phần tử cú pháp thứ nhất được gồm trong tập tham số ảnh để thu giá trị chỉ báo, trong đó

bộ giải mã được tạo cấu hình sao cho nếu giá trị chỉ báo được đặt thành giá trị thứ nhất sau đó bộ giải mã xác định rằng tiêu đề ảnh được gồm trong luồng bit bao gồm giá trị tham số tương ứng với tham số cụ thể, nếu không thì bộ giải mã xác định rằng mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể, và

bộ giải mã được tạo cấu hình như là, nếu tiêu đề ảnh bao gồm giá trị tham số

tương ứng với tham số cụ thể, sau đó bộ giải mã sử dụng giá trị tham số để giải mã dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất.

19. Bộ mã hóa (140, 600, 721), bộ mã hóa được làm thích ứng để:

tạo (s502) tập các lát thứ nhất; và

tạo (s504) tập tham số ảnh, trong đó

tập tham số ảnh gồm phần tử cú pháp thứ nhất mà chỉ báo một trong các bước:

i) mà mỗi lát được gồm trong tập các lát thứ nhất bao gồm giá trị tham số tương ứng với tham số cụ thể hoặc ii) mà tiêu đề ảnh được tạo bởi bộ mã hóa bao gồm giá trị tham số tương ứng với tham số cụ thể mà được sử dụng để mã hóa dữ liệu lát của mỗi lát được gồm trong tập các lát thứ nhất.

1/7

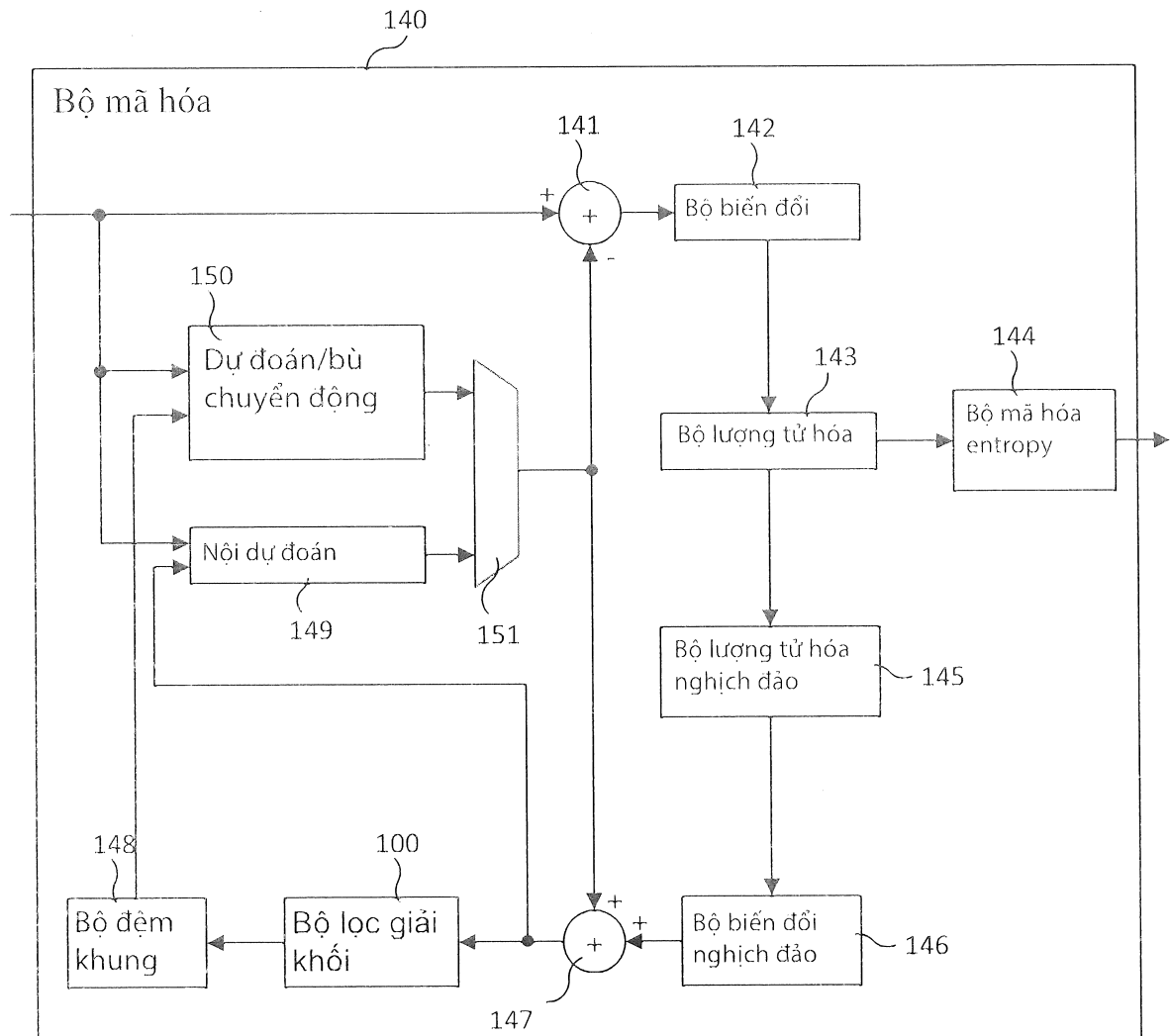


FIG. 1

2/7

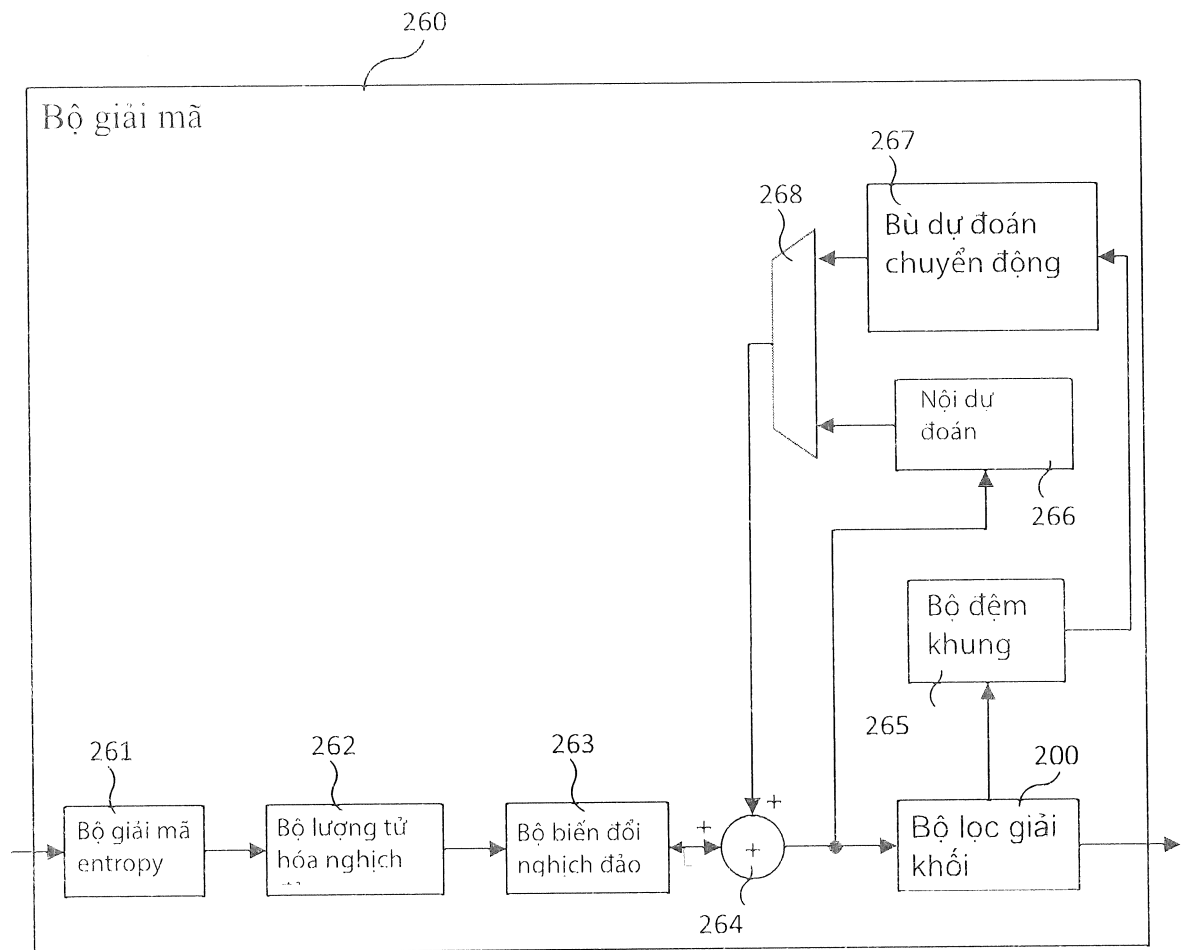


FIG. 2

3/7

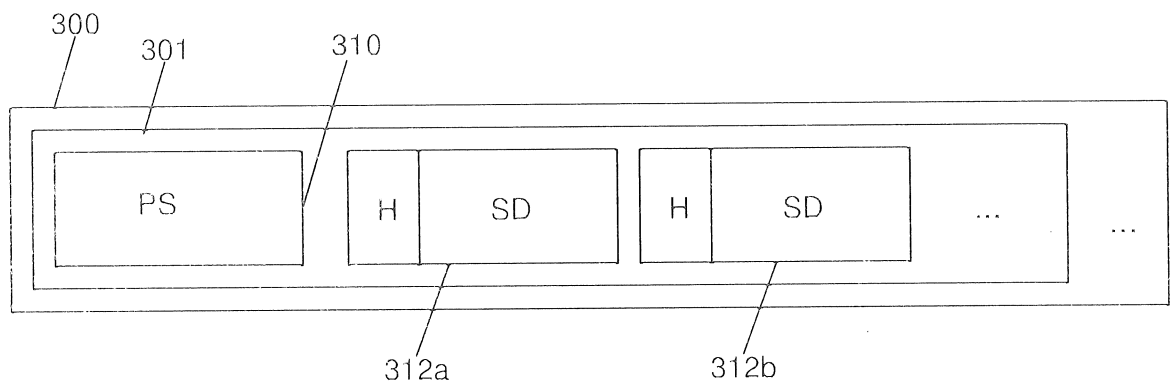


FIG. 3

4/7

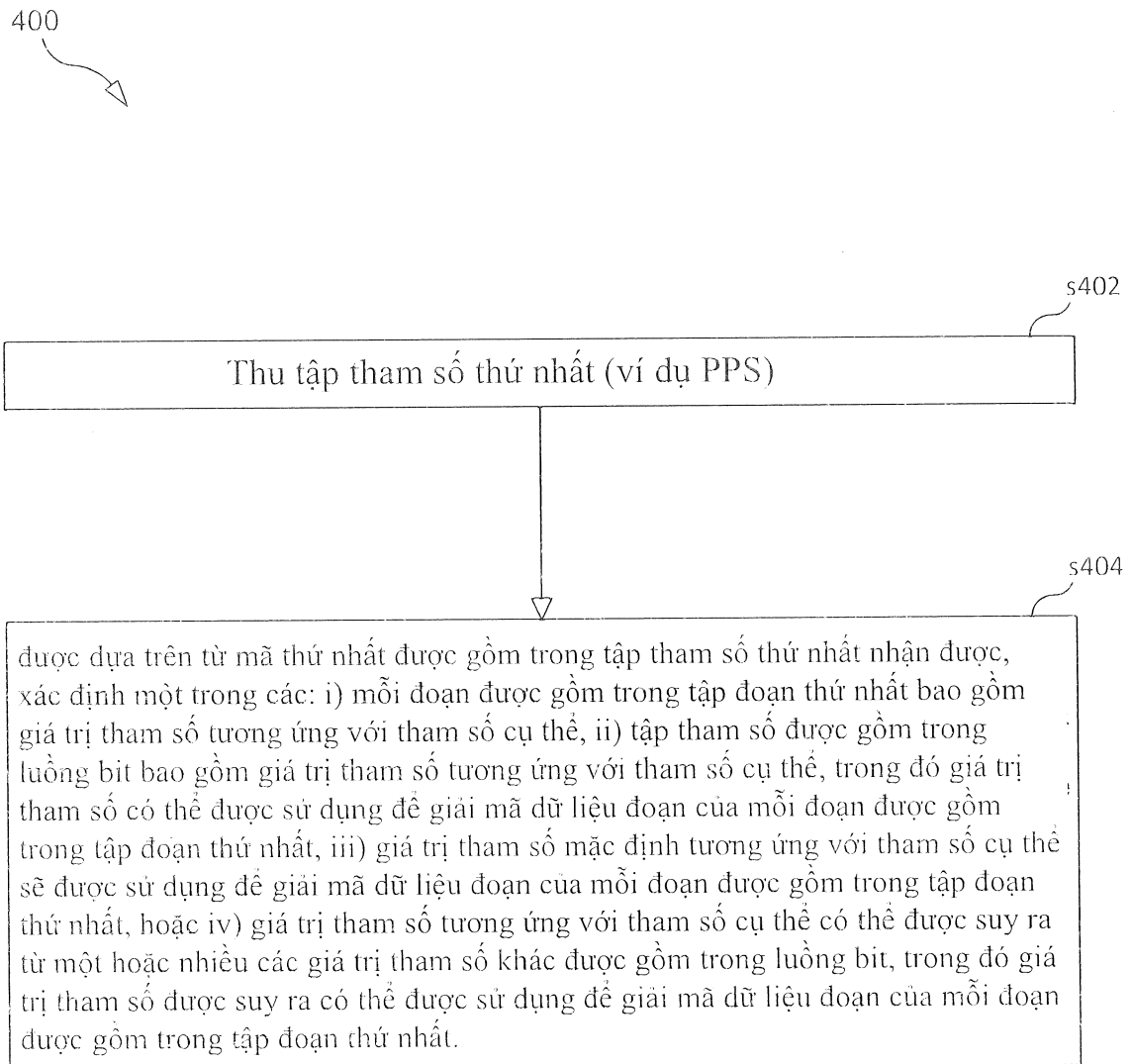


FIG. 4

5/7

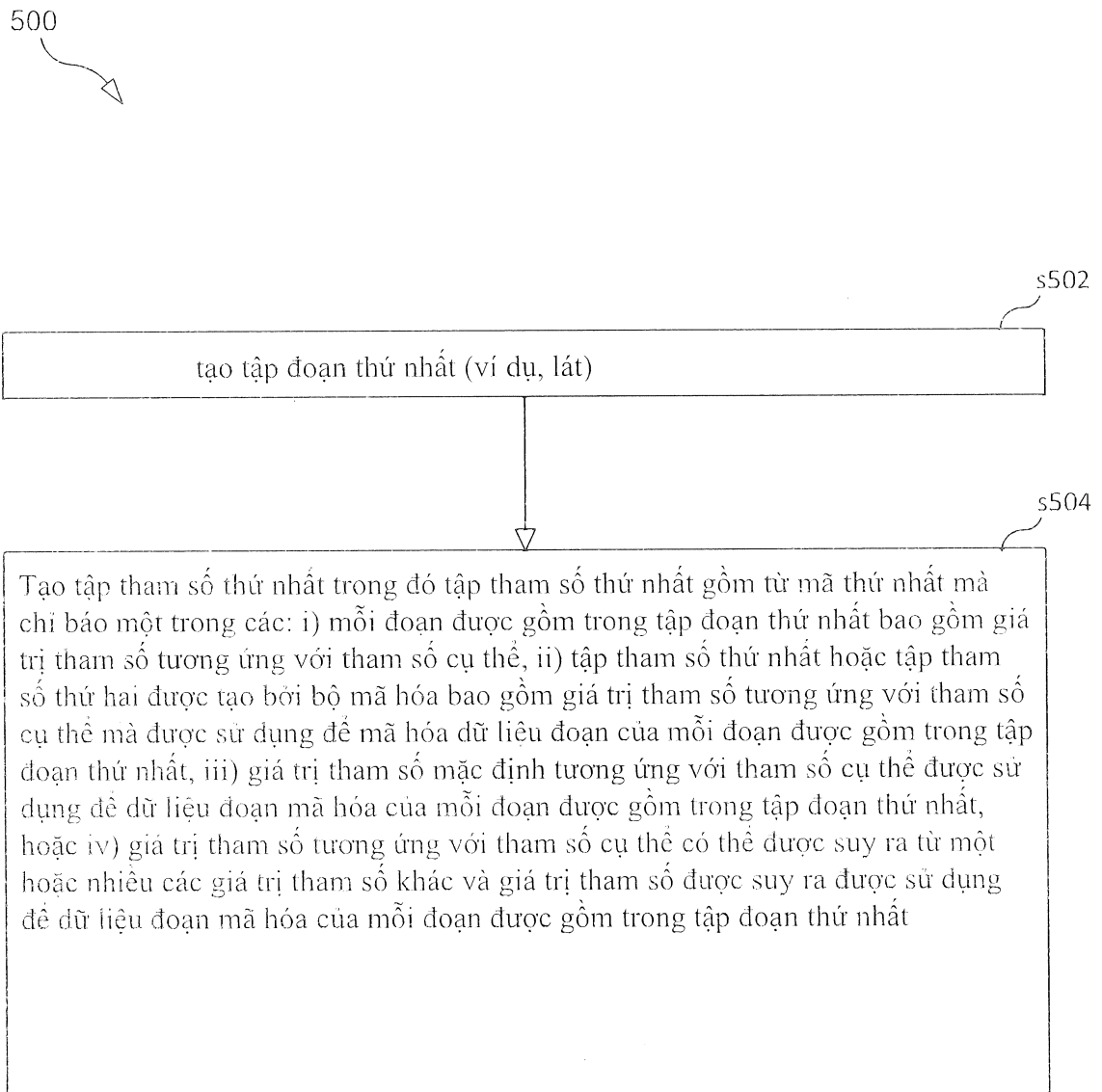


FIG. 5

6/7

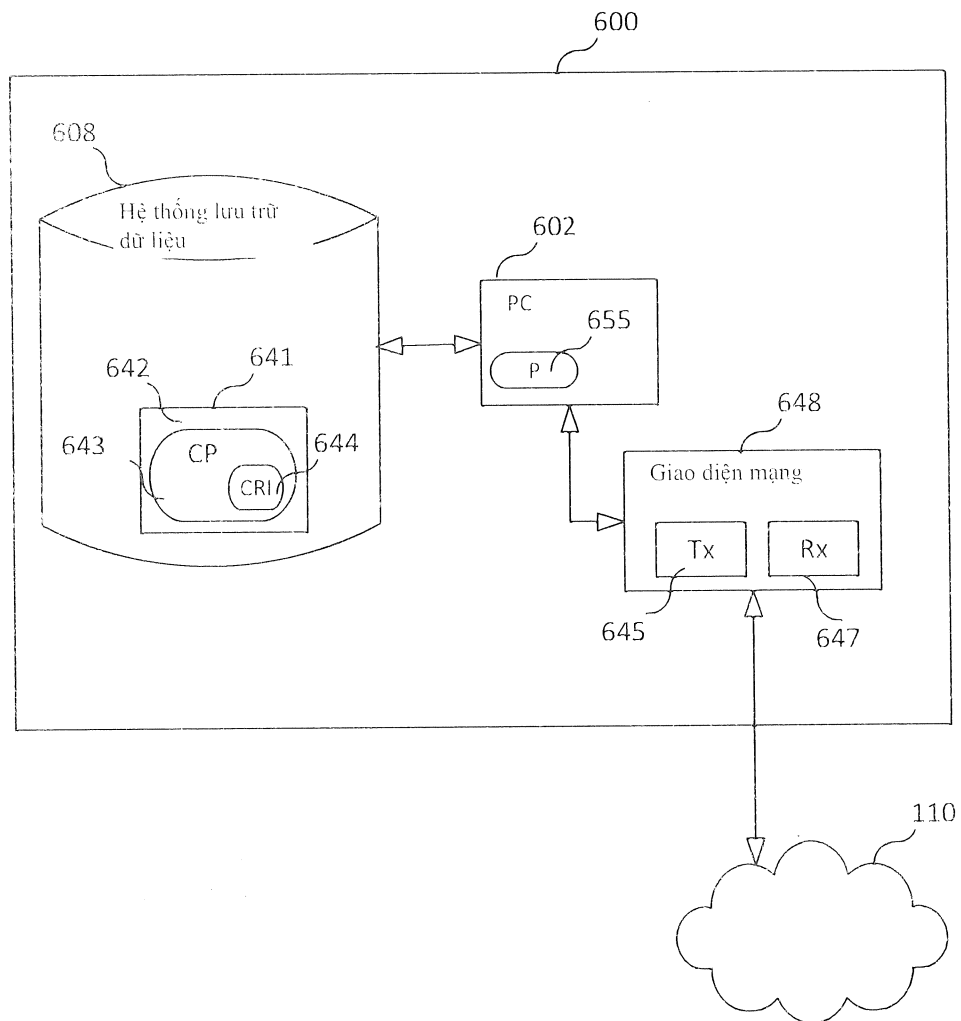


FIG. 6

7/7

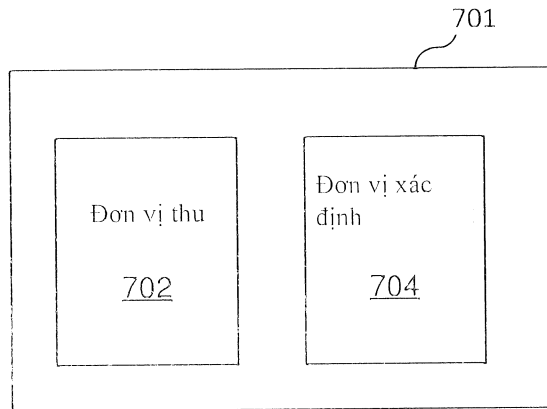


FIG. 7A

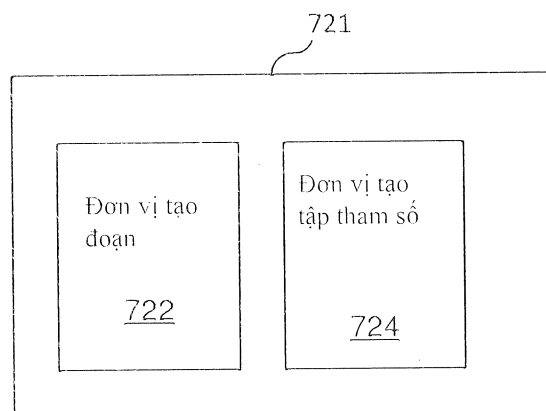


FIG. 7B