



(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)



1-0053241

(51)^{2020.01} H04N 19/117; H04N 19/176; H04N
19/86; H04N 19/187; H04N 19/80;
H04N 19/157; H04N 19/182

(13) B

(21) 1-2021-00801

(22) 27/08/2019

(86) PCT/US2019/048353 27/08/2019

(87) WO2020/046943 05/03/2020

(30) 62/723,408 27/08/2018 US; 62/742,331 06/10/2018 US; 16/551,388 26/08/2019 US

(45) 25/11/2025 452

(43) 25/05/2021 398A

(73) QUALCOMM INCORPORATED (US)

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

(72) RUSANOVSKYY, Dmytro (UA).

(74) Công ty TNHH Quốc tế D & N (D&N INTERNATIONAL CO.,LTD.)

(54) PHƯƠNG PHÁP, THIẾT BỊ VÀ MÁY ĐỂ GIẢI MÃ DỮ LIỆU VIDEO VÀ
PHƯƠNG TIỆN LƯU TRỮ ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(21) 1-2021-00801

(57) Sáng chế đề cập đến phương pháp, thiết bị và máy để giải mã dữ liệu video và phương tiện lưu trữ đọc được bằng máy tính. Bộ giải mã dữ liệu video được tạo cấu hình để thu được khối tái tạo thứ nhất của dữ liệu video bao gồm mẫu thứ nhất với trị số thứ nhất; áp dụng lọc tách khối vào khối tái tạo thứ nhất để tạo ra khối lọc tách khối; xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo và dựa vào kích thước của khối tái tạo; so sánh giá trị cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất gây ra bởi việc lọc tách khối; để đáp lại lượng sửa đổi cho mẫu thứ nhất gây ra bởi việc lọc tách khối là lớn hơn giá trị cắt thứ nhất, sửa đổi giá trị thứ nhất bởi giá trị cắt thứ nhất để xác định giá trị lọc thứ nhất cho mẫu thứ nhất; và xuất ra khối lọc tách khối của dữ liệu video với mẫu thứ nhất có giá trị lọc thứ nhất.

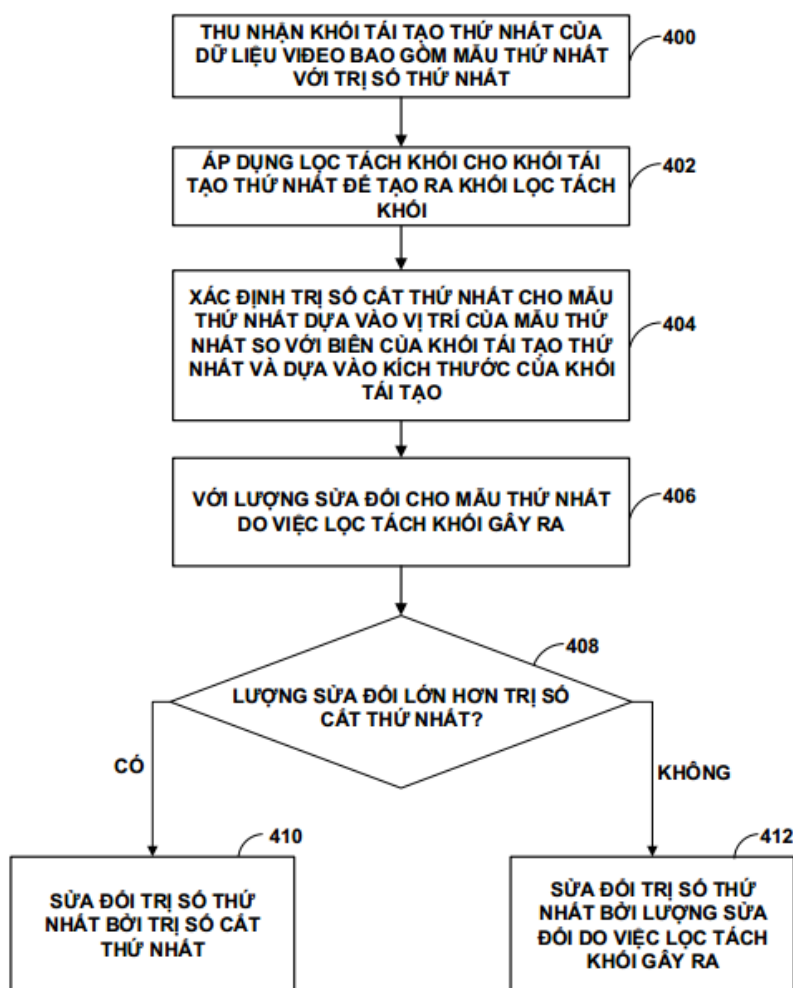


FIG.11

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến việc mã hóa và giải mã 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 thu hình số, hệ thống phát quảng bá trực tiếp số, hệ thống phát quảng bá không dây, thiết bị hỗ trợ số 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 số, thiết bị ghi số, máy phát phương tiện 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, thiết bị được gọi là “điện thoại thông minh”, thiết bị hội thảo từ xa có truyền hình, thiết bị truyền trực tiếp video, và thiết bị tương tự. Các thiết bị video số thực hiện các kỹ thuật lập mã dữ liệu video, chẳng hạn như các kỹ thuật được mô tả theo các chuẩn được quy định bởi MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Phần 10, mã hóa dữ liệu video cải tiến (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 hiệu quả hơn bằng cách thực hiện các kỹ thuật mã hóa dữ liệu video như vậy.

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

Bản chất kỹ thuật của sáng chế

Sáng chế đề cập đến các kỹ thuật liên quan đến việc lọc dữ liệu video tái tạo trong quá trình mã hóa dữ liệu video và/hoặc giải mã dữ liệu video và, cụ thể hơn, sáng chế này đề cập đến các kỹ thuật liên quan đến các kỹ thuật để thực hiện lọc tách khối. Lọc tách khối là loại lọc có thể giảm các xảo ảnh dạng khối theo các cạnh của các khối mà tạo ra từ quy trình tái tạo.

Theo một ví dụ, phương pháp giải mã dữ liệu video bao gồm bước thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất có trị số thứ nhất; áp dụng kỹ thuật lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối; xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất; so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra; đáp lại lượng sửa đổi cho mẫu thứ nhất do lọc tách khối gây ra lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bằng trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và xuất ra khối lọc tách khối của dữ liệu video, trong đó trong khối lọc tách khối, mẫu thứ nhất có trị số lọc thứ nhất.

Theo một ví dụ khác, thiết bị 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 bố trí trong mạch và được tạo cấu hình để thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất có trị số thứ nhất; áp dụng kỹ thuật lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối; xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất; so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra; để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và xuất ra khối lọc tách khối của dữ liệu video, trong đó trong khối lọc tách khối, mẫu thứ nhất có trị số lọc thứ nhất.

Theo một ví dụ khác, phương tiện đọc được bằng máy tính lưu trữ các lệnh khi được thực thi bởi một hoặc nhiều bộ xử lý khiến cho một hoặc nhiều bộ xử lý thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất có trị số thứ nhất; áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối; xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất; so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra; đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và xuất ra khối lọc tách khối của dữ liệu video, trong đó trong khối lọc tách khối, mẫu thứ nhất có trị số lọc thứ nhất.

Theo một ví dụ khác, thiết bị giải mã dữ liệu video bao gồm phương tiện để thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất có trị số thứ nhất; phương tiện để áp dụng kỹ thuật lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối; phương tiện để xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất; phương tiện để so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra; phương tiện để sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra lớn hơn trị số cắt thứ nhất; và phương tiện để xuất ra khối lọc tách khối của dữ liệu video, trong đó trong khối lọc tách khối, mẫu thứ nhất có trị số lọc thứ nhất.

Chi tiết của một hoặc nhiều ví dụ được trình bày trong các hình vẽ kèm theo và phần mô tả bên dưới. Các đặc điểm, mục đích và ưu điểm khác sẽ trở nên rõ ràng nhờ phần mô tả, các hình vẽ và các yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối minh họa ví dụ về hệ thống mã hóa và giải mã dữ liệu video có thể thực hiện các kỹ thuật của sáng chế.

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

Fig.3 thể hiện ví dụ về đoạn bốn mẫu cần tách khỏi với các khối liền kề P và Q.

Fig.4 thể hiện ví dụ về khối dữ liệu video, trong đó các đạo hàm bậc hai của bốn mẫu tuần hoàn được tính toán, trong đó trị số tổng được dùng để xác định xem việc tách khối có nên được áp dụng trên đoạn hay không.

Fig.5 thể hiện ví dụ về các mẫu của khối dữ liệu video cho một trong bốn dòng trong đoạn cần tách khỏi.

Fig.6 là sơ đồ khối minh họa bộ mã hóa dữ liệu video làm ví dụ có thể thực hiện các kỹ thuật của sáng chế.

Fig.7 là sơ đồ khối minh họa bộ giải mã dữ liệu video làm ví dụ có thể thực hiện các kỹ thuật của sáng chế.

Fig.8 là sơ đồ khối minh họa đơn vị lọc làm ví dụ để thực hiện các kỹ thuật của sáng chế này.

Fig.9 là lưu đồ minh họa một ví dụ về quy trình mã hóa dữ liệu video.

Fig.10 là lưu đồ minh họa một ví dụ về quy trình giải mã dữ liệu video.

Fig.11 là lưu đồ minh họa một ví dụ về quy trình giải mã dữ liệu video.

Mô tả chi tiết sáng chế

Mã hóa dữ liệu video (ví dụ, mã hóa dữ liệu video và/hoặc giải mã dữ liệu video) thường liên quan đến việc dự đoán khối dữ liệu video từ khối dữ liệu video đã mã hóa trong cùng một hình ảnh (ví dụ, dự đoán nội ảnh) hoặc khối dữ liệu video đã lập mã trong hình ảnh khác (ví dụ, dự đoán liên ảnh). Trong một số trường hợp, bộ mã hóa dữ liệu video còn tính toán dữ liệu dư bằng cách so sánh khối dự đoán với khối ban đầu. Do đó, dữ liệu dư thể hiện sự khác biệt giữa khối dự đoán và khối ban đầu. Để giảm số lượng bit cần được báo hiệu dữ liệu dư, bộ mã hóa dữ liệu video biến đổi và lượng tử hóa dữ liệu dư và báo hiệu dữ liệu dư được biến đổi và được lượng tử hóa trong luồng bit đã mã hóa. Việc nén được thực hiện bởi quy trình biến đổi và lượng tử hóa có thể bị hao hụt, tức là quy trình biến đổi và lượng tử hóa có thể làm biến dạng dữ liệu video đã giải mã.

Bộ giải mã dữ liệu video giải mã và thêm dữ liệu dư vào khối dự đoán để tạo ra khối dữ liệu video tái tạo phù hợp với khối dữ liệu video ban đầu hơn so với khối dự đoán. Vì tổn hao do biến đổi và lượng tử hóa dữ liệu dư, khối tái tạo thứ nhất có thể có độ méo hoặc xáo ảnh. Một loại xáo ảnh hoặc độ méo thông thường được gọi là có hình khối, trong đó các biên của các khối được dùng để lập mã dữ liệu video có thể nhìn thấy được.

Để cải thiện thêm chất lượng video giải mã, bộ giải mã dữ liệu video có thể thực hiện một hoặc nhiều hoạt động lọc trên các khối dữ liệu video tái tạo. Ví dụ về các hoạt động lọc này bao gồm lọc tách khối, lọc bù thích ứng mẫu (sample adaptive offset - SAO) và lọc vòng lặp thích ứng. Các thông số cho các hoạt động lọc này có thể được xác định bởi bộ mã hóa dữ liệu video và được báo hiệu một cách rõ ràng trong luồng bit video mã hóa hoặc có thể được ngầm định bởi bộ giải mã dữ liệu video mà không cần các thông số được báo hiệu một cách rõ ràng trong luồng bit video mã hóa.

Sáng chế này mô tả các kỹ thuật liên quan đến việc lọc dữ liệu video tái tạo trong quá trình mã hóa dữ liệu video và/hoặc giải mã dữ liệu video và, cụ thể hơn, sáng chế này mô tả các kỹ thuật liên quan đến việc lọc tách khối. Lọc tách khối là kiểu lọc được thiết kế riêng để giảm hình khối. Sáng chế này đề cập đến các kỹ thuật liên quan đến quy trình lọc thực hiện trên khung hoặc hình video, như khung hoặc hình video có thể bị méo do nén, làm nhòe, v.v., nhưng các kỹ thuật không được coi là hạn chế đối với các ví dụ méo ở trên. Các kỹ thuật có thể cải thiện chất lượng chủ quan và khách quan của dữ liệu video. Các kỹ thuật được mô tả ở đây có thể được dùng trong việc thiết kế các giải pháp lập mã dữ liệu video mới, như H.266, hoặc để mở rộng bất kỳ trong số các bộ mã hóa giải mã (codec - coder/decoder) dữ liệu video hiện có, như H.265/mã hóa dữ liệu video hiệu suất cao (HEVC), hoặc có thể đề xuất làm công cụ mã hóa đầy triển vọng cho các chuẩn lập mã dữ liệu video tương lai. Các kỹ thuật được mô tả cũng có thể được sử dụng làm phương pháp xử lý sau trên các khung dữ liệu video xuất ra từ các bộ mã hóa - giải mã tiêu chuẩn hoặc độc quyền.

Các phiên bản hiện tại của HEVC sử dụng kích thước khối tối đa 64x64. Tuy nhiên, các chuẩn lập mã dữ liệu video tương lai, như chuẩn Mã hóa dữ liệu video đa năng hiện đang phát triển có thể sử dụng các kích thước khối lớn hơn so với kích thước khối 64x64. Sáng chế này đề cập đến các kỹ thuật có thể cải thiện việc giảm độ méo thu được từ quy trình lọc tách khối, đặc biệt là kết hợp với kích thước khối lớn hơn. Hơn nữa, các kỹ thuật

được mô tả ở đây có thể thu được hiệu quả giảm độ méo trong khi vẫn duy trì các mức độ phức tạp tính toán và hiệu quả lập mã mong muốn.

Như được dùng trong bản mô tả này, thuật ngữ mã hóa dữ liệu video nói chung chỉ việc mã hóa dữ liệu video hoặc giải mã dữ liệu video. Tương tự, thuật ngữ bộ lập mã dữ liệu video thường dùng để chỉ bộ mã hóa dữ liệu video hoặc bộ giải mã dữ liệu video. Hơn nữa, một số kỹ thuật mô tả trong sáng chế này liên quan đến việc giải mã dữ liệu video cũng có thể áp dụng cho mã hóa dữ liệu video và ngược lại. Ví dụ, thường thì các bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video được tạo cấu hình để thực hiện cùng một quy trình, hoặc các quy trình tương hỗ. Ngoài ra, bộ mã hóa dữ liệu video thường thực hiện giải mã dữ liệu video (còn gọi là sự tái tạo) là một phần của quy trình xác định cách để mã hóa dữ liệu video. Ví dụ, bộ mã hóa dữ liệu video có thể thực hiện lọc tách khối trên khối dữ liệu video được giải mã để xác định xem sơ đồ mã hóa nào tạo ra cân bằng tốc độ méo mong muốn và còn để bộ mã hóa dữ liệu video có thể thực hiện ước lượng chuyển động bằng cách sử dụng các khối giống nhau có sẵn để bộ giải mã dữ liệu video khi bộ giải mã dữ liệu video thực hiện bù chuyển động.

Fig.1 là sơ đồ khối minh họa hệ thống mã hóa và giải mã dữ liệu video 100 làm ví dụ mà có thể thực hiện các kỹ thuật lọc tách khối của sáng chế. Các kỹ thuật của sáng chế này nói chung hướng đến việc lập mã (mã hóa và/hoặc giải mã) dữ liệu video, và cụ thể hơn, đến các kỹ thuật để lọc tách khối dữ liệu video. Nói chung, dữ liệu video bao gồm dữ liệu bất kỳ để xử lý video. Do đó, dữ liệu video có thể bao gồm video thô, video chưa lập mã, video đã mã hóa, video đã giải mã (ví dụ, được tái tạo), và siêu dữ liệu video, như dữ liệu báo hiệu chẳng hạn.

Như được thể hiện trên Fig.1, hệ thống 100 bao gồm thiết bị nguồn 102 cung cấp dữ liệu video đã mã hóa cần được giải mã và được hiển thị bởi thiết bị đích 116, theo ví dụ này. Cụ thể, thiết bị nguồn 102 cung cấp dữ liệu video cho thiết bị đích 116 thông qua phương tiện đọc được bằng máy tính 110. Thiết bị nguồn 102 và thiết bị đích 116 có thể là thiết bị bất kỳ trong số rất nhiều thiết bị, bao gồm máy tính để bàn, máy tính sở tay (tức là, máy tính xách tay), máy tính bảng, đầu thu kỹ thuật số, các máy điện thoại cầm tay như điện thoại "thông minh", ti vi, camera, thiết bị hiển thị, trình phát phương tiện kỹ thuật số, bàn điều khiển trò chơi điện tử, thiết bị truyền trực tiếp dữ liệu video hoặc các thiết bị tương tự khác. Trong một số trường hợp, thiết bị nguồn 102 và thiết bị đích 116 có thể

được trang bị để truyền thông không dây, và do đó có thể được gọi là các thiết bị truyền thông không dây.

Theo ví dụ trên Fig.1, thiết bị nguồn 102 bao gồm nguồn video 104, bộ nhớ 106, bộ mã hóa dữ liệu video 200, và giao diện đầu ra 108. Thiết bị đích 116 bao gồm giao diện đầu vào 122, bộ giải mã dữ liệu video 300, bộ nhớ 120, và thiết bị hiển thị 118. Theo sáng chế này, bộ mã hóa dữ liệu video 200 của thiết bị nguồn 102 và/hoặc bộ giải mã dữ liệu video 300 của thiết bị đích 116 có thể được tạo cấu hình để áp dụng các kỹ thuật để lọc tách khỏi được mô tả trong sáng chế này. Do đó, thiết bị nguồn 102 là ví dụ về thiết bị mã hóa dữ liệu video, còn thiết bị đích 116 là ví dụ về thiết bị giải mã dữ liệu video. 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 102 có thể nhận dữ liệu video từ nguồn video bên ngoài, như camera bên ngoài. Tương tự, thiết bị đích 116 có thể giao tiếp với thiết bị hiển thị bên ngoài, chứ không phải bao gồm thiết bị hiển thị tích hợp.

Hệ thống 100 được thể hiện trên Fig.1 chỉ là một ví dụ. Nói chung, thiết bị mã hóa và/hoặc giải mã dữ liệu video số bất kỳ có thể thực hiện các kỹ thuật để lọc tách khỏi được mô tả theo sáng chế này. Thiết bị nguồn 102 và thiết bị đích 116 chỉ là ví dụ về các thiết bị lập mã như vậy, trong đó thiết bị nguồn 102 tạo ra dữ liệu video được lập mã để truyền đến thiết bị đích 116. Sáng chế này đề cập đến thiết bị “lập mã” là thiết bị thực hiện lập mã (mã hóa và/hoặc giải mã) dữ liệu. Do đó, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 lần lượt là các ví dụ của thiết bị mã hóa, cụ thể là bộ mã hóa dữ liệu video và bộ giải mã dữ liệu video. Theo một số ví dụ, thiết bị nguồn 102 và thiết bị đích 116 có thể hoạt động theo cách gần như đối xứng sao cho mỗi thiết bị nguồn 102 và thiết bị đích 116 đều có các bộ phận mã hóa và giải mã dữ liệu video. Do đó, hệ thống 100 có thể hỗ trợ cuộc truyền dữ liệu video một chiều hoặc hai chiều giữa thiết bị nguồn 102 và thiết bị đích 116, ví dụ, để truyền trực tiếp dữ liệu video, phát lại dữ liệu video, phát quảng bá dữ liệu video, hoặc điện thoại có video.

Nói chung, nguồn video 104 là nguồn dữ liệu video (tức là, dữ liệu video thô, chưa được mã hóa) và cung cấp chuỗi các hình ảnh liên tiếp (còn được gọi là "khung") của dữ liệu video cho bộ mã hóa dữ liệu video 200, để mã hóa dữ liệu cho các hình ảnh. Nguồn dữ liệu video 104 của thiết bị nguồn 102 có thể bao gồm thiết bị quay video, như máy quay video, video lưu trữ chứa video thô đã quay trước đó, và/hoặc giao diện nạp dữ liệu video

để nhận dữ liệu video từ nhà cung cấp nội dung video. Theo một phương án khác, nguồn dữ liệu video 104 có thể tạo ra dữ liệu dựa trên đồ họa từ máy tính dưới dạng video nguồn, hoặc kết hợp của dữ liệu video trực tiếp, video đã lưu trữ, và video được tạo ra trên- máy tính. Trong mỗi trường hợp, bộ mã hóa dữ liệu video 200 mã hóa dữ liệu video được chụp, được chụp trước hoặc được máy tính tạo ra. Bộ mã hóa dữ liệu video 200 có thể sắp xếp lại các hình ảnh từ trình tự nhận được (đôi khi được gọi là “trình tự hiển thị”) thành trình tự lập mã để lập mã. Bộ mã hóa dữ liệu video 200 có thể tạo ra luồng bit bao gồm dữ liệu video được mã hóa. Thiết bị nguồn 102 sau đó có thể xuất ra dữ liệu video được mã hóa qua giao diện đầu ra 108 lên phương tiện đọc được bằng máy tính 110 để nhận và/hoặc phục hồi bằng, ví dụ, giao diện đầu vào 122 của thiết bị đích 116.

Bộ nhớ 106 của thiết bị nguồn 102 và bộ nhớ 120 của thiết bị đích 116 là các bộ nhớ đa năng. Theo một số ví dụ, các bộ nhớ 106, 120 có thể lưu trữ dữ liệu video thô, ví dụ, dữ liệu video thô từ nguồn dữ liệu video 104 và dữ liệu video thô, được giải mã từ bộ giải mã dữ liệu video 300. Ngoài ra hoặc theo cách khác, các bộ nhớ 106, 120 có thể lưu trữ các lệnh phần mềm có thể thực thi được bằng, ví dụ, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300, tương ứng. Mặc dù các bộ nhớ 106 và 120 được thể hiện riêng với bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 theo ví dụ này, nhưng cần phải hiểu rằng bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 cũng có thể bao gồm các bộ nhớ trong cho các mục đích tương tự hoặc tương đương về mặt chức năng. Hơn nữa, các bộ nhớ 106, 120 có thể lưu trữ dữ liệu video được mã hóa, ví dụ, xuất ra từ bộ mã hóa dữ liệu video 200 và nhập vào bộ giải mã dữ liệu video 300. Theo một số ví dụ, các phần của các bộ nhớ 106, 120 có thể được cấp phát dưới dạng một hoặc nhiều bộ đệm video, ví dụ, để lưu trữ dữ liệu video thô, được giải mã và/hoặc được mã hóa.

Phương tiện đọc được bằng máy tính 110 có thể là loại phương tiện hoặc thiết bị bất kỳ có khả năng truyền dữ liệu video được mã hóa từ thiết bị nguồn 102 đến thiết bị đích 116. Trong một số ví dụ, phương tiện đọc được bằng máy tính 110 là phương tiện truyền thông cho phép thiết bị nguồn 102 truyền dữ liệu video đã mã hóa trực tiếp đến thiết bị đích 116 theo thời gian thực, ví dụ, qua mạng tần số vô tuyến hoặc mạng dựa trên máy tính. Giao diện đầu ra 108 có thể điều chế tín hiệu truyền bao gồm dữ liệu video được mã hóa và giao diện đầu vào 122 có thể giải điều chế tín hiệu truyền thu được, theo chuẩn

truyền thông, như giao thức truyền thông không dây. Phương tiện truyền thông có thể bao gồm một hoặc cả hai phương tiện truyền thông không dây hoặc có dây, như phổ tần số vô tuyến (Radio Frequency - RF) hoặc một hay nhiều đường truyền vật lý. Phương tiện truyền thông có thể tạo nên một phần của mạng dựa trên gói, như mạng cục bộ, mạng diện rộng, hoặc mạng toàn cầu như mạng Internet. Phương tiện truyền thông có thể bao gồm các bộ định tuyến, bộ chuyển mạch, trạm gốc, hoặc mọi thiết bị khác có thể có ích để hỗ trợ truyền thông từ thiết bị nguồn 102 đến thiết bị đích 116.

Theo một số ví dụ, phương tiện đọc được bằng máy tính 110 có thể bao gồm thiết bị lưu trữ 112. Thiết bị nguồn 102 có thể xuất ra dữ liệu được mã hóa từ giao diện đầu ra 108 cho thiết bị lưu trữ 112. Tương tự, thiết bị đích 116 có thể truy cập dữ liệu được mã hóa từ thiết bị lưu trữ 112 thông qua giao diện đầu vào 122. Thiết bị lưu trữ 112 có thể bao gồm phương tiện bất kỳ trong số các phương tiện lưu trữ dữ liệu truy cập phân tán hoặc cục bộ như ổ đĩa cứng, đĩa Blu-ray, DVD, CD-ROM, bộ nhớ đệm nhanh, bộ nhớ khả biến hoặc không khả biến hoặc phương tiện lưu trữ số thích hợp khác bất kỳ để lưu trữ dữ liệu video được mã hóa.

Theo một số ví dụ, phương tiện đọc được bằng máy tính 110 có thể bao gồm máy chủ tệp tin 114 hoặc thiết bị lưu trữ trung gian khác mà có thể lưu trữ dữ liệu video đã mã hóa được tạo ra bởi thiết bị nguồn 102. Thiết bị nguồn 102 có thể xuất ra dữ liệu video được mã hóa cho máy chủ tệp tin 114. Thiết bị đích 116 có thể truy cập dữ liệu video đã lưu trữ từ máy chủ tệp tin 114 qua truyền liên tục hoặc tải xuống. Máy chủ tệp tin 114 có thể là loại thiết bị máy chủ bất kỳ có khả năng lưu trữ dữ liệu video được mã hóa và truyền dữ liệu video được mã hóa đó đến thiết bị đích 116. Máy chủ tệp tin 114 có thể là máy chủ web (ví dụ, cho trang web), máy chủ giao thức truyền tệp tin (File Transfer Protocol - FTP), thiết bị mạng phân phối nội dung, hoặc thiết bị lưu trữ gắn với mạng (network attached storage - NAS). Thiết bị đích 116 có thể truy cập dữ liệu video được mã hóa từ máy chủ tệp tin 114 qua kết nối dữ liệu chuẩn bất kỳ, bao gồm kết nối Internet. Kết nối này có thể bao gồm kênh không dây (ví dụ, kết nối Wi-Fi), kết nối có dây (ví dụ, đường thuê bao kỹ thuật số (Digital Subscriber Line - DSL), modem cáp, v.v.), hoặc kết hợp hai loại này phù hợp với việc truy cập dữ liệu video được mã hóa được lưu trữ trên máy chủ tệp tin 114. Máy chủ tệp tin 114 và giao diện đầu vào 122 có thể được tạo cấu hình để hoạt động theo giao thức truyền trực tuyến, giao thức truyền tải xuống hoặc kết hợp của chúng.

Giao diện đầu ra 108 và giao diện đầu vào 122 có thể đại diện cho bộ phát/bộ thu không dây, modem, các thành phần mạng có dây (ví dụ, thẻ Ethernet), các thành phần giao tiếp không dây hoạt động theo bất kỳ trong số các tiêu chuẩn IEEE 802.11, hoặc các thành phần vật lý khác. Trong các ví dụ trong đó giao diện đầu ra 108 và giao diện đầu vào 122 bao gồm các thành phần không dây, giao diện đầu ra 108 và giao diện đầu vào 122 có thể được tạo cấu hình để truyền dữ liệu, chẳng hạn như dữ liệu video được mã hóa, theo chuẩn truyền thông di động, như 4G, 4G-Dự án phát triển dài hạn (Long -Term Evolution - LTE), LTE cải tiến, 5G, hoặc tương tự. Theo một số ví dụ trong đó giao diện đầu ra 108 bao gồm bộ phát không dây, giao diện đầu ra 108 và giao diện đầu vào 122 có thể được tạo cấu hình để truyền dữ liệu, như dữ liệu video được mã hóa, theo các tiêu chuẩn không dây khác, chẳng hạn như đặc tả kỹ thuật IEEE 802.11, đặc tả kỹ thuật IEEE 802.15 (ví dụ, ZigBee™), tiêu chuẩn Bluetooth™, hoặc tương tự. Theo một số ví dụ, thiết bị nguồn 102 và/hoặc thiết bị đích 116 có thể bao gồm các thiết bị có hệ thống trên chip (system-on-a-chip - SoC) tương ứng. Ví dụ, thiết bị nguồn 102 có thể bao gồm thiết bị SoC để thực hiện chức năng được quy cho bộ mã hóa dữ liệu video 200 và/hoặc giao diện đầu ra 108 và thiết bị đích 116 có thể bao gồm thiết bị SoC để thực hiện chức năng được quy cho bộ giải mã dữ liệu video 300 và/hoặc giao diện đầu vào 122.

Các kỹ thuật theo sáng chế có thể được áp dụng để lập mã dữ liệu video để hỗ trợ ứng dụng đa phương tiện bất kỳ trong số nhiều ứng dụng đa phương tiện, như phát sóng truyền hình qua không khí, truyền hình cáp, truyền hình vệ tinh, truyền dòng dữ liệu video qua mạng internet, như truyền dòng thích ứng động qua giao thức HTTP (DASH - Dynamic Adaptive Streaming over HTTP), dữ liệu video dạng số được mã hóa trên phương tiện lưu trữ dữ liệu, giải mã dữ liệu video dạng số lưu trữ trên phương tiện lưu trữ dữ liệu, hoặc các ứng dụng khác.

Giao diện đầu vào 122 của thiết bị đích 116 nhận luồng bit video được mã hóa từ phương tiện đọc được bằng máy tính 110 (ví dụ, phương tiện truyền thông, thiết bị lưu trữ 112, máy chủ tệp tin 114 hoặc tương tự). Dòng bit video được mã hóa từ phương tiện đọc được bằng máy tính 110 có thể chứa thông tin báo hiệu được xác định bởi bộ mã hóa dữ liệu video 200, thông tin này cũng được bộ giải mã dữ liệu video 300 sử dụng, chẳng hạn như các phần tử cú pháp có các trị số mô tả các đặc điểm và/hoặc xử lý các khối dữ liệu video hoặc các đơn vị được lập mã khác (ví dụ, lát, hình, nhóm hình, chuỗi hoặc tương tự).

Thiết bị hiển thị 118 hiển thị hình ảnh được giải mã của dữ liệu video được giải mã cho người dùng. Thiết bị hiển thị 118 có thể là bất kỳ trong số nhiều thiết bị hiển thị như màn hình ống tia catốt (CRT - Cathode Ray Tube), màn hình tinh thể lỏng (LCD - Liquid Crystal Display), màn hình plasma, màn hình điốt phát sáng hữu cơ (OLED - Organic Light Emitting Diode), hoặc một loại thiết bị hiển thị khác.

Mặc dù không được thể hiện trên Fig.1, nhưng theo một số ví dụ, mỗi bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể được tích hợp với bộ mã hóa âm thanh và/hoặc 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/hoặc phần mềm khác, để xử lý các luồng được ghép kênh bao gồm cả âm thanh và video trong luồng dữ liệu chung. Nếu có thể, các bộ MUX-DEMUX có thể thích hợp với giao thức ghép kênh ITU H.223, hoặc các giao thức khác như giao thức gói dữ liệu người dùng (user datagram protocol - UDP).

Mỗi bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể được thực hiện dưới dạng bất kỳ trong số rất nhiều mạch mã hóa hoặc giải mã 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 Circuit - ASIC), mảng cửa lập trình được bằng trường (Field Programmable Gate Array - FPGA), mạch logic rời rạc, phần mềm, phần cứng, firmware hoặc kết hợp bất kỳ của các loại trên. Khi các kỹ thuật này được thực hiện một phần trong phần mềm, thiết bị có thể lưu trữ các lệnh cho phần mềm trên phương tiện bất biến đọc được bằng máy tính thích hợp và thực thi các lệnh trong phần cứng bằng cách sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế này. Mỗi bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể được bao gồm trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, mỗi trong số chúng có thể được tích hợp như là một phần của bộ mã hóa/bộ giải mã (bộ mã hóa/giải mã - CODEC) kết hợp trong thiết bị tương ứng. Thiết bị bao gồm bộ mã hóa dữ liệu video 200 và/hoặc bộ giải mã dữ liệu video 300 có thể bao gồm mạch tích hợp, bộ vi xử lý và/hoặc thiết bị truyền thông không dây, như điện thoại di động.

Bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể hoạt động theo chuẩn lập mã dữ liệu video, chẳng hạn như ITU-T H.265, còn được gọi là mã hóa dữ liệu video hiệu quả cao (High Efficiency Video Coding - HEVC) hoặc các phần mở rộng của nó, chẳng hạn như các phần mở rộng lập mã dữ liệu video đa cảnh nhìn và/hoặc có thể

thay đổi tỷ lệ. Theo các ví dụ khác, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 cũng có thể hoạt động theo các chuẩn độc quyền hoặc công nghiệp khác, như mô hình thử nghiệm khảo sát chung (Joint Exploration Test Model - JEM) hoặc ITU-T H.266, còn được gọi là mã hóa dữ liệu video đa năng (Versatile Video Coding - VVC). Dự thảo gần đây của chuẩn VVC được mô tả trong Bross, và các cộng sự. “Versatile Video Coding (Draft 6),” nhóm chuyên gia hợp tác chung về kỹ thuật video (Joint Video Experts Team - JVET) của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 15: Gothenburg, SE, từ ngày 3 đến ngày 12 tháng 7 2019, JVET-O2001-vE (sau đây gọi là “VVC Draft 6”). Tuy nhiên, các kỹ thuật của sáng chế không bị giới hạn ở chuẩn lập mã cụ thể bất kỳ.

Nói chung, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể thực hiện mã hóa các hình ảnh theo khối. Thuật ngữ “khối” thường đề cập đến cấu trúc bao gồm dữ liệu cần xử lý (ví dụ, mã hóa, giải mã, hoặc được sử dụng trong quy trình mã hóa và/hoặc giải mã). Ví dụ, khối có thể bao gồm ma trận mẫu hai chiều của dữ liệu độ chói và/hoặc sắc độ. Nói chung, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể mã hóa dữ liệu video được biểu diễn theo định dạng YUV (ví dụ, Y, Cb, Cr). Tức là, thay vì mã hóa dữ liệu màu đỏ, xanh lục và xanh lam (red, green, and blue - RGB) cho các mẫu của hình ảnh, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể mã hóa các thành phần độ chói và sắc độ, trong đó các thành phần sắc độ có thể bao gồm cả thành phần màu sắc đỏ và màu sắc xanh lam. Theo một số ví dụ, bộ mã hóa dữ liệu video 200 chuyển đổi dữ liệu định dạng RGB nhận được thành dạng biểu diễn YUV trước khi mã hóa, và bộ giải mã dữ liệu video 300 chuyển đổi dạng biểu diễn YUV thành định dạng RGB. Theo cách khác, các đơn vị xử lý trước và sau (không được thể hiện trên hình vẽ) có thể thực hiện các quy trình chuyển đổi này.

Nói chung sáng chế này có thể đề cập đến việc lập mã (ví dụ, mã hóa và giải mã) các hình ảnh để bao gồm quy trình mã hóa hoặc giải mã dữ liệu của hình ảnh. Tương tự, sáng chế này có thể đề cập đến việc lập mã các khối của hình ảnh để bao gồm quy trình mã hóa hoặc giải mã dữ liệu cho các khối, ví dụ, lập mã dự đoán và/hoặc mã hóa dư. Luồng bit video được mã hóa thường bao gồm một loạt trị số cho các phần tử cú pháp đại diện cho các quyết định lập mã (ví dụ, chế độ lập mã) và chia các hình ảnh thành các khối. Do

đó, việc tham chiếu đến lập mã hình ảnh hoặc khối thường được hiểu là lập mã các trị số cho các phần tử cú pháp tạo thành hình ảnh hoặc khối.

HEVC định nghĩa một số khối khác nhau, bao gồm các đơn vị lập mã (coding unit - CU), đơn vị dự đoán (prediction unit - PU) và đơn vị biến đổi (transform unit - TU). Theo HEVC, bộ mã hóa dữ liệu video (chẳng hạn như bộ mã hóa dữ liệu video 200) chia đơn vị cây mã hóa (coding tree unit - CTU) thành các CU theo cấu trúc cây tứ phân. Tức là, bộ lập mã dữ liệu video chia các CTU và CU thành bốn hình vuông không chồng lên nhau bằng nhau, và mỗi nút của cây tứ phân có không hoặc bốn nút con. Các nút không có nút con có thể được gọi là “nút lá” và các CU của các nút lá như vậy có thể bao gồm một hoặc nhiều PU và/hoặc một hoặc nhiều TU. Bộ lập mã dữ liệu video có thể chia tiếp các PU và các TU. Ví dụ, trong HEVC, các cây tứ phân dư (residual quadtree - RQT) biểu diễn việc phân chia các TU. Trong HEVC, các PU biểu diễn dữ liệu dự đoán nội ảnh hoặc liên ảnh, còn các TU biểu diễn dữ liệu dư. Các CU mà được dự đoán nội ảnh chứa thông tin dự đoán nội ảnh, như chỉ báo chế độ nội ảnh.

Bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể hoạt động theo chuẩn lập mã dữ liệu video, chẳng hạn như ITU-T H.265, còn được gọi là mã hóa dữ liệu video hiệu quả cao (High Efficiency mã hóa dữ liệu video - HEVC) hoặc các phần mở rộng của nó, chẳng hạn như các phần mở rộng mã hóa dữ liệu video đa cảnh nhìn và/hoặc có thể thay đổi tỷ lệ. Theo cách khác, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 cũng có thể hoạt động theo các chuẩn độc quyền hoặc công nghiệp khác, như mô hình thử nghiệm khảo sát chung (Joint Exploration Test Model - JEM) hoặc ITU-T H.266, còn được gọi là mã hóa dữ liệu video đa năng (Versatile Video Coding - VVC). Dự thảo gần đây của chuẩn VVC được mô tả trong Bross, và các cộng sự. “Versatile Video Coding (Draft 6),” nhóm chuyên gia hợp tác chung về kỹ thuật video (Joint Video Experts Team - JVET) của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 15: Gothenburg, SE, từ ngày 3 đến ngày 12 tháng 7 2019, JVET-O2001-vE (sau đây gọi là “VVC Draft 6”). Tuy nhiên, các kỹ thuật của sáng chế không bị giới hạn ở chuẩn lập mã cụ thể bất kỳ.

Nói chung, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể thực hiện lập mã theo khối các hình ảnh. Thuật ngữ “khối” thường đề cập đến cấu trúc bao gồm dữ liệu cần xử lý (ví dụ, mã hóa, giải mã, hoặc được sử dụng trong quy trình mã hóa

và/hoặc giải mã). Ví dụ, khối có thể bao gồm ma trận mẫu hai chiều của dữ liệu độ chói và/hoặc sắc độ. Nói chung, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể mã hóa dữ liệu video được biểu diễn theo định dạng YUV (ví dụ, Y, Cb, Cr). Tức là, thay vì mã hóa dữ liệu màu đỏ, xanh lục và xanh lam (red, green, and blue - RGB) cho các mẫu của hình ảnh, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể mã hóa các thành phần độ chói và sắc độ, trong đó các thành phần sắc độ có thể bao gồm cả thành phần sắc độ đỏ và xanh lam. Theo một số ví dụ, bộ mã hóa dữ liệu video 200 biến đổi dữ liệu định dạng RGB nhận được thành dạng biểu diễn YUV trước khi mã hóa, và bộ giải mã dữ liệu video 300 biến đổi dạng biểu diễn YUV thành định dạng RGB. Theo cách khác, các đơn vị xử lý trước và sau (không được thể hiện trên hình vẽ) có thể thực hiện các quy trình chuyển đổi này.

Nói chung sáng chế này có thể đề cập đến việc mã hóa (ví dụ, mã hóa và giải mã) các hình ảnh để bao gồm quy trình mã hóa hoặc giải mã dữ liệu của hình ảnh. Tương tự, sáng chế này có thể đề cập đến việc lập mã các khối của hình ảnh để bao gồm quy trình mã hóa hoặc giải mã dữ liệu cho các khối, ví dụ, lập mã dự đoán và/hoặc mã hóa dư. Luồng bit video mã hóa thường bao gồm một loạt trị số cho các phần tử cú pháp đại diện cho các quyết định lập mã (ví dụ, chế độ lập mã) và chia các hình ảnh thành các khối. Do đó, các tham chiếu đến lập mã hình ảnh hoặc khối thường được hiểu là lập mã các trị số cho các phần tử cú pháp tạo thành hình ảnh hoặc khối.

HEVC định nghĩa các khối khác nhau, bao gồm các đơn vị lập mã (coding unit - CU), đơn vị dự đoán (prediction unit - PU) và đơn vị biến đổi (transform unit - TU). Theo HEVC, bộ lập mã dữ liệu video (chẳng hạn như bộ mã hóa dữ liệu video 200) chia đơn vị cây lập mã (coding tree unit - CTU) thành các CU theo cấu trúc cây tứ phân. Tức là, bộ lập mã dữ liệu video chia các CTU và CU thành bốn hình vuông không chồng lên nhau bằng nhau, và mỗi nút của cây tứ phân có không hoặc bốn nút con. Các nút không có nút con có thể được gọi là “nút lá” và các CU của các nút lá như vậy có thể bao gồm một hoặc nhiều PU và/hoặc một hoặc nhiều TU. Bộ lập mã dữ liệu video có thể chia tiếp các PU và các TU. Ví dụ, trong HEVC, các cây tứ phân dư (residual quadtree - RQT) biểu diễn việc phân chia các TU. Trong HEVC, các PU biểu diễn dữ liệu dự đoán liên ảnh, còn các TU biểu diễn dữ liệu dư. Các CU được dự đoán nội ảnh chứa thông tin dự đoán nội ảnh, như chỉ báo chế độ nội ảnh.

Theo một ví dụ khác, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để hoạt động theo mô hình JEM hoặc VVC. Theo JEM hoặc VVC, bộ lập mã dữ liệu video (như bộ mã hóa dữ liệu video 200) phân chia hình ảnh thành nhiều đơn vị cây lập mã (CTU). Bộ mã hóa dữ liệu video 200 có thể chia CTU theo cấu trúc cây, chẳng hạn như cấu trúc cây nhị phân-cây tứ phân (QTBT - quadtree-binary tree) hoặc cấu trúc cây nhiều dạng (Multi-Type Tree-MTT). Cấu trúc QTBT loại bỏ các khái niệm về nhiều loại phân chia, như phân chia giữa các CU, PU và TU theo tiêu chuẩn HEVC. Cấu trúc QTBT bao gồm hai mức: mức thứ nhất được chia theo kỹ thuật phân chia cây tứ phân và mức thứ hai được phân chia theo kỹ thuật phân chia cây nhị phân. Nút gốc của cấu trúc QTBT tương ứng với CTU. Các nút lá của cây nhị phân tương ứng với các đơn vị lập mã (CU - coding unit).

Trong cấu trúc phân chia cây MTT, các khối có thể được phân chia bằng cách sử dụng phân chia cây tứ phân (quadtree - QT), phân chia cây nhị phân (binary tree - BT), và một hoặc nhiều loại phân chia cây tam phân (triple tree - TT) (còn được gọi là cây tam nguyên (ternary tree - TT)). Phân chia cây tam phân hoặc tam nguyên là phân chia trong đó khối được phân chia thành ba khối con. Trong một số ví dụ, phân chia cây tam phân hoặc tam nguyên chia khối thành ba khối con mà không chia khối gốc qua tâm. Các loại phân chia trong cây MTT (ví dụ, QT, BT, và TT), có thể là đối xứng hoặc bất đối xứng.

Trong một số ví dụ, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể sử dụng một cấu trúc QTBT hoặc MTT để biểu diễn mỗi thành phần độ chói và sắc độ, trong khi trong các ví dụ khác, bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể sử dụng hai hoặc nhiều cấu trúc QTBT hoặc MTT, như cấu trúc QTBT/MTT cho thành phần độ chói và cấu trúc QTBT/MTT khác cho cả hai thành phần sắc độ (hoặc hai cấu trúc QTBT/MTT cho các thành phần sắc độ tương ứng).

Bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để sử dụng phân chia cây tứ phân cho mỗi phân chia HEVC, QTBT, phân chia MTT, hoặc các cấu trúc phân chia khác. Đối với mục đích giải thích, phần mô tả về các kỹ thuật của sáng chế được trình bày liên quan đến kỹ thuật phân chia cây QTBT. Tuy nhiên, cần phải hiểu rằng các kỹ thuật của sáng chế này cũng có thể được áp dụng cho các bộ lập mã dữ liệu video được tạo cấu hình để sử dụng kỹ thuật phân chia cây tứ phân hoặc các kiểu kỹ thuật phân chia khác.

Các khối (ví dụ, CTU hoặc CU) có thể được nhóm theo nhiều cách vào một hình. Theo một ví dụ, mảnh có thể là vùng hình chữ nhật trong các hàng CTU bên trong ô cụ thể trong hình. Ô có thể là vùng hình chữ nhật của gồm các CTU trong một cột ô cụ thể và hàng ô cụ thể trong một hình ảnh. Cột ô là vùng hình chữ nhật gồm các CTU có chiều cao bằng chiều cao của hình ảnh và chiều rộng được xác định bởi các phần tử cú pháp (ví dụ, như trong tập hợp thông số hình ảnh). Hàng ô là vùng hình chữ nhật gồm các CTU có chiều cao được xác định bởi các phần tử cú pháp (ví dụ, như trong tập hợp thông số hình ảnh) và chiều rộng bằng chiều rộng của hình ảnh.

Theo một số ví dụ, ô có thể được chia thành nhiều mảnh, mỗi mảnh có thể bao gồm một hoặc nhiều hàng CTU trong ô. Ô không được phân chia thành nhiều mảnh có thể cùng được gọi là mảnh. Tuy nhiên, mảnh là một tập hợp con thực sự của ô có thể không được gọi là ô.

Các mảnh trong hình ảnh có thể còn được sắp xếp trong lát. Lát có thể là số nguyên của mảnh của các hình ảnh mà có thể chưa riêng trong một đơn vị lớp trừu tượng hóa mạng (NAL). Theo một số ví dụ, lát bao gồm một số ô hoàn chỉnh hoặc chỉ một chuỗi liên tiếp các mảnh hoàn chỉnh gồm một ô.

Sáng chế này có thể sử dụng “NxN” và “N với N” thay thế cho nhau để chỉ các kích thước mẫu của khối (chẳng hạn như khối CU hoặc khối dữ liệu video khác) về các chiều dọc và ngang, ví dụ, 16x16 mẫu hoặc 16 với 16 mẫu. Nói chung, CU16x16 sẽ có 16 mẫu theo chiều dọc ($y = 16$) và 16 mẫu theo chiều ngang ($x = 16$). Tương tự, CU NxN thường có N mẫu theo chiều dọc và N mẫu theo chiều ngang, trong đó N là trị số nguyên không âm. Các mẫu trong CU có thể được sắp xếp thành các hàng và các cột. Hơn nữa, các CU không nhất thiết phải có số lượng mẫu theo chiều ngang bằng số lượng điểm mẫu theo chiều dọc. Ví dụ, các CU có thể bao gồm NxM mẫu, trong đó M không nhất thiết bằng N.

Bộ mã hóa dữ liệu video 200 mã hóa dữ liệu video cho các CU biểu diễn thông tin dự đoán và/hoặc thông tin dư, và thông tin khác. Thông tin dự đoán biểu thị cách thức CU sẽ được dự đoán để tạo ra khối dự đoán cho CU. Thông tin dư thường biểu diễn các trị số chênh lệch từng mẫu giữa các mẫu của CU trước khi mã hóa và khối dự đoán.

Để dự đoán đơn vị CU, bộ mã hóa dữ liệu video 200 có thể thông thường tạo ra khối dự đoán cho CU thông qua dự đoán liên ảnh hoặc dự đoán nội ảnh. Dự báo liên ảnh

thường được dùng để chỉ việc dự đoán CU từ dữ liệu của hình ảnh được lập mã trước đó, trong khi dự đoán nội ảnh thường được dùng để chỉ việc dự đoán CU từ dữ liệu được lập mã trước đó của cùng một hình ảnh. Để thực hiện dự đoán liên ảnh, bộ mã hóa dữ liệu video 200 có thể tạo ra khối dự đoán bằng cách sử dụng một hoặc nhiều vector chuyển động. Bộ mã hóa dữ liệu video 200 có thể thường thực hiện tìm kiếm chuyển động để xác định khối tham chiếu phù hợp nhất với CU, ví dụ, theo các khác biệt giữa CU và khối tham chiếu. Bộ mã hóa dữ liệu video 200 có thể tính toán trị số đo chênh lệch bằng cách sử dụng tổng chênh lệch tuyệt đối (sum of absolute difference - SAD), tổng chênh lệch bình phương (sum of squared differences - SSD), chênh lệch tuyệt đối trung bình (mean absolute difference - MAD), chênh lệch bình phương trung bình (mean squared differences - MSD) hoặc các phép toán chênh lệch như vậy khác để xác định xem khối tham chiếu có phù hợp nhất với CU hiện thời không. Theo một số ví dụ, bộ mã hóa dữ liệu video 200 có thể dự đoán CU hiện thời bằng cách sử dụng kỹ thuật dự đoán một chiều hoặc dự đoán hai chiều.

Một số ví dụ của JEM và VVC còn cung cấp chế độ bù chuyển động afin, chế độ này có thể được coi là chế độ dự đoán liên ảnh. Trong chế độ bù chuyển động afin, bộ mã hóa dữ liệu video 200 có thể xác định hai hoặc nhiều vector chuyển động đại diện cho chuyển động không tịnh tiến, chẳng hạn như phóng to hoặc thu nhỏ, xoay, chuyển động phối cảnh hoặc các loại chuyển động không đều khác.

Để thực hiện dự đoán nội ảnh, bộ mã hóa dữ liệu video 200 có thể lựa chọn chế độ dự đoán nội ảnh để tạo ra khối dự đoán. Một số ví dụ của JEM và VVC cung cấp sáu mươi bảy chế độ dự đoán nội ảnh, bao gồm các chế độ định hướng khác nhau, cũng như chế độ phẳng và chế độ DC. Nói chung, bộ mã hóa dữ liệu video 200 chọn chế độ dự đoán nội ảnh để mô tả các mẫu lân cận cho khối hiện thời (ví dụ, khối của CU) mà dự đoán các mẫu của khối hiện thời từ đó. Các mẫu như vậy có thể thường ở trên, bên trên và bên trái hoặc bên trái của khối hiện thời trong hình giống như khối hiện thời, giả sử bộ mã hóa dữ liệu video 200 mã hóa các CTU và các CU theo thứ tự quét mảnh (từ trái sang phải, từ trên xuống dưới).

Bộ mã hóa dữ liệu video 200 mã hóa dữ liệu đại diện cho chế độ dự đoán cho khối hiện thời. Ví dụ, đối với các chế độ dự đoán liên ảnh, bộ mã hóa dữ liệu video 200 có thể mã hóa dữ liệu đại diện cho các chế độ dự đoán liên ảnh có sẵn được dùng, cũng như thông tin chuyển động cho chế độ tương ứng. Ví dụ, đối với dự đoán liên ảnh một chiều hoặc hai

chiều, bộ mã hóa dữ liệu video 200 có thể mã hóa các vector chuyển động bằng cách sử dụng chế độ dự đoán vector chuyển động nâng cao (advanced motion vector prediction - AMVP) hoặc chế độ hợp nhất. Bộ mã hóa dữ liệu video 200 có thể sử dụng các chế độ tương tự để mã hóa các vector chuyển động cho chế độ bù chuyển động afin.

Sau dự đoán, chẳng hạn như dự đoán nội ảnh hoặc dự đoán liên ảnh các khối, bộ mã hóa dữ liệu video 200 có thể tính toán dữ liệu dư cho khối. Dữ liệu dư, chẳng hạn như khối dư, biểu diễn các trị số chênh lệch mẫu-mẫu giữa khối và khối dự đoán cho khối, được tạo ra bằng cách sử dụng chế độ dự đoán tương ứng. Bộ mã hóa dữ liệu video 200 có thể áp dụng một hoặc nhiều phép biến đổi cho khối dư, để tạo ra dữ liệu được biến đổi trong miền biến đổi thay vì miền mẫu. Ví dụ, bộ mã hóa dữ liệu video 200 có thể áp dụng phép biến đổi cosin rời rạc (discrete cosine transform - DCT), biến đổi số nguyên, biến đổi sóng nhỏ, hoặc biến đổi tương tự về mặt khái niệm cho dữ liệu video dư. Ngoài ra, bộ mã hóa dữ liệu video 200 có thể áp dụng phép biến đổi thứ hai tiếp theo phép biến đổi thứ nhất, chẳng hạn như biến đổi thứ hai không phân chia phụ thuộc vào chế độ (mode-dependent non-separable secondary transform - MDNSST), biến đổi phụ thuộc vào tín hiệu, biến đổi Karhunen-Loeve (Karhunen-Loeve transform - KLT) hoặc tương tự. Bộ mã hóa dữ liệu video 200 tạo ra các hệ số biến đổi sau khi áp dụng một hoặc nhiều phép biến đổi.

Như nêu ở trên, sau phép biến đổi bất kỳ để tạo ra các hệ số biến đổi, bộ mã hóa dữ liệu video 200 có thể thực hiện lượng tử hóa các hệ số biến đổi. 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. Bằng cách thực hiện quy trình lượng tử hóa, bộ mã hóa dữ liệu video 200 có thể giảm độ sâu bit liên quan đến một số hoặc tất cả các hệ số. Ví dụ, bộ mã hóa dữ liệu video 200 có thể làm tròn trị số n -bit xuống trị số m -bit khi lượng tử hóa, trong đó n lớn hơn m . Theo một số ví dụ, để thực hiện lượng tử hóa, bộ mã hóa dữ liệu video 200 có thể thực hiện dịch phải phân theo bit trị số cần lượng tử hóa.

Sau khi lượng tử hóa, bộ mã hóa dữ liệu video 200 có thể quét các hệ số biến đổi, tạo ra vector một chiều từ ma trận hai chiều bao gồm các hệ số biến đổi đã 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 vector và để đặt các hệ số biến đổi năng lượng thấp hơn (và do đó tần

số cao hơn) ở phía sau của vector. Theo một số ví dụ, bộ mã hóa dữ liệu video 200 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, và sau đó mã hóa entropy các hệ số biến đổi lượng tử hóa của vector. Theo các ví dụ khác, bộ mã hóa dữ liệu video 200 có thể thực hiện quy trình quét thích ứng. Sau khi quét các hệ số biến đổi được lượng tử hóa để tạo vector một chiều, bộ mã hóa dữ liệu video 200 có thể mã hóa entropy vector một chiều, ví dụ, theo kỹ thuật mã hóa số học nhị phân thích ứng theo ngữ cảnh (context-adaptive binary arithmetic coding - CABAC). Bộ mã hóa dữ liệu video 200 cũng có thể mã hóa entropy các trị số cho các phần tử cú pháp mô tả siêu dữ liệu liên quan đến dữ liệu video được mã hóa dùng bởi bộ giải mã dữ liệu video 300 khi giải mã dữ liệu video.

Để thực hiện CABAC, bộ mã hóa dữ liệu video 200 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 xem các trị số lân cận của ký hiệu có trị số bằng không hay không. Việc xác định xác suất có thể được dựa vào ngữ cảnh được gán cho ký hiệu.

Bộ mã hóa dữ liệu video 200 còn có thể tạo ra 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, cho bộ giải mã dữ liệu video 300, ví dụ, trong phần đầu hình ảnh, phần đầu khối, phần đầu lát, hoặc dữ liệu cú pháp khác, như tập thông số chuỗi (sequence parameter set - SPS), tập thông số hình ảnh (picture parameter set - PPS), hoặc tập thông số video (video parameter set - VPS). Tương tự, bộ giải mã dữ liệu video 300 có thể giải mã dữ liệu cú pháp như vậy để xác định cách giải mã dữ liệu video tương ứng.

Theo cách này, bộ mã hóa dữ liệu video 200 có thể tạo ra luồng bit bao gồm dữ liệu video được mã hóa, ví dụ, các phần tử cú pháp mô tả việc phân chia hình ảnh thành các khối (ví dụ, các CU) và thông tin dự đoán và/hoặc thông tin dư cho các khối. Cuối cùng, bộ giải mã dữ liệu video 300 có thể nhận luồng bit và giải mã dữ liệu video được mã hóa.

Nói chung, bộ giải mã dữ liệu video 300 thực hiện quy trình thuận nghịch với quy trình được thực hiện bởi bộ mã hóa dữ liệu video 200 để giải mã dữ liệu video đã mã hóa của luồng bit. Ví dụ, bộ giải mã dữ liệu video 300 có thể giải mã các trị số cho các phần tử cú pháp của luồng bit bằng cách sử dụng kỹ thuật CABAC theo cách về cơ bản tương tự như, mặc dù thuận nghịch với, quy trình mã hóa CABAC của bộ mã hóa dữ liệu video 200. Các phần tử cú pháp có thể xác định thông tin chia hình ảnh thành các CTU, và chia

mỗi CTU theo cấu trúc chia tương ứng, chẳng hạn như cấu trúc QTBT, để xác định các CU của CTU. Các phần tử cú pháp còn có thể xác định thông tin dự đoán và thông tin dư cho các khối (ví dụ, các CU) của dữ liệu video.

Thông tin dư có thể được biểu diễn bằng, ví dụ, các hệ số biến đổi lượng tử hóa. Bộ giải mã dữ liệu video 300 có thể lượng tử hóa ngược và biến đổi ngược các hệ số biến đổi lượng tử hóa của khối để tái tạo khối dư cho khối. Bộ giải mã dữ liệu video 300 sử dụng chế độ dự đoán được báo hiệu (dự đoán nội ảnh hoặc dự đoán liên ảnh) và thông tin dự đoán liên quan (ví dụ, thông tin chuyển động của dự đoán liên ảnh) để tạo ra khối dự đoán cho khối. Sau đó, bộ giải mã dữ liệu video 300 có thể kết hợp khối dự đoán và khối dư (dựa trên từng mẫu) để tái tạo khối gốc. Bộ giải mã dữ liệu video 300 có thể thực hiện xử lý bổ sung, chẳng hạn như thực hiện quy trình tách khối để giảm các thành phần xảo ảnh trực quan dọc theo các biên của khối.

Nói chung, sáng chế có thể liên quan đến việc "báo hiệu" một số thông tin như các phần tử cú pháp. Nói chung, thuật ngữ "báo hiệu" đề cập đến truyền thông trị số cho các phần tử cú pháp và/hoặc dữ liệu khác được dùng để giải mã dữ liệu video đã mã hóa. Tức là, bộ mã hóa dữ liệu video 200 có thể báo hiệu các trị số cho các phần tử cú pháp trong luồng bit. Nói chung, việc báo hiệu đề cập đến việc tạo ra trị số trong luồng bit. Như đã lưu ý ở trên, thiết bị nguồn 102 có thể truyền luồng bit đến thiết bị đích 116 về cơ bản theo thời gian thực, hoặc không theo thời gian thực, việc này có thể xảy ra khi lưu trữ các phần tử cú pháp vào thiết bị lưu trữ 112 để thiết bị đích 116 truy xuất sau.

Các hình vẽ Fig.2A và Fig.2B là sơ đồ khái niệm minh họa một ví dụ về cấu trúc cây nhị phân tứ phân (QTBT) 130, và đơn vị cây lập mã (CTU) 132 tương ứng. Các đường nét liền biểu thị việc tách cây tứ phân và các đường nét đứt biểu thị việc tách cây nhị phân. Trong mỗi nút tách (tức là, nút không phải lá) của cây nhị phân, một cờ được báo hiệu để chỉ báo kiểu tách nào (tức là, tách ngang hay tách dọc) được sử dụng, trong đó 1 biểu thị tách ngang và 0 biểu thị tách dọc theo ví dụ này. Đối quy trình tách cây tứ phân, không cần thiết biểu thị kiểu tách, vì các nút cây tứ phân chia khối theo chiều ngang và theo chiều dọc thành 4 khối con có kích thước bằng nhau. Do đó, bộ mã hóa dữ liệu video 200 có thể mã hóa, và bộ giải mã dữ liệu video 300 có thể giải mã, các phần tử cú pháp (ví dụ như thông tin tách) cho mức cây khu vực của cấu trúc QTBT 130 (tức là, các đường nét liền) và các phần tử cú pháp (ví dụ như thông tin tách) cho mức cây dự đoán của cấu trúc QTBT

130 (tức là, các đường nét đứt). Bộ mã hóa dữ liệu video 200 có thể mã hóa, và bộ giải mã dữ liệu video 300 có thể giải mã, dữ liệu video, như dữ liệu dự đoán và biến đổi, cho các CU được biểu diễn bởi các nút lá đầu cuối của cấu trúc QTBT 130.

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

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

Trong một ví dụ của cấu trúc phân chia QTBT, kích thước CTU được thiết lập là 128x128 (mẫu độ chói và hai mẫu sắc độ 64x64 tương ứng), MinQTSIZE được thiết lập là 16x16, MaxBTSIZE được thiết lập là 64x64, MinBTSize (cho cả chiều rộng lẫn chiều cao) được thiết lập là 4 và MaxBTDepth được thiết lập là 4. Phân chia cây tứ phân được áp dụng cho CTU trước tiên để tạo ra các nút lá cây tứ phân. Các nút lá cây tứ phân có thể có

kích thước từ 16×16 (tức là MinQTSize) đến 128×128 (tức là, kích thước CTU). Nếu nút lá cây tứ phân là 128×128 , nó sẽ không được phân chia tiếp bởi cây nhị phân bởi vì kích thước này vượt quá MaxBTSIZE (tức là 64×64 theo ví dụ này). Cách khác, nút lá cây tứ phân sẽ được phân chia tiếp bởi cây nhị phân. Do đó, nút lá cây tứ phân cũng là nút gốc cho cây nhị phân và có độ sâu cây nhị phân bằng 0. Khi độ sâu cây nhị phân đạt đến MaxBTDDepth (bằng 4 trong một ví dụ), thì không được phép tách tiếp nữa. Khi nút cây nhị phân có độ rộng bằng MinBTSIZE (bằng 4 theo ví dụ này), việc này ngụ ý rằng không được phép tách theo chiều dọc nữa. Tương tự, nút cây nhị phân có chiều cao bằng MinBTSIZE ngụ ý rằng không được phép tách theo chiều ngang nữa cho nút cây nhị phân đó. Như mô tả ở trên, các nút lá của cây nhị phân được gọi là các CU, và được xử lý thêm theo kỹ thuật dự đoán và biến đổi mà không cần phân chia thêm.

Các xảo ảnh dạng khối có thể bao gồm các điểm gián đoạn theo chiều ngang và dọc trong hình ảnh tái tạo mà không tồn tại trong hình ảnh tĩnh hoặc hình ảnh video gốc và thường là kết quả của độ nén từ trung bình đến cao. Ví dụ, nếu bộ mã hóa dữ liệu video nén nhiều ảnh đầu vào, thì chất lượng thấy được của ảnh giải mã có thể bị ảnh hưởng khi nội dung hình ảnh được giải nén, tức là, được giải mã, nội dung ảnh được giải nén bị ảnh hưởng bởi các xảo ảnh dạng khối. Các xảo ảnh này trong các vùng phẳng trông giống như “tạo ô” do các xảo ảnh không bị che bởi nội dung có độ tương phản cao. Hơn nữa, các xảo ảnh dạng khối trong video đang phát có thể được quan sát là “chuyển động và lập lờ” bởi vì các điểm gián đoạn nằm ở các vị trí khác nhau trong các khung liên tục.

Như được giới thiệu ở trên, nguồn xảo ảnh dạng khối là lập mã biến đổi theo khối, bao gồm biến đổi và lượng tử hóa, tức là được thực hiện trên các lỗi dự đoán nội ảnh và liên ảnh (ví dụ, các khối dư). Quy trình lượng tử hóa thô các hệ số biến đổi có thể gây ra các điểm gián đoạn gây nhiễu trực quan tại các biên của khối. Dự đoán bù chuyển động là một nguồn có thể có khác của các xảo ảnh dạng khối. Các khối bù chuyển động được tạo ra bằng cách sao chép dữ liệu điểm ảnh nội suy từ các vị trí khác nhau trong các khung tham chiếu khác nhau có thể có. Vì gần như không bao giờ có sự phù hợp hoàn hảo cho dữ liệu này (ví dụ, các trị số mẫu cho các khối tham chiếu tạo ra từ quy trình nội suy), nên thường phát sinh các điểm gián đoạn trên biên của các khối dữ liệu sao chép. Tức là, hai khối lân cận được mã hóa bằng cách sử dụng kỹ thuật bù chuyển động có thể có các điểm gián đoạn.

Lọc tách khối (như trong HEVC, ví dụ) thường được thực hiện sau khi hình ảnh hoặc nhóm các khối được tái tạo/giải mã để giảm các xảo ảnh dạng khối. Cụ thể hơn, lọc tách khối sửa đổi trị số của các mẫu nằm gần mỗi biên khối, bằng cách lọc, cắt, hoặc theo các cách khác, sao cho điểm gián đoạn được làm mịn và, do đó, ít thấy được hơn.

Trong quy trình nén video, lọc tách khối có thể được thực hiện trên các ảnh trong bộ đệm hiển thị và bên ngoài vòng lặp dự đoán, nghĩa là các hình ảnh lọc tách khối không được sử dụng để dự đoán các hình ảnh khác. Lọc tách khối cũng có thể được thực hiện trong vòng lặp dự đoán, nghĩa là các hình ảnh tách khối được sử dụng làm các hình ảnh tham chiếu để bù chuyển động cho các hình ảnh trong tương lai. Cả hai kỹ thuật này đều có khả năng cải thiện chất lượng chủ quan của video được hiển thị, trong khi kỹ thuật thứ hai cũng có khả năng cải thiện hiệu quả nén do độ chính xác dự đoán liên ảnh được cải thiện bằng cách sử dụng các khung tham chiếu tách khối.

Việc tách khối trong HEVC là quy trình trong vòng lặp và được áp dụng cho các biên khối thỏa mãn hai điều kiện. Điều kiện thứ nhất là các biên là các biên của CU, PU, hoặc TU, và điều kiện thứ hai là tọa độ x (hoặc tọa độ y) là (và theo một số ví dụ, phải là) bội số của 8, nếu biên là dọc (hoặc ngang). Điều kiện thứ hai nghĩa là khoảng cách tối thiểu của hai biên lân cận song song cần tách khối là 8 điểm ảnh, việc này tạo điều kiện cho quá trình song song tốt hơn, như được mô tả chi tiết hơn dưới đây. Bộ lọc tách khối chia biên cần xử lý, bất kể ranh giới dài bao nhiêu, thành nhiều đoạn 4 mẫu không chồng nhau, là các đơn vị mà bộ lọc tách khối thực hiện lọc tách khối trên đó. Theo sáng chế này, các hoạt động tách khối trên đoạn sẽ được giới thiệu, và đoạn có thể được giả định là theo chiều dọc, nhưng việc xử lý các đoạn theo chiều ngang có hiệu quả như nhau.

Bộ giải mã dữ liệu video 300, khi thực hiện lọc tách khối, có thể thực hiện xác định cường độ biên. Đối với đoạn cần xử lý, bộ giải mã dữ liệu video kiểm tra các điều kiện lập mã (ví dụ, vector chuyển động (MV), chỉ số tham chiếu, và sự có mặt của các hệ số biến đổi khác không) của hai khối ở hai bên, ký hiệu là P và Q, ở hai bên của biên khối.

Fig.3 thể hiện ví dụ về khối P 142 và khối Q 144. Khối P 142 là khối 8x8, và khối Q 144 là khối 16x16. Khối P 142 và khối Q 144 dùng chung biên khối 146, được thể hiện trên Fig.3 là các đoạn 146A và 146B. Cả hai đoạn 146A và 146B đều dài bốn mẫu, phù hợp với quy trình lọc tách khối của HEVC, nhưng các kỹ thuật của sáng chế không bị giới hạn ở bất kỳ độ dài đoạn cụ thể nào.

Đối với các khối độ chói, bộ giải mã dữ liệu video xác định cường độ biên, của đoạn 146A. Cường độ biên thể hiện khả năng các xảo ảnh dạng khối mạnh xuất hiện quanh đoạn 146A. Theo một ví dụ, trị số cường độ biên có thể bằng 0, nghĩa là các điều kiện lập mã trong khối P 142 và khối Q 144 là các xảo ảnh dạng khối sao cho chúng không được kỳ vọng dọc theo đoạn 146A và việc lọc tách khối cho đoạn 146A có thể được bỏ qua. Theo một ví dụ khác, trị số cường độ biên có thể bằng 2, nghĩa là các điều kiện lập mã trong khối P 142 và khối Q 144 là các xảo ảnh dạng khối sao cho chúng được kỳ vọng có mặt và có thể mong muốn lọc tách khối mạnh hơn. Trị số cường độ biên cũng có thể bằng 1, nghĩa là các điều kiện lập mã trong khối P 142 và khối Q 144 sao cho có khả năng xuất hiện các xảo ảnh dạng khối nhẹ hơn và một số kỹ thuật lọc tách khối, nhưng không mạnh bằng kỹ thuật lọc tách khối khi cường độ biên bằng 2, có thể là mong muốn. Chi tiết về nguồn gốc cường độ biên được mô tả trong đoạn 8.7.2.4 của HEVC.

Bộ giải mã dữ liệu video có thể xác định cường độ biên cho các khối sắc độ khác với các khối độ chói. Ví dụ, đối với khối sắc độ, bộ giải mã dữ liệu video gán các đoạn liền kề với các khối lập mã nội ảnh, cường độ biên bằng 2, và nếu không gán các đoạn có cường độ biên bằng 0.

Như được mô tả ở trên, bộ giải mã dữ liệu video có thể ước lượng khả năng mà đoạn có các xảo ảnh dạng khối dựa vào các điều kiện lập mã của các khối liền kề. Tuy nhiên, bộ giải mã dữ liệu video cũng có thể thực hiện phân tích sâu hơn, dựa vào trị số của mẫu gần đoạn đó, để xác định xem đoạn có nên được tách khỏi khối hay không và như thế nào. Đầu tiên, bộ giải mã dữ liệu video đưa ra quyết định có lọc đoạn hay không bằng cách tính toán các giá trị đạo hàm bậc hai của bốn mẫu gần đoạn.

Fig.4 thể hiện ví dụ về đoạn 150, nằm trên biên giữa khối p 152 và khối q 154. Bộ giải mã dữ liệu video tính toán các giá trị đạo hàm bậc hai của bốn mẫu gần đoạn, mà được thể hiện trên Fig.4 là các mẫu tuần hoàn tròn từ 158A đến 158D. Nếu tổng của bốn đạo hàm bậc hai (xem biểu thức (1)) nhỏ hơn ngưỡng β , như được mô tả dưới đây liên quan đến các ngưỡng β và t_c , thì bộ giải mã dữ liệu video xác định rằng đoạn cần được tách khỏi. Nếu không, bộ giải mã dữ liệu video xác định rằng đoạn được coi là nằm trong vùng không phẳng, trong đó các xảo ảnh dạng khối là có thể bị che, và vì vậy không cần phải tách khỏi. Đối với đoạn, ngay cả với đoạn có cường độ biên khác không, bộ giải mã dữ liệu video có thể bỏ qua quy trình lọc tách khối nếu ngưỡng của biểu thức (1) không đạt.

$$|p_{2,0} - 2p_{1,0} + p_{0,0}| + |p_{2,3} - 2p_{1,3} + p_{0,3}| + |q_{2,0} - 2q_{1,0} + q_{0,0}| + |q_{2,3} - 2q_{1,3} + q_{0,3}| < \beta \quad (1)$$

Thứ hai, đối với đoạn cần tách khối, bộ giải mã dữ liệu video có thể đưa ra quyết định khác về việc xem có sử dụng chế độ lọc mạnh hay thông thường. Nếu sáu điều kiện sau đây (các biểu thức từ (2-1) đến (2-6)) đều đúng, tức là một vùng có thể quá mịn để che bất kỳ xáo ảnh dạng khối, thì bộ giải mã dữ liệu video xác định rằng chế độ lọc mạnh được sử dụng. Nếu không, bộ giải mã dữ liệu video xác định rằng chế độ lọc thông thường được sử dụng.

$$|p_{2,0} - 2p_{1,0} + p_{0,0}| + |q_{2,0} - 2q_{1,0} + q_{0,0}| < \beta/8 \quad (2-1)$$

$$|p_{2,3} - 2p_{1,3} + p_{0,3}| + |q_{2,3} - 2q_{1,3} + q_{0,3}| < \beta/8 \quad (2-2)$$

$$|p_{3,0} - p_{0,0}| + |q_{0,0} - q_{3,0}| < \beta/8 \quad (2-3)$$

$$|p_{3,3} - p_{0,3}| + |q_{0,3} - q_{3,3}| < \beta/8 \quad (2-4)$$

$$|p_{0,0} - q_{0,0}| < 2.5t_c \quad (2-5)$$

$$|p_{0,3} - q_{0,3}| < 2.5t_c \quad (2-6)$$

Thông số ngưỡng t_c là thông số cắt, được mô tả chi tiết hơn dưới đây.

Fig.5 thể hiện ví dụ về đoạn 160, nằm trên biên giữa khối p 162 và khối q 164. Fig.5 sẽ được sử dụng để thể hiện hoạt động lọc tách khối cho một dòng, được biểu diễn bởi các vị trí của các mẫu vật lý $p_0, p_1, p_2, p_3, q_0, q_1, q_2$, và q_3 .

Để xác định cường độ biên và các quyết định khác cho độ chói được mô tả ở trên, bộ giải mã dữ liệu video thực hiện phân tích và suy ra ở mức phân đoạn. Trong chế độ lọc mạnh đối với độ chói và chế độ lọc thông thường cho độ chói, lọc tách khối được thực hiện từng dòng (ví dụ, từng hàng nếu đoạn là dọc hoặc từng cột nếu đoạn là ngang).

Trong chế độ lọc mạnh cho độ chói, bộ giải mã dữ liệu video xử lý ba mẫu trên mỗi cạnh của đoạn 160. Theo ví dụ trên Fig.5, các trị số mẫu đối với p_0, p_1 , và p_2 trong khối P 162 được cập nhật thành p'_0, p'_1 , và p'_2 , bằng cách lọc thông thấp, như được thể hiện trong các biểu thức Từ (3-1) đến (3-3).

$$p'_0 = (p_2 + 2p_1 + 2p_0 + 2q_0 + q_1 + 4) \gg 3 \quad (3-1)$$

$$p'_1 = (p_2 + p_1 + p_0 + q_0 + 2) \gg 2 \quad (3-2)$$

$$p'_2 = (2p_3 + 3p_2 + p_1 + p_0 + q_0 + 4) \gg 3 \quad (3-3)$$

Các trị số mẫu sửa đổi p'_i ($i = 0, 1, 2$) được cắt trong phạm vi $[p_i - 2t_c, p_i + 2t_c]$.

Bộ giải mã dữ liệu video xử lý q_0, q_1 , và q_2 trong khối Q 164 bằng cách sử dụng hiệu quả các biểu thức tương tự như các biểu thức từ (3-1) đến (3-3), mặc dù đổi với q thay vì p . Quy trình cắt được áp dụng theo cách tương tự cho q như đối với p .

Trong chế độ lọc thông thường cho độ chói, bộ giải mã dữ liệu video xử lý, một hoặc hai mẫu trên mỗi cạnh của đoạn 160. Đối với cạnh trái của đoạn 160, ví dụ, bộ giải mã dữ liệu video kiểm tra điều kiện trong biểu thức (4-1) được chọn. Nếu điều kiện là đúng, thì bộ giải mã dữ liệu video xử lý các mẫu p_0 và p_1 . Nếu không thì, bộ giải mã dữ liệu video chỉ xử lý mẫu p_0 . Tương tự, đối với cạnh phải, bộ giải mã dữ liệu video kiểm tra điều kiện trong biểu thức (4-2) để xác định xem mẫu q_1 có được xử lý ngoài mẫu q_0 không. Do quyết định về số lượng mẫu cần xử lý được thực hiện độc lập ở mỗi bên của đoạn, nên có thể xử lý một mẫu ở một bên và hai mẫu ở một bên khác.

$$|p_{2,0} - 2p_{1,0} + p_{0,0}| + |p_{2,3} - 2p_{1,3} + p_{0,3}| < 3/16\beta \quad (4-1)$$

$$|q_{2,0} - 2q_{1,0} + q_{0,0}| + |q_{2,3} - 2q_{1,3} + q_{0,3}| < 3/16\beta \quad (4-2)$$

Để xử lý p_0 và q_0 , trị số trung gian δ được tính toán trước tiên trong biểu thức (5).

$$\delta = (9(q_0 - p_0) - 3(q_1 - p_1) + 8) \gg 4 \quad (5)$$

Nếu trị số tuyệt đối δ lớn hơn hoặc bằng 10 lần t_c , thì biên được coi là mép tự nhiên, cần được giữ nguyên, và bộ giải mã dữ liệu video không thực hiện lọc tách khối trên dòng hiện thời. Nếu không, bộ giải mã dữ liệu video cắt δ trong khoảng từ $-t_c$ đến t_c , như được thể hiện trong biểu thức (6).

$$\Delta_0 = \text{Clip3}(-t_c, t_c, \delta) \quad (6)$$

Bộ giải mã dữ liệu video cập nhật lần lượt các trị số p_0 và q_0 thành p'_0 và q'_0 , bằng cách cộng và trừ Δ_0 , như được thể hiện trong các biểu thức. (7-1) và (7-2).

$$p'_0 = p_0 + \Delta_0 \quad (7-1)$$

$$q'_0 = q_0 - \Delta_0 \quad (7-2)$$

Để xử lý mẫu thứ hai ở mỗi bên của đoạn 164, tức là, p_1 và q_1 , bộ giải mã dữ liệu video sử dụng các biểu thức (8-1) và (8-2) để nhận các trị số cập nhật p_1' và q_1' .

$$p_1' = p_1 + \text{Clip3}\left(-\frac{t_c}{2}, \frac{t_c}{2}, (((p_2 + p_0 + 1) \gg 1) - p_1 + \Delta_0) \gg 1\right) \quad (8-1)$$

$$q_1' = q_1 + \text{Clip3}\left(-\frac{t_c}{2}, \frac{t_c}{2}, (((q_2 + q_0 + 1) \gg 1) - q_1 - \Delta_0) \gg 1\right) \quad (8-2)$$

Bộ giải mã dữ liệu video thực hiện lọc tách khối cho sắc độ bằng cách xác định trị số cường độ biên, nhưng không thực hiện bất kỳ sự phân tích trị số mẫu nào. Bộ giải mã dữ liệu video chỉ xử lý mẫu thứ nhất ở một trong hai bên của đoạn, tức là, p_0 và q_0 , bởi các biểu thức (7-1) và (7-2), trong đó delta Δ_0 được tính toán như trong biểu thức (9).

$$\Delta_0 = \text{Clip3}(-t_c, t_c, (((q_0 - p_0) \ll 2) + p_1 - q_1 + 4) \gg 3) \quad (9)$$

Để tránh lọc quá mức, bộ giải mã dữ liệu video tính toán hai thông số β và t_c . Bộ giải mã dữ liệu video sử dụng ngưỡng β để kiểm soát cách tách khối được thực hiện, như đoạn có nên được tách khối hay không, quy trình tách khối được sử dụng mạnh hay bình thường, và/hoặc liệu một hoặc hai mẫu ở một bên của đoạn có được xử lý hay không. Khi β hoặc β định tỷ lệ đạt đến hoặc vượt quá (trong biểu thức (1), (2), và (4)), nghĩa là có sự thay đổi lớn hơn cho các trị số mẫu cục bộ, thì việc tách khối có xu hướng bảo toàn hơn, để bảo toàn các chi tiết trong hình ảnh gốc. Mặt khác, các trị số mẫu cục bộ có ít biến đổi hơn (tức là, mịn hơn), và bộ giải mã dữ liệu video thực hiện lọc tách khối linh hoạt hơn.

Bộ giải mã dữ liệu video chủ yếu sử dụng trị số cắt t_c để kiểm soát sự thay đổi lớn nhất của độ lớn mẫu, trừ các biểu thức (2-5) và (2-6). Trong HEVC, ví dụ, trong chế độ lọc thông thường hoặc lọc cho sắc độ, sự thay đổi độ lớn mẫu bị hạn chế sao cho sự thay đổi không vượt quá $\pm t_c$ cho mẫu thứ nhất ở một bên của đoạn (có thể áp dụng cho độ chói và sắc độ), hoặc $\pm t_c/2$ cho mẫu thứ hai (chỉ áp dụng cho độ chói). Trong HEVC, đối với chế độ lọc mạnh, trong đó có sự thay đổi lớn hơn về cường độ, sự thay đổi tối đa bị hạn chế ở $\pm 2t_c$ cho ba mẫu được xử lý trên mỗi bên của đoạn.

Các trị số của β và t_c chủ yếu phụ thuộc vào các trị số thông số lượng tử hóa (QP) từ khối trái P và khối phải Q. Cụ thể hơn, bộ giải mã dữ liệu video có thể sử dụng lần lượt trị số trung bình của các QP từ P và Q, ký hiệu là $QP_{ave} = (QP_P + QP_Q + 1) \gg 1$, là chỉ số để tìm kiếm hai bảng tìm kiếm 1-D (LUT) cho β và t_c . Mặc dù chỉ số tìm kiếm để tìm trị số t_c có thể được điều chỉnh bằng cách cộng hai, tức là, $(QP_{ave} + 2)$, nếu cường độ biên

bằng 2, hệ số trội xác định các trị số β và t_c vẫn là QP_{ave} . Trong cả hai LUT, các trị số mục nhập tăng đơn điệu với trị số của các chỉ số tìm kiếm, có nghĩa là QP_{ave} càng cao thì các trị số β và t_c càng lớn. Do đó, quy trình lọc tách khối nặng hơn có nhiều khả năng được chọn hơn và cho phép thay đổi độ lớn lớn hơn. Ngược lại, QP thấp hơn dẫn đến các trị số nhỏ hơn hoặc thậm chí bằng 0 cho β và t_c . Khi được lập mã với QP thấp, hình ảnh thường có các xáo ảnh dạng khối ít hơn, hoặc nhỏ hơn rõ rệt và do đó cần nhẹ hơn hoặc thậm chí không cần tách khối.

Các chỉ số sử dụng để tìm kiếm β và t_c trong LUT, ký hiệu là idx_β và idx_{t_c} , có thể được điều chỉnh thêm bằng hai thông số tc_offset_div2 và $beta_offset_div2$ (xem các biểu thức (10-1) và (10-2)),

$$idx_\beta = QP_{ave} + 2 \times beta_offset_div2 \quad (10-1)$$

$$idx_{t_c} = QP_{ave} + 2 \times (BS - 1) + 2 \times tc_offset_div2, \quad (10-2)$$

trong đó tc_offset_div2 và $beta_offset_div2$ được gửi trong phần đầu lát hoặc tập hợp thông số hình ảnh (picture parameter set - PPS). Điều này dẫn đến bộ mã hóa có khả năng làm thích ứng cường độ tách khối phụ thuộc vào các đặc điểm chuỗi, chế độ mã hóa, và các hệ số khác.

Quy trình tách khối HEVC có hai giai đoạn tuần tự, có thể cho phép song song hóa. Trong giai đoạn thứ nhất, bộ giải mã dữ liệu video lọc tất cả các biên khối dọc trong hình ảnh, và trong giai đoạn thứ hai, bộ giải mã dữ liệu video lọc tất cả các biên khối ngang. Trong giai đoạn thứ hai, các mẫu được sử dụng để quyết định chế độ và lọc là các đầu ra của giai đoạn thứ nhất. Trong mỗi giai đoạn, trong đó các biên được tách khối đều song song và cách nhau ít nhất 8 mẫu, các mẫu liên quan đến việc tách khối một biên không chồng lên các mẫu liên quan đến việc tách khối các biên khác bất kỳ. Trong bối cảnh này, các mẫu liên quan đến việc tách khối một biên bao gồm lên đến ba mẫu cần được lọc trên mỗi mặt của biên và lên đến bốn mẫu trên mỗi mặt để hỗ trợ quyết định lọc và chế độ, và do đó một biên có thể được tách khối song song với các biên khác bất kỳ.

Bộ lọc tách khối HEVC lọc các mẫu gần biên khối và cắt các thay đổi trong các độ lớn mẫu mà lớn hơn số lượng ngưỡng. Bộ lọc tách khối HEVC hoạt động theo ba chế độ với ba mức cường độ lọc khác nhau. Ba chế độ, hoặc cường độ lọc, ở đây được gọi là mạnh, bình thường, và không (tức là, không lọc), với các hoạt động cục bộ ngày càng tăng

của các mẫu gần biên khối (xem các biểu thức (1) và từ (2-1) đến (2-6)). Trong chế độ lọc mạnh, bộ lọc tách khối HEVC lọc ba mẫu ở mỗi bên của biên khối bằng cách lọc thông thấp (xem biểu thức từ (3-1) đến (3-3)). Đối với chế độ lọc thông thường, bộ lọc tách khối HEVC lọc ít nhất một mẫu gần với biên nhất và, trên hai mặt của biên, cũng có thể lọc mẫu gần nhất thứ hai nếu các mẫu bên trong đủ mịn (xem các biểu thức (4-1) và (4-2)). Bộ lọc tách khối HEVC thực hiện cắt, được kiểm soát bởi thông số t_c . Đối với lọc mạnh, sự thay đổi độ lớn mẫu bị giới hạn là không lớn hơn $2t_c$. Đối với lọc bình thường, sự thay đổi độ lớn của các mẫu thứ nhất và thứ hai bị giới hạn lần lượt là không lớn hơn t_c và $t_c/2$, (xem các biểu thức từ (6) đến (8-2)). Các trị số cường độ biên khác nhau chỉ tạo ra sự chênh lệch trong t_c , tức là, đoạn có cường độ biên bằng 2 là lớn hơn t_c so với đoạn với cường độ biên bằng 1.

Bộ lọc tách khối HEVC về cơ bản tuân theo cơ cấu tách khối H.264/AVC và kế thừa các đặc điểm chính của bộ lọc tách khối H.264/AVC, như xác định cường độ biên dựa vào các điều kiện lập mã, cường độ lọc nhiều mức từ mạnh đến không, QP và các thông số phụ thuộc cường độ biên β và t_c . So với bộ lọc tách khối H.264/AVC, các phần tử thiết kế mới trong bộ lọc tách khối HEVC cho phép xử lý song song dễ dàng hơn và phù hợp hơn với cấu trúc lập mã kích thước khối lớn hơn của HEVC, nhưng không cải thiện nhiều hiệu quả lập mã. Do đó, tách khối HEVC, được xem là sự cân bằng tốt giữa độ phức tạp tính toán và hiệu quả lập mã tại thời điểm hoàn thành HEVC, có thể được đơn giản hóa quá mức nếu xét đến các khả năng phần cứng được triển khai nhiều ngày nay. Các kỹ thuật của sáng chế này có khả năng tận dụng nhiều tài nguyên tính toán phần cứng để đạt được sự cải thiện hiệu quả lập mã đáng kể, đồng thời vẫn giữ được thiết kế thân thiện song song hóa.

Sáng chế đề xuất các kỹ thuật có thể cải thiện dựa trên các khía cạnh của quy trình lọc tách khối, bao gồm quy trình lọc tách khối HEVC. Một ví dụ về nhược điểm của lọc tách khối HEVC là chỉ sử dụng ba mức để biểu diễn độ mịn của vùng biên, tương ứng với ba mức cường độ lọc, có thể quá thô. Một ví dụ khác về nhược điểm của lọc tách khối HEVC là chỉ hai từ bốn dòng của đoạn được sử dụng khi quyết định chế độ tách khối và lựa chọn lọc, và trong mỗi dòng, chỉ bốn mẫu từ hai bên được sử dụng. Vì kích thước khối của bộ mã hóa-giải mã dữ liệu video thế hệ tiếp theo có thể lên đến 128×128 hoặc thậm

chí lớn hơn, việc sử dụng một phần nhỏ mẫu như vậy để quyết định chế độ có thể không phản ánh chính xác hoạt động thực của vùng biên và có thể nhạy đối với nhiễu.

Một ví dụ khác về nhược điểm của lọc tách khối HEVC là các mẫu cần được lọc ở hai bên của đoạn có thể bao gồm nhiều nhất 3 mẫu hoặc ít nhất một mẫu, có thể không đủ mẫu để đưa ra kết quả tốt để tách khối khối lớn. Một ví dụ khác về nhược điểm của lọc tách khối HEVC là tổng cộng có năm bộ lọc 4 bậc hoặc 5 bậc xác định trước, được gán trước cho ba mẫu trong chế độ lọc mạnh (xem các biểu thức từ (3-1) đến (3-3)) và hai mẫu trong chế độ lọc thông thường (xem các biểu thức từ (5) đến (8-2)). Số lượng tùy chọn giới hạn, độ dài ngắn, và sự không linh hoạt trong việc lựa chọn của bộ lọc tách khối HEVC có thể gây ra hiệu quả tách khối thấp hơn.

Một ví dụ khác về nhược điểm của lọc tách khối HEVC là các đoạn với cường độ biên khác nhau có thể có các hoạt động cục bộ khá khác nhau, nhưng dùng chung cùng bộ lọc tách khối, mặc dù trị số cắt lớn hơn đối với cường độ biên bằng 2. Một ví dụ khác về nhược điểm của lọc tách khối HEVC là bộ lọc tách khối HEVC không phân biệt khối P và khối Q. Cường độ lọc được xác định bởi trị số trung bình đạo hàm bậc hai của P và Q (xem các biểu thức (1) và từ (2-1) đến (2-2)), và các trị số thông số β và t_c phụ thuộc vào QP trung bình P và Q. Có thể có sự tổn thất về hiệu quả trong việc xử lý khối P và khối Q theo cùng một cách, đối với các kịch bản lập mã trong đó một trong số khối P hoặc khối Q có thể làm mịn trong khi khối khác giàu chi tiết, đây không phải là một tình huống hiếm gặp.

Sáng chế đề xuất một số kỹ thuật để cải thiện hiệu suất của quy trình lọc tách khối được dùng để lập mã dữ liệu video hoặc xử lý dữ liệu video. Theo một ví dụ, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để thực hiện phân tích lọc (ví dụ, lựa chọn lọc hoặc sự thích ứng cường độ lọc) dựa vào các thông số của việc chia khối. Theo một ví dụ, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định độ dài của bộ lọc tách khối và hỗ trợ lọc (ví dụ, hỗ trợ lọc được giảm tần suất lấy mẫu) dựa vào độ dài của khối trực giao với biên được lọc. Theo một ví dụ khác, độ dài của khối đồng chỉnh với biên được lọc ảnh hưởng đến quy trình thực hiện quyết định, ví dụ, cho phép lấy mẫu con theo không gian của các mẫu biên được sử dụng để thực hiện quyết định lọc.

Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để thích ứng khoảng tin cậy mẫu đầu ra, còn gọi là hệ số T_c , theo chức năng của khối phân chia các thông số (ví dụ, các kích thước) và vị trí liên quan trong khối của mẫu đang được xử

lý để áp dụng cho thông số giới hạn Tc. Tc giới hạn độ lệch, dương hoặc âm, trong trị số mẫu được tách khỏi từ mẫu được giải mã và là đầu ra cho quy trình tách khối.

Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để sử dụng các thông số của khối xử lý (kích thước khối) để hạn chế tập hợp mẫu có sẵn cho quy trình ra quyết định và lọc, đồng thời duy trì thiết kế về quy trình ra quyết định và lọc. Bộ giải mã dữ liệu video 300 có thể tạo ra các mẫu cần thiết cho quy trình lọc nhưng không có sẵn do các hạn chế sử dụng quy trình cụ thể, như quy trình ngoại suy hoặc đệm. Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể sử dụng các đặc tính biến đổi, như thông tin về kiểu biến đổi/chức năng cơ bản và các hệ số biến đổi báo hiệu khác không đến bộ giải mã dữ liệu video 300 và do đó có mặt trong khối tái tạo của mẫu, để lựa chọn các thông số lọc tách khối, như kiểu lọc, độ dài lọc, các thông số giới hạn Tc, bước bỏ qua mẫu theo không gian. Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể sử dụng các thông số lượng tử hóa của các khối P và Q để thiết lập các ngưỡng trên sự lựa chọn P và Q một cách độc lập.

Theo một ví dụ, đối với biên khối tách các khối có các kích thước khác nhau, bộ giải mã dữ liệu video 300 có thể thực hiện quyết định lọc tách khối bất đối xứng. Ví dụ, khi biên khối tách các khối theo kích thước khác nhau, bộ giải mã dữ liệu video 300 có thể lựa chọn lọc tách khối bất đối xứng.

Mặt khác, nếu bộ giải mã dữ liệu video 300 xác định rằng bộ lọc bất đối xứng cần được sử dụng, thì quy trình tách khối trên cả hai mặt của biên có thể khác nhau. Ví dụ, các thông số của bộ lọc tách khối, như độ dài nhánh, hệ số lọc, quy trình cắt hoặc tiêu chuẩn hóa có thể khác nhau trên cả hai cạnh của biên được lọc. Ngoài ra hoặc theo cách khác, các thông số cho các quy trình thực hiện quyết định, như độ dài bậc của lọc phân tích, các hệ số lọc phân tích, cắt, tiêu chuẩn hóa hoặc các ngưỡng có thể khác nhau trên cả hai cạnh của biên được lọc.

Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể xác định các thông số để lọc bất đối xứng dựa vào một hoặc nhiều thông số chế độ lập mã của các khối trên biên. Các ví dụ về các thông số chế độ lập mã này bao gồm chế độ lập mã, chế độ dự đoán, kiểu lát, hoặc các thông số khác.

Các ví dụ không giới hạn về phương án thực hiện của các kỹ thuật đưa vào ở trên sẽ được mô tả sau đây. Theo một ví dụ, quy trình phân tích lọc (lựa chọn lọc hoặc sự thích ứng cường độ lọc được mở rộng để bao gồm các thông số của khối phân chia. Ví dụ, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định độ dài của bộ lọc tách khối và hỗ trợ lọc (ví dụ, hỗ trợ lọc được giảm tần suất lấy mẫu) dựa vào độ dài của khối trực giao đến biên được lọc ảnh hưởng. Theo một ví dụ khác, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để tạo ra quyết định dựa vào độ dài của khối đồng chỉnh với biên được lọc, như cho phép lấy mẫu phụ theo không gian các mẫu biên được sử dụng để tạo ra quyết định lọc.

Mã giả cung cấp dưới đây đưa ra phương án thực hiện làm ví dụ về sự tìm ra thông số tách khối thực hiện bởi bộ giải mã dữ liệu video 300:

VERT_SAMPLE_SKIP là biến số xác định các thông số lấy mẫu lại theo không gian (số lượng mẫu được bỏ qua) chỉ ra số lượng mẫu sử dụng để phân loại các mẫu biên.

DB_BLOCK_SIZE_CLASSIFIER là các biến số xác định số lượng mẫu sử dụng để phân loại các mẫu biên.

Function isSmoothAsymArea(const CodingUnit& cu, const DeblockEdgeDir edgeDir, const Position& localPos)

```

if (edgeDir == EDGE_VER)
{
    blkWidthQ    =    std::min((SizeType)LARGE_BLOCK_SIZE_CLASSIFIER,
blkSizeQ.width >> 1);

    blkHeightQ = blkSizeQ.height;

    blkWidthP    =    std::min((SizeType)LARGE_BLOCK_SIZE_CLASSIFIER,
blkSizeP.width >> 1);

    blkHeightP = blkSizeP.height;

    piSrcP = piSrcQ - blkWidthP;

```

```

    }

    else // (edgeDir == EDGE_HOR)

    {

        blkWidthQ = blkSizeQ.width;

        blkHeightQ    =    std::min((SizeType)    DB_BLOCK_SIZE_CLASSIFIER,
blkSizeQ.height >> 1);

        blkWidthP = blkSizeP.width;

        blkHeightP    =    std::min((SizeType)    DB_BLOCK_SIZE_CLASSIFIER,
blkSizeP.height >> 1);

        piSrcP = piSrcQ - blkHeightP * iStride;
    }

    if (edgeDir == EDGE_VER)

    {

        Int i;

        i = localPos.y - cuPosLuma.y;

        // tính toán biến cục bộ theo chiều ngang

        for (Int j = 1; j < (blkWidthQ - 1); j+=VERT_SAMPLE_SKIP)

        {

            Int loc = i * iStride + j;

            varQ += abs(piSrcQ[loc - 1] + piSrcQ[loc + 1] - (piSrcQ[loc] << 1));

            loc = (i + 3) * iStride + j;

            varQ += abs(piSrcQ[loc - 1] + piSrcQ[loc + 1] - (piSrcQ[loc] << 1));

        }

    }

    else if (edgeDir == EDGE_HOR)

```

```

{
    Int j;

    j = localPos.x - cuPosLuma.x;

    // tính toán biên cục bộ theo chiều dọc

    for (Int i = 1; i < (blkHeightQ - 1); i+=HOR_SAMPLE_SKIP)
    {
        Int loc = i * iStride + j;

        varQ += abs(piSrcQ[loc - iStride] + piSrcQ[loc + iStride] - (piSrcQ[loc] << 1));

        loc = i * iStride + (j + 3);

        varQ += abs(piSrcQ[loc - iStride] + piSrcQ[loc + iStride] - (piSrcQ[loc] << 1));

    }
}

```

Theo các ví dụ ở trên, pSrcP có thể, ví dụ, đại diện cho con trỏ đến khối ở một bên của biên cần được lọc tách khối, và pSrcQ có thể đại diện cho con trỏ đến khối tại một bên khác của biên cần lọc tách khối.

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định khoảng tin cậy mẫu đầu ra, còn gọi là hệ số Tc, để giới hạn độ lệch của trị số mẫu được tách khối từ mẫu được giải mã. Tc có thể là đầu vào của quy trình tách khối. Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để làm thích ứng Tc dưới dạng chức năng của khối phân chia thông số, như kích thước khối, và vị trí liên quan trong khối hoặc trong biên được lọc của mẫu đang được xử lý áp dụng vào thông số giới hạn Tc. Theo một ví dụ, vị trí liên quan của mẫu đang được xử lý là vị trí của mẫu đang được xử lý so với biên được lọc.

Theo một phương án thực hiện làm ví dụ, thông số giới hạn Tc có thể được xác định là hàm của vị trí liên quan của dòng/cột đang được lọc trong biên được lọc, trong đó i là chỉ số của dòng/cột được xử lý hiện thời của biên được lọc, bắt đầu lần lượt từ vị trí đỉnh/bên phải của biên.

Mã giả dưới đây cung cấp một ví dụ về quy trình thực hiện ứng dụng Tc :

```

for( int i = 0; i < boundaryLength; i++ )
{
    for( int j = 0; j < deblockedSamplesOverBoundary; j++ )
    {
        delta = deblockingFilterResult[i][j] - deblockingFilterInpt[i][j];

        delta1 = Clip3(-Tc[i], Tc[i], delta );
    }
}

```

Theo một số ví dụ, trị số Tc có thể được xác định cho khối làm chức năng của vị trí so với vị trí trên cùng bên trái của khối đang được xử lý:

```

y = coordinateCurrentlyProcessedSample.y - coordinateTopLeftSampleOfBlock.y;
x = coordinateCurrentlyProcessedSample.x - coordinateTopLeftSampleOfBlock.x;

delta1 = Clip3( -tc2[y][x], tc2[y][x], delta[y][x] );

```

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để sử dụng các thông số của khối xử lý, như kích thước khối, để hạn chế tập hợp mẫu có sẵn để quy trình ra quyết định và lọc, với sự duy trì thiết kế về quy trình ra quyết định và lọc. Bộ giải mã dữ liệu video 300 có thể tạo ra các mẫu mà được yêu cầu cho quy trình lọc nhưng không có sẵn do các hạn chế sử dụng quy trình cụ thể, như quy trình ngoại suy hoặc đệm.

Giải mã dưới đây cung cấp phương án thực hiện làm ví dụ. Theo một số thực thi, nhóm bộ lọc để xử lý biên có thể được xác định. Sự lựa chọn của bộ lọc có thể dựa vào sự kết hợp của các kích thước khối nối vào biên khối và kết quả của việc phân loại các mẫu biên.

DB_FILTER_BLOCK_SIZE1, DB_FILTER_BLOCK_SIZE2 và DB_FILTER_BLOCK_SIZE3 là các biến số, mỗi biến số xác định số lượng mẫu được sử dụng để lọc các mẫu biên, xác định hỗ trợ lọc trong 3 bộ lọc làm ví dụ.

```

if (edgeDir == EDGE_VER)
{
    minTUSize = std::min(blkSizeP.width, blkSizeQ.width);
}
else
{
    minTUSize = std::min(blkSizeP.height, blkSizeQ.height);
}

if (minTUSize >= DB_FILTER_BLOCK_SIZE1 )
{
    if (isFilterApplicable (cu, edgeDir, localPos))
    {
        // Apply symmetrical filter 1 on P và Q samples
        apply_filter1(cu, edgeDir, localPos));
    }
}

else if ((MODE_INTRA != cuP.predMode) && (MODE_INTRA !=
cuQ.predMode))
{
    if (edgeDir == EDGE_VER)
    {
        if (
            ( blkSizeP.width >= DB_FILTER_BLOCK_SIZE1) && (blkSizeQ.width >=
8) )
            ||

```

```

    ( (blkSizeQ.width >= DB_FILTER_BLOCK_SIZE1) && (blkSizeP.width >=
8) )

    )

    {
    if (isFilterApplicable (cu, edgeDir, localPos))
    {
    // Apply asymmetric filtering
        if (blkSizeP.width >= DB_FILTER_BLOCK_SIZE1)
        {
            apply_filter2_onP(cu, edgeDir, localPos);
            apply_filter3_onQ(cu, edgeDir, localPos));
        }
    }
    }
    }
    else
    {
    if (
    ( (blkSizeP.height >= DB_FILTER_BLOCK_SIZE1) && (blkSizeQ.height >=
8) )

    ||

    ( (blkSizeQ.height >= DB_FILTER_BLOCK_SIZE1) && (blkSizeP.height >=
8) )

    )

    {
    if (isFilterApplicable (cu, edgeDir, localPos))

```

```

{
    // Apply asymmetric filtering
    if (blkSizeP.width >= DB_FILTER_BLOCK_SIZE1)
    {
        apply_filter2_onP(cu, edgeDir, localPos));
        apply_filter3_onQ(cu, edgeDir, localPos));
    }
}
}
}
}
}

```

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để sử dụng các đặc tính biến đổi, như thông tin về kiểu biến đổi/chức năng cơ bản và các hệ số biến đổi báo hiệu khác không đến bộ giải mã dữ liệu video 300 và do đó có mặt trong khối tái tạo của mẫu, để lựa chọn các thông số lọc tách khối, như kiểu lọc, độ dài lọc, các thông số giới hạn Tc, và/hoặc bước bỏ qua mẫu theo không gian.

Giải mã dưới đây cung cấp phương án thực hiện làm ví dụ. Giả sử rằng bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 sử dụng nhiều tập hợp biến đổi, ví dụ. {T1,T2,T3}, với bộ mã hóa dữ liệu video 200 lựa chọn biến đổi tối ưu có thể áp dụng cho các hướng ngang và dọc. Tại bộ giải mã dữ liệu video 300, khả năng áp dụng biến đổi cho hướng ngang/dọc có thể được suy ra từ các phần tử cú pháp của luồng bit. Quy trình tìm ra sử dụng trong WD của VVC (bản dự thảo 2) là cụ thể dưới đây,

Bảng 1 - Bảng biến đổi và ánh xạ báo hiệu

MTS_CU_flag	MTS_Hor_flag	MTS_Ver_flag	Intra		Inter	
			Horizontal	Vertical	Horizontal	Vertical

0			DCT2			
1	0	0	DST7	DST7	DCT8	DCT8
	0	1	DCT8	DST7	DST7	DCT8
	1	0	DST7	DCT8	DCT8	DST7
	1	1	DCT8	DCT8	DST7	DST7

Giải mã sau đây có thể được sử dụng cho sự tìm ra thông số lọc tách khối:

```

if (MTS_CU_flag == 0)
{
    ApplyDeblockingProcessType1();
} else
{
    Int TransformSetId = (MTS_Hor_flag << 1) + MTS_Ver_flag;
    switch( TransformSetId )
    {
        case 0:    ApplyDeblockingProcessType2(); break;
        case 1:    ApplyDeblockingProcessType3(); break;
        case 2:    ApplyDeblockingProcessType4(); break;
        case 3:    ApplyDeblockingProcessType5(); break;
    }
}

```

Dưới đây là ví dụ về chức năng ApplyDeblockingProcessType1. Ví dụ, sự thiết kế các chức năng khác, chức năng tách khối, còn có thể theo sau kiểu tương tự.

```
function ApplyDeblockingProcessType1()
```

{

1. Suy ra độ dài của bộ phân loại dựa vào kích thước khối, trực giao với biên tạo khối, kiểu biến đổi và chỉ số hệ số biến đổi được lập mã khác không, và ước lượng hoạt động cục bộ.
2. Suy ra độ dài về việc tách khối dựa vào kích thước khối, trực giao với biên tạo khối, kiểu biến đổi và chỉ số hệ số biến đổi được lập mã khác không, và ước lượng hoạt động cục bộ.
3. Suy ra các mẫu bỏ qua trong khi bộ phân loại hoặc quy trình tách khối dựa vào kích thước khối, trực giao với biên tạo khối, kiểu biến đổi và chỉ số hệ số biến đổi được lập mã khác không, và ước lượng hoạt động cục bộ.
4. Suy ra các thông số về tách khối, như Tc, các ngưỡng cục bộ, sự điều chỉnh QP dựa vào kích thước khối, trực giao với biên tạo khối, kiểu biến đổi và chỉ số hệ số biến đổi được lập mã khác không, và ước lượng hoạt động cục bộ.

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để sử dụng các thông số lượng tử hóa của các khối P và Q để thiết lập các ngưỡng trên sự lựa chọn P và Q một cách độc lập.

$iQP_p = cuP.qp;$

$iQP_q = cuQ.qp;$

Các giả mã sau được đưa ra cho các thông số khối P. Việc xử lý của khối Q có thể là tương tự.

$const\ int\ iIndexTCp = Clip3(0, MAX_QP + DEFAULT_INTRA_TC_OFFSET, Int(iQP_p + DEFAULT_INTRA_TC_OFFSET * (uiBs_loc - 1) + (tcOffsetDiv2 << 1)));$

$const\ int\ iIndexBp = Clip3(0, MAX_QP, iQP + (betaOffsetDiv2 << 1));$

$const\ int\ iTcp = sm_tcTable[iIndexTC] * iBitdepthScale;$

$const\ int\ iBetap = sm_betaTable[iIndexB] * iBitdepthScale;$

$const\ int\ iSideThresholdp = (iBeta + (iBeta >> 1)) >> 3;$

```
const int iThrCutp = iTc * 10;
```

Vẫn theo một ví dụ khác, phương án thực hiện sau đây của việc tách khối phụ thuộc vị trí có thể được sử dụng kết hợp với các đặc điểm khác ở đây.

Cho numberQSide và numberPSide là các độ dài của bộ lọc áp dụng cho dòng hiện thời. Như được giới thiệu ở trên, độ dài lọc numberPSide và numberQSide có thể được dựa vào kích thước khối. Theo ví dụ này, hai bộ lọc được phát triển áp dụng cho bộ lọc 7 và 3 điểm ảnh của biên khối. Các thông số của trị số cắt phụ thuộc vị trí được biểu diễn thông qua Tc7 và Tc3.

```
const char Tc7[7] = { 6, 5, 4, 3, 2, 1, 1};
```

```
const char Tc3[3] = { 6, 4, 2 };
```

```
const char *pTcP = (numberPSide == 3) ? Tc3 : Tc7;
```

```
const char *pTcQ = (numberQSide == 3) ? Tc3 : Tc7;
```

```
for (int thePos = 0; thePos < numberPSide; thePos++)
```

```
{
```

```
src = piSrcP[-iOffset*thePos];
```

```
int cvalue = (tc * pTcP[thePos]) >> 1;
```

```
piSrcP[-iOffset * thePos] = Clip3(src - cvalue, src + cvalue,
((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6));
```

```
}
```

```
for (int thePos = 0; thePos < numberQSide; thePos++)
```

```
{
```

```
src = piSrcQ[iOffset*thePos];
```

```
int cvalue = (tc * pTcQ[thePos]) >> 1;
```

```
piSrcQ[iOffset*thePos] = Clip3(src - cvalue, src + cvalue,
((refMiddle*dbCoeffsQ[thePos] + refQ * (64 - dbCoeffsQ[thePos]) + 32) >> 6));
```

}

Theo ví dụ ở trên, “Src=piSrcP[-iOffset*thePos]” minh họa trị số mẫu trước khi lọc giải khối. “((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6)” minh họa áp dụng hoạt động lọc giải khối. “refP” hoặc “refQ” là trị số kết hợp các mẫu tuyến tính tại một cạnh (khối P hoặc cạnh Q) của biên khối, hoặc trị số lọc tại một cạnh (khối P hoặc cạnh Q) của biên khối. “refMiddle” là trị số kết hợp các mẫu tuyến tính tại cả hai cạnh của biên khối, hoặc trị số lọc tại cả hai cạnh của biên khối. “Clip3(src - cvalue, src + cvalue, ((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6))” minh họa cắt trị số lọc tách khối trong miền src-value và src+cvalue, là trị số mẫu trước khi được lọc cộng/trừ ngưỡng cvalue, trong đó cvalue có khả năng thích nghi dựa vào vị trí của mẫu và kích thước khối.

Như được mô tả ở trên, bộ giải mã dữ liệu video 300 có thể xác định numberQSide và numberPSide dựa vào kích thước khối có lọc giải khối áp dụng. refMiddle là trị số sau khi áp dụng lọc vào các mẫu tại mỗi cạnh của biên của khối P và Q. refP và refQ là các trị số sau khi áp dụng lọc vào các mẫu trong khối P và khối Q.

Vẫn theo một ví dụ khác, phương án thực hiện sau đây của việc tách khối phụ thuộc vị trí có thể được sử dụng kết hợp với các đặc điểm khác ở đây.

Cho numberQSide và numberPSide là độ dài của bộ lọc áp dụng vào dòng hiện thời. Theo ví dụ này, hai bộ lọc được phát triển áp dụng cho bộ lọc 7 và 3 điểm ảnh của biên khối. Các thông số của trị số cắt phụ thuộc vị trí được biểu diễn thông qua Tc7 và Tc3.

```
const char Tc7[4] = { 3, 2, 1, 1 };
```

```
const char Tc3[3] = { 3, 2, 1 };
```

```
const char *pTcP = (numberPSide == 3) ? Tc3 : Tc7;
```

```
const char *pTcQ = (numberQSide == 3) ? Tc3 : Tc7;
```

```
char nSP = (numberPSide == 3) ? 0: 1;
```

```
char nSQ = (numberQSide == 3) ? 0: 1;
```

```

for (int thePos = 0; thePos < numberPSide; thePos++)
{
    src = piSrcP[-iOffset*thePos];

    int cvalue = tc * pTcP[thePos >> nSP];

    piSrcP[-iOffset * thePos] = Clip3(src - cvalue, src + cvalue,
    ((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6));
}

for (int thePos = 0; thePos < numberQSide; thePos++)
{
    src = piSrcQ[iOffset*thePos];

    int cvalue = tc * pTcQ[thePos >> nSQ];

    piSrcQ[iOffset*thePos] = Clip3(src - cvalue, src + cvalue,
    ((refMiddle*dbCoeffsQ[thePos] + refQ * (64 - dbCoeffsQ[thePos]) + 32) >> 6));
}

```

Theo ví dụ ở trên, “Src=piSrcP[-iOffset*thePos]” minh họa trị số mẫu trước khi lọc giải khối. “((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6)” minh họa áp dụng hoạt động lọc giải khối. “refP” và “refMiddle” là các ví dụ về sự kết hợp tuyến tính của các mẫu tại một cạnh của biên khối, hoặc các điểm ảnh được lọc tại một cạnh của biên khối. “Clip3(src - cvalue, src + cvalue, ((refMiddle*dbCoeffsP[thePos] + refP * (64 - dbCoeffsP[thePos]) + 32) >> 6))” minh họa cắt trị số lọc tách khối trong miền src-value và src+cvalue, là trị số mẫu trước khi được lọc cộng ngưỡng cvalue, trong đó cvalue có khả năng thích nghi dựa vào vị trí của mẫu và kích thước khối.

Như được mô tả ở trên, bộ giải mã dữ liệu video 300 có thể xác định numberQSide và numberPSide dựa vào kích thước khối có lọc giải khối áp dụng. refMiddle là trị số sau khi áp dụng lọc vào các mẫu tại mỗi cạnh của biên của khối P và Q. refP và refQ là các trị số sau khi áp dụng lọc vào các mẫu trong khối P và khối Q.

Vẫn theo một ví dụ khác, phương án thực hiện sau đây của việc tách khối phụ thuộc vị trí có thể được sử dụng kết hợp với các đặc điểm khác ở đây.

Cho numberQSide và numberPSide là độ dài của bộ lọc áp dụng vào dòng hiện thời. Theo ví dụ này, hai bộ lọc được phát triển áp dụng để lọc 7 và 3 điểm ảnh của biên khối. Các thông số của trị số cắt phụ thuộc vị trí được biểu diễn thông qua Tc7 và Tc3.

```

const char Tc7[4] = { 3, 2, 1, 1 };
const char Tc3[3] = { 3, 2, 1 };
const Pel m4 = piSrc[0];
const Pel m3 = piSrc[-iOffset];
const Pel m5 = piSrc[iOffset];
const Pel m2 = piSrc[-iOffset * 2];
const Pel m6 = piSrc[iOffset * 2];
const Pel m1 = piSrc[-iOffset * 3];
const Pel m7 = piSrc[iOffset * 3];
const Pel m0 = piSrc[-iOffset * 4];

piSrc[-iOffset] = Clip3(m3 - Tc3[0] * tc, m3 + Tc3[0] * tc, ((m1 + 2 * m2 + 2 *
m3 + 2 * m4 + m5 + 4) >> 3));

piSrc[0] = Clip3(m4 - Tc3[0] * tc, m4 + Tc3[0] * tc, ((m2 + 2 * m3 + 2 * m4 + 2
* m5 + m6 + 4) >> 3));

piSrc[-iOffset * 2] = Clip3(m2 - Tc3[1] * tc, m2 + Tc3[1] * tc, ((m1 + m2 + m3
+ m4 + 2) >> 2));

piSrc[iOffset] = Clip3(m5 - Tc3[1] * tc, m5 + Tc3[1] * tc, ((m3 + m4 + m5 + m6
+ 2) >> 2));

piSrc[-iOffset * 3] = Clip3(m1 - Tc3[2] * tc, m1 + Tc3[2] * tc, ((2 * m0 + 3 * m1
+ m2 + m3 + m4 + 4) >> 3));

piSrc[iOffset * 2] = Clip3(m6 - Tc3[2] * tc, m6 + Tc3[2] * tc, ((m3 + m4 + m5 +
3 * m6 + 2 * m7 + 4) >> 3));

```

Theo một ví dụ khác nữa, điều kiện xác định sau đây của lọc bất đối xứng có thể được sử dụng kết hợp với các đặc điểm khác được bộc lộ ở đây .

- Bộ lọc dài có thể vô hiệu hóa trong trường hợp bất đối xứng phụ thuộc vào chế độ dự đoán P hoặc Q.

```
bool    bSidePisLarge    =    (edgeDir    ==    EDGE_VER    &&
cuP.block(COMPONENT_Y).width    >=
SHARP_LARGE_BLOCKS_SIZE_LOWEST_TH)
```

```
|| (edgeDir == EDGE_HOR && cuP.block(COMPONENT_Y).height >=
SHARP_LARGE_BLOCKS_SIZE_LOWEST_TH);
```

```
bool    bSideQisLarge    =    (edgeDir    ==    EDGE_VER    &&
cuQ.block(COMPONENT_Y).width    >=
SHARP_LARGE_BLOCKS_SIZE_LOWEST_TH)
```

```
|| (edgeDir == EDGE_HOR && cuQ.block(COMPONENT_Y).height >=
SHARP_LARGE_BLOCKS_SIZE_LOWEST_TH);
```

```
bool asymmFlag = !(bSidePisLarge && bSideQisLarge);
```

```
if (asymmFlag)
```

```
if ((MODE_INTRA == cuP.predMode) || (MODE_INTRA == cuQ.predMode))
```

```
{
```

```
    bSidePisLarge = 0;
```

```
    bSideQisLarge = 0;
```

```
}
```

```
If (bSidePisLarge)
```

```
    ApplyLongFilterP();
```

```
else
```

```

        ApplyShortFilterP();

    If (bSideQisLarge)

        ApplyLongFilterQ();

    else

        ApplyShortFilterQ();

```

Fig.6 là sơ đồ khối minh họa ví dụ về bộ mã hóa dữ liệu video 200 mà có thể thực hiện các kỹ thuật của sáng chế. Fig.6 được đưa ra nhằm mục đích giải thích và không được xem là làm hạn chế các kỹ thuật như được minh họa và mô tả chung trong sáng chế này. Với mục đích giải thích, sáng chế mô tả bộ mã hóa dữ liệu video 200 trong ngữ cảnh các tiêu chuẩn mã hóa dữ liệu video như tiêu chuẩn mã hóa dữ liệu video HEVC và tiêu chuẩn mã hóa dữ liệu video H.266 đang phát triển. Tuy nhiên, các kỹ thuật của sáng chế này không bị giới hạn ở các tiêu chuẩn mã hóa dữ liệu video, và thường được áp dụng cho quy trình mã hóa và giải mã dữ liệu video.

Theo ví dụ trên Fig.6, bộ mã hóa dữ liệu video 200 bao gồm bộ nhớ dữ liệu video 230, đơn vị chọn chế độ 202, đơn vị tạo dư 204, đơn vị xử lý biến đổi 206, đơn vị lượng tử hóa 208, đơn vị lượng tử hóa ngược 210, đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, đơn vị lọc 216, bộ đệm hình ảnh giải mã (decoded picture buffer - DPB) 218 và đơn vị mã hóa entropy 220.

Bộ nhớ dữ liệu video 230 có thể lưu trữ dữ liệu video cần được mã hóa bởi các thành phần của bộ mã hóa dữ liệu video 200. Bộ mã hóa dữ liệu video 200 có thể thu dữ liệu video lưu trữ trong bộ nhớ dữ liệu video 230 từ, ví dụ, nguồn dữ liệu video 104 (Fig.1). DPB 218 có thể hoạt động như bộ nhớ hình tham chiếu lưu trữ dữ liệu video tham chiếu để sử dụng trong dự đoán dữ liệu video tiếp theo bằng bộ mã hóa dữ liệu video 200. Bộ nhớ dữ liệu video 230 và DPB 218 có thể được tạo ra bởi bất kỳ trong số nhiều thiết bị nhớ, như bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory - DRAM), bao gồm DRAM đồng bộ (synchronous DRAM - SDRAM), RAM từ điện trở (magnetoresistive RAM - MRAM), RAM điện trở (resistive RAM - RRAM), hoặc các loại

thiết bị nhớ khác. Bộ nhớ dữ liệu video 230 và DPB 218 có thể được cung cấp bởi cùng một thiết bị nhớ hoặc các thiết bị nhớ riêng. Theo các ví dụ khác nhau, bộ nhớ dữ liệu video 230 có thể nằm trên chip cùng các bộ phận khác của bộ mã hóa dữ liệu video 200, như được minh họa, hoặc ngoài chip so với các bộ phận này.

Theo sáng chế này, việc tham chiếu đến bộ nhớ dữ liệu video 230 không được hiểu là bị giới hạn ở bộ nhớ bên trong bộ mã hóa dữ liệu video 200, trừ khi được mô tả cụ thể như vậy, hoặc bộ nhớ bên ngoài bộ mã hóa dữ liệu video 200, trừ khi được mô tả cụ thể như vậy. Thay vào đó, tham chiếu đến bộ nhớ dữ liệu video 230 nên được hiểu là bộ nhớ tham chiếu lưu trữ dữ liệu video mà bộ mã hóa dữ liệu video 200 nhận để mã hóa (ví dụ, dữ liệu video cho khối hiện thời sẽ được mã hóa). Bộ nhớ 106 trên Fig.1 có thể cũng cung cấp lưu trữ tạm thời các đầu ra từ các đơn vị khác nhau của bộ mã hóa dữ liệu video 200.

Các đơn vị khác nhau trên Fig.6 được minh họa để giúp hiểu các hoạt động được thực hiện bởi bộ mã hóa dữ liệu video 200. Các đơn vị có thể được thực hiện như các mạch chức năng cố định, các mạch lập trình được, hoặc kết hợp của chúng. Các mạch chức năng cố định là các mạch có chức năng cụ thể, và thiết lập trước trên các hoạt động mà có thể được thực hiện. Các mạch lập trình được là các mạch mà có thể được lập trình để thực hiện một số nhiệm vụ, và cung cấp chức năng linh hoạt trong các hoạt động mà có thể được thực hiện. Ví dụ, các mạch lập trình được có thể thực thi phần mềm hoặc firmware khiến các mạch lập trình được hoạt động theo cách được xác định bởi các lệnh của phần mềm hoặc firmware. Các mạch chức năng cố định có thể thực thi các lệnh phần mềm (ví dụ, để nhận các thông số hoặc xuất ra các thông số), nhưng các loại hoạt động mà các mạch chức năng cố định thực hiện nói chung là không thay đổi. Theo một số ví dụ, một hoặc nhiều trong số các đơn vị có thể là các khối mạch khác nhau (chức năng cố định hoặc có thể lập trình được), và theo một số ví dụ, một hoặc nhiều đơn vị có thể là các mạch tích hợp.

Bộ mã hóa dữ liệu video 200 có thể bao gồm đơn vị logic số học (arithmetic logic unit - ALU), đơn vị chức năng cơ bản (elementary function unit - EFU), mạch kỹ thuật số, mạch tương tự và/hoặc lõi có thể lập trình, được tạo ra từ các mạch lập trình được. Theo các ví dụ mà các hoạt động của bộ mã hóa dữ liệu video 200 được thực hiện bằng cách sử dụng phần mềm được thực thi bởi các mạch lập trình được, bộ nhớ 106 (Fig.1) có thể lưu trữ các lệnh (ví dụ, mã đối tượng) của phần mềm mà bộ mã hóa dữ liệu video 200 nhận và

thực thi, hoặc một bộ nhớ khác trong bộ mã hóa dữ liệu video 200 (không được thể hiện trên hình vẽ) có thể lưu trữ các lệnh như vậy.

Bộ nhớ dữ liệu video 230 được tạo cấu hình để lưu trữ dữ liệu video nhận được. Bộ mã hóa dữ liệu video 200 có thể phục hồi hình ảnh của dữ liệu video từ bộ nhớ dữ liệu video 230 và cung cấp dữ liệu video cho đơn vị tạo dư 204 và đơn vị chọn chế độ 202. Dữ liệu video trong bộ nhớ dữ liệu video 230 có thể là dữ liệu video thô cần được mã hóa.

Đơn vị chọn chế độ 202 bao gồm đơn vị ước lượng chuyển động 222, đơn vị bù chuyển động 224 và đơn vị dự đoán nội ảnh 226. Đơn vị chọn chế độ 202 có thể bao gồm các đơn vị có chức năng bổ sung để thực hiện dự đoán video theo các chế độ dự đoán khác. Ví dụ, đơn vị lựa chọn chế độ 202 có thể bao gồm đơn vị bảng màu, đơn vị sao chép nội khối (có thể là một phần của đơn vị ước lượng chuyển động 222 và/hoặc đơn vị bù chuyển động 224), đơn vị afin, đơn vị mô hình tuyến tính (linear model - LM), hoặc tương tự.

Đơn vị lựa chọn chế độ 202 thường phối hợp nhiều lần mã hóa để kiểm tra sự kết hợp của các thông số mã hóa và các trị số biến dạng tốc độ thu được cho các kết hợp như vậy. Các thông số mã hóa có thể bao gồm chia các CTU thành các CU, các chế độ dự đoán cho các CU, các kiểu biến đổi cho dữ liệu dư của các CU, các thông số lượng tử hóa cho dữ liệu dư của các CU, v.v.. Đơn vị lựa chọn chế độ 202 có thể cuối cùng chọn tổ hợp của các thông số mã hóa có trị số biến dạng tốc độ tốt hơn các tổ hợp đã thử nghiệm khác.

Bộ mã hóa dữ liệu video 200 có thể phân chia hình ảnh được phục hồi từ bộ nhớ dữ liệu video 230 thành một loạt các CTU, và đóng gói một hoặc nhiều CTU trong lát. Đơn vị lựa chọn chế độ 202 có thể chia CTU của hình ảnh theo cấu trúc cây, chẳng hạn như cấu trúc QTBT hoặc cấu trúc cây tứ phân của HEVC được mô tả ở trên. Như mô tả ở trên, bộ mã hóa dữ liệu video 200 có thể tạo ra một hoặc nhiều CU từ quy trình chia CTU theo cấu trúc cây. CU như vậy cũng có thể được gọi chung là “khối dữ liệu video” hoặc “khối”.

Nói chung, đơn vị lựa chọn chế độ 202 cũng điều khiển các thành phần của chúng (ví dụ, đơn vị ước lượng chuyển động 222, đơn vị bù chuyển động 224, và đơn vị dự đoán nội ảnh 226) để tạo ra khối dự đoán cho khối hiện thời (ví dụ, CU hiện thời, hoặc trong HEVC, phần chồng của PU và TU). Để dự đoán liên ảnh khối hiện thời, đơn vị ước lượng chuyển động 222 có thể thực hiện tìm kiếm chuyển động để xác định một hoặc nhiều khối tham chiếu phù hợp trong một hoặc nhiều hình tham chiếu (ví dụ, một hoặc nhiều hình ảnh

được lập mã trước đó được lưu trữ trong DPB 218). Cụ thể, đơn vị ước lượng chuyển động 222 có thể tính toán trị số thể hiện khối tham chiếu có thể có là tương tự như thể nào với khối hiện thời, ví dụ, theo tổng chênh lệch tuyệt đối (sum of absolute difference - SAD), tổng chênh lệch bình phương (sum of squared differences - SSD), chênh lệch tuyệt đối trung bình (mean absolute difference - MAD), chênh lệch bình phương trung bình (mean squared difference - MSD), hoặc tương tự. Đơn vị ước lượng chuyển động 222 nói chung có thể thực hiện các tính toán này bằng cách sử dụng các trị số chênh lệch từng mẫu giữa khối hiện thời và khối tham chiếu đang được xem xét. Đơn vị ước lượng chuyển động 222 có thể xác định khối tham chiếu có trị số thấp nhất thu được từ các phép tính này, biểu thị khối tham chiếu phù hợp nhất với khối hiện thời.

Đơn vị ước lượng chuyển động 222 có thể tạo ra một hoặc nhiều vector chuyển động (MV) xác định các vị trí của các khối tham chiếu trong hình tham chiếu so với vị trí của khối hiện thời trong hình ảnh hiện thời. Sau đó, đơn vị ước lượng chuyển động 222 có thể cung cấp các vector chuyển động cho đơn vị bù chuyển động 224. Ví dụ, đối với dự đoán liên ảnh một chiều, đơn vị ước lượng chuyển động 222 có thể cung cấp một vector chuyển động, trong khi đối với dự đoán liên ảnh hai chiều, đơn vị ước lượng chuyển động 222 có thể cung cấp hai vector chuyển động. Sau đó, đơn vị bù chuyển động 224 có thể tạo ra khối dự đoán bằng cách sử dụng các vector chuyển động. Ví dụ, đơn vị bù chuyển động 224 có thể phục hồi dữ liệu của khối tham chiếu bằng cách sử dụng vector chuyển động. Một ví dụ khác, nếu vector chuyển động có độ chính xác mẫu phân số, thì đơn vị bù chuyển động 224 có thể nội suy các trị số cho khối dự đoán theo một hoặc nhiều bộ lọc nội suy. Hơn nữa, để dự đoán liên ảnh hai chiều, đơn vị bù chuyển động 224 có thể phục hồi dữ liệu cho hai khối tham chiếu được xác định bởi các vector chuyển động tương ứng và kết hợp dữ liệu đã phục hồi, ví dụ, qua phép lấy trung bình từng mẫu hoặc lấy trung bình có trọng số.

Một ví dụ khác, để dự đoán nội ảnh hoặc lập mã dự đoán nội ảnh, đơn vị dự đoán nội ảnh 226 có thể tạo ra khối dự đoán từ các mẫu lân cận với khối hiện thời. Ví dụ, đối với các chế độ định hướng, đơn vị dự đoán nội ảnh 226 có thể thường kết hợp toán học các trị số của các mẫu lân cận và điền các trị số được tính toán này theo hướng xác định trên khối hiện thời để tạo ra khối dự đoán. Một ví dụ khác, đối với chế độ DC, đơn vị dự đoán nội ảnh 226 có thể tính trị số trung bình của các mẫu lân cận với khối hiện thời và

tạo ra khối dự đoán để bao gồm trị số trung bình thu được này cho mỗi mẫu của khối dự đoán.

Đơn vị chọn chế độ 202 cung cấp khối dự đoán cho đơn vị tạo ra khối dự 204. Đơn vị tạo ra khối dự 204 nhận phiên bản thô, chưa lập mã của khối hiện thời từ bộ nhớ dữ liệu video 230 và khối dự đoán từ đơn vị chọn chế độ 202. Đơn vị tạo ra khối dự 204 tính toán các trị số chênh lệch theo từng mẫu giữa khối hiện thời và khối dự đoán. Trị số chênh lệch theo từng mẫu thu được xác định khối dự cho khối hiện thời. Theo một số ví dụ, đơn vị tạo dự 204 cũng có thể xác định trị số chênh lệch giữa các trị số mẫu trong khối dự để tạo ra khối dự bằng cách sử dụng kỹ thuật điều chế mã xung vi sai dự (residual differential pulse code modulation - RDPCM). Theo một số ví dụ, đơn vị tạo dự 204 có thể được tạo thành bằng cách sử dụng một hoặc nhiều mạch trừ thực hiện phép trừ nhị phân.

Trong các ví dụ mà đơn vị chọn chế độ 202 chia các CU thành các PU, mỗi PU có thể được kết hợp với một đơn vị dự đoán độ chói và các đơn vị dự đoán sắc độ tương ứng. Bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể hỗ trợ các PU có các kích thước khác nhau. Như nêu trên đây, kích thước của CU có thể được dùng để chỉ kích thước khối lập mã độ chói của CU và kích thước của PU có thể được dùng để chỉ kích thước của đơn vị dự đoán độ chói của PU. Giả sử kích thước của một CU cụ thể là $2N \times 2N$, bộ mã hóa dữ liệu video 200 có thể hỗ trợ các kích thước PU $2N \times 2N$ hoặc $N \times N$ đối với dự đoán nội ảnh, và các kích thước PU đối xứng $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$, hoặc tương tự để dự đoán liên ảnh. Bộ mã hóa dữ liệu video 20 và bộ giải mã dữ liệu video 30 cũng có thể hỗ trợ phân chia bất đối xứng các kích thước PU $2N \times nU$, $2N \times nD$, $nL \times 2N$, và $nR \times 2N$ để dự đoán liên ảnh.

Trong các ví dụ mà đơn vị chọn chế độ 202 không chia thêm CU thành các PU, mỗi CU có thể được kết hợp với một khối lập mã độ chói và các khối lập mã sắc độ tương ứng. Như nêu trên đây, kích thước của CU có thể là kích thước của khối lập mã độ chói của CU. Bộ mã hóa dữ liệu video 200 và bộ giải mã dữ liệu video 300 có thể hỗ trợ các kích thước CU $2N \times 2N$, $2N \times N$ hoặc $N \times 2N$.

Đối với các kỹ thuật mã hóa dữ liệu video khác, như mã hóa chế độ sao chép khối nội ảnh, mã hóa chế độ afin, và mã hóa chế độ mô hình tuyến tính (LM), chẳng hạn, đơn vị chọn chế độ 202, qua các đơn vị tương ứng liên quan đến các kỹ thuật mã hóa, tạo ra khối dự đoán cho khối hiện thời đang mã hóa. Theo một số ví dụ, như kỹ thuật lập mã chế

độ bảng màu, đơn vị lựa chọn chế độ 202 có thể không tạo ra khối dự đoán, và thay vào đó tạo ra các phần tử cú pháp biểu thị cách thức tái tạo khối dựa trên bảng màu đã chọn. Trong các chế độ như vậy, đơn vị chọn chế độ 202 có thể cung cấp các phần tử cú pháp này cho đơn vị mã hóa entropy 220 cần được mã hóa.

Như mô tả trên đây, đơn vị tạo dư 204 nhận dữ liệu video cho khối hiện thời và khối dự đoán tương ứng. Đơn vị tạo dư 204 sau đó tạo ra khối dư cho khối hiện thời. Để tạo ra khối dư, đơn vị tạo khối dư 204 tính toán các trị số chênh lệch theo từng mẫu giữa khối hiện thời và khối dự đoán.

Đơn vị xử lý biến đổi 206 áp dụng một hoặc nhiều phép biến đổi cho khối dư để tạo ra khối các hệ số biến đổi (ở đây được gọi là "khối hệ số biến đổi"). Đơn vị xử lý biến đổi 206 có thể áp dụng các phép biến đổi khác nhau cho khối dư để tạo ra khối hệ số biến đổi. Ví dụ, đơn vị xử lý biến đổi 206 có thể áp dụng phép biến đổi cosin rời rạc (discrete cosine transform - DCT), biến đổi có hướng, biến đổi Karhunen-Loeve (Karhunen-Loeve transform - KLT) hoặc biến đổi tương tự về mặt khái niệm cho khối dư. Theo một số ví dụ, đơn vị xử lý biến đổi 206 có thể thực hiện nhiều phép biến đổi cho khối dư, ví dụ, biến đổi chính và biến đổi phụ, như biến đổi quay chẳng hạn. Theo một số ví dụ, đơn vị xử lý biến đổi 206 không áp dụng các phép biến đổi cho khối biến đổi.

Đơn vị lượng tử hóa 208 có thể lượng tử hóa các hệ số biến đổi trong khối hệ số biến đổi, để tạo ra khối hệ số biến đổi lượng tử hóa. Đơn vị lượng tử hóa 208 có thể lượng tử hóa các hệ số biến đổi của khối hệ số biến đổi theo trị số thông số lượng tử hóa (quantization parameter - QP) kết hợp với khối hiện thời. Bộ mã hóa dữ liệu video 200 (ví dụ, qua đơn vị chọn chế độ 202) có thể điều chỉnh mức lượng tử hóa áp dụng cho các khối hệ số biến đổi kết hợp với khối hiện thời bằng cách điều chỉnh trị số QP kết hợp với CU. Quy trình lượng tử hóa có thể làm mất thông tin, và do đó, các hệ số biến đổi lượng tử hóa có thể có độ chính xác thấp hơn các hệ số biến đổi gốc được tạo ra bởi đơn vị xử lý biến đổi 206.

Đơn vị lượng tử hóa ngược 210 và đơn vị xử lý biến đổi ngược 212 có thể áp dụng lần lượt lượng tử hóa ngược và biến đổi ngược cho khối hệ số biến đổi được lượng tử hóa để tái tạo khối dư từ khối hệ số biến đổi. Đơn vị tái tạo 214 có thể tạo ra một khối tái tạo tương ứng với khối hiện thời (mặc dù có khả năng bị biến dạng ở mức độ nào đó) dựa trên khối dư được tái tạo và khối dự đoán được tạo ra bởi đơn vị lựa chọn chế độ 202. Ví dụ,

đơn vị tái tạo 214 có thể thêm các mẫu của khối dư tái tạo vào các mẫu tương ứng từ khối dư đoán được tạo ra bởi đơn vị chọn chế độ 202 để tạo ra khối được tái tạo.

Đơn vị lọc 216 có thể thực hiện một hoặc nhiều hoạt động lọc trên các khối tái tạo. Ví dụ, đơn vị lọc 216 có thể thực hiện các hoạt động tách khối để giảm các xảo ảnh dạng khối dọc theo các cạnh của CU. Ví dụ, đơn vị lọc 216 có thể được tạo cấu hình để so sánh lượng sửa đổi với mẫu do việc lọc tách khối gây ra cho trị số cắt, và để đáp lại lượng sửa đổi cho mẫu do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số bởi trị số cắt thay vì lượng sửa đổi cho mẫu do việc lọc tách khối gây ra. Tức là, đơn vị lọc 216 có thể được tạo cấu hình để giới hạn độ lớn của lượng sửa đổi cho mẫu. Theo một số ví dụ, các hoạt động của đơn vị lọc 216 có thể được bỏ qua.

Bộ mã hóa dữ liệu video 200 lưu trữ các khối tái tạo vào DPB 218. Ví dụ, trong các ví dụ mà không thực hiện hoạt động của đơn vị lọc 216, đơn vị tái tạo 214 có thể lưu trữ các khối tái tạo vào DPB 218. Trong các ví dụ mà thực hiện các hoạt động của đơn vị lọc 216, đơn vị tái tạo 216 có thể lưu trữ các khối tái tạo đã lọc vào DPB 218. Đơn vị ước lượng chuyển động 222 và đơn vị bù chuyển động 224 có thể phục hồi hình tham chiếu từ DPB 218, được tạo ra từ các khối tái tạo (và có thể được lọc), để dự đoán liên ảnh các khối của các hình ảnh được mã hóa sau đó. Ngoài ra, đơn vị dự đoán nội ảnh 226 có thể sử dụng các khối tái tạo trong DPB 218 của hình ảnh hiện thời để dự đoán nội ảnh các khối khác trong hình ảnh hiện thời.

Nói chung, đơn vị mã hóa entropy 220 có thể mã hóa entropy các phần tử cú pháp thu được từ các thành phần chức năng khác của bộ mã hóa dữ liệu video 200. Ví dụ, đơn vị mã hóa entropy 220 có thể mã hóa entropy các khối hệ số biến đổi lượng tử hóa từ đơn vị lượng tử hóa 208. Theo một ví dụ khác, đơn vị mã hóa entropy 220 có thể mã hóa entropy các phần tử cú pháp dự đoán (ví dụ, thông tin chuyển động để dự đoán liên ảnh hoặc thông tin chế độ nội ảnh để dự đoán nội ảnh) từ đơn vị lựa chọn chế độ 202. Đơn vị mã hóa entropy 220 có thể thực hiện một hoặc nhiều hoạt động mã hóa entropy trên các phần tử cú pháp, mà là một ví dụ khác về dữ liệu video, để tạo ra dữ liệu được mã hóa entropy. Ví dụ, đơn vị mã hóa entropy 220 có thể thực hiện hoạt động mã hóa có độ dài thay đổi thích ứng theo ngữ cảnh (context-adaptive variable length coding - CAVLC), hoạt động CABAC, hoạt động mã hóa có độ dài biến đổi đến biến đổi (variable-to-variable - V2V), hoạt động mã hóa số học nhị phân thích ứng theo ngữ cảnh dựa trên cú pháp (syntax-

based context-adaptive binary arithmetic coding - SBAC), hoạt động mã hóa entropy phân chia khoảng xác suất (Probability Interval Partitioning Entropy - PIPE), hoạt động mã hóa hàm mũ- Golomb, hoặc một kiểu hoạt động mã hóa entropy khác trên dữ liệu. Theo một số ví dụ, đơn vị mã hóa entropy 220 có thể hoạt động ở chế độ bỏ qua trong đó các phần tử cú pháp không được mã hóa entropy.

Bộ mã hóa dữ liệu video 200 có thể xuất ra luồng bit bao gồm các phần tử cú pháp mã hóa entropy cần thiết để tái tạo các khối của lát hoặc hình ảnh. Cụ thể, đơn vị mã hóa entropy 220 có thể xuất ra luồng bit.

Các hoạt động mô tả ở trên được mô tả liên quan đến khối. Sự mô tả như vậy nên được hiểu là các hoạt động đối với khối lập mã độ chói và/hoặc các khối lập mã sắc độ. Như được mô tả trên đây, trong một số ví dụ, khối lập mã độ chói và các khối lập mã sắc độ là các thành phần độ chói và sắc độ của CU. Trong một số ví dụ, khối lập mã độ chói và các khối lập mã sắc độ là các thành phần độ chói và sắc độ của PU.

Trong một số ví dụ, các hoạt động được thực hiện đối với khối lập mã độ chói không cần phải lặp lại đối với các khối lập mã sắc độ. Ví dụ, các hoạt động để xác định vector chuyển động (MV) và hình tham chiếu cho khối lập mã độ chói không cần phải lặp lại để xác định MV và hình tham chiếu cho các khối sắc độ. Thay vào đó, MV cho khối lập mã độ chói có thể được định tỷ lệ để xác định MV cho các khối sắc độ, và hình tham chiếu có thể giống nhau. Một ví dụ khác, quy trình dự đoán nội ảnh có thể giống nhau đối với các khối lập mã độ chói và các khối lập mã sắc độ.

Fig.7 là sơ đồ khối minh họa ví dụ về bộ giải mã dữ liệu video 300 mà có thể thực hiện các kỹ thuật của sáng chế. Fig.7 được cung cấp cho mục đích giải thích và không hạn chế trên các kỹ thuật như được minh họa và mô tả rộng rãi trong sáng chế này. Với mục đích giải thích, sáng chế này mô tả bộ giải mã dữ liệu video 300 được mô tả theo các kỹ thuật JEM và HEVC. Tuy nhiên, các kỹ thuật của sáng chế này có thể được thực hiện bởi các thiết bị lập mã dữ liệu video được tạo cấu hình theo các tiêu chuẩn lập mã dữ liệu video khác.

Theo ví dụ trên Fig.7, bộ giải mã dữ liệu video 300 bao gồm bộ nhớ đệm hình lập mã (coded picture buffer - CPB) 320, đơn vị giải mã entropy 302, đơn vị xử lý dự đoán 304, đơn vị lượng tử hóa ngược 306, đơn vị xử lý biến đổi ngược 308, đơn vị tái tạo 310,

đơn vị lọc 312 và bộ nhớ đệm hình đã giải mã (DPB) 314. Đơn vị xử lý dự đoán 304 bao gồm đơn vị bù chuyển động 316 và đơn vị dự đoán nội ảnh 318. Đơn vị xử lý dự đoán 304 có thể bao gồm các đơn vị bổ sung để thực hiện dự đoán theo các chế độ dự đoán khác. Ví dụ, đơn vị xử lý dự đoán 304 có thể bao gồm đơn vị bảng màu, đơn vị sao chép khối nội ảnh (có thể tạo thành một phần của đơn vị bù chuyển động 316), đơn vị afin, đơn vị mô hình tuyến tính (linear model - LM), hoặc tương tự. Trong các ví dụ khác, bộ giải mã dữ liệu video 300 có thể bao gồm nhiều hơn, ít hơn thành phần chức năng hoặc thành phần chức năng khác nhau.

Bộ nhớ CPB 320 có thể lưu trữ dữ liệu video, ví dụ như luồng bit dữ liệu video được mã hóa, cần được giải mã bởi các thành phần của bộ giải mã dữ liệu video 300. Dữ liệu video lưu trữ trong bộ nhớ CPB 320 có thể thu được, ví dụ, từ phương tiện đọc được bằng máy tính 110 (Fig.1). Bộ nhớ CPB 320 có thể bao gồm bộ đệm CPB lưu trữ dữ liệu video được mã hóa (ví dụ, các phần tử cú pháp) từ luồng bit video mã hóa. Ngoài ra, bộ nhớ CPB 320 có thể lưu trữ dữ liệu video ngoài các phần tử cú pháp của hình ảnh lập mã, chẳng hạn như dữ liệu tạm thời thể hiện các đầu ra từ các đơn vị khác nhau của bộ giải mã dữ liệu video 300. DPB 314 thường lưu trữ các hình ảnh đã giải mã mà bộ giải mã dữ liệu video 300 có thể xuất ra và/hoặc sử dụng làm dữ liệu video tham chiếu khi giải mã dữ liệu hoặc các hình ảnh tiếp theo của luồng bit video mã hóa. Bộ nhớ CPB 320 và DPB 314 có thể được tạo ra bởi thiết bị bất kỳ trong số nhiều thiết bị nhớ, như DRAM, SDRAM, MRAM, RRAM, hoặc các loại thiết bị nhớ khác. Bộ nhớ CPB 320 và DPB 314 có thể được tạo ra bởi cùng một thiết bị nhớ hoặc các thiết bị nhớ riêng. Trong các ví dụ khác nhau, bộ nhớ CPB 320 có thể nằm trên chip cùng các bộ phận khác của bộ giải mã dữ liệu video 300, hoặc không nằm trên chip so với các bộ phận đó.

Ngoài ra hoặc theo cách khác, trong một số ví dụ, bộ giải mã dữ liệu video 300 có thể phục hồi dữ liệu video lập mã từ bộ nhớ 120 (Fig.1). Tức là, bộ nhớ 120 có thể lưu trữ dữ liệu như mô tả ở trên vào bộ nhớ CPB 320. Tương tự, bộ nhớ 120 có thể lưu trữ các lệnh cần được thực thi bởi bộ giải mã dữ liệu video 30, khi một số hoặc tất cả các chức năng của bộ giải mã dữ liệu video 30 được thực hiện trong phần mềm được thực hiện bằng mạch xử lý của bộ giải mã dữ liệu video 300.

Các đơn vị khác nhau được thể hiện trên Fig.7 được minh họa để hỗ trợ hiểu các hoạt động được thực hiện bởi bộ giải mã dữ liệu video 300. Các đơn vị có thể được thực

hiện dưới dạng các mạch chức năng cố định, các mạch lập trình được, hoặc kết hợp của chúng. Tương tự với Fig.6, các mạch chức năng cố định là các mạch có chức năng cụ thể, và được cài đặt trước trong các hoạt động mà có thể được thực hiện. Các mạch lập trình được là các mạch mà có thể được lập trình để thực hiện một số nhiệm vụ khác nhau, và cung cấp chức năng linh hoạt trong các hoạt động mà có thể được thực hiện. Ví dụ, các mạch lập trình được có thể thực thi phần mềm hoặc firmware khiến cho các mạch lập trình được hoạt động theo cách thức được xác định bởi các lệnh của phần mềm hoặc firmware. Các mạch chức năng cố định có thể thực thi các lệnh phần mềm (ví dụ, để thu các thông số hoặc xuất ra các thông số), nhưng các loại hoạt động mà các mạch chức năng cố định thực hiện nói chung là không thể biến đổi. Trong một số ví dụ, một hoặc nhiều trong số các đơn vị có thể là các khối mạch khác nhau (chức năng cố định hoặc có thể lập trình được), và trong một số ví dụ, một hoặc nhiều đơn vị có thể là các mạch được tích hợp.

Bộ giải mã dữ liệu video 300 có thể bao gồm các ALU, EFU, mạch kỹ thuật số, mạch tương tự và/hoặc các lõi lập trình được tạo ra từ mạch có thể lập trình. Trong các ví dụ mà các hoạt động của bộ giải mã dữ liệu video 300 được thực hiện bởi phần mềm thực thi trên các mạch lập trình được, bộ nhớ trên chip hoặc ngoài chip có thể lưu trữ các lệnh (ví dụ, mã đối tượng) của phần mềm mà bộ giải mã dữ liệu video 300 nhận và thực thi.

Đơn vị giải mã entropy 302 có thể nhận dữ liệu video được mã hóa từ bộ đệm CPB và giải mã entropy dữ liệu video để tái tạo các phân tử cú pháp. Đơn vị xử lý dự đoán 304, đơn vị lượng tử hóa ngược 306, đơn vị xử lý biến đổi ngược 308, đơn vị tái tạo 310, và đơn vị lọc 312 có thể tạo ra dữ liệu video đã giải mã dựa vào các phân tử cú pháp trích ra từ luồng bit.

Nói chung, bộ giải mã dữ liệu video 300 tái tạo hình ảnh dựa trên từng khối. Bộ giải mã dữ liệu video 300 có thể thực hiện hoạt động tái tạo trên từng khối riêng (trong đó khối hiện đang được tái tạo, tức là đã giải mã, có thể được gọi là “khối hiện thời”).

Đơn vị giải mã entropy 302 có thể giải mã entropy các phân tử cú pháp xác định hệ số biến đổi lượng tử hóa của khối hệ số biến đổi lượng tử hóa, cũng như thông tin biến đổi, như thông số lượng tử hóa (quantization parameter - QP) và/hoặc (các) chỉ báo chế độ biến đổi. Đơn vị lượng tử hóa ngược 306 có thể sử dụng QP liên kết với khối hệ số biến đổi lượng tử hóa để xác định mức lượng tử hóa và, tương tự, mức lượng tử hóa ngược cho đơn vị lượng tử hóa ngược 306 để áp dụng. Ví dụ, đơn vị lượng tử hóa ngược 306 có thể thực

hiện phép toán dịch trái theo bit để lượng tử hóa ngược các hệ số biến đổi lượng tử hóa. Do đó, đơn vị lượng tử hóa ngược 306 có thể tạo ra khối hệ số biến đổi bao gồm các hệ số biến đổi.

Sau khi đơn vị lượng tử hóa ngược 306 tạo ra khối hệ số biến đổi, đơn vị xử lý biến đổi ngược 308 có thể áp dụng một hoặc nhiều phép biến đổi ngược cho khối hệ số biến đổi để tạo ra khối dư liên quan đến khối hiện thời. Ví dụ, đơn vị xử lý biến đổi ngược 308 có thể áp dụng phép DCT ngược, biến đổi số nguyên ngược, biến đổi Karhunen-Loeve (Karhunen-Loeve transform - KLT) ngược, biến đổi quay ngược, biến đổi hướng ngược hoặc một biến đổi ngược khác cho khối hệ số.

Hơn nữa, đơn vị xử lý dự đoán 304 tạo ra khối dự đoán theo các phần tử cú pháp thông tin dự đoán đã được giải mã entropy bởi đơn vị giải mã entropy 302. Ví dụ, nếu các phần tử cú pháp thông tin dự đoán biểu thị rằng khối hiện thời được dự đoán liên ảnh, thì đơn vị bù chuyển động 316 có thể tạo ra khối dự đoán. Trong trường hợp này, các phần tử cú pháp thông tin dự đoán có thể biểu thị hình tham chiếu trong DPB 314 từ đó truy xuất khối tham chiếu, cũng như vector chuyển động xác định vị trí của khối tham chiếu trong hình tham chiếu so với vị trí của khối hiện thời trong hình ảnh hiện thời. Đơn vị bù chuyển động 316 có thể thường thực hiện quy trình dự đoán liên ảnh theo cách thức về cơ bản tương tự như quy trình được mô tả liên quan đến đơn vị bù chuyển động 224 (Fig. 6).

Theo một ví dụ khác, nếu các phần tử cú pháp thông tin dự đoán biểu thị rằng khối hiện thời được dự đoán nội ảnh, thì đơn vị dự đoán nội ảnh 318 có thể tạo ra khối dự đoán theo chế độ dự đoán nội ảnh được biểu thị bởi các phần tử cú pháp thông tin dự đoán. Một lần nữa, đơn vị dự đoán nội ảnh 318 có thể thường thực hiện quy trình dự đoán nội ảnh theo cách thức về cơ bản tương tự như quy trình được mô tả liên quan đến đơn vị dự đoán nội ảnh 226 (Fig.6). Đơn vị dự đoán nội ảnh 318 có thể phục hồi dữ liệu của các mẫu lân cận cho khối hiện thời từ DPB 314.

Đơn vị tái tạo 310 có thể tái tạo khối hiện thời bằng cách sử dụng khối dự đoán và khối dư. Ví dụ, đơn vị tái tạo 310 có thể thêm các mẫu của khối dư vào các mẫu tương ứng của khối dự đoán để tái tạo khối hiện thời.

Đơn vị lọc 312 có thể thực hiện một hoặc nhiều hoạt động lọc trên các khối tái tạo. Ví dụ, đơn vị lọc 312 có thể thực hiện các hoạt động tách khối để giảm các xảo ảnh dạng

khối dọc theo các cạnh của khối tái tạo. Ví dụ, đơn vị lọc 312 có thể được tạo cấu hình để so sánh lượng sửa đổi với mẫu do việc lọc tách khối gây ra cho trị số cắt, và để đáp lại lượng sửa đổi cho mẫu do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số bởi trị số cắt thay vì lượng sửa đổi cho mẫu do việc lọc tách khối gây ra. Tức là, đơn vị lọc 312 có thể được tạo cấu hình để giới hạn độ lớn của lượng sửa đổi cho mẫu. Các hoạt động của đơn vị lọc 312 không cần thiết được thực hiện trong tất cả các ví dụ.

Bộ giải mã dữ liệu video 300 có thể lưu trữ các khối tái tạo vào DPB 314. Ví dụ, trong các ví dụ mà không cần đến hoạt động của đơn vị lọc 312, đơn vị tái tạo 310 có thể lưu trữ các khối tái tạo vào DPB 314. Trong các ví dụ mà cần đến các hoạt động của đơn vị lọc 312, đơn vị tái tạo 312 có thể lưu trữ các khối tái tạo đã lọc vào DPB 314. Như đã mô tả ở trên, DPB 314 có thể cung cấp thông tin tham chiếu, chẳng hạn như các mẫu của hình ảnh hiện thời để dự đoán nội ảnh và các hình ảnh giải mã trước đó để bù chuyển động tiếp theo, cho đơn vị xử lý dự đoán 304. Hơn nữa, bộ giải mã dữ liệu video 300 có thể xuất ra hình ảnh giải mã (ví dụ, video được giải mã) từ DPB 314 để trình diễn sau trên thiết bị hiển thị, chẳng hạn như thiết bị hiển thị 118 trên Fig.1.

Fig.8 thể hiện phương án thực hiện ví dụ của đơn vị lọc 312 trên Fig.7. Đơn vị lọc 216 trên Fig.6 có thể được thực thi theo cách giống nhau hoặc tương tự. Các đơn vị lọc 216 và 312 có thể thực hiện các kỹ thuật theo sáng chế này, có thể kết hợp với các thành phần khác của bộ mã hóa dữ liệu video 200 hoặc bộ giải mã dữ liệu video 300. Theo ví dụ trên Fig.8, đơn vị lọc 312 bao gồm bộ lọc tách khối 342, bộ lọc SAO 344 và bộ lọc ALF/GALF 346. Ví dụ, bộ lọc SAO 344 có thể được tạo cấu hình để xác định trị số bù cho các mẫu của khối. ALF/GALF 346 có thể còn lọc các khối của dữ liệu video bằng cách sử dụng bộ lọc vòng thích ứng và/hoặc lọc vòng thích ứng hình học.

Đơn vị lọc 312 có thể bao gồm vài bộ lọc và/hoặc có thể bao gồm các bộ lọc bổ sung. Ngoài ra, các bộ lọc cụ thể được thể hiện trên Fig.8 có thể được thực hiện theo trình tự khác. Các bộ lọc vòng lặp khác (trong vòng lặp lặp mã hoặc sau vòng lặp lặp mã) cũng có thể được sử dụng để làm trơn các chuyển tiếp điểm ảnh hoặc theo cách khác cải thiện chất lượng video. Các khối video tái tạo được lọc xuất ra bởi đơn vị lọc 312 có thể được lưu trữ trong DPB 314, lưu trữ các hình tham chiếu sử dụng để bù chuyển động tiếp theo. DPB 314 có thể là một phần của hoặc tách rời với bộ nhớ bổ sung lưu trữ video đã giải mã để sau đó trình diễn trên thiết bị hiển thị, như thiết bị hiển thị 118 trên Fig.1.

Bộ giải mã dữ liệu video 300, ví dụ, bộ lọc tách khối 342 của đơn vị lọc 312, có thể được tạo cấu hình để thu được khối tái tạo thứ nhất của dữ liệu video và áp dụng lọc tách khối vào khối tái tạo thứ nhất theo bất kỳ kỹ thuật hoặc sự kết hợp của các kỹ thuật được mô tả theo sáng chế này. Để thu được khối tái tạo thứ nhất của dữ liệu video, bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định khối dự đoán dữ liệu video thêm khối dư của dữ liệu video vào khối dự đoán dữ liệu video. Để áp dụng việc lọc tách khối vào khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể áp dụng việc lọc tách khối vào các mẫu được đặt tại bờ của khối tái tạo thứ nhất của dữ liệu video và khối tái tạo thứ hai của dữ liệu video. Để áp dụng việc lọc tách khối vào khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 xác định cường độ lọc cho việc lọc tách khối.

Để áp dụng việc lọc tách khối vào khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể chọn bộ lọc cho việc lọc tách khối. Để lựa chọn bộ lọc cho việc lọc tách khối, bộ giải mã dữ liệu video 300 có thể chọn bộ lọc dựa vào độ dài của khối trực giao với biên cần được lọc. Để lựa chọn bộ lọc cho việc lọc tách khối, bộ giải mã dữ liệu video 300 lựa chọn bộ lọc dựa vào độ dài của khối đồng chỉnh với biên cần được lọc. Để lựa chọn bộ lọc cho việc lọc tách khối, bộ giải mã dữ liệu video 300 có thể chọn bộ lọc từ nhiều bộ lọc tách khối có các độ dài khác nhau. Để lựa chọn bộ lọc cho việc lọc tách khối, bộ giải mã dữ liệu video 300 lựa chọn bộ lọc từ nhiều bộ lọc tách khối có các hỗ trợ lọc khác nhau.

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định hệ số tin cậy và xác định lượng sửa đổi mẫu dựa vào hệ số tin cậy. Bộ giải mã dữ liệu video 300 có thể áp dụng việc lọc tách khối vào khối tái tạo thứ nhất bằng cách sửa đổi mẫu của khối tái tạo thứ nhất bởi lượng sửa đổi mẫu. Bộ giải mã dữ liệu video 300 có thể xác định lượng sửa đổi mẫu dựa vào hệ số tin cậy bằng cách giới hạn lượng sửa đổi mẫu cho khoảng trị số. Ví dụ, hệ số tin cậy có thể là một khoảng cách.

Bộ giải mã dữ liệu video 300 có thể được tạo cấu hình để xác định khối dự đoán thứ hai của dữ liệu video; thêm khối dư thứ hai của dữ liệu video vào khối dự đoán dữ liệu video thứ hai để xác định khối tái tạo thứ hai của dữ liệu video mà liền kề với khối tái tạo thứ nhất; xác định thông số lượng tử hóa thứ nhất cho khối dư; xác định thông số lượng tử hóa thứ hai cho khối dư thứ hai; áp dụng lọc tách khối vào khối tái tạo thứ nhất dựa vào thông số lượng tử hóa thứ nhất; và áp dụng lọc tách khối vào khối tái tạo thứ hai dựa vào thông số lượng tử hóa thứ hai.

Để áp dụng lọc tách khối vào khối tái tạo thứ nhất dựa vào thông số lượng tử hóa thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ nhất dựa vào thông số lượng tử hóa thứ nhất, và áp dụng lọc tách khối vào khối tái tạo thứ hai dựa vào thông số lượng tử hóa thứ hai bằng cách xác định lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ hai dựa vào thông số lượng tử hóa thứ hai. Để xác định lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ nhất dựa vào thông số lượng tử hóa thứ nhất, bộ giải mã dữ liệu video 300 có thể giới hạn lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ nhất ở khoảng trị số thứ nhất và xác định lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ hai dựa vào thông số lượng tử hóa thứ hai bằng cách giới hạn lượng sửa đổi mẫu cho mẫu của khối tái tạo thứ hai ở khoảng trị số thứ hai là khác với khoảng trị số thứ nhất.

Để áp dụng việc lọc tách khối vào khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể chọn các thông số lọc tách khối cho việc lọc tách khối. Ví dụ, các thông số lọc tách khối có thể bao gồm một hoặc nhiều kiểu lọc, độ dài lọc, các thông số giới hạn, hoặc bước bỏ qua mẫu theo không gian. Để lựa chọn các thông số lọc tách khối cho việc lọc tách khối, bộ giải mã dữ liệu video 300 có thể chọn các thông số lọc tách khối dựa vào một hoặc cả hai đặc tính biến đổi hoặc các hệ số biến đổi cho khối dư.

Để áp dụng việc lọc tách khối vào khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cho các mẫu không có sẵn và áp dụng việc lọc tách khối sử dụng các trị số được xác định. Để xác định các trị số cho các mẫu không có sẵn, bộ giải mã dữ liệu video 300 có thể thực hiện quy trình ngoại suy và/hoặc quy trình đệm. Bộ giải mã dữ liệu video 300 có thể xác định các mẫu không có sẵn sẽ là không có sẵn dựa vào đặc điểm khối của khối tái tạo thứ nhất.

Bộ giải mã dữ liệu video 300, ví dụ, bộ lọc tách khối 342 của đơn vị lọc 312, có thể được tạo cấu hình để thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất với trị số thứ nhất; áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối; xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất; so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra; để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số

lọc thứ nhất cho mẫu thứ nhất; và xuất ra khối lọc tách khối của dữ liệu video, trong đó trong khối lọc tách khối, mẫu thứ nhất có trị số lọc thứ nhất.

Trị số lọc thứ nhất có thể bằng trị số thứ nhất cộng với trị số cắt thứ nhất hoặc có thể bằng trị số thứ nhất trừ trị số cắt thứ nhất. Bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất cho mẫu thứ nhất còn dựa vào chế độ lọc tách khối cho khối tái tạo thứ nhất.

Khối tái tạo thứ nhất có thể bao gồm mẫu thứ hai trong trị số thứ hai, và bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất và để đáp lại lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là ít hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi lượng sửa đổi cho mẫu thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai. Theo ví dụ này, trong khối lọc tách khối, mẫu thứ hai có trị số lọc thứ hai.

Nếu khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất bằng khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, thì bộ giải mã dữ liệu video 300 có thể thiết lập trị số cắt thứ nhất bằng trị số cắt thứ hai cho mẫu thứ hai. Nếu khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất là khác nhau so với khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, thì bộ giải mã dữ liệu video 300 có thể thiết lập trị số cắt thứ nhất là khác biệt so với trị số cắt thứ hai.

Dựa vào kích thước của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 chỉ có thể xác định bộ lọc tách khối áp dụng vào các mẫu là sáu mẫu hoặc ít hơn đã loại bỏ từ biên của khối tái tạo thứ nhất. Dựa vào kích thước của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định bộ lọc tách khối áp dụng vào các mẫu là hai mẫu hoặc ít hơn đã loại bỏ từ biên của khối tái tạo thứ nhất.

trị số cắt thứ nhất có thể giảm xuống với khoảng cách từ biên của khối tái tạo thứ nhất. Ví dụ, để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 6. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là một mẫu đã loại bỏ

khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 5. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 4.

Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là ba mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 3. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là bốn mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 2. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là năm mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 1.

Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 6. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là một mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 4. Để đáp lại xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, bộ giải mã dữ liệu video 300 có thể xác định trị số cắt thứ nhất bằng 2.

Bộ giải mã dữ liệu video 300 có thể còn được tạo cấu hình để thu được khối tái tạo thứ hai của dữ liệu video, trong đó khối tái tạo thứ hai bao gồm mẫu thứ hai với trị số thứ hai, trong đó khối tái tạo thứ hai dùng chung biên với khối tái tạo thứ nhất và trong đó khối tái tạo thứ nhất và khối tái tạo thứ hai có kích thước khác nhau; áp dụng lọc tách khối vào khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai; xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ hai và dựa vào

kích thước của khối tái tạo thứ hai; để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ hai, sửa đổi trị số thứ nhất bởi trị số cắt thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và xuất ra khối lọc tách khối thứ hai của dữ liệu video, trong đó trong khối lọc tách khối thứ hai, mẫu thứ hai có trị số lọc thứ hai. Bộ giải mã dữ liệu video 300 có thể áp dụng lọc tách khối vào khối tái tạo thứ nhất để xác định khối lọc tách khối thứ nhất bao gồm lọc các mẫu trong khối tái tạo thứ nhất với bộ lọc thứ nhất có độ dài thứ nhất và áp dụng lọc tách khối vào khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai bao gồm lọc các mẫu trong khối tái tạo thứ hai với bộ lọc thứ hai có độ dài thứ hai là khác nhau so với độ dài thứ nhất.

Fig.9 là lưu đồ minh họa phương pháp ví dụ để mã hóa khối hiện thời. Khối hiện thời có thể bao gồm CU hiện thời. Mặc dù được mô tả liên quan đến bộ mã hóa dữ liệu video 200 (các Fig.1 và 6), cần hiểu rằng các thiết bị khác có thể được tạo cấu hình để thực hiện phương pháp giống với phương pháp trên Fig.9.

Trong ví dụ này, bộ mã hóa dữ liệu video 200 dự đoán ban đầu khối hiện thời (350). Ví dụ, bộ mã hóa dữ liệu video 200 có thể hình thành khối dự đoán cho khối hiện thời. Sau đó, bộ mã hóa dữ liệu video 200 có thể tính toán khối dư cho khối hiện thời (352). Để tính toán khối dư, bộ mã hóa dữ liệu video 200 có thể tính toán sự chênh lệch giữa khối gốc, khối chưa mã hóa và khối dự đoán cho khối hiện thời. Sau đó, bộ mã hóa dữ liệu video 200 có thể biến đổi và lượng tử hóa các hệ số của khối dư (354). Tiếp theo, bộ mã hóa dữ liệu video 200 có thể quét các hệ số biến đổi được lượng tử hóa của khối dư (356). Trong khi quét, hoặc sau khi quét, bộ mã hóa dữ liệu video 200 có thể mã hóa entropy các hệ số (358). Ví dụ, bộ mã hóa dữ liệu video 200 có thể mã hóa các hệ số bằng cách sử dụng CAVLC hoặc CABAC. Sau đó, bộ mã hóa dữ liệu video 200 có thể xuất ra dữ liệu được mã hóa entropy cho các hệ số của khối (360).

Fig.10 là lưu đồ minh họa phương pháp ví dụ để giải mã khối hiện thời của dữ liệu video. Khối hiện thời có thể bao gồm CU hiện thời. Mặc dù được mô tả liên quan đến bộ giải mã dữ liệu video 300 (các Fig.1 và 7), cần hiểu rằng các thiết bị khác có thể được tạo cấu hình để thực hiện phương pháp giống với phương pháp trên Fig.10.

Bộ giải mã dữ liệu video 300 có thể thu dữ liệu đã lập mã entropy cho khối hiện thời, như thông tin dự đoán đã lập mã entropy và dữ liệu đã lập mã entropy cho các hệ số của khối dư tương ứng với khối hiện thời (370). Bộ giải mã dữ liệu video 300 có thể giải

mã entropy dữ liệu đã lập mã entropy để xác định thông tin dự đoán cho khối hiện thời và để tái tạo các hệ số của khối dư (372). Bộ giải mã dữ liệu video 300 có thể dự đoán khối hiện thời (374), ví dụ, bằng cách sử dụng chế độ dự đoán nội ảnh hoặc liên ảnh như được chỉ báo bởi thông tin dự đoán cho khối hiện thời, để tính toán khối dự đoán cho khối hiện thời. Sau đó, bộ giải mã dữ liệu video 300 có thể quét ngược các hệ số được tái tạo (376), để tạo ra khối của các hệ số biến đổi đã lượng tử hóa. Sau đó, bộ giải mã dữ liệu video 300 có thể lượng tử hóa ngược và biến đổi ngược các hệ số để sản xuất ra khối dư (378). Sau cùng, bộ giải mã dữ liệu video 300 có thể giải mã khối hiện thời bằng cách kết hợp khối dự đoán và khối dư (380). Sau khi kết hợp các khối dự đoán và khối dư để tạo ra khối tái tạo, bộ giải mã dữ liệu video 300 có thể áp dụng một hoặc nhiều lọc (ví dụ, tách khối, SAO, và/hoặc ALF/GALF) cho khối tái tạo không được lọc để tạo ra khối tái tạo được lọc (382).

Fig.11 là lưu đồ minh họa kỹ thuật giải mã dữ liệu video làm ví dụ được mô tả theo sáng chế này. Các kỹ thuật trên Fig.11 sẽ mô tả dựa vào bộ giải mã dữ liệu video chung, nhưng không bị giới hạn ở bộ giải mã dữ liệu video 300. Theo một số ví dụ, các kỹ thuật trên Fig.11 có thể được thực hiện bởi bộ mã hóa dữ liệu video như bộ mã hóa dữ liệu video 200, là một phần của quy trình mã hóa dữ liệu video, trong đó trường hợp bộ giải mã dữ liệu video chung tương ứng với vòng giải mã (ví dụ, đơn vị lượng tử hóa ngược 210, đơn vị xử lý biến đổi ngược 212, đơn vị tái tạo 214, và đơn vị lọc 216) của bộ mã hóa dữ liệu video 200.

Theo ví dụ trên Fig.11, bộ giải mã dữ liệu video thu được khối tái tạo thứ nhất của dữ liệu video bao gồm mẫu thứ nhất với trị số thứ nhất (400). Bộ giải mã dữ liệu video áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối (402). Bộ giải mã dữ liệu video xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất (404). Bộ giải mã dữ liệu video 300 so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra (406). Để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất (408, CÓ), bộ giải mã dữ liệu video sửa đổi trị số thứ nhất bằng trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất (410). Để đáp lại lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là không lớn hơn trị số cắt thứ nhất (408, KHÔNG), bộ giải mã dữ liệu video sửa đổi trị số thứ nhất bởi

lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khỏi gây ra (412). Ví dụ, bộ giải mã dữ liệu video có thể giữ trị số lọc thứ nhất bằng trị số lọc được xác định bởi việc lọc giải khối thay vì thiết lập trị số lọc thứ nhất bằng trị số thứ nhất cộng hoặc trừ trị số cắt.

Phải thừa nhận rằng tùy thuộc vào ví dụ, một số hoạt động hoặc sự kiện của kỹ thuật bất kỳ trong số các kỹ thuật được mô tả trong sáng chế này có thể được thực hiện theo các trình tự khác nhau, có thể được thêm, hợp nhất, hoặc bỏ hoàn toàn (ví dụ, không phải tất cả các hoạt động hoặc sự kiện được mô tả là cần thiết để thực hiện các kỹ thuật). Ngoài ra, trong một số ví dụ, các hoạt động hoặc biến cố có thể được thực hiện đồng thời, ví dụ, thông qua quy trình xử lý đa khâu chuỗi, quy trình xử lý ngắt, hoặc nhiều bộ xử lý, chứ không phải chỉ có thực hiện tuần tự.

Trong một hoặc nhiều ví dụ, các chức năng đã mô tả có thể được thực hiện dưới dạng phần cứng, phần mềm, firmware, hoặc mọi dạng kết hợp của chúng. Nếu được thực hiện bằng phần mềm, các hàm 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 đơn vị xử lý dựa trên phần cứng. Phương tiện đọc được bằng máy tính có thể bao gồm các phương tiện lưu trữ đọc được bằng máy tính 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 có thể thường tương ứng với (1) phương tiện lưu trữ hữu hình đọc được bằng máy tính là phương tiện bất khả 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 chứa 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 một hoặc nhiều trong số bộ nhớ truy nhập ngẫu nhiên (Random Access Memory - RAM), bộ nhớ chỉ đọc (Read Only Memory ROM), bộ nhớ chỉ đọc lập trình được xóa được bằng điện (Electrically Erasable Programmable Read Only Memory - EEPROM), đĩa compac-bộ nhớ chỉ đọc (CD-ROM) hoặc các thiết bị lưu trữ đĩa quang khác, thiết bị lưu trữ đĩa từ, hoặc các thiết bị lưu trữ từ tính khác, bộ nhớ tác động nhanh,

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, kết nối bất kỳ được gọi theo cách thích hợp là phương tiện đọc được bằng máy tính. Ví dụ, nếu các lệnh được truyền từ trang web, máy chủ hoặc nguồn từ xa khác bằng cách sử dụng cáp đồng trục, cáp quang sợi, cáp dây xoắn, đường thuê bao kỹ thuật số (digital subscriber line - DSL) hoặc các công nghệ không dây như hồng ngoại, vô tuyến, và vi sóng, thì cáp đồng trục, cáp quang sợi, cáp dây xoắn, đường DSL hoặc các công nghệ không dây như hồng ngoại, vô tuyến và vi sóng được bao gồm trong định nghĩa phương tiện. Tuy nhiên, cần phải hiểu rằng phương tiện đọc được bằng máy tính và phương tiện lưu dữ liệu không bao gồm kết nối, sóng mang, tín hiệu hoặc phương tiện chuyển tiếp khác, nhưng thay vào đó được hướng đến phương tiện hữu hình, bất biến. Đĩa từ và đĩa quang, như được sử dụng ở đây, bao gồm đĩa compac (CD), đĩa laze, đĩa quang, đĩa số đa năng (DVD - Digital Versatile Disc), đĩa mềm và đĩa bluray, trong đó đĩa từ thường tái tạo dữ liệu bằng phương pháp từ tính, còn đĩa quang thì tái tạo dữ liệu bằng phương pháp quang học sử dụng laze. 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 thi hành bởi một hoặc nhiều bộ xử lý, chẳng hạn như một hoặc nhiều bộ xử lý tín hiệu số (Digital Signal bộ xử lý - DSP), các bộ vi xử lý đa năng, các mạch tích hợp chuyên dụng (Application Specific Integrated Circuits - ASIC), các mảng logic lập trình được bằng trường (field programmable logic arrays - FPGAs), hoặc mạch logic rời rạc hoặc mạch tích hợp tương đương khác. Do đó, thuật ngữ “bộ xử lý”, như được sử dụng trong sáng chế, có thể dùng để chỉ mọi cấu trúc nêu trên hoặc mọi cấu trúc khác phù hợp để thực hiện các kỹ thuật nêu trong sáng chế. Ngoài ra, theo một số khía cạnh, chức năng được mô tả ở đây có thể được tạo ra trong module 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 của sáng chế có thể được thực hiện trong nhiều loại cơ cấu hoặc thiết bị khác nhau, bao gồm máy cầm tay không dây, mạch tích hợp (integrated circuit - IC) hoặc bộ IC (ví dụ, bộ chip). Các thành phần, module hoặc đơn vị khác nhau được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình

để thực hiện các kỹ thuật đã mô tả, chứ không nhất thiết phải được thực hiện bằng các đơn vị phần cứng khác nhau. Thay vào đó, như được mô tả ở trên, một số đơn vị có thể được kết hợp trong đơn vị phần cứng codec hoặc được cung cấp bởi tập hợp các đơn vị phần cứng tương tác, bao gồm một hoặc nhiều bộ xử lý như được mô tả ở trên, kết hợp với phần mềm và/hoặc firmware thích 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 giải mã dữ liệu video, phương pháp này bao gồm các bước:

thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất với trị số thứ nhất;

áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối thứ nhất;

xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra;

để đáp lại việc lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và

xuất ra khối lọc tách khối thứ nhất của dữ liệu video, trong đó trong khối lọc tách khối thứ nhất, mẫu thứ nhất có trị số lọc thứ nhất,

phương pháp này còn bao gồm các bước:

thu được khối tái tạo thứ hai của dữ liệu video, trong đó khối tái tạo thứ hai bao gồm mẫu thứ hai với trị số thứ hai, trong đó khối tái tạo thứ hai có chung biên với khối tái tạo thứ nhất và trong đó khối tái tạo thứ nhất và khối tái tạo thứ hai có kích thước khác nhau;

áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai;

xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ hai và dựa vào kích thước của khối tái tạo thứ hai;

để đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi trị số cắt thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

xuất ra khối lọc tách khối thứ hai của dữ liệu video, trong đó trong khối lọc tách khối thứ hai, mẫu thứ hai có trị số lọc thứ hai,

trong đó:

bước áp dụng lọc tách khối cho khối tái tạo thứ nhất để xác định khối lọc tách khối thứ nhất bao gồm lọc các mẫu trong khối tái tạo thứ nhất với bộ lọc thứ nhất có độ dài thứ nhất; và

bước áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai bao gồm lọc các mẫu trong khối tái tạo thứ hai với bộ lọc thứ hai có độ dài thứ hai khác với độ dài thứ nhất.

2. Phương pháp theo điểm 1, trong đó trị số lọc thứ nhất bằng trị số thứ nhất cộng trị số cắt thứ nhất.

3. Phương pháp theo điểm 1, trong đó trị số lọc thứ nhất bằng trị số thứ nhất trừ đi trị số cắt thứ nhất.

4. Phương pháp theo điểm 1, phương pháp này còn bao gồm bước:

xác định trị số cắt thứ nhất cho mẫu thứ nhất còn dựa vào chế độ lọc tách khối cho khối tái tạo thứ nhất.

5. Phương pháp theo điểm 1, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ hai với trị số thứ hai, phương pháp này còn bao gồm các bước;

xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra ít hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi lượng sửa đổi cho mẫu thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

trong đó trong khối lọc tách khối thứ nhất, mẫu thứ hai có trị số lọc thứ hai.

6. Phương pháp theo điểm 5, trong đó khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất bằng khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, và trong đó trị số cắt thứ nhất bằng trị số cắt thứ hai cho mẫu thứ hai.

7. Phương pháp theo điểm 5, trong đó khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất khác với khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, và trong đó trị số cắt thứ nhất khác với trị số cắt thứ hai.

8. Phương pháp theo điểm 1, phương pháp này còn bao gồm bước:

dựa vào kích thước của khối tái tạo thứ nhất, xác định bộ lọc tách khối chỉ được áp dụng vào các mẫu là sáu mẫu hoặc ít hơn đã loại bỏ khỏi biên của khối tái tạo thứ nhất.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 6.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại việc mẫu thứ nhất là một mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 5.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 4.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là ba mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 3.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là bốn mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 2.

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

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là năm mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 1.

15. Phương pháp theo điểm 1, phương pháp này còn bao gồm bước:

dựa vào kích thước của khối tái tạo thứ nhất, xác định bộ lọc tách khối được áp dụng vào các mẫu là hai mẫu hoặc ít hơn đã loại bỏ khỏi biên của khối tái tạo thứ nhất.

16. Phương pháp theo điểm 15, phương pháp này còn bao gồm bước:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 6.

17. Phương pháp theo điểm 15, phương pháp này còn bao gồm bước:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là một mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 4.

18. Phương pháp theo điểm 15, phương pháp này còn bao gồm bước:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 2.

19. Phương pháp theo điểm 1, trong đó phương pháp được thực hiện là một phần của quy trình mã hóa dữ liệu video.

20. Thiết bị giải mã dữ liệu video, thiết bị này bao gồm:

bộ nhớ được tạo cấu hình để lưu trữ dữ liệu video;

một hoặc nhiều bộ xử lý lắp đặt trong hệ mạch và được tạo cấu hình để:

thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất với trị số thứ nhất;

áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối thứ nhất;

xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra;

để đáp lại việc lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra

là lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và

xuất ra khối lọc tách khối thứ nhất của dữ liệu video, trong đó trong khối lọc tách khối thứ nhất, mẫu thứ nhất có trị số lọc thứ nhất,

trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

thu được khối tái tạo thứ hai của dữ liệu video, trong đó khối tái tạo thứ hai bao gồm mẫu thứ hai với trị số thứ hai, trong đó khối tái tạo thứ hai có chung biên với khối tái tạo thứ nhất và trong đó khối tái tạo thứ nhất và khối tái tạo thứ hai có kích thước khác nhau;

áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai;

xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ hai và dựa vào kích thước của khối tái tạo thứ hai;

để đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi trị số cắt thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

xuất ra khối lọc tách khối thứ hai của dữ liệu video, trong đó trong khối lọc tách khối thứ hai, mẫu thứ hai có trị số lọc thứ hai.

trong đó:

để áp dụng lọc tách khối cho khối tái tạo thứ nhất để xác định khối lọc tách khối thứ nhất, một hoặc nhiều bộ xử lý còn được tạo cấu hình để lọc các mẫu trong khối tái tạo thứ nhất với bộ lọc thứ nhất có độ dài thứ nhất; và

để áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai, một hoặc nhiều bộ xử lý còn được tạo cấu hình để lọc các mẫu trong khối tái tạo thứ hai với bộ lọc thứ hai có độ dài thứ hai khác với độ dài thứ nhất.

21. Thiết bị theo điểm 20, trong đó trị số lọc thứ nhất bằng trị số thứ nhất cộng trị số cắt thứ nhất.

22. Thiết bị theo điểm 20, trong đó trị số lọc thứ nhất bằng trị số thứ nhất trừ trị số cắt thứ nhất.

23. Thiết bị theo điểm 20, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

xác định trị số cắt thứ nhất cho mẫu thứ nhất còn dựa vào chế độ lọc tách khối cho khối tái tạo thứ nhất.

24. Thiết bị theo điểm 20, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ hai với trị số thứ hai, và trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để;

xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

để đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là ít hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi lượng sửa đổi cho mẫu thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

trong đó trong khối lọc tách khối thứ nhất, mẫu thứ hai có trị số lọc thứ hai.

25. Thiết bị theo điểm 24, trong đó khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất bằng khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, và trong đó trị số cắt thứ nhất bằng trị số cắt thứ hai cho mẫu thứ hai.

26. Thiết bị theo điểm 24, trong đó khoảng cách giữa mẫu thứ nhất và biên của khối tái tạo thứ nhất khác với khoảng cách giữa mẫu thứ hai và biên của khối tái tạo thứ nhất, và trong đó trị số cắt thứ nhất khác với trị số cắt thứ hai.

27. Thiết bị theo điểm 20, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

dựa vào kích thước của khối tái tạo thứ nhất, xác định bộ lọc tách khối chỉ được áp dụng vào các mẫu là sáu mẫu hoặc ít hơn đã loại bỏ khỏi biên của khối tái tạo thứ nhất.

28. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 6.

29. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là một mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 5.

30. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 4.

31. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là ba mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 3.

32. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là bốn mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 2.

33. Thiết bị theo điểm 27, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 6 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là năm mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 1.

34. Thiết bị theo điểm 20, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

dựa vào kích thước của khối tái tạo thứ nhất, xác định bộ lọc tách khối được áp dụng vào các mẫu là hai mẫu hoặc ít hơn đã loại bỏ khỏi biên của khối tái tạo thứ nhất.

35. Thiết bị theo điểm 34, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là liền kề với biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 6.

36. Thiết bị theo điểm 34, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là một mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 4.

37. Thiết bị theo điểm 34, trong đó một hoặc nhiều bộ xử lý còn được tạo cấu hình để:

để đáp lại việc xác định bộ lọc tách khối được áp dụng vào các mẫu trong 2 hàng hoặc cột của biên của khối tái tạo thứ nhất và để đáp lại mẫu thứ nhất là hai mẫu đã loại bỏ khỏi biên của khối tái tạo thứ nhất, xác định trị số cắt thứ nhất bằng 2.

38. Thiết bị theo điểm 20, trong đó thiết bị bao gồm bộ mã hóa dữ liệu video được tạo cấu hình để giải mã dữ liệu video là một phần của quy trình mã hóa dữ liệu video.

39. Phương tiện lưu trữ đọc được bằng máy tính lưu trữ các lệnh mà khi được thực thi bởi một hoặc nhiều bộ xử lý khiến cho một hoặc nhiều bộ xử lý:

thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất với trị số thứ nhất;

áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối thứ nhất;

xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra;

để đáp lại việc lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất, sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất; và

xuất ra khối lọc tách khối thứ nhất của dữ liệu video, trong đó trong khối lọc tách khối thứ nhất, mẫu thứ nhất có trị số lọc thứ nhất,

các lệnh, khi được thực thi bởi một hoặc nhiều bộ xử lý, còn khiến cho một hoặc nhiều bộ xử lý:

thu được khối tái tạo thứ hai của dữ liệu video, trong đó khối tái tạo thứ hai bao gồm mẫu thứ hai với trị số thứ hai, trong đó khối tái tạo thứ hai có chung biên với khối tái tạo thứ nhất và trong đó khối tái tạo thứ nhất và khối tái tạo thứ hai có kích thước khác nhau;

áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai;

xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ hai và dựa vào kích thước của khối tái tạo thứ hai;

để đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi trị số cắt thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

xuất ra khối lọc tách khối thứ hai của dữ liệu video, trong đó trong khối lọc tách khối thứ hai, mẫu thứ hai có trị số lọc thứ hai.

trong đó:

việc áp dụng lọc tách khối cho khối tái tạo thứ nhất để xác định khối lọc tách khối thứ nhất bao gồm lọc các mẫu trong khối tái tạo thứ nhất với bộ lọc thứ nhất có độ dài thứ nhất; và

việc áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai bao gồm lọc các mẫu trong khối tái tạo thứ hai với bộ lọc thứ hai có độ dài thứ hai khác với độ dài thứ nhất.

40. Máy giải mã dữ liệu video, máy này bao gồm:

phương tiện để thu được khối tái tạo thứ nhất của dữ liệu video, trong đó khối tái tạo thứ nhất bao gồm mẫu thứ nhất với trị số thứ nhất;

phương tiện để áp dụng lọc tách khối cho khối tái tạo thứ nhất để tạo ra khối lọc tách khối thứ nhất;

phương tiện để xác định trị số cắt thứ nhất cho mẫu thứ nhất dựa vào vị trí của mẫu thứ nhất so với biên của khối tái tạo thứ nhất và dựa vào kích thước của khối tái tạo thứ nhất;

phương tiện để so sánh trị số cắt thứ nhất với lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra;

phương tiện để sửa đổi trị số thứ nhất bởi trị số cắt thứ nhất để xác định trị số lọc thứ nhất cho mẫu thứ nhất để đáp lại việc lượng sửa đổi cho mẫu thứ nhất do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ nhất; và

phương tiện để xuất ra khối lọc tách khối thứ nhất của dữ liệu video, trong đó trong khối lọc tách khối thứ nhất, mẫu thứ nhất có trị số lọc thứ nhất,

máy còn bao gồm:

phương tiện để thu được khối tái tạo thứ hai của dữ liệu video, trong đó khối tái tạo thứ hai bao gồm mẫu thứ hai với trị số thứ hai, trong đó khối tái tạo thứ hai có chung biên với khối tái tạo thứ nhất và trong đó khối tái tạo thứ nhất và khối tái tạo thứ hai có kích thước khác nhau;

phương tiện để áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai;

phương tiện để xác định trị số cắt thứ hai cho mẫu thứ hai dựa vào vị trí của mẫu thứ hai so với biên của khối tái tạo thứ hai và dựa vào kích thước của khối tái tạo thứ hai;

phương tiện để, đáp lại việc lượng sửa đổi cho mẫu thứ hai do việc lọc tách khối gây ra là lớn hơn trị số cắt thứ hai, sửa đổi trị số thứ hai bởi trị số cắt thứ hai để xác định trị số lọc thứ hai cho mẫu thứ hai; và

phương tiện để xuất ra khối lọc tách khối thứ hai của dữ liệu video, trong đó trong khối lọc tách khối thứ hai, mẫu thứ hai có trị số lọc thứ hai,

trong đó:

phương tiện để áp dụng lọc tách khối cho khối tái tạo thứ nhất để xác định khối lọc tách khối thứ nhất bao gồm phương tiện để lọc các mẫu trong khối tái tạo thứ nhất với bộ lọc thứ nhất có độ dài thứ nhất; và

phương tiện để áp dụng lọc tách khối cho khối tái tạo thứ hai để xác định khối lọc tách khối thứ hai bao gồm phương tiện để lọc các mẫu trong khối tái tạo thứ hai với bộ lọc thứ hai có độ dài thứ hai khác với độ dài thứ nhất.

1 / 11

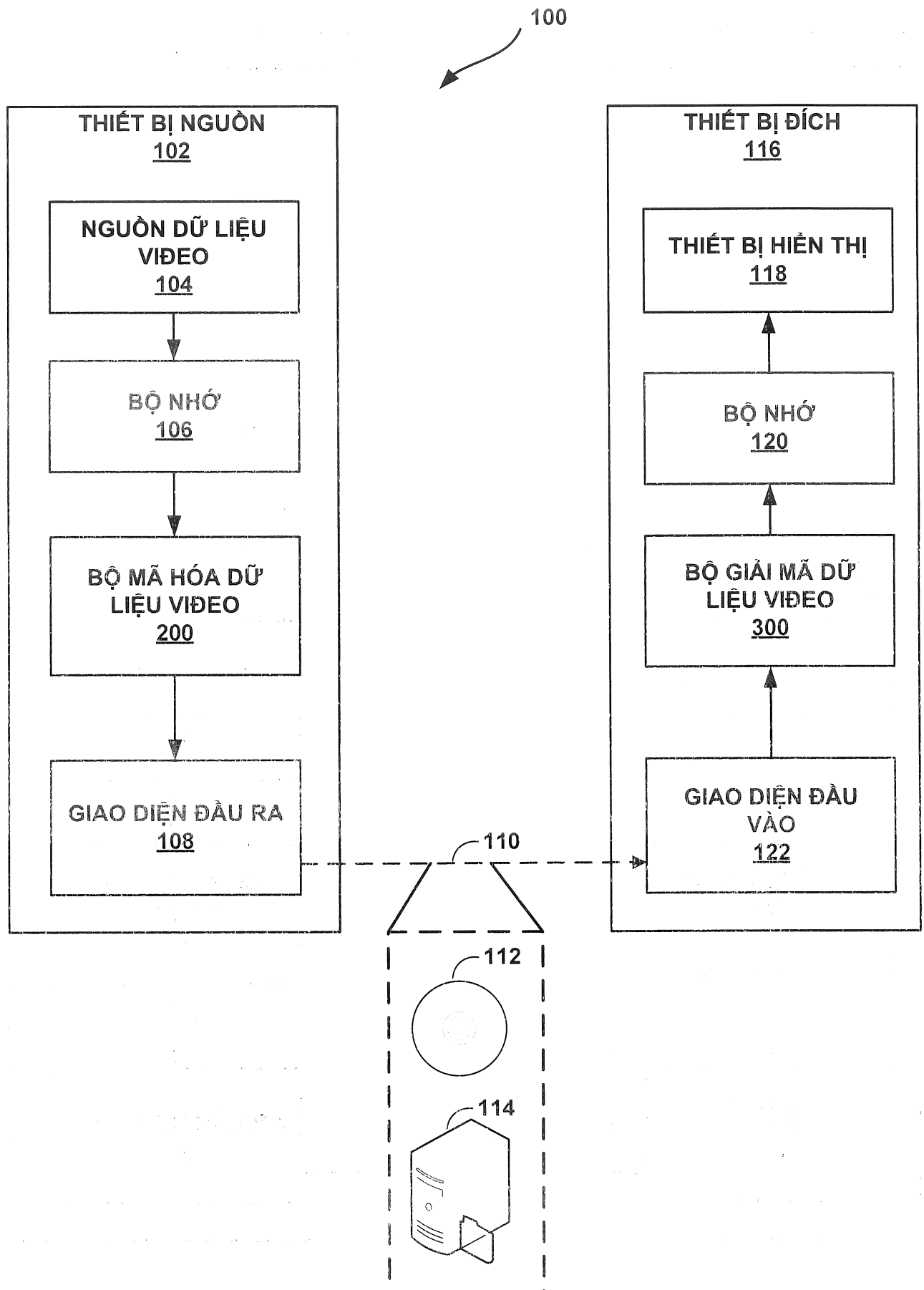


FIG.1

2 / 11

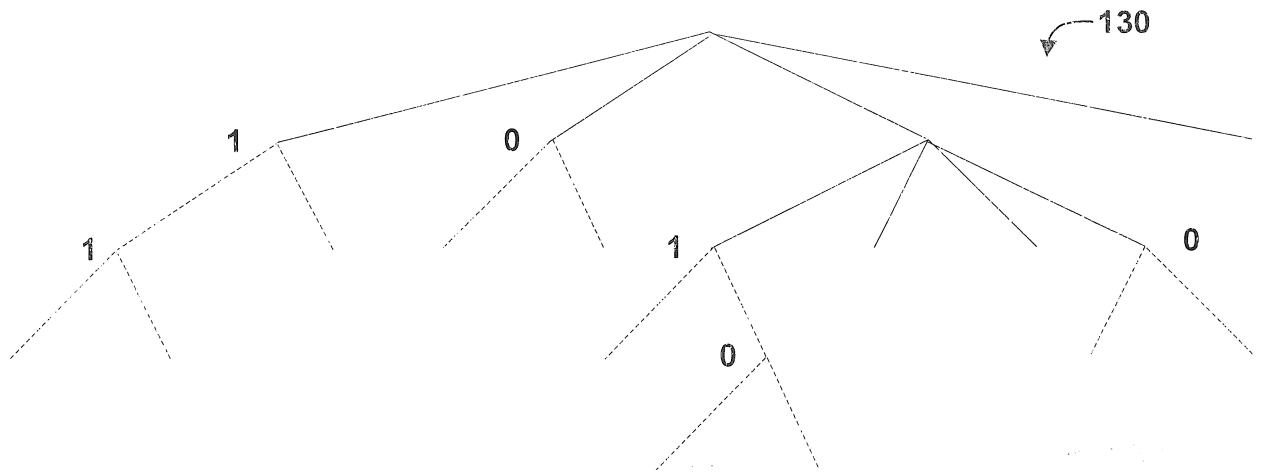


FIG.2A

132

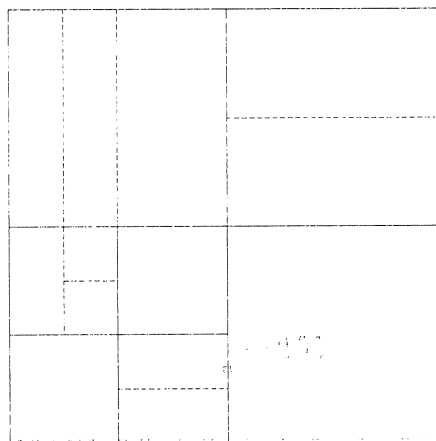


FIG.2B

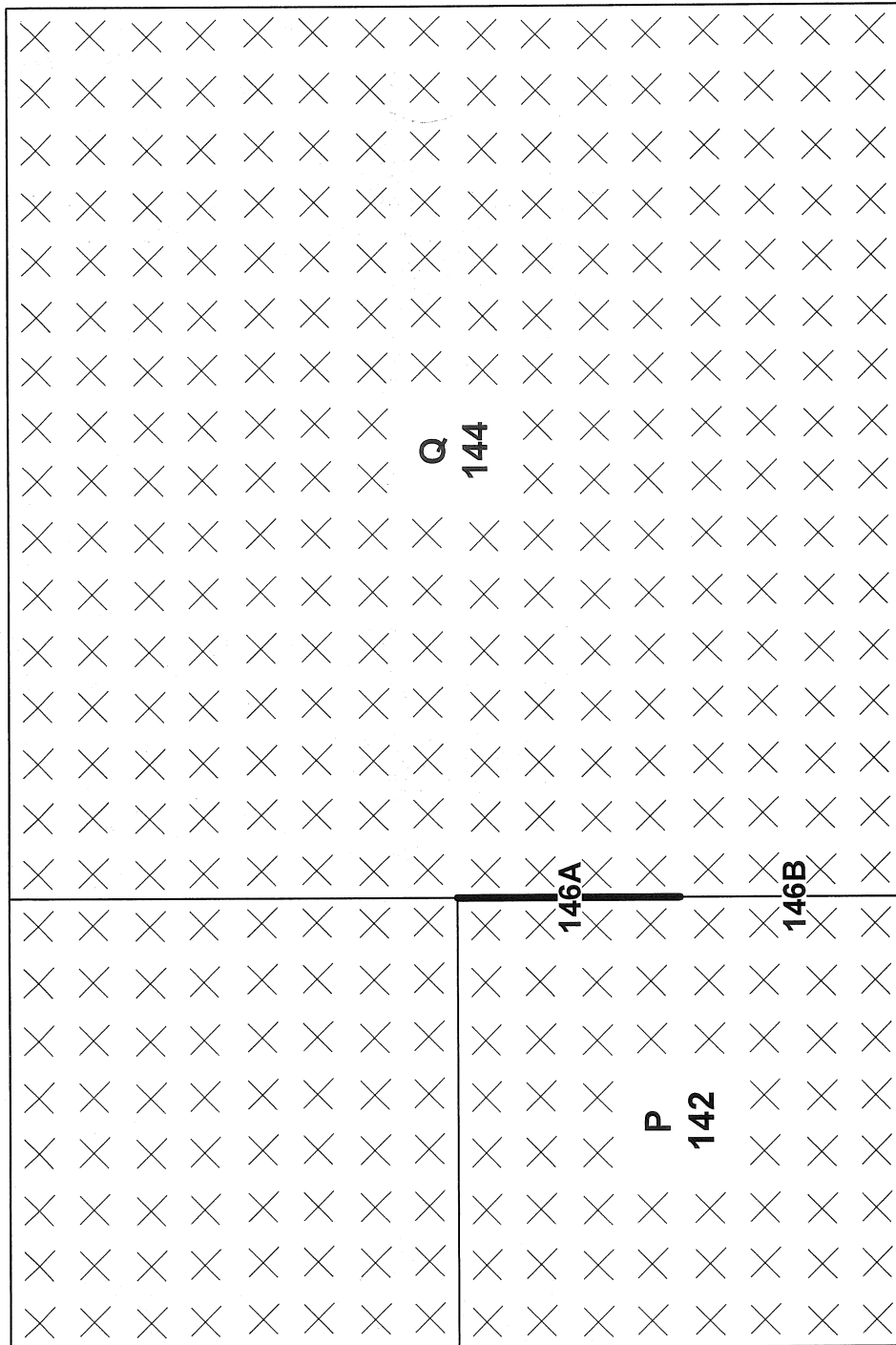


FIG.3

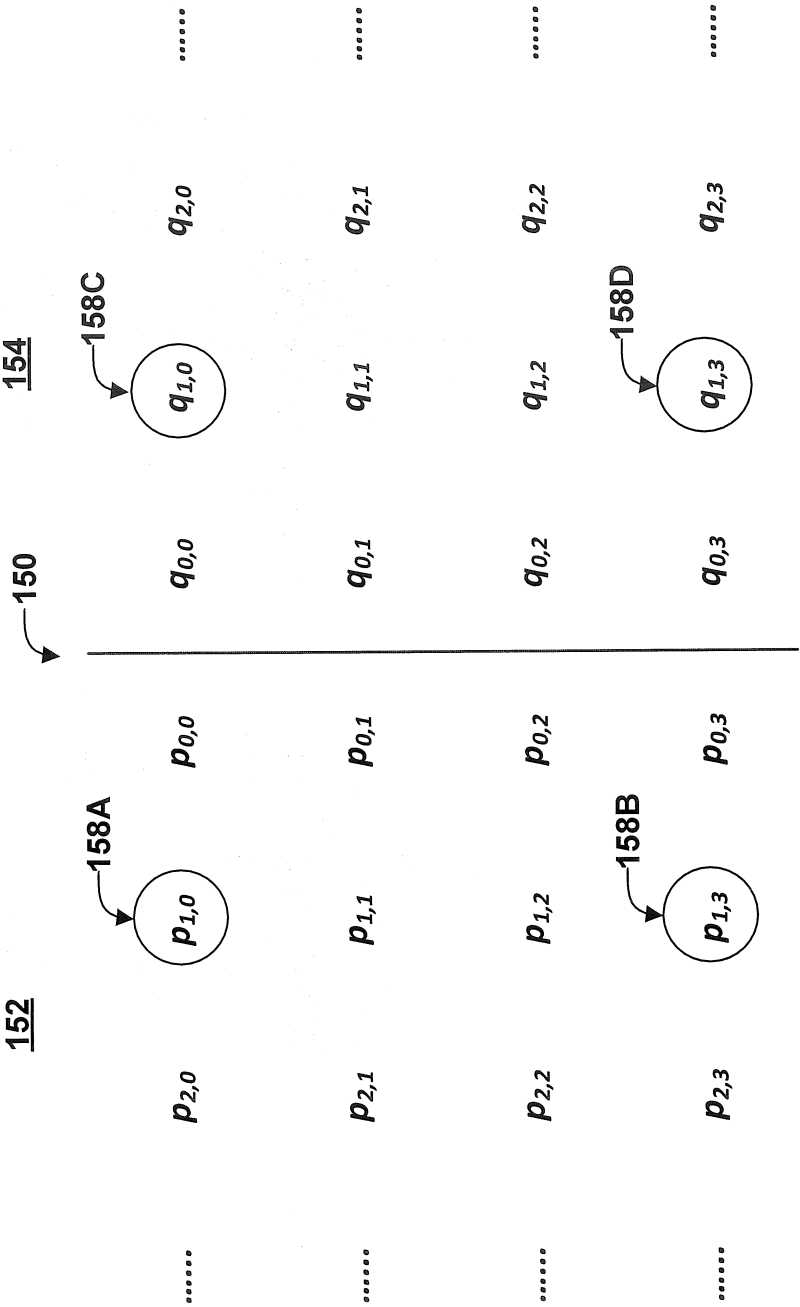


FIG.4

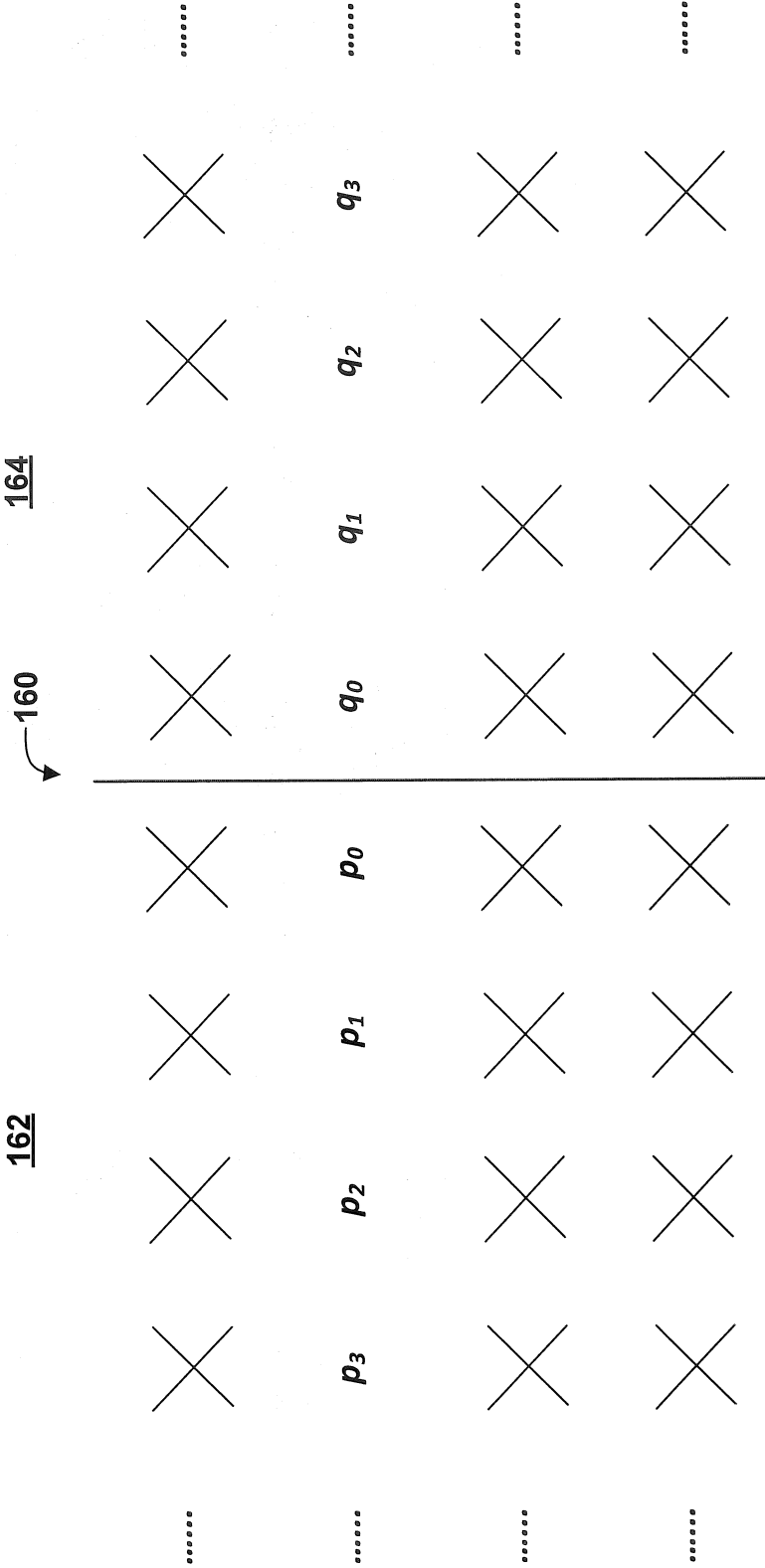
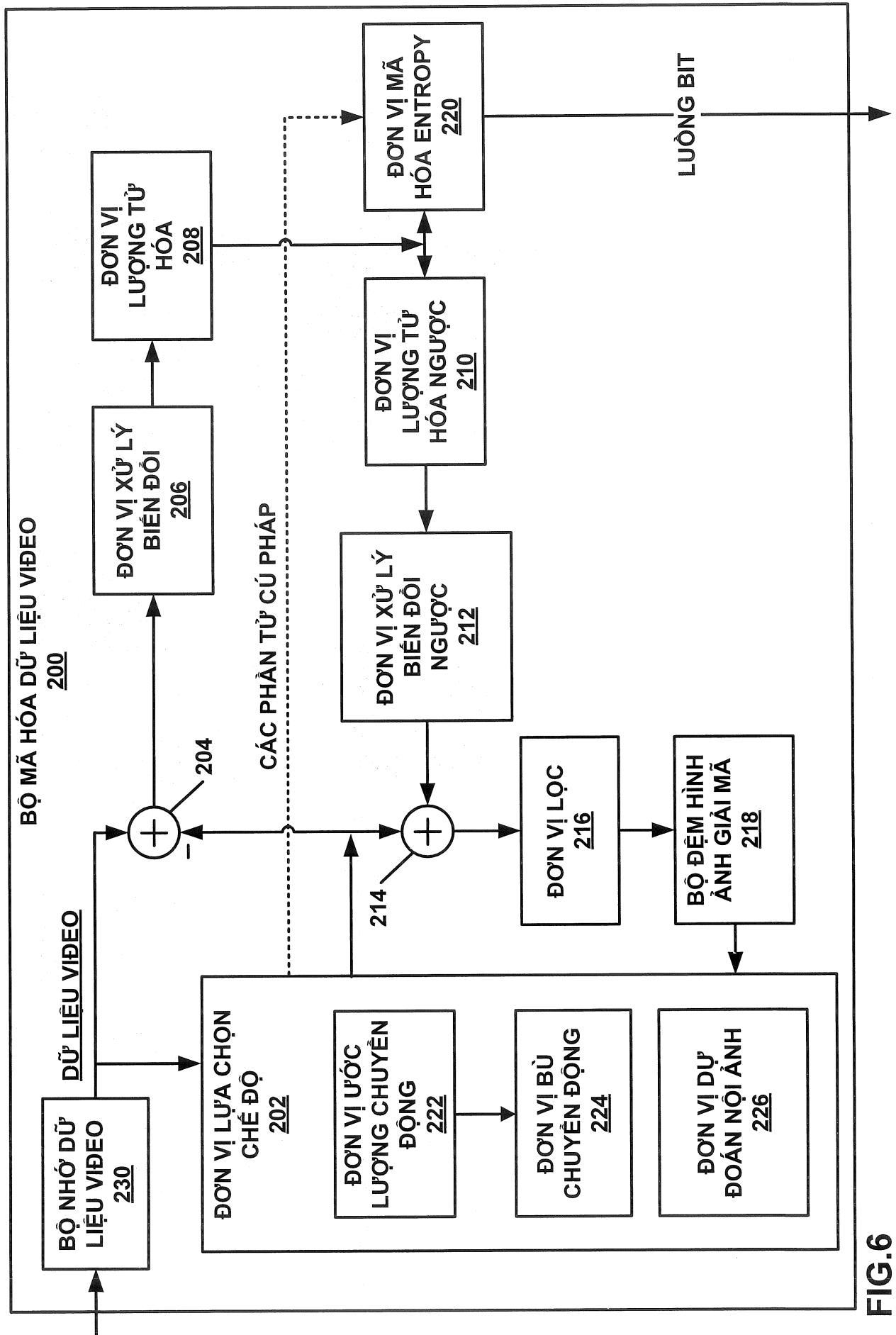


FIG.5

6 / 11



7 / 11

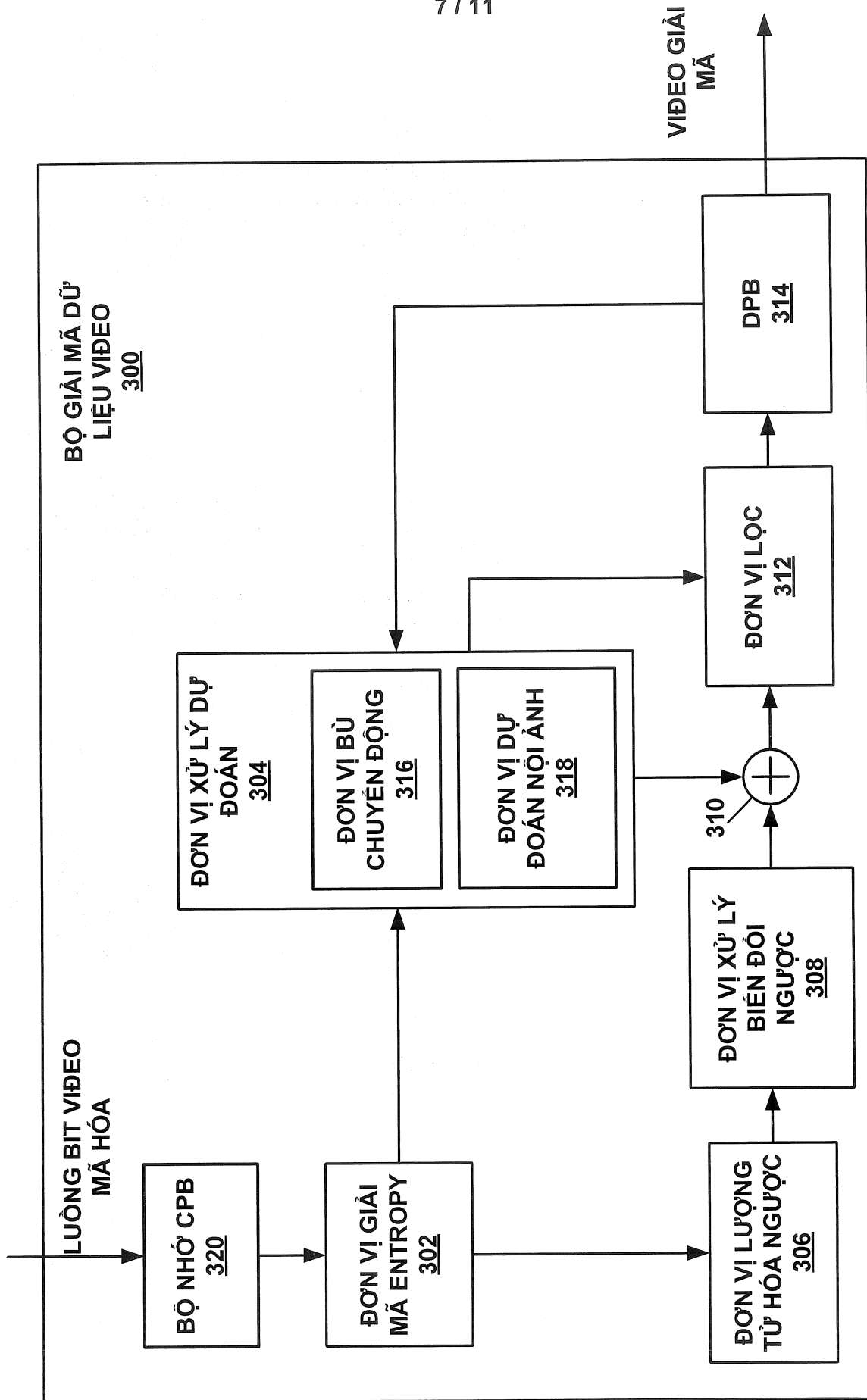


FIG.7

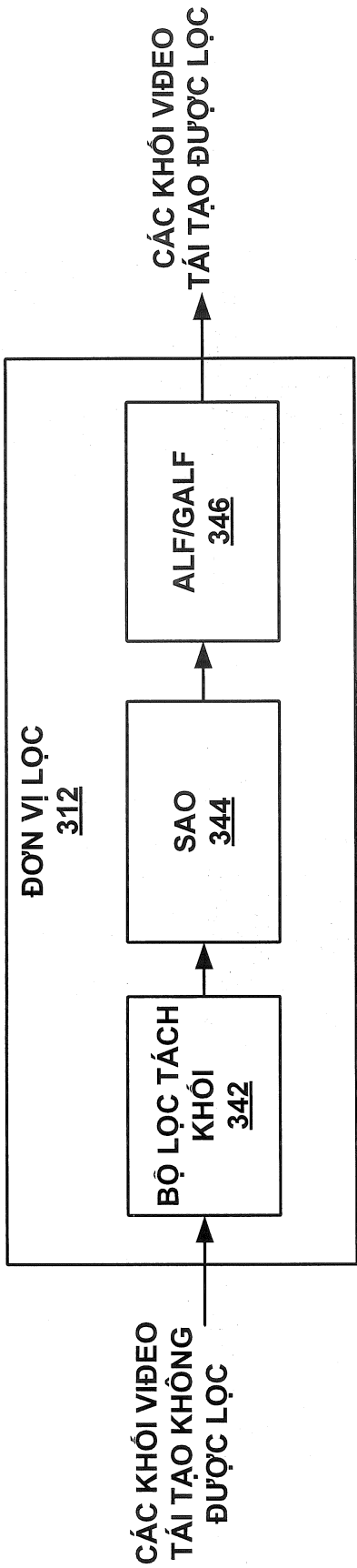


FIG.8

9 / 11

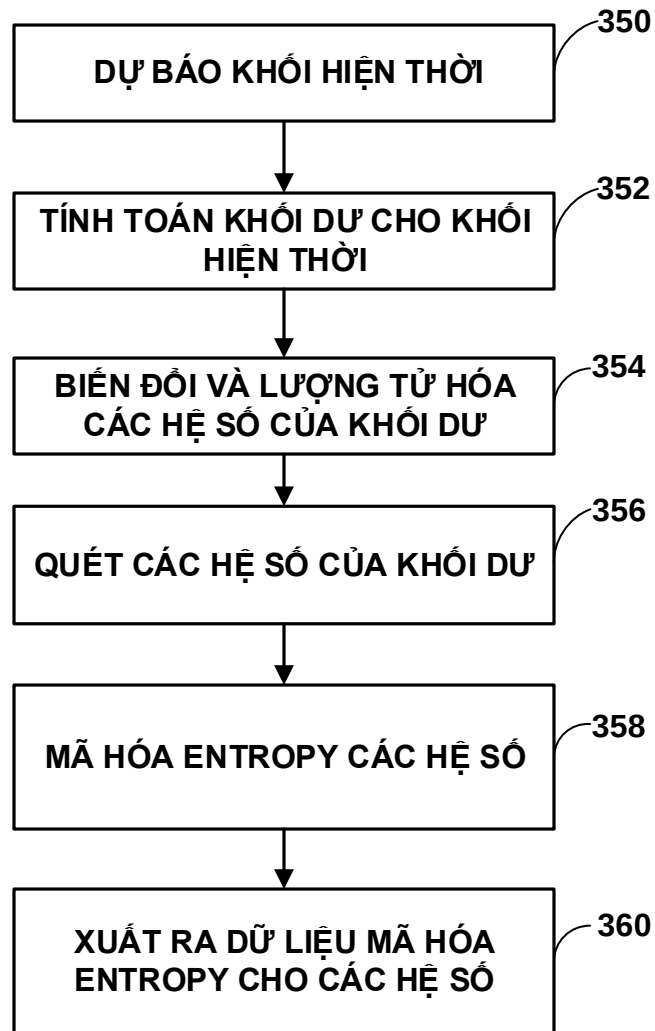


FIG.9

10 / 11

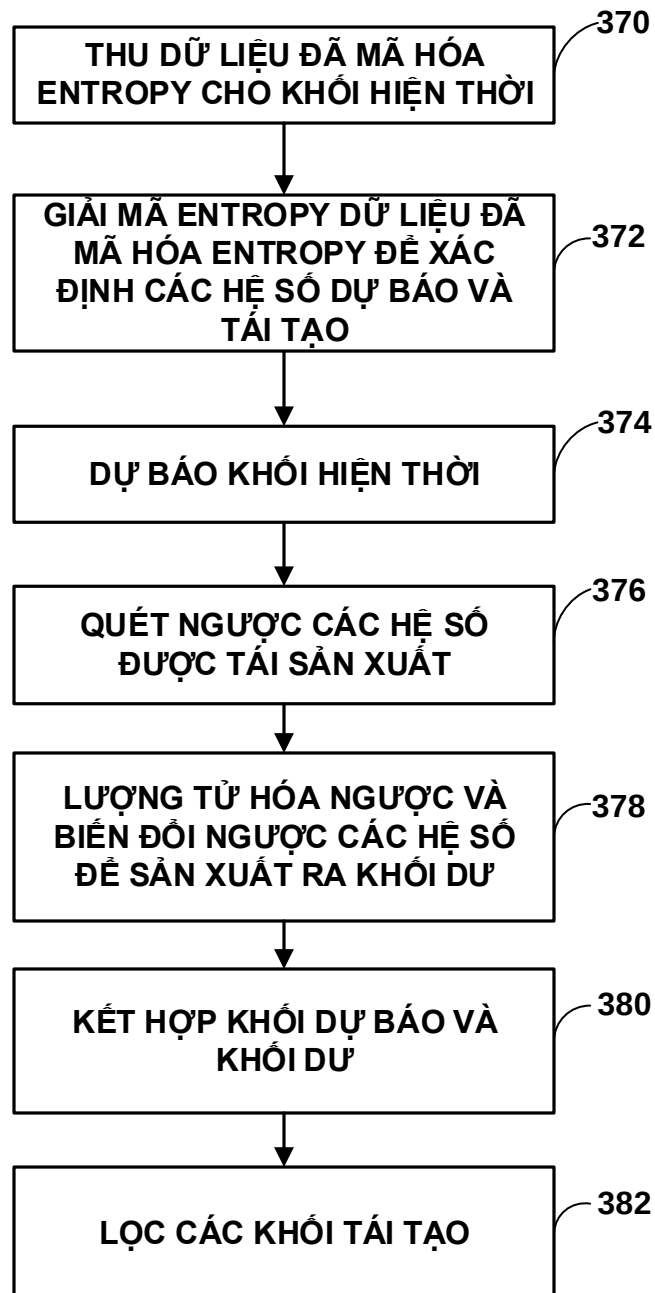


FIG.10

11 / 11

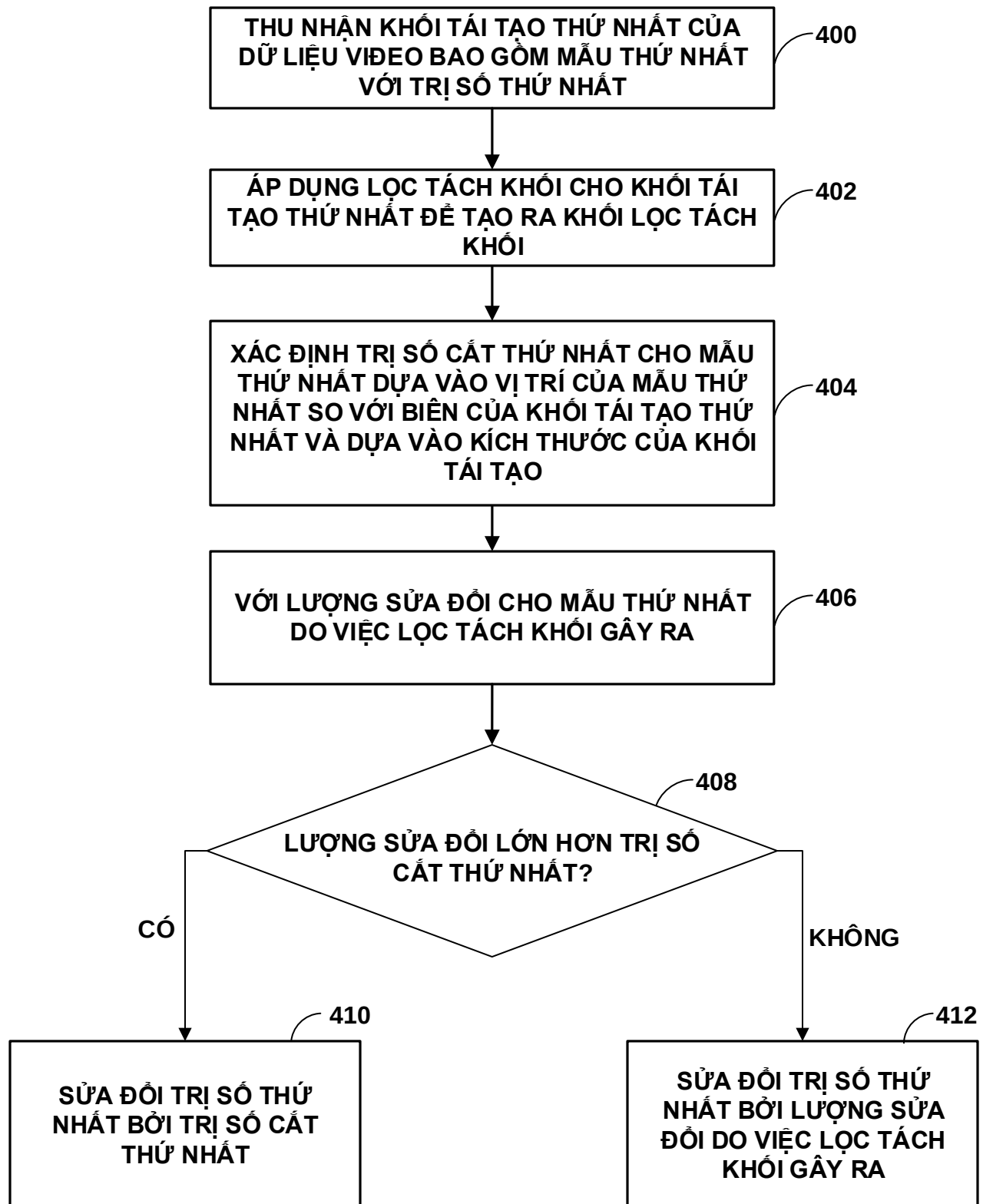


FIG.11