



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

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

(11)



1-0041320

(51)^{2020.01} H04N 19/105; H04N 19/54; H04N
19/52; H04N 19/137; H04N 19/176

(13) B

(21) 1-2020-01700

(22) 02/10/2018

(86) PCT/US2018/053936 02/10/2018

(87) WO 2019/070683 A1 11/04/2019

(30) 62/567,598 03/10/2017 US; 16/148,738 01/10/2018 US

(45) 25/10/2024 439

(43) 27/07/2020 388

(73) QUALCOMM INCORPORATED (US)

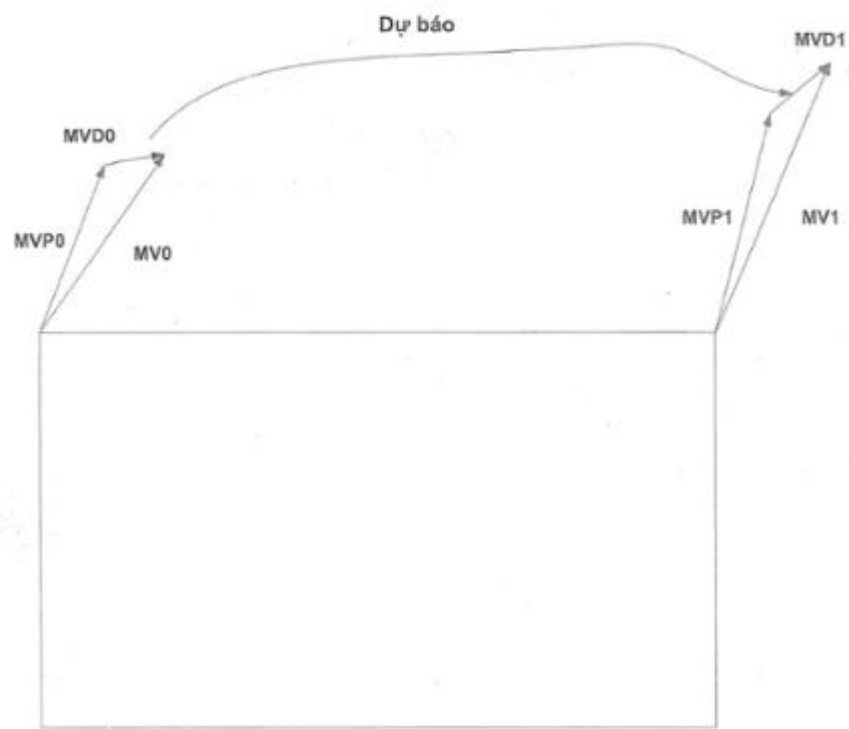
ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, CA
92121-1714, United States of America

(72) ZHANG, Kai (CN); CHEN, Jianle (CN); LI, Xiang (CN); CHIEN, Wei-Jung (TW);
CHEN, Yi-Wen (TW); ZHANG, Li (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 DỮ LIỆU VIDEO, VÀ PHƯƠNG TIỆN
LƯU TRỮ BẤT BIẾN ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(57) Sáng chế đề cập đến phương pháp và thiết bị lập mã dữ liệu video, và phương tiện lưu trữ bất biến đọc được bằng máy tính. Thiết bị làm ví dụ để lập mã dữ liệu video bao gồm bộ nhớ được tạo cấu hình để lưu trữ dữ liệu video, và một hoặc nhiều bộ xử lý được lắp đặt trong hệ mạch và được tạo cấu hình để lập mã chênh lệch vectơ chuyển động (motion vector difference - MVD) thứ nhất biểu diễn chênh lệch giữa vectơ chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vectơ chuyển động (motion vector predictor - MVP) thứ nhất cho vectơ chuyển động thứ nhất, dự đoán MVD thứ hai từ MVD thứ nhất cho vectơ chuyển động thứ hai của khối hiện thời, và lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vectơ chuyển động thứ nhất và vectơ chuyển động thứ hai dự đoán MVD thứ hai từ MVD thứ nhất theo cách này có thể giảm tốc độ bit của dòng bit chứa dữ liệu video mã hóa, cũng như cải thiện hiệu quả xử lý.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến việc mã hóa video, và cụ thể hơn, đến mã hóa thông tin chuyển động của dữ liệu video.

Tình trạng kỹ thuật của sáng chế

Các tính năng video số có thể được đưa vào áp dụng trong rất nhiều thiết bị bao gồm máy truyền hình kỹ thuật số, hệ thống phát quảng bá trực tiếp kỹ thuật số, hệ thống phát quảng bá không dây, thiết bị kỹ thuật số hỗ trợ cá nhân (personal digital assistant - PDA), máy tính xách tay hoặc máy tính để bàn, máy tính bảng, thiết bị đọc sách điện tử, máy ảnh kỹ thuật số, thiết bị ghi số, thiết bị phát phương tiện kỹ thuật số, thiết bị chơi trò chơi điện tử, bàn giao tiếp trò chơi điện tử, điện thoại di động hoặc điện thoại vô tuyến vệ tinh, được gọi là “điện thoại thông minh”, thiết bị hội nghị truyền hình, thiết bị phát trực tiếp video, và các thiết bị tương tự khác. Các thiết bị video số thực hiện các kỹ thuật mã hóa 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 video nâng cao (Advanced Video Coding - AVC), chuẩn mã hóa dữ liệu video hiệu suất cao (High Efficiency Video Coding - HEVC), ITU-T H.265/mã hóa dữ liệu video hiệu suất cao (HEVC), và các phiên bản mở rộng của các chuẩn như vậy. Các thiết bị video có thể truyền, nhận, mã hóa, giải mã và/hoặc lưu trữ thông tin video kỹ thuật số một cách hiệu quả hơn bằng cách thực hiện các kỹ thuật mã hóa video này.

Các kỹ thuật mã hóa video bao gồm dự đoán không gian (nội hình ảnh) và/hoặc dự đoán thời gian (liên hình ảnh) để giảm hoặc loại bỏ dữ liệu dư vốn có trong các chuỗi video. Đối với kỹ thuật mã hóa video dựa vào khối, lát video (ví dụ, hình ảnh video hoặc một phần hình ảnh video) có thể được tách thành các khối video, khối video này cũng có thể được gọi là đơn vị cây mã hóa (CTU - coding tree unit), đơn vị mã hóa (CU - coding unit) và/hoặc nút mã hóa. Các khối video trong lát mã hóa nội hình ảnh (I) của hình ảnh 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 ảnh. Các khối video trong lát mã hóa liên hình

ảnh (P hoặc B) của hình ảnh 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 ảnh, hoặc dự đoán thời gian đối với các mẫu tham chiếu trong các hình ảnh tham chiếu khác. Hình ảnh có thể được gọi là khung, và hình ảnh tham chiếu có thể được gọi là khung tham chiếu.

Kỹ thuật dự đoán không gian hoặc thời gian tạo ra khối dự đoán cho khối cần mã hóa. Dữ liệu dư biểu diễn chênh lệch điểm ảnh giữa khối ban đầu cần mã hóa và khối dự đoán. Khối mã hóa liên hình ảnh được mã hóa theo vector chuyển động trở đến khối mẫu tham chiếu tạo thành khối dự đoán, và dữ liệu dư biểu thị chênh lệch giữa khối mã hóa và khối dự đoán. Khối mã hóa nội hình ảnh được mã hóa theo chế độ mã hóa nội hình ảnh 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. Các hệ số biến đổi đã lượng tử hóa, ban đầu được sắp xếp theo mảng hai chiều, có thể được quét để tạo ra vector một chiều gồm các hệ số biến đổi, và kỹ thuật mã hóa entropy có thể được áp dụng để đạt hiệu quả nén thậm chí cao hơn.

Bản chất kỹ thuật của sáng chế

Nói chung, sáng chế mô tả các kỹ thuật liên quan đến dự đoán liên hình ảnh. Ví dụ, các kỹ thuật theo sáng chế bao gồm việc lập mã (mã hóa và/hoặc giải mã) vector chuyển động để bù chuyển động afin trong lập mã video dựa vào khối. Các kỹ thuật này có thể được áp dụng cho các chuẩn mã hóa video hiện có và/hoặc trong tương lai.

Theo một ví dụ, phương pháp lập mã (ví dụ, mã hóa hoặc giải mã) dữ liệu video bao gồm bước lập mã chênh lệch vector chuyển động (motion vector difference - MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán sử dụng dự đoán afin và bộ dự đoán vector chuyển động (motion vector predictor - MVP) thứ nhất cho vector chuyển động thứ nhất, dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, và lập mã khối hiện thời sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

Theo ví dụ khác, thiết bị lập mã (ví dụ, mã hóa hoặc giải mã) dữ liệu video bao gồm bộ nhớ được tạo cấu hình để lưu trữ dữ liệu video, và một hoặc nhiều bộ xử lý được lắp đặt trong hệ mạch và được tạo cấu hình để lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện

thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất, dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, và lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

Theo ví dụ khác, phương tiện lưu trữ đọc được bằng máy tính lưu trữ trên đó các lệnh mà, khi thực thi, khiến cho bộ xử lý lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất, dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, và lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

Theo một ví dụ khác, thiết bị lập mã (ví dụ, mã hóa hoặc giải mã) dữ liệu video bao gồm phương tiện lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất, phương tiện dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, và phương tiện lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

Các chi tiết của một hoặc nhiều ví dụ được trình bày trên các hình vẽ kèm theo và phân mô tả dưới đây. Các dấu hiệu, mục đích và ưu điểm khác sẽ trở nên rõ ràng qua phần mô tả và các hình vẽ, và qua các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Các hình vẽ Fig.1A và Fig.1B là sơ đồ khái niệm minh họa các ví dụ về các ứng viên lân cận theo không gian cho các chế độ hợp nhất và dự đoán vector chuyển động cải tiến (advanced motion vector prediction - AMVP) của mã hóa video hiệu suất cao (HEVC).

Fig.2 là sơ đồ khái niệm minh họa ví dụ về afin vector chuyển động hai điểm với bốn tham số afin cho khối hiện thời.

Fig.3 là sơ đồ khái niệm minh họa ví dụ về chế độ dự đoán liên hình ảnh afin cho khối hiện thời.

Các hình vẽ Fig.4A và Fig.4B là các sơ đồ khái niệm minh họa ví dụ về chế độ hợp nhất afin cho khối hiện thời.

Fig.5 là sơ đồ khái niệm minh họa ví dụ về mô hình afin với sáu tham số (ba vector chuyển động).

Fig.6 là sơ đồ khối minh họa ví dụ về hệ thống mã hóa và giải mã video mà có thể sử dụng kỹ thuật theo sáng chế để lập mã có hiệu quả thông tin chuyển động cho dự đoán afin.

Fig.7 là sơ đồ khối minh họa ví dụ về bộ mã hóa video mà có thể thực hiện các kỹ thuật theo sáng chế để mã hóa thông tin chuyển động dự đoán afin

Fig.8 là sơ đồ khối minh họa ví dụ về bộ giải mã video 30 mà có thể thực hiện các kỹ thuật theo sáng chế để giải mã thông tin chuyển động dự đoán afin

Fig.9 là sơ đồ khái niệm minh họa ví dụ về dự đoán chênh lệch vector chuyển động (MVD) cho dự đoán thông tin chuyển động afin.

Fig.10 là sơ đồ khái niệm minh họa ví dụ về dự đoán MVD cho dự đoán afin với ba vector chuyển động (dự đoán afin sáu tham số).

Fig.11 là lưu đồ minh họa ví dụ về phương pháp mã hóa khối dữ liệu video hiện thời theo các kỹ thuật của sáng chế.

Fig.12 là lưu đồ minh họa ví dụ về phương pháp giải mã khối dữ liệu video hiện thời theo các kỹ thuật của sáng chế.

Mô tả chi tiết sáng chế

Các chuẩn mã hóa video bao gồm 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 (cũng được biết đến là ISO/IEC MPEG-4 AVC), bao gồm các phần mở rộng mã hóa video có khả năng mở rộng (Scalable Video Coding - SVC) và mã hóa video đa khung hình (Multiview Video Coding - MVC).

Gần đây, thiết kế chuẩn mã hóa video mới, cụ thể là ITU-T H.265/Mã Hóa Video Hiệu Suất Cao (High Efficiency Video Coding - HEVC), đã được hoàn thiện bởi Nhóm Cộng tác Chung về mã hóa Video (Joint Collaboration Team on Video Coding - JCT-VC) của Nhóm Chuyên gia về mã hóa Video ITU-T (Video Coding Experts Group - VCEG) và Nhóm Chuyên Gia về Ảnh Động ISO/IEC (Motion Picture Experts Group - MPEG).

Đặc tả dự thảo HEVC mới nhất, và sau đây được gọi là HEVC WD, có thể truy cập được từ 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 JCTVC phát triển. Dự thảo làm việc (Working Draft - WD) của các phần mở rộng, sau đây gọi là RExt WD6, có thể truy cập được từ phenix.int-evry.fr/jct/doc_end_user/documents/16_San%20Jose/wg11/JCTVC-P1005-v1.zip.

Khảo sát về các công cụ mã hóa mới để mã hóa video trong tương lai đang diễn ra (ví dụ, như được nghiên cứu trong Nhóm khảo sát video chung - JVET) và các công nghệ cải thiện hiệu quả mã hóa cho mã hóa video đã được đề xuất. Có bằng chứng cho thấy có thể đạt được những cải thiện đáng kể trong hiệu quả mã hóa bằng cách khai thác các đặc điểm của 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. Các công ty và tổ chức được mời đệ trình các đề xuất trong giai đoạn khảo sát để có thể cải thiện hơn nữa hiệu quả mã hóa video.

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

Đối với mỗi hướng dự đoán, theo H.265/HEVC, thông tin chuyển động phải chứa chỉ số tham chiếu và vectơ chuyển động. Trong một số trường hợp, để đơn giản hóa, bản thân vectơ chuyển động có thể được gọi theo cách mà giả định rằng vectơ chuyển động có chỉ số tham chiếu gắn kèm. Chỉ số tham chiếu được dùng để nhận dạng hình ảnh tham chiếu trong danh sách hình ảnh tham chiếu hiện thời (RefPicList0 hoặc RefPicList1). Vectơ chuyển động có thành phần ngang hoặc dọc.

Số đếm thứ tự hình (POC - Picture Order Count) được sử dụng rộng rãi trong các chuẩn mã hóa video để nhận dạng thứ tự hiển thị của hình. Mặc dù có các trường hợp trong

đó hai hình trong một chuỗi video mã hóa có thể có cùng giá trị POC, nhưng điều này thường không xảy ra trong một chuỗi video mã hóa. Khi nhiều chuỗi video mã hóa có mặt trong dòng bit, thì các hình có cùng giá trị POC có thể ở gần nhau về mặt thứ tự giải mã. Các giá trị POC của các hình thường được dùng để tạo ra danh sách hình ảnh tham chiếu, suy ra tập hợp hình ảnh tham chiếu như trong HEVC, và định tỷ lệ vector chuyển động.

Trong HEVC, đơn vị mã hóa lớn nhất trong lát được gọi là khối cây mã hóa (CTB - coding tree block). CTB chứa cây tứ phân, các nút của nó là các đơn vị mã hóa. Kích thước của CTB có thể nằm trong khoảng từ 16x16 điểm ảnh đến 64x64 điểm ảnh theo profin chính của chuẩn HEVC (mặc dù về mặt kỹ thuật các kích thước CTB 8x8 có thể được hỗ trợ). Đơn vị mã hóa (CU) có thể có cùng kích thước với CTB và nhỏ bằng 8x8 điểm ảnh. Mỗi đơn vị mã hóa có thể được mã hóa với một chế độ. Khi CU được mã hóa liên hình ảnh, thì CU có thể được phân chia tiếp thành hai hoặc nhiều đơn vị dự đoán (PU) hoặc chỉ trở thành một PU khi không áp dụng quy trình phân chia thêm nữa. Khi hai PU có mặt trong một CU, thì 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 ảnh, 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 ảnh 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, có hai chế độ dự đoán liên hình ảnh, được gọi là chế độ hợp nhất (chế độ bỏ qua được coi là trường hợp đặc biệt của chế độ hợp nhất) và các chế độ dự đoán vector chuyển động cải tiến (AMVP - advanced motion vector prediction), cho đơn vị dự đoán (prediction unit-PU). Ở chế độ AMVP hoặc chế độ hợp nhất, danh sách ứng viên vector chuyển động (motion vector - MV) được duy trì cho nhiều bộ dự đoán vector chuyển động. (Các) vector chuyển động, cũng như các chỉ số tham chiếu trong chế độ hợp nhất, của PU hiện thời được tạo ra bằng cách lấy một ứng viên từ danh sách ứng viên MV.

Danh sách ứng viên MV, theo HEVC, chứa tối đa 5 ứng viên cho chế độ hợp nhất và chỉ 2 ứng viên cho chế độ AMVP. Ứng viên hợp nhất có thể chứa tập hợp thông tin chuyển động, ví dụ, các vector chuyển động tương ứng với cả danh sách hình ảnh tham chiếu (danh sách 0 và danh sách 1) và các chỉ số tham chiếu. Nếu ứng viên hợp nhất được nhận dạng bởi chỉ số hợp nhất, thì các hình ảnh tham chiếu được sử dụng để dự đoán các

khối hiện thời, cũng như các vector chuyển động gắn kèm được xác định. Tuy nhiên, trong chế độ AMVP, đối với mỗi chiều dự đoán tiềm năng từ danh sách 0 hoặc danh sách 1, chỉ số tham chiếu cần được báo hiệu rõ ràng, cùng với chỉ số bộ dự đoán vector chuyển động (MVP) đến danh sách ứng viên MV bởi vì ứng viên AMVP chỉ chứa một vector chuyển động. 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 trong khi ứng viên AMVP chỉ chứa một vector chuyển động cho một chiều dự đoán cụ thể và chỉ số tham chiếu. Các ứng viên cho cả hai chế độ được suy ra một cách tương tự từ cùng một khối lân cận không gian và thời gian.

Các hình vẽ Fig.1A và Fig.1B là sơ đồ khái niệm minh họa ví dụ về các ứng viên lân cận theo không gian cho các chế độ hợp nhất và AMVP của HEVC. Cụ thể, Fig.1A minh họa các ứng viên vector chuyển động (MV - motion vector) lân cận theo không gian cho chế độ hợp nhất, trong khi đó Fig.1B minh họa các ứng viên MV lân cận theo không gian cho chế độ AMVP. Theo HEVC, các ứng viên MV không gian được suy ra từ các khối lân cận được thể hiện trên các hình vẽ Fig.1A và Fig.1B, cho một PU riêng (PU_0), mặc dù các kỹ thuật để tạo ra các ứng viên từ các khối khác nhau đối với các chế độ hợp nhất và AMVP.

Ở chế độ hợp nhất của HEVC, tối đa bốn ứng viên MV không gian có thể được suy ra theo thứ tự được thể hiện trên Fig.1A bằng các số, và thứ tự này như sau: bên trái (0), phía trên (1), bên phải phía trên (2), bên trái phía dưới (3), và bên phải phía trên (4), như được thể hiện trên Fig.1A.

Ở chế độ AMVP của HEVC, các khối lân cận được chia thành hai 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.1B. Đối với mỗi nhóm, ứng viên tiềm năng trong khối lân cận tham chiếu đến cùng một hình ảnh tham chiếu như hình 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 không có khối lân cận nào chứa vector chuyển động trỏ đến cùng một hình ảnh tham chiếu. Do đó, nếu không thể tìm thấy ứng viên như vậy, thì ứng viên có sẵn thứ nhất sẽ được định tỷ lệ để tạo ra ứng viên cuối cùng. Do đó, các chênh lệch khoảng cách thời gian có thể được bù.

Quy trình bù chuyển động trong H.265/HEVC được dùng để tạo ra bộ dự đoán cho khối mã hóa liên hình ảnh hiện thời. Vector chuyển động có độ chính xác một phần tư điểm ảnh có thể được dùng, và các giá trị điểm ảnh tại các vị trí phân số có thể được nội suy nhờ sử dụng các giá trị điểm ảnh nguyên lân cận cho cả các thành phần độ chói và màu.

Fig.2 là sơ đồ khái niệm minh họa ví dụ về afin vector chuyển động hai điểm với bốn tham số afin cho khối hiện thời. Trong các tiêu chuẩn bộ mã hóa-giải mã video tồn tại hiện nay, chỉ mô hình chuyển động tịnh tiến được áp dụng để dự đoán bù chuyển động (motion compensation prediction - MCP). Tuy nhiên, trên thực tế thực, có nhiều loại chuyển động, ví dụ chuyển động phóng to/ thu nhỏ, chuyển động quay, chuyển động phối cảnh và các chuyển động không đều khác. Nếu chỉ mô hình chuyển động tịnh tiến cho MCP được áp dụng trong các trình tự kiểm tra này với các chuyển động không đều, thì việc đó sẽ ảnh hưởng đến độ chính xác dự đoán và dẫn đến hiệu suất mã hóa thấp. Trong nhiều năm, nhiều chuyên gia video đã cố gắng thiết kế nhiều thuật toán để cải thiện MCP cho hiệu suất mã hóa cao hơn. Các chế độ hợp nhất afin và liên (AMVP) afin đã được đề xuất để xử lý các mô hình chuyển động afin với 4 tham số như:

$$\begin{cases} V_x = ax + by + e \\ V_y = cx + dy + f \end{cases} \quad (1)$$

Trong phương trình (1) trên đây, (v_{x0}, v_{y0}) là vector chuyển động điểm điều khiển tại góc trái trên cùng của khối hiện thời trên Fig.2, và (v_{x1}, v_{y1}) là vector chuyển động điểm điều khiển khác tại góc phải trên của khối hiện thời trên Fig.2. Mô hình afin chung quy lại là:

$$\begin{cases} V_x = \frac{(v_{1x}-v_{0x})}{w}x + \frac{(v_{2x}-v_{0x})}{h}y + v_{0x} \\ V_y = \frac{(v_{1y}-v_{0y})}{w}x + \frac{(v_{2y}-v_{0y})}{h}y + v_{0y} \end{cases} \quad (2)$$

Trong phần mềm JEM hiện tại, dự đoán chuyển động afin chỉ được áp dụng cho các khối vuông. Như phần mở rộng tự nhiên, dự đoán chuyển động afin có thể được áp dụng cho các khối không vuông.

Fig.3 là sơ đồ khái niệm minh họa ví dụ về chế độ dự đoán liên hình ảnh afin cho khối hiện thời. Khối hiện thời này có thể là CU hiện thời hoặc PU hiện thời. Theo ví dụ này, khối hiện thời bao gồm hai khối được đánh dấu “V0” tại góc trái trên và “V1” tại góc

phải trên, và các khối lân cận được đánh dấu A, B, C, D, và E. Cụ thể, khối “V0” gần các khối A, B, và C, trong khi khối “V1” gần các khối D và E.

Đối với mỗi CU/PU có kích thước bằng hoặc lớn hơn 16x16, chế độ dự đoán liên hình ảnh afin (chế độ AF_INTER) có thể được áp dụng như sau. Nếu CU/PU hiện thời là ở chế độ AF_INTER, thì cờ afin tại mức CU/PU có thể được báo hiệu trong dòng bit. Danh sách ứng viên $\{(v_0, v_1) | v_0 = \{v_A, v_B, v_C\}, v_1 = \{v_D, v_E\}\}$ được tạo ra nhờ sử dụng các khối được tái tạo hợp lệ lân cận.

Như được thể hiện trên Fig.3, thông tin chuyển động v_0 được chọn từ các vector chuyển động của các khối A, B và/hoặc C. Vector chuyển động từ khối lân cận được định tỷ lệ theo danh sách tham chiếu và mối quan hệ giữa POC của hình ảnh tham chiếu cho khối lân cận, POC của hình ảnh tham chiếu cho CU/PU hiện thời và POC của CU/PU hiện thời. Và cách tiếp cận để chọn v_1 từ các khối lân cận D và E là tương tự. Nếu số lượng của danh sách ứng viên là nhỏ hơn 2, thì các ứng viên của AMVP được gán cho v_0 and v_1 . Chi phí tối ưu hóa tốc độ-độ méo (rate distortion optimization - RDO) của CU/PU hiện thời được dùng để xác định (v_0, v_1) nào được chọn làm dự đoán vector chuyển động của điểm điều khiển (control point motion vector prediction - CPMVP) của CU/PU hiện thời. Và chỉ số để biểu thị vị trí của CPMVP trong danh sách ứng viên được báo hiệu trong dòng bit.

Sau khi CPMVP của CU/PU afin hiện thời được xác định, việc ước lượng chuyển động afin được áp dụng và CPMV được tìm thấy. Sau đó chênh lệch của CPMV và CPMVP được mã hóa trong dòng bit. Dự đoán bù chuyển động afin nêu trên được áp dụng để tạo ra các phần dư của CU/PU hiện thời. Cuối cùng, phần dư của CU/PU hiện thời được biến đổi, lượng tử hóa, và mã hóa thành dòng bit, theo các quy trình thông thường.

Các hình vẽ Fig.4A và Fig.4B là sơ đồ khái niệm minh họa ví dụ về chế độ hợp nhất afin cho khối hiện thời. Khối hiện thời này có thể là CU hiện thời hoặc PU hiện thời. Theo ví dụ này, khối hiện thời có năm khối lân cận, được đánh dấu A, B, C, D, và E, như được thể hiện trên Fig.4A.

Khi CU/PU hiện thời được áp dụng trong chế độ hợp nhất afin (chế độ AF_MERGE), thì nó nhận khối thứ nhất được mã hóa bằng chế độ afin từ các khối được tái tạo lân cận hợp lệ của A, B, C, D, và E. Thứ tự chọn cho khối ứng viên là từ trái, trên,

phải trên, trái dưới cùng đến trái trên như được thể hiện trên Fig.4A. Ví dụ, nếu khối lân cận trái dưới cùng A được mã hóa ở chế độ affin như được thể hiện trên Fig.4B, thì các vector chuyển động v_2 , v_3 và v_4 của góc trái trên cùng, góc phải trên và góc trái dưới cùng của CU/PU chứa khối A được suy ra. Vector chuyển động v_0 của góc trái trên cùng trên CU/PU hiện thời được tính toán theo v_2 , v_3 và v_4 . Tương tự, vector chuyển động v_1 của góc phải trên của CU/PU hiện thời được tính toán dựa trên v_2 , v_3 , và v_4 .

Sau khi CPMV của CU/PU hiện thời v_0 và v_1 được tính toán, theo mô hình chuyển động affin rút gọn được xác định trong phương trình (2) trên đây, MVF của CU/PU hiện thời được tạo ra. Sau khi, MCP affin được áp dụng. Để nhận dạng xem CU/PU hiện thời có được mã hóa bằng chế độ AF_MERGE hay không, cờ affin được báo hiệu trong dòng bit khi có ít nhất một khối lân cận được mã hóa ở chế độ affin. Nếu không có khối affin gần khối hiện thời như được thể hiện trên Fig.4A, thì không có cờ affin được ghi trong dòng bit.

Trong HEVC, mã hóa số học nhị phân thích ứng theo ngữ cảnh (context adaptive binary arithmetic coding-CABAC) bao gồm quy trình nhị phân hóa được dùng để chuyển đổi ký hiệu thành giá trị nhị phân. Việc nhị phân hóa cho phép mã hóa số học nhị phân hiệu quả thông qua việc ánh xạ duy nhất các phần tử cú pháp không nhị phân đến chuỗi các bit, chúng được gọi là các bin. Trong phần mềm tham chiếu JEM2.0, đối với chế độ hợp nhất affin, chỉ cờ affin được mã hóa, và chỉ số hợp nhất được suy luận là mô hình affin lân cận có sẵn thứ nhất theo thứ tự kiểm tra định trước $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$. Đối với chế độ liên cấu trúc affin, hai cú pháp MVD được mã hóa cho mỗi danh sách dự đoán, biểu thị chênh lệch vector chuyển động giữa vector chuyển động affin được suy ra và vector chuyển động được dự đoán.

Fig.5 là sơ đồ khái niệm minh họa ví dụ về mô hình affin với sáu tham số (ba vector chuyển động). Trong Zou et al., "AFFINE MOTION PREDICTION FOR VIDEO CODING", Đơn sáng chế Mỹ số 15/587,044, nộp ngày 4 tháng 5 năm 2017, sơ đồ dự đoán chuyển động affin ngắt được đã được mô tả. Khối với dự đoán affin có thể sử dụng mô hình affin bốn tham số hoặc mô hình affin sáu tham số một cách thích hợp. Mô hình affin với sáu tham số có thể được xác định như sau:

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

Mô hình afin với 6 tham số có ba điểm điều khiển. Nói cách khác, mô hình afin với sáu tham số được xác định bởi ba vectơ chuyển động (MV0, MV1, và MV2), ví dụ, như được thể hiện trên Fig.5. MV0 là vectơ chuyển động điểm điều khiển thứ nhất tại góc trái trên cùng của khối hiện thời, MV1 là vectơ chuyển động điểm điều khiển thứ hai tại góc phải trên của khối hiện thời, và MV2 là vectơ chuyển động điểm điều khiển thứ ba tại góc trái dưới cùng của khối hiện thời, như được thể hiện trên Fig.5. Mô hình afin được tạo ra bởi ba vectơ chuyển động được tính toán như sau:

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

Phương trình (4) trên đây là cho khối vuông có các cạnh bằng w . Với khối không vuông (ví dụ, khối hình chữ nhật) có chiều rộng của w và chiều cao h , mô hình afin sau đây có thể được sử dụng:

$$\begin{cases} mv_x = \frac{(mv_{1x}-mv_{0x})}{w}x + \frac{(mv_{2x}-mv_{0x})}{h}y + mv_{0x} \\ mv_y = \frac{(mv_{1y}-mv_{0y})}{w}x + \frac{(mv_{2y}-mv_{0y})}{h}y + mv_{0y} \end{cases} \quad (5)$$

Cách tương tự như afin hợp nhất để suy ra các vectơ chuyển động thuộc góc trái trên cùng và góc phải trên như được mô tả liên quan đến Fig.4 trên đây có thể cũng được sử dụng để suy ra các MVP cho góc trái trên cùng, góc phải trên và góc trái dưới. Các ví dụ bổ sung được mô tả trong Chen et al., “MOTION VECTOR PREDICTION FOR AFFINE MOTION MODEL”, Đơn tạm thời Mỹ số 62/404,719, nộp ngày 5 tháng 10 năm 2016.

Fig.6 là sơ đồ khối minh họa hệ thống mã hóa và giải mã video 10 làm ví dụ mà có thể sử dụng kỹ thuật theo sáng chế để mã hóa hiệu quả thông tin chuyển động cho dự đoán afin. Như được thể hiện trên Fig.6, hệ thống 10 bao gồm thiết bị nguồn 12 cung cấp dữ liệu video mã hóa sẽ giải mã sau đó bởi 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 bao gồm bất kỳ trong số rất nhiều thiết bị, bao gồm máy tính để bàn, máy tính notebook (tức là, máy tính xách tay), máy tính bảng, hộp giải

mã tín hiệu, các máy điện thoại cầm tay như điện thoại “thông minh”, điện thoại bảng “thông minh”, ti vi, camera, thiết bị hiển thị, thiết bị phát phương tiện số, bàn giao tiếp chơi trò chơi video, thiết bị phát trực tiếp video, hoặc thiết bị tương tự. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị để truyền thông không dây.

Thiết bị đích 14 có thể thu 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. Theo một ví dụ, phương tiện đọc được bằng máy tính 16 có thể bao gồm phương tiện truyền thông để cho phép thiết bị nguồn 12 truyền dữ liệu video mã hóa trực tiếp cho thiết bị đích 14 theo thời gian thực. Dữ liệu video mã hoá có thể được điều chế theo chuẩn truyền thông, như giao thức truyền thông không dây, và được truyền cho thiết bị đích 14. Phương tiện truyền thông có thể là phương tiện truyền thông không dây hoặc có dây bất kỳ, 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 của mạng dựa vào gói, như mạng cục bộ, mạng diện rộng, hoặc mạng toàn cầu như Internet. Phương tiện truyền thông có thể bao gồm bộ định tuyến, bộ chuyển mạch, trạm gốc, hoặc mọi thiết bị khác có thể hữu dụng để hỗ trợ truyền thông từ thiết bị nguồn 12 đến thiết bị đích 14.

Theo một số ví dụ, dữ liệu đã mã hóa có thể được xuất ra từ giao diện xuất 22 đến 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 nhập theo cách phân tán hoặc cục bộ như ổ cứng, đĩa Blu-ray, đĩa số đa năng (DVD - Digital Versatile Disc), đĩa compac-bộ nhớ chỉ đọc (CD-ROM - Compact Disc-Read Only Memory), bộ nhớ flash, 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 bởi 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. Máy chủ cung cấp tập tin làm ví

dự bao gồm máy chủ web (ví dụ, đối với website), máy chủ FTP, thiết bị lưu trữ gắn vào mạng (Network attached storage -NAS), hoặc ổ đĩa cục bộ. Thiết bị đích 14 có thể truy cập dữ liệu video 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 sự kết hợp của 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 sự kết hợp của cả hai loại này.

Các kỹ thuật theo sáng chế không nhất thiết bị giới hạn ở các ứng dụng hoặc thiết lập không dây. Các kỹ thuật này có thể được áp dụng cho việc mã hóa video để hỗ trợ cho ứng dụng bất kỳ trong số nhiều ứng dụng đa phương tiện, như phát quảng bá truyền hình qua không trung, truyền hình cáp, truyền hình vệ tinh, truyền trực tiếp video qua Internet, như truyền trực tiếp thích ứng động trên HTTP (dynamic adaptive streaming over HTTP - DASH), video số được mã hóa lên phương tiện lưu trữ dữ liệu, giải mã video số được lưu trữ trên phương tiện lưu trữ dữ liệu, hoặc các ứng dụng khác. Theo một số ví dụ, hệ thống 10 có thể được tạo cấu hình để hỗ trợ truyền video một chiều hoặc hai chiều nhằm hỗ trợ các ứng dụng như truyền trực tiếp video, phát lại video, phát quảng bá video, và/hoặc điện thoại có truyền hình.

Theo ví dụ trên Fig.6, thiết bị nguồn 12 bao gồm nguồn video 18, bộ mã hóa video 20, và giao diện xuất 22. Thiết bị đích 14 bao gồm giao diện nhập 28, bộ giải mã video 30 và thiết bị hiển thị 32. Theo sáng chế, bộ mã hóa video 20 của thiết bị nguồn 12 có thể được tạo cấu hình để áp dụng các kỹ thuật này để mã hóa có hiệu quả thông tin chuyển động cho dự đoán afin. Theo các ví dụ khác, thiết bị nguồn và thiết bị đích có thể bao gồm các thành phần hoặc thiết bị khác. Ví dụ, thiết bị nguồn 12 có thể thu dữ liệu video từ nguồn video bên ngoài 18, như camera bên ngoài. Tương tự, thiết bị đích 14 có thể giao tiếp với thiết bị hiển thị bên ngoài, chứ không bao gồm cả thiết bị hiển thị tích hợp.

Hệ thống 10 được minh họa trên Fig.6 chỉ là một ví dụ. Các kỹ thuật để mã hóa có hiệu quả thông tin chuyển động cho dự đoán afin có thể được thực hiện bởi thiết bị mã hóa và/hoặc giải mã video số bất kỳ. Mặc dù nói chung các kỹ thuật theo sáng chế được thực hiện bởi thiết bị mã hóa video, nhưng các kỹ thuật này cũng có thể được thực hiện bởi bộ

mã hóa/bộ giải mã video, thường được gọi là “bộ mã hóa-giải mã (CODEC - bộ mã hóa/decoder)”. Hơn nữa, các kỹ thuật theo sáng chế cũng có thể được thực hiện bởi bộ tiền xử lý video. Thiết bị nguồn 12 và thiết bị đích 14 chỉ là các ví dụ của 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, các thiết bị 12, 14 có thể hoạt động theo cách về cơ bản là đối xứng sao cho mỗi thiết bị 12, 14 bao gồm các thành phần mã hóa và giải mã video. Do vậy, hệ thống 10 có thể hỗ trợ truyền video một chiều hoặc hai chiều giữa các thiết bị video 12, 14, ví dụ, để truyền trực tiếp video, phát lại video, phát quảng bá video, và/hoặc điện thoại truyền hình.

Nguồn video 18 của thiết bị nguồn 12 có thể bao gồm thiết bị thu video, như máy quay video, kho video chứa video đã thu trước đó, và/hoặc giao diện nạp video để nhận video từ nhà cung cấp nội dung video. Theo một cách khác nữa, nguồn video 18 có thể tạo ra dữ liệu dựa vào đồ họa máy tính làm video nguồn, hoặc sự kết hợp của video thu phát trực tiếp, video lưu trữ, và video được tạo ra trên máy tính. Trong một số trường hợp, nếu nguồn video 18 là máy quay video, thì thiết bị nguồn 12 và thiết bị đích 14 có thể tạo thành máy điện thoại có camera hoặc điện thoại có hình ảnh. Tuy nhiên, như đã nêu trên, các kỹ thuật được mô tả theo sáng chế có thể áp dụng cho quy trình mã hóa video nói chung và có thể được áp dụng cho các ứng dụng không dây và/hoặc có dây. Trong mỗi trường hợp, video được thu, thu trước hoặc được tạo ra bằng máy tính có thể được mã hóa bằng bộ mã hóa video 20. Sau đó, thông tin video mã hóa có thể được xuất ra bởi giao diện xuất 22 lên phương tiện đọc được bằng máy tính 16.

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

vậy, phương tiện đọc được bằng máy tính 16 có thể được hiểu bao gồm một hoặc nhiều phương tiện đọc được bằng máy tính có các dạng khác nhau, theo một số ví dụ.

Giao diện nhập 28 của thiết bị đích 14 nhận 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 bằng bộ mã hóa video 20, mà cũng được bộ giải mã video 30 sử dụng, bao gồm phần tử cú pháp mô tả các đặc điểm và/hoặc việc xử lý các khối và các đơn vị mã hóa khác. 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.

Bộ mã hóa video 20 và bộ giải mã video 30 có thể hoạt động theo chuẩn mã hóa video, như chuẩn mã hóa video hiệu suất cao (HEVC), còn được gọi là ITU-T H.265. Theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể hoạt động theo các tiêu chuẩn độc quyền hoặc công nghiệp khác, như tiêu chuẩn ITU-T H.264, còn được gọi là MPEG-4, Phần 10, mã hóa video cải tiến (AVC), hoặc các phiên bản mở rộng của các chuẩn này. Tuy nhiên, các kỹ thuật theo sáng chế không bị giới hạn ở bất kỳ chuẩn mã hóa cụ thể nào. Ví dụ khác về chuẩn mã hóa video bao gồm MPEG-2 và ITU-T H.263. Mặc dù không được thể hiện trên Fig.6, nhưng theo một số khía cạnh, mỗi bộ mã hóa video 20 và bộ giải mã video 30 có thể được tích hợp với bộ mã hóa và bộ giải mã âm thanh, và có thể bao gồm các bộ ghép kênh-giải ghép kênh (MUX-DEMUX) thích hợp, hoặc phần cứng và phần mềm khác, để xử lý mã hóa cả âm thanh và video trong dòng dữ liệu chung hoặc các dòng dữ liệu riêng. Nếu áp dụng được, các bộ MUX-DEMUX có thể phù hợp với giao thức ghép kênh ITU H.223, hoặc giao thức khác như giao thức gói dữ liệu người dùng (User datagram protocol-UDP).

Mỗi bộ mã hóa video 20 và bộ giải mã video 30 có thể được triển khai dưới dạng sơ đồ mạch bất kỳ trong nhiều sơ đồ mạch mã hóa thích hợp, 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 circuits-ASIC), mảng cổng lập trình được theo trường (Field programmable gate arrays- FPGAs), logic rời rạc, phần mềm, phần cứng, firmware

hoặc các kết hợp bất kỳ của chúng. Khi kỹ thuật này được thực thi 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 sử dụng một hoặc nhiều bộ xử lý để thực hiện kỹ thuật theo sáng chế. Mỗi trong số bộ mã hóa video 20 và bộ giải mã video 30 có thể nằm trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, mỗi bộ này có thể được tích hợp làm một phần của bộ mã hóa/bộ giải mã (encoder/decoder - CODEC) kết hợp trong thiết bị tương ứng.

Nói chung, theo ITU-T H.265, hình ảnh video có thể được chia thành chuỗi các đơn vị cây mã hóa (CTU) (hoặc đơn vị mã hóa lớn nhất (largest coding unit - LCUs)) mà có thể bao gồm cả các mẫu độ màu và các mẫu độ chói. Theo cách khác, các CTU có thể bao gồm dữ liệu đơn sắc (tức là, chỉ các mẫu độ chói). Dữ liệu cú pháp trong dòng bit có thể xác định kích thước cho CTU, là đơn vị mã hóa lớn nhất về số lượng điểm ảnh. Lát bao gồm một số CTU liên tiếp theo thứ tự mã hóa. Hình ảnh video có thể được phân chia thành một hoặc nhiều lát. Mỗi CTU có thể được tách thành các đơn vị mã hóa (CU) theo cây tứ phân. Nói chung, cấu trúc dữ liệu cây tứ phân bao gồm một nút cho mỗi CU, với nút gốc tương ứng với CTU. Nếu CU được phân chia thành bốn CU con, thì nút tương ứng với CU có bốn nút lá, mỗi nút lá tương ứng với một trong các CU con.

Mỗi nút của cấu trúc dữ liệu cây tứ phân có thể cung cấp dữ liệu cú pháp cho CU tương ứng. Ví dụ, nút trên cây tứ phân có thể bao gồm cờ phân tách, chỉ báo liệu CU tương ứng với nút có được tách thành các CU con không. Các phần tử cú pháp của CU có thể được định nghĩa theo cách đệ quy, và có thể tùy thuộc vào việc liệu CU có được tách thành các CU con không. Nếu CU không được tách nữa, thì nó sẽ được gọi là CU lá. Trong bản mô tả này, bốn CU con của CU lá sẽ cũng được gọi là CU lá ngay cả khi không có sự phân tách rõ ràng CU lá ban đầu. Ví dụ, nếu CU với kích thước 16x16 không được tách thêm, thì bốn CU con 8x8 cũng sẽ được gọi là CU lá mặc dù CU 16x16 chưa được tách lần nào.

CU có mục đích tương tự với khối macro của chuẩn H.264, ngoại trừ CU không có sự khác biệt về kích thước. Ví dụ, CTU có thể được tách thành bốn nút con (còn được gọi là CU con), và mỗi nút con đến lượt nó có thể là nút cha và được tách thành bốn nút con khác. Nút con cuối cùng, không tách, được gọi là nút lá của cây tứ phân, bao gồm nút mã hóa, cũng được gọi là CU lá. Dữ liệu cú pháp liên quan đến dòng bit mã hóa có thể xác

định số lần tối đa mà CTU có thể được tách, được gọi là chiều sâu CU tối đa và cũng có thể xác định kích thước tối thiểu của các nút mã hóa. Do đó, dòng bit cũng có thể xác định đơn vị mã hóa nhỏ nhất (smallest coding unit - SCU). Sáng chế sử dụng thuật ngữ “khối” để chỉ đơn vị bất kỳ trong số các đơn vị CU, đơn vị dự đoán (PU) hoặc đơn vị biến đổi (TU), trong ngữ cảnh của chuẩn HEVC, hoặc các cấu trúc dữ liệu tương tự trong ngữ cảnh của các chuẩn khác (ví dụ, khối macro và khối con của khối macro theo chuẩn H.264/AVC).

CU bao gồm nút mã hóa và các đơn vị dự đoán (PU) và đơn vị biến đổi (TU) liên quan đến nút mã hóa này. Kích thước của CU tương ứng với kích thước của nút mã hóa và thường có dạng hình vuông. Kích thước của CU có thể nằm trong khoảng từ 8x8 điểm ảnh lên đến kích thước của CTU với kích thước tối đa, ví dụ, 64x64 điểm ảnh hoặc lớn hơn. Mỗi CU có thể chứa một hoặc nhiều PU và một hoặc nhiều TU. Dữ liệu cú pháp liên quan đến CU có thể mô tả, ví dụ, phép phân chia CU thành một hoặc nhiều PU. Các chế độ phân chia có thể khác nhau giữa việc CU được bỏ qua hay là được mã hóa ở chế độ trực tiếp, được mã hóa ở chế độ dự đoán trong hình ảnh, hay được mã hóa ở chế độ dự đoán liên hình ảnh. Các PU có thể được phân chia thành dạng không phải hình vuông. Dữ liệu cú pháp liên quan đến CU cũng có thể mô tả, ví dụ, việc phân chia CU thành một hoặc nhiều TU theo cấu trúc cây tứ phân. TU có thể có dạng hình vuông hoặc không phải hình vuông (ví dụ, hình chữ nhật).

Chuẩn HEVC cho phép các biến đổi theo các TU, có thể là khác nhau đối với các CU khác nhau. Các TU thường có kích thước dựa vào kích thước của các PU (hoặc các phần chia của CU) trong một CU nhất định được xác định cho CTU đã phân chia, tuy nhiên có thể không phải lúc nào cũng đúng như vậy. Các TU thường có cùng kích thước hoặc nhỏ hơn các PU (hoặc các phần chia của CU, ví dụ, trong trường hợp dự đoán trong hình ảnh). Theo một số ví dụ, các mẫu dư tương ứng với CU có thể được chia nhỏ thành các đơn vị nhỏ hơn bằng cách sử dụng cấu trúc cây tứ phân gọi là “cây tứ phân dư” (residual quad tree - RQT). Các nút lá của RQT có thể được gọi là các đơn vị biến đổi (Transform Unit - TU). Các giá trị vi sai điểm ảnh liên quan đến các TU có thể được biến đổi để tạo ra các hệ số biến đổi, các hệ số biến đổi này có thể được lượng tử hoá.

CU lá có thể bao gồm một hoặc nhiều đơn vị dự đoán (prediction unit - PU) khi được dự đoán bằng cách sử dụng dự đoán liên hình ảnh. Nói chung, PU biểu diễn vùng không gian tương ứng với toàn bộ hoặc một phần của CU tương ứng, và có thể chứa dữ liệu để truy hồi và/hoặc tạo ra mẫu tham chiếu cho PU. Ngoài ra, PU chứa dữ liệu liên quan đến việc dự đoán. Khi CU được mã hóa ở chế độ liên cấu trúc, một hoặc nhiều PU của CU có thể bao gồm dữ liệu xác định thông tin chuyển động, như một hoặc nhiều vector chuyển động, hoặc các PU có thể được mã hóa ở chế độ bỏ qua. Dữ liệu xác định vector chuyển động cho PU có thể mô tả, ví dụ, thành phần ngang của vector chuyển động, thành phần dọc của vector chuyển động, độ phân giải của vector chuyển động (ví dụ, độ chính xác điểm ảnh một phần tư hoặc độ chính xác điểm ảnh một phần tám), hình ảnh tham chiếu mà vector chuyển động trỏ tới, và/hoặc danh sách hình ảnh tham chiếu (ví dụ, danh sách 0, danh sách 1) cho vector chuyển động.

Các CU lá cũng có thể được dự đoán ở chế độ nội cấu trúc. Nói chung, dự đoán trong hình ảnh bao gồm dự đoán CU lá (hoặc các phần chia của chúng) bằng cách sử dụng chế độ nội cấu trúc. Bộ mã hóa video có thể chọn tập hợp các điểm ảnh lân cận được mã hóa trước đó để CU lá sử dụng để dự đoán CU lá (hoặc các phần chia của chúng).

CU lá cũng có thể bao gồm một hoặc nhiều đơn vị biến đổi (transform unit - TU). Các đơn vị biến đổi có thể được xác định bằng cách sử dụng RQT (còn được gọi là cấu trúc cây tứ phân TU), như nêu trên. Ví dụ, cờ phân tách có thể chỉ báo liệu CU lá có được phân tách thành bốn đơn vị biến đổi hay không. Sau đó, mỗi TU có thể được tách tiếp thành các TU con khác. Khi TU không được tách nữa, thì nó có thể được gọi là TU lá. Nói chung, để mã hóa nội hình ảnh, tất cả các TU lá thuộc CU lá dùng chung cùng chế độ dự đoán trong hình ảnh. Tức là, cùng một chế độ dự đoán trong hình ảnh thường được áp dụng để tính các giá trị được dự đoán cho tất cả TU của CU lá. Để mã hóa nội hình ảnh, bộ mã hóa video 20 có thể tính giá trị dư cho mỗi TU lá bằng cách sử dụng chế độ dự đoán trong hình ảnh, là vi sai giữa phần của CU tương ứng với TU và khối ban đầu. TU không nhất thiết bị giới hạn ở kích cỡ của PU. Do đó, các TU có thể lớn hơn hoặc nhỏ hơn PU. Để mã hóa nội hình ảnh, các phần chia của CU, hoặc bản thân CU, có thể được đặt cùng vị trí với TU lá tương ứng cho CU này. Theo một số ví dụ, kích cỡ lớn nhất của TU lá có thể tương đương với kích cỡ của CU lá tương ứng.

Ngoài ra, các TU của CU lá cũng có thể được gắn với các cấu trúc dữ liệu cây tứ phân tương ứng, được gọi là các cây tứ phân dư (residual quadtree - RQT). Tức là, CU lá có thể bao gồm cây tứ phân biểu thị cách thức CU lá được phân chia thành các TU. Nút gốc của cây tứ phân TU thường tương ứng với CU lá, trong khi nút gốc của cây tứ phân CU thường tương ứng với CTU (hoặc LCU). TU của RQT mà không được tách được gọi là TU lá. Nói chung, bản mô tả này sử dụng các thuật ngữ CU và TU để lần lượt chỉ CU lá và TU lá, trừ khi được quy định khác.

Chuỗi video thường bao gồm dãy các hình hoặc khung video, bắt đầu với hình điểm truy cập ngẫu nhiên (random access point - RAP). Chuỗi video có thể bao gồm dữ liệu cú pháp trong tập tham số chuỗi (sequence parameter set - SPS) mô tả đặc tính của chuỗi video. Mỗi lát của hình có thể chứa dữ liệu cú pháp lát để mô tả chế độ mã hóa cho lát tương ứng. Bộ mã hóa video 20 thường hoạt động trên các khối video trong các lát video riêng lẻ để mã hóa dữ liệu video. Khối video có thể tương ứng với nút mã hóa trong CU. Các khối video có thể có kích thước cố định hoặc thay đổi, và có thể khác nhau về kích thước theo chuẩn mã hóa được quy định.

Ví dụ, kỹ thuật dự đoán có thể được thực hiện cho các PU có các kích thước khác nhau. Giả định rằng kích thước của CU cụ thể là $2N \times 2N$, dự đoán trong hình ảnh có thể được thực hiện trên các kích thước PU là $2N \times 2N$ hoặc $N \times N$, và dự đoán liên hình ảnh có thể được thực hiện trên các kích thước PU đối xứng là $2N \times 2N$, $2N \times N$, $N \times 2N$, hoặc $N \times N$. Phân chia không đối xứng cho dự đoán liên hình ảnh cũng có thể được thực hiện cho các kích thước PU là $2N \times nU$, $2N \times nD$, $nL \times 2N$, và $nR \times 2N$. Trong phân chia không đối xứng, một chiều của CU không được phân chia, trong khi chiều còn lại được phân chia thành 25% và 75%. Phần của CU tương ứng với phân chia 25% được biểu thị bằng “n” tiếp đó là chỉ báo “Lên”, “Xuống”, “Trái” hoặc “Phải”. Do vậy, ví dụ, “ $2N \times nU$ ” chỉ CU $2N \times 2N$ được phân chia theo chiều ngang với PU $2N \times 0,5N$ ở trên và PU $2N \times 1,5$ ở dưới.

Trong bản mô tả này, “ $N \times N$ ” và “N nhân N” có thể được sử dụng thay thế nhau để chỉ kích thước điểm ảnh của khối video theo chiều dọc và chiều ngang, ví dụ, 16×16 điểm ảnh hoặc 16 nhân 16 điểm ảnh. Nói chung, khối 16×16 sẽ có 16 điểm ảnh theo chiều dọc ($y = 16$) và 16 điểm ảnh theo chiều ngang ($x = 16$). Tương tự, khối $N \times N$ thường có N điểm ảnh theo chiều dọc và N điểm ảnh theo chiều ngang, trong đó N là giá trị nguyên không

âm. Các điểm ảnh trong một khối có thể được sắp xếp thành các hàng và các cột. Hơn nữa, các khối không nhất thiết phải có số lượng điểm ảnh theo chiều ngang bằng số lượng điểm ảnh theo chiều dọc. Ví dụ, các khối có thể có $N \times M$ điểm ảnh, trong đó M không nhất thiết bằng N .

Sau khi mã hóa dự đoán trong hình ảnh hoặc liên hình ảnh sử dụng các PU của CU, bộ mã hóa video 20 có thể tính toán dữ liệu dư cho các TU của CU. Các PU có thể bao gồm dữ liệu cú pháp mô tả phương pháp hoặc chế độ tạo ra dữ liệu điểm ảnh dự đoán trong miền không gian (còn được gọi là miền điểm ảnh) và các TU có thể bao gồm các hệ số trong miền biến đổi sau khi áp dụng kỹ thuật biến đổi, ví dụ, biến đổi cosin rời rạc (DCT - discrete cosine transform), biến đổi số nguyên, biến đổi sóng nhỏ hoặc kỹ thuật biến đổi tương tự về mặt khái niệm cho dữ liệu video dư. Dữ liệu dư có thể tương ứng với vi sai điểm ảnh giữa các điểm ảnh của hình không mã hóa và các giá trị dự đoán tương ứng với các PU. Bộ mã hóa video 20 có thể tạo ra các TU để chứa các hệ số biến đổi được lượng tử hóa biểu diễn dữ liệu dư cho CU. Tức là, bộ mã hóa video 20 có thể tính dữ liệu dư (dưới dạng khối dữ liệu dư), biến đổi khối dữ liệu dư để tạo ra khối hệ số biến đổi, và sau đó lượng tử hóa các hệ số biến đổi để tạo ra các hệ số biến đổi lượng tử hóa. Bộ mã hóa video 20 có thể tạo ra TU chứa các hệ số biến đổi lượng tử hóa, cũng như thông tin cú pháp khác (ví dụ, thông tin tách đối với TU).

Như nêu trên, sau phép biến đổi bất kỳ để tạo ra các hệ số biến đổi, bộ mã hóa video 20 có thể thực hiện việc lượng tử hóa các hệ số biến đổi này. Nói chung, lượng tử hóa chỉ quy trình trong đó các hệ số biến đổi được lượng tử hóa để có thể giảm lượng dữ liệu dùng để biểu diễn các hệ số, cho phép nén thêm nữa. Quá trình lượng tử hóa có thể làm giảm độ sâu bit liên quan tới một số hoặc tất cả các hệ số này. Ví dụ, giá trị n bit có thể được làm tròn xuống giá trị m bit trong quá trình lượng tử hóa, trong đó n lớn hơn m .

Sau khi lượng tử hóa, bộ mã hóa video có thể quét các hệ số biến đổi, tạo ra một vector một chiều từ ma trận hai chiều bao gồm các hệ số biến đổi lượng tử hóa. Quy trình quét được thiết kế để đặt các hệ số năng lượng cao hơn (và do đó tần số thấp hơn) ở phía trước mảng và đặt hệ số năng lượng thấp hơn (và do đó tần số cao hơn) ở phía sau mảng. Theo một số ví dụ, bộ mã hóa video 20 có thể sử dụng thứ tự quét định trước để quét các hệ số biến đổi lượng tử hóa để tạo ra vector nối tiếp có thể được mã hóa entropy. Trong các

ví dụ khác, bộ mã hóa video 20 có thể thực hiện quét thích ứng. Sau khi quét hệ số biến đổi lượng tử hóa để tạo ra vector một chiều, bộ mã hóa video 20 có thể mã hóa entropy vector một chiều, ví dụ, theo phương thức mã hóa độ dài thay đổi tương thích với ngữ cảnh (context-adaptive variable length coding-CAVLC), mã hóa số học nhị phân tương thích với ngữ cảnh (context-adaptive binary arithmetic coding-CABAC), mã hóa số học nhị phân tương thích với ngữ cảnh dựa trên cú pháp (syntax-based context-adaptive binary arithmetic coding-SBAC), mã hóa entropy phân chia khoảng xác suất (Probability Interval Partitioning Entropy-PIPE) hoặc các phương thức mã hóa entropy khác. Bộ mã hóa video 20 cũng có thể mã hóa entropy các phần tử cú pháp liên quan tới dữ liệu video mã hóa để bộ giải mã video 30 sử dụng trong việc giải mã dữ liệu video.

Để thực hiện CABAC, bộ mã hóa video 20 có thể gán ngữ cảnh trong mô hình ngữ cảnh cho ký hiệu cần được truyền. Ngữ cảnh có thể liên quan tới, ví dụ, việc các giá trị lân cận của ký hiệu có khác 0 hay không. Để thực hiện CAVLC, bộ mã hóa video 20 có thể chọn mã độ dài biến đổi cho ký hiệu cần được truyền. Các từ mã trong VLC có thể được cấu tạo sao cho các mã tương đối ngắn hơn tương ứng với các ký hiệu xác suất cao hơn, trong khi đó các mã dài hơn tương ứng với các ký hiệu có xác suất thấp hơn. Theo cách này, việc sử dụng VLC có thể tiết kiệm bit so với, ví dụ, sử dụng các từ mã độ dài bằng nhau cho mỗi ký hiệu cần được truyền. Việc xác định xác suất có thể dựa vào ngữ cảnh được gán cho ký hiệu.

Nói chung, bộ giải mã video 30 thực hiện quy trình gần như tương tự, tuy là nghịch đảo, với quy trình thực hiện bởi bộ mã hóa video 20 để giải mã dữ liệu mã hóa. Ví dụ, bộ giải mã video 30 lượng tử hóa ngược và biến đổi ngược các hệ số của TU thu được để tái tạo khối dữ liệu dư. Bộ giải mã video 30 sử dụng chế độ dự đoán đã báo hiệu (dự đoán trong hình ảnh hoặc liên hình ảnh) để tạo ra khối dự đoán. Sau đó bộ giải mã video 30 kết hợp khối dự đoán và khối dữ liệu dư (theo từng điểm ảnh một) để tái tạo khối ban đầu. Việc xử lý bổ sung có thể được thực hiện, như thực hiện quy trình tách khối để giảm các thành phần lạ thấy được dọc theo các biên khối. Ngoài ra, bộ giải mã video 30 có thể giải mã các phần tử cú pháp sử dụng quy trình CABAC theo cách gần như tương tự, tuy là nghịch đảo, với quy trình mã hóa CABAC của bộ mã hóa video 20.

Nói chung, bộ mã hóa video 20 và bộ giải mã video 30 có thể được tạo cấu hình để mã hóa hiệu quả hơn (mã hóa hoặc giải mã, tương ứng) thông tin chuyển động cho việc dự đoán afin, theo các kỹ thuật của sáng chế. Bộ mã hóa video 20 và bộ giải mã video 30 có thể được tạo cấu hình để áp dụng kỹ thuật bất kỳ trong số các kỹ thuật khác nhau được mô tả dưới đây, riêng lẻ hoặc theo dạng kết hợp bất kỳ.

Theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể dùng chênh lệch vector chuyển động (MVD) của một vector chuyển động (MV) để dự đoán MVD của một MV khác trong khối được dự đoán bằng dự đoán afin. MVD có thể được xác định là chênh lệch giữa MV và dự đoán vector chuyển động (MVP): $MVD = MV - MVP$. Cụ thể hơn, nếu vector chuyển động (MV_x, MV_y) được biểu thị bằng thành phần ngang (MV_x) và thành phần dọc (MV_y) của nó, và bộ dự đoán vector chuyển động có các thành phần (MVP_x, MVP_y) thì thành phần ngang (dọc) của MVD được xác định lần lượt là chênh lệch của các thành phần ngang (dọc) của MV và MVP. Do đó, MVD có thể được xác định bằng (MVD_x, MVD_y), trong đó $MVD_x = MV_x - MVP_x$, và $MVD_y = MV_y - MVP_y$.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dùng MVD của MV thứ nhất để dự đoán (các) MVD của một hoặc nhiều MV khác trong dự đoán afin. Fig.9 là sơ đồ khái niệm minh họa ví dụ về dự đoán MVD này. Theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để sử dụng MVD của MV thứ nhất để dự đoán MVD của MV thứ hai trong dự đoán afin (ví dụ, afin 4 tham số). Fig.9 dưới đây thể hiện ví dụ về dự đoán MVD cho dự đoán afin với hai vector chuyển động, trong đó MVD1 được dự đoán bởi MVD0.

Fig.10 là sơ đồ khái niệm minh họa ví dụ về dự đoán MVD cho dự đoán afin với ba vector chuyển động (dự đoán afin sáu tham số). Với dự đoán afin sáu tham số, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể dùng MVD của MV thứ nhất để dự đoán MVD của MV thứ hai. Hơn nữa, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể dùng MVD của MV thứ nhất để dự đoán MVD của MV thứ ba trong dự đoán afin với ba vector chuyển động. Fig.10 thể hiện ví dụ về dự đoán MVD cho dự đoán afin với ba vector chuyển động, trong đó MVD1 được dự đoán bởi MVD0 và MVD2 cũng được dự đoán bởi MVD0.

Trở lại Fig.6, theo một số ví dụ, bộ mã hóa video 20 và bộ giải mã video 30 có thể được tạo cấu hình sao cho MV thứ nhất trong các ví dụ trên đây được xác định là MV liên quan đến điểm điều khiển trên cùng bên trái biểu thị “MV0” trong các Fig.3, 9, và 10. Theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể được tạo cấu hình để lựa chọn một cách thích ứng điểm điều khiển thứ nhất liên quan đến MV thứ nhất. Ví dụ, điểm điều khiển thứ nhất có thể phụ thuộc vào thông tin được mã hóa, như là hình dạng khối. Theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể ngầm suy ra điểm điều khiển thứ nhất liên quan đến MV thứ nhất.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để áp dụng dự đoán MVD giữa hai MVD bất kỳ cho dự đoán afin. Ví dụ, bộ mã hóa video 20 và bộ giải mã video 30 có thể dự đoán MVD0 từ MVD1 cho dự đoán afin với hai vector chuyển động. Theo ví dụ khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể dự đoán MVD1 từ MVD0, và dự đoán MVD2 từ MVD1, cho dự đoán afin với ba vector chuyển động.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dự đoán MV_a từ MV_b. Bộ mã hóa video 20 có thể tính toán $MV_a' = MV_a - MV_b$, và mã hóa thông tin biểu diễn MV_a' dưới dạng một phần của dòng bit, sao cho bộ giải mã video 30 có thể giải mã thông tin này để xác định MV_a'. Bộ giải mã video 30 sau đó có thể tính toán $MV_a = MV_a' + MV_b$. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán afin với bốn tham số.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dự đoán MV_a từ MV_b. Bộ mã hóa video 20 sau đó có thể tính toán $MV_a' = MV_a - w * MV_b$, và mã hóa thông tin biểu diễn MV_a' dưới dạng một phần của dòng bit, sao cho bộ giải mã video 30 có thể giải mã thông tin này để xác định MV_a'. Bộ giải mã video 30 sau đó có thể tính toán $MV_a = MV_a' + w * MV_b$. Theo ví dụ này, w là giá trị trọng số, như 0,5. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán afin với hai tham số. Ví dụ này có thể được thực hiện dưới dạng số nguyên, như $MVD1' = MVD1 - ((MVD0+1) >> 1)$ khi $w = 0,5$, hoặc $MVD1' = MVD1 - ((MVD0+2) >> 2)$ khi $w = 0,25$. Theo một ví dụ, bộ mã hóa video 20 mã hóa dữ liệu biểu diễn w dưới dạng một phần của dòng bit, ví dụ, tại mức chuỗi (như tập tham số chuỗi

(SPS)), mức hình ảnh (như tập tham số hình ảnh (PPS)), mức lát (như trong phần đầu lát), hoặc mức khối (như trong phần đầu khối). Bộ giải mã video 30 có thể còn trích xuất thông tin báo hiệu này từ mức thông tin tương ứng.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dự đoán MVD của một điểm điều khiển từ các MVD của nhiều hơn một điểm điều khiển khác. Ví dụ, bộ mã hóa video 20 và bộ giải mã video 30 có thể dự đoán MVD2 từ MVD0 và MVD1 trong mô hình afin sáu tham số. Ví dụ, bộ mã hóa video 20 có thể tính toán $MVD2' = MVD2 - ((MVD0 + MVD1) >> 1)$, và mã hóa thông tin biểu diễn MVD2' dưới dạng một phần của dòng bit, sao cho bộ giải mã video 30 có thể giải mã thông tin này để xác định MVD2'. Bộ giải mã video 30 sau đó có thể sử dụng thông tin này để tính toán $MVD2 = MVD2' + ((MVD0 + MVD1) >> 1)$.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để xác định xem có dự đoán MVD dựa vào giá trị MVD hay không. Ví dụ, bộ mã hóa video 20 có thể xác định để dự đoán MVDa từ MVDb nếu $|MVDb^x + MVDb^y| < T$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Theo ví dụ khác, bộ mã hóa video 20 có thể xác định để dự đoán MVDa từ MVDb nếu $\max(|MVDb^x|, |MVDb^y|) < T$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Theo ví dụ khác nữa, bộ mã hóa video 20 có thể dự đoán MVDa từ MVDb nếu $|MVDb^x + MVDb^y| > T$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Theo ví dụ khác nữa, bộ mã hóa video 20 có thể dự đoán MVDa từ MVDb nếu $\min(|MVDb^x|, |MVDb^y|) > T$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Trong các ví dụ trên đây, T biểu diễn ngưỡng, mà có thể là số cố định hoặc được báo hiệu bởi bộ mã hóa video 20 và được giải mã bởi bộ giải mã video 30. Bộ mã hóa video 20 có thể mã hóa dữ liệu biểu diễn xem có dự đoán MVDa từ MVDb hay không dựa vào sự xác định bất kỳ trong số các xác định làm ví dụ trên đây, và bộ giải mã video 30 có thể giải mã dữ liệu mã hóa này để xác định xem có dự đoán MVDa từ MVDb hay không.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dự đoán các thành phần ngang (x) và/hoặc dọc (y) của MVD theo các cách khác nhau. Ví dụ, bộ mã hóa video 20 và bộ giải mã video

30 có thể chỉ dự đoán thành phần x của MVDb từ thành phần x của MVDa, nhưng không dự đoán thành phần y của MVDb từ một MVD khác (ví dụ, MVDa).

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để xác định xem có dự đoán MVD dựa vào giá trị của MVP hay không. Ví dụ, bộ mã hóa video 20 có thể xác định để dự đoán MVDa từ MVDb nếu $|MVPa^x - MVPb^x| + |MVPa^y - MVPb^y| < S$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Theo ví dụ khác, bộ mã hóa video 20 có thể xác định để dự đoán MVDa từ MVDb nếu $\max(|MVPa^x - MVPb^x|, |MVPa^y - MVPb^y|) < S$; nếu không thì bộ mã hóa video 20 không dự đoán MVDa từ MVDb. Trong các ví dụ trên đây, S biểu diễn ngưỡng, mà có thể là số cố định hoặc được báo hiệu bởi bộ mã hóa video 20 và được giải mã bởi bộ giải mã video 30. Bộ mã hóa video 20 có thể mã hóa dữ liệu biểu diễn xem có dự đoán MVDa từ MVDb hay không dựa vào sự xác định bất kỳ trong số các xác định làm ví dụ trên đây, và bộ giải mã video 30 có thể giải mã dữ liệu mã hóa này để xác định xem có dự đoán MVDa từ MVDb hay không.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để xác định xem có dự đoán MVD hay không cho khối được dự đoán bằng cách sử dụng dự đoán afin dựa vào phương pháp dự đoán chuyển động. Ví dụ, nếu MVP đến từ phương pháp suy ra MVP trong JEM như được mô tả trên đây liên quan đến Fig.3, thì bộ mã hóa video 20 và bộ giải mã video 30 có thể xác định không sử dụng dự đoán MVD. Theo ví dụ khác, nếu MVP đến từ phương pháp suy ra MVP tương tự với hợp nhất afin như được mô tả trên đây liên quan đến đơn tạm thời Mỹ số 62/404,719, thì bộ mã hóa video 20 và bộ giải mã video 30 có thể xác định để sử dụng dự đoán MVD. Ngoài ra hoặc theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể xác định xem có dùng dự đoán MVD hay không dựa vào việc xem quy trình bù sáng có được dùng cho khối nguồn của MVP không.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để xác định xem có dự đoán MVD hay không dựa vào kích thước và/hoặc hình dạng của khối hiện thời. Ví dụ, bộ mã hóa video 20 và bộ giải mã video 30 có thể xác định để sử dụng dự đoán MVD khi $W \cdot H > T$, trong đó W là chiều rộng của khối hiện thời, H là chiều cao của khối hiện thời, và T là giá trị ngưỡng.

T có thể là số cố định hoặc được báo hiệu từ bộ mã hóa video 20 đến bộ giải mã video 30 trong dòng bit. Theo ví dụ khác, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể xác định sử dụng dự đoán MVD khi $W \cdot H < T$.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để mã hóa dữ liệu biểu diễn xem có dự đoán MVD (hoặc thành phần của nó) trong dòng bit hay không. Tức là, bộ mã hóa video 20 có thể mã hóa dữ liệu biểu diễn xem có dự đoán một hoặc cả hai thành phần (ngang và dọc) của MVD trong dòng bit hay không, và bộ giải mã video 30 có thể xác định xem có dự đoán một hoặc cả hai thành phần của MVD hay không từ dữ liệu mã hóa của dòng bit (bằng cách giải mã dữ liệu).

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để mã hóa dữ liệu biểu diễn (các) điểm điều khiển nào sẽ dùng làm tham chiếu cho dự đoán MVD trong dòng bit. Tức là, bộ mã hóa video 20 có thể mã hóa dữ liệu này, và bộ giải mã video 30 có thể giải mã dữ liệu này để xác định (các) điểm điều khiển nào sẽ dùng làm tham chiếu cho dự đoán MVD.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tạo ra MVP của MV từ MVD của MV khác trong khối được dự đoán bằng cách sử dụng dự đoán afin.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để dùng MVD của MV thứ nhất để tạo ra (các) MVP của một hoặc nhiều MV khác trong dự đoán afin. Theo một ví dụ, bộ mã hóa video 20 và bộ giải mã video 30 có thể dùng MVD của MV thứ nhất để tạo ra MVP của MV thứ hai trong dự đoán afin (ví dụ, afin bốn tham số). Theo ví dụ khác, với dự đoán afin sáu tham số, bộ mã hóa video 20 và bộ giải mã video 30 có thể dùng MVD của MV thứ nhất để tạo ra MVP của MV thứ hai trong dự đoán afin với ba vector chuyển động. Ngoài ra, bộ mã hóa video 20 và bộ giải mã video 30 có thể dùng MVD của MV thứ nhất để tạo ra MVP của MV thứ ba trong dự đoán afin với ba vector chuyển động. Theo cách khác, bộ mã hóa video 20 và bộ giải mã video 30 có thể dùng MVD của MV thứ hai để tạo ra MVP của MV thứ ba trong dự đoán afin với ba vector chuyển động.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tạo ra MVP0 từ MVD1 cho dự đoán affin với hai vector chuyển động. Theo ví dụ khác, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tạo ra MVP1 từ MVD0, và để tạo ra MVP2 từ MVD1, cho dự đoán affin với ba vector chuyển động.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tính toán MVPa theo $MVPa = MVP'a + MVD_b$. MVP'a là MVP được tạo ra theo cách ban đầu mà không xét đến MVD_b như được mô tả trên đây cho chế độ AMVP và hợp nhất để dự đoán affin. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán affin với bốn tham số (hai vector chuyển động).

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tính toán MVPa theo $MVPa = MVP'a + w * MVD_b$, trong đó w là giá trị trọng số, như 0,5. Theo ví dụ này, MVP'a là MVP được tạo ra theo cách ban đầu mà không xét đến MVD_b như được mô tả trên đây cho chế độ AMVP và hợp nhất để dự đoán affin. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán affin với hai tham số. Ví dụ này có thể được thực hiện dưới dạng số nguyên, như $MVP1 = MVP'1 + ((MVD0+1) >> 1)$ khi $w = 0,5$, hoặc $MVP1 = MVP'1 + ((MVD0+2) >> 2)$ khi $w = 0,25$. Theo một ví dụ, bộ mã hóa video 20 xác định w và báo hiệu giá trị w trong dòng bit tại mức chuỗi, mức hình ảnh, mức lát, hoặc mức khối. Bộ giải mã video 30, theo đó, sẽ giải mã giá trị w từ mức thông tin thích hợp.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tạo ra MVP của một điểm điều khiển từ các MVD của nhiều điểm điều khiển khác. Ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể tạo ra MVP2 từ MVD0 và MVD1 trong mô hình affin sáu tham số. Bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể tính toán MVP2 là $MVP2 = MVP'2 + ((MVD0 + MVD1) >> 1)$.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để sử dụng MV của một điểm điều khiển để tạo ra (các) MVP của (các) MV của một hoặc nhiều điểm điều khiển khác trong khối dự đoán được bằng cách sử dụng dự đoán affin. Theo một ví dụ, bộ mã hóa video 20 và/hoặc bộ giải

mã video 30 có thể dùng MV thứ nhất để tạo ra MVP của MV thứ hai trong dự đoán afin (ví dụ, afin bốn tham số). Theo ví dụ khác, với dự đoán afin sáu tham số, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể dùng MV thứ nhất để tạo ra MVP của MV thứ hai trong dự đoán afin với ba vector chuyển động, và dùng MV thứ nhất để tạo ra MVP của MV thứ ba trong dự đoán afin với ba vector chuyển động. Theo cách khác, với dự đoán afin sáu tham số, bộ mã hóa 20 và/hoặc bộ giải mã video 30 có thể dùng MV thứ hai để tạo ra MVP của MV thứ ba trong dự đoán afin với ba vector chuyển động.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tính toán MVPa là $MVPa = (MVP'a + MVb) \gg 1$. MVP'a là MVP được tạo ra theo cách ban đầu mà không xét đến MVb như được mô tả trên đây trong các đoạn thảo luận về chế độ AMVP và hợp nhất cho dự đoán afin. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán afin với bốn tham số.

Ngoài ra hoặc theo cách khác, theo một số ví dụ, bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể được tạo cấu hình để tính toán MVPa là $MVPa = w1 * MVP'a + w2 * MVb$. Theo ví dụ này, $w1$ và $w2$ là các giá trị trọng số mà có thể có các giá trị giống nhau hoặc khác nhau, ví dụ, $w1 = w2 = 0,5$. MVP'a là MVP được tạo ra theo cách ban đầu mà không xét đến MVb như được mô tả trong các đoạn trên ở chế độ AMVP và hợp nhất cho dự đoán afin. Theo một ví dụ, $a=1$ và $b=0$ cho dự đoán afin với bốn tham số. Ví dụ này có thể được thực hiện dưới dạng số nguyên, như $MVP1 = (3 * MVP'1 + MV0 + 2) \gg 2$ khi $w1 = 0,75$ và $w2 = 0,25$. Theo một ví dụ, bộ mã hóa video 20 mã hóa dữ liệu cho $w1$ và $w2$ trong dòng bit tại mức bất kỳ trong số các mức chuỗi, mức hình ảnh, mức lát, hoặc mức khối. Tương tự, bộ giải mã video 30 sẽ xác định $w1$ và $w2$ bằng cách giải mã dữ liệu này tại mức thích hợp.

Bộ mã hóa video 20 còn có thể truyền dữ liệu cú pháp, như dữ liệu cú pháp dựa trên khối, dữ liệu cú pháp dựa trên hình ảnh, và dữ liệu cú pháp dựa trên chuỗi, đến bộ giải mã video 30, ví dụ, trong phần đầu hình ảnh, phần đầu khối, phần đầu lát, hoặc dữ liệu cú pháp khác, như tập tham số chuỗi (SPS), tập tham số hình ảnh (PPS), hoặc tập tham số video (video parameter set - VPS).

Mỗi bộ mã hóa video 20 và bộ giải mã video 30 có thể được thực thi dưới dạng bất kỳ trong nhiều sơ đồ mạch mã hóa hoặc giải mã thích hợp, khi có thể áp dụng được, như

một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (DSP), mạch tích hợp chuyên dụng (Application specific integrated circuits-ASIC), mảng cổng lập trình được theo trường (Field programmable gate arrays- FPGAs), mạch logic rời rạc, phần mềm, phần cứng, firmware hoặc sự kết hợp bất kỳ của chúng. Mỗi bộ mã hóa video 20 và bộ giải mã video 30 có thể được bao gồm trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, một trong hai bộ này có thể được tích hợp dưới dạng một phần của bộ mã hóa/giải mã video kết hợp (CODEC). Thiết bị bao gồm bộ mã hóa video 20 và/hoặc bộ giải mã video 30 có thể bao gồm mạch tích hợp, bộ vi xử lý và/hoặc thiết bị truyền thông không dây, như điện thoại di động.

Fig.7 là sơ đồ khối minh họa ví dụ về bộ mã hóa video 20 mà có thể thực hiện các kỹ thuật theo sáng chế để mã hóa thông tin chuyển động dự đoán afin. Bộ mã hóa video 20 có thể thực hiện mã hóa nội hình ảnh và liên hình ảnh các khối video trong các lát video. Quy trình mã hóa nội hình ảnh dựa vào dự đoán không gian để giảm hoặc loại bỏ dữ liệu dư không gian trong video trong khung hoặc hình ảnh video nhất định. Mã hóa dự đoán liên hình ảnh dựa vào dự đoán thời gian để giảm hoặc loại bỏ dữ liệu dư thời gian trong video trong các khung hoặc hình ảnh liên kế của chuỗi video. Chế độ dự đoán trong hình ảnh (chế độ I) có thể chỉ bất kỳ trong số các chế độ mã hóa theo không gian. Chế độ dự đoán liên hình ảnh, như chế độ dự đoán một chiều (chế độ P) hoặc dự đoán hai chiều (chế độ B), có thể chỉ bất kỳ trong số các chế độ mã hóa theo thời gian.

Như được thể hiện trên Fig.7, bộ mã hóa video 20 thu khối video hiện thời trong khung video cần được mã hóa. Theo ví dụ trên Fig.7, bộ mã hóa video 20 bao gồm bộ chọn chế độ 40, bộ nhớ hình ảnh tham chiếu 64 (còn có thể được gọi là bộ đệm hình ảnh giải mã (decoded picture buffer - DPB), bộ cộng 50, bộ xử lý biến đổi 52, bộ lượng tử hóa 54 và bộ mã hóa entropy 56. Bộ chọn chế độ 40, lần lượt, bao gồm bộ bù chuyển động 44, bộ ước lượng chuyển động 42, bộ dự đoán trong hình ảnh 46, và bộ chia 48. Để tái tạo khối video, bộ mã hóa video 20 còn bao gồm bộ lượng tử hóa ngược 58, bộ biến đổi ngược 60 và bộ cộng 62. Bộ lọc tách khối (không được thể hiện trên Fig.7) có thể cũng được đưa vào để lọc các biên khối để loại bỏ các thành phần lạ dạng khối khối video đã tái tạo. Nếu muốn, bộ lọc tách khối thường lọc đầu ra của bộ cộng 62. Các bộ lọc bổ sung (trong vòng lặp hoặc sau vòng lặp) cũng có thể được sử dụng ngoài bộ lọc tách khối. Các bộ lọc này

không được thể hiện một cách vắn tắt, nhưng nếu muốn, có thể lọc đầu ra của bộ cộng 50 (như bộ lọc trong vòng lặp).

Trong quy trình mã hóa, bộ mã hóa video 20 thu khung hoặc lát dữ liệu video cần được mã hóa. Khung hoặc lát video có thể được chia thành nhiều khối video. Bộ ước lượng chuyển động 42 và bộ bù chuyển động 44 thực hiện mã hóa dự đoán liên hình ảnh đối với khối video thu được liên quan tới một hoặc nhiều khối trong một hoặc nhiều khung tham chiếu để tạo ra dự đoán thời gian. Bộ dự đoán trong hình ảnh 46 theo cách khác có thể thực hiện mã hóa dự đoán trong hình ảnh đối với khối video thu được liên quan tới một hoặc nhiều khối lân cận trong cùng một khung hoặc lát với khối cần được mã hóa để cung cấp dự đoán không gian. Bộ mã hóa video 20 có thể thực hiện nhiều bước mã hóa, ví dụ, để chọn chế độ mã hóa thích hợp cho mỗi khối dữ liệu video.

Hơn nữa, bộ chia 48 có thể chia các khối dữ liệu video thành các khối con, dựa vào việc đánh giá các sơ đồ phân chia trước đó trong các bước mã hóa trước đó. Ví dụ, bộ chia 48 có thể ban đầu phân chia khung hoặc lát thành các CTU và phân chia mỗi trong số các CTU thành các CU con dựa vào phân tích tốc độ méo (ví dụ, tối ưu hóa tốc độ méo). Bộ chọn chế độ 40 có thể còn tạo ra cấu trúc dữ liệu cây tứ phân biểu thị việc phân chia CTU thành các CU con. CU nút lá của cây tứ phân có thể bao gồm một hoặc nhiều PU và một hoặc nhiều TU.

Bộ chọn chế độ 40 có thể chọn một trong số các chế độ dự đoán, trong hình ảnh hoặc liên hình ảnh, ví dụ, dựa vào các kết quả sai số, và cung cấp khối dự đoán thu được cho bộ cộng 50 để tạo ra dữ liệu dư và cho bộ cộng 62 để tái tạo khối mã hóa để dùng làm khung tham chiếu. Bộ chọn chế độ 40 còn cung cấp các phần tử cú pháp, như vector chuyển động, chỉ báo chế độ mã hóa dự đoán trong hình ảnh, thông tin phân chia và các thông tin cú pháp như vậy khác cho bộ mã hóa entropy 56.

Bộ ước lượng chuyển động 42 và bộ bù chuyển động 44 có thể được tích hợp ở mức cao, nhưng được minh họa riêng cho mục đích giải thích khái niệm. Quy trình ước lượng chuyển động, được thực hiện bởi bộ ước lượng chuyển động 42, là quy trình tạo ra các vector chuyển động để ước lượng chuyển động cho các khối video. Vector chuyển động, ví dụ, có thể biểu thị sự dịch chuyển của PU của khối video trong khung video hoặc hình ảnh hiện thời liên quan đến khối dự đoán trong khung tham chiếu (hoặc bộ mã hóa khác) so

với khối hiện thời được mã hóa trong khung hiện thời (hoặc bộ mã hóa khác). Khối dự đoán là khối được thấy là phù hợp nhất với khối cần mã hóa, xét về chênh lệch điểm ảnh, giá trị này có thể được xác định bằng tổng chênh lệch tuyệt đối (sum of absolute difference - SAD), tổng chênh lệch bình phương (sum of square difference - SSD), hoặc các giá trị đo chênh lệch khác. Theo một số ví dụ, bộ mã hóa video 20 có thể tính toán các giá trị cho các vị trí điểm ảnh dưới số nguyên của các hình ảnh tham chiếu được lưu trữ trong bộ nhớ hình ảnh tham chiếu 64. Ví dụ, bộ mã hóa video 20 có thể nội suy các giá trị của các vị trí điểm ảnh một phần tư, các vị trí điểm ảnh một phần tám, hoặc các vị trí điểm ảnh phân số khác của hình ảnh tham chiếu. Do vậy, bộ ước lượng chuyển động 42 có thể thực hiện tìm kiếm chuyển động liên quan đến toàn bộ vị trí điểm ảnh và các vị trí điểm ảnh phân số và xuất ra vector chuyển động có độ chính xác điểm ảnh phân số.

Bộ ước lượng chuyển động 42 tính toán vector chuyển động cho PU của khối video trong lát được mã hóa liên hình ảnh bằng cách so sánh vị trí của PU với vị trí của khối dự đoán của hình ảnh tham chiếu. Hình ảnh tham chiếu có thể được chọn từ danh sách hình ảnh tham chiếu thứ nhất (danh sách 0) hoặc danh sách hình ảnh tham chiếu thứ hai (danh sách 1), mỗi trong số các danh sách này nhận diện một hoặc nhiều hình ảnh tham chiếu lưu trữ trong bộ nhớ hình ảnh tham chiếu 64. Bộ ước lượng chuyển động 42 gửi vector chuyển động đã tính toán cho bộ mã hóa entropy 56 và bộ bù chuyển động 44.

Quy trình bù chuyển động, được thực hiện bởi bộ bù chuyển động 44, có thể bao gồm bước tìm nạp hoặc tạo ra khối dự đoán dựa trên vector chuyển động được xác định bởi bộ ước lượng chuyển động 42. Ngoài ra, bộ ước lượng chuyển động 42 và bộ bù chuyển động 44 có thể được tích hợp về mặt chức năng, theo một số ví dụ. Khi thu vector chuyển động cho PU của khối video hiện thời, bộ bù chuyển động 44 có thể định vị khối dự đoán mà vector chuyển động trở vào trong một trong số các danh sách hình ảnh tham chiếu. Bộ cộng 50 tạo ra khối dữ liệu video dự bằng cách lấy giá trị điểm ảnh của khối video hiện thời được mã hóa trừ đi giá trị điểm ảnh của khối dự đoán, tạo ra giá trị chênh lệch điểm ảnh, như được nêu dưới đây. Nói chung, bộ ước lượng chuyển động 42 thực hiện ước lượng chuyển động đối với các thành phần độ chói, và bộ bù chuyển động 44 sử dụng các vector chuyển động được tính toán dựa vào các thành phần độ chói cho cả thành phần độ màu và thành phần độ chói. Bộ bù chuyển động 44 có thể tạo ra khối dự đoán sử dụng các

vector chuyển động, có thể bao gồm việc nội suy hoặc nếu không thì thực hiện thao tác toán học khác đối với các giá trị của các khối dự đoán được tham chiếu bởi các vector chuyển động. Bộ chọn chế độ 40 cũng có thể tạo ra các phần tử cú pháp liên quan đến các khối video và lát video để bộ giải mã video 30 sử dụng để giải mã khối video của lát video.

Bộ mã hóa video 20 có thể được tạo cấu hình để thực hiện kỹ thuật bất kỳ trong số các kỹ thuật khác nhau theo sáng chế đã thảo luận trên đây liên quan đến Fig.6. Ví dụ, bộ bù chuyển động 44 có thể được tạo cấu hình để mã hóa thông tin chuyển động cho khối dữ liệu video bằng cách sử dụng chế độ AMVP hoặc hợp nhất theo HEVC, và/hoặc có thể được tạo cấu hình để mã hóa thông tin chuyển động afin hoặc khối theo dữ liệu video bằng cách sử dụng chế độ liên hình ảnh afin hoặc chế độ hợp nhất afin theo các kỹ thuật của sáng chế.

Bộ dự đoán trong hình ảnh 46 có thể dự đoán trong hình ảnh khối hiện thời, thay cho dự đoán liên hình ảnh được thực hiện bởi bộ ước lượng chuyển động 42 và bộ bù chuyển động 44, như đã mô tả trên đây. Cụ thể là, bộ dự đoán trong hình ảnh 46 có thể xác định chế độ dự đoán trong hình ảnh sử dụng để mã hóa khối hiện thời. Theo một số ví dụ, bộ dự đoán trong hình ảnh 46 có thể mã hóa khối hiện thời bằng cách sử dụng các chế độ dự đoán trong hình ảnh khác nhau, ví dụ, trong các bước mã hóa riêng rẽ, và bộ dự đoán trong hình ảnh 46 (hoặc bộ chọn chế độ 40, theo một số ví dụ) có thể chọn chế độ dự đoán trong hình ảnh phù hợp để sử dụng từ các chế độ đã thử nghiệm.

Ví dụ, bộ dự đoán trong hình ảnh 46 có thể tính toán giá trị tốc độ - độ méo bằng cách sử dụng phân tích tốc độ - độ méo cho các chế độ dự đoán trong hình ảnh được thử nghiệm khác nhau và chọn chế độ dự đoán trong hình ảnh có đặc điểm tốc độ - độ méo tốt nhất trong số các chế độ đã thử nghiệm. Phân tích tốc độ - độ méo thường xác định lượng méo (hoặc sai số) giữa khối mã hóa và khối chưa mã hóa ban đầu mà được mã hóa để tạo ra khối mã hóa, cũng như tốc độ bit (tức là, số lượng bit) được dùng để tạo ra khối mã hóa. Bộ dự đoán trong hình ảnh 46 có thể tính toán tỷ lệ giữa các mức độ méo và tốc độ đối với các khối mã hóa khác nhau để xác định chế độ dự đoán trong hình ảnh nào có giá trị tốc độ - độ méo tốt nhất cho khối này.

Sau khi chọn chế độ dự đoán trong hình ảnh cho một khối, bộ dự đoán trong hình ảnh 46 có thể cung cấp thông tin chỉ báo về chế độ dự đoán trong hình ảnh đã chọn cho

khối đến bộ mã hóa entropy 56. Bộ mã hóa entropy 56 có thể mã hóa thông tin chỉ báo chế độ dự đoán trong hình ảnh được chọn. Bộ mã hóa video 20 có thể bao gồm vào trong dòng bit đã được truyền dữ liệu cấu hình, mà có thể bao gồm các bảng chỉ số về chế độ dự đoán trong hình ảnh và các bảng chỉ số chế độ dự đoán trong hình ảnh được sửa đổi (còn được gọi là các bảng ánh xạ từ mã), các định nghĩa về các ngữ cảnh mã hóa cho các khối khác nhau, và các chỉ báo về chế độ dự đoán trong hình ảnh nhiều khả năng xảy ra nhất, bảng chỉ số về chế độ dự đoán trong hình ảnh, và bảng chỉ số về chế độ dự đoán trong hình ảnh được sửa đổi để sử dụng cho mỗi ngữ cảnh.

Bộ mã hóa video 20 tạo ra khối video dư bằng cách lấy khối video ban đầu được mã hóa trừ đi dữ liệu dự đoán của bộ chọn chế độ 40. Bộ cộng 50 là thành phần hoặc các thành phần thực hiện phép trừ này. Bộ xử lý biến đổi 52 áp dụng phép biến đổi, như phép biến đổi cosin rời rạc (discrete cosine transform - DCT) hoặc phép biến đổi tương tự về mặt khái niệm, cho khối dữ liệu dư, tạo ra khối video chứa các giá trị hệ số biến đổi. Biến đổi sóng nhỏ, biến đổi số nguyên, biến đổi băng phụ, biến đổi sin rời rạc (discrete sine transform - DST), hoặc các loại biến đổi khác có thể được dùng thay cho DCT. Trong trường hợp bất kỳ, bộ xử lý biến đổi 52 áp dụng phép biến đổi cho khối dữ liệu dư, tạo ra khối hệ số biến đổi. Việc biến đổi có thể chuyển đổi thông tin dư từ miền điểm ảnh thành miền biến đổi, như miền tần số. Bộ xử lý biến đổi 52 có thể truyền các hệ số biến đổi thu được cho bộ lượng tử hóa 54. Bộ lượng tử hóa 54 lượng tử hóa các hệ số biến đổi để giảm thêm tốc độ bit. Quy trình lượng tử hóa có thể giảm chiều sâu bit liên quan đến một số hoặc tất cả các hệ số. Mức lượng tử hóa có thể được sửa đổi bằng cách điều chỉnh tham số lượng tử hóa.

Sau khi lượng tử hóa, bộ mã hóa entropy 56 mã hóa entropy các hệ số biến đổi đã lượng tử hóa. Ví dụ, bộ mã hóa entropy 56 có thể thực hiện mã hóa độ dài thay đổi thích ứng theo ngữ cảnh (Context adaptive variable length coding - CAVLC), mã hóa thuật toán nhị phân thích ứng theo ngữ cảnh (Context adaptive binary arithmetic coding - CABAC), mã hóa thuật toán nhị phân thích ứng theo ngữ cảnh dựa vào cú pháp (syntax-based context-adaptive binary arithmetic coding - SBAC), mã hóa entropy phân chia khoảng xác suất (probability interval partitioning entropy - PIPE) hoặc kỹ thuật mã hóa entropy khác. Trong trường hợp mã hóa entropy dựa vào ngữ cảnh, ngữ cảnh có thể dựa vào các khối lân

cận. Sau khi mã hóa entropy bằng bộ mã hóa entropy 56, dòng bit mã hóa có thể được truyền cho thiết bị khác, (ví dụ, bộ giải mã video 30) hoặc được lưu trữ để truyền hoặc tìm kiếm sau đó.

Bộ lượng tử hóa ngược 58 và bộ biến đổi ngược 60 lần lượt áp dụng lượng tử hóa ngược và biến đổi ngược để tái tạo khối dữ liệu dư trong miền điểm ảnh. Cụ thể, bộ cộng 62 cộng khối dữ liệu dư được tái tạo vào khối dự đoán được bù chuyển động đã tạo ra trước đó bởi bộ bù chuyển động 44 hoặc bộ dự đoán trong hình ảnh 46 để tạo ra khối video tái tạo để lưu trữ trong bộ nhớ hình ảnh tham chiếu 64. Khối video tái tạo có thể được bộ ước lượng chuyển động 42 và bộ bù chuyển động 44 sử dụng làm khối tham chiếu để mã hóa liên hình ảnh khối trong khung video tiếp theo.

Fig.8 là sơ đồ khối minh họa ví dụ về bộ giải mã video 30 mà có thể thực hiện các kỹ thuật theo sáng chế để giải mã thông tin chuyển động dự đoán afin Theo ví dụ trên Fig.8, bộ giải mã video 30 bao gồm bộ giải mã entropy 70, bộ bù chuyển động 72, bộ dự đoán trong hình ảnh 74, bộ lượng tử hóa ngược 76, bộ biến đổi ngược 78, bộ nhớ hình ảnh tham chiếu 82 và bộ cộng 80. Bộ giải mã video 30 có thể, theo một số ví dụ, thực hiện bước giải mã thường ngược với bước mã hóa được mô tả liên quan đến bộ mã hóa video 20 (Fig.7). Bộ bù chuyển động 72 có thể tạo ra dữ liệu dự đoán dựa vào vector chuyển động thu được từ bộ giải mã entropy 70, còn bộ dự đoán trong hình ảnh 74 có thể tạo ra dữ liệu dự đoán dựa vào bộ chỉ báo chế độ dự đoán trong hình ảnh thu được từ bộ giải mã entropy 70.

Trong quá trình giải mã, bộ giải mã video 30 thu dòng bit video mã hóa biểu diễn các khối video của lát video mã hóa và các phần tử cú pháp gắn kèm từ bộ mã hóa video 20. Bộ giải mã entropy 70 của bộ giải mã video 30 giải mã entropy dòng bit để tạo ra các hệ số lượng tử hóa, các vector chuyển động hoặc các chỉ báo chế độ dự đoán trong hình ảnh, và các phần tử cú pháp khác. Bộ giải mã entropy 70 chuyển tiếp các vector chuyển động và các phần tử cú pháp khác đến bộ bù chuyển động 72. Bộ giải mã video 30 có thể thu phần tử cú pháp ở mức lát video và/hoặc mức khối video.

Khi lát video được mã hóa dưới dạng lát được mã hóa nội hình ảnh (I), thì bộ dự đoán trong hình ảnh 74 có thể tạo ra dữ liệu dự đoán cho khối video của lát video hiện thời dựa vào chế độ dự đoán trong hình ảnh được báo hiệu và dữ liệu của khối được giải mã

trước đó của khung hoặc hình ảnh hiện thời. Khi khung video được mã hóa dưới dạng lát được mã hóa liên hình ảnh (tức là, B, P hoặc GPB), thì bộ bù chuyển động 72 tạo ra các khối dự đoán cho khối video của lát video hiện thời dựa vào các vector chuyển động và các phần tử cú pháp khác thu được từ bộ giải mã entropy 70. Các khối dự đoán có thể được tạo ra từ một trong các hình ảnh tham chiếu trong một trong các danh sách hình ảnh tham chiếu. Bộ giải mã video 30 có thể tạo ra các danh sách khung tham chiếu, danh sách 0 và danh sách 1, bằng cách sử dụng các kỹ thuật xây dựng mặc định dựa vào các hình ảnh tham chiếu được lưu trữ trong bộ nhớ hình ảnh tham chiếu 82. Bộ bù chuyển động 72 xác định thông tin dự đoán cho khối video của lát video hiện thời bằng cách phân tích cú pháp các vector chuyển động và các phần tử cú pháp khác, và sử dụng thông tin dự đoán để tạo ra các khối dự đoán cho khối video hiện thời đang được giải mã. Ví dụ, bộ bù chuyển động 72 sử dụng một số phần tử cú pháp thu được để xác định chế độ dự đoán (ví dụ, dự đoán trong hình ảnh hoặc liên hình ảnh) dùng để mã hóa các khối video của lát video, loại lát dự đoán liên hình ảnh (ví dụ, lát B hoặc lát P), thông tin cấu trúc cho một hoặc nhiều danh sách hình ảnh tham chiếu cho lát này, các vector chuyển động cho mỗi khối video được mã hóa liên hình ảnh của lát này, trạng thái dự đoán liên hình ảnh dành cho mỗi khối video mã hóa liên hình ảnh của lát này và các thông tin khác để giải mã các khối video trong lát video hiện thời.

Bộ giải mã video 30 có thể được tạo cấu hình để thực hiện kỹ thuật bất kỳ trong số các kỹ thuật khác nhau của sáng chế đã thảo luận trên đây liên quan đến Fig.6. Ví dụ, bộ bù chuyển động 72 có thể được tạo cấu hình để thực hiện dự đoán vector chuyển động bằng cách sử dụng chế độ AMVP hoặc hợp nhất theo HEVC, và/hoặc có thể được tạo cấu hình để thực hiện thông tin chuyển động afin hoặc khối dữ liệu video bằng cách sử dụng chế độ liên hình ảnh afin hoặc chế độ hợp nhất afin theo các kỹ thuật của sáng chế. Bộ giải mã entropy 70 có thể giải mã một hoặc nhiều phần tử cú pháp biểu diễn cách thông tin chuyển động (ví dụ, thông tin chuyển động afin) được mã hóa cho khối hiện thời.

Bộ bù chuyển động 72 còn có thể thực hiện nội suy dựa vào bộ lọc nội suy. Bộ bù chuyển động 72 có thể sử dụng các bộ lọc nội suy như được sử dụng bởi bộ mã hóa video 20 trong khi mã hóa các khối video để tính toán các giá trị đã nội suy cho các điểm ảnh dưới số nguyên của các khối tham chiếu. Trong trường hợp này, bộ bù chuyển động 72 có

thể xác định các bộ lọc nội suy dùng bởi bộ mã hóa video 20 từ các phần tử cú pháp nhận được và sử dụng các bộ lọc nội suy này để tạo ra các khối dự đoán.

Bộ lượng tử hóa ngược 76 lượng tử hóa ngược, tức là -khử lượng tử hóa các hệ số biến đổi đã lượng tử hóa có trong dòng bit và được giải mã bởi bộ giải mã entropy 70. Quy trình lượng tử hóa ngược có thể bao gồm bước sử dụng tham số lượng tử hóa QP_Y được tính bởi bộ giải mã video 30 cho mỗi khối video trong lát video để xác định mức độ lượng tử hóa và, tương tự, mức độ lượng tử hóa ngược cần được áp dụng.

Bộ biến đổi ngược 78 áp dụng phép biến đổi ngược, ví dụ DCT ngược, phép biến đổi số nguyên ngược, hoặc quá trình biến đổi ngược tương tự về mặt khái niệm, cho các hệ số biến đổi để tạo ra các khối dữ liệu dư trong miền điểm ảnh.

Sau khi bộ bù chuyển động 72 tạo ra khối dự đoán cho khối video hiện thời dựa vào các vectơ chuyển động và các phần tử cú pháp khác, bộ giải mã video 30 tạo ra khối video được giải mã bằng cách cộng các khối dữ liệu dư từ bộ biến đổi ngược 78 với các khối dự đoán tương ứng được tạo bởi bộ bù chuyển động 72. Bộ cộng 80 là thành phần hoặc các thành phần thực hiện phép tính cộng này. Nếu cần, bộ lọc tách khối cũng có thể được áp dụng để lọc các khối đã được giải mã để loại bỏ các thành phần lạ dạng khối. Các bộ lọc vòng lặp khác (trong vòng lặp mã hóa hoặc sau vòng lặp mã hóa) cũng có thể được dùng để làm trơn các chuyển tiếp điểm ảnh hoặc nếu không thì cải thiện chất lượng video. Các khối video đã giải mã trong một khung hoặc hình ảnh nhất định sau đó được lưu trữ vào bộ nhớ hình ảnh tham chiếu 82, bộ nhớ này lưu trữ các hình ảnh tham chiếu được dùng để bù chuyển động tiếp theo. Bộ nhớ hình ảnh tham chiếu 82 còn lưu trữ video đã giải mã để sau đó trình diễn trên thiết bị hiển thị, như thiết bị hiển thị 32 trên Fig.6.

Fig.11 là lưu đồ minh họa ví dụ về phương pháp mã hóa khối dữ liệu video hiện thời theo các kỹ thuật của sáng chế. Nhằm mục đích làm ví dụ và giải thích, phương pháp trên Fig.11 được giải thích liên quan tới bộ mã hóa video 20 trên các hình vẽ Fig.6 và Fig.7. Tuy nhiên, cần hiểu rằng các thiết bị khác có thể được tạo cấu hình để thực hiện phương pháp này hoặc phương pháp tương tự.

Ban đầu, mặc dù không được thể hiện trên Fig.11, nhưng bộ chọn chế độ 40 có thể xác định chế độ dự đoán để được sử dụng để dự đoán khối hiện thời. Theo ví dụ này, giả định rằng bộ chọn chế độ 40 chọn chế độ dự đoán afin, bao gồm dự đoán sử dụng ít nhất

hai vector chuyển động. Do vậy, bộ chọn chế độ 40 khiến cho bộ ước lượng chuyển động 42 thực hiện quy trình tìm kiếm chuyển động để xác định vector chuyển động thứ nhất và vector chuyển động thứ hai (100). Bộ chọn chế độ 40 có thể so sánh các kết quả tốc độ - độ méo giữa các phương pháp dự đoán khác nhau, như dự đoán trong hình ảnh, dự đoán liên hình ảnh, và dự đoán afin, và xác định rằng dự đoán afin tạo ra các kết quả tốc độ - độ méo tốt nhất trong số các chế độ dự đoán kiểm tra khác nhau.

Sau khi bộ ước lượng chuyển động 42 xác định các vector chuyển động thứ nhất và thứ hai, bộ mã hóa video 20 có thể tính toán bộ dự đoán vector chuyển động (MVP) thứ nhất (102) cho vector chuyển động thứ nhất. MVP thứ nhất có thể tương ứng với vector chuyển động của khối lân cận. Bộ mã hóa video 20 sau đó có thể tính toán chênh lệch vector chuyển động (MVD) thứ nhất là chênh lệch giữa vector chuyển động thứ nhất và bộ dự đoán vector chuyển động thứ nhất (104). Cụ thể, bộ mã hóa video 20 có thể tính toán lần lượt các chênh lệch giữa các thành phần x và y của vector chuyển động và MVD.

Bộ mã hóa video 20 sau đó có thể xác định MVP thứ hai (106) cho vector chuyển động thứ hai. Bộ mã hóa video 20 còn có thể tính toán MVD thứ hai (108) là chênh lệch giữa vector chuyển động thứ hai và MVP thứ hai.

Bộ bù chuyển động 44 còn có thể dự đoán khối hiện thời (110) bằng cách sử dụng dự đoán afin để tạo ra khối dự đoán cho khối hiện thời. Mặc dù hai vector chuyển động được thảo luận để phục vụ mục đích làm ví dụ, nhưng cần hiểu rằng ba vector chuyển động có thể được dùng cho dự đoán afin để tạo ra khối dự đoán. Tương tự, bộ mã hóa video 20 có thể tạo ra MVD thứ ba và MVP thứ ba theo các kỹ thuật của sáng chế cho vector chuyển động thứ ba, như được thảo luận trên đây.

Sau khi tạo ra khối dự đoán, bộ mã hóa video 20 có thể tính toán khối dữ liệu dư (112) biểu diễn các chênh lệch theo từng điểm ảnh giữa khối hiện thời và khối dự đoán. Cụ thể, bộ cộng 50 có thể tính toán các chênh lệch theo từng điểm ảnh giữa khối hiện thời và khối dự đoán. Bộ mã hóa video 20 sau đó có thể mã hóa MVD thứ nhất, MVD thứ hai, và khối dữ liệu dư (114) để mã hóa khối hiện thời. Tức là, theo các kỹ thuật của sáng chế, bộ mã hóa video 20 có thể mã hóa MVD thứ nhất bằng cách sử dụng, ví dụ, các kỹ thuật dự đoán vector chuyển động, như chế độ hợp nhất hoặc chế độ AMVP, và mã hóa MVD thứ hai bằng cách dự đoán MVD thứ hai từ MVD thứ nhất. Vì vậy, để mã hóa MVD thứ

hai, bộ mã hóa video 20 có thể mã hóa dữ liệu biểu diễn chênh lệch giữa MVD thứ nhất và MVD thứ hai, như các chênh lệch giữa các thành phần x và y của MVD thứ nhất và MVD thứ hai. Để mã hóa khối dữ liệu dư, bộ xử lý biến đổi 52 có thể biến đổi khối dữ liệu dư, bộ lượng tử hóa 54 có thể lượng tử hóa các hệ số biến đổi của khối biến đổi thu được, và bộ mã hóa entropy 56 có thể mã hóa entropy các hệ số biến đổi lượng tử hóa thu được.

Theo phương án này, phương pháp trên Fig.11 là một ví dụ của phương pháp bao gồm bước lập mã (cụ thể là, mã hóa) chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất; dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời; và lập mã (cụ thể là, mã hóa) khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai. Bằng việc thực hiện phương pháp trên Fig.11, bộ mã hóa video 20 có thể tạo ra dòng bit mà có hiệu quả về mặt băng thông hơn so với khi thực hiện các kỹ thuật trước đó, vì dữ liệu biểu diễn MVD thứ hai có thể nhỏ hơn vì MVD thứ hai được dự đoán từ MVD thứ nhất.

Fig.12 là lưu đồ minh họa ví dụ về phương pháp giải mã khối dữ liệu video hiện thời theo các kỹ thuật của sáng chế. Phương pháp trên Fig.12 được giải thích liên quan tới bộ giải mã video 30 trên các hình vẽ Fig.6 và/hoặc Fig.8 phục vụ mục đích làm ví dụ. Tuy nhiên, cần hiểu rằng các thiết bị khác có thể được tạo cấu hình để thực hiện các kỹ thuật của phương pháp này hoặc phương pháp tương tự.

Bộ giải mã video 30 có thể giải mã chênh lệch vector chuyển động (MVD) thứ nhất, MVD thứ hai, và khối dữ liệu dư (120) của khối hiện thời. Tức là, theo các kỹ thuật của sáng chế, bộ giải mã video 30 có thể giải mã MVD thứ nhất bằng cách sử dụng, ví dụ, các kỹ thuật dự đoán vector chuyển động, như chế độ hợp nhất hoặc chế độ AMVP, và giải mã MVD thứ hai bằng cách dự đoán MVD thứ hai từ MVD thứ nhất. Vì vậy, để giải mã MVD thứ hai, bộ giải mã video 30 có thể giải mã dữ liệu biểu diễn chênh lệch giữa MVD thứ nhất và MVD thứ hai, như các chênh lệch giữa các thành phần x và y của MVD thứ nhất và MVD thứ hai. Để giải mã khối dữ liệu dư, bộ giải mã entropy 70 có thể giải mã entropy các hệ số biến đổi lượng tử hóa, bộ lượng tử hóa ngược 76 có thể lượng tử hóa ngược các

hệ số biến đổi, và bộ biến đổi ngược 78 có thể biến đổi ngược các hệ số biến đổi để tái tạo khối dữ liệu dư.

Bộ bù chuyển động 72 sau đó có thể xác định bộ dự đoán vector chuyển động (MVP) thứ nhất (122) cho vector chuyển động thứ nhất của khối hiện thời và tính toán vector chuyển động thứ nhất (124) từ MVP thứ nhất. Cụ thể là, bộ bù chuyển động 72 có thể thêm MVD thứ nhất vào MVP thứ nhất để tính toán vector chuyển động thứ nhất. Tương tự, bộ bù chuyển động 72 có thể xác định MVP thứ hai (126) cho vector chuyển động thứ hai của khối hiện thời và tính toán vector chuyển động thứ hai (128) từ MVP thứ hai. Cụ thể, bộ bù chuyển động 72 có thể thêm MVD thứ hai vào MVP thứ hai để tính toán vector chuyển động thứ hai. Theo một số ví dụ, vector chuyển động thứ ba có thể còn bao gồm, trong trường hợp đó bộ giải mã entropy 70 có thể giải mã entropy dữ liệu biểu diễn chênh lệch giữa, ví dụ, MVD thứ nhất và MVD thứ ba cho vector chuyển động thứ ba, và bộ bù chuyển động 72 có thể tính toán vector chuyển động thứ ba từ MVD thứ ba và MVP thứ ba theo cách tương tự.

Bộ bù chuyển động 72 sau đó có thể dự đoán khối hiện thời (130), ví dụ, theo dự đoán chuyển động afin bằng cách sử dụng các vector chuyển động thứ nhất và thứ hai (và có khả năng là thứ ba). Bộ giải mã video 30 sau đó có thể giải mã khối hiện thời (132), ví dụ, bằng cách khiến cho bộ cộng 80 thêm các giá trị của khối dự đoán vào các giá trị của khối dữ liệu dư theo từng điểm ảnh một.

Theo phương án này, phương pháp trên Fig.12 là một ví dụ của phương pháp bao gồm bước lập mã (cụ thể là giải mã) chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất; dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời; và lập mã (cụ thể là giải mã) khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai. Bằng cách thực hiện phương pháp trên Fig.12, bộ giải mã video 30 có thể giải mã dòng bit mà có hiệu quả về mặt băng thông hơn so với khi thực hiện các kỹ thuật trước đó, vì dữ liệu biểu diễn MVD thứ hai có thể là nhỏ hơn vì MVD thứ hai được dự đoán từ MVD thứ nhất.

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 các trình tự khác nhau, có thể được bỏ sung, hợp nhất hoặc được loại bỏ hoàn toàn (ví dụ, không phải tất cả các hành động hoặc sự kiện được mô tả đều cần thiết cho việc thực hiện các kỹ thuật theo sáng chế). Ngoài ra, theo một số phương án làm ví dụ, các thao tác hoặc sự kiện có thể được thực hiện đồng thời, ví dụ, thông qua quy trình xử lý đa xâu chuỗi, quy trình xử lý gián đoạn, hoặc nhiều bộ xử lý, chứ không phải chỉ có thực hiện tuần tự.

Theo một hoặc nhiều phương án làm ví dụ, các chức năng được mô tả có thể được thực hiện dưới dạng phần cứng, phần mềm, firmware, hoặc dạng kết hợp bất kỳ của chúng. Nếu được thực hiện bằng phần mềm, các chức năng có thể được lưu trữ hoặc truyền dưới dạng một hoặc nhiều lệnh hoặc mã trên phương tiện đọc được bằng máy tính và được thực hiện bằng bộ xử lý dựa trên phần cứng. Phương tiện đọc được bằng máy tính có thể bao gồm các phương tiện lưu trữ đọc được bằng máy tính tương ứng với phương tiện hữu hình như phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông có phương tiện bất kỳ hỗ trợ truyền chương trình máy tính từ nơi này đến nơi khác, ví dụ, theo giao thức truyền thông. Theo cách này, phương tiện đọc được bằng máy tính thông thường có thể tương ứng với (1) phương tiện lưu trữ đọc được bằng máy tính hữu hình là phương tiện bất biến hoặc (2) phương tiện truyền thông như tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ liệu có thể là phương tiện khả dụng bất kỳ mà có thể được truy cập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để truy hồi các lệnh, mã và/hoặc các cấu trúc dữ liệu để thực hiện các kỹ thuật được mô tả trong sáng chế. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được 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 nhập ngẫu nhiên (RAM - Random Access Memory), bộ nhớ chỉ đọc (ROM - Read Only Memory), bộ nhớ chỉ đọc lập trình được xóa được bằng điện (EEPROM - Electrically Erasable Programmable Read Only Memory), đĩa compac-bộ nhớ chỉ đọc (CD-ROM - Compact Disc-Read Only Memory) 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 website, máy chủ hoặc nguồn từ xa khác sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn đôi, đường thuê bao số (Digital Subscriber Line - DS), hoặc công nghệ không dây như hồng ngoại, vô tuyến và vi ba, thì cáp đồng trục, cáp quang sợi, cáp xoắn đôi, DSL, hoặc công nghệ không dây như hồng ngoại, vô tuyến và vi ba đó cũng nằm trong định nghĩa về phương tiện. Tuy nhiên, cần phải hiểu rằng phương tiện lưu trữ đọc được bằng máy tính và phương tiện lưu dữ liệu không bao gồm các kết nối, sóng mang, tín hiệu hoặc phương tiện chuyển tiếp khác, nhưng thay vào đó được hướng đến phương tiện lưu trữ hữu hình, bất biến. Đĩa từ và đĩa quang, như mô tả ở đây, bao gồm CD (đĩa compact), đĩa laser, đĩa quang, đĩa số đa năng (DVD - digital versatile disc), đĩa mềm và đĩa blu-ray trong đó các đĩa từ thường tái tạo dữ liệu bằng từ tính, trong khi các đĩa quang thường tái tạo dữ liệu bằng quang học có laser. Dạng kết hợp của các loại phương tiện nêu trên cũng được bao gồm trong phạm vi phương tiện đọc được bằng máy tính.

Các lệnh có thể được thực thi bởi một hoặc nhiều bộ xử lý, chẳng hạn như một hoặc nhiều bộ xử lý tín hiệu số (digital signal processor - DSP), các bộ vi xử lý đa năng, mạch tích hợp chuyên dụng (application specific integrated circuit - ASIC), mảng cổng lập trình được theo trường (field programmable gate array- FPGA), hoặc các mạch logic tích hợp hoặc rời rạc 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, theo một số khía cạnh, chức năng mô tả ở đây có thể được tạo ra trong mô đun phần cứng và/hoặc phần mềm chuyên dụng được tạo cấu hình để mã hoá và giải mã, hoặc được kết hợp thành bộ mã hoá-giải mã kết hợp. Hơn nữa, các kỹ thuật có thể được thực hiện hoàn toàn trong một hoặc nhiều mạch hoặc phần tử logic.

Các kỹ thuật theo sáng chế có thể được thực hiện bằng rất nhiều thiết bị hoặc máy, trong đó có thiết bị cầm tay không dây, mạch tích hợp (IC - Integrated Circuit) hoặc bộ IC (ví dụ, bộ chip). Các thành phần, mô đun 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 được bộc lộ, chứ không nhất thiết phải được thực hiện bằng các bộ phận phần cứng khác nhau. Thực ra, như đã nêu trên, các bộ phận khác nhau có thể được kết hợp lại

thành một bộ phận phần cứng mã hoá-giải mã hoặc được cung cấp bởi một tập hợp các bộ phận phần cứng tương tác với nhau, bao gồm một hoặc nhiều bộ xử lý như đã nêu trên, kết hợp với phần mềm và/hoặc firmware phù hợp.

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

YÊU CẦU BẢO HỘ

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

lập mã chênh lệch vector chuyển động (motion vector difference - MVD) thứ nhất biểu diễn sự chênh lệch thứ nhất giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (motion vector predictor - MVP) thứ nhất cho vector chuyển động thứ nhất;

dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, MVD thứ hai biểu diễn sự chênh lệch thứ hai giữa vector chuyển động thứ hai và MVP thứ hai cho vector chuyển động thứ hai, vector chuyển động thứ hai là khác với vector chuyển động thứ nhất; và

lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

2. Phương pháp theo điểm 1, trong đó MVD thứ nhất bao gồm thành phần ngang ($MVDx1$) và thành phần dọc ($MVDy1$), MVP thứ nhất bao gồm thành phần ngang ($MVPx1$) và thành phần dọc ($MVPy1$), vector chuyển động thứ nhất bao gồm thành phần ngang ($MVx1$) và thành phần dọc ($MVy1$), $MVDx1 = MVx1 - MVPx1$, và $MVDy1 = MVy1 - MVPy1$.

3. Phương pháp theo điểm 1, trong đó bước lập mã khối hiện thời bao gồm lập mã khối hiện thời theo mô hình afin bốn tham số.

4. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước dự đoán MVD thứ ba từ ít nhất một trong MVD thứ nhất hoặc MVD thứ hai cho vector chuyển động thứ ba của khối hiện thời, trong đó bước lập mã khối hiện thời bao gồm lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất, vector chuyển động thứ hai, và vector chuyển động thứ ba.

5. Phương pháp theo điểm 4, trong đó bước lập mã khối hiện thời bao gồm lập mã khối hiện thời theo mô hình afin sáu tham số.

6. Phương pháp theo điểm 1, trong đó vector chuyển động thứ nhất bắt đầu từ góc trên-trái của khối hiện thời, và trong đó vector chuyển động thứ hai bắt đầu từ góc trên-phải của

khởi hiện thời.

7. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước lập mã dữ liệu xác định các điểm điều khiển cho vector chuyển động thứ nhất và vector chuyển động thứ hai.

8. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước xác định các điểm điều khiển cho vector chuyển động thứ nhất và vector chuyển động thứ hai dựa vào hình dạng của khởi hiện thời.

9. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước ngầm suy ra các điểm điều khiển cho vector chuyển động thứ nhất và vector chuyển động thứ hai.

10. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước lập mã dữ liệu biểu diễn MVD'2 cho MVD thứ hai, trong đó MVD'2 biểu diễn giá trị dư cho MVD thứ hai liên quan đến MVD thứ nhất.

11. Phương pháp theo điểm 10, trong đó MVD thứ nhất là MVD1, MVD thứ hai là MVD2, w là giá trị trọng số, và $MVD'2 = MVD1 - w * MVD2$.

12. Phương pháp theo điểm 10, trong đó MVD thứ nhất là MVD1, MVD thứ hai là MVD2, và $MVD'2 = MVD2 - ((MVD1 + 1) >> 1)$ cho giá trị trọng số là 0,5, trong đó ">>" là toán tử dịch phải phân theo bit.

13. Phương pháp theo điểm 10, trong đó MVD thứ nhất là MVD1, MVD thứ hai là MVD2, và $MVD'2 = MVD2 - ((MVD1 + 2) >> 2)$ cho giá trị trọng số là 0,25, trong đó ">>" là toán tử dịch phải phân theo bit.

14. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước quyết định dự đoán MVD thứ hai từ MVD thứ nhất, trong đó việc dự đoán MVD thứ hai từ MVD thứ nhất bao gồm dự đoán MVD thứ hai từ MVD thứ nhất để đáp lại việc quyết định dự đoán MVD thứ hai từ MVD thứ nhất.

15. Phương pháp theo điểm 14, trong đó bước quyết định dự đoán MVD thứ hai từ MVD thứ nhất bao gồm quyết định dự đoán MVD thứ hai từ MVD thứ nhất dựa vào hình dạng của khối hiện thời.

16. Phương pháp theo điểm 14, trong đó bước quyết định dự đoán MVD thứ hai từ MVD thứ nhất bao gồm quyết định dự đoán MVD thứ hai từ MVD thứ nhất dựa vào phương pháp dự đoán chuyển động cho khối hiện thời.

17. Phương pháp theo điểm 16, trong đó bước quyết định dự đoán MVD thứ hai từ MVD thứ nhất bao gồm xác định rằng phương pháp dự đoán chuyển động là chế độ hợp nhất afin.

18. Phương pháp theo điểm 1, trong đó MVD thứ hai bao gồm thành phần ngang ($MVD2^x$) và thành phần dọc ($MVD2^y$), và trong đó dự đoán MVD thứ hai bao gồm dự đoán $MVD2^x$ khác với dự đoán $MVD2^y$.

19. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước tạo ra MVP thứ ba cho vector chuyển động thứ ba của khối hiện thời từ ít nhất một trong MVD thứ nhất hoặc MVD thứ hai.

20. Phương pháp theo điểm 1, trong đó MVD thứ nhất là MVD1, phương pháp này còn bao gồm các bước:

xác định MVP trung gian thứ hai ($MVP'2$) cho vector chuyển động thứ hai từ vector chuyển động của một hoặc nhiều khối lân cận với khối hiện thời; và

tạo ra MVP thứ hai ($MVP2$) cho vector chuyển động thứ hai từ $MVP'2$ và MVD1.

21. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước tạo ra MVP thứ hai cho vector chuyển động thứ hai từ MVD thứ nhất và MVD thứ ba cho vector chuyển động thứ ba của khối hiện thời, trong đó MVP thứ hai là MVP2, MVD thứ nhất là MVD1, MVD thứ ba là MVD3, phương pháp này còn bao gồm bước xác định MVP trung gian thứ hai ($MVP'2$) cho vector chuyển động thứ hai từ vector chuyển động của một hoặc nhiều

khối lân cận với khối hiện thời, trong đó việc tạo ra MVP2 bao gồm tạo ra MVP2 dưới dạng $MVP2 = MVP'2 + ((MVD1 + MVD3) \gg 1)$, trong đó “ $\gg$ ” là toán tử dịch phải phân theo bit.

22. Phương pháp theo điểm 1, trong đó bước lập mã khối hiện thời bao gồm giải mã khối hiện thời, bao gồm:

- thêm MVD thứ nhất vào MVP thứ nhất để tái tạo vector chuyển động thứ nhất;
- xác định MVP thứ hai cho vector chuyển động thứ hai;
- tái tạo MVD thứ hai bằng cách sử dụng dự đoán từ MVD thứ nhất;
- thêm MVD thứ hai vào MVP thứ hai để tái tạo vector chuyển động thứ hai;
- tạo ra khối dự đoán cho khối hiện thời bằng cách sử dụng vector chuyển động thứ nhất và vector chuyển động thứ hai;
- giải mã các khối dữ liệu dư cho khối hiện thời; và
- thêm khối dữ liệu dư và khối dự đoán để tái tạo khối hiện thời.

23. Phương pháp theo điểm 1, trong đó bước lập mã khối hiện thời bao gồm mã hóa khối hiện thời, bao gồm:

- trừ MVP thứ nhất từ vector chuyển động thứ nhất để tạo ra MVD thứ nhất;
- xác định MVP thứ hai cho vector chuyển động thứ hai;
- trừ vector chuyển động thứ hai từ MVP thứ hai để tạo ra MVD thứ hai;
- mã hóa MVD thứ nhất;
- mã hóa dữ liệu biểu diễn MVD thứ hai được dự đoán từ MVD thứ nhất;
- tạo ra khối dự đoán cho khối hiện thời bằng cách sử dụng vector chuyển động thứ nhất và vector chuyển động thứ hai;
- trừ khối dự đoán từ khối hiện thời để tạo ra khối dữ liệu dư; và
- mã hóa khối dữ liệu dư.

24. Thiết bị lập mã dữ liệu video, thiết bị này bao gồm:

- bộ nhớ được tạo cấu hình để lưu trữ dữ liệu video; và
- một hoặc nhiều bộ xử lý lắp đặt trong hệ mạch và được tạo cấu hình để:
 - lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán

bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất;

dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, MVD thứ hai biểu diễn sự chênh lệch thứ hai giữa vector chuyển động thứ hai và MVP thứ hai cho vector chuyển động thứ hai, vector chuyển động thứ hai là khác với vector chuyển động thứ nhất; và

lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

25. Thiết bị theo điểm 24, trong đó một hoặc nhiều bộ xử lý được tạo cấu hình để:

thêm MVD thứ nhất vào MVP thứ nhất để tái tạo vector chuyển động thứ nhất;

xác định MVP thứ hai cho vector chuyển động thứ hai;

tái tạo MVD thứ hai bằng cách sử dụng dự đoán từ MVD thứ nhất;

thêm MVD thứ hai vào MVP thứ hai để tái tạo vector chuyển động thứ hai;

tạo ra khối dự đoán cho khối hiện thời bằng cách sử dụng vector chuyển động thứ nhất và vector chuyển động thứ hai;

giải mã các khối dữ liệu dư cho khối hiện thời;

thêm khối dữ liệu dư và khối dự đoán để tái tạo khối hiện thời.

26. Thiết bị theo điểm 24, trong đó một hoặc nhiều bộ xử lý được tạo cấu hình để:

trừ MVP thứ nhất từ vector chuyển động thứ nhất để tạo ra MVD thứ nhất;

xác định MVP thứ hai cho vector chuyển động thứ hai;

trừ vector chuyển động thứ hai từ MVP thứ hai để tạo ra MVD thứ hai;

mã hóa MVD thứ nhất;

mã hóa dữ liệu biểu diễn MVD thứ hai được dự đoán từ MVD thứ nhất;

tạo ra khối dự đoán cho khối hiện thời bằng cách sử dụng vector chuyển động thứ nhất và vector chuyển động thứ hai;

trừ khối dự đoán từ khối hiện thời để tạo ra khối dữ liệu dư; và

mã hóa khối dữ liệu dư.

27. Thiết bị theo điểm 24, trong đó thiết bị này còn bao gồm màn hình được tạo cấu hình để hiển thị dữ liệu video được giải mã.

28. Thiết bị theo điểm 24, trong đó thiết bị này bao gồm một hoặc nhiều trong số camera, máy tính, thiết bị di động, thiết bị thu phát quảng bá, hoặc thiết bị giải mã tín hiệu truyền hình.

29. Phương tiện lưu trữ bất biến đọc được bằng máy tính lưu trữ trên đó các lệnh mà, khi được thực thi, khiến cho bộ xử lý của thiết bị lập mã dữ liệu video thực hiện:

lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất;

dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, MVD thứ hai biểu diễn sự chênh lệch thứ hai giữa vector chuyển động thứ hai và MVP thứ hai cho vector chuyển động thứ hai, vector chuyển động thứ hai là khác với vector chuyển động thứ nhất; và

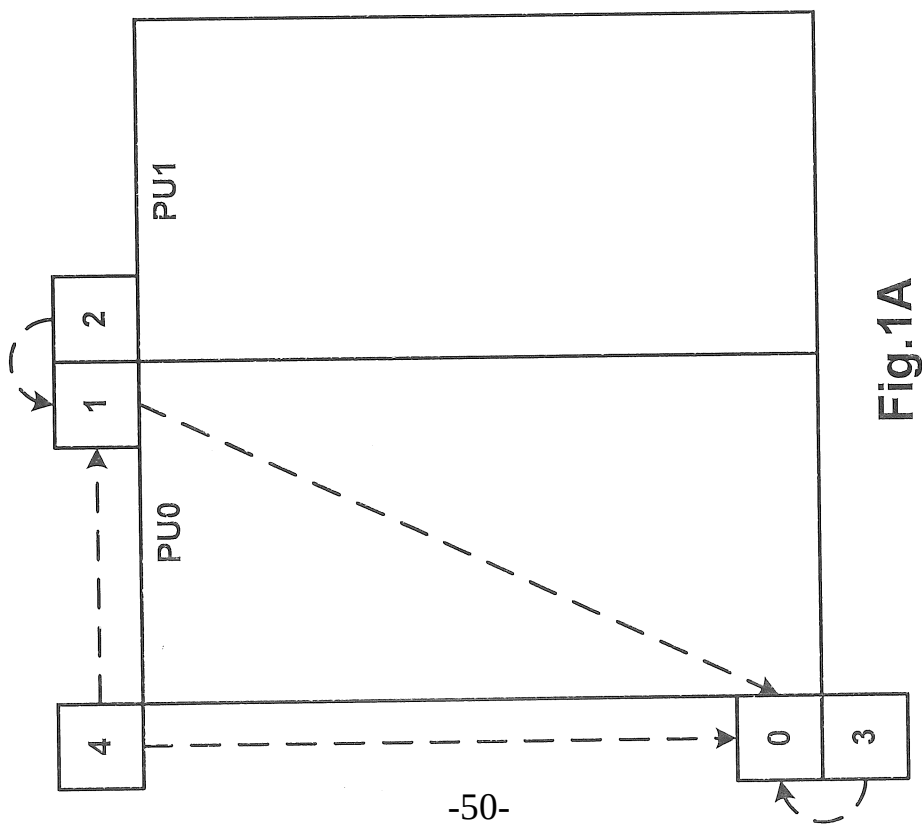
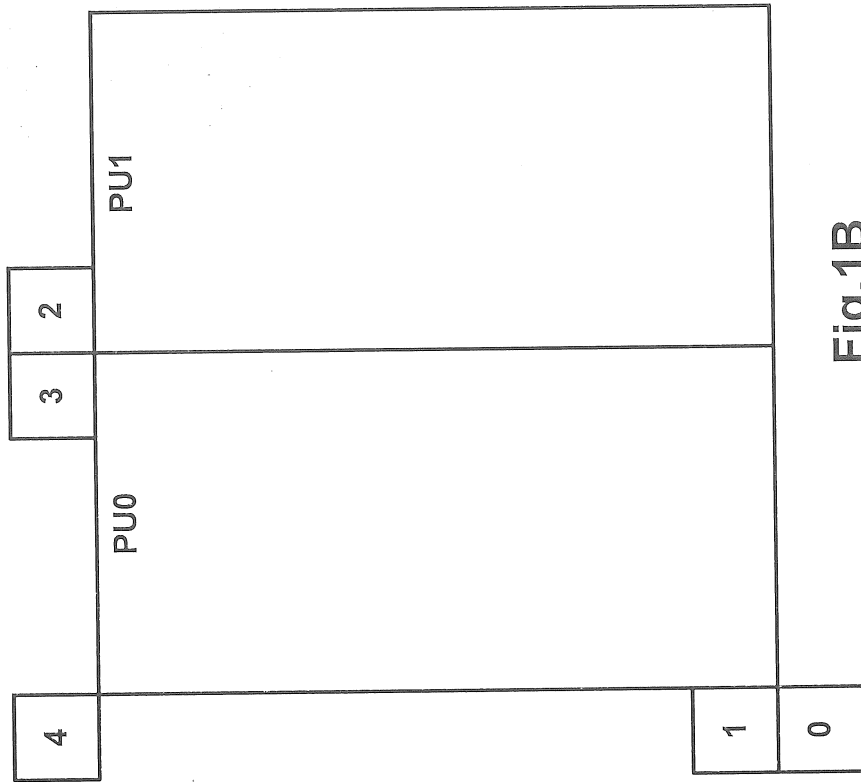
lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.

30. Thiết bị lập mã dữ liệu video, thiết bị này bao gồm:

phương tiện lập mã chênh lệch vector chuyển động (MVD) thứ nhất biểu diễn chênh lệch giữa vector chuyển động thứ nhất của khối dữ liệu video hiện thời được dự đoán bằng cách sử dụng dự đoán afin và bộ dự đoán vector chuyển động (MVP) thứ nhất cho vector chuyển động thứ nhất;

phương tiện dự đoán MVD thứ hai từ MVD thứ nhất cho vector chuyển động thứ hai của khối hiện thời, MVD thứ hai biểu diễn sự chênh lệch thứ hai giữa vector chuyển động thứ hai và MVP thứ hai cho vector chuyển động thứ hai, vector chuyển động thứ hai là khác với vector chuyển động thứ nhất; và

phương tiện lập mã khối hiện thời bằng cách sử dụng dự đoán afin theo vector chuyển động thứ nhất và vector chuyển động thứ hai.



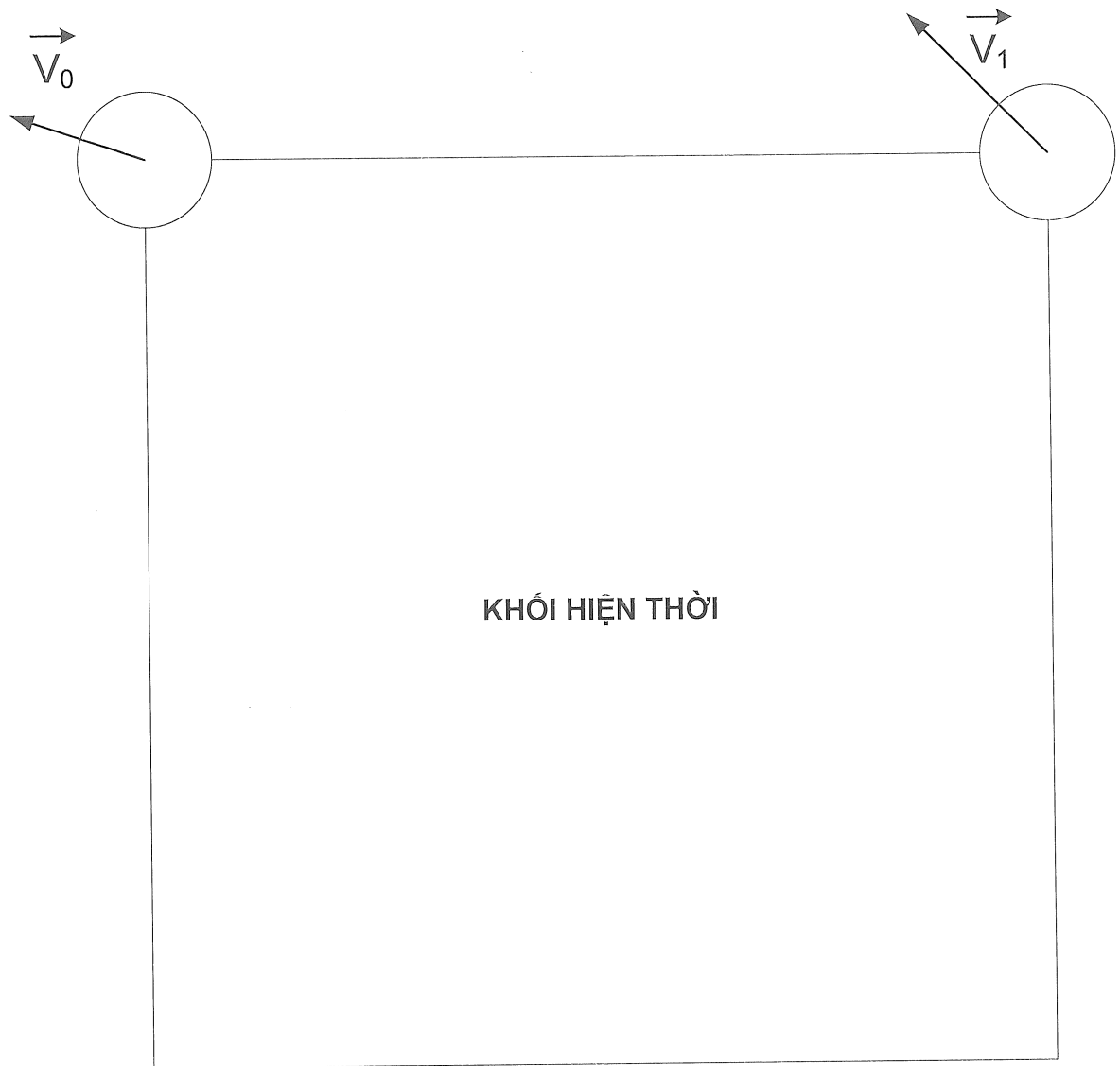


Fig.2



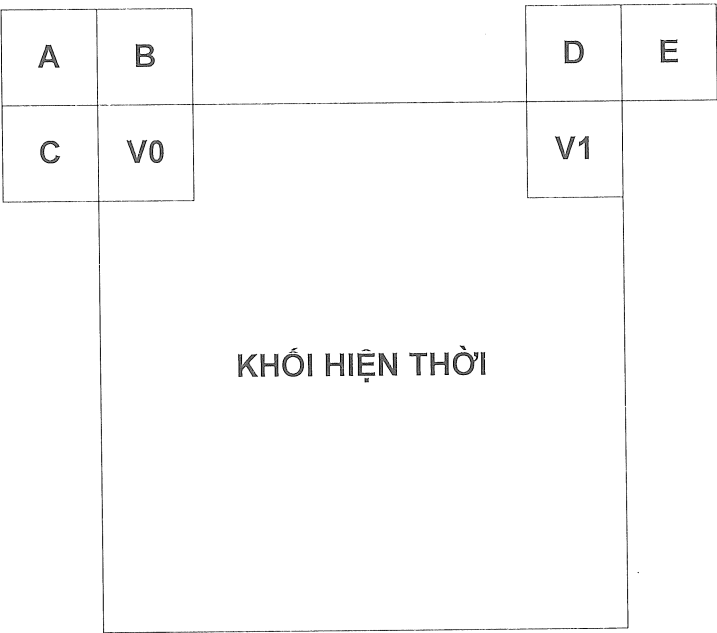


Fig.4A

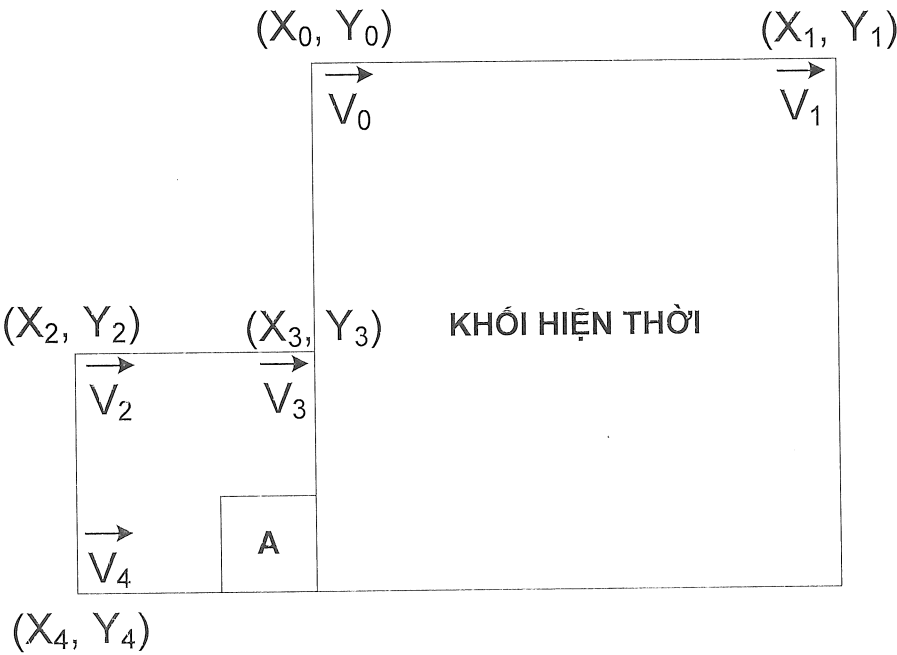
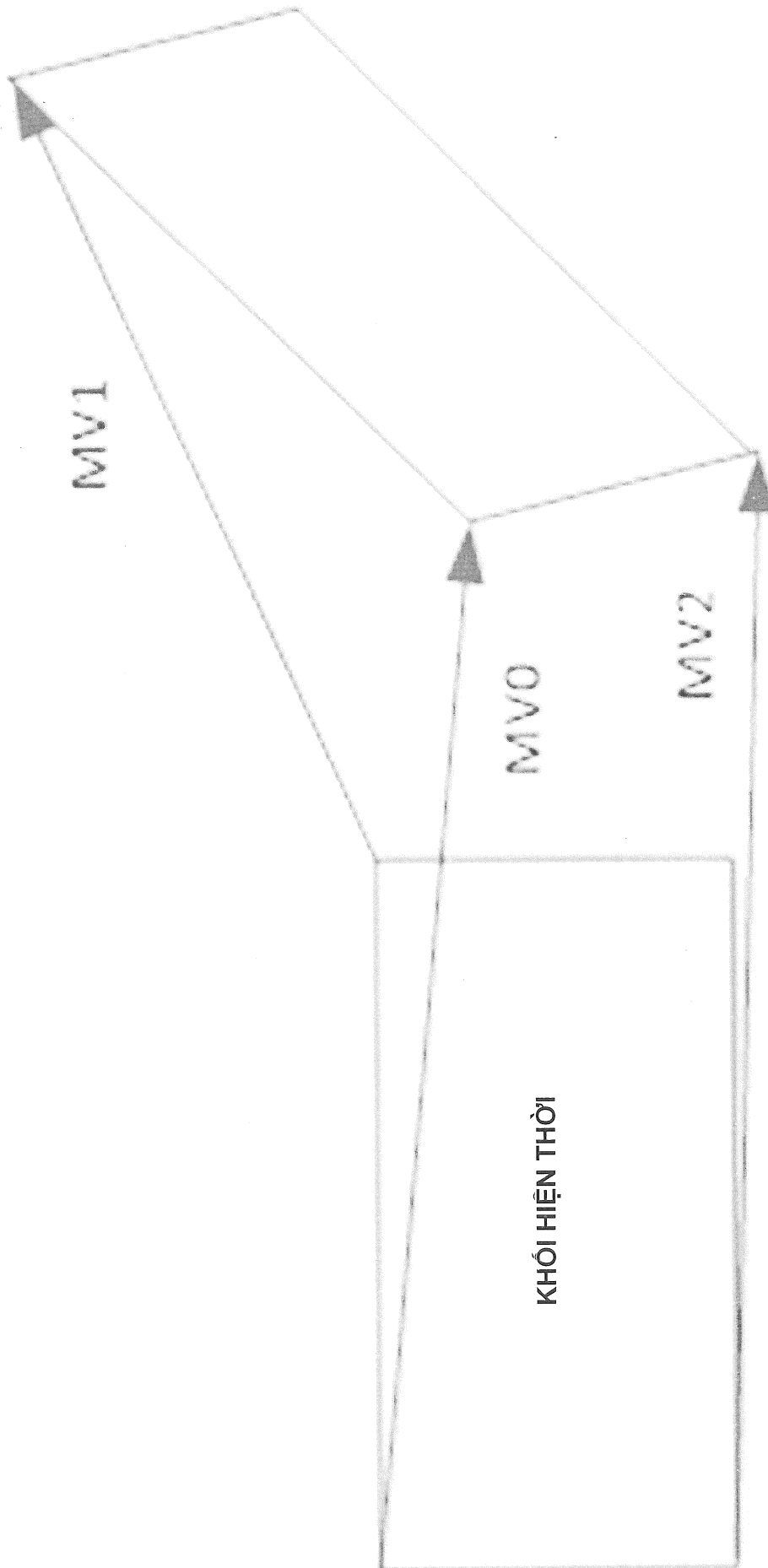


Fig.4B



6 / 12

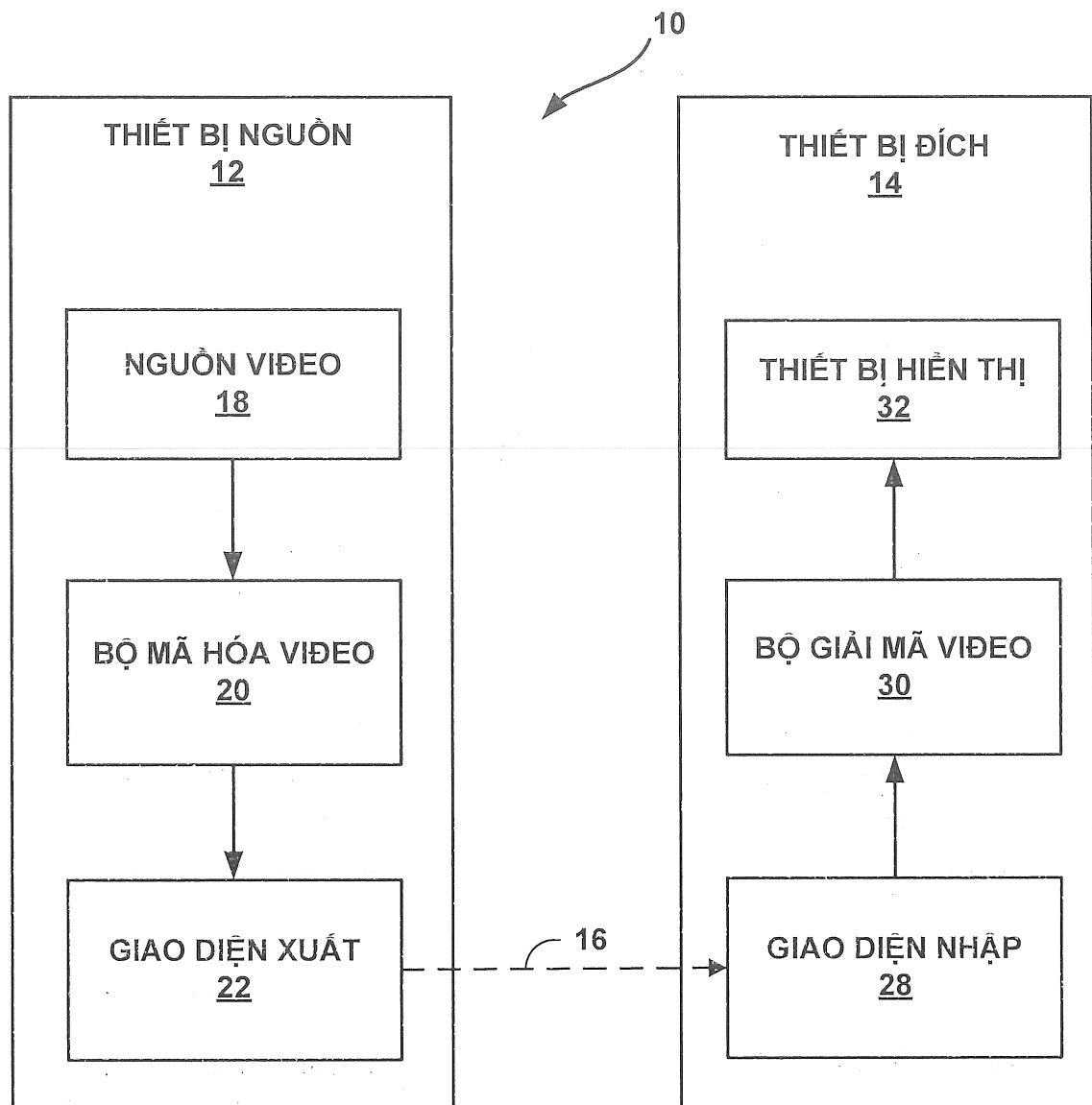


Fig.6

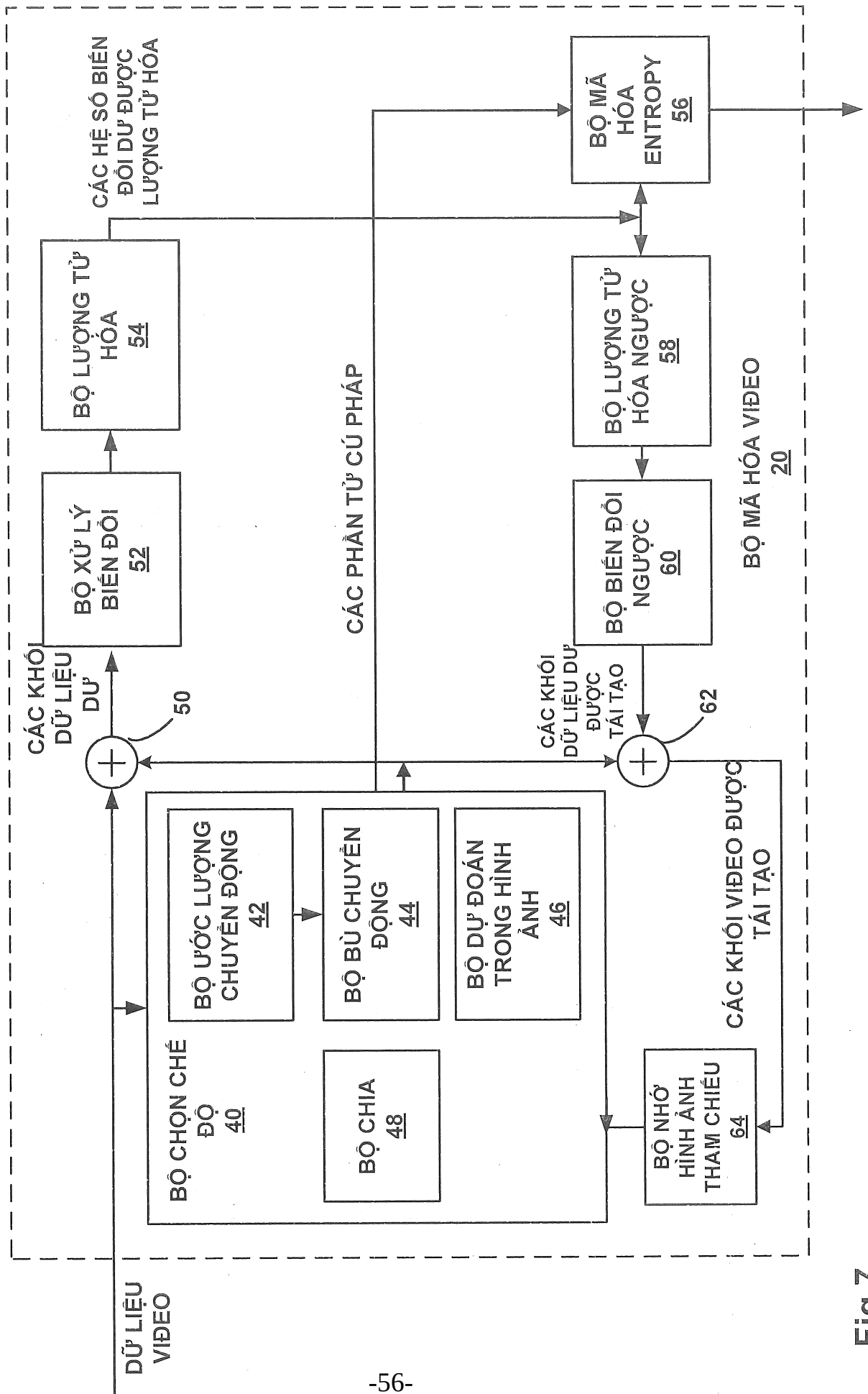


Fig.7

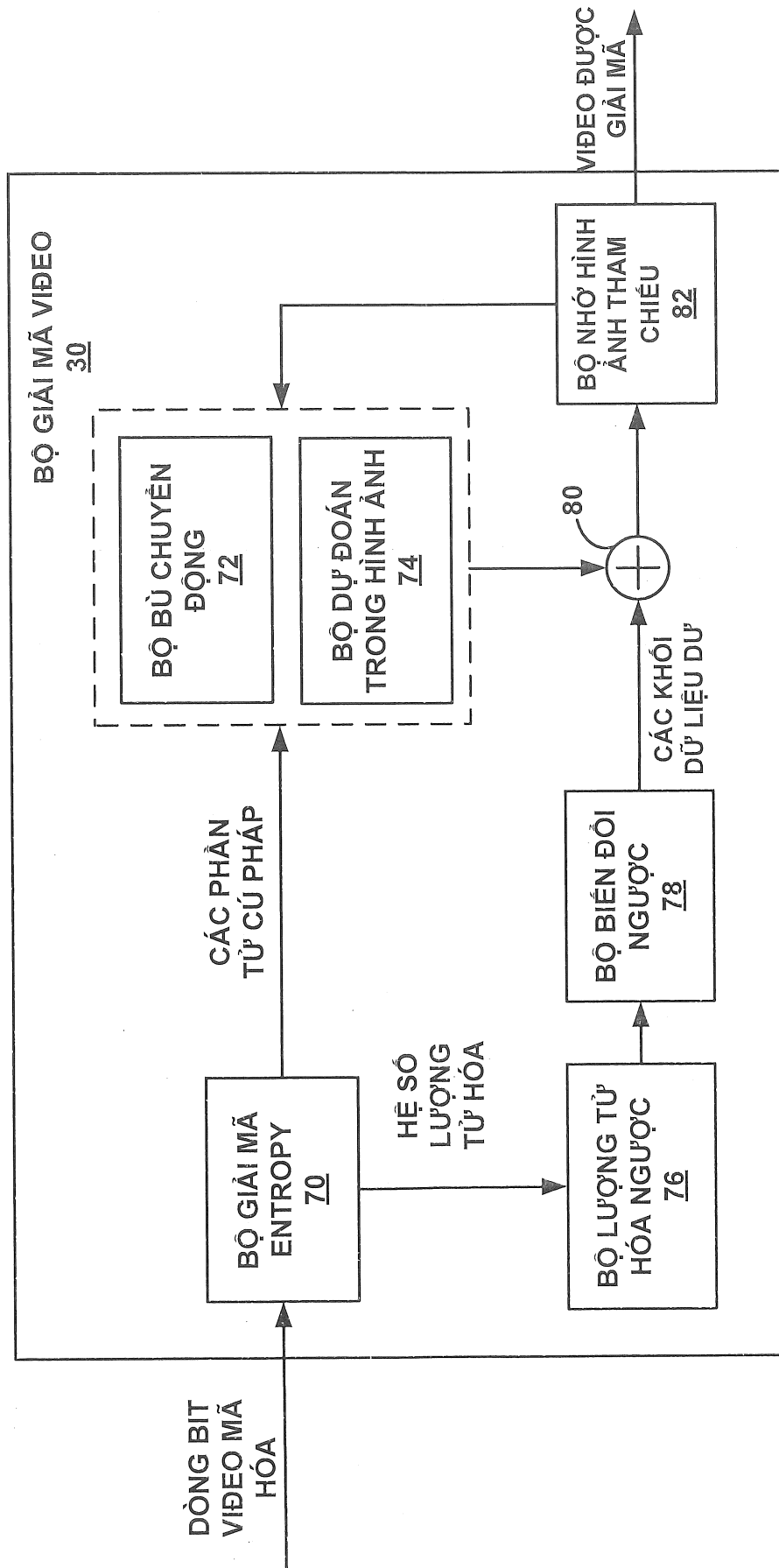


Fig.8

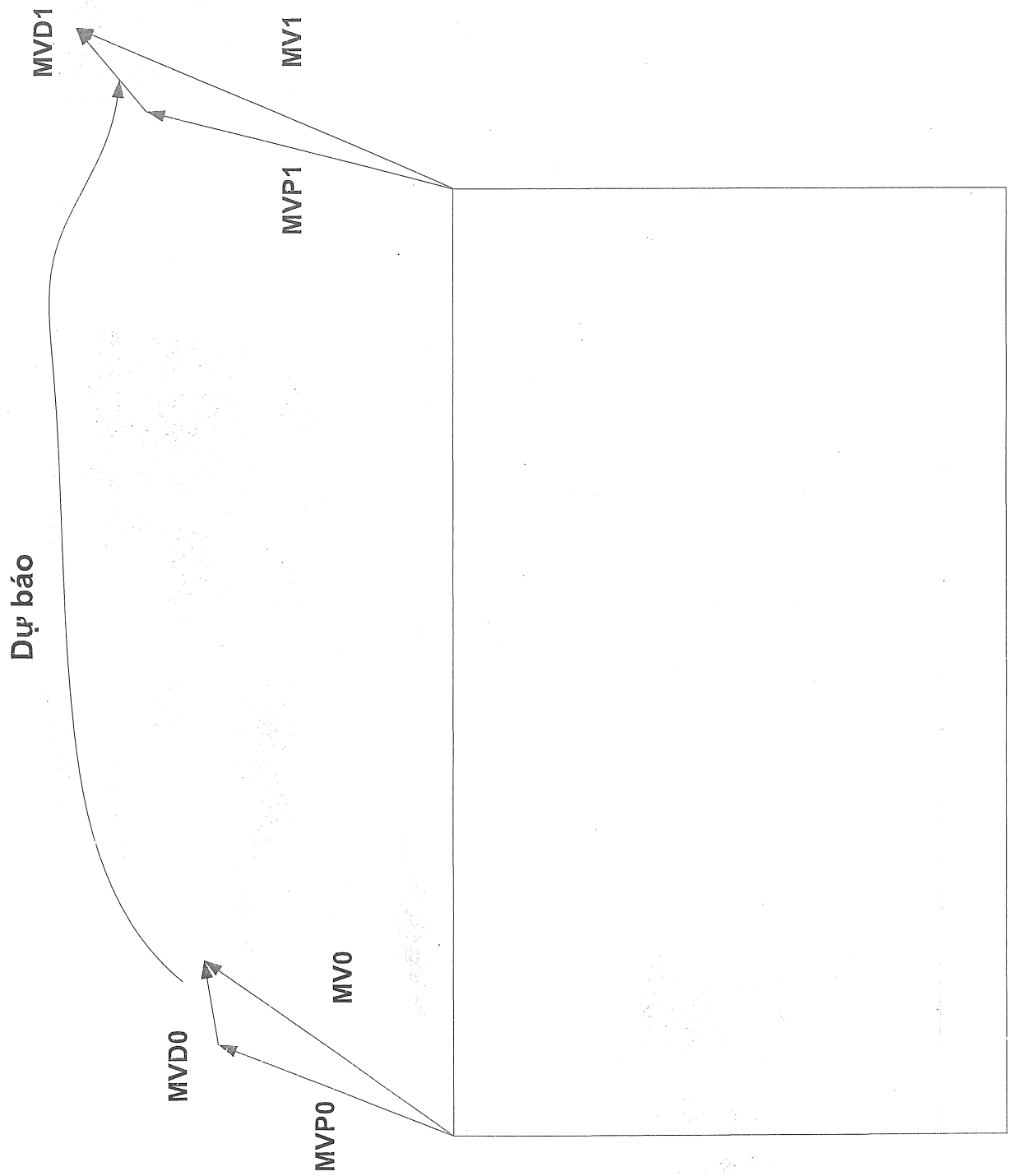


Fig.9

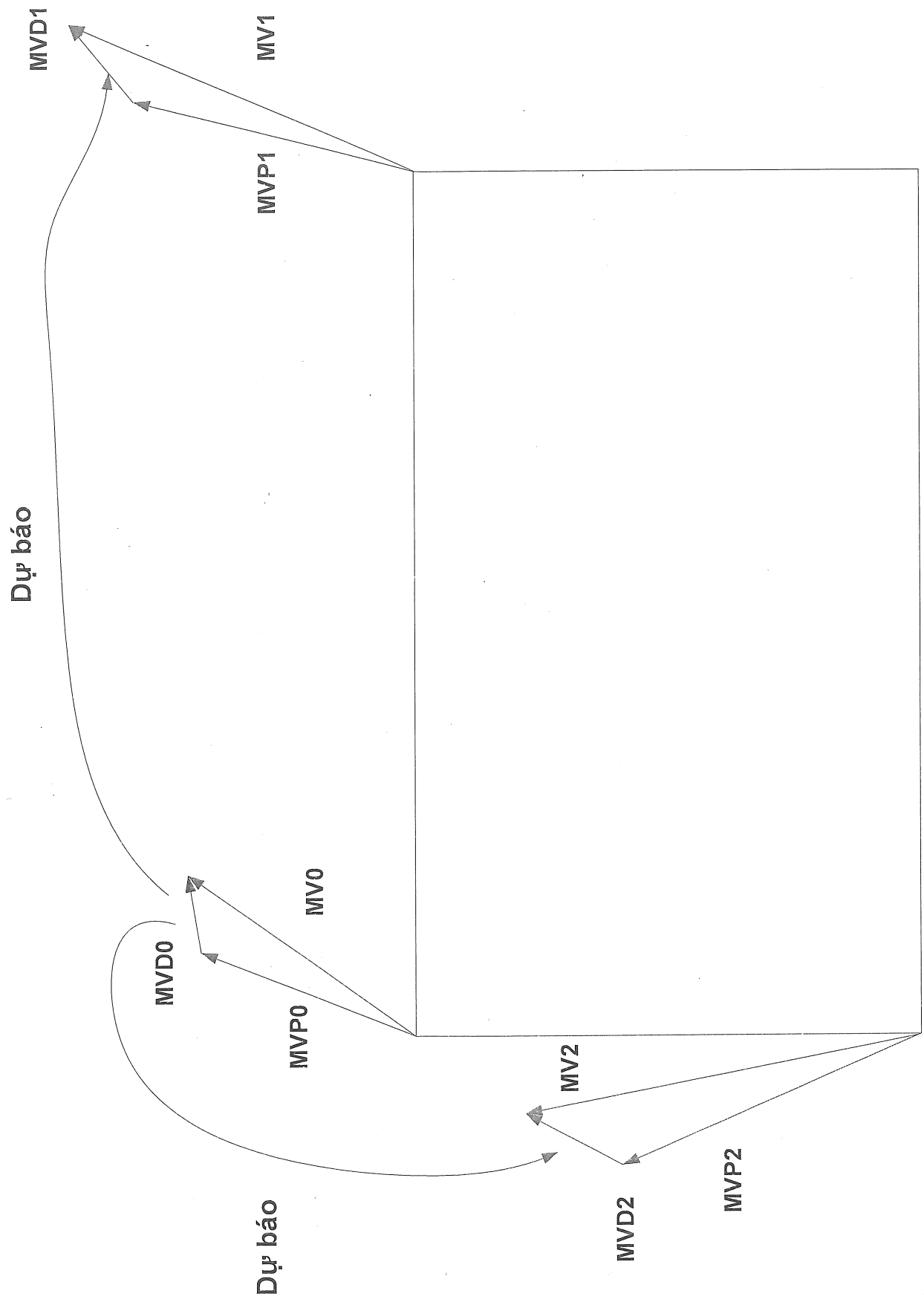


Fig.10

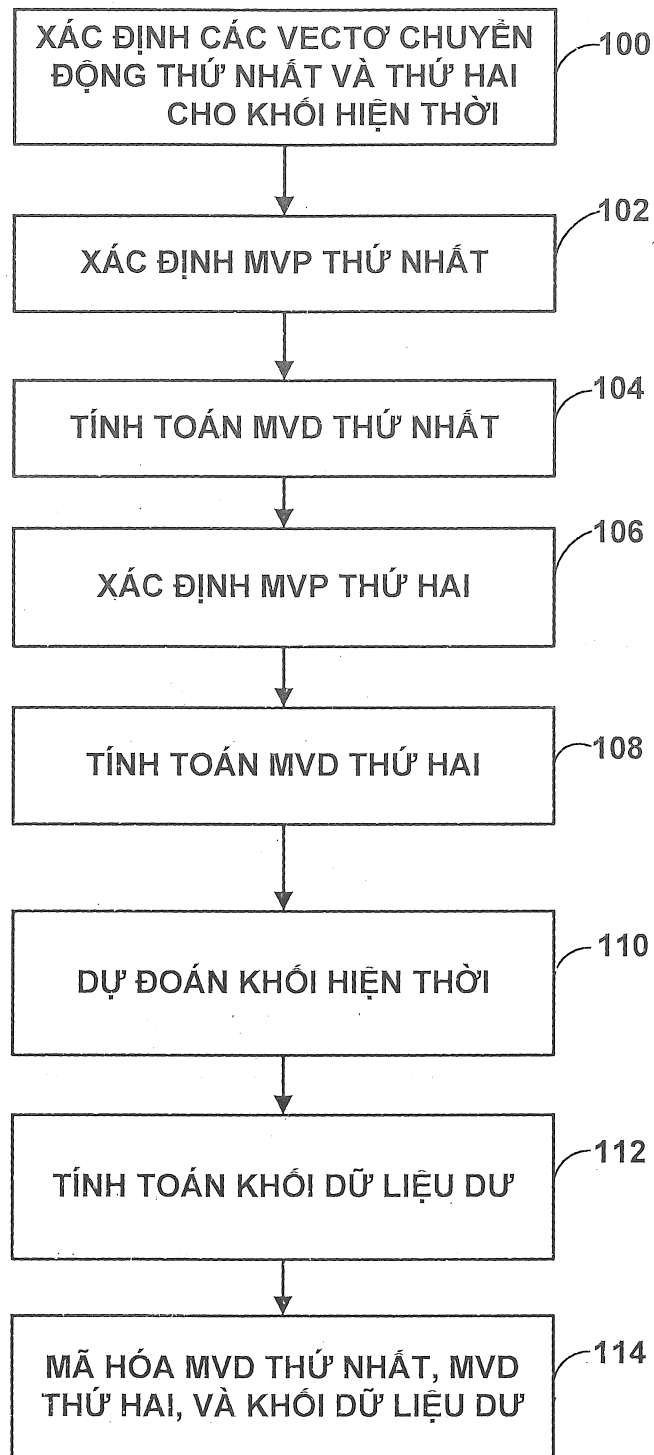


Fig.11

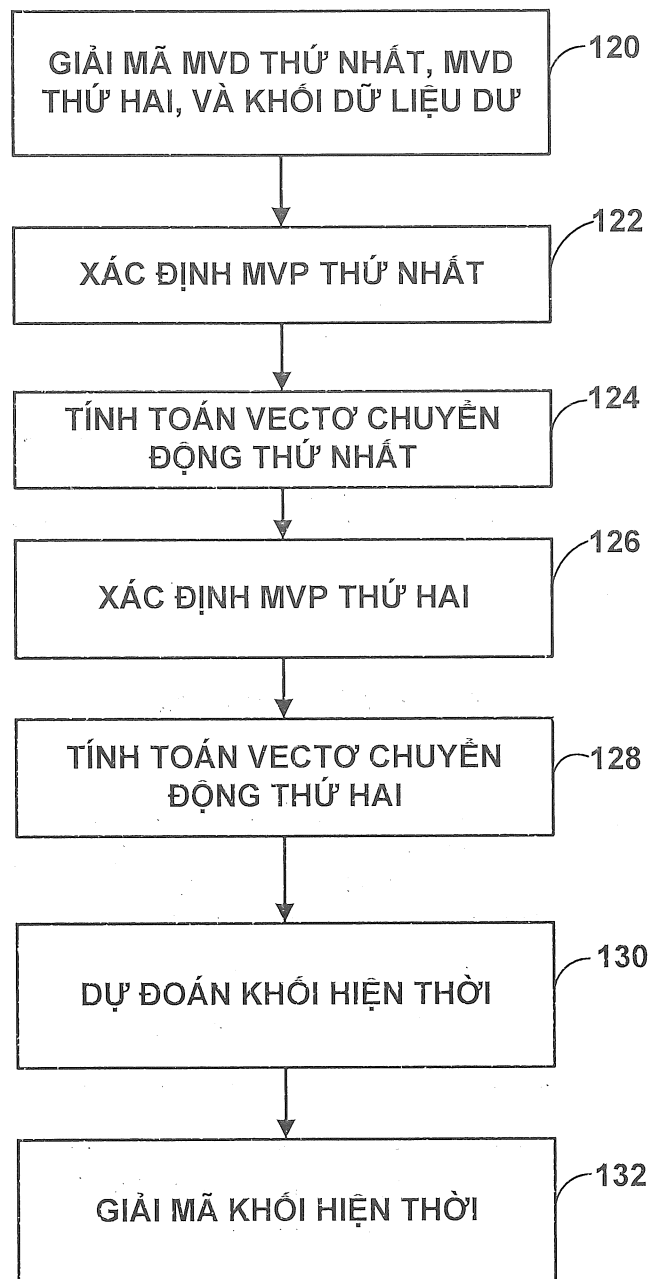


Fig.12