



(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



1-0050271

(51)^{2020.01} H04N 19/52; H04N 19/573; H04N 19/54 (13) B

(21) 1-2020-02565 (22) 14/11/2018
(86) PCT/US2018/061085 14/11/2018 (87) WO 2019/099540 A1 23/05/2019
(30) 62/586,069 14/11/2017 US; 16/189,481 13/11/2018 US
(45) 25/08/2025 449 (43) 25/09/2020 390A
(73) QUALCOMM INCORPORATED (US)
ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, CA
92121-1714, United States of America
(72) ZHANG, Kai (CN); CHEN, Jianle (CN); LI, Xiang (CN); CHUANG, Hsiao-Chiang
(CN); KARCZEWICZ, Marta (US).
(74) Công ty TNHH Quốc tế D & N (D&N INTERNATIONAL CO.,LTD.)

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ MÃ HÓA VÀ GIẢI MÃ DỮ LIỆU VIDEO

(21) 1-2020-02565

(57) Sáng chế đề cập đến phương pháp và thiết bị mã hóa và giải mã dữ liệu video. Cụ thể là, sáng chế đề xuất phương pháp và hệ thống mã hóa dữ liệu video bằng cách sử dụng kỹ thuật bù chuyển động afin. Phương pháp này có thể bao gồm bước nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin và xây dựng danh sách bộ dự đoán vectơ chuyển động (motion vector predictor - MVP) afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vectơ chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vectơ chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời. Bộ mã hóa dữ liệu video có thể xác định các vectơ chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin và mã hóa khối dữ liệu video hiện thời với các vectơ chuyển động được xác định cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời.

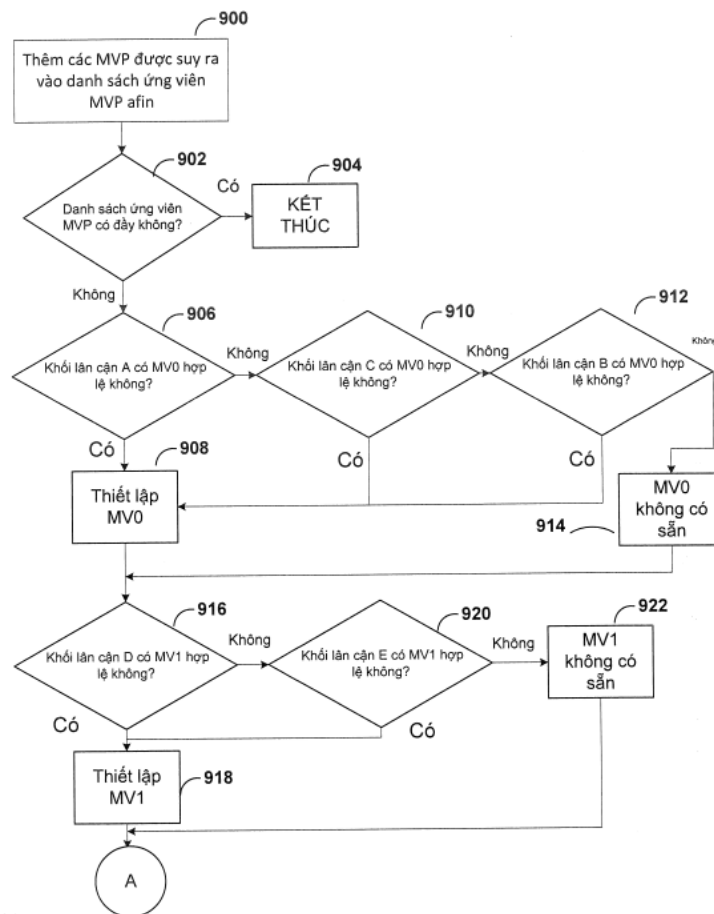


FIG.9A

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến thiết bị, hệ thống, và phương pháp mã hoá dữ liệu video.

Tình trạng kỹ thuật của sáng chế

Các tính năng video kỹ thuật số có thể được kết hợp vào nhiều loại thiết bị bao gồm các thiết bị truyền hình kỹ thuật số, các hệ thống phát quảng bá trực tiếp kỹ thuật số, các hệ thống phát quảng bá không dây, các thiết bị kỹ thuật số hỗ trợ cá nhân (Personal Digital Assistant - PDA), các máy tính xách tay hoặc các máy tính để bàn, các máy tính bảng, các máy đọc sách điện tử, các camera số hóa, các thiết bị ghi số hóa, các máy nghe nhạc số hóa, các thiết bị trò chơi điện tử, các bàn giao tiếp trò chơi điện tử, các điện thoại di động hoặc các điện thoại vô tuyến vệ tinh, các thiết bị được-gọi-là “các điện thoại thông minh”, các thiết bị hội thảo từ xa có truyền hình, các thiết bị truyền trực tiếp video, và các thiết bị tương tự. Các thiết bị video kỹ thuật số thực hiện các kỹ thuật nén dữ liệu video, chẳng hạn như các kỹ thuật được mô tả trong các chuẩn được quy định bởi MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Phần 10, mã hóa dữ liệu video nâng cao (Advanced Video Coding - AVC), chuẩn ITU-T H.265 mã hóa dữ liệu video hiệu suất cao (High Efficiency Video Coding - HEVC), và các phần mở rộng của các chuẩn này. Các thiết bị video có thể truyền, nhận, mã hóa, giải mã, và/hoặc lưu trữ thông tin video kỹ thuật số hiệu quả hơn bằng cách thực hiện các kỹ thuật nén dữ liệu video như vậy.

Các kỹ thuật nén dữ liệu video thực hiện dự đoán không gian (nội hình) và/hoặc dự đoán thời gian (liên hình) để giảm bớt hoặc loại bỏ phần dư vốn có trong các chuỗi dữ liệu video. Đối với việc mã hóa dữ liệu video dựa trên khối, lát video (tức là khung video hoặc một phần của khung video) có thể được phân chia thành các khối dữ liệu video, khối dữ liệu video còn có thể được gọi là khối cây, đơn vị mã hóa (coding unit - CU) và/hoặc nút mã hóa. Các khối dữ liệu video trong lát mã hóa nội hình (I) của hình được mã hóa bằng cách sử dụng kỹ thuật dự đoán không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng một hình. Các khối dữ liệu video trong lát mã hóa liên hình (P hoặc B) của hình có thể sử dụng kỹ thuật dự đoán không gian đối với các mẫu tham chiếu trong các khối lân cận trong cùng một hình, hoặc dự đoán thời gian đối với các mẫu tham chiếu trong các hình tham chiếu khác. Kỹ thuật dự đoán không gian hoặc thời gian tạo ra khối dự đoán

cho khối cần mã hóa. Dữ liệu dư đại diện cho độ lệch điểm ảnh giữa khối ban đầu cần mã hóa và khối dự đoán. Khối mã hóa liên hình được mã hóa theo vector chuyển động trở đến khối gồm các mẫu tham chiếu tạo thành khối dự đoán, và dữ liệu dư biểu thị độ lệch giữa khối mã hóa và khối dự đoán. Khối mã hóa nội hình được mã hóa theo chế độ mã hóa nội hình và dữ liệu dư. Để nén thêm, dữ liệu dư có thể được biến đổi từ miền điểm ảnh sang miền biến đổi, tạo ra các hệ số biến đổi dư, sau đó các hệ số này có thể được lượng tử hóa.

Bản chất kỹ thuật của sáng chế

Nói chung, sáng chế mô tả các kỹ thuật liên quan đến dự đoán liên hình. Cụ thể hơn, sáng chế này mô tả các kỹ thuật mã hóa vector chuyển động để bù chuyển động afin trong quá trình mã hóa dữ liệu video dựa trên khối. Các kỹ thuật của sáng chế này bao gồm bước xây dựng danh sách bộ dự đoán vector chuyển động afin để sử dụng trong quy trình mã hóa các vector chuyển động của một hoặc nhiều điểm điều khiển của mô hình chuyển động afin cho một khối hiện thời. Bộ mã hóa dữ liệu video (ví dụ, bộ mã hóa dữ liệu video hoặc bộ giải mã dữ liệu video) có thể xây dựng danh sách bộ dự đoán vector chuyển động afin cho khối hiện thời bằng cách sử dụng các vector chuyển động gắn với các khối lân cận.

Bộ mã hóa dữ liệu video có thể thêm vector chuyển động của các khối lân cận vào danh sách bộ dự đoán vector chuyển động afin nếu vector chuyển động trở đến hình tham chiếu giống hệt với hình tham chiếu đích. Hình tham chiếu đích là hình tham chiếu được xác định bởi danh sách tham chiếu và chỉ số hình tham chiếu được báo hiệu cho khối hiện thời.

Các kỹ thuật này có thể được áp dụng cho bộ mã hóa-giải mã dữ liệu video bất kỳ, ví dụ như Mã hóa dữ liệu video hiệu suất cao (High Efficiency Video Coding - HEVC) hoặc là công cụ mã hóa hiệu quả cho chuẩn mã hóa dữ liệu video bất kỳ trong tương lai, như H.266/mã hóa dữ liệu video đa năng (Versatile Video Coding - VVC).

Theo một ví dụ, sáng chế đề xuất phương pháp giải mã dữ liệu video, phương pháp này bao gồm bước nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin, xây dựng danh sách bộ dự đoán vector chuyển động (motion vector predictor - MVP) afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với

hình tham chiếu đích cho khối dữ liệu video hiện thời, xác định vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin, và giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời.

Theo ví dụ khác, sáng chế đề xuất thiết bị được tạo cấu hình để giải mã dữ liệu video, thiết bị này bao gồm bộ nhớ được tạo cấu hình để lưu trữ khối dữ liệu video hiện thời, và một hoặc nhiều bộ xử lý truyền thông với bộ nhớ, một hoặc nhiều bộ xử lý được tạo cấu hình để nhận khối dữ liệu video hiện thời cần được giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin, xây dựng danh sách MVP afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, xác định các vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin, và giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời.

Theo ví dụ khác, sáng chế đề xuất thiết bị được tạo cấu hình để giải mã dữ liệu video, thiết bị này bao gồm phương tiện nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin, phương tiện xây dựng danh sách MVP afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, phương tiện xác định các vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin, và phương tiện giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời.

Theo ví dụ khác, sáng chế đề xuất thiết bị được tạo cấu hình để mã hóa dữ liệu video, thiết bị này bao gồm bộ nhớ được tạo cấu hình để lưu trữ khối dữ liệu video hiện thời, và một hoặc nhiều bộ xử lý truyền thông với bộ nhớ, một hoặc nhiều bộ xử lý được tạo cấu hình để nhận khối dữ liệu video hiện thời cần được mã hóa bằng cách sử dụng kỹ thuật bù chuyển động afin, xây dựng danh sách MVP afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ

liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, xác định các vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin, và mã hóa khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời

Chi tiết về một hoặc nhiều khía cạnh của sáng chế được mô tả trong các hình vẽ kèm theo và phần mô tả chi tiết dưới đây. Các dấu hiệu, mục đích và ưu điểm khác của sáng chế sẽ được hiểu rõ hơn khi xem phần mô tả chi tiết sáng chế và hình vẽ, cùng với các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Fig. 1 là sơ đồ khối minh họa hệ thống mã hóa và giải mã dữ liệu video làm ví dụ mà có thể sử dụng một hoặc nhiều kỹ thuật được mô tả trong sáng chế này.

Các hình vẽ trên Fig.2A và Fig.2B là sơ đồ khái niệm minh họa ví dụ về cấu trúc cây nhị phân tứ phân (quadtree binary tree - QTBT), và đơn vị cây mã hóa (coding tree unit - CTU) tương ứng.

Fig.3A minh họa các ứng viên vector chuyển động (motion vector - MV) lân cận theo không gian của chế độ hợp nhất.

Fig.3B minh họa các ứng viên MV lân cận theo không gian của chế độ dự đoán vector chuyển động nâng cao (Advanced Motion Vector Prediction - AMVP).

Fig.4 là sơ đồ khái niệm minh họa ví dụ về chế độ afin MV hai điểm có bốn thông số afin.

Fig.5 minh họa trường vector chuyển động (motion vector field - MVF) afin trên mỗi khối con.

Fig.6 là sơ đồ khối minh họa khối hiện thời và các khối lân cận được sử dụng ở chế độ AF_INTER.

Fig.7A thể hiện các khối lân cận được sử dụng khi mã hóa khối hiện thời ở chế độ AF_MERGE.

Fig.7B minh họa chế độ AF_MERGE khi khối dưới bên trái lân cận được mã hóa trong chế độ afin.

Fig.8 là sơ đồ khái niệm của chế độ affin có sáu thông số.

Các hình vẽ trên Fig.9A và Fig.9B là lưu đồ làm ví dụ minh họa thủ tục xây dựng danh sách ứng viên dự đoán vector chuyển động affin được đề xuất.

Fig.10 là sơ đồ khối minh họa một ví dụ về bộ mã hóa dữ liệu video mà có thể thực hiện một hoặc nhiều kỹ thuật được mô tả trong sáng chế.

Fig.11 là sơ đồ khối minh họa một ví dụ về bộ giải mã dữ liệu video mà có thể thực hiện một hoặc nhiều kỹ thuật được mô tả trong sáng chế.

Fig.12 là lưu đồ minh họa một ví dụ về phương pháp mã hóa theo sáng chế.

Fig.13 là lưu đồ minh họa một ví dụ về phương pháp giải mã theo sáng chế.

Mô tả chi tiết sáng chế

Sáng chế này mô tả các kỹ thuật dự đoán vector chuyển động affin trong quy trình mã hóa dữ liệu video. Việc sử dụng các mô hình chuyển động affin đã được đề xuất để cung cấp khả năng nén hơn nữa dữ liệu video. Mô hình chuyển động affin cho khối dữ liệu video thể là kỹ thuật quay của khối trong một loạt các hình. Mô hình chuyển động affin của khối có thể được xác định dựa vào các vector chuyển động của các điểm điều khiển của khối. Theo một số phương án thực hiện, các điểm điều khiển của khối là các góc trên bên trái và trên bên phải của khối. Theo một số phương án thực hiện, các điểm điều khiển của khối còn bao gồm góc dưới bên trái của khối. Bộ mã hóa dữ liệu video (tức là, bộ mã hóa dữ liệu video hoặc bộ giải mã dữ liệu video) có thể xác định các vector chuyển động của các khối con của khối dựa vào các vector chuyển động của các điểm điều khiển của khối.

Hai kỹ thuật làm ví dụ đã được đề xuất để báo hiệu các vector chuyển động của các điểm điều khiển của khối. Kỹ thuật thứ nhất đôi khi được gọi là chế độ liên kết affin. Kỹ thuật thứ hai đôi khi được gọi là chế độ hợp nhất affin. Ở chế độ liên kết affin, bộ mã hóa dữ liệu video tạo ra danh sách ứng viên tập hợp MVP affin cho khối hiện thời. Danh sách ứng viên tập hợp MVP affin là danh sách gồm các tập hợp MVP affin. Mỗi tập hợp MVP affin là tập hợp các MVP tương ứng với các điểm điều khiển khác nhau của khối hiện thời. Bộ mã hóa dữ liệu video báo hiệu cho bộ giải mã dữ liệu video chỉ số để nhận dạng tập hợp MVP affin được chọn trong danh sách ứng viên tập hợp MVP affin. Ngoài ra, bộ mã hóa dữ liệu video báo hiệu giá trị chênh lệch vector chuyển động (motion vector difference - MVD) cho mỗi trong số các điểm điều khiển của khối hiện thời. Vector chuyển động của

điểm điều khiển có thể bằng MVD cho điểm điều khiển cộng với bộ dự đoán vector chuyển động cho điểm điều khiển trong tập hợp MVP afin được chọn. Bộ mã hóa dữ liệu video cũng báo hiệu chỉ số tham chiếu nhận dạng hình tham chiếu mà bộ giải mã dữ liệu video sử dụng với khối hiện thời. Bộ giải mã dữ liệu video tạo ra cùng danh sách ứng viên tập hợp MVP afin và sử dụng chỉ số đã báo hiệu để xác định tập hợp MVP afin được chọn. Bộ giải mã dữ liệu video có thể bổ sung giá trị chênh lệch MVD vào các vector chuyển động của tập hợp MVP afin được chọn để xác định vector chuyển động của các điểm điều khiển của khối hiện thời.

Ở chế độ hợp nhất afin, bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video nhận dạng cùng khối nguồn afin cho khối hiện thời. Khối nguồn afin có thể là khối được mã hóa afin mà lân cận theo không gian với khối hiện thời. Bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video ngoại suy các vector chuyển động của các điểm điều khiển của khối hiện thời từ các vector chuyển động của các điểm điều khiển của khối nguồn afin. Ví dụ, bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video có thể xây dựng mô hình chuyển động afin mô tả các vector chuyển động của các vị trí trong khối hiện thời. Mô hình chuyển động afin được xác định bởi tập hợp các thông số afin. Bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video có thể xác định các thông số afin dựa vào các vector chuyển động của các điểm điều khiển của khối hiện thời. Bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video có thể xác định các vector chuyển động của các điểm điều khiển của khối hiện thời dựa vào các vector chuyển động của các điểm điều khiển của khối nguồn afin.

Theo một số ví dụ, các kỹ thuật để xây dựng danh sách ứng viên tập hợp MVP afin có thể không hiệu quả. Tức là, một số kỹ thuật làm ví dụ để xây dựng danh sách ứng viên tập hợp MVP afin có thể đưa vào danh sách các ứng viên mà có thể dẫn đến mã hóa dữ liệu video gần điểm tối ưu về cả tốc độ mã hóa và/hoặc biến dạng. Sáng chế này mô tả các kỹ thuật xây dựng danh sách ứng viên tập hợp MVP afin hiệu quả hơn.

Fig.1 là sơ đồ khối minh họa một ví dụ về hệ thống mã hóa và giải mã dữ liệu video 10 mà có thể sử dụng các kỹ thuật theo sáng chế để xây dựng danh sách bộ dự đoán vector chuyển động afin. Như được thể hiện trên Fig.1, hệ thống 10 bao gồm thiết bị nguồn 12 cung cấp dữ liệu video được mã hóa cần giải mã sau đó bằng thiết bị đích 14. Cụ thể là, thiết bị nguồn 12 cung cấp dữ liệu video cho thiết bị đích 14 qua phương tiện đọc được bằng máy tính 16. Thiết bị nguồn 12 và thiết bị đích 14 có thể là thiết bị bất kỳ trong số

rất nhiều thiết bị, như máy tính để bàn, máy tính sổ tay (tức là, máy tính xách tay), máy tính bảng, đầu thu giải mã tín hiệu, máy điện thoại cầm tay như điện thoại “thông minh”, máy tính bảng “thông minh”, máy thu hình, camera, thiết bị hiển thị, máy phát phương tiện kỹ thuật số, bàn giao tiếp trò chơi có hình, thiết bị truyền trực tiếp dữ liệu video, hoặc các thiết bị tương tự khác. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị để truyền thông không dây. Do đó, thiết bị nguồn 12 và thiết bị đích 14 có thể là các thiết bị truyền thông không dây. Thiết bị nguồn 12 là thiết bị mã hóa dữ liệu video làm ví dụ (tức là, thiết bị để mã hóa dữ liệu video). Thiết bị đích 14 là thiết bị giải mã dữ liệu video làm ví dụ tức là, thiết bị để giải mã dữ liệu video).

Theo ví dụ trên Fig.1, thiết bị nguồn 12 bao gồm nguồn dữ liệu video 18, phương tiện lưu trữ 19 được tạo cấu hình để lưu trữ dữ liệu video, bộ mã hóa dữ liệu video 20, và giao diện xuất 22. Thiết bị đích 14 bao gồm giao diện nhập 26, phương tiện lưu trữ 28 được tạo cấu hình để lưu trữ dữ liệu video, bộ giải mã dữ liệu video 30, và thiết bị hiển thị 32. Trong các ví dụ khác, thiết bị nguồn 12 và thiết bị đích 14 có các bộ phận hoặc thiết bị khác. Ví dụ, thiết bị nguồn 12 có thể nhận dữ liệu video từ nguồn video bên ngoài, như camera bên ngoài. Tương tự, thiết bị đích 14 có thể giao tiếp với thiết bị hiển thị bên ngoài, chứ không phải bao gồm thiết bị hiển thị tích hợp.

Hệ thống 10 được minh họa trên Fig.1 chỉ là một ví dụ. Các kỹ thuật xử lý dữ liệu video có thể được thực hiện bởi thiết bị mã hóa và/hoặc giải mã dữ liệu video số bất kỳ. Mặc dù kỹ thuật theo sáng chế nhìn chung được thực hiện bởi thiết bị mã hóa dữ liệu video, nhưng các kỹ thuật này cũng có thể được thực hiện bởi bộ mã hóa/bộ giải mã dữ liệu video, thường được gọi là “bộ mã hóa-giải mã (CODEC).” Thiết bị nguồn 12 và thiết bị đích 14 chỉ là ví dụ về các thiết bị mã hóa như vậy, trong đó thiết bị nguồn 12 tạo ra dữ liệu video mã hóa để truyền cho thiết bị đích 14. Theo một số ví dụ, thiết bị nguồn 12 và thiết bị đích 14 có thể hoạt động theo cách gần như đối xứng sao cho mỗi thiết bị nguồn 12 và thiết bị đích 14 đều có các bộ phận mã hóa và giải mã dữ liệu video. Do đó, hệ thống 10 có thể hỗ trợ truyền dữ liệu video một chiều hoặc hai chiều giữa thiết bị nguồn 12 và thiết bị đích 14, ví dụ, để truyền trực tiếp dữ liệu video, phát lại dữ liệu video, phát quảng bá dữ liệu video, hoặc điện thoại có video.

Nguồn dữ liệu video 18 của thiết bị nguồn 12 có thể bao gồm thiết bị quay video, như camera ghi hình, kho lưu trữ dữ liệu video chứa dữ liệu video đã quay từ trước, và/hoặc

giao diện cung cấp dữ liệu video để nhận dữ liệu video từ nhà cung cấp nội dung video. Theo phương án thay thế khác nữa, nguồn dữ liệu video 18 có thể tạo ra dữ liệu dựa trên đồ họa từ máy tính làm dữ liệu video nguồn, hoặc tổ hợp của dữ liệu video thu phát trực tiếp, dữ liệu video đã được lưu trữ, và dữ liệu video được tạo ra trên máy tính. Thiết bị nguồn 12 có thể bao gồm một hoặc nhiều phương tiện lưu trữ dữ liệu (ví dụ, phương tiện lưu trữ 19) được tạo cấu hình để lưu trữ dữ liệu video. Các kỹ thuật nêu trong sáng chế có thể áp dụng cho mã hóa video nói chung, và có thể được áp dụng cho các ứng dụng không dây và/hoặc nối dây. Trong mỗi trường hợp, dữ liệu video được thu, thu trước hoặc tạo ra bởi máy tính có thể được mã hóa bằng bộ mã hóa dữ liệu video 20. Giao diện xuất 22 có thể xuất ra thông tin video mã hóa cho phương tiện đọc được bằng máy tính 16.

Giao diện xuất 22 có thể bao gồm các loại thành phần hoặc thiết bị khác nhau. Ví dụ, giao diện xuất 22 có thể bao gồm bộ phát không dây, môđem, thành phần mạng nối dây (ví dụ, thẻ Ethernet), hoặc thành phần vật lý khác. Trong các ví dụ mà giao diện xuất 22 bao gồm bộ thu không dây, giao diện xuất 22 có thể được tạo cấu hình để nhận dữ liệu, như luồng bit, được điều chế theo chuẩn truyền thông di động, như 4G, 4G-LTE, LTE nâng cao, 5G, và tương tự. Trong một số ví dụ mà giao diện xuất 22 bao gồm bộ thu không dây, giao diện xuất 22 có thể được tạo cấu hình để nhận dữ liệu, như luồng bit, được điều chế theo các chuẩn không dây khác, như tài liệu kỹ thuật IEEE 802.11, tài liệu kỹ thuật IEEE 802.15 (ví dụ, ZigBee™), chuẩn Bluetooth™, và tương tự. Trong một số ví dụ, hệ mạch của giao diện xuất 22 có thể được tích hợp vào hệ mạch của bộ mã hóa dữ liệu video 20 và/hoặc các thành phần khác của thiết bị nguồn 12. Ví dụ, bộ mã hóa dữ liệu video 20 và giao diện xuất 22 có thể là các bộ phận của hệ thống trên chip (system on a chip - SoC). SoC cũng có thể bao gồm các thành phần khác, như bộ vi xử lý đa dụng, bộ xử lý đồ họa, v.v.

Thiết bị đích 14 có thể nhận dữ liệu video mã hóa cần được giải mã qua phương tiện đọc được bằng máy tính 16. Phương tiện đọc được bằng máy tính 16 có thể bao gồm loại phương tiện hoặc thiết bị bất kỳ có khả năng truyền dữ liệu video mã hóa từ thiết bị nguồn 12 đến thiết bị đích 14. Trong một số ví dụ, phương tiện đọc được bằng máy tính 16 có thể bao gồm phương tiện truyền thông cho phép thiết bị nguồn 12 truyền dữ liệu video đã mã hóa trực tiếp đến thiết bị đích 14 theo thời gian thực. Dữ liệu video đã mã hóa có thể được điều chế theo chuẩn truyền thông, như giao thức truyền thông không dây, và

được truyền đến thiết bị đích 14. Phương tiện truyền thông có thể là mọi phương tiện truyền thông không dây hoặc có dây, như phổ tần số vô tuyến (Radio Frequency - RF) hoặc một hay nhiều đường truyền vật lý. Phương tiện truyền thông có thể tạo nên một phần mạng dựa vào gói, như mạng cục bộ, mạng diện rộng, hoặc mạng toàn cầu như Internet. Phương tiện truyền thông có thể có các bộ định tuyến, bộ chuyển mạch, trạm cơ sở, hoặc mọi thiết bị khác có thể có ích để hỗ trợ truyền thông từ thiết bị nguồn 12 đến thiết bị đích 14. Thiết bị đích 14 có thể bao gồm một hoặc nhiều phương tiện lưu trữ dữ liệu được tạo cấu hình để lưu trữ dữ liệu video mã hóa và dữ liệu video giải mã.

Theo một số ví dụ, dữ liệu đã mã hóa có thể được xuất ra từ giao diện xuất 22 cho thiết bị lưu trữ. Tương tự, dữ liệu đã mã hóa có thể được truy cập từ thiết bị lưu trữ bằng giao diện nhập. Thiết bị lưu trữ có thể là bất kỳ trong số nhiều phương tiện lưu trữ dữ liệu truy cập theo cách phân tán hoặc cục bộ như ổ cứng, đĩa Blu-ray, đĩa số đa năng (Digital Versatile Disc - DVD), đĩa compac-bộ nhớ chỉ đọc (Compact Disc-Read Only Memory - CD-ROM), bộ nhớ truy cập nhanh, bộ nhớ khả biến hoặc bất biến, hoặc phương tiện lưu trữ số thích hợp khác bất kỳ để lưu trữ dữ liệu video mã hóa. Theo ví dụ khác, thiết bị lưu trữ có thể tương ứng với máy chủ tập tin hoặc thiết bị lưu trữ trung gian khác có thể lưu trữ dữ liệu video mã hoá được tạo ra bằng thiết bị nguồn 12. Thiết bị đích 14 có thể truy cập dữ liệu video đã lưu trữ từ thiết bị lưu trữ thông qua cơ chế truyền trực tiếp hoặc tải xuống. Máy chủ tập tin có thể là loại máy chủ bất kỳ có khả năng lưu trữ dữ liệu video đã mã hoá và truyền dữ liệu video đã mã hoá đó cho thiết bị đích 14. Ví dụ về máy chủ tập là máy chủ web (ví dụ, cho website), máy chủ giao thức truyền tệp (File Transfer Protocol - FTP), thiết bị lưu trữ gắn với mạng (Network Attached Storage - NAS), hoặc ổ đĩa cục bộ. Thiết bị đích 14 có thể truy cập dữ liệu video đã mã hóa thông qua kết nối dữ liệu chuẩn bất kỳ, trong đó có kết nối Internet. Kết nối này có thể là kênh không dây (ví dụ, kết nối Wi-Fi), kết nối nối dây (ví dụ, đường thuê bao kỹ thuật số (Digital Subscriber Line - DSL), môđem cáp, v.v.), hoặc kết hợp hai loại này phù hợp với việc truy cập dữ liệu video mã hoá được lưu trữ trên máy chủ tập tin. Cuộc truyền dữ liệu video mã hoá từ thiết bị lưu trữ có thể là truyền trực tiếp, truyền tải xuống, hoặc kết hợp cả hai loại này.

Kỹ thuật này có thể được áp dụng để mã hóa dữ liệu video hỗ trợ ứng dụng bất kỳ trong nhiều ứng dụng đa phương tiện, như phát sóng truyền hình qua không khí, truyền hình cáp, truyền nối dây, truyền hình vệ tinh, truyền trực video qua Internet, như truyền

trực tiếp thích ứng động trên HTTP (dynamic adaptive streaming over HTTP - DASH), video số được mã hóa trên phương tiện lưu trữ dữ liệu, giải mã dữ liệu video số được lưu trữ trên phương tiện lưu trữ dữ liệu hoặc các ứng dụng khác. Trong một số ví dụ, hệ thống 10 có thể được tạo cấu hình để hỗ trợ truyền dữ liệu video một chiều hoặc hai chiều để hỗ trợ các ứng dụng như tạo dòng dữ liệu video, phát lại dữ liệu video, phát quảng bá dữ liệu video, và/hoặc điện thoại có truyền hình.

Phương tiện đọc được bằng máy tính 16 có thể là phương tiện khả biến, như phương tiện phát quảng bá không dây hoặc phương tiện truyền qua mạng nổi dây, hoặc là phương tiện lưu trữ (tức là, phương tiện lưu trữ không khả biến), như đĩa cứng, ổ đĩa flash, đĩa compac, đĩa video kỹ thuật số, đĩa Blu-ray, hoặc phương tiện khác đọc được bằng máy tính. Trong một số ví dụ, máy chủ mạng (không được thể hiện trên hình vẽ) có thể thu dữ liệu video mã hoá từ thiết bị nguồn 12 và cung cấp dữ liệu video mã hoá này cho thiết bị đích 14, ví dụ, qua cuộc truyền trên mạng. Tương tự, thiết bị tính toán của hãng sản xuất phương tiện, như hãng sản xuất đĩa, có thể thu dữ liệu video mã hoá từ thiết bị nguồn 12 và tạo ra đĩa chứa dữ liệu video mã hoá. Vì vậy, phương tiện đọc được bằng máy tính 16 có thể được hiểu là bao gồm một hoặc nhiều trong số các dạng phương tiện đọc được bằng máy tính, trong một số ví dụ.

Giao diện nhập 26 của thiết bị đích 14 thu thông tin từ phương tiện đọc được bằng máy tính 16. Thông tin của phương tiện đọc được bằng máy tính 16 có thể bao gồm thông tin cú pháp được xác định bởi bộ mã hóa dữ liệu video 20, cũng được bộ giải mã dữ liệu video 30 sử dụng, bao gồm phần tử cú pháp mô tả các đặc điểm và/hoặc quá trình xử lý các khối và các đơn vị mã hóa khác, ví dụ, các nhóm hình (group of pictures - GOP). Giao diện nhập 26 có thể bao gồm các loại thành phần hoặc thiết bị khác nhau. Ví dụ, giao diện nhập 26 có thể bao gồm bộ thu không dây, môđem, thành phần mạng nổi dây (ví dụ, thẻ Ethernet), hoặc một thành phần vật lý khác. Trong các ví dụ mà giao diện nhập 26 bao gồm bộ thu không dây, giao diện nhập 26 có thể được tạo cấu hình để thu dữ liệu, như luồng bit, được điều chế theo chuẩn truyền thông di động, như 4G, 4G-LTE, LTE nâng cao, 5G, và tương tự. Trong một số ví dụ mà giao diện nhập 26 bao gồm bộ thu không dây, giao diện nhập 26 có thể được tạo cấu hình để thu dữ liệu, như luồng bit, được điều chế theo các chuẩn không dây khác, như tài liệu kỹ thuật IEEE 802.11, tài liệu kỹ thuật IEEE 802.15 (ví dụ, ZigBee™), chuẩn Bluetooth™, và tương tự. Trong một số ví dụ, hệ mạch

của giao diện nhập 26 có thể được tích hợp vào hệ mạch của bộ giải mã dữ liệu video 30 và/hoặc các thành phần khác của thiết bị đích 14. Ví dụ, bộ giải mã dữ liệu video 30 và giao diện nhập 26 có thể là bộ phận của SoC. SoC cũng có thể bao gồm các thành phần khác, như bộ vi xử lý đa dụng, bộ xử lý đồ họa, v.v..

Phương tiện lưu trữ 28 có thể được tạo cấu hình để lưu trữ dữ liệu video mã hóa, như dữ liệu video mã hóa (ví dụ, luồng bit) thu được bởi giao diện nhập 26. Thiết bị hiển thị 32 hiển thị dữ liệu video đã giải mã cho người dùng và có thể bao gồm thiết bị bất kỳ trong số nhiều thiết bị hiển thị như màn hình ống phóng tia âm cực (cathode ray tube - CRT), màn hình tinh thể lỏng (liquid crystal display - LCD), màn hình plasma, màn hình điốt phát sáng hữu cơ (organic light emitting diode - OLED), hoặc loại thiết bị hiển thị khác.

Mỗi bộ mã hóa video 20 và bộ giải mã dữ liệu video 30 có thể được thực hiện dưới dạng mạch mã hóa hoặc mạch giải mã bất kỳ trong số rất nhiều mạch mã hóa hoặc giải mã phù hợp, nếu có, như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (digital signal processor - DSP), mạch tích hợp chuyên dụng (Application Specific Integrated Circuit - ASIC), mảng cổng lập trình được theo trường (Field Programmable Gate Array - FPGA), mạch logic rời rạc, phần mềm, phần cứng, phần sụn hoặc sự kết hợp bất kỳ của các loại trên. Khi các kỹ thuật này được thực hiện một phần trong phần mềm, thiết bị có thể lưu trữ các lệnh cho phần mềm trên phương tiện bất biến đọc được bằng máy tính thích hợp và thực thi các lệnh trong phần cứng bằng cách sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế này. Mỗi bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể nằm trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, mỗi bộ phận này có thể được tích hợp làm một phần của bộ mã hóa/giải mã (encoder/decoder - CODEC) kết hợp trong thiết bị tương ứng.

Trong một số ví dụ, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể hoạt động theo chuẩn mã hóa dữ liệu video như chuẩn hiện có hoặc chuẩn trong tương lai. Ví dụ về các chuẩn mã hóa dữ liệu video bao gồm, nhưng không bị giới hạn ở, ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 hoặc ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual và ITU-T H.264 (còn được biết đến là ISO/IEC MPEG-4 AVC), bao gồm các phiên bản mở rộng mã hóa dữ liệu video có khả năng điều chỉnh

(Scalable video Coding - SVC) và mã hóa dữ liệu video đa khung hình (Multiview video Coding - MVC).

Ngoài ra, chuẩn mã hóa dữ liệu video mới, cụ thể là chuẩn mã hóa dữ liệu video hiệu suất cao (HEVC) hoặc ITU-T H.265, bao gồm các phiên bản mở rộng mã hóa dải và nội dung màn hình của nó, các phiên bản mở rộng mã hóa dữ liệu video 3D (3D-HEVC) và đa khung nhìn (MV-HEVC) và phiên bản mở rộng có thể thay đổi tỷ lệ (SHVC), gần đây đã được phát triển bởi nhóm hợp tác liên kết về mã hóa dữ liệu video (Joint Collaboration Team on Video Coding - JCT-VC) cũng như Nhóm cộng tác chung về phát triển mở rộng mã hóa dữ liệu video 3D (Joint Collaboration Team on 3D Video Coding Extension Development - JCT-3V) thuộc Nhóm chuyên gia về mã hóa dữ liệu video (Video Coding Experts Group - VCEG) thuộc tổ chức ITU-T và Nhóm chuyên gia về hình động (Motion Picture Experts Group - MPEG) thuộc tổ chức ISO/IEC. Tài liệu của Wang và các đồng tác giả tên là "High Efficiency Video Coding (HEVC) Defect Report 2", JCT-VC của tổ chức ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC29/WG 11, Hội nghị lần thứ 15 tại Geneva, Thụy Sĩ, từ 23/10 đến 1/11/2013, tài liệu JCTVC-O1003_v2 là bản đặc tả kỹ thuật dự thảo cho HEVC, sau đây được gọi là HEVC WD, có thể xem tại http://phenix.int-evry.fr/jct/doc_end_user/documents/15_Geneva/wg11/JCTVC-O1003-v2.zip. Các phần mở rộng phạm vi của HEVC, gọi là HEVC RExt, cũng đang được JCT-VC phát triển. Flynn và các đồng tác giả, "High Efficiency Video Coding (HEVC) Range Extensions text specification: Draft 6," JCT-VC của tổ chức ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 16, tại San Jose, Hoa Kỳ, từ 9 đến 17/01/2014, tài liệu JCTVC-R1005_v1 là bản dự thảo làm việc mới về các phần mở rộng phạm vi, sau đây được gọi là RExt WD6, có thể xem tại: http://phenix.int-evry.fr/jct/doc_end_user/documents/16_San%20Jose/wg11/JCTVC-P1005-v1.zip. HEVC cũng được xuất bản dưới dạng Khuyến nghị ITU-T H.265, sêri H: Hệ thống nghe nhìn và đa phương tiện, các dịch vụ cơ sở hạ tầng nghe nhìn - mã hóa dữ liệu video chuyển động, mã hóa dữ liệu video hiệu quả cao, tháng 12/2016.

ITU-T VCEG (Q6/16) và ISO/IEC MPEG (JTC 1/SC 29/WG 11) hiện nay đang nghiên cứu nhu cầu tiềm năng để chuẩn hóa công nghệ mã hóa dữ liệu video trong tương lai với khả năng nén cao hơn đáng kể so với khả năng nén của chuẩn HEVC hiện tại (bao gồm các phiên bản mở rộng hiện tại của nó và các phiên bản mở rộng sắp tới để mã hóa

nội dung màn hình và mã hóa dải động cao). Nỗ lực chung này được gọi là Nhóm khảo sát dữ liệu video chung (Joint Video Exploration Team - JVET). Có bằng chứng cho thấy có thể đạt được những cải thiện đáng kể về hiệu quả mã hóa bằng cách khai thác các đặc điểm nội dung video, đặc biệt là đối với nội dung có độ phân giải cao như 4K, với các công cụ mã hóa chuyên dụng mới ngoài H.265/HEVC.

Nhóm JVET họp lần đầu tiên từ ngày 19 đến 21/10/2015. Phiên bản mới nhất của phần mềm tham chiếu, tức là Mô hình khảo sát chung 7 (Joint Exploration Model 7 - JEM 7) có thể được tải về từ: https://jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/tags/HM-16.6-JEM-7.0/. Mô tả thuật toán này cho JEM7 này có thể tham khảo đến tài liệu của J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm, J. Boyce "Algorithm description of Joint Exploration Test Model 7 (JEM7)", JVET-G1001, Torino, tháng 7 năm 2017.

Bản dự thảo ban đầu của chuẩn mã hóa dữ liệu video mới, được gọi là chuẩn H.266/Mã hóa dữ liệu video đa năng (Versatile Video Coding - VVC), có sẵn trong tài liệu JVET-J1001 "Versatile Video Coding (Draft 1)" của Benjamin Bross, và mô tả thuật toán của nó có sẵn trong tài liệu JVET-J1002 "Algorithm description for Versatile Video Coding and Test Model 1 (VTM 1)" của Jianle Chen và Elena Alshina. Tuy nhiên, các kỹ thuật theo sáng chế không chỉ giới hạn ở bất kỳ chuẩn mã hóa cụ thể nào. Phần mềm kiểm tra của VVC được gọi là mô hình thử nghiệm VVC (VVC Test Model - VTM).

Các kỹ thuật của sáng chế này có thể được sử dụng trong ngữ cảnh của bộ mã hóa giải mã dữ liệu video tiên tiến, như các phần mở rộng của HEVC hoặc thế hệ tiếp theo của các chuẩn mã hóa dữ liệu video. Mặc dù các kỹ thuật của sáng chế này thường được mô tả có liên quan đến HEVC và chuẩn mã hóa dữ liệu video thế hệ tiếp theo (ví dụ, JEM/VVC/VTM), cần phải hiểu rằng các kỹ thuật của sáng chế này có thể được sử dụng cùng với kỹ thuật mã hóa dữ liệu video bất kỳ để dự đoán vector chuyển động affin.

Theo một ví dụ, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để nhận khối dữ liệu video hiện thời cần mã hóa/giải mã bằng cách sử dụng kỹ thuật bù chuyển động affin, xây dựng danh sách bộ dự đoán vector chuyển động (MVP) affin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ một khối dữ liệu video lân cận vào danh sách MVP affin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu

đích cho khối dữ liệu video hiện thời, xác định vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin, và giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời.

Trong chuẩn HEVC và các tài liệu kỹ thuật mã hóa dữ liệu video khác, dữ liệu video bao gồm một loạt hình. Các hình có thể còn được gọi là các “khung”. Hình có thể bao gồm một hoặc nhiều mảng mẫu. Mỗi mảng mẫu tương ứng của hình có thể bao gồm mảng các mẫu cho thành phần màu tương ứng. Trong HEVC, hình có thể chứa ba mảng mẫu, được ký hiệu là S_L , S_{Cb} , và S_{Cr} . S_L là mảng hai chiều (tức là, khối) gồm các mẫu độ chói. S_{Cb} là mảng hai chiều gồm các mẫu màu Cb. S_{Cr} là mảng hai chiều gồm các mẫu màu Cr. Trong các trường hợp khác, hình có thể là hình đơn sắc và có thể chỉ bao gồm mảng các mẫu độ chói.

Là một phần của quá trình mã hóa dữ liệu video, bộ mã hóa dữ liệu video 20 có thể mã hóa các hình của dữ liệu video. Nói cách khác, bộ mã hóa dữ liệu video 20 có thể tạo ra các dạng biểu diễn mã hóa của các hình của dữ liệu video. Dạng biểu diễn mã hóa của hình có thể được gọi ở đây là “hình mã hóa” hoặc “hình được mã hóa”.

Để tạo ra dạng biểu diễn mã hóa của hình, bộ mã hóa dữ liệu video 20 có thể mã hóa các khối của hình. Bộ mã hóa dữ liệu video 20 có thể bao gồm, trong luồng bit, dạng biểu diễn được mã hóa của khối dữ liệu video. Ví dụ, để tạo ra dạng biểu diễn mã hóa của hình, bộ mã hóa dữ liệu video 20 có thể chia mỗi mảng mẫu của hình thành các khối cây mã hóa (coding tree block - CTB) và mã hóa các CTB. CTB có thể là khối mẫu $N \times N$ trong mảng mẫu của hình. Trong biên dạng chính HEVC, kích thước của CTB có thể nằm trong khoảng từ 16×16 đến 64×64 , mặc dù về mặt kỹ thuật thì các kích thước CTB 8×8 có thể được hỗ trợ.

Đơn vị cây mã hóa (coding tree unit - CTU) của hình có thể bao gồm một hoặc nhiều CTB và có thể bao gồm các cấu trúc cú pháp dùng để mã hóa các mẫu của một hoặc nhiều CTB. Ví dụ, mỗi CTU có thể bao gồm CTB gồm các mẫu độ chói, hai CTB tương ứng gồm các mẫu màu, và các cấu trúc cú pháp dùng để mã hóa các mẫu của các CTB. Trong các hình đơn sắc hoặc các hình có ba mặt phẳng màu riêng biệt, CTU có thể bao gồm CTB đơn và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu của CTB. CTU cũng có thể được gọi là “khối cây” hoặc “đơn vị mã hóa lớn nhất” (largest coding unit -

LCU). Trong bản mô tả này, “cấu trúc cú pháp” có thể được định nghĩa là 0 hoặc nhiều phần tử cú pháp có mặt cùng nhau luồng bit theo thứ tự được xác định. Trong một số bộ mã hóa-giải mã codec, hình mã hóa là một dạng biểu diễn mã hóa chứa tất cả các CTU của hình.

Để mã hóa CTU của hình, bộ mã hóa dữ liệu video 20 có thể chia các CTB của CTU thành một hoặc nhiều khối mã hóa. Khối mã hóa là khối mẫu $N \times N$. Để tạo ra CTU mã hóa, bộ mã hóa dữ liệu video 20 có thể thực hiện phân chia cây tứ phân đệ quy trên các khối cây mã hóa của CTU để chia khối cây mã hóa này thành các khối mã hóa, do đó có tên là “đơn vị cây mã hóa”. Đơn vị mã hóa (coding unit - CU) có thể bao gồm một hoặc nhiều khối mã hóa và các cấu trúc cú pháp dùng để mã hóa các mẫu của một hoặc nhiều khối mã hóa. Ví dụ, CU có thể bao gồm khối mã hóa gồm các mẫu độ chói và hai khối mã hóa tương ứng gồm các mẫu màu của hình có mảng mẫu độ chói, mảng mẫu Cb và mảng mẫu Cr và các cấu trúc cú pháp dùng để mã hóa các mẫu của khối mã hóa. Trong các hình đơn sắc hoặc các hình có ba mặt phẳng màu riêng biệt, CU có thể bao gồm một khối mã hóa và các cấu trúc cú pháp dùng để mã hóa các mẫu của khối mã hóa.

Hơn nữa, bộ mã hóa dữ liệu video 20 có thể mã hóa các CU của hình của dữ liệu video. Trong một số bộ mã hóa-giải mã codec, như là một phần của quá trình mã hóa CU, bộ mã hóa dữ liệu video 20 có thể phân chia khối mã hóa của CU thành một hoặc nhiều khối dự đoán. Khối dự đoán là khối hình chữ nhật (tức là, hình vuông hoặc không phải hình vuông) gồm các mẫu mà cùng một quy trình dự đoán được áp dụng trên đó. Đơn vị dự đoán (prediction unit - PU) của CU có thể bao gồm một hoặc nhiều khối dự đoán của CU và các cấu trúc cú pháp dùng để dự đoán một hoặc nhiều khối dự đoán. Ví dụ, PU có thể bao gồm khối dự đoán gồm các mẫu độ chói, hai khối dự đoán tương ứng gồm các mẫu màu và các cấu trúc cú pháp dùng để dự đoán các khối dự đoán này. Trong các hình đơn sắc hoặc các hình có ba mặt phẳng màu riêng biệt, PU có thể bao gồm một khối dự đoán và các cấu trúc cú pháp dùng để dự đoán khối dự đoán.

Bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự đoán (ví dụ, các khối dự đoán chói, Cb và Cr) cho các khối dự đoán (ví dụ, khối dự đoán chói, Cb và Cr) của mỗi PU của CU. Bộ mã hóa dữ liệu video 20 có thể sử dụng quy trình dự đoán nội hình hoặc dự đoán liên hình để tạo ra các khối dự đoán cho PU. Nếu bộ mã hóa dữ liệu video 20 sử dụng dự đoán nội hình để tạo ra khối dự đoán, thì bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự

đoán dựa vào các mẫu giải mã của hình bao gồm CU. Nếu bộ mã hóa dữ liệu video 20 sử dụng quy trình dự đoán liên hình để tạo ra khối dự đoán của CU của hình hiện thời, thì bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự đoán của CU dựa trên các mẫu giải mã của hình tham chiếu (tức là, hình khác với hình hiện thời).

Do đó, trong chuẩn HEVC, đơn vị mã hóa lớn nhất trong lát được gọi là khối cây mã hóa (coding tree block - CTB). CTB chứa cây tứ phân, các nút của chúng là các đơn vị mã hóa. Kích thước của CTB có thể nằm trong khoảng từ 16x16 đến 64x64 trong profin chính của HEVC (mặc dù về mặt kỹ thuật các kích thước CTB 8x8 có thể được hỗ trợ). Dù vậy, đơn vị mã hóa (CU) có thể có cùng kích thước với CTB và có thể có kích thước nhỏ đến 8x8. Mỗi đơn vị mã hóa được mã hóa với một chế độ. Khi CU được mã hóa liên hình, nó có thể tiếp tục được chia thành hai PU hoặc chỉ trở thành một PU khi không áp dụng việc chia thêm. Khi hai PU có mặt trong một CU, chúng có thể là các hình chữ nhật có kích thước bằng một nửa hoặc hai hình chữ nhật có kích thước bằng $\frac{1}{4}$ hoặc $\frac{3}{4}$ kích thước của CU. Khi CU được mã hóa liên hình, một tập hợp thông tin chuyển động xuất hiện trong mỗi PU. Hơn nữa, mỗi PU được mã hóa với chế độ dự đoán liên hình duy nhất để suy ra tập hợp thông tin chuyển động. Trong HEVC, các kích thước PU nhỏ nhất là 8x4 và 4x8.

Trong HEVC và các bộ mã hóa-giải mã cụ thể khác, bộ mã hóa dữ liệu video 20 mã hóa CU bằng cách chỉ sử dụng một chế độ dự đoán (tức là, dự đoán nội hình hoặc dự đoán liên hình). Do đó, trong HEVC và các bộ mã hóa-giải mã codec cụ thể khác, bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự đoán của CU bằng cách sử dụng dự đoán nội hình hoặc bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự đoán của CU bằng cách sử dụng dự đoán liên hình. Khi bộ mã hóa dữ liệu video 20 sử dụng kỹ thuật dự đoán liên hình để mã hóa CU, bộ mã hóa dữ liệu video 20 có thể phân chia CU thành 2 hoặc 4 PU, hoặc một PU tương ứng với toàn bộ CU. Khi hai PU có mặt trong một CU, hai PU này có thể là các hình chữ nhật có kích thước bằng một nửa hoặc hai hình chữ nhật có kích thước bằng $\frac{1}{4}$ hoặc $\frac{3}{4}$ kích thước của CU. Trong chuẩn HEVC, có tám chế độ phân chia cho CU được mã hóa với chế độ dự đoán liên hình, tức là, PART_2Nx2N, PART_2NxN, PART_Nx2N, PART_NxN, PART_2NxnU, PART_2NxnD, PART_nLx2N và PART_nRx2N. Khi CU được mã hóa nội hình, 2Nx2N và NxN chỉ là các dạng PU cho

phép, và trong mỗi PU, một chế độ dự đoán nội hình được mã hóa (trong khi đó chế độ dự đoán màu được báo hiệu ở mức CU).

Để dự đoán CU, bộ mã hóa dữ liệu video 20 thường có thể tạo ra khối dự đoán cho CU thông qua dự đoán liên hình hoặc dự đoán nội hình. Dự đoán liên hình thường được dùng để chỉ việc dự đoán CU từ dữ liệu của hình được mã hóa trước đó, trong khi dự đoán nội hình thường được dùng để chỉ việc dự đoán CU từ dữ liệu được mã hóa trước đó của cùng một hình. Để thực hiện dự đoán liên hình, bộ mã hóa dữ liệu video 20 có thể tạo ra khối dự đoán bằng cách sử dụng một hoặc nhiều vector chuyển động. Bộ mã hóa dữ liệu video 20 thường có thể thực hiện tìm kiếm chuyển động để xác định khối tham chiếu phù hợp nhất với CU, ví dụ, khác biệt giữa CU và khối tham chiếu. Bộ mã hóa dữ liệu video 20 có thể tính toán metric chênh lệch bằng cách sử dụng tổng chênh lệch tuyệt đối (sum of absolute difference - SAD), tổng chênh lệch bình phương (sum of squared difference - SSD), chênh lệch tuyệt đối trung bình (mean absolute difference - MAD), chênh lệch bình phương trung bình (mean squared difference - MSD) hoặc các phép toán chênh lệch như vậy khác để xác định xem khối tham chiếu có phù hợp nhất với CU hiện thời không.

Trong một số ví dụ, bộ mã hóa dữ liệu video 20 có thể dự đoán CU hiện thời bằng cách sử dụng dự đoán một chiều hoặc dự đoán hai chiều.

JEM/VVC cũng cung cấp chế độ bù chuyển động afin, chế độ này có thể được coi là chế độ dự đoán liên hình. Trong chế độ bù chuyển động afin, bộ mã hóa dữ liệu video 20 có thể xác định hai hoặc nhiều vector chuyển động đại diện cho chuyển động không dịch, chẳng hạn như phóng to hoặc thu nhỏ, xoay, chuyển động phối cảnh hoặc các loại chuyển động không đều khác.

Để thực hiện dự đoán nội hình, bộ mã hóa dữ liệu video 20 có thể lựa chọn chế độ dự đoán nội hình để tạo khối dự đoán. JEM/VVC cung cấp sáu mươi bảy chế độ dự đoán nội hình, bao gồm các chế độ định hướng khác nhau, cũng như chế độ phẳng và chế độ DC. Nói chung, bộ mã hóa video 20 chọn chế độ dự đoán nội ảnh để mô tả các mẫu lân cận cho khối hiện thời (ví dụ, khối của CU) để từ đó dự đoán các mẫu của khối hiện thời. Các mẫu như vậy thường có thể ở trên, bên trên và bên trái hoặc bên trái của khối hiện thời trong một hình giống như khối hiện thời, giả sử bộ mã hóa dữ liệu video 20 mã hóa CTU và CU theo thứ tự quét mảnh (từ trái sang phải, từ trên xuống dưới).

Bộ mã hóa dữ liệu video 20 mã hóa dữ liệu đại diện cho chế độ dự đoán cho khối hiện thời. Ví dụ, đối với các chế độ dự đoán liên hình, bộ mã hóa dữ liệu video 20 có thể mã hóa dữ liệu đại diện cho chế độ dự đoán liên hình nào trong số các chế độ dự đoán liên hình có sẵn, cũng như thông tin chuyển động cho chế độ tương ứng. Ví dụ, đối với dự đoán liên hình một chiều hoặc hai chiều, bộ mã hóa dữ liệu video 20 có thể mã hóa các vector chuyển động bằng cách sử dụng chế độ dự đoán vector chuyển động nâng cao (advanced motion vector prediction - AMVP) hoặc chế độ hợp nhất. Bộ mã hóa dữ liệu video 20 có thể sử dụng các chế độ tương tự để mã hóa các vector chuyển động cho chế độ bù chuyển động afin.

Bộ mã hóa dữ liệu video 20 có thể tạo ra một hoặc nhiều khối dữ liệu dư cho CU. Ví dụ, bộ mã hóa dữ liệu video 20 có thể tạo ra khối dữ liệu dư độ chói cho CU. Mỗi mẫu trong khối dữ liệu dư độ chói của CU biểu thị sự chênh lệch giữa mẫu độ chói trong một trong số các khối độ chói dự đoán của CU và mẫu tương ứng trong khối mã hóa độ chói ban đầu của CU. Ngoài ra, bộ mã hóa dữ liệu video 20 có thể tạo ra khối dữ liệu dư Cb cho CU. Mỗi mẫu trong khối dữ liệu dư Cb của CU có thể biểu thị sự chênh lệch giữa mẫu Cb trong một trong số các khối Cb dự đoán của CU và mẫu tương ứng trong khối mã hóa Cb ban đầu của CU. Bộ mã hóa dữ liệu video 20 còn có thể tạo ra khối dữ liệu dư Cr cho CU. Mỗi mẫu trong khối dữ liệu dư Cr của CU có thể biểu thị chênh lệch giữa mẫu Cr trong một trong số các khối Cr dự đoán của CU và mẫu tương ứng trong khối mã hóa Cr ban đầu của CU.

Hơn nữa, bộ mã hóa dữ liệu video 20 có thể phân tích các khối dữ liệu dư của CU thành một hoặc nhiều khối biến đổi. Hơn nữa, bộ mã hóa dữ liệu video 20 có thể sử dụng kỹ thuật phân chia cây tứ phân để phân chia các khối dư của CU thành một hoặc nhiều khối biến đổi. Khối biến đổi là khối hình chữ nhật (ví dụ, hình vuông hoặc không phải hình vuông) gồm các mẫu áp dụng cùng một phép biến đổi trên đó. Đơn vị biến đổi (transform unit - TU) của CU có thể bao gồm một hoặc nhiều khối biến đổi. Ví dụ, TU có thể bao gồm khối biến đổi gồm mẫu độ chói, hai khối biến đổi tương ứng gồm mẫu màu và cấu trúc cú pháp dùng để biến đổi các mẫu khối biến đổi. Do vậy, mỗi TU của CU có thể có khối biến đổi độ chói, khối biến đổi Cb, và khối biến đổi Cr. Khối biến đổi độ chói của TU có thể là khối con của khối dữ liệu dư độ chói của CU. Khối biến đổi Cb có thể là khối con của khối dữ liệu dư Cb của CU. Khối biến đổi Cr có thể là khối con của khối dữ liệu

liệu dư Cr của CU. Trong các hình đơn sắc hoặc các hình có ba mặt phẳng màu riêng biệt, TU có thể bao gồm một khối biến đổi và các cấu trúc cú pháp dùng để biến đổi các mẫu của khối biến đổi.

Trong JEM7 và các ví dụ về VVC, thay vì sử dụng cấu trúc phân chia cây tứ phân của HEVC được mô tả ở trên, có thể sử dụng cấu trúc phân chia cây nhị phân tứ phân (QTBT). Cấu trúc QTBT loại bỏ các khái niệm về nhiều loại phân chia. Tức là, cấu trúc QTBT loại bỏ sự tách biệt các khái niệm CU, PU và TU và hỗ trợ linh hoạt hơn cho các dạng phân chia CU. Trong cấu trúc khối QTBT, CU có thể có hình vuông hoặc hình chữ nhật. Cấu trúc QTBT của các ví dụ về JEM/VVC bao gồm hai mức: mức thứ nhất được phân chia theo kỹ thuật phân chia cây tứ phân và mức thứ hai được phân chia theo kỹ thuật phân chia cây nhị phân. Nút gốc của cấu trúc QTBT tương ứng với CTU. Các nút lá của cây nhị phân tương ứng với các đơn vị mã hóa (CU - coding unit). Theo một ví dụ, CU được phân chia trước tiên bởi cấu trúc cây tứ phân. Các nút lá cây tứ phân còn được phân chia bởi cấu trúc cây nhị phân.

Trong một số ví dụ, có hai loại chia: chia ngang đối xứng và chia dọc đối xứng. Các nút lá cây nhị phân được gọi là CU, và phân đoạn đó (tức là, CU) được sử dụng để xử lý dự đoán và biến đổi mà không cần phân chia thêm. Việc này có nghĩa là CU, PU và TU có cùng kích thước khối trong cấu trúc khối mã hóa QTBT. Trong JEM/VVC, CU đôi khi bao gồm các khối mã hóa (coding block - CB) có các thành phần màu khác nhau. Ví dụ, một CU chứa một CB độ chói và hai CB màu trong trường hợp các lát P và B có định dạng 4:2:0 và đôi khi bao gồm CB của một thành phần. Ví dụ, một CU chỉ chứa một CB độ chói hoặc chỉ hai CB màu trong trường hợp I lát.

Trong một số ví dụ, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng một cấu trúc QTBT duy nhất để biểu diễn từng thành phần độ chói và màu, trong khi trong các ví dụ khác, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng hai hoặc nhiều cấu trúc QTBT, chẳng hạn như cấu trúc QTBT cho thành phần độ chói và cấu trúc QTBT khác cho cả hai thành phần màu (hoặc hai cấu trúc QTBT cho các thành phần màu tương ứng).

Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để sử dụng kỹ thuật phân chia cây tứ phân cho mỗi HEVC, kỹ thuật phân chia QTBT theo các ví dụ JEM/VVC, hoặc các cấu trúc phân chia khác. Đối với mục đích giải thích, phần

mô tả về các kỹ thuật của sáng chế được trình bày liên quan đến kỹ thuật phân chia cây QTBT. Tuy nhiên, cần phải hiểu rằng các kỹ thuật của sáng chế này cũng có thể được áp dụng cho các bộ mã hóa dữ liệu video được tạo cấu hình để sử dụng kỹ thuật phân chia cây tứ phân hoặc các kiểu kỹ thuật phân chia khác.

Bộ mã hóa dữ liệu video 20 có thể áp dụng một hoặc nhiều phép biến đổi cho khối biến đổi của TU để tạo ra khối hệ số cho TU. Khối hệ số có thể là mảng hai chiều gồm các hệ số biến đổi. Hệ số biến đổi có thể là đại lượng vô hướng. Trong một số ví dụ, một hoặc nhiều phép biến đổi thực hiện biến đổi khối biến đổi từ miền điểm ảnh sang miền tần số. Do đó, trong các ví dụ này, hệ số biến đổi có thể là đại lượng vô hướng được coi là ở trong miền tần số. Mức hệ số biến đổi là đại lượng nguyên biểu diễn giá trị gắn với chỉ số tần số 2 chiều cụ thể trong quy trình giải mã trước khi định tỷ lệ để tính toán giá trị của hệ số biến đổi.

Trong một số ví dụ, bộ mã hóa dữ liệu video 20 bỏ qua việc áp dụng các phép biến đổi cho khối biến đổi. Trong các ví dụ như vậy, bộ mã hóa dữ liệu video 20 có thể xử lý các giá trị mẫu dữ liệu dư theo cách giống như các hệ số biến đổi. Do đó, trong các ví dụ trong đó bộ mã hóa dữ liệu video 20 bỏ qua việc áp dụng các hệ số biến đổi, phần mô tả sau đây về các hệ số biến đổi và các khối hệ số có thể áp dụng được cho các khối biến đổi gồm các mẫu dữ liệu dư.

Để tạo ra khối hệ số, bộ mã hóa dữ liệu video 20 có thể lượng tử hóa khối hệ số. Lượng tử hóa thường là quy trình trong đó các hệ số biến đổi được lượng tử hóa để có thể giảm bớt lượng dữ liệu được sử dụng để biểu diễn các hệ số biến đổi, nhằm đạt được hiệu quả nén cao hơn. Trong một số ví dụ, bộ mã hóa dữ liệu video 20 bỏ qua quy trình lượng tử hóa. Sau khi bộ mã hóa dữ liệu video 20 lượng tử hóa khối hệ số, bộ mã hóa dữ liệu video 20 có thể tạo ra các phần tử cú pháp biểu thị các hệ số biến đổi đã lượng tử hóa. Bộ mã hóa dữ liệu video 20 có thể mã hóa entropy một hoặc nhiều phần tử cú pháp chỉ báo các hệ số biến đổi được lượng tử hóa. Ví dụ, bộ mã hóa dữ liệu video 20 có thể thực hiện mã hóa số học nhị phân thích ứng theo ngữ cảnh (Context-Adaptive Binary Arithmetic Coding - CABAC) trên các phần tử cú pháp biểu thị các hệ số biến đổi được lượng tử hóa. Do đó, khối mã hóa (ví dụ, CU mã hóa) có thể bao gồm các phần tử cú pháp được mã hóa entropy chỉ báo các hệ số biến đổi được lượng tử hóa.

Bộ mã hóa dữ liệu video 20 có thể xuất ra luồng bit bao gồm dữ liệu video được mã hóa. Nói cách khác, bộ mã hóa dữ liệu video 20 có thể xuất ra luồng bit bao gồm dạng biểu diễn được mã hóa của dữ liệu video. Ví dụ, luồng bit có thể bao gồm chuỗi các bit tạo thành dạng biểu diễn của các hình mã hóa của dữ liệu video và dữ liệu kết hợp. Trong một số ví dụ, dạng biểu diễn của hình mã hóa có thể bao gồm dạng biểu diễn mã hóa của các khối.

Luồng bit có thể bao gồm chuỗi các đơn vị lớp trừu tượng hóa mạng (network abstraction layer - NAL). Đơn vị NAL là cấu trúc cú pháp chứa thông tin chỉ báo về loại dữ liệu trong đơn vị NAL và các byte chứa dữ liệu đó ở dạng tải tin chứa chuỗi byte thô (raw byte sequence payload - RBSP) được lồng ghép với các bit chống giả lập nếu cần. Mỗi đơn vị NAL có thể bao gồm phần đầu đơn vị NAL và đóng gói RBSP. Phần đầu đơn vị NAL có thể bao gồm phần tử cú pháp biểu thị mã kiểu đơn vị NAL. Mã kiểu đơn vị NAL được xác định bởi phần đầu đơn vị NAL của đơn vị NAL biểu thị kiểu đơn vị NAL. RBSP có thể là cấu trúc cú pháp chứa số byte nguyên được đóng gói trong đơn vị NAL. Trong một số trường hợp, RBSP bao gồm các bit 0.

Bộ giải mã dữ liệu video 30 có thể nhận luồng bit được tạo ra bởi bộ mã hóa dữ liệu video 20. Như đã nêu trên, luồng bit có thể bao gồm dạng biểu diễn mã hóa của dữ liệu video. Bộ giải mã dữ liệu video 30 có thể giải mã luồng bit để tái tạo các hình của dữ liệu video. Trong khi giải mã luồng bit, bộ giải mã dữ liệu video 30 có thể phân tích cú pháp luồng bit để thu được các phần tử cú pháp từ luồng bit. Bộ giải mã dữ liệu video 30 có thể tái tạo các hình của dữ liệu video dựa ít nhất một phần vào các phần tử cú pháp thu được từ luồng bit. Quy trình tái tạo các hình của dữ liệu video có thể thường ngược với quy trình được thực hiện bởi bộ mã hóa dữ liệu video 20 để mã hóa các hình. Ví dụ, bộ giải mã dữ liệu video 30 có thể sử dụng kỹ thuật dự đoán liên hình hoặc dự đoán nội hình để tạo ra một hoặc nhiều khối dự đoán cho mỗi PU của CU hiện thời có thể sử dụng các vector chuyển động của PU để xác định các khối dự đoán cho các PU của CU hiện thời. Ngoài ra, bộ giải mã dữ liệu video 30 có thể lượng tử hóa ngược các khối hệ số của các TU của CU hiện thời. Bộ giải mã dữ liệu video 30 có thể thực hiện biến đổi ngược trên các khối hệ số để tái tạo các khối biến đổi của các TU của CU hiện thời. Trong một số ví dụ, bộ giải mã dữ liệu video 30 có thể tái tạo các khối mã hóa của CU hiện thời bằng cách bổ sung các mẫu của các khối dự đoán cho các PU của CU hiện thời vào các mẫu được giải mã

tương ứng của các khối biến đổi của các TU của CU hiện thời. Bằng cách tái tạo các khối mã hóa cho mỗi CU của một hình, bộ giải mã dữ liệu video 30 có thể tái tạo hình.

Lát của hình có thể bao gồm số nguyên các CTU của hình. Các CTU của lát có thể được sắp xếp liên tiếp theo thứ tự quét, như thứ tự quét mảnh. Trong HEVC, lát có thể được định nghĩa là số nguyên các CTU có trong một đoạn lát độc lập và tất cả các đoạn lát phụ thuộc kế tiếp (nếu có) đứng trước đoạn lát độc lập tiếp theo (nếu có) trong cùng một đơn vị truy cập. Hơn nữa, trong HEVC, đoạn lát được định nghĩa là một số nguyên các đơn vị cây mã hóa được sắp xếp liên tiếp theo thứ tự quét ô và được chứa trong một đơn vị NAL. Quét ô là sự sắp xếp theo trình tự cụ thể của các CTB phân chia hình trong đó các CTB được sắp xếp liên tiếp theo thứ tự quét mảnh CTB trong ô, trong khi đó các ô trong hình được sắp xếp liên tiếp theo thứ tự quét mảnh của các ô của hình. Ô là vùng hình chữ nhật của các CTB trong một cột ô cụ thể và hàng ô cụ thể trong một hình. Phần đầu đoạn lát là một phần của đoạn lát mã hóa chứa các phần tử dữ liệu liên quan đến đơn vị cây mã hóa thứ nhất hoặc tất cả các đơn vị cây mã hóa được biểu diễn trong đoạn lát. Thuật ngữ "phần đầu lát" là phần đầu đoạn lát của đoạn lát độc lập tức là đoạn lát hiện thời hoặc đoạn lát độc lập gần nhất đứng trước đoạn lát phụ thuộc hiện thời theo thứ tự giải mã.

Các hình vẽ trên Fig.2A và Fig.2B là sơ đồ khái niệm minh họa một ví dụ về cấu trúc cây nhị phân tứ phân (quadtree binary tree - QTBT) 130, và đơn vị cây mã hóa (coding tree unit - CTU) 132 tương ứng. Các đường nét liền biểu thị việc tách cây tứ phân và các đường nét đứt biểu thị việc tách cây nhị phân. Trong mỗi nút tách (tức là, nút không phải lá) của cây nhị phân, một cờ được báo hiệu để chỉ báo kiểu tách nào (tức là, tách ngang hay tách dọc) được sử dụng, trong đó 0 biểu thị tách ngang và 1 biểu thị tách dọc trong ví dụ này. Đối với việc tách cây tứ phân, không cần thiết biểu thị kiểu tách, vì các nút cây tứ phân tách một khối theo chiều ngang và theo chiều dọc thành 4 khối con có kích thước bằng nhau. Do đó, bộ mã hóa dữ liệu video 20 có thể mã hóa, và bộ giải mã dữ liệu video 30 có thể giải mã, các phần tử cú pháp (ví dụ như thông tin tách) cho mức cây khu vực của QTBT 130 (tức là, các đường nét liền) và các phần tử cú pháp (ví dụ như thông tin tách) cho mức cây dự đoán của QTBT 130 (tức là, các đường nét đứt). Bộ mã hóa dữ liệu video 20 có thể mã hóa, và bộ giải mã video 30 có thể giải mã, dữ liệu video, như dữ liệu dự đoán và biến đổi, cho các CU được biểu diễn bởi các nút lá đầu cuối của cấu trúc QTBT 130.

Nói chung, CTU 132 trên Fig.2B có thể được kết hợp với các thông số xác định kích thước của các khối tương ứng với các nút của cấu trúc QTBT 130 ở mức thứ nhất và thứ hai. Các thông số này có thể bao gồm kích thước CTU (đại diện cho kích thước CTU 132 trong các mẫu), kích thước cây tứ phân tối thiểu (MinQTSIZE, đại diện cho kích thước nút lá cây tứ phân tối thiểu được phép), kích thước cây nhị phân tối đa (MaxBTSIZE, đại diện cho kích thước nút gốc cây nhị phân tối đa được phép), độ sâu cây nhị phân tối đa (MaxBTDepth, đại diện cho độ sâu cây nhị phân tối đa được phép) và kích thước cây nhị phân tối thiểu (MinBTSIZE, đại diện cho kích thước nút lá cây nhị phân tối thiểu được phép).

Nút gốc của cấu trúc QTBT tương ứng với CTU có thể có bốn nút con ở mức thứ nhất của cấu trúc QTBT, mỗi nút có thể được phân chia theo kỹ thuật phân chia cây tứ phân. Tức là, các nút của mức thứ nhất là các nút lá (không có nút con) hoặc có bốn nút con. Ví dụ về cấu trúc QTBT 130 đại diện cho các nút như vậy bao gồm nút cha và nút con có các đường liền nét cho các nhánh. Nếu các nút của mức thứ nhất không lớn hơn kích thước nút gốc cây nhị phân lớn nhất cho phép (MaxBTSIZE), thì chúng có thể được các cây nhị tương ứng phân chia tiếp. Việc phân chia cây nhị phân của một nút có thể được lặp lại cho đến khi các nút tạo ra từ quy trình phân tách đạt đến kích thước nút lá cây nhị phân tối thiểu được phép (MinBTSIZE) hoặc độ sâu cây nhị phân tối đa được phép (MaxBTDepth). Ví dụ về cấu trúc QTBT 130 đại diện cho các nút như vậy có các đường đứt nét cho các nhánh. Nút lá cây nhị phân được gọi là một đơn vị mã hóa (CU), đơn vị này được sử dụng để dự đoán (ví dụ, dự đoán nội hình hoặc dự đoán liên hình) và chuyển đổi, mà không cần phân vùng thêm. Như mô tả ở trên, các CU cũng có thể được gọi là các “khối dữ liệu video” hoặc “các khối”.

Trong một ví dụ của cấu trúc phân tách QTBT, kích thước CTU được thiết lập là 128x128 (mẫu độ chói và hai mẫu màu 64x64 tương ứng), MinQTSIZE được thiết lập là 16x16, MaxBTSIZE được thiết lập là 64x64, MinBTSIZE (cả chiều rộng lẫn chiều cao) được thiết lập là 4 và MaxBTDepth được thiết lập là 4. Phân tách cây tứ phân được áp dụng cho CTU trước tiên để tạo ra các nút lá cây tứ phân. Các nút lá cây tứ phân có thể có kích thước từ 16x16 (tức là MinQTSIZE) đến 128x128 (tức là, kích thước CTU). Nếu nút lá cây tứ phân là 128x128, nó sẽ không được cây nhị phân phân tách tiếp bởi vì kích thước này vượt quá MaxBTSIZE (tức là 64x64 trong ví dụ này). Nếu không, nút lá cây tứ phân sẽ được cây

nhị phân phân chia tiếp. Do đó, nút lá cây tứ phân cũng là nút gốc cho cây nhị phân và có độ sâu cây nhị phân bằng 0. Khi độ sâu cây nhị phân đạt đến `MaxBTDepth` (trong ví dụ này là bằng 4), thì không được phép tách tiếp nữa. Khi nút cây nhị phân có độ rộng bằng `MinBTSize` (bằng 4 trong ví dụ này), điều này biểu thị rằng không cho phép tách ngang thêm nữa. Tương tự, nút cây nhị phân có chiều cao bằng `MinBTSize` ngụ ý rằng không cho phép tách dọc đối với nút cây nhị phân đó. Như mô tả ở trên, các nút lá của cây nhị phân được gọi là các CU, và được xử lý thêm theo kỹ thuật dự đoán và biến đổi mà không cần phân chia thêm.

Trong H.265/HEVC, đối với mỗi khối, tập hợp thông tin chuyển động có thể có sẵn. Tập hợp thông tin chuyển động này chứa thông tin chuyển động cho các chiều dự đoán ngược và xuôi. Ở đây, các chiều dự đoán xuôi và ngược là hai chiều dự đoán của chế độ dự đoán hai chiều và các thuật ngữ "xuôi" và "ngược" không nhất thiết có ý nghĩa hình học. Thay vào đó, các thuật ngữ "xuôi" và "ngược" lần lượt tương ứng với danh sách hình tham chiếu 0 (`RefPicList0`) và danh sách hình tham chiếu 1 (`RefPicList1`) của hình hiện thời. Khi duy nhất một danh sách hình tham chiếu có sẵn cho hình hoặc lát, chỉ `RefPicList0` có sẵn và thông tin chuyển động của mỗi khối của lát là luôn theo chiều xuôi.

Đối với mỗi chiều dự đoán, thông tin chuyển động chứa chỉ số tham chiếu và vector chuyển động. Trong một số trường hợp, để đơn giản hóa, bản thân vector chuyển động có thể được tham chiếu theo cách mà giả định rằng nó có chỉ số tham chiếu gắn kèm. Chỉ số tham chiếu được dùng để nhận ra hình tham chiếu trong danh sách hình tham chiếu hiện thời (ví dụ, danh sách hình tham chiếu thứ nhất (`RefPicList0`) hoặc danh sách hình tham chiếu thứ hai (`RefPicList1`)). Vector chuyển động có thành phần ngang (x) và dọc (y) biểu thị sự dịch chuyển giữa khối hiện thời và khối dự đoán.

Số đếm thứ tự hình (POC - Picture Order Count) được sử dụng rộng rãi trong các chuẩn mã hóa dữ liệu video để nhận dạng thứ tự hiển thị của hình. Mặc dù có trường hợp hai hình trong một chuỗi video mã hóa có thể có cùng giá trị POC, điều này thường không xảy ra trong một chuỗi dữ liệu video mã hóa. Khi nhiều chuỗi dữ liệu video được mã hóa có mặt trong luồng bit, các hình có cùng giá trị POC có thể ở gần nhau xét theo thứ tự giải mã. Các giá trị POC của các hình thường được dùng để tạo ra danh sách hình tham chiếu, suy ra tập hợp hình tham chiếu như trong HEVC và định tỷ lệ vector chuyển động.

Như được mô tả ở trên, trong chuẩn HEVC, đơn vị mã hóa lớn nhất trong lát được gọi là khối cây mã hóa (coding tree block - CTB). CTB chứa cây tứ phân, các nút của chúng là các CU. Kích thước của CTB có thể nằm trong khoảng từ 16x16 đến 64x64 trong profin chính của HEVC (mặc dù về mặt kỹ thuật các kích thước CTB 8x8 có thể được hỗ trợ). Dù vậy, đơn vị mã hóa (CU) có thể có cùng kích thước với CTB và có thể có kích thước nhỏ đến 8x8. Mỗi đơn vị mã hóa được mã hóa với một chế độ. Khi CU được mã hóa liên hình, nó có thể tiếp tục được chia thành hai PU hoặc chỉ trở thành một PU khi không áp dụng việc chia thêm. Khi hai PU có mặt trong một CU, chúng có thể là các hình chữ nhật có kích thước bằng một nửa hoặc hai hình chữ nhật có kích thước bằng $\frac{1}{4}$ hoặc $\frac{3}{4}$ kích thước của CU.

Khi CU được mã hóa liên hình, thì có một tập hợp thông tin chuyển động cho mỗi PU. Hơn nữa, mỗi PU được mã hóa với chế độ dự đoán liên hình duy nhất để suy ra tập hợp thông tin chuyển động. Trong HEVC, các kích thước PU nhỏ nhất là 8x4 và 4x8.

Bộ mã hóa dữ liệu video (ví dụ, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30) có thể thực hiện dự đoán liên hình một chiều hoặc dự đoán liên hình hai chiều cho khối hiện thời (ví dụ, CU hoặc PU). Khi thực hiện dự đoán liên hình một chiều cho khối hiện thời, bộ mã hóa dữ liệu video sử dụng vector chuyển động để xác định vị trí trong hình tham chiếu. Bộ mã hóa dữ liệu video sau đó có thể tạo ra khối dự đoán cho khối hiện thời. Khối dự đoán có thể bao gồm khối mẫu trong hình tham chiếu tại vị trí được chỉ báo bởi vector chuyển động, hoặc khối mẫu được nội suy từ các mẫu của hình tham chiếu. Khi thực hiện dự đoán liên hình hai chiều, bộ mã hóa dữ liệu video có thể thực hiện quy trình này với hình tham chiếu thứ hai và vector chuyển động thứ hai, do đó tạo ra khối dự đoán thứ hai cho khối hiện thời. Trong dự đoán liên hình hai chiều, các khối dự đoán được tạo ra từ một hình tham chiếu trong đây có thể được gọi là các khối dự đoán sơ bộ. Hơn nữa, trong dự đoán liên hình hai chiều, bộ mã hóa dữ liệu video có thể tạo ra, dựa vào hai khối sơ bộ, khối dự đoán cuối cùng cho khối hiện thời. Trong một số ví dụ, bộ mã hóa dữ liệu video có thể tạo ra khối dự đoán cuối cùng sao cho mỗi khối trong khối dự đoán cuối cùng là giá trị trung bình có trọng số của các mẫu tương ứng trong các khối dự đoán sơ bộ.

Để hỗ trợ dự đoán liên hình trong hình, bộ mã hóa dữ liệu video (chẳng hạn, bộ mã hóa dữ liệu video 20 hoặc bộ giải mã dữ liệu video 30) tạo ra hai danh sách hình tham chiếu cho hình hiện thời. Các danh sách hình tham chiếu cho hình hiện thời bao gồm các

hình tham chiếu có sẵn để sử dụng trong việc thực hiện dự đoán liên hình cho các khối trong hình hiện thời. Hai danh sách hình tham chiếu thường được gọi là danh sách 0 và danh sách 1 (còn được gọi là RefPicList0 và RefPicList1). Theo một ví dụ, mỗi hình tham chiếu trong danh sách 0 của hình hiện thời xuất hiện trước hình hiện thời theo thứ tự xuất ra. Theo một ví dụ, mỗi hình tham chiếu trong danh sách 1 của hình hiện thời xuất hiện sau hình hiện thời theo thứ tự xuất ra. Do đó, việc sử dụng hình tham chiếu trong danh sách 0 có thể được xem là chiều dự đoán liên hình thứ nhất và việc sử dụng hình tham chiếu trong danh sách 1 có thể được xem là chiều dự đoán liên hình thứ hai. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tạo ra danh sách 0 cho hình hiện thời với các hình tham chiếu theo cùng thứ tự. Cũng giống như vậy, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tạo ra danh sách 1 cho hình hiện thời với các hình tham chiếu theo cùng thứ tự. Do đó, bộ mã hóa dữ liệu video 20 có thể chỉ báo cho bộ giải mã dữ liệu video 30 hình tham chiếu trong danh sách hình tham chiếu bằng cách báo hiệu chỉ số tham chiếu chỉ báo vị trí trong danh sách hình tham chiếu của hình tham chiếu.

Chuẩn HEVC bao gồm nhiều chế độ dự đoán liên hình, bao gồm chế độ hợp nhất (chế độ bỏ qua được xem là trường hợp hợp nhất đặc biệt) và chế độ dự đoán vectơ chuyển động cải tiến (advanced motion vector prediction - AMVP). Ở chế độ AMVP hoặc chế độ hợp nhất, danh sách ứng viên vectơ chuyển động (motion vector - MV) được duy trì cho nhiều bộ dự đoán vectơ chuyển động. (Các) vectơ chuyển động, cũng như các chỉ số tham chiếu trong chế độ hợp nhất, của PU hiện thời được xác định bằng cách lấy một ứng viên từ danh sách ứng viên MV. Danh sách ứng viên MV chứa tối đa 5 ứng viên cho chế độ hợp nhất và 2 ứng viên cho chế độ AMVP. Ứng viên hợp nhất có thể chứa tập hợp thông tin chuyển động, ví dụ, các vectơ chuyển động tương ứng với một hoặc nhiều trong số các danh sách hình tham chiếu (ví dụ, danh sách 0 và danh sách 1) và các chỉ số tham chiếu. Ứng viên hợp nhất được nhận dạng bởi chỉ số hợp nhất, và bộ giải mã dữ liệu video 30 xác định và sử dụng các hình tham chiếu cho việc dự đoán của các khối hiện thời, cũng như các vectơ chuyển động liên quan. Tuy nhiên, trong chế độ AMVP, đối với mỗi chiều dự đoán có thể có từ danh sách 0 hoặc danh sách 1, chỉ số tham chiếu được báo hiệu một cách rõ ràng, cùng với chỉ số bộ dự đoán vectơ chuyển động (motion vector predictor - MVP) cho danh sách ứng viên MV vì ứng viên AMVP chỉ chứa vectơ chuyển động. Chỉ số MVP

biểu thị ứng viên trong danh sách ứng viên MV. Trong chế độ AMVP, các vector chuyển động được dự đoán có thể được lọc thêm.

Như có thể thấy trên đây, ứng viên hợp nhất tương ứng với tập hợp đầy đủ thông tin chuyển động (ví dụ, chiều dự đoán, chỉ số tham chiếu và vector chuyển động) trong khi ứng viên AMVP chỉ chứa một vector chuyển động cho chỉ số tham chiếu và chiều dự đoán được báo hiệu cụ thể. Các ứng viên cho cả hai chế độ được suy ra một cách tương tự từ cùng một khối lân cận theo không gian và thời gian.

Cụ thể hơn, trong chế độ hợp nhất, mỗi bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tạo ra danh sách ứng viên vector chuyển động hợp nhất cho khối (ví dụ, PU hoặc CU). Danh sách ứng viên MV hợp nhất cho khối bao gồm một hoặc nhiều ứng viên hợp nhất, các ứng viên này cũng có thể được gọi là bộ dự đoán vector chuyển động (MVP - motion vector predictor). Trong HEVC, danh sách ứng viên MV hợp nhất có thể chứa tối đa 5 ứng viên hợp nhất. Mỗi ứng viên hợp nhất tương ứng trong danh sách ứng viên MV hợp nhất chỉ rõ một hoặc nhiều vector chuyển động và một hoặc nhiều chỉ số tham chiếu. Ví dụ, ứng viên hợp nhất có thể chỉ rõ vector chuyển động trong danh sách 0 và/hoặc vector chuyển động trong danh sách 1, và có thể chỉ rõ chỉ số tham chiếu danh sách 0 và/hoặc chỉ số tham chiếu danh sách 1. Vector chuyển động danh sách 0 là vector chuyển động chỉ báo vị trí trong hình tham chiếu trong danh sách 0. Vector chuyển động danh sách 1 là vector chuyển động chỉ báo vị trí trong hình tham chiếu trong danh sách 1. Bộ mã hóa dữ liệu video 20 có thể báo hiệu chỉ số hợp nhất biểu thị vị trí trong danh sách ứng viên MV hợp nhất của ứng viên hợp nhất được chọn cho PU. Bộ giải mã dữ liệu video 30 có thể sử dụng chỉ số hợp nhất để nhận dạng ứng viên hợp nhất được chọn. Bộ giải mã dữ liệu video 30 sau đó có thể sử dụng các vector chuyển động và chỉ số tham chiếu của ứng viên hợp nhất được chọn làm các vector chuyển động và chỉ số tham chiếu của PU.

Ở chế độ AMVP, bộ mã hóa dữ liệu video 20 tạo ra danh sách ứng viên AMVP danh sách 0 và/hoặc danh sách ứng viên AMVP danh sách 1 cho khối (ví dụ, PU hoặc CU), cả hai đều có thể được gọi là danh sách ứng viên AMVP. Bộ giải mã dữ liệu video 30 tạo ra các danh sách ứng viên AMVP phù hợp với các danh sách ứng viên AMVP được tạo ra bởi bộ mã hóa dữ liệu video 20. Trong HEVC, danh sách ứng viên AMVP chứa hai ứng viên AMVP. Mỗi ứng viên AMVP tương ứng trong danh sách ứng viên AMVP trong danh sách 0 chỉ rõ vector chuyển động trong danh sách 0 tương ứng. Mỗi ứng viên AMVP

tương ứng trong danh sách ứng viên AMVP trong danh sách 1 chỉ rõ vector chuyển động trong danh sách 1 tương ứng. Ở chế độ AMVP, nếu PU được dự đoán liên hình một chiều từ danh sách 0 hoặc được dự đoán liên hình hai chiều, bộ mã hóa dữ liệu video 20 báo hiệu chỉ số MVP danh sách 0, chỉ số tham chiếu danh sách 0, và giá trị chênh lệch vector chuyển động (MVD) danh sách 0. Chỉ số MVP danh sách 0 chỉ rõ vị trí của ứng viên AMVP được chọn trong danh sách ứng viên AMVP danh sách 0. Chỉ số tham chiếu danh sách 0 chỉ rõ vị trí của hình tham chiếu danh sách 0 được chọn. MVD danh sách 0 chỉ rõ giá trị chênh lệch giữa vector chuyển động danh sách 0 của PU và vector chuyển động danh sách 0 được chỉ rõ bởi ứng viên AMVP được chọn trong danh sách ứng viên AMVP danh sách 0. Do đó, bộ giải mã dữ liệu video 30 có thể sử dụng chỉ số MVP danh sách 0 và MVD danh sách 0 để xác định vector chuyển động danh sách 0 của PU. Bộ giải mã dữ liệu video 30 sau đó có thể xác định khối dự đoán sơ bộ hoặc cuối cùng cho PU bao gồm các mẫu tương ứng với vị trí trong hình tham chiếu danh sách 0 được chọn được nhận dạng bởi vector chuyển động danh sách 0 của PU. Bộ mã hóa dữ liệu video 20 có thể báo hiệu các phần tử cú pháp tương tự cho danh sách 1 và bộ giải mã dữ liệu video 30 có thể sử dụng các phần tử cú pháp cho danh sách 1 theo cách tương tự.

Các ứng viên MV không gian được suy ra từ các khối lân cận được thể hiện trên Fig.3A và Fig.3B, cho PU cụ thể (PU_0), mặc dù các phương pháp tạo ra các ứng viên từ các khối khác chế độ hợp nhất và AMVP. Fig.3A minh họa các ứng viên MV lân cận theo không gian cho chế độ hợp nhất. Ở chế độ hợp nhất, tối đa 4 ứng viên MV không gian có thể được suy ra theo thứ tự được thể hiện trên Fig.3A bằng các số, và thứ tự như sau: trái (0), trên (1), trên bên phải (2), dưới bên trái (3), và trên bên trái (4).

Fig.3B minh họa các ứng viên MV lân cận theo không gian đối với chế độ AMVP. Ở chế độ AMVP, các khối lân cận được chia thành 2 nhóm: nhóm bên trái bao gồm khối 0 và 1, và nhóm bên trên bao gồm các khối 2, 3, và 4 như được thể hiện trên Fig.3B. Đối với mỗi nhóm, ứng viên có thể có trong khối lân cận tham chiếu đến cùng một hình tham chiếu như được chỉ báo bởi chỉ số tham chiếu được báo hiệu có mức ưu tiên cao nhất cần chọn để tạo ra ứng viên cuối cùng của nhóm này. Có thể tất cả các khối lân cận không chứa vector chuyển động trở đến cùng một hình tham chiếu. Do đó, nếu không thể tìm thấy ứng viên như vậy, thì ứng viên sẵn có thứ nhất có thể được định tỷ lệ để tạo ra ứng viên cuối cùng, do đó giá trị chênh lệch khoảng cách thời gian có thể được bù.

Ví dụ, là một phần của quy trình tạo ra danh sách ứng viên AMVP danh sách 0, bộ mã hóa dữ liệu video (chẳng hạn, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30) kiểm tra xem khối 0 có được dự đoán từ danh sách 0 không, và nếu đúng như vậy thì kiểm tra xem hình tham chiếu danh sách 0 của khối 0 có giống với hình tham chiếu danh sách 0 của PU hiện thời hay không. Nếu khối 0 được dự đoán từ danh sách 0 và hình tham chiếu danh sách 0 của khối 0 giống với hình tham chiếu danh sách 0 của PU hiện thời, thì bộ mã hóa dữ liệu video bao gồm vector chuyển động danh sách 0 của khối 0 trong danh sách ứng viên AMVP danh sách 0. Nếu không, bộ mã hóa dữ liệu video kiểm tra xem khối 0 có được dự đoán từ danh sách 1 không, và nếu đúng như vậy, kiểm tra xem hình tham chiếu danh sách 1 của khối 0 có giống với hình tham chiếu danh sách 0 của PU hiện thời không. Nếu khối 0 được dự đoán từ danh sách 0 và hình tham chiếu danh sách 1 của khối 0 giống với hình tham chiếu danh sách 0 của PU hiện thời, thì bộ mã hóa dữ liệu video bao gồm vector chuyển động danh sách 1 của khối 0 trong danh sách ứng viên AMVP trong danh sách 0. Nếu hình tham chiếu danh sách 1 của khối 0 khác hình tham chiếu danh sách 0 của PU hiện thời, thì bộ mã hóa dữ liệu video lặp lại quy trình này với khối 1 thay vì khối 0.

Tuy nhiên, nếu khối 1 không được dự đoán từ danh sách 1 hoặc hình tham chiếu danh sách 1 của khối 1 khác hình tham chiếu danh sách 0 của PU hiện thời, thì bộ mã hóa dữ liệu video xác định xem khối 0 có được dự đoán từ danh sách 0 không và, nếu đúng như vậy, xác định xem hình tham chiếu danh sách 0 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là hình tham chiếu dài hạn hay đều là hình tham chiếu ngắn hạn. Nếu hình tham chiếu danh sách 0 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là các hình tham chiếu dài hạn hoặc hình tham chiếu danh sách 0 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là các hình tham chiếu ngắn hạn, thì bộ mã hóa dữ liệu video có thể định tỷ lệ vector chuyển động danh sách 0 của khối 0 dựa vào hiệu số tạm thời giữa hình tham chiếu danh sách 0 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời. Bộ mã hóa dữ liệu video đưa vector chuyển động danh sách 0 được định tỷ lệ vào danh sách ứng viên AMVP danh sách 0. Nếu hình tham chiếu danh sách 0 của khối 0 là hình tham chiếu dài hạn, và hình tham chiếu danh sách 0 của PU hiện thời là hình tham chiếu ngắn hạn, hoặc ngược lại, bộ mã hóa dữ liệu video xác định xem khối 0 có được dự đoán từ danh sách 1 không, và nếu đúng như vậy thì xác định xem hình

tham chiếu danh sách 1 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là hình tham chiếu dài hạn hay đều là hình tham chiếu ngắn hạn. Nếu hình tham chiếu danh sách 1 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là các hình tham chiếu dài hạn hoặc hình tham chiếu danh sách 1 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời đều là các hình tham chiếu ngắn hạn, thì bộ mã hóa dữ liệu video có thể định tỷ lệ vector chuyển động danh sách 1 của khối 0 dựa vào hiệu số tạm thời giữa hình tham chiếu danh sách 1 của khối 0 và hình tham chiếu danh sách 0 của PU hiện thời. Bộ mã hóa dữ liệu video đưa vector chuyển động danh sách 0 được định tỷ lệ vào danh sách ứng viên AMVP danh sách 0. Nếu hình tham chiếu danh sách 1 của khối 0 là hình tham chiếu dài hạn và hình tham chiếu danh sách 0 của PU hiện thời là hình tham chiếu ngắn hạn, hoặc ngược lại, thì bộ mã hóa dữ liệu video lặp lại quy trình này với khối 1 thay vì khối 0.

Bộ mã hóa dữ liệu video có thể thực hiện quy trình tương tự với các khối 2, 3, và 4 để đưa ứng viên thứ hai vào danh sách ứng viên AMVP danh sách 0 của PU hiện thời. Ngoài ra, bộ mã hóa dữ liệu video có thể lặp lại toàn bộ quy trình này, hoán đổi tham chiếu đến danh sách 0 với danh sách 1 và tham chiếu đến danh sách 1 với danh sách 0, để tạo ra danh sách ứng viên AMVP danh sách 1 của PU hiện thời.

Do đó, ở chế độ AMVP, các khối lân cận được chia thành 2 nhóm: nhóm bên trái bao gồm khối 0 và 1, và nhóm bên trên bao gồm các khối 2, 3, và 4 như được thể hiện trên Fig.3B. Đối với mỗi nhóm, ứng viên có thể có trong khối lân cận tham chiếu đến cùng một hình tham chiếu như được chỉ báo bởi chỉ số tham chiếu được báo hiệu có mức ưu tiên cao nhất cần chọn để tạo ra ứng viên cuối cùng của nhóm này. Có thể tất cả các khối lân cận không chứa vector chuyển động trỏ đến cùng một hình tham chiếu. Do đó, nếu không thể tìm thấy ứng viên như vậy, thì ứng viên sẵn có thứ nhất có thể được định tỷ lệ để tạo ra ứng viên cuối cùng, do đó hiệu số khoảng cách tạm thời có thể được bù.

Bộ mã hóa dữ liệu video có thể bao gồm ứng viên dự đoán vector chuyển động thời gian (temporal motion vector predictor - TMVP), nếu được kích hoạt và có sẵn, trong danh sách ứng viên MV hợp nhất sau ứng viên vector chuyển động không gian hoặc danh sách ứng viên AMVP. Ví dụ, trong trường hợp AMVP, bộ mã hóa dữ liệu video có thể bao gồm ứng viên TMVP trong danh sách ứng viên AMVP nếu các khối lân cận theo không gian không có sẵn (ví dụ, bởi vì các khối lân cận theo không gian ở ngoài biên hình, lát, hoặc

ô, bởi vì các khối lân cận theo không gian được dự đoán nội hình, v.v.). Ở chế độ hợp nhất, ứng viên TMVP có thể chỉ rõ các vector chuyển động trong danh sách 0 và/hoặc danh sách 1 của khối lân cận theo thời gian. Các chỉ số tham chiếu cho ứng viên TMVP trong chế độ hợp nhất luôn được thiết lập bằng 0. Ở chế độ AMVP, ứng viên TMVP chỉ rõ vector chuyển động danh sách 0 của khối lân cận theo thời gian hoặc vector chuyển động danh sách 1 của khối lân cận theo thời gian. Khối lân cận theo thời gian là khối trong hình tham chiếu. Quy trình suy ra vector chuyển động đối với ứng viên TMVP có thể giống với cả chế độ hợp nhất và chế độ AMVP.

Như đã nêu trên đây, bộ mã hóa dữ liệu video có thể định tỷ lệ vector chuyển động. Khi định tỷ lệ vector chuyển động, giả định rằng giá trị của vector chuyển động tỷ lệ với khoảng cách của các hình theo thời gian trình diễn. Vector chuyển động kết hợp hai hình, hình tham chiếu, và hình chứa vector chuyển động (gọi là hình chứa). Khi vector chuyển động được sử dụng để dự đoán vector chuyển động khác, khoảng cách của hình chứa và hình tham chiếu được tính dựa vào giá trị đếm thứ tự hình (Picture Order Count - POC) của hình tham chiếu và hình chứa.

Đối với vector chuyển động cần dự đoán, cả hình chứa gắn kèm và hình tham chiếu của vector chuyển động đều có thể khác nhau. Do đó khoảng cách mới (dựa vào POC) được tính toán. Vector chuyển động được định tỷ lệ dựa vào hai khoảng cách POC này. Đối với ứng viên lân cận trong không gian, các hình chứa cho hai vector chuyển động là giống nhau, còn các hình tham chiếu là khác nhau. Trong HEVC, việc định tỷ lệ vector chuyển động áp dụng cho cả TMVP và AMVP đối với các ứng viên lân cận theo thời gian và không gian.

Hơn nữa, theo một số phương án thực hiện, nếu danh sách ứng viên MV (ví dụ, danh sách ứng viên MV hợp nhất hoặc danh sách ứng viên AMVP) không hoàn chỉnh, bộ mã hóa dữ liệu video có thể tạo ra và chèn các ứng viên vector chuyển động giả vào cuối danh sách ứng viên MV cho tới khi danh sách ứng viên MV có số lượng ứng viên cần thiết. Ở chế độ hợp nhất, có hai loại ứng viên MV giả: ứng viên kết hợp được suy ra chỉ cho các lát B và ứng viên không. Ứng viên kết hợp chỉ rõ sự kết hợp của vector chuyển động trong danh sách 0 từ một ứng viên hợp nhất với vector chuyển động trong danh sách 1 cho ứng viên hợp nhất khác. Các ứng viên không được sử dụng để dự đoán vector chuyển động chỉ

khi loại thứ nhất (tức là, ứng viên kết hợp) không cung cấp đủ ứng viên giả. Ứng viên không là ứng viên chỉ rõ MV mà mỗi thành phần ngang và dọc của nó đều bằng 0.

Đối với mỗi cặp ứng viên đã có trong danh sách ứng viên và có thông tin chuyển động cần thiết, các ứng viên vector chuyển động kết hợp hai chiều được suy ra bằng kết hợp của vector chuyển động của ứng viên thứ nhất tham chiếu đến một hình trong danh sách 0 và vector chuyển động của ứng viên thứ hai tham chiếu đến một hình trong danh sách 1.

Ngoài ra, bộ mã hóa dữ liệu video có thể áp dụng quy trình lược bớt để chèn ứng viên. Các ứng viên từ các khối khác nhau có thể giống nhau, điều này có thể giảm hiệu suất của danh sách ứng viên hợp nhất/AMVP. Quy trình lược bớt được áp dụng để giải quyết vấn đề này. Quy trình lược bớt so sánh một ứng viên với các ứng viên khác trong danh sách ứng viên hiện thời để tránh chèn ứng viên giống nhau vào phạm vi nhất định. Để giảm độ phức tạp, chỉ một số quy trình lược bớt giới hạn được áp dụng thay vì so sánh mỗi ứng viên tiềm năng với tất cả các ứng viên hiện có khác.

Bù chuyển động trong H.265/HEVC được dùng để tạo ra bộ dự đoán cho khối liên hình hiện thời (ví dụ, khối được mã hóa bằng cách sử dụng kỹ thuật dự đoán liên hình). Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng vector chuyển động có độ chính xác một phần tư điểm ảnh và có thể nội suy các giá trị điểm ảnh ở các vị trí phân đoạn bằng cách sử dụng các giá trị điểm ảnh nguyên lân cận cho cả hai thành phần độ chói mà màu.

Trong một số tiêu chuẩn codec dữ liệu video làm ví dụ, chỉ một mô hình chuyển động dịch được áp dụng để dự đoán bù chuyển động (motion compensation prediction - MCP) (còn được gọi là dự đoán liên hình). Tuy nhiên, trên thực tế, có nhiều loại chuyển động, ví dụ chuyển động phóng to/ thu nhỏ, quay, phối cảnh và các chuyển động không đều khác. Nếu chỉ áp dụng mô hình chuyển động tịnh tiến cho MCP trong các chuỗi thử nghiệm như vậy với các mức chuyển động không đều, việc sử dụng chế độ chuyển động tịnh tiến như vậy sẽ ảnh hưởng đến độ chính xác dự đoán và có thể dẫn đến hiệu quả mã hóa thấp. Các chuyên gia về dữ liệu video đã cố gắng thiết kế các thuật toán để cải thiện MCP cho hiệu quả mã hóa cao hơn. Trong mô hình JEM, kỹ thuật dự đoán bù chuyển động biến đổi afin được áp dụng để cải thiện hiệu quả mã hóa. Chế độ hợp nhất afin và chế độ

liên hình afin được đề xuất để xử lý các mô hình chuyển động afin với bốn thông số, như sau:

$$\begin{cases} mv_x = ax + by + c \\ mv_y = dx + ey + f \end{cases} \quad (1)$$

Trong phương trình (1), mv_x là thành phần ngang của vector chuyển động cho vị trí (x, y) trong khối, và mv_y là thành phần dọc của vector chuyển động cho vị trí (x, y) trong khối. Trong phương trình (1), a, b, c, d, e , và f là các thông số. Lưu ý rằng trong mô hình chuyển động afin, các vị trí khác nhau trong khối có các vector chuyển động khác nhau.

Mô hình chuyển động afin 4 thông số có thể được biểu diễn bằng vector chuyển động của điểm điều khiển trên bên trái (V_0) và vector chuyển động của điểm điều khiển trên bên phải (V_1). Fig.4 minh họa mô hình chuyển động afin đơn giản hóa cho khối hiện thời 300. Như được thể hiện trên Fig.4, trường chuyển động afin của khối hiện thời 300 được mô tả bởi hai vector chuyển động điểm điều khiển MV_0 và MV_1 . MV_0 là vector chuyển động điểm điều khiển cho điểm điều khiển trên cùng bên trái 302 của khối hiện thời 300. MV_1 là vector chuyển động điểm điều khiển cho điểm điều khiển trên cùng bên phải 304 của khối hiện thời 300. Như vậy, (mv_{x_0}, mv_{y_0}) là vector chuyển động điểm điều khiển trên góc trên cùng bên trái, và (mv_{x_1}, mv_{y_1}) là vector chuyển động điểm điều khiển khác trên góc phải bên trên của khối hiện thời 300, như được thể hiện trên Fig.4.

Trường vector chuyển động (motion vector field - MVF) của khối được mô tả bằng phương trình sau đây:

$$\begin{cases} mv_x = \frac{(mv_{1x} - mv_{0x})}{w} x - \frac{(mv_{1y} - mv_{0y})}{w} y + mv_{0x} \\ mv_y = \frac{(mv_{1y} - mv_{0y})}{w} x + \frac{(mv_{1x} - mv_{0x})}{w} y + mv_{0y} \end{cases} \quad (2)$$

Trong phương trình (2), mv_x là thành phần ngang của vector chuyển động cho vị trí (x, y) trong khối; mv_y là thành phần dọc của vector chuyển động cho vị trí (x, y) trong khối; (mv_{0x}, mv_{0y}) là vector chuyển động của điểm điều khiển góc trên cùng bên trái (ví dụ, điểm điều khiển trên bên trái 302); (mv_{1x}, mv_{1y}) là vector chuyển động của điểm điều khiển góc trên cùng bên phải (ví dụ, điểm điều khiển trên cùng bên phải 304); và w là chiều rộng của khối. Do đó, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng phương trình (2) để “nội suy” các vector chuyển động cho các vị trí (x, y) dựa vào các vector chuyển động của các điểm điều khiển của khối. Theo các ví dụ của phần mềm JEM, dự đoán chuyển động affin chỉ được áp dụng cho các khối vuông. Là một phần mở rộng tự nhiên, dự đoán chuyển động affin cũng có thể được áp dụng cho các khối không vuông.

Trong một số ví dụ, để đơn giản hóa thêm dự đoán bù chuyển động, dự đoán biến đổi affin dựa trên khối có thể được áp dụng. Do đó, thay vì tạo ra các vector chuyển động cho từng vị trí trong khối, bộ mã hóa dữ liệu video (ví dụ, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30) có thể tạo ra các vector chuyển động cho các khối phụ của khối. Trong JEM, các khối con là các khối 4x4. Để tạo ra vector chuyển động của khối con, bộ mã hóa dữ liệu video có thể tính toán vector chuyển động của mẫu trung tâm của khối con theo phương trình (2). Sau đó bộ mã hóa dữ liệu video có thể làm tròn vector chuyển động đã tính đến độ chính xác phân số 1/16. Vector chuyển động đã làm tròn ở đây có thể được gọi là vector chuyển động có độ chính xác cao. Sau đó, bộ mã hóa dữ liệu video có thể áp dụng các bộ lọc nội suy bù chuyển động để tạo ra các dự đoán (tức là, các khối dự đoán) cho mỗi trong số các khối con có các vector chuyển động được tạo ra.

Fig.5 minh họa trường vector chuyển động (MVF) affin làm ví dụ trên mỗi khối con. Như thể hiện trong ví dụ trên Fig.5, khối hiện thời 400 có điểm điều khiển trên bên trái 402 và điểm điều khiển trên bên phải 404. Bộ mã hóa dữ liệu video có thể tính toán, dựa vào vector chuyển động 406 cho điểm điều khiển trên bên trái 402 và vector chuyển động 408 cho điểm điều khiển trên bên phải 404, các vector chuyển động cho các khối con của khối hiện thời 400. Fig.5 thể hiện các vector chuyển động của các khối con dưới dạng các mũi tên nhỏ.

Sau quy trình dự đoán MCP, vector chuyển động có độ chính xác cao của mỗi khối con được làm tròn và được lưu là có cùng độ chính xác với vector chuyển động thông thường. Trong một số ví dụ, việc làm tròn vector chuyển động có độ chính xác cao chỉ được

thực hiện khi độ chính xác của các vectơ chuyển động được lưu trữ nhỏ hơn các vectơ chuyển động có độ chính xác cao.

Theo một ví dụ, hai chế độ chuyển động afin được sử dụng trong JEM: Chế độ AF_INTER và chế độ AF_MERGE. Theo một ví dụ của JEM, chế độ AF_INTER có thể được áp dụng cho các CU có cả chiều rộng lẫn chiều cao lớn hơn 8. Cờ afin được báo hiệu ở mức CU trong luồng bit để chỉ báo việc chế độ AF_INTER có được sử dụng hay không. Ở chế độ AF_INTER, bộ mã hóa dữ liệu video 20 báo hiệu chỉ số tham chiếu danh sách 0 và/hoặc danh sách 1 chỉ số tham chiếu cho khối hiện thời để biểu thị hình tham chiếu danh sách 0 và/hoặc hình tham chiếu danh sách 1.

Trong chế độ AF_INTER, mỗi bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xây dựng một hoặc nhiều danh sách ứng viên (tức là danh sách ứng viên tập hợp MVP afin) cho khối hiện thời. Ví dụ, mỗi bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xây dựng danh sách ứng viên tập hợp MVP afin danh sách 0 và/hoặc danh sách ứng viên tập hợp MVP afin danh sách 1. Mỗi trong số các danh sách ứng viên tập hợp MVP afin bao gồm tập hợp tương ứng trong số các tập hợp MVP afin. Trong mô hình chuyển động afin 4 thông số, tập hợp MVP afin trong danh sách ứng viên tập hợp MVP afin trong danh sách 0 chỉ rõ hai vectơ chuyển động trong danh sách 0 (tức là, cặp vectơ chuyển động). Trong mô hình chuyển động afin 4 thông số, tập hợp MVP afin trong danh sách ứng viên tập hợp MVP afin danh sách 1 định rõ hai vectơ chuyển động danh sách 1.

Đối với mỗi CU/PU mà kích thước của nó bằng hoặc lớn hơn 16x16, chế độ AF_INTER có thể được áp dụng như sau. Nếu CU/PU hiện thời ở trong chế độ AF_INTER, cờ afin ở mức CU/PU được báo hiệu trong luồng bit. Danh sách ứng viên dự đoán vectơ chuyển động (MVP) afin có hai ứng viên là $\{(MVP^0_0, MVP^0_1), (MVP^1_0, MVP^1_1)\}$ được xây dựng. Chi phí tối ưu hóa tốc độ-độ méo (rate-distortion optimization - RDO) được dùng để xác định xem (MVP^0_0, MVP^0_1) hay (MVP^1_0, MVP^1_1) được chọn làm dự đoán vectơ chuyển động afin của CU/PU hiện. Nếu (MVP^x_0, MVP^x_1) được chọn, thì MV_0 được mã hóa với MVP^x_0 làm dự đoán và MV_1 được mã hóa với MVP^x_1 làm dự đoán. Chỉ số để biểu thị vị trí của ứng viên được chọn trong danh sách được báo hiệu cho khối hiện thời trong luồng bit.

Fig.6 là sơ đồ khối minh họa khối 500 hiện thời và các khối lân cận như được dùng trong chế độ AF_INTER. Trong một ví dụ, các khối lân cận từ A-F được thể hiện trên Fig.6 có thể được dùng để xây dựng danh sách ứng viên MVP afin cho khối hiện thời 500. Trong ví dụ này, các khối lân cận có thể được chia thành ba nhóm: Nhóm 0 (G0), Nhóm 1 (G1) và Nhóm 2 (G2). Nhóm G0 bao gồm khối trên bên trái 502 (A), khối trên thứ nhất 504 (B) và khối trái thứ nhất 506 (C). Các khối lân cận G0 có thể được dùng để xác định ứng viên cho MV0 của điểm điều khiển phía trên bên trái của khối hiện thời 500. Nhóm G1 bao gồm khối trên thứ hai 508 (D) và khối trên bên phải 510 (E). Các khối lân cận G1 có thể được dùng để xác định ứng viên cho MV1 của điểm điều khiển phía trên bên phải khối hiện thời 500. Nhóm G2 bao gồm khối thứ hai bên trái 512 (G) và khối dưới bên phải 514 (F). Các khối lân cận G2 có thể được dùng để xác định ứng viên cho MV2 của điểm điều khiển phía dưới bên trái khối 500 hiện thời.

Thủ tục xây dựng danh sách ứng viên MVP afin như sau.

- Thu thập các MV từ ba nhóm (G0, G1, G2)
 - G0: {MV-A, MV-B, MV-C}, G1: {MV-D, MV-E}, G2 {MV-F, MV-G}. Khối A, B, C, D, E, F và G được thể hiện trên Fig.6 của khối hiện thời 500. MV-A là vector chuyển động được liên kết với khối A, MV-B là vector chuyển động được liên kết với khối B, v.v.
 - Trước tiên đặt MV làm hình tham chiếu đích.
 - Sau đó, các MV được định tỷ lệ nếu không đề cập đến hình tham chiếu đích.
- Đối với bộ ba (MV0, MV1, MV2) từ G0, G1, G2, tạo ra MV2' từ MV0 và MV1 với mô hình afin để thu được $D(MV0, MV1, MV2) = |MV2 - MV2'|$
- Đi qua tất cả các bộ ba từ G0, G1 và G2, tìm bộ ba (MV00, MV01, MV02) mà tạo ra D tối thiểu, sau đó đặt $MVP^0_0 = MV00, MVP^0_1 = MV01$.
- Nếu có nhiều hơn một bộ ba có sẵn, tìm (MV10, MV11, MV12) mà tạo ra D tối thiểu thứ hai, sau đó đặt $MVP^1_0 = MV10, MVP^1_1 = MV11$.
- Nếu các ứng viên chưa được hoàn thành, thì các ứng viên MVP cho khối dự đoán không phải afin được tạo ra cho khối hiện thời. Ví dụ, các ứng viên MVP cho khối dự đoán không

phải afin là $MVP_nonaff0$ và $MVP_nonaff1$. Nếu (MVP^1_0, MVP^1_1) không thể được tìm thấy từ tìm kiếm bộ ba, thì sẽ đặt $MVP^1_0 = MVP^1_1 = MVP_nonaff0$.

Sau khi MVP của CU/PU afin hiện thời được xác định, ước lượng chuyển động afin được áp dụng và (MV^0_0, MV^0_1) được tìm thấy. Sau đó, hiệu số (MV^0_0, MV^0_1) và (MVP^x_0, MVP^x_1) được mã hóa trong luồng bit.

Dự đoán bù chuyển động afin nêu trên được áp dụng để tạo ra các giá trị dư của CU/PU hiện thời. Cuối cùng, các giá trị dư của CU/PU hiện thời được biến đổi, lượng tử hóa và mã hóa thành luồng bit như thủ tục truyền thống.

Khi CU/PU hiện thời được mã hóa bằng chế độ AF_MERGE , bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể thu được thông tin liên quan đến khối thứ nhất được mã hóa với chế độ afin từ các khối tái tạo lân cận hợp lệ. Fig.7A thể hiện các khối lân cận được sử dụng khi mã hóa khối hiện thời 700 ở chế độ AF_MERGE . Thứ tự lựa chọn cho các khối ứng viên liên quan đến khối hiện thời 700 là từ trái 702 (A), trên 704 (B), phía trên bên phải 706 (C), phía dưới bên trái 708 (D) đến phía trên bên trái 710 (E), như thể hiện trong 7A.

Trong một ví dụ, nếu khối dưới bên trái lân cận 702 (A) được mã hóa ở chế độ afin, như được thể hiện trên Fig.7B, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tạo ra các vector chuyển động mv_2 , mv_3 và mv_4 của góc trên bên trái, góc trên bên phải và góc dưới bên trái của CU/PU 712 chứa khối dưới bên trái 702 (A). Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tính toán vector chuyển động v_0 của góc trên bên trái của CU/PU 712 hiện thời theo mv_2 , mv_3 và mv_4 . Tương tự, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tính toán vector chuyển động v_1 của góc trên bên phải của CU/PU 712 hiện thời dựa vào mv_2 , mv_3 và mv_4 . Sau khi vector chuyển động điểm điều khiển (control point motion vector - CPMV) của CU/PU hiện thời mv_0 và mv_1 được xác định, ví dụ, theo mô hình chuyển động afin rút gọn được xác định trong phương trình (2), bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 tạo ra trường MVD của khối hiện thời 700 (chẳng hạn, CU/PU). Sau đó, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 áp dụng MCP afin, như được mô tả ở nơi khác trong bản mô tả này. Để xác định xem khối hiện thời 700 (CU/PU) có được mã hóa bằng chế độ AF_MERGE không, bộ mã hóa dữ liệu video 20 có thể báo hiệu cờ afin trong luồng bit cho bộ giải mã

dữ liệu video 30 khi có ít nhất một khối lân cận được mã hóa ở chế độ afin. Nếu khối afin lân cận khối hiện thời 700 không tồn tại (ví dụ, các khối lân cận được thể hiện trên Fig.7A), thì cờ afin không được viết trong luồng bit.

Trong một ví dụ, đối với Danh sách X trong đó $X = 0$ và/hoặc $X = 1$, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể ngoại suy vector chuyển động Danh sách X của điểm điều khiển trên cùng bên trái của khối lân cận 704 (B) để tạo ra vector chuyển động Danh sách X của điểm điều khiển trên cùng bên trái của khối hiện thời 700, sử dụng chỉ số tham chiếu Danh sách X của điểm điều khiển trên cùng bên trái của khối lân cận 704 (B) làm chỉ số tham chiếu Danh sách X của điểm điều khiển trên cùng bên trái của khối hiện thời 700, ngoại suy vector chuyển động Danh sách X của điểm điều khiển trên cùng bên phải của khối lân cận 704 (B) để tạo vector chuyển động danh sách X của điểm điều khiển trên cùng bên phải của khối hiện thời 700 và sử dụng chỉ số tham chiếu danh sách X của điểm điều khiển trên cùng bên phải của khối lân cận 704 (B) làm chỉ số tham chiếu Danh sách X của điểm điều khiển trên cùng bên phải của khối hiện thời 700.

Trong ví dụ này, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể sử dụng phương trình (2) nêu trên để ngoại suy vector chuyển động của điểm điều khiển của khối lân cận 704 (B) để xác định vector chuyển động của điểm điều khiển của khối hiện thời 700, sử dụng vị trí (x, y) của điểm điều khiển của khối hiện thời 700 là x và y theo phương trình (2).

Trong HEVC, kỹ thuật mã hóa số học nhị phân thích ứng ngữ cảnh (context adaptive binary arithmetic coding - CABAC) được dùng để chuyển đổi ký hiệu thành giá trị nhị phân. Kỹ thuật này được gọi là kỹ thuật nhị phân hóa. Quy trình nhị phân hóa cho phép mã hóa số học nhị phân hiệu quả qua phép ánh xạ duy nhất các phần tử cú pháp không nhị phân lên chuỗi các bit, chúng được gọi là các bin.

Trong phần mềm tham chiếu JEM2.0, đối với chế độ hợp nhất afin, chỉ có cờ afin được mã hóa, và chỉ số hợp nhất được suy ra là mô hình afin lân cận có sẵn thứ nhất theo thứ tự kiểm tra định trước A-B-C-D-E (ví dụ, các khối 702, 704, 706, 708 và 710 trên Fig.

7A). Đối với chế độ liên kết afin, hai cú pháp MVD được mã hóa cho mỗi danh sách dự

đoán biểu thị hiệu số vector chuyển động giữa vector chuyển động afin được suy ra và vector chuyển động được dự đoán.

Trong Đơn xin cấp bằng sáng chế Hoa Kỳ số 15/587,044, nộp ngày 4 tháng 5 năm 2017, sơ đồ dự đoán chuyển động afin có thể chuyển đổi được mô tả. Đối với khối được mã hóa bằng cách sử dụng dự đoán afin, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xác định sử dụng mô hình afin bốn thông số hoặc mô hình afin sáu thông số một cách thích ứng.

Mô hình afin có 6 thông số được định nghĩa là:

$$\begin{cases} mv_x = ax + by + e \\ mv_y = cx + dy + f \end{cases} \quad (3)$$

Mô hình afin với 6 thông số có ba điểm điều khiển. Nói cách khác, mô hình afin với 6 thông số được xác định bởi ba vector chuyển động, như thể hiện trên Fig.8. MV0 802 là CPMV thứ nhất ở góc trên bên trái của khối hiện thời 800, MV1 804 là CPMV thứ hai ở góc trên bên phải của khối hiện thời 800 và MV2 806 là vector chuyển động điểm điều khiển thứ ba ở góc dưới bên trái của khối hiện thời 800, như được thể hiện trên Fig.

8. Mô hình afin có thể mô tả chuyển động của khối hiện thời 800 vào vị trí thể hiện cho khối tham chiếu 808. Mô hình afin được tạo ra bởi ba vector chuyển động được tính toán như sau:

$$\begin{cases} mv_x = \frac{(mv_{1x} - mv_{0x})}{w}x - \frac{(mv_{2x} - mv_{0x})}{h}y + mv_{0x} \\ mv_y = \frac{(mv_{1y} - mv_{0y})}{w}x + \frac{(mv_{2y} - mv_{0y})}{h}y + mv_{0y} \end{cases} \quad (4)$$

trong đó w và h lần lượt là chiều rộng và chiều cao của khối.

Các kỹ thuật mô tả ở trên để thu được các vector chuyển động của góc trên bên trái và góc trên bên phải của khối hiện thời cũng có thể được dùng để thu được các bộ dự đoán MVP cho góc trên bên trái, góc trên bên phải và góc dưới bên trái. Các ví dụ khác cũng được mô tả trong Đơn sáng chế Hoa Kỳ số 15/725,052, nộp ngày 4 tháng 10 năm 2017. MVD1 có thể được dự đoán từ MVD ở chế độ afin, như được mô tả trong Đơn tạm thời

số 62,570,417 của Hoa Kỳ nộp ngày 10 tháng 10 năm 2017. Ví dụ, ở chế độ AMVP, MV cuối cùng được tính là MVP cộng với chênh lệch vector chuyển động (MVD). Chênh lệch vector chuyển động được xác định bởi bộ mã hóa dữ liệu video 20 là chênh lệch giữa vector chuyển động thực tế cho khối và MVP. Ở chế độ afin, có thể có nhiều hơn một vector CPMV và do đó, có thể có nhiều giá trị MVD. Một MVD có thể được dự đoán từ MVD khác. Trong một ví dụ, giá trị MVD của vector CPMV trên cùng bên trái được chọn để dự đoán giá trị MVD của các vector CPMV khác.

Sáng chế này đề xuất một số kỹ thuật để cải thiện dự đoán vector chuyển động afin. Sáng chế này mô tả các kỹ thuật để xây dựng danh sách MVP afin hiệu quả hơn. Các kỹ thuật phân loại sau đây có thể được áp dụng riêng. Ngoài ra, kết hợp bất kỳ của các kỹ thuật sau đây có thể được áp dụng cùng nhau.

Trong ví dụ thứ nhất của sáng chế, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để sử dụng vector chuyển động của khối lân cận làm MVP cho điểm điều khiển của khối hiện thời với chế độ afin chỉ khi nó có một hình tham chiếu giống với hình tham chiếu đích.

Hình tham chiếu đích là hình tham chiếu được xác định bởi danh sách tham chiếu và chỉ số hình tham chiếu được báo hiệu cho khối hiện thời. Một hình tham chiếu của khối lân cận giống với hình tham chiếu đích nếu chúng nằm trong cùng một danh sách tham chiếu và có cùng chỉ số hình tham chiếu. Theo một ví dụ khác, một hình tham chiếu của khối lân cận giống với hình tham chiếu đích nếu chúng có cùng số thứ tự hình ảnh (picture order count - POC).

Theo ví dụ thứ hai, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo0 cấu hình để tạo ra một MVP0 cho MV0 như sau:

- 1) Các khối lân cận A, B và C như được thể hiện trên Fig.6 được kiểm tra theo thứ tự xác định. Ví dụ, ba khối được kiểm tra theo thứ tự A, C và B. Theo một ví dụ khác, ba khối được kiểm tra theo thứ tự A, B và C. Theo một ví dụ khác nữa, ba khối được kiểm tra theo thứ tự C, B và A.

- 2) Khi kiểm tra các khối lân cận A, B và C theo thứ tự xác định, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để chọn khối có sẵn thứ nhất được mã hóa liên hình và có một hình tham chiếu, được đặt tên là RefX, giống hệt với hình tham chiếu đích.
- 3) Vector chuyển động của khối lân cận được chọn tham chiếu đến RefX được xuất ra dưới dạng MVP0. Nếu không có khối nào thỏa mãn điều kiện kiểm tra, thì MVP0 được đánh dấu là không có sẵn trong bước này.

Theo ví dụ thứ ba của sáng chế, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tạo ra một MVP1 cho MV1 như sau:

- 1) Các khối lân cận D và E như thể hiện trên Fig.6 được kiểm tra theo thứ tự xác định. Ví dụ, hai khối được kiểm tra theo thứ tự D và E. Theo một ví dụ khác, hai khối được kiểm tra theo thứ tự E và D.
- 2) Khi kiểm tra các khối lân cận D và E theo thứ tự xác định, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để chọn khối có sẵn thứ nhất được mã hóa liên hình và có một hình tham chiếu, được đặt tên là RefX, giống hệt với hình tham chiếu đích.
- 3) Vector chuyển động của khối lân cận được chọn tham chiếu đến RefX là đầu ra của MVP1. Nếu không có khối nào thỏa mãn điều kiện kiểm tra, thì MVP1 được đánh dấu là không có sẵn trong bước này.

Theo ví dụ thứ tư, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tạo ra một MVP2 cho MV2 như sau:

- 1) Các khối lân cận F và G như thể hiện trên Fig.6 được kiểm tra theo thứ tự xác định. Ví dụ, hai khối được kiểm tra theo thứ tự F và G. Theo một ví dụ khác, hai khối được kiểm tra theo thứ tự G và F.
- 2) Khi kiểm tra các khối lân cận F và G theo thứ tự xác định, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để chọn khối có sẵn thứ nhất được mã hóa liên hình và có một hình tham chiếu, được đặt tên là RefX, giống hệt với hình tham chiếu đích.

- 3) Vector chuyển động của khối lân cận được chọn tham chiếu đến RefX là đầu ra của MVP2. Nếu không có khối nào thỏa mãn điều kiện kiểm tra, thì MVP2 được đánh dấu là không có sẵn trong bước này.

Theo ví dụ thứ năm của sáng chế, nếu MVP0 (ví dụ thứ hai) và MVP1 (ví dụ thứ ba) có sẵn, nhưng MVP2 (ví dụ thứ tư) không có sẵn, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tính MVP2 là:

$$\begin{cases} mvp_2^x = -\frac{(mvp_1^y - mvp_0^y)}{w}h + mvp_0^x \\ mvp_2^y = \frac{(mvp_1^x - mvp_0^x)}{w}h + mvp_0^y \end{cases}$$

trong đó MVP2= (mvp^x₂, mvp^y₂), MVP1= (mvp^x₁, mvp^y₁) và MVP0= (mvp^x₀, mvp^y₀), và w và h lần lượt là chiều rộng và chiều cao của khối hiện thời. Theo một ví dụ khác, hoạt động phân chia có thể được thay thế bằng hoạt động dịch. Ví dụ,

$$\begin{cases} mvp_2^x = -(((mvp_1^y - mvp_0^y)h + offset) >> wb) + mvp_0^x \\ mvp_2^y = (((mvp_1^x - mvp_0^x)h + offset) >> wb) + mvp_0^y \end{cases}$$

trong đó w=2^{wb}. offset là số nguyên bất kỳ, như 0 hoặc w/2. Theo một ví dụ khác, hoạt động phân chia có thể được thay thế bằng bảng tìm kiếm.

Theo ví dụ thứ sáu của sáng chế, nếu MVP0 (ví dụ thứ hai) và MVP2 (ví dụ thứ tư) có sẵn, nhưng MVP1 (ví dụ thứ ba) không có sẵn, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tính MVP1 là

$$\begin{cases} mvp_1^x = \frac{(mvp_2^y - mvp_0^y)}{h}w + mvp_0^x \\ mvp_1^y = -\frac{(mvp_2^x - mvp_0^x)}{h}w + mvp_0^y \end{cases}$$

trong đó MVP2= (mvp^x₂, mvp^y₂), MVP1= (mvp^x₁, mvp^y₁) và MVP0= (mvp^x₀, mvp^y₀), và w và h lần lượt là chiều rộng và chiều cao của khối hiện thời. Theo một ví dụ khác, hoạt động phân chia có thể được thay thế bằng hoạt động dịch. Ví dụ,

$$\begin{cases} mvp_1^x = (((mvp_2^y - mvp_0^y)w + offset) >> hb) + mvp_0^x \\ mvp_1^y = -(((mvp_2^x - mvp_0^x)w + offset) >> hb) + mvp_0^y \end{cases}$$

trong đó h=2^{wh}. offset là số nguyên bất kỳ, như 0 hoặc h/2. Theo một ví dụ khác, hoạt động phân chia có thể được thay thế bằng bảng tìm kiếm.

Theo ví dụ thứ bảy của sáng chế, nếu MVP1 (ví dụ thứ ba) và MVP2 (ví dụ thứ tư) có sẵn, nhưng MVP0 (có sẵn thứ hai) không có sẵn, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tính MVP1 là

$$\begin{cases} mvp_0^x = \frac{(w^2 \times mvp_1^x - w \times h \times mvp_1^y + h^2 \times mvp_2^x + w \times h \times mvp_2^y)}{w^2 + h^2} \\ mvp_0^y = \frac{(w \times h \times mvp_1^x + w^2 \times mvp_1^y - w \times h \times mvp_2^x + h^2 \times mvp_2^y)}{w^2 + h^2} \end{cases}$$

trong đó MVP2= (mvp_2^x, mvp_2^y) , MVP1= (mvp_1^x, mvp_1^y) và MVP0= (mvp_0^x, mvp_0^y) , và w và h lần lượt là chiều rộng và chiều cao của khối hiện thời. Theo một ví dụ khác, hoạt động phân chia có thể được thay thế bằng bảng tìm kiếm.

Theo ví dụ thứ tám của sáng chế, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để làm tròn MVP0, MVP1 và MVP2 đầu ra theo độ chính xác chuyển động được xác định trước của MVD. Ví dụ, nếu độ chính xác tích trữ chuyển động là 1/16 điểm ảnh, nhưng độ chính xác MVD là 1/4 điểm ảnh, thì các MVP đầu ra được làm tròn là

$$MVP0' = (MVP0 \gg 2) \ll 2, MVP1' = (MVP1 \gg 2) \ll 2, MVP3' = (MVP2 \gg 2) \ll 2$$

Trong ví dụ thứ chín, nếu một khối liên hình áp dụng mô hình afin 6 thông số, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để tạo ra và xuất ra MVP0, MVP1 và MVP2 dưới dạng một ứng viên MVP afin. Nếu một khối liên hình áp dụng mô hình afin 4 thông số, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để chỉ tạo ra và xuất MVP0, MVP1 dưới dạng một ứng viên MVP afin.

Theo ví dụ thứ mười, nếu có ít hơn hai trong số MVP0, MVP1 và MVP2 có sẵn, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để không xuất ra MVP0, MVP1 và MVP2 dưới dạng một ứng viên MVP.

Theo ví dụ thứ mười một của sáng chế, nếu MVP0 không có sẵn, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để không xuất ra MVP0, MVP1 và MVP2 dưới dạng một ứng viên MVP.

Theo một ví dụ thứ mười hai của sáng chế, các hình vẽ trên Fig.9A và 9B minh họa một sơ đồ làm ví dụ của thủ tục xây dựng danh sách ứng viên MVP afin được đề xuất.

Trong ví dụ mô tả bên dưới, khối lân cận có MV hợp lệ trong trường hợp khối lân cận có một hình tham chiếu, được đặt tên là RefX, giống với hình tham chiếu đích.

Như được thể hiện trên Fig.9A, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 trước tiên có thể thêm các MVPS được suy ra vào danh sách ứng viên MVP afin (900). Ví dụ, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể thu được MVP bằng cách sử dụng bất kỳ trong số các kỹ thuật trước đây được mô tả ở trên. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 sau đó có thể xác định xem danh sách ứng viên MVP có đầy không (ví dụ, danh sách ứng viên MVP bao gồm số lượng ứng viên xác định trước) (902). Nếu đầy, thì quá trình kết thúc (904). Nếu chưa, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xác định trong khối A lân cận trên Fig.6 có ứng viên hợp lệ cho MV0 (906). Nếu có, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đặt vector chuyển động của khối A lân cận là MV0 (908).

Nếu không, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể thực hiện các bước kiểm tra tương tự cho khối C lân cận (910), và sau đó là khối B lân cận (912). Nếu khối C lân cận có ứng viên hợp lệ cho MV0, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đặt vector chuyển động của khối C lân cận là MV0 (908). Nếu khối C lân cận không có ứng viên hợp lệ, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xác định xem khối B lân cận có ứng viên hợp lệ cho MV0 hay không (912). Nếu khối B lân cận có ứng viên hợp lệ cho MV0, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đặt vector chuyển động của khối B lân cận là MV0 (908). Nếu khối C lân cận không có ứng viên hợp lệ, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đánh dấu MV0 là không có ứng viên có sẵn (914).

Quy trình sau đó chuyển đến các khối D và E lân cận trên Fig.6. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 trước tiên xác định xem khối D lân cận có ứng viên hợp lệ cho MV1 hay không (916). Nếu có, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 thiết lập ứng viên cho MV1 làm vector chuyển động của khối D lân cận (918). Nếu không, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 xác định xem khối E lân cận có ứng viên hợp lệ cho MV1 hay không (920). Nếu có, thì bộ mã hóa

dữ liệu video 20 và bộ giải mã dữ liệu video 30 thiết lập ứng viên cho MV1 làm vector chuyển động của khối E lân cận (918). Nếu khối E lân cận không có ứng viên hợp lệ, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đánh dấu MV1 không có ứng viên có sẵn (922).

Quy trình tiếp tục trên Fig.9B, trong đó các khối lân cận F và G trên Fig.6 được kiểm tra. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 trước tiên xác định xem khối F lân cận có ứng viên hợp lệ cho MV2 hay không (924). Nếu có, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 thiết lập ứng viên cho MV2 làm vector chuyển động của khối F lân cận (926). Nếu không, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 xác định xem khối G lân cận có ứng viên hợp lệ cho MV2 hay không (928). Nếu có, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 thiết lập ứng viên cho MV2 làm vector chuyển động của khối G lân cận (926). Nếu khối G lân cận không có ứng viên hợp lệ, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể đánh dấu MV2 không có ứng viên có sẵn (930).

Tiếp theo, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể xác định xem ứng viên cho MV0 đã được xác định trước đó là có sẵn hay không (932). Nếu không, bộ mã hóa dữ liệu video 20 có thể nối thêm các ứng viên MVP bổ sung (ví dụ, các ứng viên nhân tạo được xác định bởi HEVC bổ sung) vào danh sách ứng viên MVP afin (934).

Nếu bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 xác định rằng có sẵn một ứng viên cho MV0 (932), xác định rằng có sẵn một ứng viên cho MV1 (936) và xác định rằng có sẵn một ứng viên cho MV2 (938), thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 sau đó có thể nối thêm các ứng viên là (MVP0, MVP1, MVP2) vào danh sách ứng viên MVP afin (940).

Nếu bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 xác định rằng các ứng viên chỉ có sẵn cho MV0 và MV1, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể suy ra MV2 từ giá trị của các ứng viên cho MV0 và MV1 (942). Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 sau đó có thể nối thêm các ứng viên là (MVP0, MVP1, MVP2) vào danh sách ứng viên MVP afin (940).

Nếu bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 xác định rằng các ứng viên chỉ có sẵn cho MV0 và MV2, thì bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể thu được MV1 từ giá trị của các ứng viên cho MV0 và MV2 (946). Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 sau đó có thể nối thêm các ứng viên là (MVP0, MVP1, MVP2) vào danh sách ứng viên MVP afin (940).

Sau khi nối thì ba ứng viên (MV0, MV1, MV2) là (MVP0, MVP1, MVP2) trong danh sách ứng viên MVP afin (940), bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể kiểm tra lại xem danh sách ứng viên MVP có đầy không (948).

Nếu có, thì kết thúc quy trình. Nếu không, thì bộ mã hóa dữ liệu video 20 có thể nối thêm các ứng viên MVP bổ sung (ví dụ, các ứng viên nhân tạo được xác định bởi HEVC bổ sung) vào danh sách ứng viên MVP afin (934).

Các kỹ thuật được mô tả ở trên để thu được MVP afin dưới dạng (MVP0, MVP1, MVP2) có thể cải thiện hiệu quả mã hóa các CPMV. Theo các ví dụ của sáng chế này, MVP_i từ khối lân cận có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu dữ liệu video hiện thời. Kỹ thuật như vậy có thể giảm hoặc loại bỏ các thao tác định tỷ lệ trong quy trình thu được MVP afin, quy trình này đơn giản hóa quy trình suy ra MVP. Việc thử nghiệm đã chỉ ra rằng các kỹ thuật của sáng chế này dẫn đến việc đơn giản hóa quy trình suy ra MVP với hiệu quả tối thiểu đối với hiệu suất nén.

Theo ví dụ thứ mười ba, bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để xuất ra vector chuyển động của khối lân cận được chọn có liên quan đến RefX.

Theo ví dụ thứ mười bốn, nếu một khối liên hình áp dụng mô hình afin 4 thông số, thì bộ mã hóa dữ liệu video 20 và/hoặc bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để không báo hiệu MVD1y, giá trị này là thành phần dọc của giá trị chênh lệch vector chuyển động của điểm điều khiển trên cùng bên phải.

Theo một ví dụ khác, phần tử cú pháp zoom_only_flag được báo hiệu để biểu thị liệu MVD1y có được báo hiệu hay không. Phần tử cú pháp zoom_only_flag có thể được báo hiệu ở mức chuỗi, chẳng hạn như trong tập hợp thông số tuần tự (Sequence Parameter Set - SPS), ở mức hình, như trong tập hợp thông số hình (Picture Parameter Set - PPS), ở

mức lát, như trong phần đầu lát, trong mức đơn vị mã hóa (Coding Tree Unit - CTU), mức đơn vị mã hóa (Coding Unit - CU) hoặc mức đơn vị dự đoán (Prediction Unit - PU).

1) Cờ `zoom_only` có thể được báo hiệu cho mỗi danh sách tham chiếu trong dự đoán hai chiều một cách riêng rẽ. Chuyển động tham chiếu đến các danh sách tham chiếu khác nhau trong một khối có thể có các cờ `zoom_only_flag` khác nhau.

2) Theo một ví dụ khác, phần tử cú pháp `zoom_only_flag` có thể được báo hiệu một lần cho cả hai danh sách tham chiếu trong dự đoán hai chiều. Các chuyển động tham chiếu đến các danh sách tham chiếu khác nhau trong một khối cùng dùng chung một phần tử cú pháp `zoom_only_flag`.

Theo ví dụ thứ mười lăm của sáng chế, nếu `MVD1y` không được báo hiệu, thì `MV1y` được thiết lập ẩn bằng với `MV0y`. `MV1y` là thành phần dọc của vector chuyển động của điểm điều khiển trên cùng bên phải và `MV0y` là thành phần dọc của vector chuyển động của điểm điều khiển trên cùng bên trái.

Fig.10 là sơ đồ khối minh họa một ví dụ về bộ mã hóa dữ liệu video 20 có thể thực hiện các kỹ thuật của sáng chế. Fig.10 được đưa ra nhằm mục đích giải thích và không được coi là làm giới hạn các kỹ thuật như được minh họa và mô tả rộng rãi trong sáng chế. Các kỹ thuật của sáng chế có thể áp dụng được cho các chuẩn hoặc phương pháp mã hóa khác nhau.

Theo ví dụ trên Fig.10, bộ mã hóa dữ liệu video 20 bao gồm đơn vị xử lý dự đoán 1200, bộ nhớ dữ liệu video 1201, đơn vị tạo dữ liệu dư 1202, đơn vị xử lý biến đổi 1204, đơn vị lượng tử hóa 1206, đơn vị lượng tử hóa ngược 1208, đơn vị xử lý biến đổi ngược 1210, đơn vị tái tạo 1212, đơn vị lọc 1214, bộ đệm hình giải mã 1216, và đơn vị mã hóa entropy 1218. Đơn vị xử lý dự đoán 1200 bao gồm đơn vị xử lý dự đoán liên hình 1220 và đơn vị xử lý dự đoán nội hình 1222. Đơn vị xử lý dự đoán liên hình 1220 có thể bao gồm đơn vị ước lượng chuyển động và đơn vị bù chuyển động (không thể hiện).

Bộ nhớ dữ liệu video 1201 có thể được tạo cấu hình để lưu trữ dữ liệu video cần mã hóa bởi các thành phần của bộ mã hóa dữ liệu video 20. Dữ liệu video lưu trong bộ nhớ dữ liệu video 1201 có thể được thu nhận, ví dụ, từ nguồn dữ liệu video 18. Bộ đệm hình giải mã 1216 có thể là bộ nhớ hình tham chiếu lưu trữ dữ liệu video tham chiếu dùng để mã hóa dữ liệu video bằng bộ mã hóa dữ liệu video 20, ví dụ, ở chế độ mã hóa nội hình

hoặc mã hóa liên hình. Bộ nhớ dữ liệu video 1201 và bộ đệm hình giải mã 1216 có thể được tạo ra bởi bất kỳ trong số nhiều thiết bị nhớ, chẳng hạn như bộ nhớ truy cập ngẫu nhiên động (dynamic random-access memory - DRAM), bao gồm DRAM đồng bộ (synchronous DRAM - SDRAM), RAM từ tính (magnetoresistive RAM - MRAM), RAM điện trở (resistive RAM - RRAM) hoặc các loại thiết bị nhớ khác. Bộ nhớ dữ liệu video 1201 và bộ đệm hình giải mã 1216 có thể được cung cấp bởi cùng một thiết bị nhớ hoặc các thiết bị nhớ riêng. Trong một số ví dụ khác nhau, bộ nhớ dữ liệu video 1201 có thể là bộ nhớ trên chip có các thành phần khác của bộ mã hóa dữ liệu video 20 hoặc bộ nhớ ngoài chip so với các thành phần đó. Bộ nhớ dữ liệu video 1201 có thể giống với hoặc là một phần của phương tiện lưu trữ 19 trên Fig.1.

Bộ mã hóa dữ liệu video 20 nhận dữ liệu video. Bộ mã hóa dữ liệu video 20 có thể mã hóa mỗi CTU trong một lát của hình của dữ liệu video. Mỗi trong số các CTU có thể được kết hợp với các CTB độ chói kích thước bằng nhau và các CTB tương ứng của hình. Như một phần của quá trình mã hóa CTU, đơn vị xử lý dự đoán 1200 có thể thực hiện kỹ thuật phân chia để chia các CTB của CTU thành các khối nhỏ dần. Các khối nhỏ hơn có thể là các khối mã hóa của các CU. Ví dụ, đơn vị xử lý dự đoán 1200 có thể chia CTB gắn với CTU theo cấu trúc cây.

Bộ mã hóa dữ liệu video 20 có thể mã hóa các CU của CTU để tạo ra các dạng biểu diễn đã mã hóa của các CU (tức là, các CU mã hóa). Như một phần của quá trình mã hóa CU, đơn vị xử lý dự đoán 1200 có thể phân chia các khối mã hóa gắn với CU trong số một hoặc nhiều PU của CU. Do đó, mỗi PU có thể được kết hợp với khối dự đoán độ chói và các khối dự đoán màu tương ứng. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể hỗ trợ các PU có các kích thước khác nhau. Như được mô tả ở trên, kích thước của CU có thể được dùng để chỉ kích thước khối mã hóa độ chói của CU và kích thước của PU có thể được dùng để chỉ kích thước của khối dự đoán độ chói của PU. Giả sử kích thước của một CU cụ thể là $2N \times 2N$, bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 có thể hỗ trợ các kích thước PU $2N \times 2N$ hoặc $N \times N$ đối với dự đoán nội hình, và các kích thước PU đối xứng $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$, hoặc tương tự đối với dự đoán liên hình. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 cũng có thể hỗ trợ phân chia không đối xứng các kích thước PU $2N \times nU$, $2N \times nD$, $nL \times 2N$, và $nR \times 2N$ để dự đoán liên hình.

Đơn vị xử lý dự đoán liên hình 1220 có thể tạo ra dữ liệu dự đoán cho PU. Để tạo ra dữ liệu dự đoán cho PU, đơn vị xử lý dự đoán liên hình 1220 thực hiện dự đoán liên hình trên PU. Dữ liệu dự đoán cho PU có thể bao gồm các khối dự đoán của PU và thông tin chuyển động cho PU. Đơn vị xử lý dự đoán liên hình 1220 có thể thực hiện các thao tác khác nhau cho PU của CU tùy thuộc vào việc PU nằm trong lát I, lát P hay lát B. Trong lát I, tất cả PU đều được dự đoán nội hình. Do đó, nếu PU ở trong lát I, thì đơn vị xử lý dự đoán liên hình 1220 không thực hiện dự đoán liên hình trên PU. Do đó, đối với các khối được mã hóa ở chế độ I, khối dự đoán được tạo ra bằng cách sử dụng dự đoán không gian từ các khối lân cận được mã hóa trước đó trong cùng một khung. Nếu PU ở trong lát P, thì đơn vị xử lý dự đoán liên hình 1220 có thể sử dụng kỹ thuật dự đoán liên hình một chiều để tạo ra khối dự đoán cho PU. Nếu PU ở trong lát B, thì đơn vị xử lý dự đoán nội hình 1220 có thể sử dụng kỹ thuật dự đoán liên hình một chiều hoặc hai chiều để tạo ra khối dự đoán cho PU. Đơn vị xử lý dự đoán liên hình 1220 có thể áp dụng các kỹ thuật này cho các mô hình chuyển động afin được mô tả trong các phần khác của bản mô tả này.

Đơn vị xử lý dự đoán nội hình 1222 có thể tạo ra dữ liệu dự đoán cho PU bằng cách thực hiện dự đoán nội hình trên PU. Dữ liệu dự đoán cho PU có thể bao gồm các khối dự đoán của PU và các phần tử cú pháp khác nhau. Đơn vị xử lý dự đoán nội hình 1222 có thể thực hiện dự đoán nội hình trên các PU trong các lát I, lát P và lát B.

Để thực hiện dự đoán nội hình trên PU, đơn vị xử lý dự đoán nội hình 1222 có thể sử dụng nhiều chế độ dự đoán nội hình để tạo ra nhiều tập hợp dữ liệu dự đoán cho PU. Đơn vị xử lý dự đoán nội hình 1222 có thể sử dụng các mẫu từ các khối mẫu của các PU lân cận để tạo ra khối dự đoán cho PU. Các PU lân cận có thể ở trên, bên trên và bên phải, bên trên và bên trái hoặc bên trái của PU, giả sử thứ tự mã hóa từ trái sang phải, từ trên xuống dưới cho PU, CU và CTU. Đơn vị xử lý dự đoán nội hình 1222 có thể sử dụng một số chế độ dự đoán nội hình, ví dụ, 33 chế độ dự đoán nội hình có hướng. Trong một số ví dụ, số lượng chế độ dự đoán nội hình có thể phụ thuộc vào kích thước của vùng liên quan tới PU đó.

Đơn vị xử lý dự đoán 1200 có thể chọn dữ liệu dự đoán cho các PU của CU từ trong số dữ liệu dự đoán được tạo ra bởi đơn vị xử lý dự đoán liên hình 1220 cho các PU hoặc dữ liệu dự đoán được tạo ra bởi đơn vị xử lý dự đoán nội hình 1222 cho các PU. Trong một số ví dụ, đơn vị xử lý dự đoán 1200 chọn dữ liệu dự đoán cho các PU của CU dựa vào

số đo tốc độ/độ méo của các tập hợp dữ liệu dự đoán. Các khối dự đoán của dữ liệu dự đoán được chọn ở đây có thể được gọi là các khối dự đoán được chọn.

Dựa trên các khối mã hóa (ví dụ, các khối mã hóa độ sáng, Cb và Cr) của CU và các khối dự đoán đã chọn (ví dụ, các khối dự đoán độ sáng, Cb và Cr) của PU trong CU, bộ tạo ra dữ liệu dư 1202 có thể tạo ra các khối dữ liệu dư (ví dụ, các khối dữ liệu dư độ sáng, Cb và Cr) cho CU. Ví dụ, bộ tạo ra dữ liệu tạo ra dữ liệu dư 1202 có thể tạo ra các khối dữ liệu dư của CU sao cho mỗi mẫu trong các khối dữ liệu dư này có giá trị bằng hiệu số giữa mẫu trong khối mã hóa của CU và mẫu tương ứng trong khối dự đoán được chọn tương ứng của PU trong CU.

Đơn vị xử lý biến đổi 1204 có thể thực hiện phân chia các khối dữ liệu dư của CU thành các khối biến đổi của các TU của CU. Ví dụ, đơn vị xử lý biến đổi 1204 có thể thực hiện phân chia cây tứ phân để phân chia các khối dữ liệu dư của CU thành các khối biến đổi của các TU của CU. Do đó, TU có thể được kết hợp với một khối biến đổi độ chói và hai khối biến đổi màu. Kích thước và vị trí của các khối biến đổi độ chói và màu của các TU trong CU có thể dựa vào hoặc không dựa vào kích thước và vị trí của các khối dự đoán của PU trong CU. Cấu trúc cây tứ phân được biết đến là “cây tứ phân dư” (residual quad-tree - RQT) có thể bao gồm các nút kết hợp với mỗi trong số các vùng. Các TU của CU có thể tương ứng với các nút lá của RQT.

Đơn vị xử lý biến đổi 1204 có thể tạo ra các khối hệ số biến đổi cho mỗi TU của CU bằng cách áp dụng một hoặc nhiều giá trị biến đổi cho các khối biến đổi của TU. Đơn vị xử lý biến đổi 1204 có thể áp dụng một số giá trị biến đổi cho một khối biến đổi kết hợp với một TU. Ví dụ, đơn vị xử lý biến đổi 1204 có thể áp dụng kỹ thuật biến đổi cosin rời rạc (discrete cosine transform - DCT), biến đổi có hướng hoặc biến đổi có khái niệm tương tự cho khối biến đổi. Trong một số ví dụ, đơn vị xử lý biến đổi 1204 không áp dụng phép biến đổi cho khối biến đổi. Trong các ví dụ như vậy, khối biến đổi có thể được đối xử như khối hệ số biến đổi.

Bộ lượng tử hóa 1206 có thể lượng tử hóa các hệ số biến đổi trong khối hệ số. Quy trình lượng tử hóa có thể làm giảm độ sâu bit liên quan đến một số hoặc tất cả các hệ số biến đổi. Ví dụ, hệ số biến đổi n bit có thể được làm tròn xuống hệ số biến đổi m -bit trong quy trình lượng tử hóa, trong đó n lớn hơn m . Đơn vị lượng tử hóa 1206 có thể lượng tử hóa khối hệ số gắn với TU của CU dựa trên giá trị thông số lượng tử hóa (quantization

parameter - QP) gắn với CU. Bộ mã hóa dữ liệu video 20 có thể điều chỉnh mức độ lượng tử hóa áp dụng cho các khối hệ số kết hợp với CU bằng cách điều chỉnh giá trị QP liên quan đến CU. Quy trình lượng tử hóa có thể làm mất thông tin. Do đó, các hệ số biến đổi được lượng tử hóa có thể có độ chính xác thấp hơn so với các hệ số biến đổi ban đầu.

Đơn vị lượng tử hóa ngược 1208 và đơn vị xử lý biến đổi ngược 1210 có thể áp dụng kỹ thuật lượng tử hóa ngược và biến đổi ngược cho khối hệ số, tương ứng, để tái tạo một khối dư từ khối hệ số. Đơn vị tái tạo 1212 có thể bổ sung khối dữ liệu dư đã tái tạo cho các mẫu tương ứng từ một hoặc nhiều khối dự đoán tạo ra bởi đơn vị xử lý dự đoán 1200 để tạo ra khối biến đổi đã tái tạo gắn với TU. Bằng cách tái tạo các khối biến đổi cho từng TU của CU theo cách này, bộ mã hóa dữ liệu video 20 có thể tái tạo các khối mã hóa của CU này.

Đơn vị lọc 1214 có thể thực hiện một hoặc nhiều hoạt động tách khối để giảm các thành phần lạ dạng khối trong các khối mã hóa liên quan đến CU. Bộ đệm hình giải mã 1216 có thể lưu trữ các khối mã hóa đã tái tạo sau khi đơn vị lọc 1214 thực hiện một hoặc nhiều hoạt động tách khối trên các khối mã hóa đã tái tạo. Đơn vị xử lý dự đoán liên hình 1220 có thể sử dụng hình tham chiếu chứa các khối mã hóa tái tạo để thực hiện dự đoán liên hình trên PU của các hình khác. Ngoài ra, đơn vị xử lý dự đoán nội hình 1222 có thể sử dụng các khối mã hóa tái tạo trong bộ đệm hình giải mã 1216 để thực hiện dự đoán cấu trúc trên các PU khác trong cùng một hình dưới dạng CU.

Đơn vị mã hóa entropy 1218 có thể nhận dữ liệu từ các thành phần chức năng khác của bộ mã hóa dữ liệu video 20. Ví dụ, đơn vị mã hóa entropy 1218 có thể nhận các khối hệ số từ đơn vị lượng tử hóa 1206 và có thể nhận các phần tử cú pháp từ đơn vị xử lý dự đoán 1200. Đơn vị mã hóa entropy 1218 có thể thực hiện một hoặc nhiều hoạt động mã hóa entropy trên dữ liệu để tạo dữ liệu mã hóa entropy. Ví dụ, đơn vị mã hóa entropy 1218 có thể thực hiện hoạt động mã hóa CABAC, hoạt động mã hóa độ dài biến đổi theo ngữ cảnh (context-adaptive variable length coding - CAVLC), hoạt động mã hóa độ dài biến đổi (variable-to-variable - V2V), mã hóa số học nhị phân thích ứng theo ngữ cảnh (syntax-based context-adaptive binary arithmetic coding - SBAC)), hoạt động mã hóa entropy phân chia khoảng xác suất (Probability Interval Partitioning Entropy - PIPE), hoạt động mã hóa hàm mũ Golomb hoặc loại hoạt động mã hóa entropy khác trên dữ liệu. Bộ mã hóa dữ liệu video 20 có thể xuất ra luồng bit bao gồm dữ liệu được mã hóa entropy tạo ra bởi đơn vị

mã hóa entropy 1218. Ví dụ, luồng bit có thể chứa dữ liệu biểu diễn giá trị của các hệ số biến đổi cho CU.

Fig.11 là sơ đồ khối minh họa bộ giải mã dữ liệu video 30 làm ví dụ mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Fig.11 được cung cấp cho mục đích giải thích và không hạn chế trên các kỹ thuật như được mô tả rộng rãi và được mô tả trong sáng chế này. Nhằm mục đích giải thích, sáng chế mô tả bộ giải mã dữ liệu video 30 theo ngữ cảnh mã hóa HEVC. Tuy nhiên, các kỹ thuật trong bản mô tả này có thể áp dụng được cho các phương pháp hoặc chuẩn mã hóa khác.

Theo ví dụ trên Fig.11, bộ giải mã dữ liệu video 30 bao gồm bộ giải mã entropy 1300, bộ nhớ dữ liệu video 1301, bộ xử lý dự đoán 1302, bộ lượng tử hóa ngược 1304, bộ xử lý biến đổi ngược 1306, bộ tái tạo 1308, bộ lọc 1310, và bộ đệm hình giải mã 1312. Bộ xử lý dự đoán 1302 bao gồm đơn vị bù chuyển động 1314 và đơn vị xử lý dự đoán nội hình 1316. Theo các ví dụ khác, bộ giải mã dữ liệu video 30 có thể bao gồm nhiều hơn, ít hơn hoặc các thành phần chức năng khác nhau.

Bộ nhớ dữ liệu video 1301 có thể lưu trữ dữ liệu video đã mã hóa, ví dụ như luồng bit video đã mã hóa, mà cần được giải mã bởi các thành phần của bộ giải mã dữ liệu video 30. Dữ liệu video được lưu trữ trong bộ nhớ dữ liệu video 1301 có thể được thu, ví dụ, từ phương tiện đọc được bằng máy tính 16, ví dụ, từ nguồn video cục bộ, như máy quay, qua truyền dữ liệu video trên mạng có dây hoặc không dây, hoặc bằng cách truy cập phương tiện lưu trữ dữ liệu vật lý. Bộ nhớ dữ liệu video 1301 có thể tạo ra bộ đệm hình mã hóa (coded picture buffer - CPB) lưu trữ dữ liệu video mã hóa từ luồng bit video mã hóa. Bộ đệm hình giải mã 1312 có thể là bộ nhớ hình tham chiếu mà lưu trữ dữ liệu video tham chiếu dùng để giải mã dữ liệu video bằng bộ giải mã dữ liệu video bộ giải mã dữ liệu video 30, ví dụ, các chế độ mã hóa liên hình hoặc nội hình, hoặc để xuất ra. Bộ nhớ dữ liệu video 1301 và bộ đệm hình giải mã 1312 có thể được tạo ra bởi bất kỳ trong số nhiều thiết bị nhớ, chẳng hạn như bộ nhớ truy cập ngẫu nhiên động (dynamic random-access memory - DRAM), bao gồm DRAM đồng bộ (synchronous DRAM - SDRAM), RAM từ tính (magnetoresistive RAM - MRAM), RAM điện trở (resistive RAM - RRAM) hoặc các loại thiết bị nhớ khác. Bộ nhớ dữ liệu video 1301 và bộ đệm hình giải mã 1312 có thể được cung cấp bởi cùng một thiết bị nhớ hoặc các thiết bị nhớ riêng. Theo một số ví dụ, bộ nhớ dữ liệu video 1301 có thể là bộ nhớ trên chip có các thành phần khác của bộ giải mã dữ

liệu video 30 hoặc bộ nhớ ngoài chip liên quan đến các thành phần này. Bộ nhớ dữ liệu video 1301 có thể giống với hoặc là một phần của phương tiện lưu trữ 28 trên Fig.1.

Bộ nhớ dữ liệu video 1301 nhận và lưu trữ dữ liệu video đã mã hóa (ví dụ, các đơn vị NAL) từ luồng bit. Bộ giải mã entropy 1300 có thể nhận dữ liệu video đã mã hóa (ví dụ, đơn vị NAL) từ bộ nhớ dữ liệu video 1301 và có thể phân tích các đơn vị NAL để thu được các phần tử cú pháp. Đơn vị giải mã entropy 1300 có thể giải mã entropy các phần tử cú pháp được mã hóa entropy trong các đơn vị NAL. Đơn vị xử lý dự đoán 1302, đơn vị lượng tử hóa ngược 1304, đơn vị xử lý biến đổi ngược 1306, đơn vị tái tạo 1308, và đơn vị lọc 1310 có thể tạo ra dữ liệu video đã giải mã dựa vào các phần tử cú pháp trích ra từ luồng bit. Đơn vị giải mã entropy 1300 có thể thực hiện quy trình thường ngược với quy trình của đơn vị mã hóa entropy 1218.

Ngoài việc thu nhận các phần tử cú pháp từ luồng bit, bộ giải mã dữ liệu video 30 có thể thực hiện hoạt động tái tạo trên CU. Để thực hiện hoạt động tái tạo này, bộ giải mã dữ liệu video 30 có thể thực hiện hoạt động tái tạo trên từng TU của CU. Bằng cách thực hiện hoạt động tái tạo cho từng TU của CU, bộ giải mã dữ liệu video 30 có thể tái tạo các khối dữ liệu dư của CU.

Như một phần của quá trình thực hiện hoạt động tái tạo trên TU của CU, đơn vị lượng tử hóa ngược 1304 có thể lượng tử hóa ngược, tức là, khử lượng tử hóa, các khối hệ số liên quan đến TU. Sau khi đơn vị lượng tử hóa ngược 1304 lượng tử hóa ngược khối hệ số, đơn vị xử lý biến đổi ngược 1306 có thể áp dụng một hoặc nhiều phần biến đổi ngược cho khối hệ số để tạo ra khối dữ liệu dư liên quan đến TU. Ví dụ, đơn vị xử lý biến đổi ngược 1306 có thể áp dụng phép DCT ngược, biến đổi số nguyên ngược, biến đổi Karhunen-Loeve (Karhunen-Loeve transform - KLT) ngược, biến đổi quay ngược, biến đổi hướng ngược hoặc biến đổi ngược khác cho khối hệ số.

Đơn vị lượng tử hóa ngược 1304 có thể thực hiện các kỹ thuật cụ thể theo sáng chế. Ví dụ, đối với ít nhất một nhóm lượng tử hóa tương ứng trong số nhiều nhóm lượng tử hóa trong CTB của CTU trong hình của dữ liệu video, đơn vị lượng tử hóa ngược 1304 có thể suy ra, dựa ít nhất một phần vào thông tin lượng tử hóa cục bộ được báo hiệu trong luồng bit, thông số lượng tử hóa tương ứng cho nhóm lượng tử hóa tương ứng. Ngoài ra, trong ví dụ này, đơn vị lượng tử hóa ngược 1304 có thể lượng tử hóa ngược, dựa vào thông số lượng tử hóa tương ứng cho nhóm lượng tử hóa tương ứng, ít nhất một hệ số biến đổi của

khối biến đổi của TU của CU của CTU. Trong ví dụ này, nhóm lượng tử hóa tương ứng được định nghĩa là nhóm gồm các CU hoặc khối mã hóa liên tiếp theo thứ tự mã hóa sao cho các biên của nhóm lượng tử hóa tương ứng phải là các biên của các CU hoặc khối mã hóa và kích thước của nhóm lượng tử hóa tương ứng lớn hơn hoặc bằng ngưỡng. Bộ giải mã dữ liệu video 30 (ví dụ, đơn vị xử lý biến đổi ngược 1306, đơn vị tái tạo 1308, và đơn vị lọc 1310) có thể tái tạo, dựa vào các hệ số biến đổi được lượng tử hóa ngược của khối biến đổi, khối mã hóa của CU.

Nếu PU được mã hóa bằng cách sử dụng kỹ thuật dự đoán nội hình, thì đơn vị xử lý dự đoán nội hình 1316 có thể thực hiện dự đoán nội hình để tạo ra các khối dự đoán cho PU này. Đơn vị xử lý dự đoán nội hình 1316 có thể sử dụng chế độ dự đoán nội hình để tạo ra các khối dự đoán của PU dựa trên các khối lân cận theo không gian. Đơn vị xử lý dự đoán nội hình 1316 có thể xác định chế độ dự đoán nội hình cho PU dựa trên một hoặc nhiều phân tử cú pháp thu được từ luồng bit.

Nếu PU được mã hóa nhờ sử dụng kỹ thuật dự đoán liên hình, thì đơn vị bù chuyển động 1314 có thể xác định thông tin chuyển động cho PU này. Đơn vị bù chuyển động 1314 có thể xác định, dựa vào thông tin chuyển động của PU, một hoặc nhiều khối tham chiếu. Đơn vị bù chuyển động 1314 có thể tạo ra, dựa trên một hoặc nhiều khối tham chiếu, khối dự đoán (ví dụ, khối dự đoán độ chói, Cb và Cr) cho PU. Đơn vị bù chuyển động 1314 có thể áp dụng các kỹ thuật cho các mô hình chuyển động afin được mô tả trong các phần khác của bản mô tả này.

Đơn vị tái tạo 1308 có thể sử dụng các khối biến đổi (ví dụ, khối biến đổi độ chói, Cb và Cr) cho các đơn vị TU của CU và các khối dự đoán (ví dụ, khối độ chói, Cb và Cr) của các PU của CU, (chẳng hạn, dữ liệu dự đoán nội hình hoặc dữ liệu dự đoán liên hình, nếu có thể, để tái tạo các khối mã hóa (ví dụ, khối mã hóa độ chói, Cb và Cr) cho CU. Ví dụ, đơn vị tái tạo 1308 có thể thêm các mẫu của các khối biến đổi (ví dụ, khối biến đổi độ chói, Cb và Cr) vào các mẫu tương ứng của các khối dự đoán (ví dụ, khối dự đoán độ chói, Cb và Cr) để tái tạo các khối mã hóa (ví dụ, khối mã hóa độ chói, Cb và Cr) của CU.

Đơn vị lọc 1310 có thể thực hiện hoạt động tách khối để giảm các thành phần lạ tạo khối liên quan đến các khối mã hóa của CU. Bộ giải mã dữ liệu video 30 có thể lưu trữ các khối mã hóa của CU vào bộ đệm hình giải mã 1312. Bộ đệm hình giải mã 1312 có thể cung cấp các hình tham chiếu cho hoạt động bù chuyển động tiếp theo, dự đoán nội hình,

và biểu diễn trên thiết bị hiển thị, chẳng hạn như thiết bị hiển thị 32 trên Fig.1. Ví dụ, bộ giải mã dữ liệu video 30 có thể thực hiện, dựa trên các khối trong bộ đệm hình được giải mã 1312, các hoạt động dự đoán nội hình hoặc dự đoán liên hình cho các PU của các CU khác.

Fig.12 là lưu đồ minh họa một ví dụ về phương pháp mã hóa theo sáng chế. Các kỹ thuật trên Fig.12 có thể được thực hiện bởi một hoặc nhiều thành phần cấu trúc của bộ mã hóa dữ liệu video 20, bao gồm đơn vị xử lý dự đoán liên hình 1222.

Theo một ví dụ của sáng chế, bộ mã hóa dữ liệu video 20 có thể được tạo cấu hình để nhận khối dữ liệu video hiện thời cần được mã hóa bằng cách sử dụng kỹ thuật bù chuyển động afin (200), và xây dựng danh sách MVP afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời (202). Bộ mã hóa dữ liệu video 20 có thể được tạo cấu hình thêm để

xác định vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin (204), và mã hóa khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời (206).

Theo một ví dụ, một hoặc nhiều điểm điều khiển bao gồm ít nhất là điểm điều khiển thứ nhất. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ nhất, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để kiểm tra các khối lân cận thứ nhất của điểm điều khiển thứ nhất theo thứ tự định trước thứ nhất, trong đó các khối lân cận thứ nhất bao gồm khối lân cận bên trái bên trên (A), khối lân cận bên trên thứ nhất (B) và khối lân cận bên trái thứ nhất (C), và trong đó thứ tự định trước thứ nhất là A rồi C rồi B, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ nhất (MVP0), vector chuyển động thứ nhất của các khối lân cận thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ nhất.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để đánh dấu bộ dự đoán vector chuyển động thứ nhất cho điểm điều khiển thứ nhất là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ nhất của điểm điều khiển thứ nhất có hình tham chiếu kết hợp giống với hình tham khảo đích.

Theo một ví dụ khác của sáng chế, một hoặc nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ hai. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ hai, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để kiểm tra các khối lân cận thứ hai của điểm điều khiển thứ hai theo thứ tự định trước thứ hai, trong đó các khối lân cận thứ hai bao gồm khối lân cận bên trên thứ hai (D), khối lân cận bên trên bên phải (E), và trong đó thứ tự định trước thứ hai là D rồi E, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ hai (MVP1), vector chuyển động thứ hai của các khối lân cận thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ hai.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để đánh dấu bộ dự đoán vector chuyển động thứ hai cho điểm điều khiển thứ nhất thứ hai là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ hai của điểm điều khiển thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, một hoặc nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ ba. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ ba, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để kiểm tra các khối lân cận thứ ba của điểm điều khiển thứ ba theo thứ tự định trước thứ ba, trong đó các khối lân cận thứ ba bao gồm khối lân cận bên trái thứ hai (F), và khối lân cận bên dưới bên trái (G), và trong đó thứ tự định trước thứ ba là F rồi G, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ ba (MVP2), vector chuyển động thứ ba của các khối lân cận thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ ba.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để đánh dấu bộ dự đoán vector chuyển động thứ ba cho điểm điều khiển thứ nhất thứ ba là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ ba của điểm điều khiển thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0), và bộ dự đoán vector chuyển động thứ hai (MVP1) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ nhất (MVP0) dưới dạng hàm của bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ mã hóa dữ liệu video 20 được tạo cấu hình thêm để làm tròn bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) theo độ chính xác chuyển động xác định trước.

Theo một ví dụ khác của sáng chế, kỹ thuật bù chuyển động affin là mô hình affin 6 thông số, và danh sách MVP affin bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng một ứng viên MVP duy nhất.

Theo một ví dụ khác của sáng chế, kỹ thuật bù chuyển động affin là mô hình affin 4 thông số và danh sách MVP affin bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), và bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng một ứng viên MVP duy nhất.

Theo một ví dụ khác của sáng chế, hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp nằm trong cùng danh sách hình tham chiếu là hình tham chiếu đích và có cùng chỉ số tham chiếu như hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp có cùng số đếm thứ tự hình với hình tham chiếu đích.

Fig.13 là lưu đồ minh họa một ví dụ về phương pháp giải mã theo sáng chế. Các kỹ thuật trên Fig.13 có thể được thực hiện bởi một hoặc nhiều thành phần cấu trúc của bộ giải mã dữ liệu video 30, bao gồm đơn vị bù chuyển động 1314.

Theo một ví dụ của sáng chế, bộ giải mã dữ liệu video 30 có thể được tạo cấu hình để nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng bù chuyển động afin (210), và xây dựng danh sách MVP afin cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ một khối dữ liệu video lân cận vào danh sách MVP afin trong trường hợp vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời (212). Bộ giải mã dữ liệu video 30 có thể còn được tạo cấu hình để xác định vector chuyển động cho một hoặc nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin (214), và giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định được cho một hoặc nhiều điểm điều khiển của khối dữ liệu video hiện thời (216).

Theo một ví dụ, một hoặc nhiều điểm điều khiển bao gồm ít nhất là điểm điều khiển thứ nhất. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ nhất, bộ giải mã dữ liệu video 30 được tạo cấu hình thêm để kiểm tra các khối lân cận thứ nhất của điểm điều khiển thứ nhất theo thứ tự định trước thứ nhất, trong đó các khối lân cận thứ nhất bao gồm khối lân cận bên trên bên trái (A), khối lân cận bên trên thứ nhất (B) và khối lân cận thứ nhất bên trái (C), và trong đó thứ tự định trước thứ nhất là A rồi C rồi B, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ nhất (MVP0), vector chuyển động thứ nhất của các khối lân cận thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ nhất.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 được tạo cấu hình thêm để đánh dấu bộ dự đoán vector chuyển động thứ nhất cho điểm điều khiển thứ nhất là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ nhất của điểm điều khiển thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, một hoặc nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ hai. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ hai, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để kiểm tra các khối lân cận thứ hai của điểm điều khiển thứ hai theo thứ tự định trước thứ hai, trong đó các khối lân cận thứ hai bao gồm khối lân cận bên trên thứ hai (D), và khối lân cận bên trên bên phải (E), và trong

đó thứ tự định trước thứ hai là D rồi E, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ hai (MVP1), vector chuyển động thứ hai của các khối lân cận thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ hai.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để đánh dấu bộ dự đoán vector chuyển động thứ hai cho điểm điều khiển thứ hai là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ hai của điểm điều khiển thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, một hoặc nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ ba. Để xây dựng danh sách MVP afin cho điểm điều khiển thứ ba, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để kiểm tra các khối lân cận thứ ba của điểm điều khiển thứ ba theo thứ tự định trước thứ ba, trong đó các khối lân cận thứ ba bao gồm khối lân cận bên trái thứ hai (F), và khối lân cận bên dưới bên trái (G), và trong đó thứ tự định trước thứ ba là F rồi G, và thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ ba (MVP2), vector chuyển động thứ ba của các khối lân cận thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ ba.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để đánh dấu bộ dự đoán vector chuyển động thứ ba cho điểm điều khiển thứ nhất thứ ba là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ ba của điểm điều khiển thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng hàm của bộ dự đoán

vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn, và tính toán bộ dự đoán vector chuyển động thứ nhất (MVP0) dưới dạng hàm của bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn.

Theo một ví dụ khác của sáng chế, bộ giải mã dữ liệu video 30 còn được tạo cấu hình để làm tròn bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) theo độ chính xác chuyển động xác định trước.

Theo một ví dụ khác của sáng chế, kỹ thuật bù chuyển động afin là mô hình afin 6 thông số, và danh sách MVP afin bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng một ứng viên MVP duy nhất.

Theo một ví dụ khác của sáng chế, kỹ thuật bù chuyển động afin là mô hình afin 4 thông số và danh sách MVP afin bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), và bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng một ứng viên MVP duy nhất.

Theo một ví dụ khác của sáng chế, hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp nằm trong cùng danh sách hình tham chiếu là hình tham chiếu đích và có cùng chỉ số tham chiếu như hình tham chiếu đích.

Theo một ví dụ khác của sáng chế, hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp có cùng số đếm thứ tự hình với hình tham chiếu đích.

Một số khía cạnh của sáng chế này đã được mô tả liên quan đến HEVC, các phần mở rộng của chuẩn HEVC, JEM và VVC cho mục đích minh họa. Tuy nhiên, các kỹ thuật được mô tả trong sáng chế này có thể hữu ích cho các quy trình mã hóa dữ liệu video khác, bao gồm các quy trình mã hóa dữ liệu video chuẩn hoặc độc quyền khác hiện đang phát triển hoặc vẫn chưa được phát triển.

Bộ mã hóa dữ liệu video, như được mô tả trong sáng chế, có thể được dùng để chỉ bộ mã hóa dữ liệu video hoặc bộ giải mã dữ liệu video. Tương tự, đơn vị mã hóa dữ liệu video có thể dùng để chỉ bộ mã hóa dữ liệu video hoặc bộ giải mã dữ liệu video. Tương tự, mã hóa dữ liệu video có thể chỉ mã hóa dữ liệu video hoặc giải mã dữ liệu video, tùy trường hợp. Theo sáng chế, cụm từ “dựa vào” có thể biểu thị chỉ dựa vào, dựa ít nhất một phần vào, hoặc dựa theo một cách nào đó vào. Sáng chế có thể sử dụng thuật ngữ “đơn vị dữ liệu video” hoặc “khối dữ liệu video” hoặc “khối” để chỉ một hoặc nhiều khối mẫu và các cấu trúc cú pháp được dùng để mã hóa các mẫu của một hoặc nhiều khối mẫu. Các dạng đơn vị dữ liệu video làm ví dụ có thể bao gồm CTU, CU, PU, các đơn vị biến đổi (transform unit - TU), các khối macrô, các phần chia khối macrô, v.v. Trong một số trường hợp, phần mô tả các PU có thể được dùng thay thế cho phần mô tả các khối macrô hoặc các phần chia khối macrô. Ví dụ về các loại khối dữ liệu video có thể bao gồm các khối cây mã hóa, các khối mã hóa, và các loại khối dữ liệu video khác.

Cần hiểu rằng tùy thuộc vào ví dụ, các hành động hoặc sự kiện nhất định của kỹ thuật bất kỳ được mô tả ở đây có thể được thực hiện theo một trình tự khác, có thể được bổ sung, hợp nhất hoặc loại bỏ hoàn toàn (ví dụ, không phải tất cả các hành động hoặc sự kiện được mô tả đều cần thiết để thực hiện các kỹ thuật theo sáng chế). Ngoài ra, theo một số phương án làm ví dụ, các thao tác hoặc biến cố có thể được thực hiện đồng thời, ví dụ, thông qua quy trình xử lý đa khâu chuỗi, quy trình xử lý ngắt, hoặc nhiều bộ xử lý, chứ không phải chỉ có thực hiện tuần tự.

Theo một hoặc nhiều phương án làm ví dụ, các hàm đã mô tả có thể được thực hiện dưới dạng phần cứng, phần mềm, phần sụn, hoặc mọi dạng kết hợp của chúng. Nếu được thực hiện bằng phần mềm, các chức năng có thể được lưu trữ trên hoặc truyền qua, dưới dạng một hoặc nhiều lệnh hoặc mã, phương tiện đọc được bằng máy tính và được thực thi bởi bộ xử lý dựa trên phần cứng. Phương tiện đọc được bằng máy tính có thể là phương tiện lưu trữ đọc được bằng máy tính, tương ứng với phương tiện lưu trữ hữu hình chẳng hạn như phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông bao gồm mọi phương tiện tạo điều kiện thuận lợi cho việc truyền chương trình máy tính từ nơi này sang nơi khác, ví dụ theo giao thức truyền thông. Theo cách này, phương tiện đọc được bằng máy tính thường có thể tương ứng với (1) phương tiện lưu trữ đọc được bằng máy tính hữu hình hoặc (2) phương tiện truyền thông như tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ

liệu có thể là mọi phương tiện có sẵn có thể truy cập được bằng một hoặc nhiều máy tính hay một hoặc nhiều mạch xử lý để tìm kiếm các lệnh, mã và/hoặc cấu trúc dữ liệu để thực hiện các kỹ thuật nêu trong sáng chế. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được bằng máy tính.

Ví dụ, và không giới hạn, các phương tiện lưu trữ đọc được bởi máy tính như vậy có thể bao gồm bộ nhớ truy cập ngẫu nhiên (Random Access Memory - RAM), bộ nhớ chỉ đọc (Read Only Memory ROM), bộ nhớ chỉ đọc lập trình được xóa được bằng điện (Electrically Erasable Programmable Read Only Memory - EEPROM), đĩa compac-bộ nhớ chỉ đọc (CD-ROM) hoặc các thiết bị lưu trữ đĩa quang khác, thiết bị lưu trữ đĩa từ, hoặc các thiết bị lưu trữ từ tính khác, bộ nhớ flash, hoặc phương tiện khác bất kỳ mà có thể được dùng để lưu trữ mã chương trình mong muốn ở dạng các lệnh hoặc các cấu trúc dữ liệu và có thể được truy cập bởi máy tính. Ngoài ra, mọi dạng kết nối được gọi theo cách thích hợp là phương tiện đọc được bằng máy tính. Ví dụ, nếu các lệnh được truyền từ trang web, máy chủ hoặc nguồn từ xa khác bằng cách sử dụng cáp đồng trục, cáp quang, cáp dây xoắn, đường thuê bao kỹ thuật số (digital subscriber line - DSL) hoặc các công nghệ không dây như hồng ngoại, vô tuyến, và vi sóng, thì cáp đồng trục, cáp quang, cáp dây xoắn, đường DSL hoặc các công nghệ không dây như hồng ngoại, vô tuyến và vi sóng được bao gồm trong định nghĩa phương tiện. Tuy nhiên, cần phải hiểu rằng phương tiện lưu trữ đọc được bằng máy tính và phương tiện lưu trữ dữ liệu không bao gồm các kết nối, sóng mang, tín hiệu hoặc phương tiện tạm thời khác, mà thay vào đó được hướng đến phương tiện lưu trữ hữu hình, không tạm thời.

Đĩa từ và đĩa quang, như được sử dụng ở đây, bao gồm đĩa nén (Compact Disc - CD), đĩa laze, đĩa quang, đĩa kỹ thuật số đa năng (Digital Versatile Disc - DVD), đĩa mềm và đĩa Blu-ray, trong đó các đĩa từ thường sao chép dữ liệu theo cách từ tính, trong khi đó đĩa quang sao chép dữ liệu theo cách quang học bằng các tia laze. Kết hợp của cả hai loại này cũng được bao gồm trong phạm vi của phương tiện đọc được bằng máy tính.

Chức năng được mô tả trong bản mô tả này có thể được thực hiện bằng chức năng cố định và/hoặc hệ mạch xử lý lập trình được. Ví dụ, các lệnh có thể được thực thi bằng chức năng cố định và/hoặc hệ mạch xử lý lập trình được. Các mạch xử lý này có thể bao gồm một hoặc nhiều bộ xử lý, chẳng hạn như một hoặc nhiều bộ xử lý tín hiệu kỹ thuật số (Digital Signal Processor - DSP), các bộ vi xử lý đa năng, các mạch tích hợp chuyên dụng

(Application Specific Integrated Circuit - ASIC), các mảng logic lập trình được theo trường (field programmable logic array - FPGA), hoặc mạch logic rời rạc hoặc mạch tích hợp tương đương khác. Do vậy, thuật ngữ “bộ xử lý”, như được sử dụng ở đây có thể chỉ cấu trúc bất kỳ trong các cấu trúc nêu trên hoặc cấu trúc khác bất kỳ phù hợp để thực hiện các kỹ thuật được mô tả ở đây. Ngoài ra, trong một số ví dụ, chức năng được mô tả trong sáng chế này có thể được cung cấp trong các môđun phần cứng và/hoặc phần mềm chuyên dụng được định cấu hình để mã hóa và giải mã hoặc kết hợp trong bộ codec kết hợp. Ngoài ra, các kỹ thuật có thể được thực hiện đầy đủ trong một hoặc nhiều mạch hoặc các phần tử logic. Các mạch xử lý có thể được nối với các thành phần khác theo nhiều cách khác nhau. Ví dụ, mạch xử lý có thể được nối với các thành phần khác qua liên kết bằng thiết bị bên trong, kết nối mạng có dây hoặc không dây, hoặc một phương tiện truyền thông khác.

Các kỹ thuật theo sáng chế có thể được thực hiện bằng rất nhiều thiết bị hoặc cơ cấu, như thiết bị cầm tay không dây, mạch tích hợp (Integrated Circuit - IC) hoặc bộ IC (ví dụ, bộ chip). Các thành phần, môđun hoặc bộ phận khác nhau được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật đã bộc lộ, chứ không nhất thiết phải được thực hiện bằng các bộ phận phần cứng khác nhau. Thay vào đó, như được mô tả ở trên, một số đơn vị có thể được kết hợp trong đơn vị phần cứng codec hoặc được cung cấp bởi tập hợp các đơn vị phần cứng tương tác, bao gồm một hoặc nhiều bộ xử lý như được mô tả ở trên, kết hợp với phần mềm và/hoặc phần sụn thích hợp.

Các ví dụ khác nhau đã được mô tả. Những ví dụ này và những ví dụ khác đều nằm trong phạm vi của các điểm yêu cầu bảo hộ sau đây.

YÊU CẦU BẢO HỘ

1. Phương pháp mã hóa dữ liệu video, phương pháp này bao gồm các bước:

nhận khối dữ liệu video hiện thời cần được giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin;

xây dựng danh sách bộ dự đoán vector chuyển động (motion vector predictor - MVP) afin cho nhiều điểm điều khiển của mô hình afin để mã hóa khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, và không thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động không có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời,

trong đó việc xây dựng danh sách MVP afin còn bao gồm thu được một ứng viên từ nhiều bộ dự đoán vector chuyển động, trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) dựa vào mô hình afin là mô hình afin 6 thông số, và trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) dựa vào mô hình afin là mô hình afin 4 thông số;

xác định các vector chuyển động cho nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin; và

giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định cho nhiều điểm điều khiển của khối dữ liệu video hiện thời.

2. Phương pháp theo điểm 1, trong đó nhiều điểm điều khiển bao gồm ít nhất là điểm điều khiển thứ nhất, và trong đó việc xây dựng danh sách MVP afin cho điểm điều khiển thứ nhất bao gồm các bước:

kiểm tra các khối lân cận thứ nhất của điểm điều khiển thứ nhất theo thứ tự định trước thứ nhất, trong đó các khối lân cận thứ nhất bao gồm khối lân cận ở trên bên trái (A), khối lân cận ở trên thứ nhất (B), và khối lân cận bên trái thứ nhất (C), và trong đó thứ tự định trước thứ nhất là A rồi C rồi B; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ nhất (MVP0), vector chuyển động thứ nhất của các khối lân cận thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước.

3. Phương pháp theo điểm 2, phương pháp này còn bao gồm bước:

đánh dấu bộ dự đoán vector chuyển động thứ nhất cho điểm điều khiển thứ nhất là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ nhất của điểm điều khiển thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích.

4. Phương pháp theo điểm 2, trong đó nhiều điểm điều khiển bao gồm thêm điểm điều khiển thứ hai, và trong đó việc xây dựng danh sách MVP afin cho điểm điều khiển thứ hai bao gồm các bước:

kiểm tra các khối lân cận thứ hai của điểm điều khiển thứ hai theo thứ tự định trước thứ hai, trong đó các khối lân cận thứ hai bao gồm khối lân cận bên trên thứ hai (D), và khối lân cận bên trên bên phải (E), và trong đó thứ tự định trước thứ hai là D rồi E; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ hai (MVP1), vector chuyển động thứ hai của các khối lân cận thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ hai.

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

đánh dấu bộ dự đoán vector chuyển động thứ hai cho điểm điều khiển thứ nhất thứ hai là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ hai của điểm điều khiển thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích.

6. Phương pháp theo điểm 4, trong đó nhiều điểm điều khiển bao gồm thêm điểm điều khiển thứ ba, và trong đó việc xây dựng danh sách MVP afin cho điểm điều khiển thứ ba bao gồm các bước:

kiểm tra các khối lân cận thứ ba của điểm điều khiển thứ ba theo thứ tự định trước thứ ba, trong đó các khối lân cận thứ ba bao gồm khối lân cận bên trái thứ hai (F), và khối lân cận bên dưới bên trái (G), và trong đó thứ tự định trước thứ ba là F rồi G; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ ba (MVP2), vector chuyển động thứ ba của các khối lân cận thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ ba.

7. Phương pháp theo điểm 6, phương pháp này còn bao gồm bước:

đánh dấu bộ dự đoán vector chuyển động thứ ba cho điểm điều khiển thứ nhất thứ ba là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ ba của điểm điều khiển thứ ba có hình tham chiếu giống với hình tham chiếu đích.

8. Phương pháp theo điểm 6, phương pháp này còn bao gồm các bước:

xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng một hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn.

9. Phương pháp theo điểm 6, phương pháp này còn bao gồm các bước:

xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn.

10. Phương pháp theo điểm 6, phương pháp này còn bao gồm các bước:

xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ nhất (MVP0) dưới dạng hàm của bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn.

11. Phương pháp theo điểm 6, phương pháp này còn bao gồm bước:

làm tròn bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) theo độ chính xác chuyển động định trước.

12. Phương pháp theo điểm 1, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp nằm trong

cùng danh sách hình tham chiếu với hình tham chiếu đích và có cùng chỉ số tham chiếu với hình tham chiếu đích.

13. Phương pháp theo điểm 1, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp có số đếm thứ tự hình giống với hình tham chiếu đích.

14. Thiết bị được tạo cấu hình để giải mã dữ liệu video, thiết bị này bao gồm:

bộ nhớ được tạo cấu hình để lưu trữ khối dữ liệu video hiện thời; và

một hoặc nhiều bộ xử lý truyền thông với bộ nhớ, trong đó thiết bị được tạo cấu hình để:

nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin;

xây dựng danh sách bộ dự đoán vector chuyển động (motion vector predictor - MVP) afin cho nhiều điểm điều khiển của mô hình afin để mã hóa khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, và không thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động không có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời,

trong đó để xây dựng danh sách MVP afin, thiết bị còn được tạo cấu hình để thu được một ứng viên từ nhiều bộ dự đoán vector chuyển động, trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) dựa vào mô hình afin là mô hình afin 6 thông số, và trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) dựa vào mô hình afin là mô hình afin 4 thông số;

xác định các vector chuyển động cho nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin; và

giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định cho nhiều điểm điều khiển của khối dữ liệu video hiện thời.

15. Thiết bị theo điểm 14, trong đó nhiều điểm điều khiển bao gồm ít nhất là điểm điều khiển thứ nhất, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ nhất, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ nhất của điểm điều khiển thứ nhất theo thứ tự định trước thứ nhất, trong đó các khối lân cận thứ nhất bao gồm khối lân cận ở trên bên trái (A), khối lân cận bên trên thứ nhất (B), và khối lân cận bên trái thứ nhất (C), và trong đó thứ tự định trước thứ nhất là A rồi C rồi B; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ nhất (MVP0), vector chuyển động thứ nhất của các khối lân cận thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước.

16. Thiết bị theo điểm 15, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vector chuyển động thứ nhất cho điểm điều khiển thứ nhất là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ nhất của điểm điều khiển thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích.

17. Thiết bị theo điểm 15, trong đó nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ hai, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ hai, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ hai của điểm điều khiển thứ hai theo thứ tự định trước thứ hai, trong đó các khối lân cận thứ hai bao gồm khối lân cận bên trên thứ hai (D), và khối lân cận bên trên bên phải (E), và trong đó thứ tự định trước thứ hai là D rồi E; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ hai (MVP1), vector chuyển động thứ hai của các khối lân cận thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ hai.

18. Thiết bị theo điểm 17, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vector chuyển động thứ hai cho điểm điều khiển thứ nhất thứ hai là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ hai của điểm điều khiển thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích.

19. Thiết bị theo điểm 17, trong đó nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ ba, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ ba, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ ba của điểm điều khiển thứ ba theo thứ tự định trước thứ ba, trong đó các khối lân cận thứ ba bao gồm khối lân cận bên trái thứ hai (F), và khối lân cận bên dưới bên trái (G), và trong đó thứ tự định trước thứ ba là F rồi G; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ ba (MVP2), vector chuyển động thứ ba của các khối lân cận thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ ba.

20. Thiết bị theo điểm 19, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vector chuyển động thứ ba cho điểm điều khiển thứ nhất thứ ba là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ ba của điểm điều khiển thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích.

21. Thiết bị theo điểm 19, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn.

22. Thiết bị theo điểm 19, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn.

23. Thiết bị theo điểm 19, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ nhất (MVP0) dưới dạng hàm của bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn.

24. Thiết bị theo điểm 19, trong đó thiết bị còn được tạo cấu hình để:

làm tròn bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) theo độ chính xác chuyển động định trước.

25. Thiết bị theo điểm 14, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp nằm trong cùng danh sách hình tham chiếu với hình tham chiếu đích và có cùng chỉ số tham chiếu với hình tham chiếu đích.

26. Thiết bị theo điểm 14, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp có cùng số đếm thứ tự hình với hình tham chiếu đích.

27. Thiết bị được tạo cấu hình để giải mã dữ liệu video, thiết bị này bao gồm:

phương tiện nhận khối dữ liệu video hiện thời cần giải mã bằng cách sử dụng kỹ thuật bù chuyển động afin;

phương tiện xây dựng danh sách bộ dự đoán vector chuyển động (MVP) afin cho nhiều điểm điều khiển của mô hình afin để mã hóa khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, và không thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động không có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời,

trong đó phương tiện xây dựng danh sách MVP afin còn bao gồm phương tiện thu được một ứng viên từ nhiều bộ dự đoán vector chuyển động, trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) dựa

vào mô hình afin là mô hình afin 6 thông số, và trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) dựa vào mô hình afin là mô hình afin 4 thông số;

phương tiện xác định các vector chuyển động cho nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin; và

phương tiện giải mã khối dữ liệu video hiện thời với các vector chuyển động xác định cho nhiều điểm điều khiển của khối dữ liệu video hiện thời.

28. Thiết bị được tạo cấu hình để mã hóa dữ liệu video, thiết bị này bao gồm:

bộ nhớ được tạo cấu hình để lưu trữ khối dữ liệu video hiện thời; và

một hoặc nhiều bộ xử lý truyền thông với bộ nhớ, trong đó thiết bị được tạo cấu hình để:

nhận khối dữ liệu video hiện thời cần mã hóa bằng cách sử dụng kỹ thuật bù chuyển động afin;

xây dựng danh sách bộ dự đoán vector chuyển động (MVP) afin cho nhiều điểm điều khiển của mô hình afin để mã hóa khối dữ liệu video hiện thời, bao gồm việc thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời, và không thêm vector chuyển động từ khối dữ liệu video lân cận vào danh sách MVP afin dưới dạng bộ dự đoán vector chuyển động dựa vào vector chuyển động không có hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời,

trong đó để xây dựng danh sách MVP afin, thiết bị còn được tạo cấu hình để thu được một ứng viên từ nhiều bộ dự đoán vector chuyển động, trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) dựa vào mô hình afin là mô hình afin 6 thông số, và trong đó nhiều bộ dự đoán vector chuyển động bao gồm bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) dựa vào mô hình afin là mô hình afin 4 thông số;

xác định các vectơ chuyển động cho nhiều điểm điều khiển bằng cách sử dụng danh sách MVP afin; và

mã hóa khối dữ liệu video hiện thời với các vectơ chuyển động xác định cho nhiều điểm điều khiển của khối dữ liệu video hiện thời.

29. Thiết bị theo điểm 28, trong đó nhiều điểm điều khiển bao gồm ít nhất là điểm điều khiển thứ nhất, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ nhất, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ nhất của điểm điều khiển thứ nhất theo thứ tự định trước thứ nhất, trong đó các khối lân cận thứ nhất bao gồm khối lân cận ở trên bên trái (A), khối lân cận bên trên thứ nhất (B), và khối lân cận bên trái thứ nhất (C), và trong đó thứ tự định trước thứ nhất là A rồi C rồi B; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vectơ chuyển động thứ nhất (MVP0), vectơ chuyển động thứ nhất của các khối lân cận thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước.

30. Thiết bị theo điểm 29, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vectơ chuyển động thứ nhất cho điểm điều khiển thứ nhất là không có sẵn nếu không có vectơ chuyển động nào của các khối lân cận thứ nhất của điểm điều khiển thứ nhất có hình tham chiếu kết hợp giống với hình tham chiếu đích.

31. Thiết bị theo điểm 29, trong đó nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ hai, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ hai, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ hai của điểm điều khiển thứ hai theo thứ tự định trước thứ hai, trong đó các khối lân cận thứ hai bao gồm khối lân cận bên trên thứ hai (D), và khối lân cận bên trên bên phải (E), và trong đó thứ tự định trước thứ hai là D rồi E; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vectơ chuyển động thứ hai (MVP1), vectơ chuyển động thứ hai của các khối lân cận thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ hai.

32. Thiết bị theo điểm 31, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vector chuyển động thứ hai cho điểm điều khiển thứ nhất thứ hai là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ hai của điểm điều khiển thứ hai có hình tham chiếu kết hợp giống với hình tham chiếu đích.

33. Thiết bị theo điểm 31, trong đó nhiều điểm điều khiển còn bao gồm điểm điều khiển thứ ba, và trong đó để xây dựng danh sách MVP afin cho điểm điều khiển thứ ba, thiết bị còn được tạo cấu hình thêm để:

kiểm tra các khối lân cận thứ ba của điểm điều khiển thứ ba theo thứ tự định trước thứ ba, trong đó các khối lân cận thứ ba bao gồm khối lân cận bên trái thứ hai (F), và khối lân cận bên dưới bên trái (G), và trong đó thứ tự định trước thứ ba là F rồi G; và

thêm vào danh sách MVP afin, dưới dạng bộ dự đoán vector chuyển động thứ ba (MVP2), vector chuyển động thứ ba của các khối lân cận thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích xuất hiện đầu tiên theo thứ tự định trước thứ ba.

34. Thiết bị theo điểm 33, trong đó thiết bị còn được tạo cấu hình để:

đánh dấu bộ dự đoán vector chuyển động thứ ba cho điểm điều khiển thứ nhất thứ ba là không có sẵn nếu không có vector chuyển động nào của các khối lân cận thứ ba của điểm điều khiển thứ ba có hình tham chiếu kết hợp giống với hình tham chiếu đích.

35. Thiết bị theo điểm 33, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ ba (MVP2) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ hai (MVP1) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ ba (MVP2) không có sẵn.

36. Thiết bị theo điểm 33, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ hai (MVP1) dưới dạng hàm của bộ dự đoán vector chuyển động thứ nhất (MVP0) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ hai (MVP1) không có sẵn.

37. Thiết bị theo điểm 33, trong đó thiết bị còn được tạo cấu hình để:

xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn; và

tính toán bộ dự đoán vector chuyển động thứ nhất (MVP0) dưới dạng hàm của bộ dự đoán vector chuyển động thứ hai (MVP1) và bộ dự đoán vector chuyển động thứ ba (MVP2) để đáp lại việc xác định rằng bộ dự đoán vector chuyển động thứ nhất (MVP0) không có sẵn.

38. Thiết bị theo điểm 33, trong đó thiết bị còn được tạo cấu hình để:

làm tròn bộ dự đoán vector chuyển động thứ nhất (MVP0), bộ dự đoán vector chuyển động thứ hai (MVP1), và bộ dự đoán vector chuyển động thứ ba (MVP2) theo độ chính xác chuyển động định trước.

39. Thiết bị theo điểm 28, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp nằm trong cùng danh sách hình tham chiếu với hình tham chiếu đích và có cùng chỉ số tham chiếu với hình tham chiếu đích.

40. Thiết bị theo điểm 28, trong đó hình tham chiếu kết hợp giống với hình tham chiếu đích cho khối dữ liệu video hiện thời khi hình tham chiếu kết hợp có cùng số đếm thứ tự hình với hình tham chiếu đích.

1 / 14

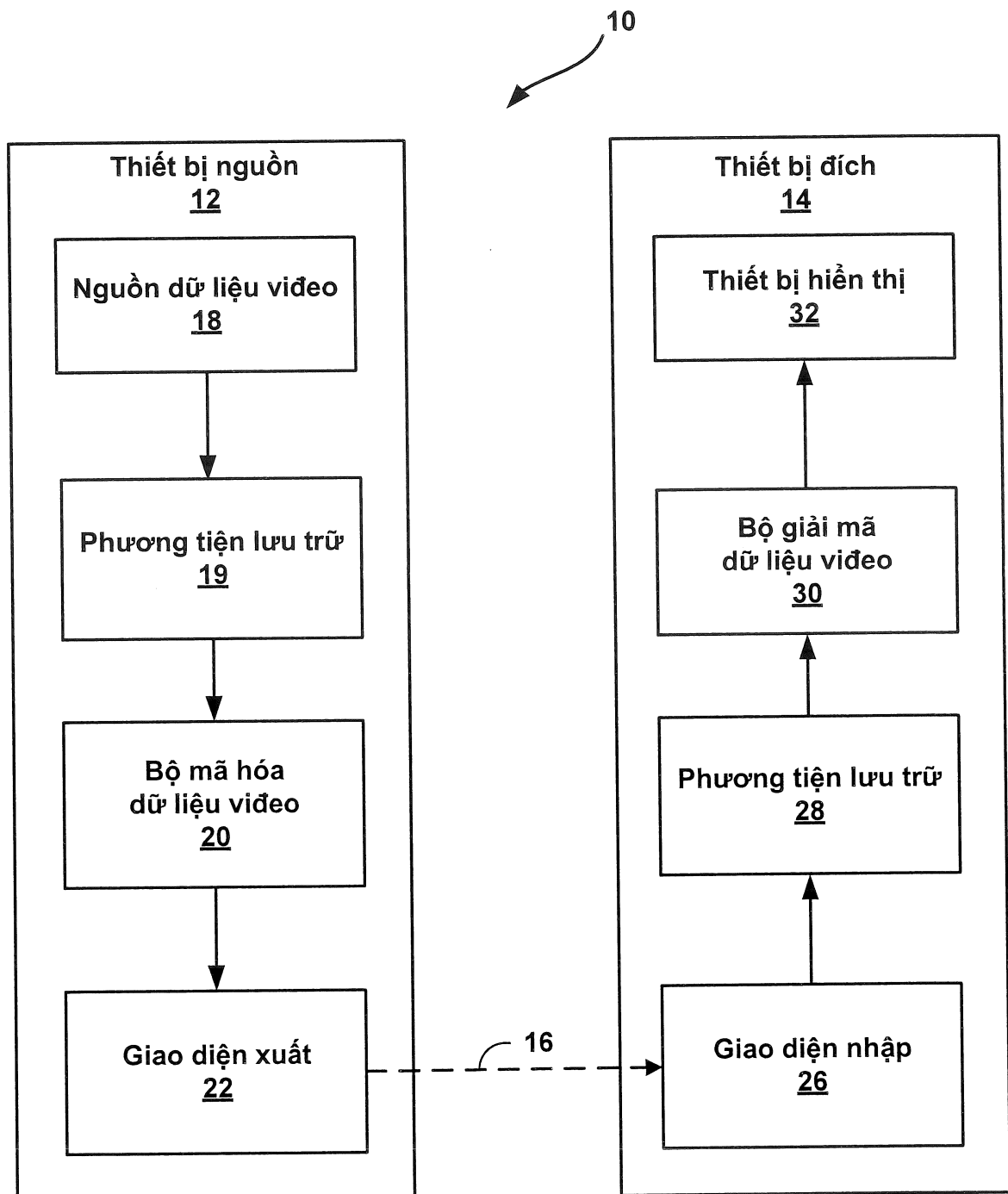


FIG.1

2 / 14

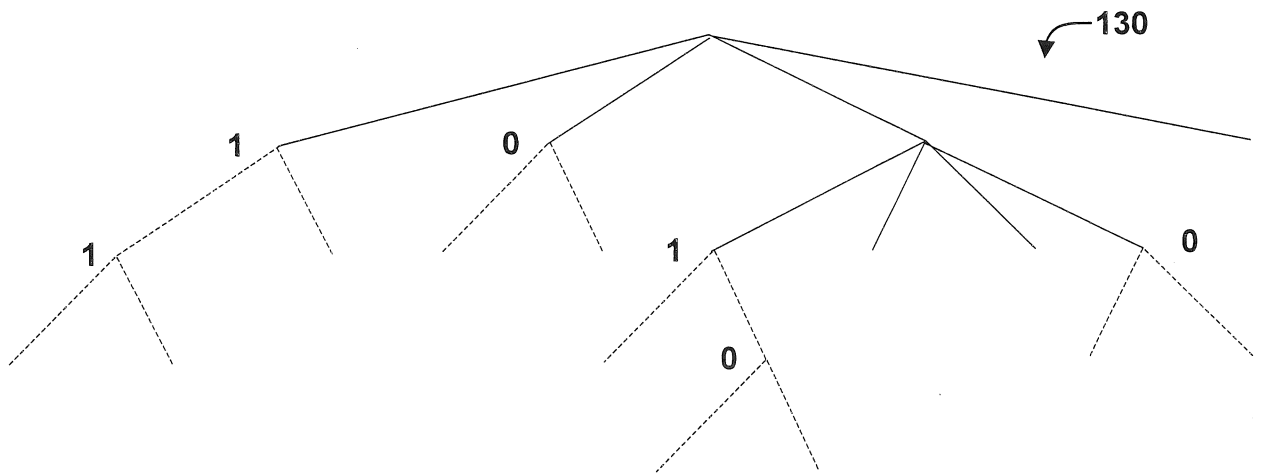


FIG.2A

132

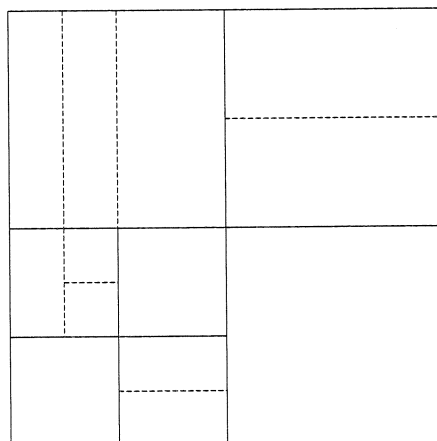


FIG.2B

3 / 14

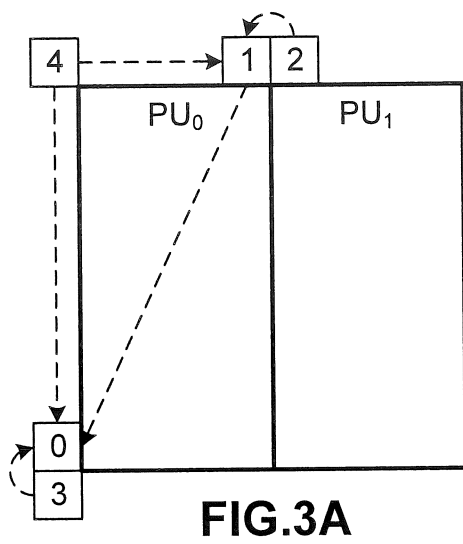


FIG.3A

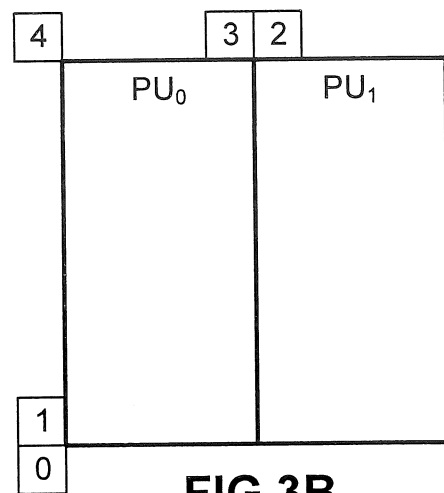


FIG.3B

4 / 14

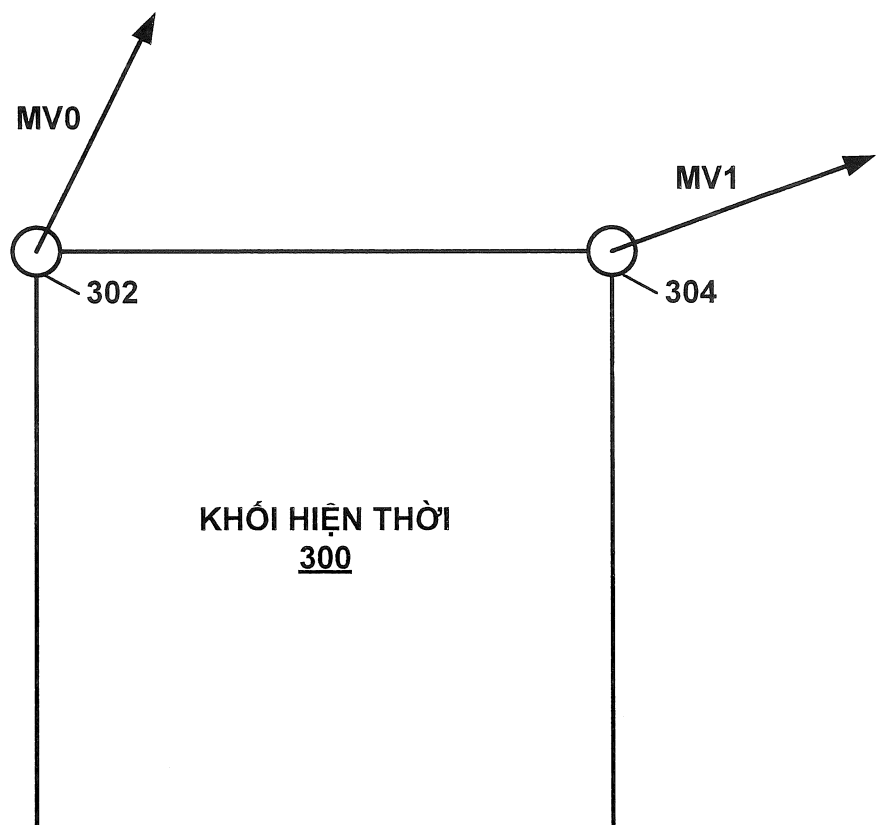


FIG.4

5 / 14

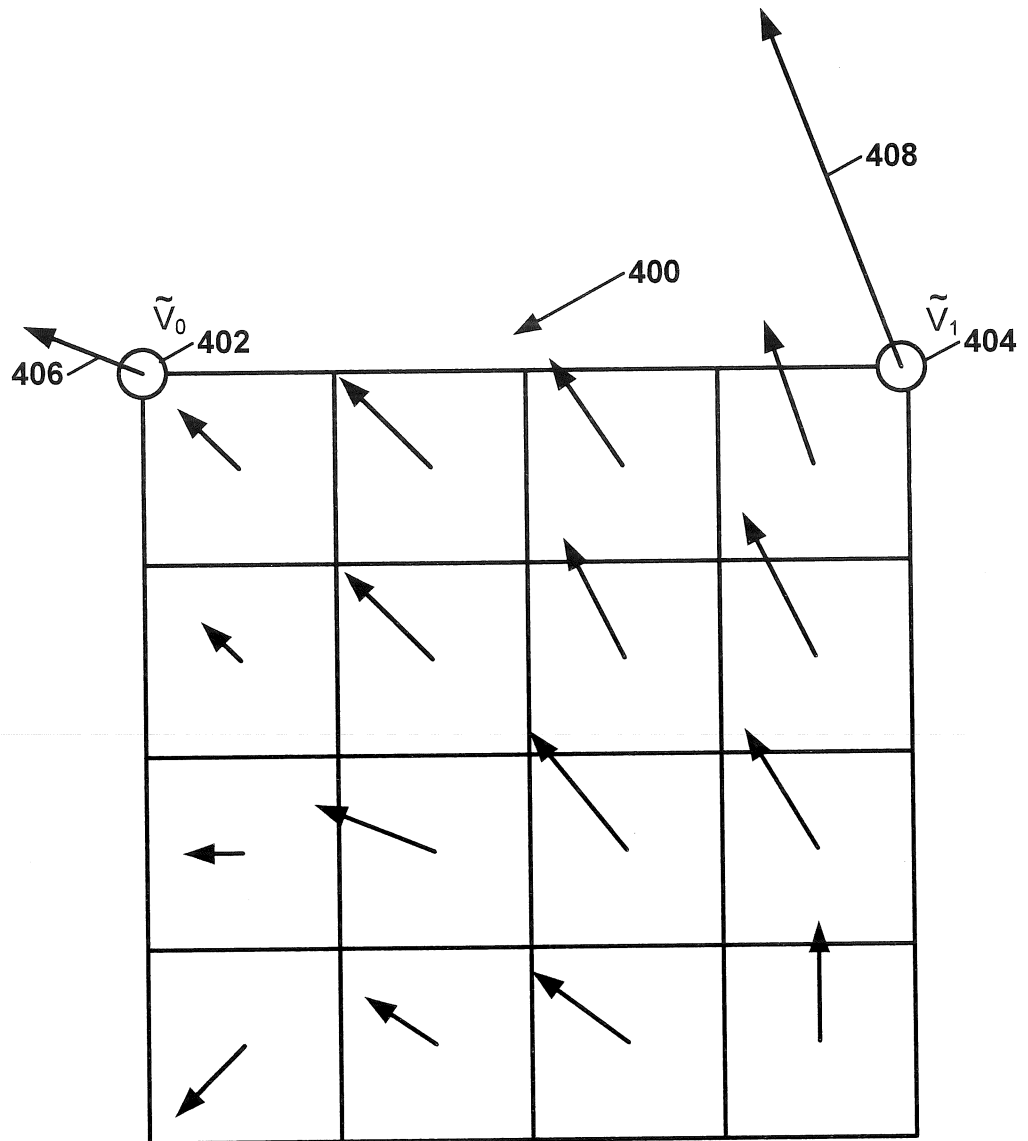


FIG.5

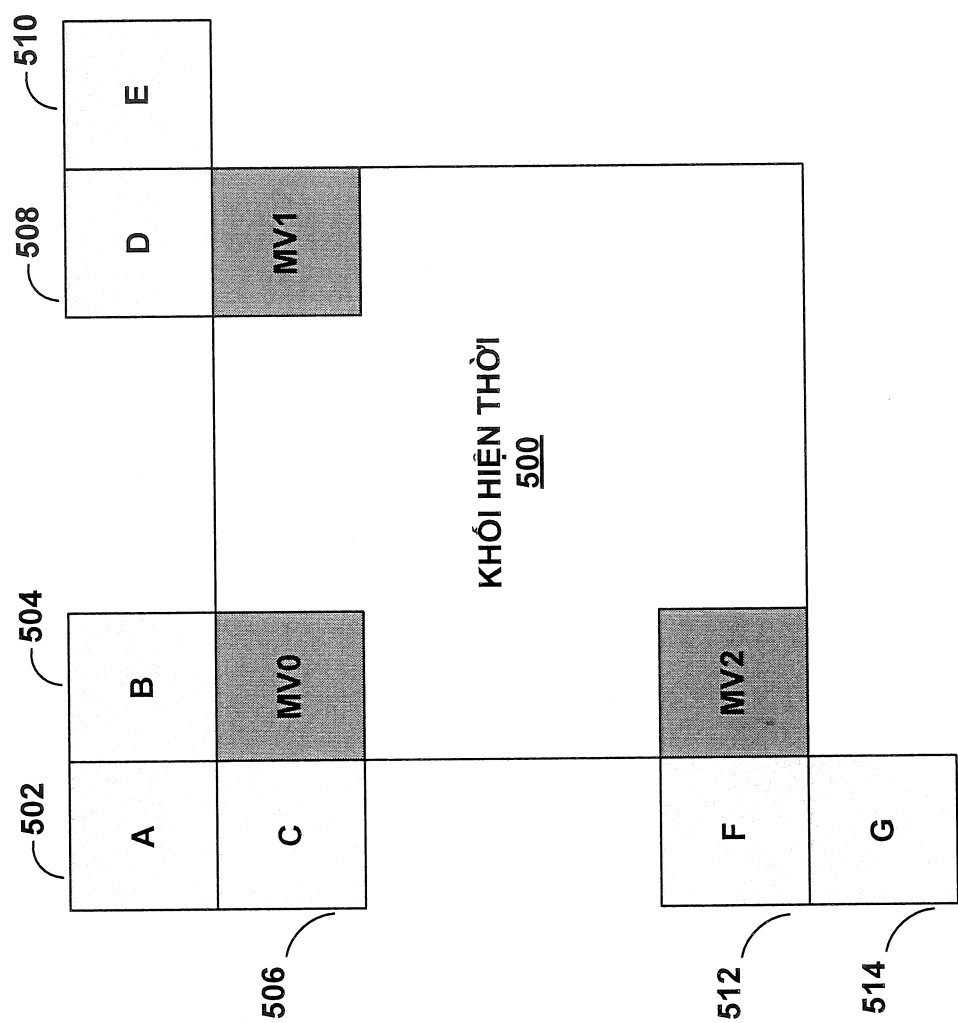


FIG.6

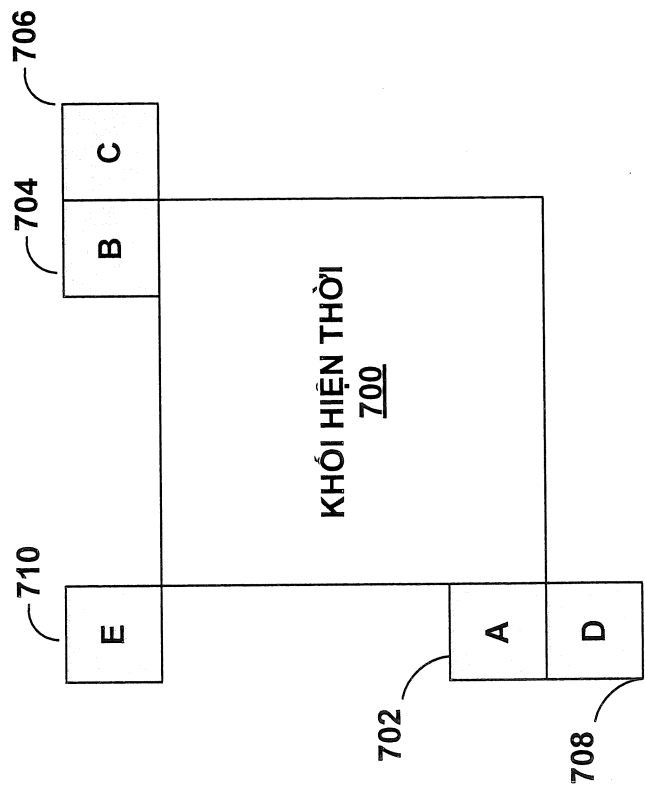


FIG. 7A

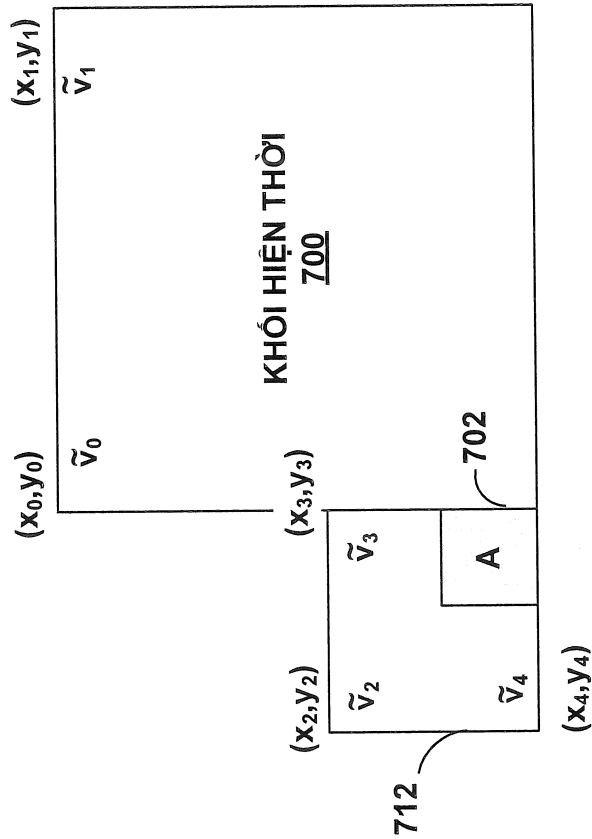


FIG. 7B

8 / 14

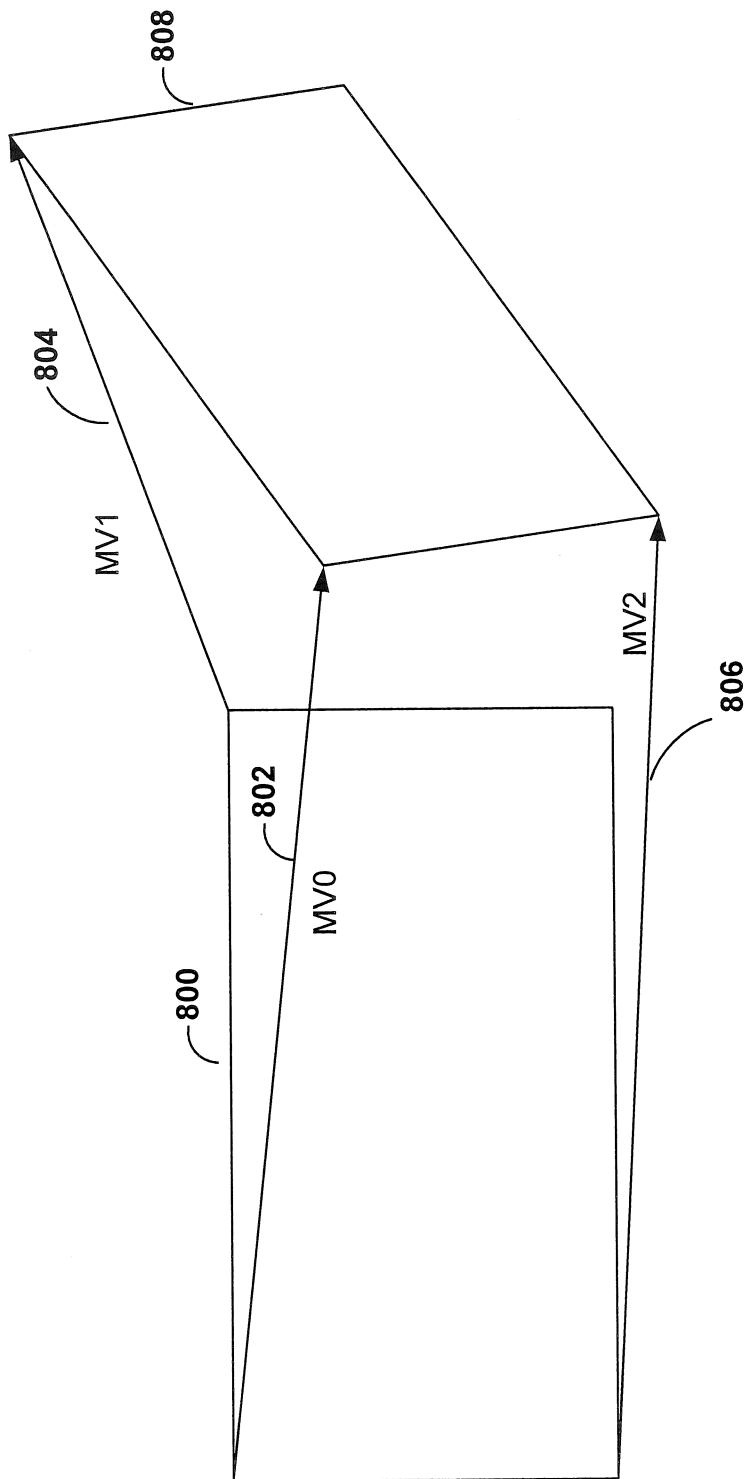


FIG.8

9 / 14

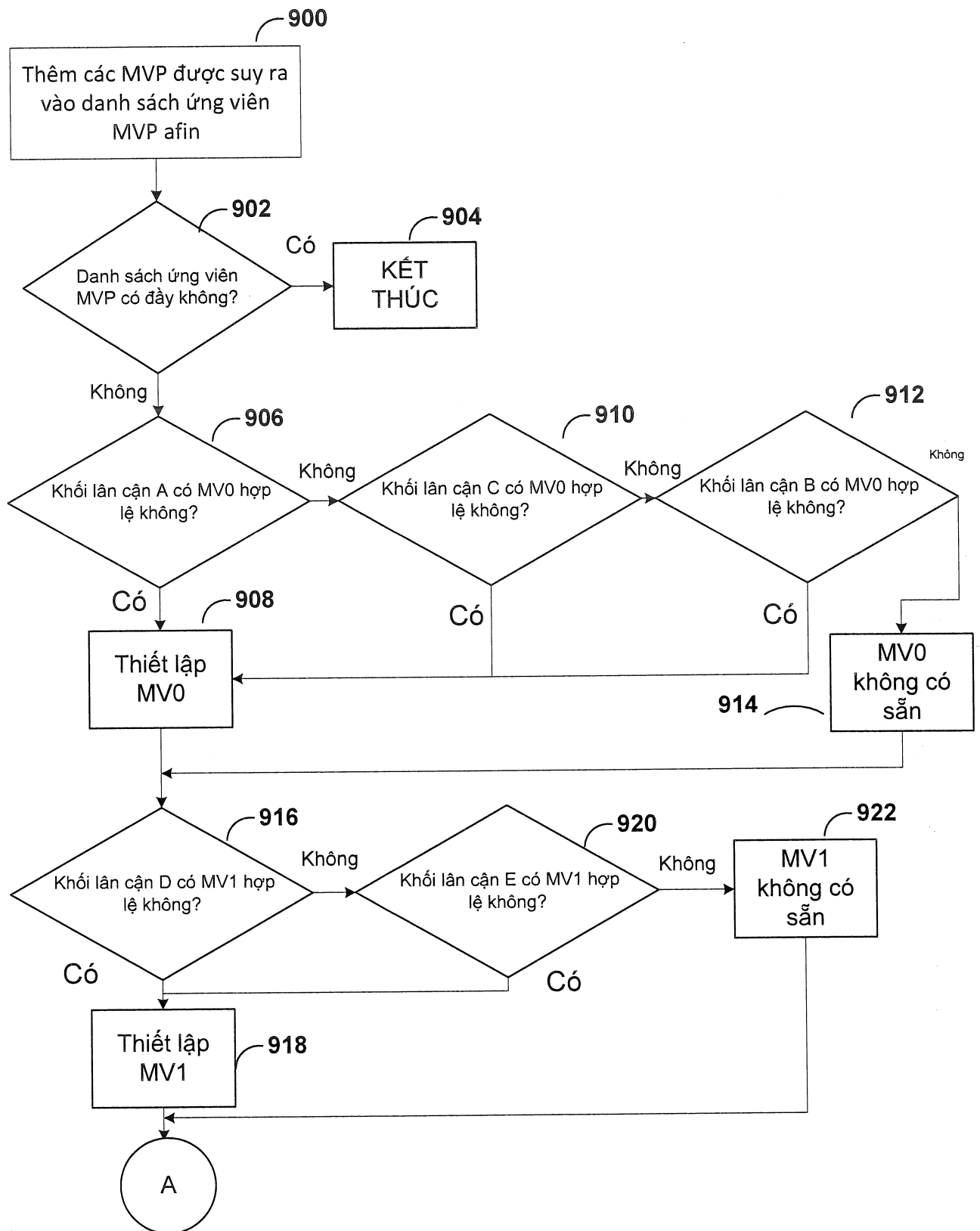


FIG.9A

10 / 14

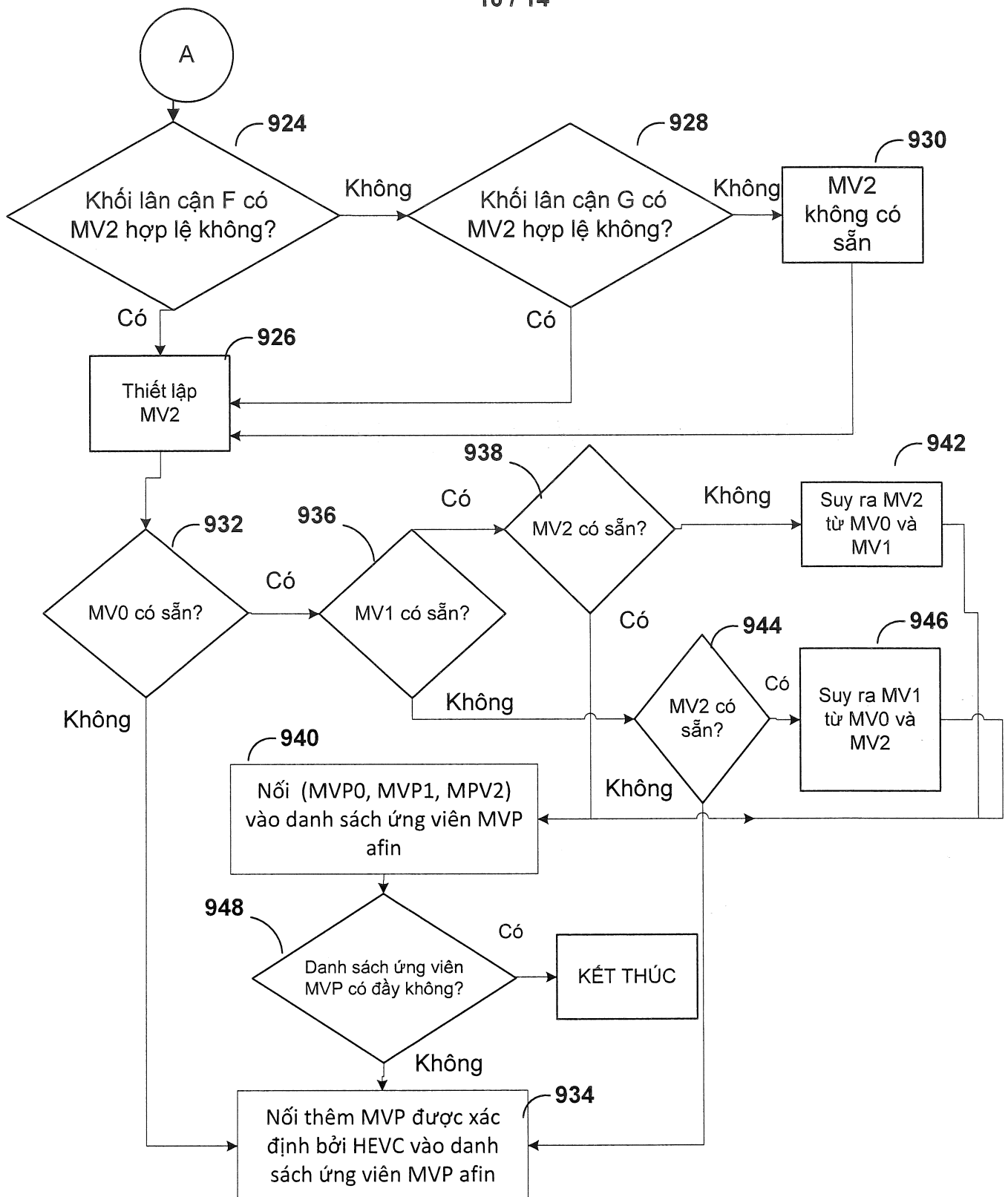


FIG.9B

11 / 14

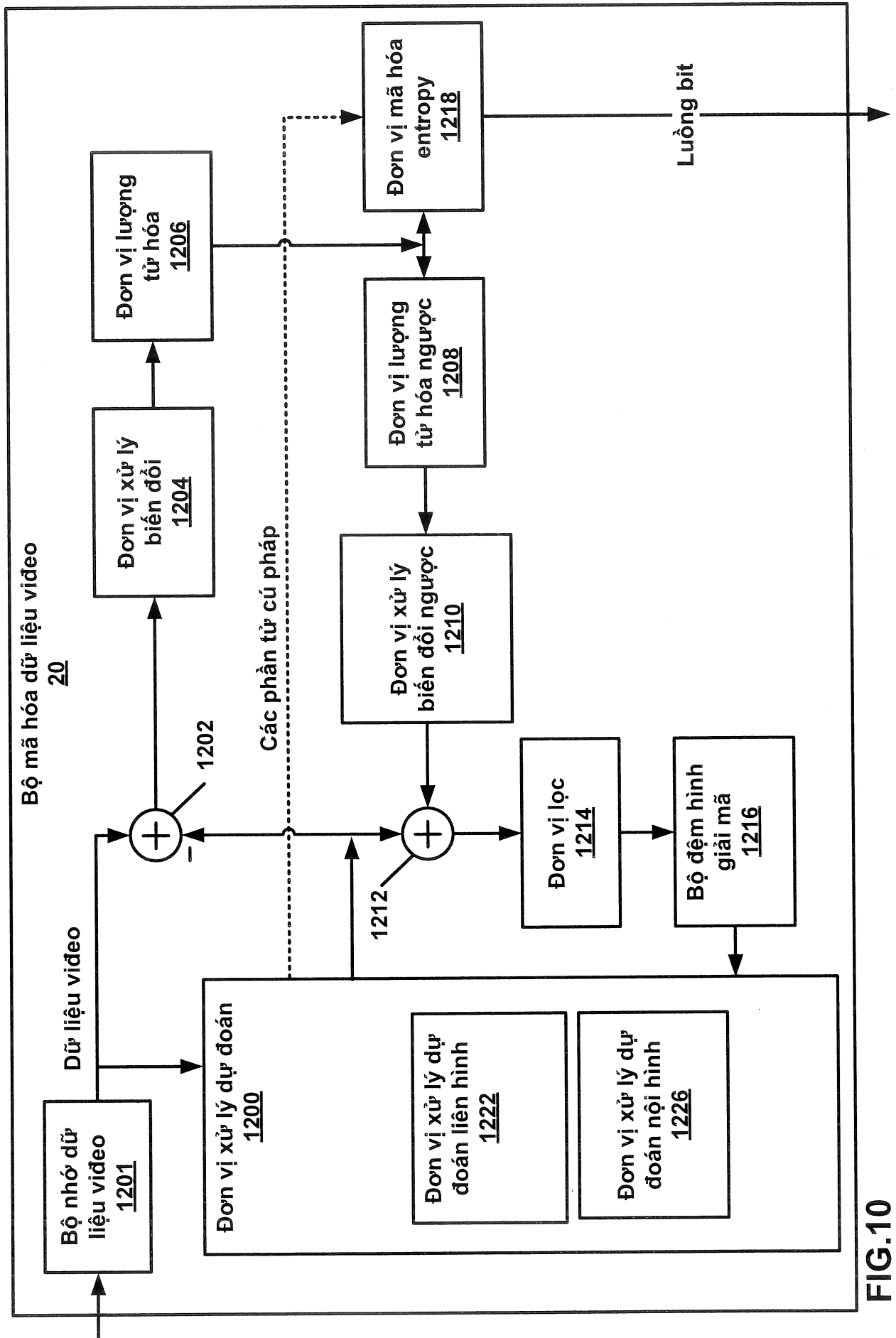


FIG.10

12 / 14

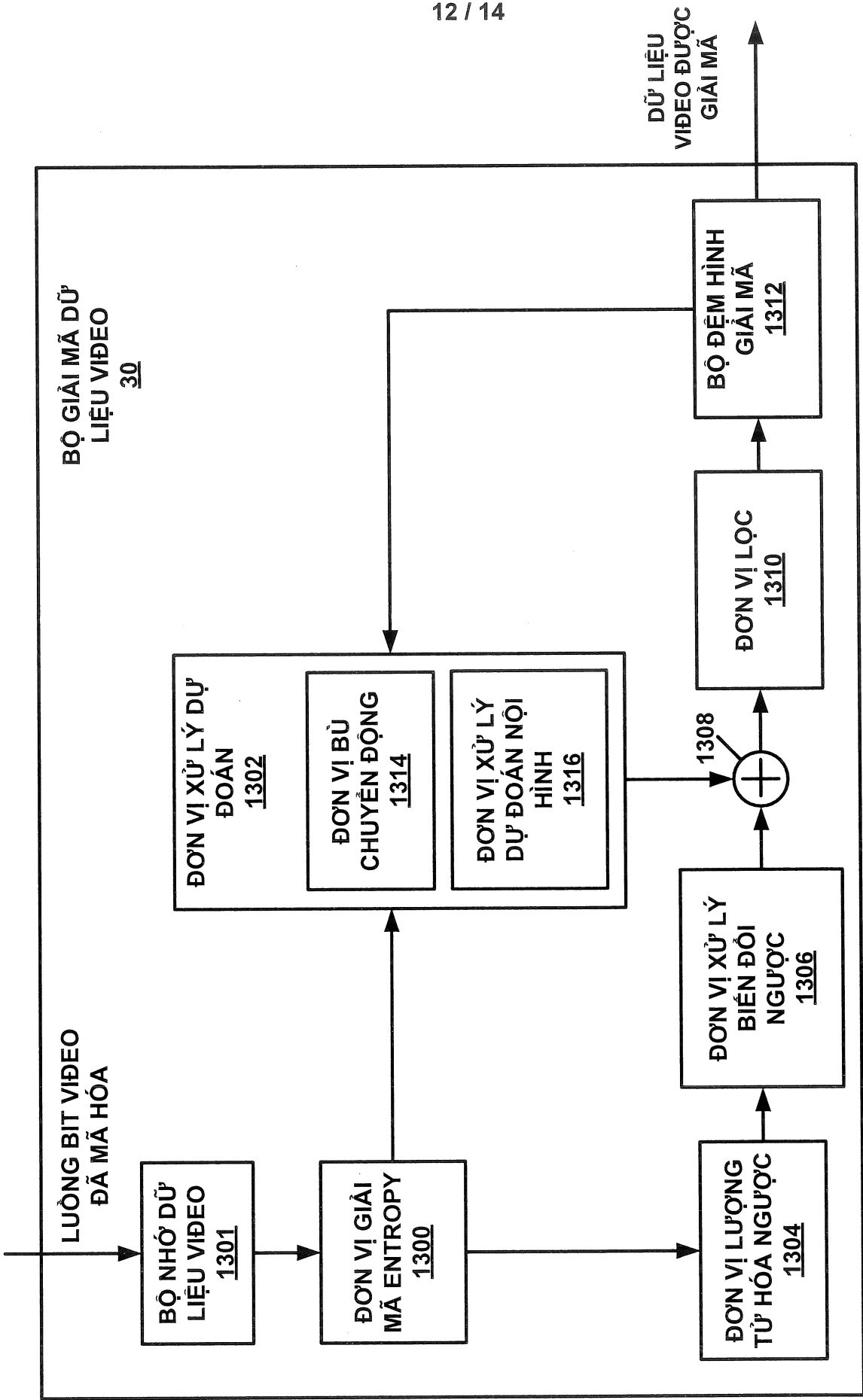


FIG.11

13 / 14

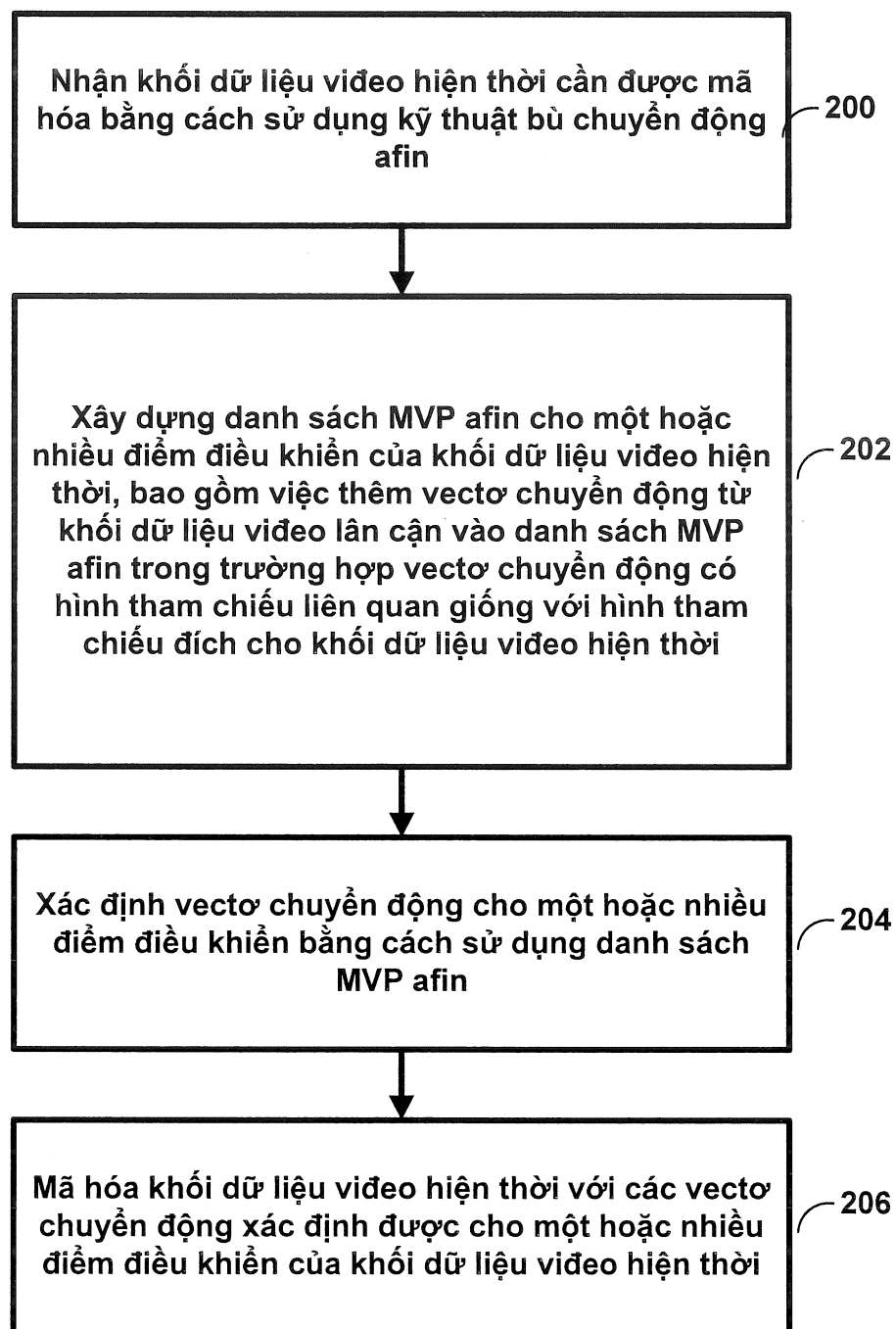


FIG.12

14 / 14

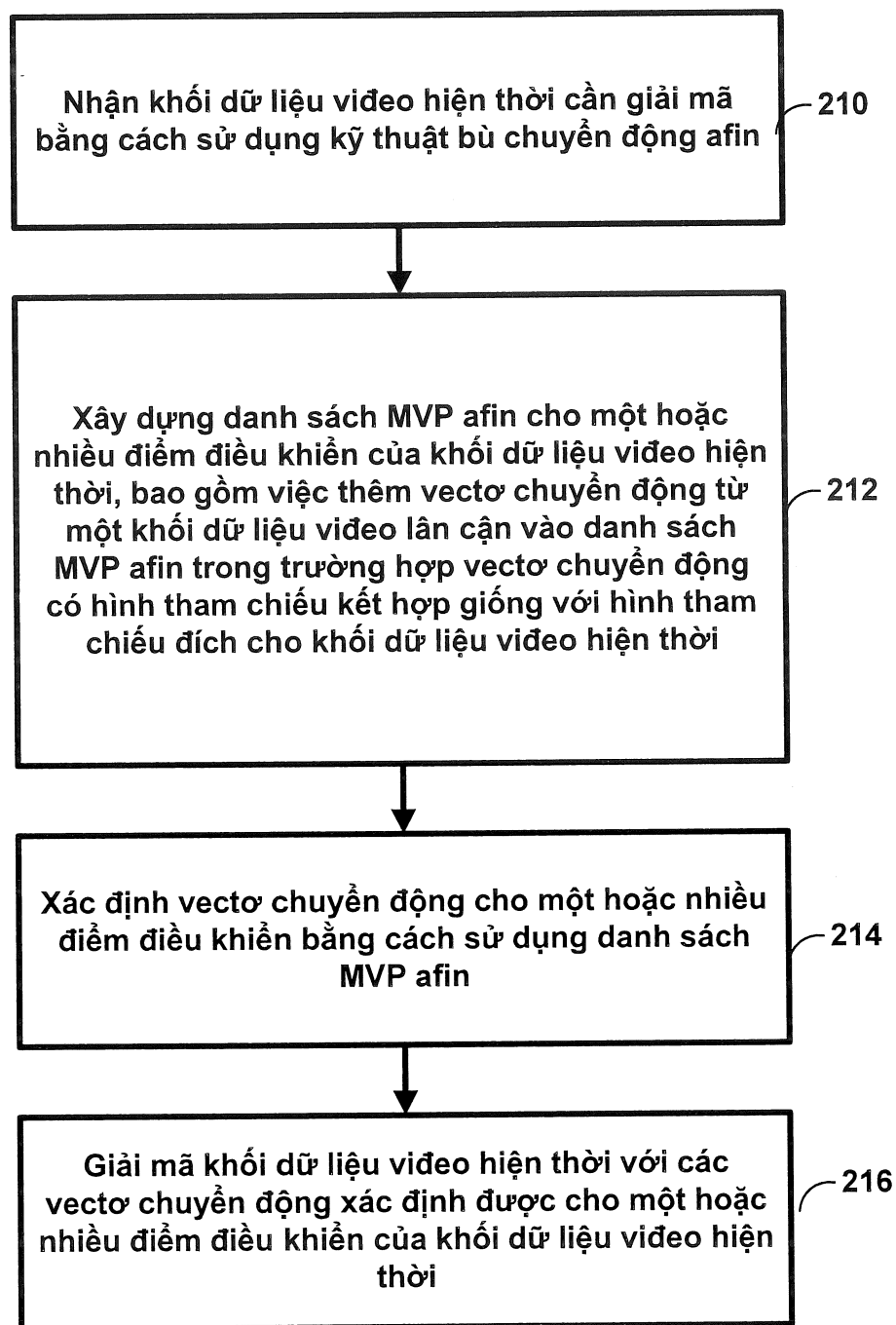


FIG.13