



- (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
1-0054018
(51)^{2021.01} H04N 19/103; H04N 19/513; H04N 19/174; H04N 19/157; H04N 19/172 (13) B

-
- (21) 1-2022-07203 (22) 07/04/2021
(86) PCT/US2021/026271 07/04/2021 (87) WO2021/207423 14/10/2021
(30) 63/007,355 08/04/2020 US; 63/009,404 13/04/2020 US; 63/010,005 14/04/2020 US;
63/010,619 15/04/2020 US; 63/015,663 26/04/2020 US
(45) 25/11/2025 452 (43) 27/01/2023 418A
(73) BEIJING DAJIA INTERNET INFORMATION TECHNOLOGY CO., LTD. (CN)
Room 101D1-7, 1st Floor, Building 1, No. 6, Shangdi West Road, Haidian District,
Beijing 100085, China
(72) CHEN, Yi-Wen (TW); XIU, Xiaoyu (CN); MA, Tsung-Chuan (TW); JHU, Hong-
Jheng (TW); CHEN, Wei (CN); WANG, Xianglin (US); YU, Bing (CN).
(74) Công ty TNHH Trường Xuân (AGELESS CO.,LTD.)

-
- (54) PHƯƠNG PHÁP GIẢI MÃ VIDEO, THIẾT BỊ LẬP MÃ VIDEO VÀ PHƯƠNG
TIỆN LƯU TRỮ CÓ THỂ ĐỌC ĐƯỢC TRÊN MÁY TÍNH KHÔNG TẠM THỜI

(21) 1-2022-07203

(57) Sáng chế đề xuất phương pháp giải mã video, thiết bị lập mã video và phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời. Phương pháp này bao gồm bước bộ giải mã xác định liệu cờ vô hiệu hóa có mặt trong tiêu đề hình ảnh (Picture header: PH) kết hợp với hình ảnh, trong đó cờ vô hiệu hóa chỉ rõ liệu công cụ lập mã được vô hiệu hóa trong một hoặc nhiều lát kết hợp với PH hay không. Ngoài ra, phương pháp này bao gồm bước bộ giải mã suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong tập hợp tham số chuỗi (sequence parameter set: SPS) của hình ảnh đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH.

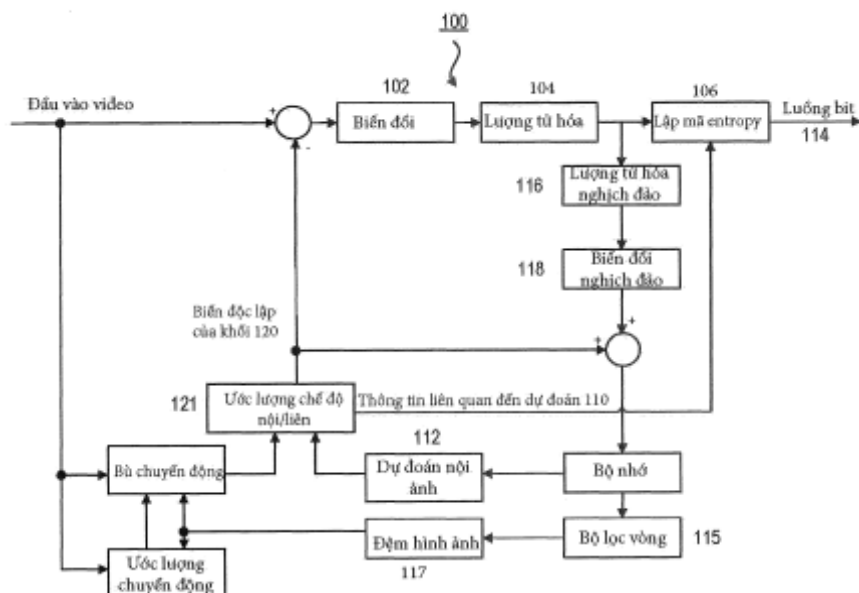


FIG. 1

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lập mã và nén video, và cụ thể là nhưng không giới hạn ở, phương pháp giải mã video, thiết bị lập mã video và phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời.

Tình trạng kỹ thuật của sáng chế

Nhiều công nghệ lập mã video khác nhau có thể được sử dụng để nén dữ liệu video. Lập mã video được thực hiện theo một hoặc nhiều tiêu chuẩn lập mã video. Ví dụ, tiêu chuẩn lập mã video bao gồm lập mã video linh hoạt (VVC), mô hình kiểm tra thăm dò chung (joint exploration test model: JEM), lập mã video hiệu năng cao (H.265/HEVC), lập mã video nâng cao (H.264/AVC), lập mã của nhóm chuyên gia hình ảnh động (moving picture experts group: MPEG), hoặc tương tự. Lập mã video thường sử dụng phương pháp dự đoán (ví dụ, dự đoán liên khung ảnh, dự đoán nội khung ảnh hoặc tương tự) tận dụng sự dư thừa có trong các hình ảnh hoặc chuỗi video. Mục đích quan trọng của công nghệ lập mã video là để nén dữ liệu video thành dạng sử dụng tốc độ bit thấp hơn, đồng thời tránh hoặc giảm thiểu sự suy giảm chất lượng video.

Bản chất kỹ thuật của sáng chế

Sáng chế đề xuất các ví dụ về công nghệ liên quan đến báo hiệu phần tử cú pháp trong lập mã video.

Theo khía cạnh thứ nhất, sáng chế đề xuất phương pháp lập mã video. Phương pháp này bao gồm bộ giải mã xác định liệu cờ vô hiệu hóa có mặt trong tiêu đề hình ảnh (picture header: PH) kết hợp với một ảnh, trong đó cờ vô hiệu hóa chỉ rõ liệu công cụ lập mã được vô hiệu hóa trong một hoặc nhiều lát (slice) kết hợp với PH. Ngoài ra, phương pháp này bao gồm bước bộ giải mã suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong tập hợp tham số chuỗi (sequence parameter set: SPS) của hình ảnh đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH.

Theo khía cạnh thứ hai, sáng chế đề xuất phương pháp lập mã video. Phương pháp này bao gồm bước bộ giải mã xác định liệu cờ có mặt trong PH kết hợp với hình ảnh và cờ chỉ rõ liệu hình ảnh được sử dụng cho dự đoán vector chuyển động theo thời gian (temporal motion vector prediction: TMVP) được suy ra từ danh mục hình ảnh tham chiếu từ nhiều danh mục hình ảnh tham chiếu kết hợp với hình ảnh đó. Ngoài ra, phương pháp này bao gồm bước bộ giải mã suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu đáp lại việc xác định rằng cờ không có mặt trong PH.

Theo khía cạnh thứ ba, sáng chế đề xuất phương pháp lập mã video. Phương pháp này bao gồm bước bộ giải mã xác định liệu cờ có mặt trong PH kết hợp với hình ảnh, cờ chỉ rõ số lượng của trọng số được báo hiệu trong danh mục hình ảnh tham chiếu theo cờ dự đoán trọng số (weighed prediction: WP) thứ nhất trong tập hợp tham số hình ảnh (PPS) của hình ảnh và cờ WP thứ hai trong PH của hình ảnh, và cờ trong cú pháp WP kết hợp với hình ảnh. Ngoài ra, phương pháp này bao gồm bước bộ giải mã suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu từ nhiều danh mục hình ảnh tham chiếu kết hợp với hình ảnh đáp lại việc xác định rằng cờ không có mặt trong PH.

Theo khía cạnh thứ tư, sáng chế đề xuất phương pháp lập mã video. Phương pháp này bao gồm bước bộ giải mã sử dụng cờ được kích hoạt để chỉ rõ liệu một hoặc nhiều biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng cho dự đoán liên ảnh (inter prediction) để một hoặc nhiều lát kết hợp với PH của hình ảnh. Ngoài ra, phương pháp này bao gồm bước bộ giải mã ràng buộc giá trị của cờ được kích hoạt theo nhiều độ lệch được áp dụng cho kích thước của hình ảnh để chọn thang chia tỷ lệ tính toán tỷ lệ.

Theo khía cạnh thứ năm, sáng chế đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý và bộ nhớ được cấu hình để lưu lệnh có thể chạy được bằng một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý, khi chạy các lệnh, được cấu hình để thực hiện phương pháp lập mã video theo khía cạnh thứ nhất của sáng chế.

Theo khía cạnh thứ sáu, sáng chế đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý và bộ nhớ được cấu hình để lưu lệnh có thể được chạy

bằng một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý, khi chạy lệnh, được cấu hình để thực hiện phương pháp lập mã video theo khía cạnh thứ hai của sáng chế.

Theo khía cạnh thứ bảy, sáng chế đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý và bộ nhớ được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý, khi chạy lệnh, được cấu hình để thực hiện phương pháp lập mã video theo khía cạnh thứ ba của sáng chế.

Theo khía cạnh thứ tám, sáng chế đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý và bộ nhớ được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý, khi chạy lệnh, được cấu hình để thực hiện phương pháp lập mã video theo khía cạnh thứ tư của sáng chế.

Theo khía cạnh thứ chín, sáng chế đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời để lập mã video, lưu lệnh chạy trên máy tính. Các lệnh, khi được chạy bởi một hoặc nhiều bộ xử lý máy tính, làm cho một hoặc nhiều bộ xử lý máy tính thực hiện phương pháp lập mã video theo khía cạnh thứ nhất của sáng chế.

Theo khía cạnh thứ mười, sáng chế đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời để lập mã video, lưu lệnh chạy trên máy tính. Các lệnh, khi được chạy bởi một hoặc nhiều bộ xử lý máy tính, làm cho một hoặc nhiều bộ xử lý máy tính thực hiện phương pháp lập mã video theo khía cạnh thứ hai của sáng chế.

Theo khía cạnh thứ mười một, sáng chế đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời để lập mã video, lưu lệnh chạy trên máy tính. Các lệnh, khi được chạy bởi một hoặc nhiều bộ xử lý máy tính, làm cho một hoặc nhiều bộ xử lý máy tính thực hiện phương pháp lập mã video theo khía cạnh thứ ba của sáng chế.

Theo khía cạnh thứ mười hai, sáng chế đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời để lập mã video, lưu lệnh chạy trên máy tính. Các lệnh, khi được chạy bởi một hoặc nhiều bộ xử lý máy tính, làm cho một hoặc nhiều bộ xử lý máy tính thực hiện phương pháp lập mã video theo khía cạnh thứ tư của sáng chế.

Mô tả vắn tắt các hình vẽ

Mô tả cụ thể hơn về các ví dụ của sáng chế sẽ được nêu bằng cách tham khảo các ví dụ cụ thể được minh họa trong các hình vẽ đi kèm. Cho rằng các hình vẽ này chỉ mô tả một số ví dụ và do đó không được coi là giới hạn phạm vi, các ví dụ sẽ được mô tả và

giải thích với các đặc điểm cụ thể và chi tiết bổ sung thông qua việc sử dụng các hình vẽ đi kèm.

FIG. 1 là sơ đồ khối minh họa bộ lập mã video điển hình phù hợp với một số phương án thực hiện của sáng chế.

FIG. 2 là sơ đồ khối minh họa bộ giải mã video điển hình phù hợp với một số phương án thực hiện của sáng chế.

FIG. 3 minh họa ví dụ về hình ảnh được chia thành nhiều đơn vị cây lập mã (coding tree unit: CTU) phù hợp với một số phương án thực hiện của sáng chế.

FIG. 4A-4D là sơ đồ minh họa chế độ phân tách cây nhiều loại phù hợp với một số phương án thực hiện của sáng chế.

FIG. 5 là sơ đồ khối minh họa thiết bị lập mã video điển hình phù hợp với một số phương án thực hiện của sáng chế.

FIG. 6 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

FIG. 7 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

FIG. 8 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

FIG. 9 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

Mô tả chi tiết sáng chế

Bây giờ, sự tham khảo sẽ được nêu chi tiết đến các phương án thực hiện cụ thể, các ví dụ được minh họa trong các hình vẽ đi kèm. Trong phần mô tả chi tiết sau đây, nhiều chi tiết cụ thể không giới hạn được trình bày để hỗ trợ việc hiểu đối tượng được trình bày ở đây. Nhưng đối với người có hiểu biết trung bình trong lĩnh vực này sẽ hiểu rằng có thể sử dụng nhiều phương án thay thế khác nhau. Ví dụ, đối với người có hiểu biết trung bình trong lĩnh vực này, đối tượng được trình bày ở đây có thể được thực hiện trên nhiều loại thiết bị điện tử có khả năng chạy video kỹ thuật số.

Việc tham chiếu trong toàn bộ bản mô tả này đến “một phương án”, “phương án”, “một ví dụ”, “một số phương án”, “một số ví dụ” hoặc tương tự có nghĩa là một tính năng, cấu trúc hoặc đặc tính cụ thể được mô tả nằm trong ít nhất một phương án hoặc ví dụ. Các tính năng, cấu trúc, phần tử hoặc đặc điểm được mô tả liên quan đến một hoặc một số phương án cũng có thể áp dụng cho các phương án khác, trừ khi được quy định rõ ràng khác.

Xuyên suốt bản mô tả, các thuật ngữ “thứ nhất,” “thứ hai”, “thứ ba” và v.v. đều được sử dụng làm thuật ngữ mục chỉ để tham chiếu đến các yếu tố có liên quan, ví dụ, thiết bị, thành phần, bộ cục, các bước và v.v., mà không ngụ ý bất kỳ thứ tự không gian hoặc trình tự thời gian nào, trừ khi được chỉ định rõ ràng khác. Ví dụ: “thiết bị thứ nhất” và “thiết bị thứ hai” có thể đề cập đến hai thiết bị được tạo thành riêng biệt hoặc hai bộ phận, thành phần hoặc trạng thái hoạt động của cùng một thiết bị và có thể được đặt tên tùy ý.

Các thuật ngữ “môđun”, “môđun phụ”, “mạch”, “mạch phụ”, “hệ mạch”, “hệ mạch phụ”, “đơn vị” hoặc “đơn vị phụ” có thể bao gồm bộ nhớ (dùng chung, dành riêng hoặc nhóm) lưu trữ mã hoặc lệnh có thể được chạy bởi một hoặc nhiều bộ xử lý. Môđun có thể bao gồm một hoặc nhiều mạch có hoặc không có mã hoặc lệnh được lưu. Môđun hoặc mạch có thể bao gồm một hoặc nhiều thành phần mà được nối trực tiếp hoặc gián tiếp. Các thành phần này có thể hoặc không thể được gắn vật lý với, hoặc được đặt gần với nhau.

Như được sử dụng ở đây, thuật ngữ “nếu” hoặc “khi nào” có thể được hiểu là “khi” hoặc “đáp lại với” tùy thuộc vào ngữ cảnh. Các thuật ngữ này, nếu xuất hiện trong yêu cầu bảo hộ, có thể không chỉ ra rằng các giới hạn hoặc tính năng liên quan là có điều kiện hoặc tùy chọn. Ví dụ, phương pháp có thể bao gồm các bước: i) khi hoặc nếu điều kiện X có, chức năng hoặc hành động X’ được thực hiện, và ii) khi hoặc nếu điều kiện Y có, chức năng hoặc hành động Y’ được thực hiện. Phương pháp này có thể được thực hiện với cả khả năng thực hiện chức năng hoặc hành động X’ và khả năng thực hiện chức năng hoặc hành động Y’. Do đó, cả hai chức năng X’ và Y’ đều có thể được thực hiện, vào những thời điểm khác nhau, trên nhiều lần thực hiện phương pháp này.

Đơn vị hoặc môđun có thể được triển khai hoàn toàn bằng phần mềm, hoàn toàn bằng phần cứng hoặc bằng sự kết hợp giữa phần cứng và phần mềm. Ví dụ, trong triển

khai phần mềm thuần túy, đơn vị hoặc môđun có thể bao gồm các khối mã hoặc thành phần phần mềm liên quan đến chức năng, được liên kết trực tiếp hoặc gián tiếp với nhau, để thực hiện một chức năng cụ thể.

FIG. 1 thể hiện sơ đồ khối minh họa bộ mã hóa video lai dựa trên khối 100 điển hình mà có thể được sử dụng kết hợp với nhiều tiêu chuẩn lập mã video sử dụng xử lý dựa trên khối. Trong bộ mã hóa 100, khung video được phân vùng thành nhiều khối video để xử lý. Đối với mỗi khối video nhất định, phép dự đoán được tạo ra dựa trên cách tiếp cận dự đoán liên ảnh hoặc cách tiếp cận dự đoán nội ảnh. Trong dự đoán liên ảnh, một hoặc nhiều biến độc lập dự đoán được tạo ra thông qua ước lượng chuyển động và bù chuyển động, căn cứ vào các pixel (điểm ảnh) từ các khung được tái cấu trúc trước đó. Trong dự đoán nội ảnh, biến độc lập dự đoán được tạo ra dựa trên pixel tái cấu trúc trên khung hiện thời. Thông qua việc quyết định chế độ, biến độc lập dự đoán tốt nhất có thể được chọn để dự đoán khối hiện thời.

Phần dư dự đoán, thể hiện sự khác nhau giữa khối video hiện thời và biến độc lập của nó, được gửi đến hệ mạch biến đổi 102. Sau đó, các hệ số biến đổi được gửi từ hệ mạch biến đổi 102 đến hệ mạch lượng tử hóa 104 để giảm entropy. Sau đó, các hệ số lượng tử hóa được nạp vào hệ mạch lập mã entropy 106 để tạo ra luồng bit video được nén. Như được thể hiện trong FIG. 1, thông tin liên quan đến dự đoán 110 từ hệ mạch dự đoán liên và/hoặc hệ mạch dự đoán nội 112, như thông tin phân vùng khối video, vector chuyển động, chỉ số hình ảnh tham chiếu, và chế độ dự đoán nội, cũng được nạp qua hệ mạch lập mã entropy 106 và được lưu vào luồng bit video được nén 114.

Trong bộ mã hóa 100, hệ mạch liên quan đến bộ giải mã cũng cần thiết để tái cấu trúc pixel cho mục đích dự đoán. Trước tiên, phần dư dự đoán được tái cấu trúc thông qua lượng tử hóa nghịch đảo 116 và hệ mạch biến đổi nghịch đảo 118. Phần dư dự đoán đã tái cấu trúc này được kết hợp với biến độc lập của khối 120 để tạo ra pixel tái cấu trúc không được lọc cho khối video hiện thời.

Dự đoán nội (còn được đề cập đến là “dự đoán không gian”) sử dụng các pixel từ các mẫu của khối lân cận đã được lập mã (mà được gọi là các mẫu tham chiếu) trong cùng một hình ảnh và/hoặc lát video để dự đoán khối video hiện thời. Dự đoán không gian làm giảm dư thừa không gian vốn có trong tín hiệu video.

Dự đoán liên ảnh (còn được đề cập đến là “dự đoán thời gian”) sử dụng pixel tái cấu trúc từ các hình ảnh video đã được lập mã sẵn để dự đoán khối video hiện thời. Dự đoán thời gian làm giảm dư thừa thời gian vốn có trong tín hiệu video. Tín hiệu dự đoán thời gian cho đơn vị lập mã (coding unit: CU) thu được hoặc khối lập mã thường được báo hiệu bằng một hoặc nhiều vector chuyển động (motion vector: MV) mà chỉ ra lượng và hướng của chuyển động giữa CU hiện thời và tham chiếu thời gian của nó. Ngoài ra, nếu nhiều hình ảnh tham chiếu được hỗ trợ, một chỉ số hình ảnh tham chiếu ngoài ra được gửi thêm, mà được sử dụng để xác định từ đó hình ảnh tham chiếu trong kho hình ảnh tham chiếu tín hiệu dự đoán thời gian đi đến.

Sau khi thực hiện dự đoán không gian và/hoặc dự đoán thời gian, hệ mạch quyết định chế độ liên/nội 121 trong bộ mã hóa 100 lựa chọn chế độ dự đoán tốt nhất, ví dụ, căn cứ vào phương pháp tối ưu hóa tốc độ-biến dạng. Sau đó, biến độc lập dự đoán của khối 120 được trừ khỏi khối video hiện thời; và phần dư dự đoán thu được được loại bỏ tương quan sử dụng hệ mạch biến đổi 102 và hệ mạch lượng tử hóa 104. Các hệ số dư được lượng tử hóa thu được được biến đổi nghịch đảo bằng hệ mạch lượng tử hóa nghịch đảo 116 và được biến đổi nghịch đảo bằng hệ mạch biến đổi nghịch đảo 118 để tạo ra phần dư tái cấu trúc, mà sau đó được cộng trở lại khối dự đoán để tạo ra tín hiệu tái cấu trúc của CU. Ngoài ra, bộ lọc vòng lặp 115, như bộ lọc khử khối, bù tương thích mẫu (sample adaptive offset: SAO), và/hoặc bộ lọc vòng lặp tương thích (adaptive in-loop filter: ALF) có thể được áp dụng lên CU tái cấu trúc trước khi nó được đặt trong kho hình ảnh tham chiếu của đệm hình ảnh 117 và được sử dụng để lập mã khối video tương lai. Để tạo ra luồng bit video đầu ra 114, chế độ lập mã (liên hoặc nội), thông tin chế độ dự đoán, thông tin chuyển động, và hệ số dư lượng tử hóa đều được gửi đến bộ lập mã entropy 106 để nén thêm và được đóng gói để tạo ra luồng bit.

Ví dụ, bộ lọc khử khối có sẵn trong AVC, HEVC cũng như là phiên bản hiện thời của VVC. Trong HEVC, bộ lọc vòng lặp bổ sung gọi là SAO (sample adaptive offset) được định rõ là để cải thiện thêm hiệu quả lập mã. Trong phiên bản hiện thời của tiêu chuẩn VVC, bộ lọc vòng lặp khác nữa gọi là bộ lọc vòng lặp tương thích ALF (adaptive loop filter) đang được xem xét tích cực và nó có cơ hội tốt để nằm trong tiêu chuẩn cuối cùng.

Thao tác lọc trong vòng là tùy chọn. Thực hiện các thao tác này giúp cải thiện hiệu quả lập mã và chất lượng hình ảnh. Chúng cũng có thể được tắt dưới dạng một quyết định được đưa ra bởi bộ mã hóa 100 để tiết kiệm độ phức tạp tính toán.

Nên lưu ý rằng, dự đoán nội thường được sử dụng căn cứ vào pixel được tái cấu trúc không lọc, trong khi dự đoán liên căn cứ vào pixel tái cấu trúc lọc nếu các sự lựa chọn này được bật lên bởi bộ mã hóa 100.

FIG. 2 là sơ đồ khối minh họa bộ giải mã video dựa vào khối 200 điển hình mà có thể được sử dụng kết hợp với nhiều tiêu chuẩn lập mã video. Bộ giải mã 200 này tương tự với phần liên quan đến tái cấu trúc ở trong bộ mã hóa 100 của FIG. 1. Trong bộ giải mã 200, luồng bit video đến 201 trước tiên được giải mã thông qua giải mã entropy 202 để suy ra các mức hệ số lượng tử hóa và thông tin liên quan đến dự đoán. Sau đó các mức hệ số lượng tử hóa được xử lý thông qua phép lượng tử hóa nghịch đảo 204 và biến đổi ngược 206 để thu được phần dư dự đoán tái cấu trúc. Thiết bị hàm biến độc lập của khối, được thực hiện trong bộ lựa chọn chế độ nội/liên 212, được cấu hình để thực hiện hoặc là phép dự đoán nội 208, hoặc phép bù chuyển động 210, căn cứ vào thông tin dự đoán giải mã. Tập hợp pixel tái cấu trúc không lọc thu được bằng cách cộng phần dư dự đoán tái cấu trúc từ phép biến đổi nghịch đảo 206 và đầu ra dự đoán được tạo ra bởi thiết bị hàm biến độc lập của khối, sử dụng bộ cộng 214.

Khối tái cấu trúc có thể còn đi qua bộ lọc vòng lặp 209 trước khi nó được lưu trong đệm hình ảnh 213 mà thực hiện chức năng như kho lưu hình ảnh tham chiếu. Video tái cấu trúc trong đệm hình ảnh 213 có thể được gửi đến để điều khiển thiết bị hiển thị, cũng như là được sử dụng để dự đoán khối video tương lai. Trong tình huống mà bộ lọc vòng lặp 209 được bật, thao tác lọc được thực hiện trên các pixel tái cấu trúc này để suy ra đầu ra video tái cấu trúc cuối cùng 222.

Tiêu chuẩn lập mã/giải mã video được đề cập đến ở trên như VVC, JEM, HEVC, MPEG-4, Part 10 là cùng một khái niệm. Ví dụ, tất cả chúng sử dụng quy trình căn cứ vào khối. Các sơ đồ phân vùng khối trong một số tiêu chuẩn được xây dựng dưới đây.

Lập mã video đa năng (VVC)

Ở cuộc họp JVET thứ mười được tổ chức từ ngày 10-20/04/2018 tại San Diego, US, JVET xác định bản thảo đầu tiên của lập mã video đa năng (VVC) và chế độ kiểm tra VVC 1 (VTM1) dưới dạng triển khai phần mềm tham chiếu của nó. Nó đã được

quyết định bao gồm một cây tứ phân với cây đa loại lồng nhau làm tính năng mã hóa mới ban đầu của VVC. Cây đa loại là một cấu trúc phân vùng khối mã hóa bao gồm cả phân tách nhị phân và bậc ba. Kể từ đó, phần mềm tham chiếu VTM, với cả quá trình mã hóa và giải mã được thực hiện, đã được phát triển và cập nhật thông qua các cuộc họp JVET sau.

Trong VVC, hình ảnh của video đầu vào được phân vùng thành các khối gọi là CTU. CTU được phân tách thành các CU sử dụng cây tứ phân với cấu trúc cây nhiều loại lồng nhau, với CU định rõ ra một vùng các pixel chia sẻ chung một chế độ dự đoán (ví dụ, nội hoặc liên). Thuật ngữ ‘đơn vị’ để chỉ một vùng của hình ảnh bao gồm tất cả thành phần như luma (độ chói) và chroma (sắc độ). Thuật ngữ ‘khối’ có thể được sử dụng để định rõ vùng mà bao gồm các thành phần cụ thể (ví dụ, luma) và khối của các thành phần khác (ví dụ, luma so với chroma) có thể khác nhau về vị trí không gian khi xem xét định dạng lấy mẫu chroma như 4:2:0.

Sự phân vùng của hình ảnh thành các CTU

FIG. 3 minh họa ví dụ về hình ảnh 300 được chia thành nhiều CTU 302 phù hợp với một số phương án thực hiện của sáng chế.

Các hình ảnh được chia thành chuỗi CTU. Khái niệm về CTU giống với khái niệm của HEVC. Đối với hình ảnh mà có ba mảng mẫu, CTU bao gồm khối $N \times N$ của mẫu luma cùng với hai khối tương ứng của mẫu chroma.

Kích thước cho phép tối đa của khối luma trong CTU được chỉ rõ là 128×128 (mặc dù kích thước tối đa của các khối biến đổi luma là 64×64).

Sự phân vùng của CTU sử dụng cấu trúc cây

Trong HEVC, CTU được phân tách thành các CU bằng cách sử dụng cấu trúc cây tứ phân được thể hiện là cây lập mã để tương thích với các đặc điểm vị trí khác nhau. Quyết định liệu có lập mã vùng hình ảnh sử dụng dự đoán liên ảnh (thời gian) hoặc nội ảnh (không gian) được tạo ra ở mức độ CU lá hay không. Mỗi CU lá có thể được phân tách thêm thành một, hai, hoặc bốn PU theo loại phân tách PU. Bên trong một PU, cùng một quy trình dự đoán được áp dụng và thông tin liên quan được truyền đến bộ giải mã trên cơ sở PU. Sau khi thu được khối dư bằng cách áp dụng xử lý dự đoán căn cứ vào loại phân tách PU, thì CU lá có thể được phân vùng thành các đơn vị biến đổi (transform

unit: TU) theo cấu trúc cây tứ phân khác tương tự với cây lập mã cho CU. Một trong số đặc điểm chính của cấu trúc HEVC là nó có nhiều khái niệm phân vùng bao gồm CU, PU, và TU.

Trong VVC, cây tứ phân với cây đa loại lồng nhau sử dụng cấu trúc đoạn phân tách tam phân và nhị phân thay thế các khái niệm về loại đơn vị đa phân vùng, tức là, nó loại bỏ sự tách rời của khái niệm CU, PU và TU ngoại trừ các khái niệm cần thiết cho CU có kích thước quá lớn so với chiều dài biến đổi tối đa và hỗ trợ linh hoạt hơn cho các hình dạng phân vùng CU. Trong cấu trúc cây lập mã, CU có thể có hình vuông hoặc hình chữ nhật. Trước tiên, CTU được phân vùng bởi cấu trúc cây tứ phân (còn gọi là cây tứ phân). Sau đó, các nút lá của cây tứ phân có thể được phân chia thêm bằng cấu trúc cây nhiều loại.

FIG. 4A-4D là sơ đồ minh họa chế độ phân tách cây nhiều loại phù hợp với một số phương án thực hiện của sáng chế. Như được thể hiện trong FIG. 4A-4D, có bốn loại phân tách trong cấu trúc cây đa loại, phân tách nhị phân thẳng đứng 402 (SPLIT_BT_VER), phân tách nhị phân ngang 404 (SPLIT_BT_HOR), phân tách tam phân thẳng đứng 406 (SPLIT_TT_VER), và phân tách tam phân ngang 408 (SPLIT_TT_HOR). Nút lá cây nhiều loại được gọi là CU, và trừ khi CU quá lớn so với độ dài biến đổi tối đa, sự phân đoạn này được sử dụng cho phép dự đoán và xử lý biến đổi mà không cần phân vùng thêm. Điều này có nghĩa là, trong hầu hết các trường hợp, CU, PU, và TU có cùng một kích thước khối trong cây tứ phân với cấu trúc khối lập mã cây nhiều loại lồng nhau. Ngoại lệ xảy ra khi chiều dài biến đổi được hỗ trợ tối đa nhỏ hơn chiều rộng hoặc chiều cao của thành phần màu của CU.

Cú pháp trong VVC

Trong VVC, lớp thứ nhất của luồng bit của báo hiệu cú pháp là lớp trừu tượng mạng (Network Abstraction Layer: NAL), trong đó luồng bit được chia thành tập hợp các đơn vị NAL. Vài đơn vị NAL báo hiệu các tham số điều khiển chung cho bộ giải mã, như tập hợp tham số chuỗi (sequence parameter set: SPS) và tập hợp tham số hình ảnh (picture parameter set: PPS). Các đơn vị khác chứa dữ liệu video. Đơn vị NAL của lớp lập mã video (video coding layer: VCL) chứa các lát (slice) của video được lập mã. Hình ảnh đã lập mã được gọi là đơn vị truy cập và có thể được mã hóa dưới dạng một hoặc nhiều lát.

Trình tự video được lập mã bắt đầu với hình ảnh làm mới bộ giải mã tức thời (Instantaneous Decoder Refresh: IDR). Tất cả hình ảnh video sau được lập mã dưới dạng lát. Hình ảnh IDR mới báo hiệu rằng đoạn video trước đó đã kết thúc, và đoạn mới bắt đầu. Mỗi đơn vị NAL bắt đầu với tiêu đề 1 byte, sau đó là phụ tải có ích theo trình tự thô (Raw Byte Sequence Payload: RBSP). RBSP chứa các lát được mã hóa. Các lát được lập mã nhị phân, nên chúng có thể được đệm bằng các bit 0 để đảm bảo rằng chiều dài là một số nguyên của byte. Lát bao gồm các tiêu đề lát và dữ liệu lát. Dữ liệu lát được chỉ rõ dưới dạng dãy các CU.

Khái niệm PH đã được thông qua trong cuộc họp JVET lần thứ 16 để truyền được một lần mỗi bức ảnh với tư cách là đơn vị NAL của VCL đầu tiên của hình ảnh. Nó cũng được đề xuất để nhóm vài phần tử cú pháp trước đó trong tiêu đề lát với tiêu đề hình ảnh. Các phần tử cú pháp mà về mặt chức năng chỉ cần được truyền một lần mỗi bức ảnh có thể được chuyển đến tiêu đề hình ảnh thay cho được truyền nhiều lần trong lát cho mỗi hình ảnh thu được.

Trong mô tả đặc điểm kỹ thuật VVC, các bảng cú pháp chỉ rõ tập cha cú pháp của tất cả các luồng bit được phép. Ràng buộc bổ sung trên cú pháp có thể được chỉ rõ, hoặc trực tiếp hoặc gián tiếp, trong các mệnh đề khác. Bảng 1 dưới đây là bảng cú pháp của tiêu đề lát và tiêu đề hình ảnh trong VVC. Ngữ nghĩa của một số cú pháp cũng được minh họa sau bảng cú pháp.

Bảng 1

<code>slice_header() {</code>	Bộ mô tả
<code>picture_header_in_slice_header_flag</code>	<code>u(1)</code>
<code>if(picture_header_in_slice_header_flag)</code>	
<code>picture_header_structure()</code>	
<code>if(subpic_info_present_flag)</code>	
<code>slice_subpic_id</code>	<code>u(v)</code>

<pre> if((rect_slice_flag && NumSlicesInSubpic[CurrSubpicIdx] > 1) (!rect_slice_flag && NumTilesInPic > 1)) </pre>	
slice_address	u(v)
<pre> for(i = 0; i < NumExtraShBits; i++) </pre>	
sh_extra_bit[i]	u(1)
<pre> if(!rect_slice_flag && NumTilesInPic > 1) </pre>	
num_tiles_in_slice_minus1	ue(v)
<pre> if(ph_inter_slice_allowed_flag) </pre>	
slice_type	ue(v)
<pre> if(sps_alf_enabled_flag && !alf_info_in_ph_flag) { </pre>	
slice_alf_enabled_flag	u(1)
<pre> if(slice_alf_enabled_flag) { </pre>	
slice_num_alf_aps_ids_luma	u(3)
<pre> for(i = 0; i < slice_num_alf_aps_ids_luma; i++) </pre>	
slice_alf_aps_id_luma[i]	u(3)
<pre> if(ChromaArrayType != 0) </pre>	
slice_alf_chroma_idc	u(2)
<pre> if(slice_alf_chroma_idc) </pre>	
slice_alf_aps_id_chroma	u(3)
<pre> if(sps_ccalf_enabled_flag) { </pre>	

slice_cc_alf_cb_enabled_flag	u(1)
if(slice_cc_alf_cb_enabled_flag)	
slice_cc_alf_cb_aps_id	u(3)
slice_cc_alf_cr_enabled_flag	u(1)
if(slice_cc_alf_cr_enabled_flag)	
slice_cc_alf_cr_aps_id	u(3)
}	
}	
}	
if(separate_colour_plane_flag == 1)	
colour_plane_id	u(2)
if(!rpl_info_in_ph_flag && ((nal_unit_type != IDR_W_RADL && nal_unit_type != IDR_N_LP) sps_idr_rpl_present_flag))	
ref_pic_lists()	
if((rpl_info_in_ph_flag ((nal_unit_type != IDR_W_RADL && nal_unit_type != IDR_N_LP) sps_idr_rpl_present_flag)) && ((slice_type != I && num_ref_entries[0][RplsIdx[0]] > 1) (slice_type == B && num_ref_entries[1][RplsIdx[1]] > 1))) {	
num_ref_idx_active_override_flag	u(1)

if(num_ref_idx_active_override_flag)	
for(i = 0; i < (slice_type == B ? 2 : 1); i++)	
if(num_ref_entries[i][RplIdx[i]] > 1)	
num_ref_idx_active_minus1[i]	ue(v)
}	
if(slice_type != I) {	
if(cabac_init_present_flag)	
cabac_init_flag	u(1)
if(ph_temporal_mvp_enabled_flag && !rpl_info_in_ph_flag) {	
if(slice_type == B)	
slice_collocated_from_l0_flag	u(1)
if((slice_collocated_from_l0_flag && NumRefIdxActive[0] > 1) (! slice_collocated_from_l0_flag && NumRefIdxActive[1] > 1))	
slice_collocated_ref_idx	ue(v)
}	
if(!wp_info_in_ph_flag && ((pps_weighted_pred_flag && slice_type == P) (pps_weighted_bipred_flag && slice_type == B)))	
pred_weight_table()	
}	

if(!qp_delta_info_in_ph_flag)	
slice_qp_delta	se(v)
if(pps_slice_chroma_qp_delta_present_flag) {	
slice_cb_qp_offset	se(v)
slice_cr_qp_offset	se(v)
if(sps_joint_cbr_enabled_flag)	
slice_joint_cbr_qp_offset	se(v)
}	
if(pps_cu_chroma_qp_offset_list_enabled_flag)	
cu_chroma_qp_offset_enabled_flag	u(1)
if(sps_sao_enabled_flag && !sao_info_in_ph_flag) {	
slice_sao_luma_flag	u(1)
if(ChromaArrayType != 0)	
slice_sao_chroma_flag	u(1)
}	
if(deblocking_filter_override_enabled_flag && !dbf_info_in_ph_flag)	
slice_deblocking_filter_override_flag	u(1)
if(slice_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)
slice_cb_beta_offset_div2	se(v)
slice_cb_tc_offset_div2	se(v)
slice_cr_beta_offset_div2	se(v)
slice_cr_tc_offset_div2	se(v)
}	
}	
slice_ts_residual_coding_disabled_flag	u(1)
if(ph_lmcs_enabled_flag)	
slice_lmcs_enabled_flag	u(1)
if(ph_explicit_scaling_list_enabled_flag)	
slice_explicit_scaling_list_used_flag	u(1)
if(NumEntryPoints > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < NumEntryPoints; i++)	
entry_point_offset_minus1[i]	u(v)
}	
if(slice_header_extension_present_flag) {	
slice_header_extension_length	ue(v)

for(i = 0; i < slice_header_extension_length; i++)	
slice_header_extension_data_byte[i]	u(8)
}	

byte_alignment()	
}	
picture_header_structure() {	Descriptor
gdr_or_irap_pic_flag	u(1)
if(gdr_or_irap_pic_flag)	
gdr_pic_flag	u(1)
ph_inter_slice_allowed_flag	u(1)
if(ph_inter_slice_allowed_flag)	
ph_intra_slice_allowed_flag	u(1)
non_reference_picture_flag	u(1)
ph_pic_parameter_set_id	ue(v)
ph_pic_order_cnt_lsb	u(v)
if(gdr_or_irap_pic_flag)	
no_output_of_prior_pics_flag	u(1)
if(gdr_pic_flag)	
recovery_poc_cnt	ue(v)
for(i = 0; i < NumExtraPhBits; i++)	
ph_extra_bit[i]	u(1)
if(sps_poc_msb_flag) {	
ph_poc_msb_present_flag	u(1)

if(ph_poc_msb_present_flag)	
poc_msb_val	u(v)
}	
if(sps_alf_enabled_flag && alf_info_in_ph_flag) {	
ph_alf_enabled_flag	u(1)
if(ph_alf_enabled_flag) {	
ph_num_alf_aps_ids_luma	u(3)
for(i = 0; i < ph_num_alf_aps_ids_luma; i++)	
ph_alf_aps_id_luma[i]	u(3)
if(ChromaArrayType != 0)	
ph_alf_chroma_idc	u(2)
if(ph_alf_chroma_idc > 0)	

ph_alf_aps_id_chroma	u(3)
if(sps_ccalf_enabled_flag) {	
ph_cc_alf_cb_enabled_flag	u(1)
if(ph_cc_alf_cb_enabled_flag)	
ph_cc_alf_cb_aps_id	u(3)
ph_cc_alf_cr_enabled_flag	u(1)
if(ph_cc_alf_cr_enabled_flag)	
ph_cc_alf_cr_aps_id	u(3)
}	
}	
}	
if(sps_lmcs_enabled_flag) {	
ph_lmcs_enabled_flag	u(1)
if(ph_lmcs_enabled_flag) {	
ph_lmcs_aps_id	u(2)
if(ChromaArrayType != 0)	
ph_chroma_residual_scale_flag	u(1)
}	
}	
if(sps_explicit_scaling_list_enabled_flag) {	

ph_explicit_scaling_list_enabled_flag	u(1)
if(ph_explicit_scaling_list_enabled_flag)	
ph_scaling_list_aps_id	u(3)
}	
if(sps_virtual_boundaries_enabled_flag && !sps_virtual_boundaries_present_flag) {	
ph_virtual_boundaries_present_flag	u(1)
if(ph_virtual_boundaries_present_flag) {	
ph_num_ver_virtual_boundaries	u(2)
for(i = 0; i < ph_num_ver_virtual_boundaries; i++)	
ph_virtual_boundaries_pos_x[i]	u(13)
ph_num_hor_virtual_boundaries	u(2)
for(i = 0; i < ph_num_hor_virtual_boundaries; i++)	
ph_virtual_boundaries_pos_y[i]	u(13)
}	
}	
if(output_flag_present_flag)	
pic_output_flag	u(1)
if(rpl_info_in_ph_flag)	
ref_pic_lists()	
if(partition_constraints_override_enabled_flag)	

partition_constraints_override_flag	u(1)
if(ph_intra_slice_allowed_flag) {	
if(partition_constraints_override_flag) {	
ph_log2_diff_min_qt_min_cb_intra_slice_luma	ue(v)
ph_max_mtt_hierarchy_depth_intra_slice_luma	ue(v)
if(ph_max_mtt_hierarchy_depth_intra_slice_luma != 0) {	
ph_log2_diff_max_bt_min_qt_intra_slice_luma	ue(v)
ph_log2_diff_max_tt_min_qt_intra_slice_luma	ue(v)
}	
if(qtbtt_dual_tree_intra_flag) {	
ph_log2_diff_min_qt_min_cb_intra_slice_chroma	ue(v)
ph_max_mtt_hierarchy_depth_intra_slice_chroma	ue(v)
if(ph_max_mtt_hierarchy_depth_intra_slice_chroma != 0) {	
ph_log2_diff_max_bt_min_qt_intra_slice_chroma	ue(v)
ph_log2_diff_max_tt_min_qt_intra_slice_chroma	ue(v)
}	
}	
}	
if(cu_qp_delta_enabled_flag)	
ph_cu_qp_delta_subdiv_intra_slice	ue(v)

if(pps_cu_chroma_qp_offset_list_enabled_flag)	
ph_cu_chroma_qp_offset_subdiv_intra_slice	ue(v)
}	
if(ph_inter_slice_allowed_flag) {	
if(partition_constraints_override_flag) {	
ph_log2_diff_min_qt_min_cb_inter_slice	ue(v)
ph_max_mtt_hierarchy_depth_inter_slice	ue(v)
if(ph_max_mtt_hierarchy_depth_inter_slice != 0) {	
ph_log2_diff_max_bt_min_qt_inter_slice	ue(v)
ph_log2_diff_max_tt_min_qt_inter_slice	ue(v)
}	
}	
if(cu_qp_delta_enabled_flag)	
ph_cu_qp_delta_subdiv_inter_slice	ue(v)
if(pps_cu_chroma_qp_offset_list_enabled_flag)	
ph_cu_chroma_qp_offset_subdiv_inter_slice	ue(v)
if(sps_temporal_mvp_enabled_flag) {	
ph_temporal_mvp_enabled_flag	u(1)
if(ph_temporal_mvp_enabled_flag && rpl_info_in_ph_flag) {	
ph_collocated_from_l0_flag	u(1)

<pre> if((ph_collocated_from_l0_flag && num_ref_entries[0][RplIdx[0]] > 1) (!ph_collocated_from_l0_flag && num_ref_entries[1][RplIdx[1]] > 1)) </pre>	
ph_collocated_ref_idx	ue(v)
}	
}	
mvd_l1_zero_flag	u(1)
if(sps_fpel_mmvd_enabled_flag)	
ph_fpel_mmvd_enabled_flag	u(1)
if(sps_bdof_pic_present_flag)	
ph_disable_bdof_flag	u(1)
if(sps_dmvr_pic_present_flag)	
ph_disable_dmvr_flag	u(1)
if(sps_prof_pic_present_flag)	
ph_disable_prof_flag	u(1)
if((pps_weighted_pred_flag pps_weighted_bipred_flag) && wp_info_in_ph_flag)	
pred_weight_table()	
}	
if(qp_delta_info_in_ph_flag)	
ph_qp_delta	se(v)

if(sps_joint_cbr_enabled_flag)	
ph_joint_cbr_sign_flag	u(1)
if(sps_sao_enabled_flag && sao_info_in_ph_flag) {	
ph_sao_luma_enabled_flag	u(1)
if(ChromaArrayType != 0)	
ph_sao_chroma_enabled_flag	u(1)
}	
if(sps_dep_quant_enabled_flag)	
ph_dep_quant_enabled_flag	u(1)
if(sps_sign_data_hiding_enabled_flag && !ph_dep_quant_enabled_flag)	
pic_sign_data_hiding_enabled_flag	u(1)
if(deblocking_filter_override_enabled_flag && dbf_info_in_ph_flag) {	
ph_deblocking_filter_override_flag	u(1)
if(ph_deblocking_filter_override_flag) {	
ph_deblocking_filter_disabled_flag	u(1)
if(!ph_deblocking_filter_disabled_flag) {	
ph_beta_offset_div2	se(v)
ph_tc_offset_div2	se(v)
ph_cb_beta_offset_div2	se(v)

ph_cb_tc_offset_div2	se(v)
ph_cr_beta_offset_div2	se(v)
ph_cr_tc_offset_div2	se(v)
}	
}	
}	
if(picture_header_extension_present_flag) {	
ph_extension_length	ue(v)
for(i = 0; i < ph_extension_length; i++)	
ph_extension_data_byte[i]	u(8)
}	
}	

Ngữ nghĩa của các phần tử cú pháp đã chọn

ph_temporal_mvp_enabled_flag chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên ảnh cho các lát (slice) kết hợp với PH hay không. Nếu **ph_temporal_mvp_enabled_flag** bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác, nếu **ph_temporal_mvp_enabled_flag** bằng 1, biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của **ph_temporal_mvp_enabled_flag** được suy ra là bằng 0. Khi không có hình ảnh tham chiếu trong đệm hình ảnh được giải mã (Decoded Picture Buffer: DPB) có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của **ph_temporal_mvp_enabled_flag** sẽ bằng 0.

Số lượng tối đa của ứng viên MVP trộn dựa vào khối con, `MaxNumSubblockMergeCand`, được suy ra như sau:

$$\begin{aligned} &\text{Nếu } (\text{sps_affine_enabled_flag}) \\ &\quad \text{MaxNumSubblockMergeCand} = 5 - \text{five_minus_max_num_subblock_merge_cand} \end{aligned} \quad (87)$$

Hoặc là

`MaxNumSubblockMergeCand = sps_sbtmvp_enabled_flag && ph_temporal_mvp_enabled_flag;`

Trong đó giá trị của `MaxNumSubblockMergeCand` sẽ trong phạm vi bao gồm 0 đến 5 (bao gồm cả 0 và 5).

`slice_collocated_from_10_flag` bằng 1 chỉ rõ rằng hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian được suy ra từ danh mục hình ảnh tham chiếu 0. `slice_collocated_from_10_flag` bằng 0 chỉ rõ rằng hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian được suy ra từ danh mục hình ảnh tham chiếu 1.

Khi `slice_type` bằng B hoặc P, `ph_temporal_mvp_enabled_flag` bằng 1, và `slice_collocated_from_10_flag` không có mặt, áp dụng điều sau:

- Nếu `rpl_info_in_ph_flag` bằng 1, `slice_collocated_from_10_flag` được suy ra là bằng `ph_collocated_from_10_flag`.
- Trường hợp khác (`rpl_info_in_ph_flag` bằng 0 và `slice_type` bằng P), giá trị của `slice_collocated_from_10_flag` được suy ra là bằng 1.

`slice_collocated_ref_idx` chỉ rõ chỉ số tham chiếu của hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian.

Khi `slice_type` bằng P hoặc khi `slice_type` bằng B và `slice_collocated_from_10_flag` bằng 1, `slice_collocated_ref_idx` để chỉ mục nhập trong danh mục hình ảnh tham chiếu 0, và giá trị của `slice_collocated_ref_idx` sẽ trong phạm vi bao gồm từ 0 đến `NumRefIdxActive[0] - 1`.

Khi `slice_type` bằng B và `slice_collocated_from_10_flag` bằng 0, `slice_collocated_ref_idx` để chỉ mục nhập trong danh mục hình ảnh tham chiếu 1, và giá

trị của `slice_collocated_ref_idx` sẽ trong phạm vi bao gồm từ 0 đến `NumRefIdxActive[ 1 ] - 1`.

Khi `slice_collocated_ref_idx` không có mặt, áp dụng điều sau:

- Nếu `rpl_info_in_ph_flag` bằng 1, giá trị của `slice_collocated_ref_idx` được suy ra là bằng `ph_collocated_ref_idx`.
- Trường hợp khác (`rpl_info_in_ph_flag` bằng 0), giá trị của `slice_collocated_ref_idx` được suy ra là bằng 0.

Là một yêu cầu của sự phụ hợp luồng bit rằng hình ảnh được đề cập bởi `slice_collocated_ref_idx` shall giống nhau cho tất cả các lát của hình ảnh được lập mã.

Là một yêu cầu của sự phụ hợp luồng bit rằng các giá trị của `pic_width_in_luma_samples` và `pic_height_in_luma_samples` của hình ảnh tham chiếu được đề cập bởi `slice_collocated_ref_idx` sẽ bằng các giá trị của `pic_width_in_luma_samples` và `pic_height_in_luma_samples` tương ứng của hình ảnh hiện thời và `RprConstraintsActive[slice_collocated_from_l0_flag ? 0:1][slice_collocated_ref_idx]` sẽ bằng 0.

Giá trị của `RprConstraintsActive[i][j]` được suy ra trong phần 8.3.2 trong mô tả đặc điểm kỹ thuật VVC. Phép suy ra của giá trị của `RprConstraintsActive[i][j]` được mô tả dưới đây.

Quy trình giải mã cho sự cấu trúc các danh mục hình ảnh tham chiếu

Quy trình giải mã cho tái cấu trúc các danh mục hình ảnh tham chiếu được gọi ở thời điểm bắt đầu quy trình giải mã cho từng lát của hình ảnh không IDR.

Hình ảnh tham chiếu được đề cập đến thông qua các chỉ số tham chiếu. Chỉ số tham chiếu là chỉ số vào trong danh mục hình ảnh tham chiếu. Khi giải mã lát I, không có danh mục hình ảnh tham chiếu nào được sử dụng trong giải mã của dữ liệu lát. Khi giải mã lát P, chỉ danh mục hình ảnh tham chiếu 0 (tức là, `RefPicList[ 0 ]`), được sử dụng trong giải mã dữ liệu lát. Khi giải mã lát B, cả danh mục hình ảnh tham chiếu 0 và danh mục hình ảnh tham chiếu 1 (tức là, `RefPicList[ 1 ]`) được sử dụng trong giải mã của dữ liệu lát.

Tại thời điểm bắt đầu quy trình giải mã đối với mỗi lát của hình ảnh không IDR, danh mục hình ảnh tham chiếu `RefPicList[ 0 ]` và `RefPicList[ 1 ]` được suy ra. Danh mục

hình ảnh tham chiếu được sử dụng trong đánh dấu hình ảnh tham chiếu được chỉ rõ trong tiêu chuẩn lập mã video hoặc trong giải mã của dữ liệu slice.

Đối với slice I của hình ảnh không IDR mà nó không phải là lát thứ nhất của hình ảnh, RefPicList[0] và RefPicList[1] có thể được suy ra cho mục đích kiểm tra sự phù hợp của luồng bit, nhưng phép suy ra của nó không cần thiết để giải mã hình ảnh hiện thời hoặc hình ảnh sau hình ảnh hiện thời trong thứ tự giải mã. Đối với slice P mà nó không phải là lát thứ nhất của hình ảnh, RefPicList[1] có thể được suy ra cho mục đích kiểm tra sự phù hợp của luồng bit, nhưng phép suy ra của nó không cần thiết để giải mã hình ảnh hiện thời hoặc hình ảnh sau hình ảnh hiện thời trong thứ tự giải mã.

Danh mục hình ảnh tham chiếu RefPicList[0] và RefPicList[1], tỷ lệ chia hình ảnh tham chiếu RefPicScale[i][j][0] và RefPicScale[i][j][1], và các cờ được chia tỷ lệ của hình ảnh tham chiếu RprConstraintsActive[0][j] và RprConstraintsActive[1][j] được suy ra như sau:

```

for( i = 0; i < 2; i++ ) {
    for( j = 0, k = 0, pocBase = PicOrderCntVal; j <
        num_ref_entries[ i ][ RplsIdx[ i ] ]; j++ ) {
        if( !inter_layer_ref_pic_flag[ i ][ RplsIdx[ i ] ][ j ] ) {
            if( st_ref_pic_flag[ i ][ RplsIdx[ i ] ][ j ] ) {
                RefPicPocList[ i ][ j ] =
                pocBase - DeltaPocValSt[ i ][ RplsIdx[ i ] ][ j ]
                if( có hình ảnh tham chiếu picA trong DPB với cùng một nuh_layer_id
                    dưới dạng hình ảnh hiện thời
                    và PicOrderCntVal bằng RefPicPocList[ i ][ j ] )
                    RefPicList[ i ][ j ] = picA
                hoặc là
                    RefPicList[ i ][ j ] = "không có hình ảnh tham chiếu"
                pocBase = RefPicPocList[ i ][ j ]
            } else {
                if( !delta_poc_msb_cycle_lt[ i ][ k ] ) {
                    if( có hình ảnh tham chiếu picA trong DPB với cùng một
                        nuh_layer_id dưới dạng hình ảnh hiện thời và
                        PicOrderCntVal & ( MaxPicOrderCntLsb - 1 )
                    bằng

```

```

PocLsbLt[ i ][ k ] )
    RefPicList[ i ][ j ] = picA
    hoặc là
    RefPicList[ i ][ j ] = "không có hình ảnh tham chiếu"
    RefPicLtPocList[ i ][ j ] = PocLsbLt[ i ][ k ]
} else {
    if( có hình ảnh tham chiếu picA trong DPB với cùng một
    nuh_layer_id dưới dạng hình ảnh hiện thời và
        PicOrderCntVal bằng FullPocLt[ i ][ k ] )
        RefPicList[ i ][ j ] = picA
        hoặc là
        RefPicList[ i ][ j ] = "không có hình ảnh tham chiếu"
        RefPicLtPocList[ i ][ j ] = FullPocLt[ i ][ k ]
    }
    k++
}
} hoặc là {
    layerIdx =
DirectRefLayerIdx[ GeneralLayerIdx[ nuh_layer_id ] ][ ilrp_idx[ i ][ RplsIdx ][ j
]]
    refPicLayerId = vps_layer_id[ layerIdx ]
    if( có hình ảnh tham chiếu picA trong DPB với nuh_layer_id bằng
    refPicLayerId và
        cùng một PicOrderCntVal dưới dạng hình ảnh hiện thời )
        RefPicList[ i ][ j ] = picA
        hoặc là
        RefPicList[ i ][ j ] = "không có hình ảnh tham chiếu"
    }
    fRefWidth được thiết lập bằng PicOutputWidthL của hình ảnh tham chiếu
    RefPicList[ i ][ j ]
    fRefHeight được thiết lập bằng PicOutputHeightL của hình ảnh tham chiếu
    RefPicList[ i ][ j ]

```


refPicWidth, refPicHeight, refScalingWinLeftOffset,
 refScalingWinRightOffset, refScalingWinTopOffset,
 và refScalingWinBottomOffset, được thiết lập bằng các giá trị của
 pic_width_in_luma_samples,
 pic_height_in_luma_samples, scaling_win_left_offset,
 scaling_win_right_offset,
 scaling_win_top_offset, và scaling_win_bottom_offset, respectively, của
 hình ảnh tham chiếu
 RefPicList[i][j]

```

RefPicScale[ i ][ j ][ 0 ]=( ( fRefWidth << 14 ) + ( PicOutputWidthL >> 1 )
) / PicOutputWidthL
RefPicScale[ i ][ j ][ 1 ]=( ( fRefHeight << 14 ) + ( PicOutputHeightL >> 1
) ) / PicOutputHeightL
RprConstraintsActive[ i ][ j ]=( pic_width_in_luma_samples !=
refPicWidth ||
pic_height_in_luma_samples != refPicHeight ||
scaling_win_left_offset != refScalingWinLeftOffset ||
scaling_win_right_offset != refScalingWinRightOffset ||
scaling_win_top_offset != refScalingWinTopOffset ||
scaling_win_bottom_offset != refScalingWinBottomOffset )
}
}

```

scaling_win_left_offset, scaling_win_right_offset, scaling_win_top_offset, và scaling_win_bottom_offset chỉ rõ độ lệch mà được áp dụng cho kích thước hình ảnh để chọn thang chia tính toán tỷ lệ. Khi không có mặt, các giá trị của **scaling_win_left_offset, scaling_win_right_offset, scaling_win_top_offset, và scaling_win_bottom_offset** được suy ra bằng **pps_conf_win_left_offset, pps_conf_win_right_offset, pps_conf_win_top_offset, và pps_conf_win_bottom_offset** tương ứng.

Giá trị của $\text{SubWidthC} * (\text{scaling_win_left_offset} + \text{scaling_win_right_offset})$ sẽ nhỏ hơn $\text{pic_width_in_luma_samples}$, và giá trị của $\text{SubHeightC} * (\text{scaling_win_top_offset} + \text{scaling_win_bottom_offset})$ sẽ nhỏ hơn $\text{pic_height_in_luma_samples}$.

Các biến PicOutputWidthL và PicOutputHeightL được suy ra như sau:

$$\begin{aligned} \text{PicOutputWidthL} &= \text{pic_width_in_luma_samples} - \\ &\quad \text{SubWidthC} * (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}) \\ \text{PicOutputHeightL} &= \text{pic_height_in_luma_samples} - \\ &\quad \text{SubHeightC} * (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}). \end{aligned} \tag{78}$$

Lấy $\text{refPicOutputWidthL}$ và $\text{refPicOutputHeightL}$ tương ứng là PicOutputWidthL và PicOutputHeightL của hình ảnh tham chiếu của hình ảnh hiện thời để chỉ PPS này. Là một yêu cầu của sự phù hợp luồng bit rằng tất cả các điều kiện sau được thỏa mãn:

- $\text{PicOutputWidthL} * 2$ sẽ lớn hơn hoặc bằng $\text{refPicWidthInLumaSamples}$.
- $\text{PicOutputHeightL} * 2$ sẽ lớn hơn hoặc bằng $\text{refPicHeightInLumaSamples}$.
- PicOutputWidthL sẽ nhỏ hơn hoặc bằng $\text{refPicWidthInLumaSamples} * 8$.
- PicOutputHeightL sẽ nhỏ hơn hoặc bằng $\text{refPicHeightInLumaSamples} * 8$.
- $\text{PicOutputWidthL} * \text{pic_width_max_in_luma_samples}$ sẽ lớn hơn hoặc bằng $\text{refPicOutputWidthL} * (\text{pic_width_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY}))$.
- $\text{PicOutputHeightL} * \text{pic_height_max_in_luma_samples}$ sẽ lớn hơn hoặc bằng $\text{refPicOutputHeightL} * (\text{pic_height_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY}))$.

Trong VVC hiện thời, **mvd_l1_zero_flag** được báo hiệu trong PH mà không có ràng buộc điều kiện nào. Tuy nhiên, đặc điểm được điều khiển bằng cờ **mvd_l1_zero_flag** chỉ có thể áp dụng khi slice là slice dự đoán hai chiều slice (B-slice). Do đó, báo hiệu cờ là thừa khi slice kết hợp với tiêu đề hình ảnh không phải là slice B.

Trong ví dụ khác, **ph_disable_bdof_flag** và **ph_disable_dmvr_flag** được báo hiệu trong PH chỉ khi cờ kích hoạt tương ứng (**sps_bdof_pic_present_flag**, **sps_dmvr_pic_present_flag**) được báo hiệu trong tập hợp tham số chuỗi (sequence parameter set: SPS) tương ứng là đúng. Tuy nhiên, như được thể hiện trong bảng 2 dưới đây, các đặc điểm được điều khiển bởi cờ **ph_disable_bdof_flag** và

ph_disable_dmvr_flag chỉ có thể áp dụng khi slice là slice dự đoán hai chiều (B-slice). Do đó, báo hiệu của hai cờ này là thừa hoặc vô ích khi slice kết hợp với tiêu đề hình ảnh không phải là slice B.

Bảng 2

if(sps_bdof_pic_present_flag)
ph_disable_bdof_flag
if(sps_dmvr_pic_present_flag)
ph_disable_dmvr_flag

Thêm một ví dụ cũng có thể thấy về phần tử cú pháp **ph_collocated_from_l0_flag** để chỉ hình ảnh được sắp xếp là từ list0 hoặc list1. Và ví dụ khác có thể thấy trên cú pháp **pred_weight_table()** mà là phần tử cú pháp liên quan đến bảng trọng số cho phép dự đoán hai chiều, như được thể hiện dưới đây.

if(ph_temporal_mvp_enabled_flag && rpl_info_in_ph_flag) {
ph_collocated_from_l0_flag

if((pps_weighted_pred_flag pps_weighted_bipred_flag) && wp_info_in_ph_flag)
pred_weight_table()

pred_weight_table() {	Bộ mô tả
luma_log2_weight_denom	ue(v)
if(ChromaArrayType != 0)	
delta_chroma_log2_weight_denom	se(v)
if(wp_info_in_ph_flag)	
num_l0_weights	ue(v)
for(i = 0; i < NumWeightsL0; i++)	
luma_weight_l0_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumWeightsL0; i++)	
chroma_weight_l0_flag[i]	u(1)
for(i = 0; i < NumWeightsL0; i++) {	
if(luma_weight_l0_flag[i]) {	
delta_luma_weight_l0[i]	se(v)
luma_offset_l0[i]	se(v)
}	
if(chroma_weight_l0_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l0[i][j]	se(v)
delta_chroma_offset_l0[i][j]	se(v)

}	
}	
if(pps_weighted_bipred_flag && wp_info_in_ph_flag)	
num_l1_weights	ue(v)
for(i = 0; i < NumWeightsL1; i++)	
luma_weight_l1_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumWeightsL1; i++)	
chroma_weight_l1_flag[i]	u(1)
for(i = 0; i < NumWeightsL1; i++) {	
if(luma_weight_l1_flag[i]) {	
delta_luma_weight_l1[i]	se(v)
luma_offset_l1[i]	se(v)
}	
if(chroma_weight_l1_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l1[i][j]	se(v)
delta_chroma_offset_l1[i][j]	se(v)
}	
}	

}	
---	--

Vấn đề thứ ba là kết hợp với cú pháp `ph_temporal_mvp_enabled_flag`. Trong VVC hiện thời, bởi vì độ phân giải của hình ảnh được sắp xếp được lựa chọn cho phép suy ra TMVP sẽ giống với độ phân giải của hình ảnh hiện thời, có sự ràng buộc phù hợp luồng bit để kiểm tra giá trị của `ph_temporal_mvp_enabled_flag` như được minh họa dưới đây.

Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

Tuy nhiên, trong VVC hiện thời, không chỉ độ phân giải của hình ảnh được sắp xếp sẽ ảnh hưởng đến sự kích hoạt của TMVP, mà cả độ lệch mà được áp dụng cho kích thước hình ảnh để chọn thang chia tính toán tỷ lệ ảnh hưởng đến khả năng kích hoạt của TMVP. Tuy nhiên, trong VVC hiện thời, độ lệch không được xem xét trong sự phù hợp luồng bit của **`ph_temporal_mvp_enabled_flag`**.

Ngoài ra, có yêu cầu về sự phù hợp luồng bit là hình ảnh được đề cập bởi **`slice_collocated_ref_idx`** sẽ giống cho tất cả các slice của hình ảnh được lập mã. Tuy nhiên, khi hình ảnh được lập mã có nhiều slice và không có hình ảnh tham chiếu chung tồn tại trong số tất cả các slice này, sự phù hợp luồng bit không có cơ hội được đáp ứng. Và trong trường hợp này, **`ph_temporal_mvp_enabled_flag`** nên được ràng buộc bằng 0.

Vài phương pháp được đề xuất nhằm vào vấn đề được mô tả ở trên. Phương pháp được đề xuất có thể được áp dụng độc lập hoặc kết hợp.

Vì các đặc điểm được điều khiển bằng cờ **`mvd_l1_zero_flag`**, **`ph_disable_bdof_flag`** và **`ph_disable_dmvr_flag`** chỉ có thể áp dụng được khi slice là slice dự đoán hai chiều (slice B), theo phương pháp của sáng chế, đề xuất báo hiệu cờ này chỉ khi các slice kết hợp là các slice B. Lưu ý rằng khi danh mục hình ảnh tham chiếu được báo hiệu trong PH (ví dụ, `rpl_info_in_ph_flag=1`), có nghĩa là tất cả các lát của hình ảnh lập mã sử dụng cùng một hình ảnh tham chiếu được báo hiệu trong PH. Do đó, khi danh mục hình ảnh tham chiếu được báo hiệu trong PH và các danh mục hình ảnh tham chiếu được báo hiệu chỉ ra rằng hình ảnh hiện thời không phải là dự đoán hai

chiều, cờ `mvd_l1_zero_flag`, `ph_disable_bdof_flag` và `ph_disable_dmvr_flag` cần không được báo hiệu.

Trong vài ví dụ, vài điều kiện được bổ sung vào tập hợp cú pháp này trong PH để ngăn chặn báo hiệu dư thừa hoặc hành vi giải mã không xác định do các giá trị được gửi không đúng đối với một số cú pháp trong tiêu đề hình ảnh. Vài ví dụ được minh họa dưới đây, trong đó các biến `num_ref_entries[i][ RplIdx[ i ] ]` thể hiện số lượng của hình ảnh tham chiếu trong danh mục `i`.

Trong vài ví dụ, điều kiện có thể được thể hiện như dưới đây:

Nếu $(\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[0][\text{RplIdx}[0]] > 1 \ \&\& \ \text{num_ref_entries}[1][\text{RplIdx}[1]] > 1)$

`mvd_l1_zero_flag`

Trong vài ví dụ, điều kiện có thể được thể hiện như dưới đây:

Nếu $(\neg \text{rpl_info_in_ph_flag} \ \parallel \ (\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[0][\text{RplIdx}[0]] > 1 \ \&\& \ \text{num_ref_entries}[1][\text{RplIdx}[1]] > 1))$

`mvd_l1_zero_flag`

Trong vài ví dụ, điều kiện có thể được thể hiện như dưới đây:

Nếu $(\neg \text{rpl_info_in_ph_flag} \ \parallel \ (\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[0][\text{RplIdx}[0]] > 0 \ \&\& \ \text{num_ref_entries}[1][\text{RplIdx}[1]] > 0))$

`mvd_l1_zero_flag`

Theo cách khác, các điều kiện có thể được viết dưới dạng nhỏ gọn hơn, mang lại kết quả tương tự. Bởi vì slice dự đoán hai chiều (slice B) hoặc hình ảnh dự đoán hai chiều phải có ít nhất một hình ảnh tham chiếu list1, nó có thể chỉ kiểm tra liệu hình ảnh/slice hiện thời có hình ảnh tham chiếu list1 hay không. Ví dụ về kiểm tra điều kiện thay thế được minh họa dưới đây:

Nếu $(\neg \text{rpl_info_in_ph_flag} \ \parallel \ (\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[1][\text{RplIdx}[1]] > 0))$

`mvd_l1_zero_flag`

Ngữ nghĩa của `mvd_l1_zero_flag` cũng được biến đổi để xử lý trường hợp khi nó không được báo hiệu.

mvd_l1_zero_flag bằng 1 chỉ ra rằng cấu trúc cú pháp `mvd_coding( x0, y0, 1 )` không được phân tích và `MvdL1[ x0 ][ y0 ][ compIdx ]` và `MvdCpL1[ x0 ][ y0 ][ cpIdx ][ compIdx ]` được thiết lập bằng 0 cho `compIdx = 0..1` và `cpIdx = 0..2`. **mvd_l1_zero_flag** bằng 0 chỉ ra rằng cấu trúc cú pháp `mvd_coding( x0, y0, 1 )` được phân tích. Khi không có mặt, giá trị của **mvd_l1_zero_flag** được suy ra bằng 0.

Vài ví dụ về báo hiệu có điều kiện phần tử cú pháp **ph_disable_dmvr_flag** được minh họa dưới đây:

Nếu `(sps_dmvr_pic_present_flag && rpl_info_in_ph_flag && num_ref_entries[ 0 ][ RplsIdx[ 0 ] ] > 1 && num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] > 1 )`

ph_disable_dmvr_flag

hoặc nếu `(sps_dmvr_pic_present_flag && (!rpl_info_in_ph_flag || (rpl_info_in_ph_flag && num_ref_entries[ 0 ][ RplsIdx[ 0 ] ] > 1 && num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] > 1)))`

ph_disable_dmvr_flag

hoặc nếu `(sps_dmvr_pic_present_flag && (!rpl_info_in_ph_flag || (rpl_info_in_ph_flag && num_ref_entries[ 0 ][ RplsIdx[ 0 ] ] > 0 && num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] > 0)))`

ph_disable_dmvr_flag

Ví dụ khác về kiểm tra điều kiện thay thế được minh họa dưới đây:

Nếu `(sps_dmvr_pic_present_flag && (!rpl_info_in_ph_flag || (rpl_info_in_ph_flag && num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] > 0)))`

ph_disable_dmvr_flag

Ngữ nghĩa của **ph_disable_dmvr_flag** cũng được biến đổi để xử lý trường hợp khi mà nó không được báo hiệu.

ph_disable_dmvr_flag bằng 1 chỉ rõ rằng tinh chỉnh vector chuyển động của bộ giải mã dựa vào dự đoán kép liên ảnh được vô hiệu hóa trong các slice kết hợp với PH. **ph_disable_dmvr_flag** bằng 0 chỉ rõ rằng tinh chỉnh vector chuyển động của bộ giải mã

kết hợp dự đoán kép liên ảnh có thể hoặc không thể được kích hoạt trong các slice kết hợp với PH.

Khi `ph_disable_dmvr_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 0.
- Hoặc là nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 1, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.
- Trường hợp khác (`sps_dmvr_enabled_flag` bằng 0), giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Ví dụ khác để suy ra giá trị của `ph_disable_dmvr_flag` khi nó không có mặt được minh họa dưới đây:

Nếu tất cả điều kiện được xem xét cho phép suy ra của giá trị của `ph_disable_dmvr_flag` khi nó hoặc được báo hiệu rõ ràng hoặc được suy ra hoàn toàn:

- Nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 0.
- Hoặc nếu `sps_dmvr_enabled_flag` bằng 0 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.
- Hoặc nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng X. (X được báo hiệu rõ ràng)
- Hoặc nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] > 0`, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng X. (X được báo hiệu rõ ràng)
- Hoặc (`sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] == 0`), giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Vì phần tử cú pháp `ph_disable_dmvr_flag` được báo hiệu rõ ràng dưới điều kiện thứ ba và thứ tư, chúng có thể được loại bỏ khỏi phép suy ra của `ph_disable_dmvr_flag` khi `ph_disable_dmvr_flag` không có mặt.

Khi `ph_disable_dmvr_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 0.
- Hoặc nếu `sps_dmvr_enabled_flag` bằng 0 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.
- Hoặc (`sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] == 0`), giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Điều kiện có thể được chỉnh sửa đơn giản hóa như sau:

Khi `ph_disable_dmvr_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_dmvr_enabled_flag` bằng 1 và `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 0.
- Trường hợp khác (`sps_dmvr_enabled_flag` bằng 0 or `sps_dmvr_pic_present_flag` bằng 1), giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Ví dụ khác để suy ra giá trị của `ph_disable_dmvr_flag` khi nó không có mặt được minh họa dưới đây:

Khi `ph_disable_dmvr_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1- `sps_dmvr_enabled_flag`.
- Hoặc nếu `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1- `sps_dmvr_enabled_flag`.
- Hoặc nếu `sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] > 0`, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1- `sps_dmvr_enabled_flag`.
- Hoặc (`sps_dmvr_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] == 0`), giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Trong vài ví dụ, vì phần tử cú pháp `ph_disable_dmvr_flag` được báo hiệu rõ ràng dưới điều kiện thứ hai và thứ ba nêu trên, nên chúng có thể được loại bỏ khỏi phép suy ra của `ph_disable_dmvr_flag` khi nó không có mặt.

Trong vài ví dụ, khi `ph_disable_dmvr_flag` không có mặt, áp dụng điều sau: nếu `sps_dmvr_pic_present_flag` bằng 0, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng `1 - sps_dmvr_enabled_flag`; mặt khác, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.

Vài ví dụ của báo hiệu có điều kiện phần tử cú pháp `ph_disable_bdof_flag` được minh họa dưới đây:

Nếu $(\text{sps_bdof_pic_present_flag} \ \&\& \ \text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[0][\text{RplsIdx}[0]] > 1 \ \&\& \ \text{num_ref_entries}[1][\text{RplsIdx}[1]] > 1)$

`ph_disable_bdof_flag`

hoặc nếu $(\text{sps_bdof_pic_present_flag} \ \&\& \ (!\text{rpl_info_in_ph_flag} \ || \ (\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[0][\text{RplsIdx}[0]] > 1 \ \&\& \ \text{num_ref_entries}[1][\text{RplsIdx}[1]] > 1))$

`ph_disable_bdof_flag`

Trong vài ví dụ, ví dụ về kiểm tra điều kiện thay thế được minh họa dưới đây:

Nếu $(\text{sps_bdof_pic_present_flag} \ \&\& \ (!\text{rpl_info_in_ph_flag} \ || \ (\text{rpl_info_in_ph_flag} \ \&\& \ \text{num_ref_entries}[1][\text{RplsIdx}[1]] > 0)))$

`ph_disable_bdof_flag`

Ngữ nghĩa của `ph_disable_bdof_flag` cũng được biến đổi để xử lý trường hợp khi mà nó không được báo hiệu.

`ph_disable_bdof_flag` bằng 1 chỉ rõ rằng dự đoán liên luồng quang hai chiều căn cứ vào dự đoán kép liên ảnh được vô hiệu hóa trong các slice kết hợp với PH. `ph_disable_bdof_flag` bằng 0 chỉ rõ rằng dự đoán liên luồng quang hai chiều căn cứ vào dự đoán kép liên ảnh có thể hoặc không thể được kích hoạt trong các slice kết hợp với PH.

Khi `ph_disable_bdof_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 0.
- Hoặc nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 1, giá trị của `ph_disable_dmvr_flag` được suy ra là bằng 1.
- Trường hợp khác (`sps_bdof_enabled_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Cách khác để suy ra giá trị của `ph_disable_bdof_flag` khi nó không có mặt được minh họa dưới đây:

Nếu tất cả các điều kiện được xem xét cho phép suy ra của giá trị của `ph_disable_bdof_flag` khi nó hoặc được báo hiệu rõ ràng hoặc được suy ra hoàn toàn:

- Nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 0.
- Hoặc nếu `sps_bdof_enabled_flag` bằng 0 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.
- Hoặc nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng X. (X được báo hiệu rõ ràng)
- Hoặc nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] > 0`, giá trị của `ph_disable_bdof_flag` được suy ra là bằng X. (X được báo hiệu rõ ràng)
- Hoặc (`sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplIdx[ 1 ] ] == 0`), giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Vì phần tử cú pháp `ph_disable_bdof_flag` được báo hiệu rõ ràng dưới điều kiện thứ ba và thứ tư, nên chúng có thể được loại bỏ khỏi phép suy ra của `ph_disable_bdof_flag` khi `ph_disable_bdof_flag` không có mặt:

Khi `ph_disable_bdof_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 0.

- Hoặc nếu `sps_bdof_enabled_flag` bằng 0 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.
- Hoặc (`sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] == 0`), giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Các điều kiện có thể được sửa đổi đơn giản hóa như sau:

Khi `ph_disable_bdof_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_bdof_enabled_flag` bằng 1 và `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 0.
- Trường hợp khác (`sps_bdof_enabled_flag` bằng 0 hoặc `sps_bdof_pic_present_flag` bằng 1), giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Cách khác để suy ra giá trị của `ph_disable_bdof_flag` khi nó không có mặt được minh họa dưới đây:

Khi `ph_disable_bdof_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1- `sps_bdof_enabled_flag`.
- Hoặc nếu `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1- `sps_bdof_enabled_flag`.
- Hoặc nếu `sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] > 0`, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1- `sps_bdof_enabled_flag`.
- Hoặc (`sps_bdof_pic_present_flag` bằng 1 và `rpl_info_in_ph_flag` bằng 1 và `num_ref_entries[ 1 ][ RplsIdx[ 1 ] ] == 0`), giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Trong vài ví dụ, vì phần tử cú pháp `ph_disable_bdof_flag` được báo hiệu rõ ràng dưới điều kiện thứ hai và thứ ba, nên chúng có thể được loại bỏ khỏi phép suy ra của `ph_disable_bdof_flag` khi nó không có mặt.

Khi `ph_disable_bdof_flag` không có mặt, áp dụng điều sau:

- Nếu `sps_bdof_pic_present_flag` bằng 0, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1- `sps_bdof_enabled_flag`..
- Trường hợp khác, giá trị của `ph_disable_bdof_flag` được suy ra là bằng 1.

Ngoài ra, điều kiện báo hiệu đối với phần tử cú pháp **`ph_collocated_from_l0_flag`** và `weight_table( )` được sửa đổi bởi vì hai loại phần tử cú pháp chỉ có thể được áp dụng khi các slice kết hợp là slice B. Ví dụ về báo hiệu phần tử cú pháp sửa đổi được minh họa dưới đây.

```
if( ph_temporal_mvp_enabled_flag && (num_ref_entries[ 0 ][ RplIdx[ 0 ] ] > 1
&& num_ref_entries[ 1 ][ RplIdx[ 1 ] ] > 1))) {
```

`ph_collocated_from_l0_flag`

Hoặc

```
if( ph_temporal_mvp_enabled_flag && (num_ref_entries[ 0 ][ RplIdx[ 0 ] ] > 0
&& num_ref_entries[ 1 ][ RplIdx[ 1 ] ] > 0))) {
```

`ph_collocated_from_l0_flag`

Ngữ nghĩa của `ph_collocated_from_l0_flag` cũng được sửa đổi để xử lý trường hợp khi mà nó không được báo hiệu.

`ph_collocated_from_l0_flag` bằng 1 chỉ rõ rằng hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian được suy ra từ danh mục hình ảnh tham chiếu 0. `ph_collocated_from_l0_flag` bằng 0 chỉ rõ rằng hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian được suy ra từ danh mục hình ảnh tham chiếu 1.

Khi `ph_collocated_from_l0_flag` không có mặt, áp dụng điều sau:

- Nếu `num_ref_entries[ 0 ][ RplIdx[ 0 ] ]` lớn hơn 1, giá trị của `ph_collocated_from_l0_flag` được suy ra bằng 1.
- Trường hợp khác (`num_ref_entries[ 1 ][ RplIdx[ 1 ] ]` lớn hơn 1), giá trị của `ph_collocated_from_l0_flag` được suy ra bằng 0.

if((pps_weighted_pred_flag pps_weighted_bipred_flag) && wp_info_in_ph_flag flag)

pred_weight_table()

pred_weight_table() {	Bộ mô tả
luma_log2_weight_denom	ue(v)
if(ChromaArrayType != 0)	
delta_chroma_log2_weight_denom	se(v)
if(wp_info_in_ph_flag)	
num_l0_weights	ue(v)
for(i = 0; i < NumWeightsL0; i++)	
luma_weight_l0_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumWeightsL0; i++)	
chroma_weight_l0_flag[i]	u(1)
for(i = 0; i < NumWeightsL0; i++) {	
if(luma_weight_l0_flag[i]) {	
delta_luma_weight_l0[i]	se(v)
luma_offset_l0[i]	se(v)
}	
if(chroma_weight_l0_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l0[i][j]	se(v)
delta_chroma_offset_l0[i][j]	se(v)

}	
}	
if(pps_weighted_bipred_flag && wp_info_in_ph_flag && (!rpl_info_in_ph_flag (rpl_info_in_ph_flag && num_ref_entries[0][RplsIdx[0]] > 0 && num_ref_entries[1][RplsIdx[1]] > 0)))	
num_l1_weights	ue(v)

for(i = 0; i < NumWeightsL1; i++)	
luma_weight_l1_flag[i]	u(1)
if(ChromaArrayType != 0)	
for(i = 0; i < NumWeightsL1; i++)	
chroma_weight_l1_flag[i]	u(1)
for(i = 0; i < NumWeightsL1; i++) {	
if(luma_weight_l1_flag[i]) {	
delta_luma_weight_l1[i]	se(v)
luma_offset_l1[i]	se(v)
}	
if(chroma_weight_l1_flag[i])	
for(j = 0; j < 2; j++) {	
delta_chroma_weight_l1[i][j]	se(v)
delta_chroma_offset_l1[i][j]	se(v)
}	
}	
}	

Tương tự, ví dụ về kiểm tra điều kiện thay thế được minh họa dưới đây:

```

if( pps_weighted_bipred_flag && wp_info_in_ph_flag &&
(!rpl_info_in_ph_flag || (rpl_info_in_ph_flag && num_ref_entries[ 1 ][ RplsIdx[ 1 ] ]
> 0)))

```

num_l1_weights

Ngữ nghĩa của phần tử cú pháp trong `pred_weight_table()` cũng được thay đổi để xử lý trường hợp khi mà nó không được báo hiệu.

`num_l1_weights` chỉ rõ số lượng của trọng số được báo hiệu cho các mục nhập trong danh mục hình ảnh tham chiếu 1 khi `pps_weighted_bipred_flag` và `wp_info_in_ph_flag` đều bằng 1. Giá trị của `num_l1_weights` sẽ trong phạm vi bao gồm 0 đến $\text{Min}(15, \text{num_ref_entries}[1][\text{RplsIdx}[1]])$.

Biến `NumWeightsL1` được suy ra như sau:

```
if( !pps_weighted_bipred_flag)
    NumWeightsL1 = 0
hoặc nếu (wp_info_in_ph_flag    &&    rpl_info_in_ph_flag    &&
(num_ref_entries[ 0 ][ RplsIdx[ 0 ] ] ==0 || num_ref_entries[ 1 ][ RplsIdx[ 1 ] ]
>=0))
    NumWeightsL1 = 0
hoặc if( wp_info_in_ph_flag )
    NumWeightsL1 = num_l1_weights
hoặc
    NumWeightsL1 = NumRefIdxActive[ 1 ]
```

Trong ngữ nghĩa của phần tử cú pháp trong `pred_weight_table()`, cách khác để suy ra giá trị của `num_l1_weights` khi nó không có mặt được minh họa dưới đây:

`num_l1_weights` chỉ rõ số lượng của trọng số được báo hiệu cho các mục nhập trong danh mục hình ảnh tham chiếu 1 khi `pps_weighted_bipred_flag` và `wp_info_in_ph_flag` đều bằng 1. Giá trị của `num_l1_weights` sẽ trong phạm vi bao gồm 0 đến $\text{Min}(15, \text{num_ref_entries}[1][\text{RplsIdx}[1]])$. Khi không có mặt, giá trị của `num_l1_weights` được suy ra bằng 0.

Biến `NumWeightsL1` được suy ra như sau:

```
if( !pps_weighted_bipred_flag)
    NumWeightsL1 = 0
hoặc if( wp_info_in_ph_flag )
    NumWeightsL1 = num_l1_weights
```

hoặc

```
NumWeightsL1 = NumRefIdxActive[ 1 ]
```

Trong ngữ nghĩa của phần tử cú pháp trong `pred_weight_table()`, cách khác nữa để suy ra giá trị của `num_l1_weights` khi nó không có mặt được minh họa dưới đây:

```
if( !pps_weighted_bipred_flag || ( wp_info_in_ph_flag && num_ref_entries
[ 1 ][ RplIdx[ 1 ] ] == 0 ) )
```

```
    NumWeightsL1 = 0
```

hoặc `if( wp_info_in_ph_flag )`

```
    NumWeightsL1 = num_l1_weights
```

hoặc

```
NumWeightsL1 = NumRefIdxActive[ 1 ]
```

Về mặt khái niệm, đề xuất thêm điều kiện báo hiệu để kiểm tra xem liệu hình ảnh hiện thời có hình ảnh tham chiếu từ cả các danh mục hình ảnh tham chiếu `list0` và `list1` cho bất kỳ phần tử cú pháp nào mà chỉ có khả năng áp dụng trong slice B hay không để tránh báo hiệu các bit dư thừa. Điều kiện kiểm tra không bị giới hạn ở các phương pháp nêu trên để kiểm tra kích thước của cả danh mục hình ảnh tham chiếu (ví dụ, các danh mục hình ảnh tham chiếu `list0/list1`) và điều kiện kiểm tra có thể là phương pháp bất kỳ khác để chỉ ra liệu hình ảnh hiện thời có hình ảnh tham chiếu từ cả danh mục hình ảnh tham chiếu `list0` và `list1` hay không. Ví dụ, cờ có thể được báo hiệu để chỉ ra liệu hình ảnh hiện thời có cả hai hình ảnh tham chiếu `list0` và `list1` hay không.

Khi phần tử cú pháp không được báo hiệu và thông tin danh mục hình ảnh tham chiếu được báo hiệu trong tiêu đề hình ảnh (picture header: PH), các giá trị của phần tử cú pháp được suy ra sử dụng thông tin liệu hình ảnh hiện thời có cả hai hình ảnh tham chiếu `list0` và `list1` hay nó chỉ có hình ảnh tham chiếu `list0` hoặc `list1`. Trong một ví dụ, khi `ph_collocated_from_l0_flag` không được báo hiệu, giá trị của nó được suy ra chỉ là hình ảnh tham chiếu mà hình ảnh hiện thời có. Trong ví dụ khác, khi `sps_bdo_f_enabled_flag` bằng 1 và `sps_bdo_f_pic_present_flag` bằng 1 nhưng `ph_disable_bdo_f_flag` không được báo hiệu, nó ngụ ý rằng hoặc `num_ref_entries[ 0 ][ RplIdx[ 0 ] ]` bằng 0 hoặc `num_ref_entries[ 1 ][ RplIdx[ 1 ] ]` bằng 0 theo điều kiện báo hiệu được đề xuất trên `ph_disable_bdo_f_flag`. Do đó, dưới điều kiện này, `ph_disable_bdo_f_flag` không được báo hiệu và được suy ra là 1. Trong

VVC hiện thời, không chỉ độ phân giải của hình ảnh được sắp xếp có ảnh hưởng đến việc kích hoạt TMVP mà cả độ lệch được áp dụng cho kích thước hình ảnh để chọn thang chia tính toán tỷ lệ có thể ảnh hưởng đến việc kích hoạt TMVP. Tuy nhiên, trong VVC hiện thời, độ lệch không được xem xét trong sự phù hợp luồng bit của `ph_temporal_mvp_enabled_flag`. Trong phương án thứ hai, đề xuất bổ sung rằng buộc phù hợp luồng bit vào VVC hiện thời, yêu cầu rằng giá trị của `ph_temporal_mvp_enabled_flag` sẽ phụ thuộc vào độ lệch mà được áp dụng cho kích thước hình ảnh để chọn thang chia tính toán tỷ lệ, như được minh họa dưới đây.

Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian và cùng một độ lệch mà được áp dụng cho kích thước hình ảnh để chọn thang chia tính toán tỷ lệ dưới dạng hình ảnh hiện thời, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

Câu trên cũng có thể được viết theo cách khác như sau:

khi không có hình ảnh tham chiếu trong DPB có giá trị biến được kết hợp `RprConstraintsActive[i][j]` bằng 0, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

Trong VVC hiện thời, có yêu cầu về sự phù hợp luồng bit mà hình ảnh được đề cập bởi `slice_collocated_ref_idx` sẽ giống với tất cả các slice của hình ảnh được lập mã. Tuy nhiên, khi hình ảnh được lập mã có nhiều slice và không có hình ảnh tham chiếu chung tồn tại trong số tất cả các slice, sự phù hợp luồng bit này không có cơ hội được đáp ứng. Trong phương án thứ ba của sáng chế, yêu cầu của sự phù hợp luồng bit trên `ph_temporal_mvp_enabled_flag` được biến đổi để xem xét liệu có hình ảnh tham chiếu chung tồn tại trong số tất cả các slice trong hình ảnh hiện thời hay không. Căn cứ vào phương án này, vài sửa đổi điển hình cho mô tả đặc điểm kỹ thuật VVC được minh họa dưới đây.

`ph_temporal_mvp_enabled_flag` chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên cho các slice kết hợp với PH. Nếu `ph_temporal_mvp_enabled_flag` bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác (`ph_temporal_mvp_enabled_flag` bằng 1), biến độc lập dự đoán của vector chuyển động

theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của `ph_temporal_mvp_enabled_flag` được suy ra là bằng 0. Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0. Khi không có hình ảnh tham chiếu chung tồn tại trong tất cả các slice liên kết hợp với PH, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

`ph_temporal_mvp_enabled_flag` chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên cho các slice kết hợp với PH hay không. Nếu `ph_temporal_mvp_enabled_flag` bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác (`ph_temporal_mvp_enabled_flag` bằng 1), biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của `ph_temporal_mvp_enabled_flag` được suy ra là bằng 0. Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0. Khi không có hình ảnh tham chiếu chung tồn tại trong tất cả các slice không nội kết hợp với PH, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

`ph_temporal_mvp_enabled_flag` chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên cho các slice kết hợp với PH. Nếu `ph_temporal_mvp_enabled_flag` bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác (`ph_temporal_mvp_enabled_flag` bằng 1), biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của `ph_temporal_mvp_enabled_flag` được suy ra là bằng 0. Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0. Khi không có hình ảnh tham chiếu chung tồn tại trong tất cả các slice không nội (non-intra) kết hợp với PH, giá trị của `ph_temporal_mvp_enabled_flag` sẽ bằng 0.

Trong một ví dụ, sự phù hợp luồng bit trên `slice_collocated_ref_idx` được đơn giản hóa như sau:

Là một yêu cầu của sự phù hợp luồng bit rằng các giá trị của `pic_width_in_luma_samples` và `pic_height_in_luma_samples` của hình ảnh tham chiếu được đề cập tương ứng bởi `slice_collocated_ref_idx` sẽ bằng các giá trị của `pic_width_in_luma_samples` và `pic_height_in_luma_samples`, của hình ảnh hiện thời, và `RprConstraintsActive[ slice_collocated_from_l0_flag ? 0 : 1 ][ slice_collocated_ref_idx ]` sẽ bằng 0.

Phương pháp trên có thể được thực hiện sử dụng thiết bị bao gồm một hoặc nhiều hệ mạch, mà bao gồm mạch tích hợp ứng dụng chuyên dụng (application specific integrated circuit: ASIC), bộ xử lý tín hiệu kỹ thuật số (DSP), thiết bị xử lý tín hiệu kỹ thuật số (digital signal processing device: DSPD), thiết bị logic lập trình được (programmable logic devices: PLD), mảng cổng lập trình được (field programmable gate array: FPGA), bộ điều khiển, bộ vi điều khiển, bộ vi xử lý, hoặc thành phần điện tử khác. Thiết bị này có thể sử dụng hệ mạch kết hợp với phần cứng hoặc phần mềm để thực hiện phương pháp mô tả ở trên. Mỗi môđun, môđun con, bộ phận hoặc bộ phận phụ được bộc lộ ở trên có thể được thực hiện ít nhất một phần sử dụng một hoặc nhiều hệ mạch.

Trong mô tả đặc điểm kỹ thuật VVC hiện tại, có hai sự phù hợp luồng bit về giới hạn độ phân giải của hình ảnh được sắp xếp. Một được đặt trên `PH_temporal_mvp_enabled_flag` và sự phù hợp còn lại được đặt trên sự phù hợp luồng bit ở `slice_collocated_ref_idx`. Tuy nhiên, việc tuân thủ hai luồng bit trong tham số kỹ thuật VVC về mặt chức năng là dư thừa vì cả hai đều cấm sử dụng hình ảnh được sắp xếp có độ phân giải khác nhau và/hoặc độ lệch chia độ khác nhau với hình ảnh hiện thời. Để giảm bớt gánh nặng của bộ mã hóa để kiểm tra sự phù hợp luồng bit, trong phương án thứ năm, đề xuất áp dụng chỉ sự phù hợp luồng bit `slice_collocated_ref_idx`. Ví dụ về mô tả đặc điểm kỹ thuật VVC đã sửa đổi được minh họa dưới đây. Phần thay đổi đã được đánh dấu.

ph_temporal_mvp_enabled_flag chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên cho các slice kết hợp với PH. Nếu `ph_temporal_mvp_enabled_flag` bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác

(**ph_temporal_mvp_enabled_flag** bằng 1), biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của **ph_temporal_mvp_enabled_flag** được suy ra là bằng 0. Cái còn lại được đặt trên sự phù hợp luồng bit trên **slice_collocated_ref_idx** như được minh họa dưới đây:

slice_collocated_ref_idx chỉ rõ chỉ số tham chiếu của hình ảnh được sắp xếp được sử dụng cho dự đoán vector chuyển động theo thời gian.

...

Là một yêu cầu của sự phù hợp luồng bit rằng các giá trị của **pic_width_in_luma_samples** và **pic_height_in_luma_samples** của hình ảnh tham chiếu được đề cập bởi **slice_collocated_ref_idx** sẽ bằng các giá trị của **pic_width_in_luma_samples** và **pic_height_in_luma_samples** tương ứng của hình ảnh hiện thời, và **RprConstraintsActive[slice_collocated_from_l0_flag ? 0 : 1][slice_collocated_ref_idx]** sẽ bằng 0.

Trong một ví dụ, đề xuất áp dụng chỉ sự phù hợp luồng bit **ph_temporal_mvp_enabled_flag**. Ví dụ về mô tả VVC sửa đổi được minh họa dưới đây.

ph_temporal_mvp_enabled_flag chỉ rõ liệu biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng cho dự đoán liên ảnh đối với các slice kết hợp với PH. Nếu **ph_temporal_mvp_enabled_flag** bằng 0, phần tử cú pháp của các slice kết hợp với PH sẽ bị ràng buộc sao cho không có biến độc lập dự đoán của vector chuyển động theo thời gian được sử dụng trong giải mã của các slice. Trường hợp khác (**ph_temporal_mvp_enabled_flag** bằng 1), biến độc lập dự đoán của vector chuyển động theo thời gian có thể được sử dụng trong giải mã của các slice kết hợp với PH. Khi không có mặt, giá trị của **ph_temporal_mvp_enabled_flag** được suy ra là bằng 0. Khi không có hình ảnh tham chiếu trong DPB có cùng một độ phân giải không gian dưới dạng hình ảnh hiện thời, giá trị của **ph_temporal_mvp_enabled_flag** sẽ bằng 0.

Cái còn lại được đặt trên sự phù hợp luồng bit trên slice_collocated_ref_idx như được minh họa dưới đây:

slice_collocated_ref_idx chỉ rõ chỉ số tham chiếu của hình ảnh được sắp xếp được sử dụng cho dự đoán vectơ chuyển động theo thời gian.

FIG. 5 là sơ đồ khối minh họa thiết bị lập mã video điển hình phù hợp với một số phương án thực hiện của sáng chế. Thiết bị này 500 có thể là thiết bị đầu cuối, chẳng hạn như điện thoại di động, máy tính bảng, thiết bị đầu cuối phát sóng kỹ thuật số, thiết bị máy tính bảng hoặc trợ lý kỹ thuật số cá nhân.

Như được thể hiện trong FIG. 5, thiết bị này 500 có thể bao gồm một hoặc nhiều các thành phần sau: bộ xử lý 502, bộ nhớ 504, bộ cấp điện 506, bộ phận đa phương tiện 508, bộ phận âm thanh 510, giao diện đầu vào/đầu ra (input/output: I/O) 512, bộ phận cảm ứng 514, và bộ phận truyền thông 516.

Bộ xử lý 502 thường điều khiển hoạt động tổng thể của thiết bị này 500, như các thao tác liên quan đến hiển thị, gọi điện thoại, truyền thông dữ liệu, vận hành camera và ghi. Bộ xử lý 502 có thể bao gồm một hoặc nhiều bộ xử lý 520 để chạy lệnh để hoàn thành tất cả hoặc một phần các bước của phương pháp trên. Ngoài ra, bộ xử lý 502 có thể bao gồm một hoặc nhiều mô đun để thuận tiện cho tương tác giữa bộ xử lý 502 và các thành phần khác. Ví dụ, bộ xử lý 502 có thể bao gồm mô đun đa phương tiện để thuận tiện cho tương tác giữa bộ phận đa phương tiện 508 và bộ xử lý 502.

Bộ nhớ 504 được cấu hình để lưu các loại dữ liệu khác nhau để hỗ trợ các thao tác của thiết bị này 500. Ví dụ về dữ liệu này bao gồm các lệnh, dữ liệu tiếp xúc, dữ liệu danh bạ, tin nhắn, hình ảnh, video, và v.v. cho bất kỳ ứng dụng hoặc phương pháp mà vận hành trên thiết bị này 500. Bộ nhớ 504 có thể được thực thi bằng loại bất kỳ trong số các thiết bị lưu trữ khả biến hoặc không khả biến hoặc kết hợp của chúng, và bộ nhớ 504 có thể là bộ nhớ truy cập ngẫu nhiên tĩnh (SRAM), bộ nhớ chỉ đọc lập trình có thể xóa bằng điện (EEPROM), bộ nhớ chỉ đọc lập trình có thể xóa (EPROM), Bộ nhớ chỉ đọc có thể lập trình (PROM), bộ nhớ chỉ đọc (ROM), bộ nhớ từ tính, bộ nhớ flash, đĩa từ tính hoặc đĩa nén.

Bộ cấp điện 506 cấp điện cho các thành phần khác nhau của thiết bị này 500. Bộ cấp điện 506 có thể bao gồm hệ thống quản lý cấp nguồn, một hoặc nhiều bộ phận cấp

điện và các thành phần khác kết hợp với việc sản xuất, quản lý và phân phối điện cho thiết bị này 500.

Bộ phận đa phương tiện 508 bao gồm màn hình cung cấp giao diện đầu ra giữa thiết bị này 500 và người dùng. Trong vài ví dụ, màn hình có thể bao gồm màn hình tinh thể lỏng (Liquid Crystal Display: LCD) và bảng cảm ứng chạm (Touch Panel: TP). Nếu màn hình bao gồm bảng cảm ứng chạm, màn hình này có thể được thực hiện dưới dạng màn hình chạm nhận tín hiệu đầu vào từ người dùng. Bảng cảm ứng chạm có thể bao gồm một hoặc nhiều cảm biến chạm để cảm ứng một cú chạm, trượt và một cử chỉ trên bảng điều khiển cảm ứng. Cảm biến chạm có thể không chỉ cảm nhận ranh giới của một hành động chạm hoặc trượt, mà còn phát hiện thời lượng và áp lực liên quan đến hoạt động chạm hoặc trượt. Trong vài ví dụ, bộ phận đa phương tiện 508 có thể bao gồm camera trước và/hoặc camera sau. Khi thiết bị này 500 ở chế độ vận hành, như chế độ chụp hoặc chế độ video, camera trước và/hoặc camera sau có thể nhận dữ liệu đa phương tiện bên ngoài.

Bộ phận âm thanh 510 được cấu hình để xuất ra và/hoặc nhập vào tín hiệu âm thanh. Ví dụ, bộ phận âm thanh 510 bao gồm micrô (MIC). Khi thiết bị này 500 ở chế độ vận hành, như chế độ cuộc gọi, chế độ ghi và chế độ nhận dạng giọng nói, micrô được cấu hình để nhận tín hiệu âm thanh bên ngoài. Tín hiệu âm thanh đã nhận có thể còn được lưu trong bộ nhớ 504 hoặc được gửi thông qua bộ phận truyền thông 516. Trong vài ví dụ, bộ phận âm thanh 510 còn bao gồm loa để phát ra tín hiệu âm thanh.

Giao diện I/O 512 cung cấp giao diện giữa bộ xử lý 502 và môđun giao diện ngoại vi. Môđun giao diện ngoại vi trên có thể là bàn phím, con lăn nhấp chuột, một nút hoặc những thứ tương tự. Những nút này có thể bao gồm nhưng không giới hạn ở, nút trang chủ, nút âm lượng, nút khởi động và nút khóa.

Bộ phận cảm ứng 514 bao gồm một hoặc nhiều cảm biến để cung cấp đánh giá trạng thái ở các khía cạnh khác nhau cho thiết bị này 500. Ví dụ, bộ phận cảm ứng 514 có thể phát hiện ra trạng thái tắt/mở của thiết bị này 500 và vị trí tương đối của các thành phần. Ví dụ, các thành phần là màn hình và bàn phím của thiết bị này 500. Bộ phận cảm ứng 514 có thể cũng phát hiện ra sự thay đổi vị trí của thiết bị này 500 hoặc thành phần của thiết bị này 500, có hoặc không có danh bạ người dùng trên thiết bị này 500, định hướng hoặc tăng/giảm tốc độ thiết bị này 500, và sự thay đổi nhiệt độ của thiết bị này

500. Bộ phận cảm ứng 514 có thể bao gồm một cảm biến độ gần được định cấu hình để phát hiện sự hiện diện của một đối tượng gần đó mà không cần bất kỳ chạm vật lý nào. Bộ phận cảm ứng 514 có thể còn bao gồm cảm biến quang, như CMOS hoặc cảm biến hình ảnh CCD được sử dụng trong ứng dụng hình ảnh. Trong vài ví dụ, bộ phận cảm ứng 514 có thể bao gồm cảm biến gia tốc, cảm biến con quay hồi chuyển, cảm biến từ tính, cảm biến áp suất hoặc cảm biến nhiệt độ.

Bộ phận truyền thông 516 được cấu hình để thuận tiện cho việc truyền thông có dây hoặc không dây giữa thiết bị này 500 và các thiết bị khác. Thiết bị này 500 có thể truy cập mạng không dây dựa trên tiêu chuẩn giao tiếp, chẳng hạn như WiFi, 4G hoặc sự kết hợp của chúng. Trong một ví dụ, bộ phận truyền thông 516 nhận tín hiệu phát sóng hoặc phát thông tin liên quan từ hệ thống quản lý chương trình phát sóng bên ngoài qua kênh phát rộng. Trong một ví dụ, bộ phận truyền thông 516 có thể bao gồm thêm môđun giao tiếp trường gần (near field communication: NFC) để thúc đẩy giao tiếp trong phạm vi ngắn. Ví dụ: môđun NFC có thể được triển khai dựa trên công nghệ nhận dạng tần số vô tuyến (radio frequency identification: RFID), công nghệ kết hợp dữ liệu hồng ngoại (infrared data association: IrDA), công nghệ băng tần siêu rộng (ultra-wide band: UWB), công nghệ bluetooth (BT) và công nghệ khác.

Trong một ví dụ, thiết bị này 500 có thể được chạy bằng một hoặc nhiều mạch tích hợp ứng dụng chuyên dụng (Application Specific Integrated Circuits: ASIC), bộ xử lý tín hiệu kỹ thuật số (DSP), thiết bị xử lý tín hiệu kỹ thuật số (Digital Signal Processing Device: DSPD), thiết bị logic có thể lập trình được (Programmable Logic Device: PLD), mảng cổng lập trình trường (Field Programmable Gate Array: FPGA), bộ điều khiển, bộ vi điều khiển, bộ vi xử lý hoặc bộ phận điện tử khác để thực hiện phương pháp trên.

Phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời có thể là, ví dụ, ổ đĩa cứng (HDD), ổ cứng thể rắn (SSD), bộ nhớ Flash, ổ lai hoặc ổ lai thể rắn (SSHD), bộ nhớ chỉ đọc (ROM), bộ nhớ chỉ đọc đĩa nhỏ gọn (CD-ROM), băng từ, đĩa mềm và v.v..

FIG. 6 là lưu đồ minh họa quy trình diễn hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

Trong bước 602, bộ xử lý 520 xác định liệu cờ vô hiệu hóa có mặt trong PH kết hợp với hình ảnh.

Trong vài ví dụ, cờ vô hiệu hóa chỉ rõ liệu công cụ lập mã được vô hiệu hóa trong một hoặc nhiều lát kết hợp với PH hay không.

Trong bước 604, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH.

Trong vài ví dụ, bộ xử lý 520 vô hiệu hóa công cụ lập mã trong giải mã một hoặc nhiều lát đáp lại việc xác định rằng giá trị của cờ vô hiệu hóa bằng 1 và kích hoạt công cụ lập mã trong giải mã một hoặc nhiều lát đáp lại việc xác định rằng giá trị của cờ vô hiệu hóa bằng 0.

Trong vài ví dụ, công cụ lập mã bao gồm ít nhất một trong số phép dự đoán kép liên ảnh dựa vào DMVR và dự đoán kép liên ảnh dựa vào BDOF.

Trong vài ví dụ, bộ xử lý 520 bỏ qua phân tích cú pháp cờ vô hiệu hóa đáp lại việc xác định rằng một hoặc nhiều danh mục hình ảnh tham chiếu chỉ ra rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều.

Trong vài ví dụ, cờ vô hiệu hóa chỉ rõ liệu DMVR dựa trên dự đoán kép liên ảnh được vô hiệu hóa trong một hoặc nhiều lát kết hợp với PH, bộ xử lý 520 vô hiệu hóa công cụ lập mã trong giải mã một hoặc nhiều lát bằng cách vô hiệu hóa DMVR dựa trên dự đoán kép liên ảnh trong giải mã một hoặc nhiều lát, và bộ xử lý 520 kích hoạt công cụ lập mã trong giải mã một hoặc nhiều lát bằng cách kích hoạt DMVR dựa trên dự đoán kép liên ảnh trong giải mã một hoặc nhiều lát.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 0 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1 và cờ kích hoạt thứ hai trong SPS bằng 0.

Trong vài ví dụ, cờ kích hoạt thứ nhất trong SPS chỉ rõ liệu DMVR dựa trên dự đoán kép liên ảnh được kích hoạt, cờ kích hoạt thứ nhất bằng 1 chỉ rõ rằng DMVR dựa trên dự đoán kép liên ảnh được kích hoạt, và cờ kích hoạt thứ nhất bằng 0 chỉ rõ rằng DMVR dựa trên dự đoán kép liên ảnh được vô hiệu hóa.

Trong vài ví dụ, cờ kích hoạt thứ hai trong SPS chỉ rõ liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS, cờ kích hoạt thứ hai bằng 0 chỉ rõ rằng cờ vô hiệu hóa không có mặt trong PH để chỉ SPS, và cờ kích hoạt thứ hai bằng 1 chỉ rõ rằng cờ vô hiệu hóa có mặt trong PH để chỉ SPS.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1 và cờ kích hoạt thứ hai trong SPS bằng 1 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0 và cờ kích hoạt thứ hai trong SPS bằng 0 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1, cờ kích hoạt thứ hai trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0. Một hoặc nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phép phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1.

Trong vài ví dụ, bộ xử lý 520 xác định giá trị của cờ kích hoạt thứ nhất trong SPS là W, và cờ kích hoạt thứ nhất trong SPS chỉ rõ liệu DMVR dựa trên dự đoán kép liên ảnh được kích hoạt, cờ kích hoạt thứ nhất bằng 1 chỉ rõ rằng DMVR dựa trên dự đoán kép liên ảnh được kích hoạt, và cờ kích hoạt thứ nhất bằng 0 chỉ rõ rằng DMVR dựa trên dự đoán kép liên ảnh được vô hiệu hóa. Bộ xử lý 520 suy ra giá trị của cờ vô hiệu

hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng $1-W$ đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 0. Cờ kích hoạt thứ hai trong SPS chỉ rõ liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS, cờ kích hoạt thứ hai bằng 0 chỉ rõ rằng cờ vô hiệu hóa không có mặt trong PH để chỉ SPS, và cờ kích hoạt thứ hai bằng 1 chỉ rõ rằng cờ vô hiệu hóa có mặt trong PH để chỉ SPS.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phép phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0 bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0. Một hoặc nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt trong SPS đáp lại việc xác định rằng cờ vô hiệu hóa được báo hiệu trong PH.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt trong SPS đáp lại việc xác định rằng cờ vô hiệu hóa được báo hiệu trong PH bằng cách: xác định giá trị của cờ kích hoạt thứ nhất trong SPS là W ; suy ra giá trị của cờ vô hiệu hóa bằng $1-W$ đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1 và một hoặc nhiều danh mục hình ảnh tham chiếu không được báo hiệu trong PH; suy ra giá trị của cờ vô hiệu hóa bằng $1-W$ đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu 1 lớn hơn 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt trong SPS đáp lại việc xác định rằng cờ vô hiệu hóa được báo hiệu trong PH bằng cách: suy ra giá trị của cờ vô hiệu hóa là giá trị của cờ vô hiệu hóa được báo hiệu rõ ràng trong PH đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1, cờ kích hoạt thứ hai trong SPS bằng 1, và một hoặc nhiều danh mục hình ảnh tham chiếu không được báo hiệu trong PH; suy ra giá trị của cờ vô hiệu hóa là giá trị của cờ vô hiệu hóa được báo hiệu rõ ràng trong PH đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1, cờ kích hoạt thứ hai trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu 1 lớn hơn 0.

Trong vài ví dụ, cờ vô hiệu hóa chỉ rõ liệu dự đoán kép liên ảnh dựa trên BDOF được vô hiệu hóa trong một hoặc nhiều lát kết hợp với PH. Bộ xử lý 520 vô hiệu hóa công cụ lập mã trong giải mã một hoặc nhiều lát bằng cách vô hiệu hóa dự đoán kép liên ảnh dựa trên BDOF trong giải mã một hoặc nhiều lát và kích hoạt công cụ lập mã trong giải mã một hoặc nhiều lát bằng cách kích hoạt dự đoán kép liên ảnh dựa trên BDOF trong giải mã một hoặc nhiều lát.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 0 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 1 và cờ kích hoạt thứ tư trong SPS bằng 0. Cờ kích hoạt thứ ba trong SPS chỉ rõ liệu dự đoán kép liên ảnh dựa trên BDOF được kích hoạt, cờ kích hoạt thứ ba bằng 1 chỉ rõ rằng dự đoán kép liên ảnh dựa trên BDOF được kích hoạt, và cờ kích hoạt thứ ba bằng 0 chỉ rõ rằng dự đoán kép liên ảnh dựa trên BDOF được vô hiệu hóa. Cờ kích hoạt thứ tư trong SPS chỉ rõ liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS, cờ kích hoạt thứ tư bằng 0 chỉ rõ rằng cờ vô hiệu hóa không có mặt trong PH để chỉ SPS, và cờ kích hoạt thứ tư bằng 1 chỉ rõ rằng cờ vô hiệu hóa có mặt trong PH để chỉ SPS.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 1 và cờ

kích hoạt thứ tư trong SPS bằng 1 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng việc phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 0 và cờ kích hoạt thứ tư trong SPS bằng 0 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 1, cờ kích hoạt thứ tư trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0. Một hoặc nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 0 và suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS bằng 1.

Trong vài ví dụ, bộ xử lý 520 xác định giá trị của cờ kích hoạt thứ ba trong SPS là V và suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1-V đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS bằng 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ vô hiệu hóa được bỏ qua bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS khác 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS khác 0 bằng cách suy ra giá trị của cờ vô hiệu hóa bằng 1 đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0. Một hoặc nhiều

danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 xác định giá trị của cờ kích hoạt thứ ba trong SPS là V và suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt trong SPS đáp lại việc xác định rằng cờ vô hiệu hóa được báo hiệu trong PH bằng cách: suy ra giá trị của cờ vô hiệu hóa bằng 1-V đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS bằng 1 và một hoặc nhiều danh mục hình ảnh tham chiếu không được báo hiệu trong PH; và suy ra giá trị của cờ vô hiệu hóa bằng 1-V đáp lại việc xác định rằng cờ kích hoạt thứ tư trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu 1 lớn hơn 0.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ vô hiệu hóa là giá trị của cờ vô hiệu hóa được báo hiệu rõ ràng trong PH đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 1, cờ kích hoạt thứ tư trong SPS bằng 1, và một hoặc nhiều danh mục hình ảnh tham chiếu không được báo hiệu trong PH và suy ra giá trị của cờ vô hiệu hóa là giá trị của cờ vô hiệu hóa được báo hiệu rõ ràng trong PH đáp lại việc xác định rằng cờ kích hoạt thứ ba trong SPS bằng 1, cờ kích hoạt thứ tư trong SPS bằng 1, một hoặc nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu 1 lớn hơn 0.

FIG. 7 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

Trong 702, bộ xử lý 520 xác định liệu cờ có mặt trong PH kết hợp với hình ảnh.

Trong vài ví dụ, cờ chỉ rõ liệu hình ảnh được sử dụng cho TMVP được suy ra từ danh mục hình ảnh tham chiếu từ nhiều danh mục hình ảnh tham chiếu kết hợp với hình ảnh.

Trong bước 704, bộ xử lý 520 suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu đáp lại việc xác định rằng cờ không có mặt trong PH.

Trong vài ví dụ, bộ xử lý 520 bỏ qua phân tích cú pháp cờ đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu chỉ ra rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều.

Trong vài ví dụ, nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, cờ bằng 1 chỉ rõ rằng hình ảnh được sử dụng cho TMVP được suy ra từ danh mục hình ảnh tham chiếu thứ nhất, và cờ bằng 0 chỉ rõ rằng hình ảnh được sử dụng cho TMVP được suy ra từ danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu đáp lại việc xác định rằng phân tích cú pháp của cờ được bỏ qua bằng cách suy ra giá trị của cờ bằng 1 đáp lại việc xác định rằng số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ nhất lớn hơn 1 và suy ra giá trị của cờ bằng 0 đáp lại việc xác định rằng số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai lớn hơn 1.

FIG. 8 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

Trong bước 802, bộ xử lý 520 xác định liệu cờ có mặt trong PH kết hợp với hình ảnh.

Trong vài ví dụ, cờ chỉ rõ số lượng của trọng số được báo hiệu trong danh mục hình ảnh tham chiếu theo cờ WP thứ nhất trong PPS của hình ảnh và cờ WP thứ hai trong PH của hình ảnh và cờ trong cú pháp WP kết hợp với hình ảnh.

Trong bước 804, bộ xử lý 520 suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu từ nhiều danh mục hình ảnh tham chiếu kết hợp với hình ảnh đáp lại việc xác định rằng cờ không có mặt trong PH.

Trong vài ví dụ, bộ xử lý 520 bỏ qua phân tích cú pháp cờ đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu chỉ ra rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều.

Trong vài ví dụ, nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai, cờ chỉ rõ số lượng của

trọng số được báo hiệu trong danh mục hình ảnh tham chiếu thứ hai đáp lại việc xác định rằng cờ WP thứ nhất trong PPS bằng 1 và cờ WP thứ hai trong PH bằng 1.

Trong vài ví dụ, bộ xử lý 520 suy ra giá trị của cờ theo số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu kết hợp với hình ảnh đáp lại việc xác định rằng phân tích cú pháp của cờ được bỏ qua bằng cách suy ra giá trị của cờ bằng 0 đáp lại việc xác định rằng cờ WP thứ nhất trong PPS bằng 0, cờ WP thứ hai trong PH bằng 1, và số lượng của hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0, suy ra giá trị của cờ là giá trị của cờ được báo hiệu rõ ràng trong PH đáp lại việc xác định rằng cờ WP thứ nhất trong PPS khác 0 và cờ WP thứ hai trong PH bằng 1, và suy ra giá trị của cờ là giá trị của NumRefIdxActive[1] đáp lại việc xác định rằng cờ WP thứ nhất trong PPS khác 0 và cờ WP thứ hai trong PH khác 1.

Trong vài ví dụ, giá trị của NumRefIdxActive[i] – 1 chỉ rõ chỉ số tham chiếu tối đa cho danh mục hình ảnh tham chiếu i, trong đó i bằng 0 hoặc 1.

Ví dụ, giá trị của NumRefIdxActive[1] – 1 chỉ rõ chỉ số tham chiếu tối đa cho danh mục hình ảnh tham chiếu 1, đó là danh mục hình ảnh tham chiếu thứ hai.

Trong vài ví dụ, bộ xử lý 520 xác định giá trị của cờ đáp lại việc xác định rằng cờ có mặt trong PH bằng cách xác định giá trị của cờ là 0 đáp lại việc xác định cờ WP thứ nhất trong PPS bằng 0, xác định giá trị của cờ là giá trị của cờ được báo hiệu rõ ràng trong PH đáp lại việc xác định cờ WP thứ nhất trong PPS khác 0 và cờ WP thứ hai trong PH bằng 1, và xác định giá trị của cờ là giá trị của NumRefIdxActive[1] đáp lại việc xác định rằng cờ WP thứ nhất trong PPS khác 0 và cờ WP thứ hai trong PH khác 1.

FIG. 9 là lưu đồ minh họa quy trình điển hình của lập mã video phù hợp với một số phương án thực hiện của sáng chế.

Trong 902, bộ xử lý 520 sử dụng cờ được kích hoạt để chỉ rõ liệu một hoặc nhiều biến độc lập dự đoán của vectơ chuyển động theo thời gian được sử dụng cho dự đoán liên cho một hoặc nhiều lát kết hợp với PH của hình ảnh.

Trong bước 904, bộ xử lý 520 ràng buộc giá trị của cờ được kích hoạt theo nhiều độ lệch được áp dụng cho kích thước của hình ảnh để chọn thang chia tính toán tỷ lệ.

Trong vài ví dụ, bộ xử lý thiết lập cờ được kích hoạt bằng 0 đáp lại việc xác định rằng không có hình ảnh tham chiếu chung trong một hoặc nhiều lát liên. Một hoặc nhiều lát bao gồm một hoặc nhiều lát liên kết hợp với PH.

Trong vài ví dụ, bộ xử lý thiết lập cờ được kích hoạt thành 0 đáp lại việc xác định rằng không có hình ảnh tham chiếu chung trong một hoặc nhiều lát không nội (non-intra).

Trong vài ví dụ, đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý 520; và bộ nhớ 504 được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý; trong đó bộ xử lý, nhờ chạy lệnh, được cấu hình để thực hiện phương pháp như được minh họa trong FIG. 6.

Trong vài ví dụ, đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý 520; và bộ nhớ 504 được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý; trong đó bộ xử lý, nhờ chạy lệnh, được cấu hình để thực hiện phương pháp như được minh họa trong FIG. 7.

Trong vài ví dụ, đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý 520; và bộ nhớ 504 được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý; trong đó bộ xử lý, nhờ chạy lệnh, được cấu hình để thực hiện phương pháp như được minh họa trong FIG. 8.

Trong vài ví dụ, đề xuất thiết bị lập mã video. Thiết bị này bao gồm một hoặc nhiều bộ xử lý 520; và bộ nhớ 504 được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý; trong đó bộ xử lý, nhờ chạy lệnh, được cấu hình để thực hiện phương pháp như được minh họa trong FIG. 9.

Trong vài ví dụ khác, đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời 504, có lệnh được lưu ở đó. Khi các lệnh được chạy bởi một hoặc nhiều bộ xử lý 520, các lệnh này làm cho bộ xử lý thực hiện phương pháp như được minh họa trong FIG. 6.

Theo vài ví dụ khác, đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời 504, có lệnh được lưu ở đó. Khi các lệnh được chạy bằng một hoặc nhiều bộ xử lý 520, các lệnh này làm cho bộ xử lý thực hiện phương pháp như được minh họa trong FIG. 7.

Trong vài ví dụ khác, đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời 504, có lệnh được lưu ở đó. Khi lệnh được chạy bằng một hoặc nhiều bộ xử lý 520, các lệnh làm cho bộ xử lý thực hiện phương pháp như được minh họa trong FIG. 8.

Trong vài ví dụ khác, đề xuất phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời 504, có lệnh được lưu ở đó. Khi lệnh được chạy bằng một hoặc nhiều bộ xử lý 520, lệnh này làm cho bộ xử lý thực hiện phương pháp như được minh họa trong FIG. 9.

Mô tả của sáng chế đã được thể hiện cho mục đích minh họa và không dự định là đầy đủ hoặc giới hạn đối với sáng chế. Nhiều sửa đổi, biến thể và triển khai thay thế sẽ rõ ràng đối với những người có hiểu biết trung bình trong lĩnh vực này trong các mô tả ở trên và các hình vẽ liên quan.

Các ví dụ đã được chọn và mô tả để giải thích các nguyên tắc sáng chế, và cho phép những người có hiểu biết trung bình trong lĩnh vực này hiểu được sáng chế với các cách triển khai khác nhau và để sử dụng tốt nhất các nguyên tắc cơ bản và các cách triển khai khác nhau với các sửa đổi khác nhau phù hợp với đặc điểm cụ thể sử dụng dự tính. Do đó, cần phải hiểu rằng phạm vi tiết lộ không bị giới hạn trong các ví dụ cụ thể về các triển khai được tiết lộ và các sửa đổi và triển khai khác được dự định đưa vào phạm vi của sáng chế.

Sáng chế yêu cầu hưởng quyền ưu tiên của đơn tạm thời Mỹ số 63/007,355, tiêu đề “Báo hiệu phần tử cú pháp trong lập mã video,” nộp ngày 08/04/2020, đơn tạm thời Mỹ số 63/009,404, tiêu đề “Báo hiệu phần tử cú pháp trong lập mã video,” nộp ngày 13/04/2020, đơn tạm thời Mỹ số 63/010/005, tiêu đề “Báo hiệu phần tử cú pháp trong lập mã video,” nộp ngày 14/04/2020, đơn tạm thời Mỹ số 60/010,619, tiêu đề “Báo hiệu phần tử cú pháp trong lập mã video,” nộp ngày 15/04/2020, và đơn tạm thời Mỹ số 63/015,663, tiêu đề “Báo hiệu phần tử cú pháp trong lập mã video,” nộp ngày 26/04/2020, đều được tham khảo kết hợp ở đây.

YÊU CẦU BẢO HỘ

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

xác định bằng bộ giải mã, liệu cờ vô hiệu hóa có mặt trong tiêu đề hình ảnh (picture header: PH) kết hợp với hình ảnh hay không, trong đó cờ vô hiệu hóa chỉ rõ liệu công cụ lập mã được vô hiệu hóa cho một hoặc nhiều lát (slice) kết hợp với PH hay không; và

đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra, bằng bộ giải mã, giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong tập hợp tham số chuỗi (sequence parameter set: SPS) của hình ảnh;

trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt trong SPS của hình ảnh bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1 và cờ kích hoạt thứ hai trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 0.

2. Phương pháp theo điểm 1, trong đó cờ kích hoạt thứ nhất trong SPS chỉ rõ rằng liệu công cụ lập mã có được kích hoạt hay không, cờ kích hoạt thứ nhất bằng 1 chỉ rõ rằng công cụ lập mã được kích hoạt, và cờ kích hoạt thứ nhất bằng 0 chỉ rõ rằng công cụ lập mã bị vô hiệu hóa, và trong đó cờ kích hoạt thứ hai trong SPS chỉ rõ rằng liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS hay không, cờ kích hoạt thứ hai bằng 1 chỉ rõ rằng cờ vô hiệu hóa được cho phép có mặt trong PH để chỉ SPS.

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

đáp lại việc xác định rằng giá trị của cờ vô hiệu hóa bằng 1, thì vô hiệu hóa bằng bộ giải mã, công cụ lập mã trong giải mã một hoặc nhiều lát; và

đáp lại việc xác định rằng giá trị của cờ vô hiệu hóa bằng 0, thì kích hoạt bằng bộ giải mã, công cụ lập mã trong giải mã một hoặc nhiều lát.

4. Phương pháp theo điểm 1, trong đó công cụ lập mã bao gồm ít nhất một trong số phép dự đoán kép liên ảnh dựa vào tính chỉnh vector chuyển động của bộ giải mã (decoder motion vector refinement: DMVR) và phép dự đoán kép liên ảnh dựa vào luồng quang hai chiều (bi-directional optical flow: BDOF).

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

đáp lại việc xác định từ một hoặc nhiều danh mục hình ảnh tham chiếu rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều, thì xác định, bởi bộ giải mã rằng cờ vô hiệu hóa không có mặt trong PH.

6. Phương pháp theo điểm 5, trong đó cờ vô hiệu hóa chỉ rõ liệu phép dự đoán kép liên ảnh dựa vào DMVR được vô hiệu hóa cho một hoặc nhiều lát kết hợp với PH hay không;

trong đó vô hiệu hóa công cụ lập mã trong giải mã một hoặc nhiều lát bao gồm vô hiệu hóa phép dự đoán kép liên ảnh dựa vào DMVR trong giải mã một hoặc nhiều lát; và

trong đó kích hoạt công cụ lập mã trong giải mã một hoặc nhiều lát bao gồm kích hoạt phép dự đoán kép liên ảnh dựa vào DMVR trong giải mã một hoặc nhiều lát.

7. Phương pháp theo điểm 6, trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1 và cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1; và

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

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

đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng các hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0, thì xác định rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều, trong đó nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai,

trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0 và cờ kích hoạt thứ hai trong SPS bằng 0, thì suy ra cờ vô hiệu hóa bằng 1; và

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1, và cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

9. Phương pháp theo điểm 7, trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1; và

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

10. Phương pháp theo điểm 6, còn bao gồm:

xác định giá trị của cờ kích hoạt thứ nhất trong SPS là W, trong đó cờ kích hoạt thứ nhất trong SPS chỉ rõ liệu dự đoán kép liên ảnh dựa vào DMVR được kích hoạt hay không, cờ kích hoạt thứ nhất bằng 1 chỉ rõ rằng dự đoán kép liên ảnh dựa vào DMVR được kích hoạt, và cờ kích hoạt thứ nhất bằng 0 chỉ rõ rằng dự đoán kép liên ảnh dựa vào DMVR được vô hiệu hóa; và

trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1-W,

trong đó cờ kích hoạt thứ hai trong SPS chỉ rõ liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS, cờ kích hoạt thứ hai bằng 0 chỉ rõ rằng cờ vô hiệu hóa không có mặt trong PH để chỉ SPS, và cờ kích hoạt thứ hai bằng 1 chỉ rõ rằng cờ vô hiệu hóa có khả năng có mặt trong PH để chỉ SPS.

11. Phương pháp theo điểm 10, trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

12. Phương pháp theo điểm 11, còn bao gồm

đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng các hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0, thì xác định rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều, trong đó nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai,

trong đó đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1 còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

13. Phương pháp theo điểm 5, trong đó cờ vô hiệu hóa chỉ rõ liệu phép dự đoán kép liên ảnh dựa trên BDOF được vô hiệu hóa cho một hoặc nhiều lát kết hợp với PH hay không;

trong đó vô hiệu hóa công cụ lập mã trong giải mã một hoặc nhiều lát bao gồm vô hiệu hóa dự đoán kép liên ảnh dựa trên BDOF trong giải mã một hoặc nhiều lát; và

trong đó kích hoạt công cụ lập mã trong giải mã một hoặc nhiều lát bao gồm kích hoạt dự đoán kép liên ảnh dựa trên BDOF trong giải mã một hoặc nhiều lát.

14. Phương pháp theo điểm 13, trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1 và cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1; và

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1; hoặc

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1, và đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

15. Phương pháp theo điểm 14, còn bao gồm

đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng các hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0, xác định rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều, trong đó nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai,

trong đó, đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 0 và cờ kích hoạt thứ hai trong SPS bằng 0, thì suy ra cờ vô hiệu hóa bằng 1; và

đáp lại việc xác định rằng cờ kích hoạt thứ nhất trong SPS bằng 1, và cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

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

xác định giá trị của cờ kích hoạt thứ nhất trong SPS là V, trong đó cờ kích hoạt thứ nhất trong SPS chỉ rõ liệu dự đoán kép liên ảnh dựa trên BDOF được kích hoạt hay không, cờ kích hoạt thứ nhất bằng 1 chỉ rõ rằng dự đoán kép liên ảnh dựa trên BDOF được kích hoạt, và cờ kích hoạt thứ nhất bằng 0 chỉ rõ rằng dự đoán kép liên ảnh dựa trên BDOF được vô hiệu hóa; và

trong đó suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1-V, trong đó cờ kích hoạt thứ hai trong SPS chỉ rõ liệu cờ vô hiệu hóa có mặt trong PH để chỉ SPS, cờ kích hoạt thứ hai bằng 0 chỉ rõ rằng cờ vô hiệu hóa không có mặt trong PH để chỉ SPS, và cờ kích hoạt thứ hai bằng 1 chỉ rõ rằng cờ vô hiệu hóa có mặt trong PH để chỉ SPS.

17. Phương pháp theo điểm 16, trong đó đáp lại việc xác định rằng cờ vô hiệu hóa không có mặt trong PH, thì suy ra giá trị của cờ vô hiệu hóa theo một hoặc nhiều cờ có thể kích hoạt được báo hiệu trong SPS của hình ảnh còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

18. Phương pháp theo điểm 17, còn bao gồm

đáp lại việc xác định rằng nhiều danh mục hình ảnh tham chiếu được báo hiệu trong PH, và số lượng hình ảnh tham chiếu trong danh mục hình ảnh tham chiếu thứ hai bằng 0, thì xác định rằng một hoặc nhiều lát kết hợp với hình ảnh không phải là dự đoán hai chiều, trong đó nhiều danh mục hình ảnh tham chiếu bao gồm danh mục hình ảnh tham chiếu thứ nhất và danh mục hình ảnh tham chiếu thứ hai,

trong đó đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS khác 0, thì suy ra giá trị của cờ vô hiệu hóa bằng 1 còn bao gồm:

đáp lại việc xác định rằng cờ kích hoạt thứ hai trong SPS bằng 1, thì suy ra giá trị của cờ vô hiệu hóa bằng 1.

19. Thiết bị lập mã video, bao gồm:

một hoặc nhiều bộ xử lý; và

bộ nhớ được cấu hình để lưu lệnh có thể được chạy bằng một hoặc nhiều bộ xử lý; trong đó một hoặc nhiều bộ xử lý, nhờ chạy lệnh, được cấu hình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 18.

20. Phương tiện lưu trữ có thể đọc được trên máy tính không tạm thời để lập mã video, lưu luồng bit để được giải mã bằng phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 18.

1 / 6

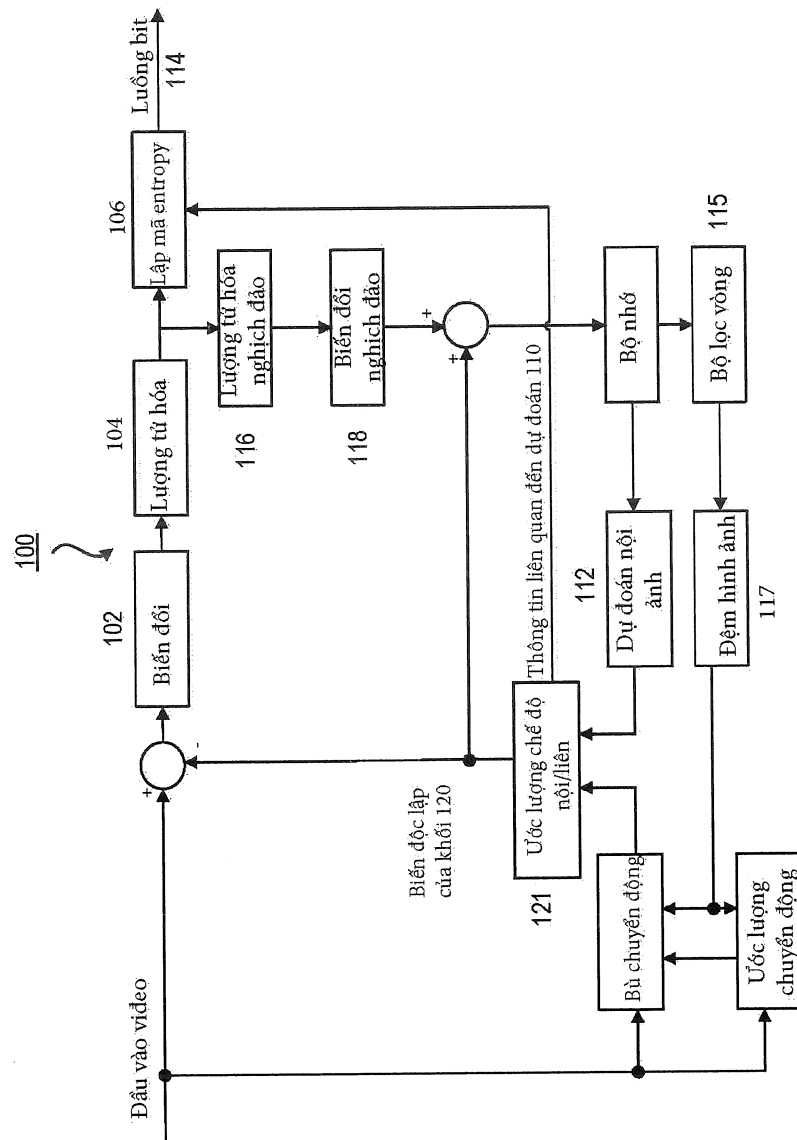
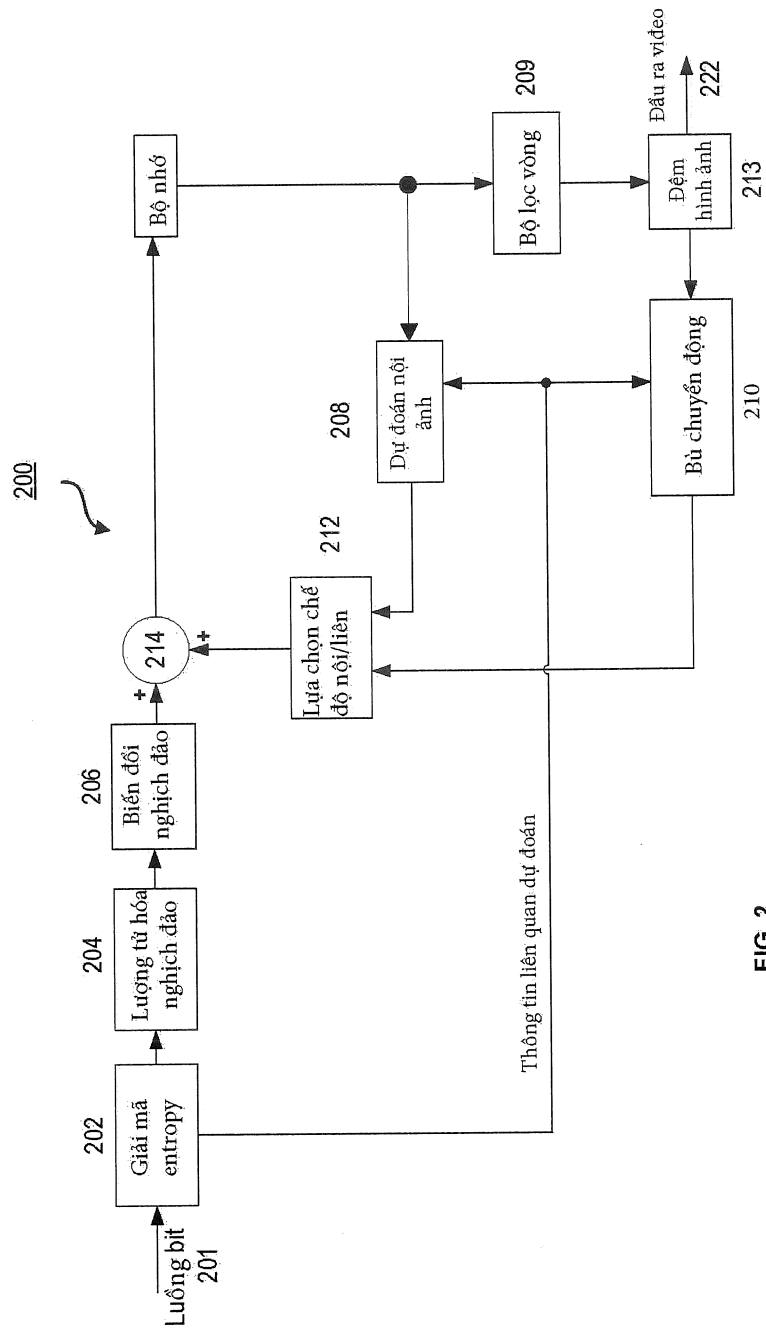


FIG. 1

2 / 6



3 / 6

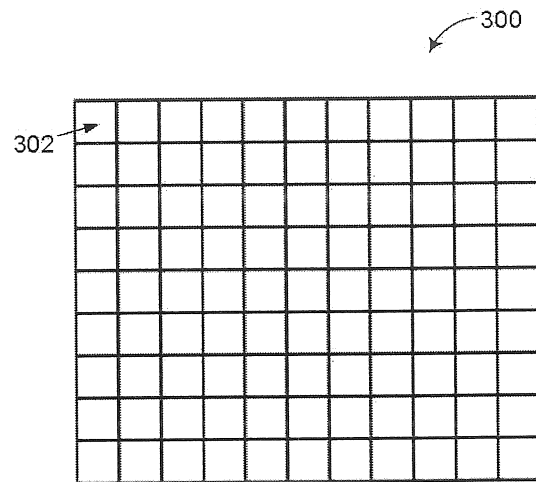
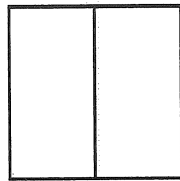


FIG. 3

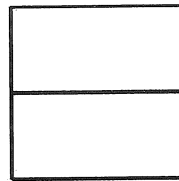
402



SPLIT_BT_VER

FIG. 4A

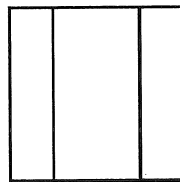
404



SPLIT_BT_HOR

FIG. 4B

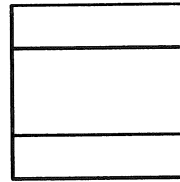
406



SPLIT_TT_VER

FIG. 4C

408



SPLIT_TT_HOR

FIG. 4D

4 / 6

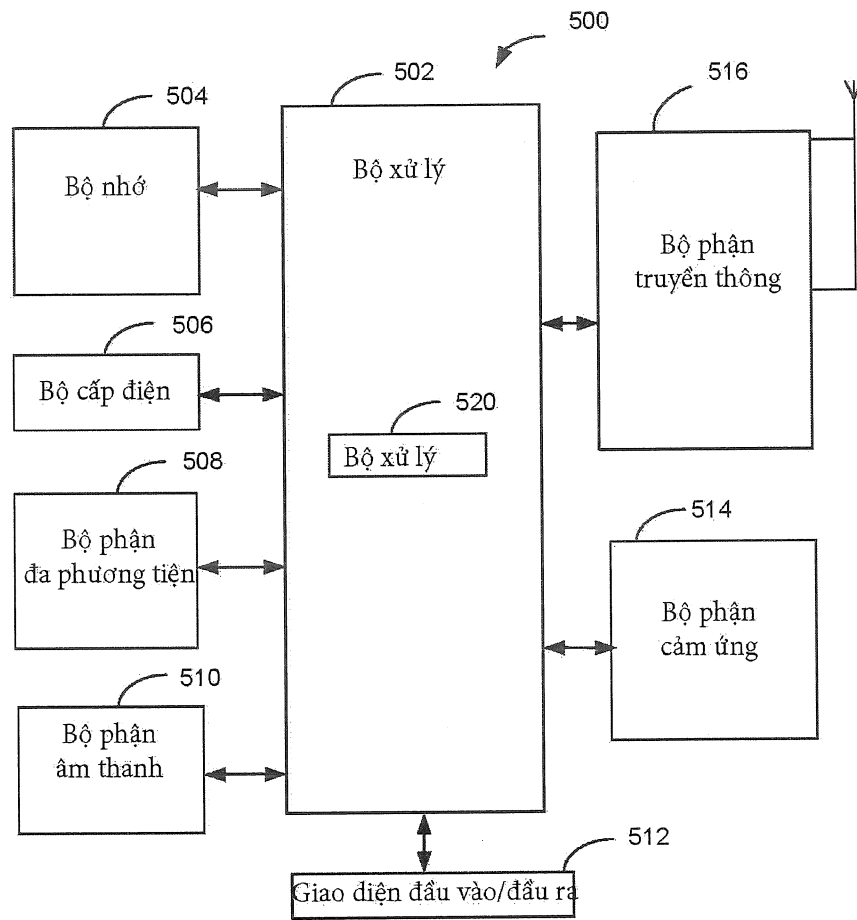


FIG. 5

5 / 6

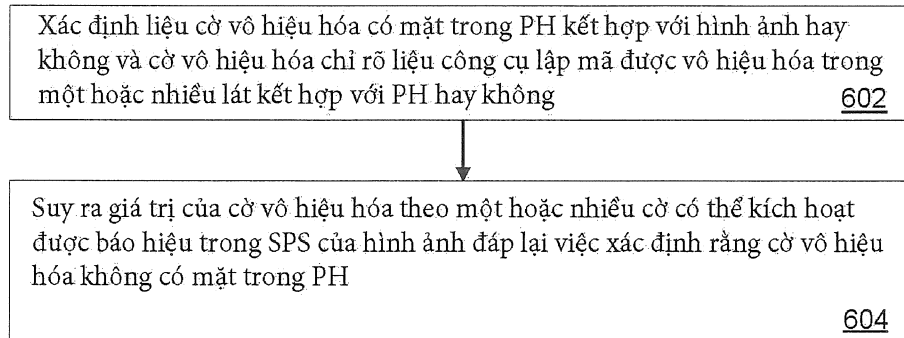


FIG. 6

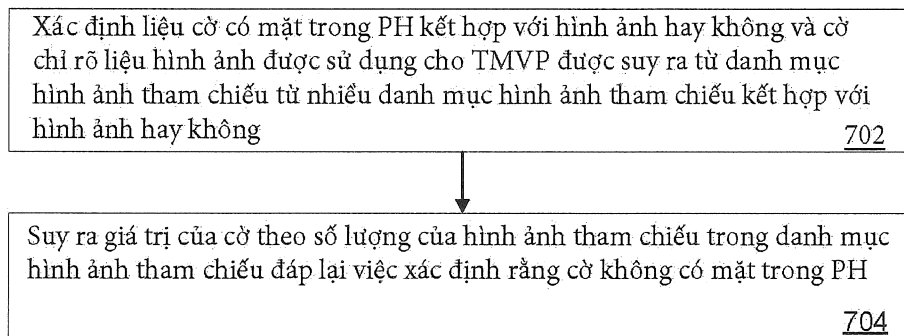


FIG. 7

6 / 6

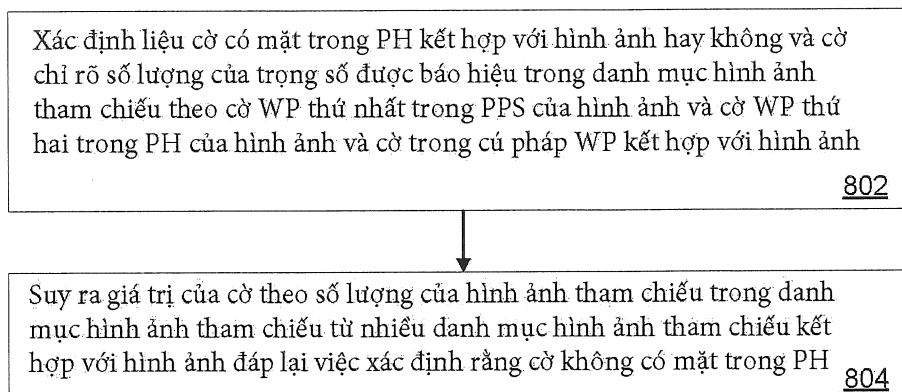


FIG. 8

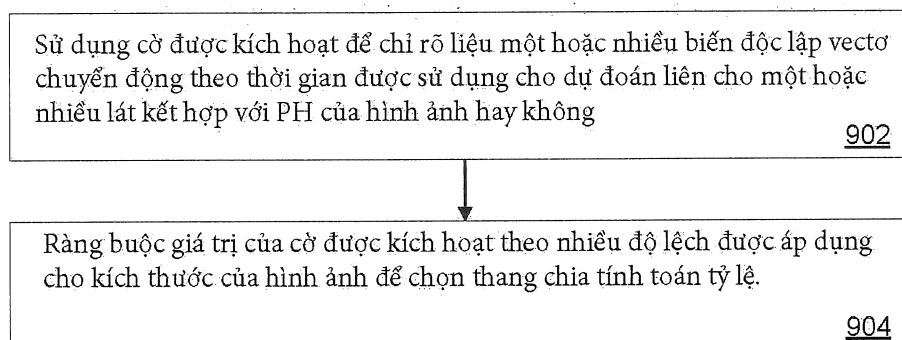


FIG. 9