



(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)



1-0049766

(51)^{2020.01} H04N 19/105; H04N 19/174; H04N
19/82; H04N 19/107

(13) B

(21) 1-2021-03769

(22) 27/11/2019

(86) PCT/US2019/063694 27/11/2019

(87) WO2020/113065 04/06/2020

(30) 62/772,066 27/11/2018 US; 62/771,941 27/11/2018 US

(45) 25/08/2025 449

(43) 27/09/2021 402A

(73) OP SOLUTIONS, LLC (US)

368 Middle Street, Amherst, Massachusetts 01002, United States of America

(72) KALVA, Hari (US); FURHT, Borivoje (US); ADZIC, Velibor (US).

(74) Công ty Luật TNHH T&G (TGVN)

(54) BỘ GIẢI MÃ

(21) 1-2021-03769

(57) Sáng chế đề cập đến bộ giải mã bao gồm hệ mạch được tạo cấu hình để nhận khối hiện tại, xác định khối hiện là khối để cập nhật khung tham chiếu không khả dụng dựa trên khối hiện tại, và cập nhật khung tham chiếu không khả dụng này nhờ sử dụng khối hiện tại. Sáng chế cũng đề cập đến các phương pháp tạo mã, thiết bị, các hệ thống, các công nghệ và các vật phẩm liên quan.

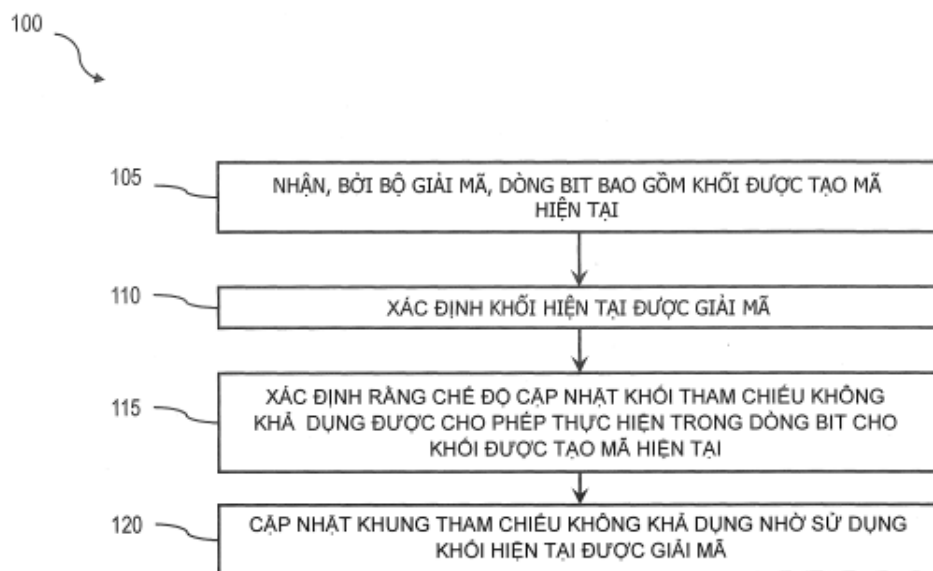


FIG. 1

Lĩnh vực kỹ thuật được đề cập

Nhìn chung, sáng chế đề cập đến lĩnh vực nén video. Cụ thể, sáng chế đề cập đến việc cập nhật khối thích ứng cho hình ảnh tham chiếu không khả dụng.

Tình trạng kỹ thuật của sáng chế

Codec video có thể bao gồm mạch điện tử hoặc phần mềm mà nó nén hoặc giải nén video kỹ thuật số. Nó có thể chuyển đổi video không được nén thành định dạng được nén hoặc ngược lại. Trong ngữ cảnh của việc nén video, thiết bị mà nó nén video (và/hoặc thực hiện một số chức năng của việc này) thường có thể được gọi là bộ mã hóa, và thiết bị mà nó giải nén video (và/hoặc thực hiện một số chức năng của việc này) có thể được gọi là bộ giải mã.

Định dạng của dữ liệu được nén có thể tuân thủ theo thông số kỹ thuật nén video tiêu chuẩn. Việc nén có thể là có mất mát ở chỗ video được nén thiếu một số thông tin có mặt trong video ban đầu. Hệ quả của điều này có thể bao gồm việc video được giải nén có thể có chất lượng thấp hơn video không được nén ban đầu do có không đủ thông tin để xây dựng lại chính xác video ban đầu.

Có thể có những mối quan hệ phức tạp giữa chất lượng video, lượng dữ liệu được sử dụng để thể hiện video (ví dụ như được xác định bởi tốc độ bit), độ phức tạp của các thuật toán mã hóa và giải mã, độ nhạy với các mất mát và sai số dữ liệu, độ dễ biên tập, việc truy cập ngẫu nhiên, độ trì hoãn đầu-mút-đầu-mút (ví dụ, độ trễ), và dạng tương tự.

Việc bù chuyển động có thể bao gồm cách tiếp cận để dự đoán khung video hoặc một phần của nó khi đã cho khung tham chiếu, chẳng hạn như các khung trước đó và/hoặc các khung tương lai, bằng cách xem xét tới chuyển động của camera và/hoặc các vật thể trong video. Nó có thể được sử dụng trong việc mã hóa và giải mã dữ liệu video để nén video, ví dụ như trong việc mã hóa và giải mã sử dụng tiêu chuẩn nhóm chuyên gia hình ảnh động (Motion Picture Experts Group, MPEG)-2 (cũng được gọi là tạo mã video cải tiến

(advanced video coding, AVC) và H.264). Việc bù chuyển động có thể mô tả hình ảnh dưới dạng việc biến đổi của hình ảnh tham chiếu thành hình ảnh hiện tại. Hình ảnh tham chiếu có thể là ở trước theo thời gian khi so sánh với hình ảnh hiện tại, từ tương lai khi so sánh với hình ảnh hiện tại, hoặc có thể bao gồm khung tham chiếu dài hạn (long term reference, LTR). Khi các ảnh có thể được tổng hợp một cách chính xác từ các ảnh được truyền và/hoặc được lưu trữ trước đó, hiệu quả nén có thể được cải thiện.

Các tiêu chuẩn hiện tại chẳng hạn như H.264 và H.265 cho phép việc cập nhật các khung chẳng hạn như các khung tham chiếu dài hạn bằng cách báo hiệu khung mới được giải mã để được lưu và làm cho khả dụng dưới dạng khung tham chiếu. Các cập nhật này được báo hiệu bởi bộ mã hóa và toàn bộ khung được cập nhật. Tuy nhiên việc cập nhật toàn bộ khung có thể là tốn kém, cụ thể khi chỉ phần nhỏ của nền tĩnh là đã thay đổi. Các cập nhật khung một phần là khả thi nhưng chúng có thể thường liên quan đến các thủ tục phức tạp và tốn kém về mặt tính toán để thực hiện các cập nhật khung.

Bản chất kỹ thuật của sáng chế

Theo một khía cạnh, bộ giải mã bao gồm hệ mạch được tạo cấu hình để nhận dòng bit bao gồm khối được tạo mã hiện tại, xác định khối hiện tại được giải mã, xác định rằng chế độ cập nhật khối tham chiếu không khả dụng được cho phép thực hiện trong dòng bit cho khối được tạo mã hiện tại, và cập nhật khung tham chiếu không khả dụng nhờ sử dụng khối hiện tại được giải mã.

Trong một khía cạnh khác, phương pháp bao gồm bước nhận, bởi bộ giải mã, dòng bit bao gồm khối được tạo mã hiện tại. Phương pháp này bao gồm bước xác định khối hiện tại được giải mã. Phương pháp này bao gồm bước xác định rằng chế độ cập nhật khối tham chiếu không khả dụng được cho phép thực hiện trong dòng bit cho khối được tạo mã hiện tại. Phương pháp này bao gồm bước cập nhật khung tham chiếu không khả dụng nhờ sử dụng khối hiện tại được giải mã.

Trong một khía cạnh khác, bộ giải mã bao gồm hệ mạch được tạo cấu hình để nhận khối hiện tại, đánh dấu khối hiện tại để cập nhật khung tham chiếu không khả dụng dựa trên khối hiện tại và cập nhật khung tham chiếu không khả dụng này nhờ sử dụng khối hiện

tại.

Trong một khía cạnh khác, phương pháp này bao gồm bước nhận, bởi bộ giải mã, khối hiện tại. Phương pháp này bao gồm bước đánh dấu khối hiện tại này để cập nhật khung tham chiếu không khả dụng dựa trên khối hiện tại này. Phương pháp này bao gồm bước cập nhật khung tham chiếu không khả dụng nhờ sử dụng khối hiện tại.

Các chi tiết của một hoặc nhiều biến thể của đối tượng của sáng chế được mô tả ở đây được nêu trên các hình vẽ kèm theo và phần mô tả dưới đây. Các dấu hiệu và các ưu điểm khác của sáng chế được mô tả ở đây sẽ trở nên rõ ràng từ phần mô tả và các hình vẽ, và từ các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Nhằm mục đích minh họa sáng chế, các hình vẽ thể hiện các khía cạnh của một hoặc nhiều phương án của sáng chế. Tuy nhiên, cần hiểu rằng sáng chế không bị giới hạn ở các cách sắp xếp và phương tiện chính xác được thể hiện trên các hình vẽ này, trong đó:

Fig.1 là sơ đồ luồng xử lý minh họa quy trình ví dụ của các cập nhật khối của khung UR trong đó khung UR đang cập nhật một phần bởi bộ giải mã và dựa trên khối hiện tại được giải mã;

Fig.2 là sơ đồ luồng xử lý minh họa quy trình ví dụ của các cập nhật khối của khung UR nhờ sử dụng việc báo hiệu ngầm định trong đó khung UR được cập nhật một phần bởi bộ giải mã và dựa trên khối hiện tại được giải mã;

Fig.3 là sơ đồ khối minh họa khung ví dụ bao gồm khối hiện tại X và ba khối lân cận có các vectơ chuyển động được liên kết mà chúng sử dụng khung UR cho việc ước lượng chuyển động;

Fig.4 là sơ đồ khối minh họa khung ví dụ khác mà nó bao gồm các khối mà chúng không sử dụng khung UR để tham chiếu và các khối mà chúng sử dụng khung UR để tham chiếu;

Fig.5 là sơ đồ khối hệ thống minh họa bộ giải mã ví dụ có khả năng giải mã dòng bit bằng việc cập nhật khối của khung UR;

Fig.6 là sơ đồ luồng xử lý minh họa quy trình ví dụ để mã hóa video bằng các cập

nhật khối của khung UR theo một số khía cạnh của sáng chế;

Fig.7 là sơ đồ luồng xử lý minh họa quy trình ví dụ để mã hóa video bằng các cập nhật khối của khung UR theo một số khía cạnh của sáng chế mà nó có thể làm giảm độ phức tạp mã hóa trong khi làm tăng hiệu quả nén;

Fig.8 là sơ đồ khối hệ thống minh họa bộ mã hóa video ví dụ có khả năng báo hiệu cho các cập nhật khối UR phía bộ giải mã;

và

Fig.9 là sơ đồ khối của hệ thống tính toán mà nó có thể được sử dụng để thi hành một hoặc nhiều phương pháp bất kỳ trong số các phương pháp được bộc lộ ở đây và một hoặc nhiều phần bất kỳ của chúng.

Các hình vẽ này không nhất thiết là đúng tỷ lệ và có thể được minh họa bởi các đường ảo, các cách biểu diễn dạng biểu đồ và các hình rời rạc. Trong các ví dụ nhất định, các chi tiết không cần thiết cho việc hiểu các phương án hoặc tạo ra các chi tiết khác khó lĩnh hội có thể đã được lược bỏ. Các ký hiệu tham chiếu giống nhau trên các hình vẽ khác nhau biểu thị các phần tử giống nhau.

Mô tả chi tiết sáng chế

Một số cách thi hành của sáng chế bao gồm các cách tiếp cận để thực hiện các cập nhật một phần cho khung tham chiếu không khả dụng (unavailable reference, UR) trên phía bộ giải mã. Khung tham chiếu không khả dụng (UR) là khung và/hoặc hình ảnh được sử dụng để tạo ra các khung và/hoặc hình ảnh được dự đoán trong một hoặc nhiều nhóm các hình ảnh (group of pictures, GOP), nhưng bản thân nó không được hiển thị trong hình ảnh video. Khung được đánh dấu là khung UR trong dòng bit video có thể là khả dụng để sử dụng làm tham chiếu cho đến khi nó được loại bỏ một cách tường minh qua việc báo hiệu dòng bit. Các khung UR có thể cải thiện hiệu quả dự đoán và nén trong các cảnh có nền tĩnh trong khoảng thời gian mở rộng (ví dụ, nền trong hội thảo video hoặc video giám sát bãi đỗ xe). Tuy nhiên, theo thời gian, nền của cảnh dần thay đổi (ví dụ, những chiếc xe hơi khi được đỗ trong chỗ trống trở thành một phần của cảnh nền). Các cập nhật khung UR một phần có thể được báo hiệu tường minh trong dòng bit nhờ sử dụng chế độ cập nhật khối

tham chiếu không khả dụng mà, khi được cho phép thực hiện đối với khối hiện tại đã cho, có thể chỉ ra rằng khối hiện tại được giải mã được sử dụng để cập nhật các điểm ảnh được đặt cùng chỗ (co-located) theo không gian bên trong khung UR. Các cập nhật khung UR một phần này có thể cải thiện việc dự đoán mà không yêu cầu việc cập nhật toàn bộ khung UR, mà điều này có thể làm giảm sai số phần dư và do đó cải thiện hiệu suất nén. Hơn nữa, qua việc báo hiệu một cách tường minh các cập nhật khối UR, các cập nhật UR có thể đạt được một cách đơn giản và không cần sử dụng các thủ tục không thỏa đáng và phức tạp.

Theo cách khác hoặc thêm vào đó, các cập nhật khung UR một phần có thể được báo hiệu ngầm định sao cho không có việc báo hiệu tường minh nào, hoặc việc báo hiệu tường minh giảm, trong dòng bit là bắt buộc khi được so sánh với một số cách tiếp cận khác. Cách tiếp cận tín hiệu ngầm định có thể chỉ ra rằng khối hiện tại được giải mã được sử dụng để cập nhật các điểm ảnh được đặt cùng chỗ theo không gian bên trong khung UR. Các cập nhật khung UR một phần này có thể cải thiện việc dự đoán mà không bắt buộc việc cập nhật toàn bộ khung UR, mà điều này có thể làm giảm sai số phần dư và do đó cải thiện hiệu suất nén. Hơn nữa, qua việc báo hiệu ngầm định các cập nhật khối UR, phí tổn tiêu đề có thể được giảm xuống, qua đó làm giảm tốc độ bit và cải thiện hoạt động nén. Thêm nữa, trong một số cách thi hành, các cập nhật UR có thể đạt được một cách đơn giản và không cần sử dụng các thủ tục không thỏa đáng và phức tạp.

Trong một số cách thi hành, việc báo hiệu ngầm định các khối video nhận dạng ra các trường hợp trong đó ngữ cảnh khối hiện tại và lân cận được sử dụng để xác định một cách ngầm định các khối nào được cập nhật trong khung UR (ví dụ, các phần nào của khung UR cần được cập nhật). Giờ đây tham chiếu đến Fig.1, sơ đồ luồng xử lý được đưa ra để minh họa quy trình ví dụ 100 của các cập nhật khối của khung UR nhờ sử dụng việc báo hiệu tường minh trong đó khung UR được cập nhật một phần bởi bộ giải mã và dựa trên khối hiện tại được giải mã. Các cập nhật khối UR này có thể cải thiện việc dự đoán mà không yêu cầu việc cập nhật toàn bộ khung UR, mà điều này có thể làm giảm sai số phần dư và do đó cải thiện hiệu suất nén.

Tại bước 105, và vẫn tham chiếu đến Fig.1, dòng bit bao gồm khối được tạo mã hiện tại được nhận bởi bộ giải mã. Dòng bit có thể bao gồm, ví dụ, dữ liệu được tìm thấy trong

dòng của các bit mà nó là đầu vào cho bộ giải mã khi sử dụng việc nén dữ liệu. Dòng bit có thể bao gồm thông tin cần thiết để giải mã video. Việc nhận có thể bao gồm việc trích xuất và/hoặc phân tích cú pháp khối và thông tin báo hiệu được liên kết với nó từ dòng bit. Trong một số cách thi hành, khối được tạo mã hiện tại có thể bao gồm đơn vị cây tạo mã (coding tree unit, CTU), đơn vị tạo mã (coding unit, CU), hoặc đơn vị dự đoán (prediction unit, PU).

Tại bước 110, và tiếp tục tham chiếu đến Fig.1, khối hiện tại được giải mã có thể được xác định. Ví dụ, khối được tạo mã hiện tại được nhận có thể được giải mã, ví dụ, bằng cách sử dụng việc dự đoán liên. Hoạt động giải mã thông qua việc dự đoán liên có thể bao gồm việc sử dụng khung trước đó, khung tương lai, hoặc khung UR làm tham chiếu để tính toán việc dự đoán, mà nó có thể được kết hợp với phần dư được chứa trong dòng bit.

Tại bước 115, và tiếp tục tham chiếu đến Fig.1, việc liệu chế độ cập nhật khối tham chiếu không khả dụng có được cho phép thực hiện trong dòng bit cho khối được tạo mã hiện tại hay không có thể được xác định. Chế độ cập nhật khối UR có thể được chứa trong tiêu đề của dòng bit và/hoặc trường được chứa trong đó. Ví dụ, có thể xác định được rằng trường chế độ cập nhật khối UR trong tiêu đề của dòng bit được cho phép thực hiện. Việc báo hiệu khối khung UR được cập nhật có thể được cho phép thực hiện nhờ sử dụng theo cách có chọn lọc tiêu đề chẳng hạn như tập tham số hình ảnh (picture parameter set, PPS) và/hoặc tập tham số chuỗi (sequence parameter set, SPS). Trường, chẳng hạn như UR_BLOCK_UPDATE có thể có, nhưng không giới hạn ở, các giá trị đúng hoặc sai (ví dụ, 0 hoặc 1).

Vẫn tham chiếu đến Fig.1, nếu chế độ cập nhật khối UR được cho phép thực hiện (ví dụ, nếu trường UR_BLOCK_UPDATE trong tiêu đề được thiết lập), thì tại 120, bộ giải mã có thể cập nhật khung tham chiếu không khả dụng nhờ sử dụng khối hiện tại được giải mã. Việc cập nhật này có thể bao gồm việc thực hiện các cập nhật khối UR, trong đó các điểm ảnh trong khung UR mà chúng được đặt cùng chỗ theo không gian với khối hiện tại, mà chế độ cập nhật khối của khung UR được cho phép thực hiện đối với nó, được cập nhật (ví dụ, được cải biến) nhờ sử dụng khối hiện tại được giải mã. Trong một số cách thi hành, các cơ chế cập nhật khối có thể là tường minh hoặc ngầm định. Ví dụ, một phần của (ví dụ,

khối của) khung UR có thể được cập nhật bằng cách cập nhật các điểm ảnh (ví dụ, các giá trị độ chói) được đặt cùng chỗ trong khung UR với các trị số điểm ảnh của khối hiện tại được giải mã; nói cách khác, việc cập nhật khung UR có thể bao gồm việc thay thế các giá trị độ chói của khung UR bằng các giá trị độ chói được đặt cùng chỗ theo không gian của khối hiện tại được giải mã. Trong một số cách thi hành, khung UR có thể được cập nhật theo cơ chế khác, chẳng hạn như việc cập nhật một phần của khung UR bằng trung bình của các giá trị điểm ảnh khung UR được đặt cùng chỗ ban đầu và các giá trị điểm ảnh của khối được giải mã hiện tại. Những người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng, khi xem xét toàn bộ sáng chế, sẽ hiểu được các cơ cấu khác nhau đa dạng có thể được sử dụng. Trong một số cách thi hành, việc cập nhật có thể bao gồm việc cập nhật nhiều khối của khung UR bằng nhiều khối được giải mã, mà việc này có thể được thực hiện nhờ sử dụng các phương pháp bất kỳ được mô tả trên đây để cập nhật khung UR bằng khối hiện tại được giải mã. Việc cập nhật có thể bao gồm, mà không giới hạn ở, việc tạo ra khối và/hoặc khung của các giá trị độ chói và/hoặc sắc độ mặc định và việc thay thế và/hoặc cập nhật của chúng từ một hoặc nhiều khối được giải mã nhờ sử dụng quy trình và/hoặc các bước quy trình bất kỳ được mô tả trên đây; việc cập nhật có thể bao gồm việc tạo ra khung UR, mà việc này có thể được hoàn thành bằng cách tạo ra khung UR có các giá trị mặc định và việc cập nhật các giá trị mặc định này như được mô tả trên đây.

Tiếp tục tham chiếu đến Fig.1, đối với các khối hiện tại tiếp theo, khung UR được cập nhật có thể được sử dụng làm khung tham chiếu cho việc dự đoán liên. Ví dụ, khối được tạo mã thứ hai có thể được nhận. Việc liệu chế độ dự đoán liên có được cho phép thực hiện đối với khối được tạo mã thứ hai này hay không có thể được xác định. Khối được giải mã thứ hai có thể được xác định nhờ sử dụng khung UR được cập nhật làm khung tham chiếu và theo chế độ dự đoán liên. Ví dụ, việc giải mã thông qua việc dự đoán liên có thể bao gồm việc sử dụng khung UR được cập nhật làm tham chiếu để tính toán việc dự đoán, mà nó có thể được kết hợp với phần dư được chứa trong dòng bit.

Vẫn tham chiếu đến Fig.1, các cập nhật khối UR có thể là khả dụng cho mỗi khối hiện tại trong quy trình giải mã. Trong một số cách thi hành, các cập nhật khối UR có thể được bỏ qua theo cách ngầm định đối với các khung được tạo mã nội. Ví dụ, dòng bit có thể bao

gồm khối được tạo mã hiện tại thứ hai bên trong khung khác với khối được tạo mã hiện tại thứ nhất; chế độ của khối được tạo mã hiện tại thứ hai này có thể bao gồm việc dự đoán nội. Cập nhật khối UR có thể được bỏ qua để đáp ứng lại việc xác định rằng khối được tạo mã hiện tại thứ hai là khối dự đoán nội. Việc bỏ qua có thể bao gồm, ví dụ, việc không cập nhật khung UR hoặc việc xác định xem liệu trường chẳng hạn như UR_BLOCK_UPDATE có được thiết lập trong tiêu đề hay không.

Trong một số cách thi hành, và tham chiếu thêm đến Fig.1, nếu trường UR_BLOCK_UPDATE trong tiêu đề được thiết lập thì bộ giải mã có thể dự kiến các bit báo hiệu cập nhật khối tường minh trong dòng bit. Trong cách tiếp cận này, khối được tạo mã của video có thể được báo hiệu trong tiêu đề khối của dòng bit video làm khối cần được cập nhật trong khung UR. Trong trường hợp đó, khối được đặt cùng chỗ trong khung UR có thể được thay thế bởi khối được báo hiệu; khung UR được cập nhật này có thể được sử dụng, ví dụ, cho các mục đích ước lượng chuyển động trong tương lai cho đến khi khung UR được cập nhật lần tới.

Fig.2 là sơ đồ luồng xử lý minh họa phương án làm ví dụ của quy trình 200 của các cập nhật khối của khung UR nhờ sử dụng việc báo hiệu ngầm định trong đó khung UR được cập nhật một phần bởi bộ giải mã và dựa trên khối hiện tại được giải mã. Việc báo hiệu ngầm định có thể bao gồm các trường hợp trong đó ngữ cảnh khối hiện tại và lân cận được sử dụng để xác định các khối nào của khung UR cần được cập nhật khi so sánh với việc báo hiệu tường minh trong chế độ cập nhật khối UR trong dòng bit, chẳng hạn như trong tiêu đề của dòng bit. Các cập nhật khối UR với việc báo hiệu ngầm định như vậy có thể cải thiện việc dự đoán và làm giảm phí tổn báo hiệu, qua đó cải thiện tốc độ bit và cải thiện hoạt động nén. Tương tự, qua việc thực hiện các cập nhật khối UR mà không yêu cầu cập nhật toàn bộ khung UR, một số cách thi hành có thể làm giảm sai số phần dư và do đó cải thiện hiệu suất nén.

Tại bước 205, và vẫn tham chiếu đến Fig.2, khối hiện tại được nhận bởi bộ giải mã. Khối hiện tại có thể được chứa trong dòng bit mà bộ giải mã nhận. Dòng bit có thể bao gồm, ví dụ, dữ liệu được tìm thấy trong dòng của các bit mà nó là đầu vào cho bộ giải mã khi sử dụng việc nén dữ liệu. Dòng bit có thể bao gồm thông tin cần thiết để giải mã video.

Việc nhận có thể bao gồm việc trích xuất và/hoặc phân tích cú pháp khối và thông tin báo hiệu được liên kết với nó từ dòng bit. Trong một số cách thi hành, khối hiện tại có thể bao gồm đơn vị cây tạo mã (coding tree unit, CTU), đơn vị tạo mã (coding unit, CU), và/hoặc đơn vị dự đoán (prediction unit, PU).

Vẫn tham chiếu đến Fig.2, trong một số cách thi hành, và vẫn tham chiếu đến Fig.2, chế độ cập nhật khối tham chiếu không khả dụng có thể không được báo hiệu tường minh trong dòng bit cho khối hiện tại. Tương tự, trong một số cách thi hành, bộ giải mã có thể xác định, nhờ sử dụng dòng bit, xem liệu chế độ cập nhật khối tham chiếu không khả dụng có được báo hiệu một cách tường minh trong dòng bit cho khối hiện tại hay không. Trong một số cách thi hành, bộ giải mã có thể xác định rằng trường trong tiêu đề của dòng bit chỉ ra rằng chế độ cập nhật khối tham chiếu không khả dụng được vô hiệu hóa hoặc không có mặt.

Tại bước 220, và tiếp tục tham chiếu đến Fig.2, khối hiện tại có thể được đánh dấu để cập nhật khung UR dựa trên khối hiện tại này; khối hiện tại này có thể được đánh dấu dựa trên hoặc để đáp ứng lại việc xác định rằng khối hiện tại được báo hiệu theo cách ngầm định để sử dụng trong cập nhật khối của khung UR. Việc đánh dấu có thể bao gồm, nhưng không giới hạn ở, việc thiết lập giá trị biến số thành giá trị chỉ ra rằng khối hiện tại cần được sử dụng cho việc cập nhật khung UR; ví dụ, biến luận lý (Boolean) được liên kết với khối hiện tại và được tạo cấu hình để chỉ ra xem liệu khối hiện tại có cần được sử dụng để cập nhật khung UR hay không có thể được thiết lập là “đúng” (“true”) hoặc “1” để chỉ ra việc đánh dấu.

Ví dụ, và vẫn tham chiếu đến Fig.2, khối hiện tại có thể được đánh dấu để cập nhật khi khối hiện tại sử dụng khung UR làm tham chiếu cho việc dự đoán liên và khối hiện tại bao gồm việc áp dụng việc bù độ chiếu sáng; việc đánh dấu có thể được thực hiện để đáp ứng lại và/hoặc dựa trên việc xác định này. Việc bù độ chiếu sáng có thể được sử dụng để cải thiện việc dự đoán trong việc tạo mã video. Việc bù độ chiếu sáng có thể cải thiện việc dự đoán trong các trường hợp mà ở đó các điểm ảnh khối thay đổi do độ chiếu sáng (ví dụ, đám mây trôi qua làm giảm độ chiếu sáng). Khối sử dụng khung UR làm tham chiếu và áp dụng việc bù độ chiếu sáng có thể được xác định là khối cập nhật và được đánh dấu theo

đó để cập nhật khung UR.

Như là một ví dụ khác, và tiếp tục tham chiếu đến Fig.2, khối hiện tại có thể được đánh dấu để cập nhật khung UR khi các khối lân cận sử dụng việc tham chiếu vector chuyển động UR và dựa trên việc các khối lân cận nào sử dụng tham chiếu vector chuyển động UR. Trong video đại diện (ví dụ, điển hình), một phần của nền của cảnh cập nhật theo thời gian. Khi các thay đổi này xuất hiện, vùng được thay đổi có thể được tạo mã dưới dạng khối nội khi không có khối tương tự được tìm thấy. Trong các trường hợp như vậy, tất cả các khối ngoại trừ vùng mới được thay đổi này, có khả năng, sử dụng khung UR để tham chiếu. Khối mà nó có nhiều khối lân cận với tham chiếu vector chuyển động UR có thể được đánh dấu để cập nhật khung UR ngầm định. Theo đó, khả năng khối hiện tại là khối cập nhật UR là cao hơn khi các khối lân cận của khối hiện tại mà chúng sử dụng khung UR tăng lên; khả năng này có thể được tính và so sánh với số lượng ngưỡng, trong đó việc khả năng này vượt quá số lượng ngưỡng có thể chỉ ra rằng khối hiện tại là khối cập nhật UR. Nếu khối hiện tại được tạo mã nội thì số lượng các khối lân cận với tham chiếu UR nhỏ hơn có thể báo hiệu cập nhật khối UR; ví dụ, số lượng ngưỡng thấp hơn có thể được sử dụng khi so sánh ngưỡng như trên. Nếu khối hiện tại được tạo mã liên thì số lượng các khối lân cận với tham chiếu UR cao hơn có thể báo hiệu ngầm định cập nhật khối UR, ví dụ bằng cách sử dụng số lượng ngưỡng cao hơn. Theo đó, để làm một ví dụ khác không nhằm mục đích giới hạn, khối hiện tại có thể được đánh dấu để cập nhật khung UR để đáp ứng lại và/hoặc dựa trên việc xác định rằng hai hay hơn hai khối lân cận với khối hiện tại sử dụng vector chuyển động UR cho việc dự đoán liên.

Vẫn tham chiếu đến Fig.2, trong một số cách thi hành, các ứng viên cập nhật khối có thể được xác định sau khi toàn bộ khung đã được giải mã sao cho toàn bộ lân cận của khối có thể được xem xét. Ví dụ, khối với tham chiếu phi UR được bao quanh bởi các khối với các tham chiếu UR trên hai hay hơn hai phía có thể là ứng viên cho các cập nhật UR. Như là một ví dụ khác, nếu khối hiện tại được tạo mã nội và nhiều khối lân cận sử dụng khung UR làm tham chiếu và có vector chuyển động là $(0,0)$ (ví dụ, vector chuyển động không) thì khối hiện tại có thể được đánh dấu để cập nhật UR. Như là một ví dụ khác nữa, nếu khối hiện tại được tạo mã nội và nhiều khối lân cận sử dụng khung UR làm tham chiếu và có

vector chuyển động giống hoặc tương tự nhau (ví dụ, vector chuyển động có thể khác nhau theo lượng nhỏ hơn lượng được xác định trước hoặc định trước), thì khối hiện tại có thể được đánh dấu để cập nhật UR. Để ví dụ thêm, nếu khối hiện tại có các khối lân cận trên ít nhất ba phía của khối hiện tại này mà chúng sử dụng khung UR làm tham chiếu cho việc dự đoán liên thì khối hiện tại có thể được đánh dấu để cập nhật UR.

Trong một số cách thi hành, và tiếp tục tham chiếu đến Fig.2, cập nhật UR cũng có thể được xác định tại mức đơn vị tạo mã (CU). Nếu tất cả các CU lân cận trên ít nhất ba phía đều sử dụng tham chiếu UR thì tất cả các khối của CU hiện tại có thể được đánh dấu để cập nhật UR. Trong một số cách thi hành, cập nhật UR có thể được xác định ở mức đơn vị cây tạo mã (CTU). Nếu tất cả các lân cận CTU trên ít nhất ba phía đều sử dụng tham chiếu UR thì tất cả các khối của CTU hiện tại mà chúng không sử dụng tham chiếu UR có thể được đánh dấu để cập nhật. Fig.3 là sơ đồ khối minh họa phương án làm ví dụ về khung 300 bao gồm khối hiện tại X và ba khối lân cận có các vector chuyển động được liên kết MV1, MV2, MV3 mà chúng sử dụng khung UR cho việc ước lượng chuyển động. Trong ví dụ được minh họa này, khối hiện tại X có thể được đánh dấu để cập nhật khung UR.

Fig.4 là sơ đồ khối minh họa một phương án làm ví dụ khác về khung 400 mà nó bao gồm các khối được gắn nhãn X, mà chúng không sử dụng khung UR để tham chiếu và các khối L mà chúng sử dụng khung UR để tham chiếu; các khối được đánh dấu là X có thể được đánh dấu để cập nhật khung UR.

Tham chiếu lại đến Fig.2, tại bước 215, khung UR có thể được cập nhật nhờ sử dụng khối hiện tại. Việc cập nhật có thể bao gồm việc thực hiện các cập nhật khối UR trong đó các điểm ảnh trong khung UR được đặt cùng chỗ theo không gian với khối hiện tại được cập nhật (ví dụ, được cải biến) nhờ sử dụng khối hiện tại này. Trong một số cách thi hành, các cơ chế cập nhật khối có thể là tường minh hoặc ngầm định. Ví dụ, một phần của (ví dụ, khối của) khung UR có thể được cập nhật bằng cách cập nhật (ví dụ, thay thế) các điểm ảnh (ví dụ, các giá trị độ chói) được đặt cùng chỗ trong khung UR bằng các giá trị điểm ảnh của khối hiện tại. Trong một số cách thi hành, khung UR có thể được cập nhật theo cơ chế khác, chẳng hạn như việc cập nhật một phần của khung UR bằng trung bình của các giá trị điểm ảnh khung UR được đặt cùng chỗ ban đầu và các giá trị điểm ảnh của khối hiện tại. Những

người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng, khi xem xét toàn bộ sáng chế, sẽ hiểu được các cơ cấu khác nhau đa dạng có thể được sử dụng. Trong một số cách thi hành, việc cập nhật có thể bao gồm việc cập nhật nhiều khối của khung UR bằng nhiều khối được giải mã, mà việc này có thể được thực hiện nhờ sử dụng các phương pháp bất kỳ được mô tả trên đây để cập nhật khung UR bằng khối hiện tại được giải mã. Việc cập nhật có thể bao gồm, mà không giới hạn ở, việc tạo ra khối và/hoặc khung của các giá trị độ chói và/hoặc sắc độ mặc định và việc thay thế và/hoặc cập nhật của chúng từ một hoặc nhiều khối được giải mã nhờ sử dụng quy trình và/hoặc các bước quy trình bất kỳ được mô tả trên đây; việc cập nhật có thể bao gồm việc tạo ra khung UR, mà việc này có thể được hoàn thành bằng cách tạo ra khung UR có các giá trị mặc định và việc cập nhật các giá trị mặc định này như được mô tả trên đây.

Đối với các khối hiện tại tiếp theo, khung UR được cập nhật có thể được sử dụng làm khung tham chiếu cho việc dự đoán liên. Ví dụ, khối được tạo mã thứ hai có thể được nhận. Việc liệu chế độ dự đoán liên có được cho phép thực hiện đối với khối được tạo mã thứ hai này hay không có thể được xác định. Khối được giải mã thứ hai có thể được xác định nhờ sử dụng khung UR được cập nhật làm khung tham chiếu và theo chế độ dự đoán liên. Ví dụ, việc giải mã thông qua việc dự đoán liên có thể bao gồm việc sử dụng khung UR được cập nhật làm tham chiếu để tính toán việc dự đoán, mà nó có thể được kết hợp với phần dư được chứa trong dòng bit.

Fig.5 là sơ đồ khối hệ thống thể hiện bộ giải mã ví dụ 500 có khả năng giải mã dòng bit 504 bằng các cập nhật khối của khung UR. Các cập nhật khối của khung UR có thể bao gồm, nhưng không giới hạn ở, các cập nhật khối của khung UR được báo hiệu tường minh, ví dụ và không giới hạn như được mô tả trên đây khi tham chiếu đến Fig.1. Các cập nhật khối của khung UR có thể bao gồm, nhưng không giới hạn ở, các cập nhật khối của khung UR mà chúng được báo hiệu ngầm định (ví dụ, các cập nhật khối của khung UR được xác định từ ngữ cảnh của các khối hiện tại và/hoặc lân cận), ví dụ như được mô tả trên đây khi tham chiếu đến Fig.2 đến Fig.4. Bộ giải mã 500 bao gồm bộ xử lý của bộ giải mã entropy 508, bộ xử lý lượng tử hóa ngược và biến đổi ngược 512, bộ lọc khử khối 516, bộ đệm khung 520, bộ xử lý bù chuyển động 524 và bộ xử lý dự đoán nội 528. Trong một số cách

thi hành, dòng bit 504 có thể bao gồm các tham số (ví dụ, trường trong tiêu đề của dòng bit) mà chúng báo hiệu chế độ cập nhật khối của khung UR. Theo cách khác hoặc thêm vào đó, dòng bit 504 có thể không báo hiệu tường minh chế độ cập nhật khối của khung UR cho khối hiện tại; dòng bit có thể chứa các tham số khác. Bộ xử lý bù chuyển động 524 có thể xây dựng lại thông tin điểm ảnh nhờ sử dụng khung UR và cập nhật khung UR này theo chế độ cập nhật khối của khung UR. Ví dụ, khi chế độ cập nhật khối của khung UR được báo hiệu tường minh cho khối hiện tại, các điểm ảnh (ví dụ, các giá trị độ chói) được đặt cùng chỗ trong khung UR có thể được thay thế bằng các giá trị điểm ảnh của khối hiện tại. Khi khối cập nhật được báo hiệu theo cách ngầm định, bộ xử lý bù chuyển động 524 có thể đánh dấu khối hiện tại để cập nhật khung UR (ví dụ, dựa trên việc liệu cập nhật khung UR có được báo hiệu theo cách ngầm định hay không), và cập nhật khung UR theo đó. Ví dụ, khi cập nhật khối của khung UR được báo hiệu theo cách ngầm định cho khối hiện tại, các điểm ảnh (ví dụ, các giá trị độ chói) được đặt cùng chỗ trong khung UR có thể được thay thế bằng các giá trị điểm ảnh của khối hiện tại.

Khi vận hành, dòng bit 504 có thể được nhận bởi bộ giải mã 500 và được nhập vào bộ xử lý của bộ giải mã entropy 508, mà nó có thể giải mã entropy dòng bit này thành các hệ số được lượng tử hóa. Các hệ số được lượng tử hóa có thể được cung cấp cho bộ xử lý lượng tử hóa ngược và biến đổi ngược 512, mà nó có thể thực hiện việc lượng tử hóa ngược và biến đổi ngược để tạo ra tín hiệu phần dư, mà nó có thể được cộng vào đầu ra của bộ xử lý bù chuyển động 524 hoặc bộ xử lý dự đoán nội 528 tùy theo chế độ xử lý. Đầu ra của bộ xử lý bù chuyển động 524 và bộ xử lý dự đoán nội 528 có thể bao gồm việc dự đoán khối dựa trên khối được giải mã trước đó hoặc khung UR. Tổng của việc dự đoán và phần dư có thể được xử lý bởi bộ lọc khử khối 516 và được lưu trữ trong bộ đệm khung 520. Đối với khối đã cho, (ví dụ, CU hoặc PU), khi dòng bit 504 báo hiệu theo cách tường minh và/hoặc ngầm định rằng chế độ cập nhật khối của khung UR được cho phép thực hiện, bộ xử lý bù chuyển động 524 có thể cập nhật khung UR, mà nó có thể được chứa trong bộ đệm khung 520, để cập nhật các điểm ảnh (ví dụ, các giá trị độ chói) được đặt cùng chỗ trong khung UR bằng các giá trị điểm ảnh của khối hiện tại.

Trong một số cách thi hành, bộ giải mã 500 có thể bao gồm bộ xử lý cập nhật khối

của khung UR 532 mà nó có thể tạo ra cập nhật khung UR dựa trên khối hiện tại và cung cấp các giá trị điểm ảnh của khung UR cho các quy trình dự đoán liên; điều này có thể được thi hành theo các bước quy trình bất kỳ được mô tả trong sáng chế này. Bộ xử lý cập nhật khối của khung UR 532 có thể tác động trực tiếp đến việc bù chuyển động. Ngoài ra, bộ xử lý cập nhật khung UR 532 có thể nhận thông tin từ bộ xử lý dự đoán nội 528, chẳng hạn như khi khối hiện tại là khối dự đoán nội.

Fig.6 là sơ đồ luồng xử lý minh họa phương án làm ví dụ của quy trình 600 để mã hóa video bằng các cập nhật khối của khung UR với việc báo hiệu tường minh theo một số khía cạnh của sáng chế mà nó có thể làm giảm độ phức tạp mã hóa trong khi làm tăng hiệu quả nén. Tại bước 605, khung video có thể trải qua việc phân đoạn khối ban đầu, ví dụ, nhờ sử dụng sơ đồ phân vùng khối macro được tạo cấu trúc cây mà nó có thể bao gồm việc phân vùng khung hình ảnh thành các CTU và các CU. Tại bước 610, khối có thể được chọn để cập nhật một phần của khung UR. Ví dụ, khối có thể được chọn dựa trên phổ tần suất và/hoặc phép đo của chuyển động bên trong khối khi so sánh với một hoặc nhiều khối được đặt cùng chỗ mà chúng cạnh kề theo thời gian (ví dụ, các khung liên kề theo thời gian, chẳng hạn như trong phạm vi một số lượng khung được xác định trước của khung hiện tại). Việc xác định số đo phổ tần suất có thể bao gồm việc xác định nhờ sử dụng ma trận biến đổi, điều này không giới hạn ở ma trận biến đổi cosin rời rạc. Việc xác định số đo phổ tần suất cho khối có thể bao gồm việc xác định nhờ sử dụng ma trận biến đổi cosin rời rạc được phổ quát hóa. Ví dụ, khi các khối như được mô tả trên đây là các khối 4 x 4 của các điểm ảnh, ma trận biến đổi cosin rời rạc được phổ quát hóa có thể bao gồm ma trận II biến đổi cosin rời rạc được phổ quát hóa có dạng:

$$T = \begin{pmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{pmatrix}$$

trong đó a bằng $\frac{1}{2}$, b bằng $\sqrt{\frac{1}{2}} \cos \frac{\pi}{8}$, và c bằng $\sqrt{\frac{1}{2}} \cos \frac{3\pi}{8}$.

Trong một số cách thi hành, phép xấp xỉ nguyên của ma trận biến đổi có thể được sử dụng, mà nó có thể được sử dụng để được các cách thi hành phần cứng và phần mềm hiệu

quả. Ví dụ, khi các khối như được mô tả trên đây là các khối 4x4 của các điểm ảnh, ma trận biến đổi cosin rời rạc được phổ quát hóa có thể bao gồm ma trận II biến đổi cosin rời rạc được phổ quát hóa có dạng:

$$T_{INT} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}.$$

Đối với khối B_i , phổ tần suất (frequency content) của khối có thể được tính nhờ sử dụng:

$$F_{Bi} = T \times B_i \times T'.$$

trong đó T' là chuyển vị của ma trận truyền cosin T , B_i là khối được biểu diễn dưới dạng ma trận của các giá trị số học tương ứng với các điểm ảnh trong khối này, chẳng hạn như ma trận 4 x 4 thể hiện khối 4 x 4 như được mô tả trên đây, và phép toán $\times$ biểu thị phép nhân ma trận. Việc chọn có thể bao gồm việc nhận biết theo quy tắc số đo rằng khối sẽ được sử dụng để cập nhật một phần của khung UR tại bộ giải mã. Quy tắc số đo có thể bao gồm việc so sánh phép đo về kết cấu và/hoặc chuyển động, bao gồm nhưng không giới hạn ở phép đo bất kỳ của phổ tần suất hoặc dạng tương tự được mô tả trên đây, với ngưỡng được xác định trước và/hoặc ngưỡng được lưu trữ. Theo ví dụ không nhằm mục đích giới hạn, khi F_{Bi} như được định ra trên đây vượt quá một giá trị nào đó, việc cập nhật có thể được kích hoạt, ví dụ do tần số rất cao có thể là không nhận biết được. Theo cách khác hoặc thêm vào đó, việc chọn có thể được thực hiện nhờ sử dụng quy trình và/hoặc bước quy trình bất kỳ được mô tả dưới đây với tham chiếu đến Fig.7. Tại bước 615, khối có thể được mã hóa và được đưa vào trong dòng bit.

Tại bước 620, các tham số cập nhật khối của khung UR tường minh có thể được xác định và được chứa trong dòng bit để báo hiệu rằng chế độ cập nhật khối của khung UR được cho phép thực hiện đối với khối hiện tại. Ví dụ, trường trong PPS hoặc SPS có thể được thiết lập (ví dụ, được cho phép thực hiện). Ví dụ, trường chẳng hạn như trường `UR_BLOCK_UPDATE` có thể được thiết lập để chỉ ra cho khối hiện tại rằng chế độ cập nhật khối của khung UR được cho phép thực hiện.

Fig.7 là sơ đồ luồng xử lý minh họa phương án làm ví dụ của quy trình 700 để mã hóa

video bằng các cập nhật khối của khung UR với việc báo hiệu ngầm định theo một số khía cạnh của sáng chế mà nó có thể làm giảm độ phức tạp mã hóa trong khi làm tăng hiệu quả nén. Tại bước 705, khung video có thể trải qua việc phân đoạn khối ban đầu, ví dụ, nhờ sử dụng sơ đồ phân vùng khối macro được tạo cấu trúc cây mà nó có thể bao gồm việc phân vùng khung hình ảnh thành các CTU và các CU.

Tại bước 710, và vẫn tham chiếu đến Fig.7, khối có thể được chọn để cập nhật một phần của khung UR; khối này có thể được chọn, ví dụ, dựa trên điều kiện cập nhật khung UR ngầm định. Việc chọn có thể bao gồm việc nhận biết theo điều kiện rằng khối sẽ được sử dụng để cập nhật một phần của khung UR tại bộ giải mã. Các điều kiện này có thể bao gồm, ví dụ, khi nào khối hiện tại sử dụng khung UR làm tham chiếu cho việc dự đoán liên và bao gồm việc áp dụng việc bù độ chiếu sáng. Như là một ví dụ khác, các điều kiện có thể liên quan đến việc các lân cận nào của khối sử dụng tham chiếu vector chuyển động UR, chẳng hạn như khi khối mà nó có nhiều khối lân cận với tham chiếu vector chuyển động UR, việc khối hiện tại có được tạo mã liên hay không và việc hai hay hơn hai khối lân cận với khối hiện tại sử dụng vector chuyển động UR cho việc dự đoán liên. Như là một ví dụ khác, nếu khối với tham chiếu phi UR được bao quanh bởi các khối với các tham chiếu UR trên hai hay hơn hai phía thì điều này có thể chỉ ra rằng khối này sẽ được sử dụng để cập nhật một phần của khung UR. Như là một ví dụ khác, nếu khối hiện tại được tạo mã nội và nhiều khối lân cận sử dụng khung UR làm tham chiếu và có vector chuyển động là $(0,0)$ (ví dụ, vector chuyển động không), khối hiện tại có thể được nhận biết là khối sẽ được sử dụng để cập nhật một phần của khung UR. Như là một ví dụ khác nữa, nếu khối hiện tại được tạo mã nội và nhiều khối lân cận sử dụng khung UR làm tham chiếu và có vector chuyển động giống nhau hoặc tương tự nhau (ví dụ, vector chuyển động có thể khác nhau theo lượng nhỏ hơn lượng được xác định trước hoặc định trước), thì khối hiện tại có thể được nhận biết là khối sẽ được sử dụng khi cập nhật khung UR.

Như là một ví dụ khác, và tham chiếu thêm đến Fig.7, trong một số cách thi hành, khối có thể bao gồm đơn vị tạo mã (CU) và điều kiện cho việc xác định có thể bao gồm điều kiện chỉ rõ rằng nếu tất cả các CU lân cận trên ít nhất ba phía đều sử dụng tham chiếu UR thì tất cả các khối của CU hiện tại có thể được chọn. Như là một ví dụ khác, trong một

số cách thi hành, khối có thể bao gồm mức đơn vị cây tạo mã (CTU) và điều kiện có thể bao gồm việc nếu tất cả các CTU lân cận trên ít nhất ba phía đều sử dụng tham chiếu UR thì tất cả các khối của CTU hiện tại mà chúng không sử dụng tham chiếu UR có thể được chọn làm các khối được sử dụng để cập nhật khung UR.

Tại bước 730, khối được chọn có thể được mã hóa và được đưa vào trong dòng bit. Việc mã hóa, ví dụ, có thể bao gồm việc sử dụng các chế độ dự đoán nội và dự đoán liên. Trong một số cách thi hành, các tham số cập nhật khối của khung UR tường minh có thể không được đưa vào trong dòng bit cho khối hiện tại.

Fig.8 là sơ đồ khối hệ thống minh họa phương án làm ví dụ của bộ mã hóa video 800 có khả năng báo hiệu tường minh và/hoặc ngầm định cho các cập nhật khối UR phía bộ giải mã. Bộ mã hóa video ví dụ 800 nhận video đầu vào 804, mà nó có thể đã được phân đoạn ban đầu hoặc chia theo sơ đồ xử lý, chẳng hạn như sơ đồ phân vùng khối macro được tạo cấu trúc cây (ví dụ, cây tứ phân cộng cây nhị phân). Ví dụ về sơ đồ phân vùng khối macro được tạo cấu trúc cây có thể bao gồm việc phân vùng khung hình ảnh thành các phần tử khối lớn được gọi là các đơn vị cây tạo mã (CTU). Trong một số cách thi hành, mỗi CTU có thể còn được phân vùng một hoặc nhiều lần thành một số lượng các khối con được gọi là các đơn vị tạo mã (CU); kết quả của việc phân vùng này có thể bao gồm nhóm các khối con mà chúng có thể được gọi các đơn vị dự đoán (PU). Các đơn vị biến đổi (Transform unit, TU) cũng có thể được sử dụng. “Đơn vị biến đổi,” như được sử dụng trong sáng chế, là đơn vị cơ bản mà tại đó việc biến đổi tần số được áp dụng, mà nó có thể bao gồm, nhưng không giới hạn ở, khối con của CU mà nó là thích hợp cho các hoạt động biến đổi tần số. Để làm ví dụ không nhằm mục đích giới hạn, TU có thể bao gồm khối biến đổi của các mẫu độ chói và hai khối biến đổi tương ứng của các mẫu sắc độ của hình ảnh và các cấu trúc cú pháp được sử dụng để biến đổi các mẫu khối biến đổi, trong đó “khối biến đổi” được định ra là khối $M \times N$ hình chữ nhật của các mẫu sinh ra từ việc biến đổi trong quy trình giải mã.

Vẫn tham chiếu đến Fig.8, bộ mã hóa video ví dụ 800 bao gồm bộ xử lý dự đoán nội 815, bộ xử lý ước lượng/bù chuyển động 812 (cũng được gọi là bộ xử lý dự đoán liên) có khả năng hỗ trợ các cập nhật khối của khung UR tường minh và/hoặc ngầm định tại bộ giải mã, bộ xử lý biến đổi/lượng tử hóa 816, bộ xử lý lượng tử hóa ngược/biến đổi ngược 820,

bộ lọc trong vòng lặp 824, bộ đệm hình ảnh được giải mã 828, và bộ xử lý tạo mã entropy 832. Trong một số cách thi hành, bộ xử lý ước lượng/bù chuyển động 812 có thể xác định đối với khối hiện tại rằng khung UR cần được cập nhật tại bộ giải mã, ví dụ, theo một hoặc nhiều điều kiện trong số các điều kiện được mô tả trên đây đối với Fig.6 và/hoặc Fig.7; bộ xử lý ước lượng/bù chuyển động 812 có thể thiết lập các tham số để báo hiệu tường minh rằng chế độ cập nhật khối của khung UR được cho phép thực hiện. Các tham số dòng bit mà chúng báo hiệu các chế độ cập nhật khối của khung UR có thể là đầu vào cho bộ xử lý tạo mã entropy 832 để được đưa vào trong dòng bit đầu ra 836. Theo cách khác hoặc thêm vào đó, bộ xử lý ước lượng/bù chuyển động 812 có thể cập nhật khung UR của bộ mã hóa bằng khối hiện tại và không có việc báo hiệu cập nhật khối của khung UR tường minh nào có thể được đưa vào trong dòng bit. Các tham số dòng bit có thể là đầu vào cho bộ xử lý tạo mã entropy 832 để được đưa vào trong dòng bit đầu ra 836. Một phần của khung UR phía bộ mã hóa có thể được cập nhật để sử dụng bởi bộ xử lý ước lượng/bù chuyển động 812 để mã hóa các khối bổ sung mà chúng có thể sử dụng khung UR làm tham chiếu cho việc dự đoán liên.

Khi vận hành, và tiếp tục tham chiếu đến Fig.8, đối với mỗi khối của khung của video đầu vào 804, việc liệu sẽ xử lý khối này thông qua việc dự đoán hình ảnh nội hay sử dụng việc ước lượng/bù chuyển động có thể được xác định. Khối có thể được cung cấp cho bộ xử lý dự đoán nội 808 hoặc bộ xử lý ước lượng/bù chuyển động 812. Nếu khối cần được xử lý thông qua việc dự đoán nội thì bộ xử lý dự đoán nội 808 có thể thực hiện việc xử lý để xuất ra bộ dự đoán. Nếu khối sẽ được xử lý thông qua việc ước lượng/bù chuyển động, bộ xử lý ước lượng/bù chuyển động 812 có thể thực hiện việc xử lý bao gồm việc sử dụng khung UR phía bộ mã hóa làm tham chiếu cho việc dự đoán liên, nếu có thể áp dụng được.

Vẫn tham chiếu đến Fig.8, phần dư có thể được tạo ra bằng việc trừ bộ dự đoán khỏi video đầu vào. Phần dư có thể được nhận bởi bộ xử lý biến đổi/lượng tử hóa 816, mà nó có thể thực hiện việc xử lý biến đổi (ví dụ, biến đổi cosin rời rạc (discrete cosine transform, DCT)) để tạo ra các hệ số, mà chúng có thể được lượng tử hóa. Các hệ số được lượng tử hóa và thông tin báo hiệu bất kỳ được liên kết với chúng có thể được cung cấp cho bộ xử lý tạo mã entropy 832 để mã hóa entropy và đưa vào trong dòng bit đầu ra 836. Bộ xử lý

mã hóa entropy 832 có thể hỗ trợ việc mã hóa thông tin báo hiệu liên quan đến các chế độ cập nhật khối của khung UR. Thêm vào đó, các hệ số được lượng tử hóa có thể được cung cấp cho bộ xử lý lượng tử hóa ngược/biến đổi ngược 820, mà nó có thể tái tạo các điểm ảnh, mà chúng có thể được kết hợp với bộ dự đoán và được xử lý bởi bộ lọc trong vòng 824, mà đầu ra của nó có thể được lưu trữ trong bộ đệm hình ảnh được giải mã 828 để sử dụng bởi bộ xử lý ước lượng/bù chuyển động 812 mà nó có khả năng hỗ trợ các cập nhật khối của khung UR tại bộ giải mã.

Mặc dù một vài biến thể đã được mô tả một cách chi tiết trên đây, có thể có các cải biến hoặc các bổ sung khác. Ví dụ, trong một số cách thi hành, các khối hiện tại có thể bao gồm các khối đối xứng bất kỳ (8x8, 16x16, 32x32, 64x64, 128 x 128, và dạng tương tự) cũng như khối bất đối xứng bất kỳ (8x4, 16x8, và dạng tương tự).

Trong một số cách thi hành, cây quyết định tứ phân cộng nhị phân (quadtree plus binary decision tree, QTBT) có thể được thi hành. Trong QTBT, tại mức đơn vị cây tạo mã, các tham số phân vùng của QTBT có thể được dẫn ra theo cách động để thích ứng với các đặc điểm cục bộ mà không cần truyền phí tổn bất kỳ. Tiếp đó, tại mức đơn vị tạo mã, cấu trúc cây quyết định bộ phân loại chung (joint-classifier) có thể loại trừ những lần lặp không cần thiết và kiểm soát rủi ro của việc dự đoán sai. Trong một số cách thi hành, chế độ cập nhật khối của khung UR có thể là khả dụng dưới dạng tùy chọn bổ sung khả dụng tại mọi nút lá của QTBT. Khối được giải mã hiện tại, như được mô tả trên đây, có thể được đưa vào trong QTBT; ví dụ, và không nhằm mục đích giới hạn, khối được giải mã hiện tại có thể là nút không phải lá của QTBT.

Trong một số cách thi hành, thêm các phần tử cú pháp có thể được báo hiệu tại các mức phân cấp khác nhau của dòng bit. Cập nhật khối của khung UR, mà nó có thể bao gồm, nhưng không giới hạn ở, cập nhật khối của khung UR tường minh và/hoặc cập nhật khối của khung UR ngầm định, có thể được cho phép thực hiện đối với toàn bộ chuỗi bằng cách bao gồm cờ cho phép thực hiện được tạo mã trong tập tham số chuỗi (Sequence Parameter Set, SPS). Hơn nữa, cờ CTU có thể được tạo mã tại mức đơn vị cây tạo mã (CTU) để chỉ ra, xem liệu các đơn vị tạo mã (CU) bất kỳ có sử dụng chế độ cập nhật khối của khung UR hay không. Cờ CU có thể được tạo mã để chỉ ra xem liệu đơn vị tạo mã hiện tại có sử dụng

chế độ cập nhật khối của khung UR hay không. Cờ CTU có thể được tạo mã tại mức đơn vị cây tạo mã (CTU) để chỉ ra xem liệu các đơn vị tạo mã (CU) bất kỳ có sử dụng việc báo hiệu cập nhật khối của khung UR tường minh hay không. Cờ CTU có thể được tạo mã tại mức đơn vị cây tạo mã (CTU) để chỉ ra xem liệu các đơn vị tạo mã (CU) bất kỳ có sử dụng việc báo hiệu cập nhật khối của khung UR ngầm định hay không. Mặc dù các phương án được bộc lộ trên đây đã được mô tả liên quan đến các cập nhật cho các khung UR, các phương án được bộc lộ trên đây có thể được áp dụng theo cách thay thế hoặc bổ sung cho các khung, các hình ảnh khác, bao gồm nhưng không giới hạn ở các khung tham chiếu dài hạn.

Đối tượng được mô tả ở đây đem lại nhiều ưu điểm kỹ thuật. Ví dụ, một số cách thi hành đối tượng của sáng chế có thể đem lại việc giải mã các khối nhờ sử dụng khung UR mà nó có thể bao gồm việc cập nhật các phần của khung UR này mà không phải cập nhật toàn bộ khung UR này. Các cách tiếp cận như vậy có thể làm giảm độ phức tạp trong khi làm tăng hiệu quả nén. Hơn nữa, khi báo hiệu theo cách ngầm định các cập nhật khối UR, phí tổn tiêu đề có được giảm xuống, qua đó làm giảm tốc độ bit và cải thiện hoạt động nén.

Cần lưu ý rằng một hoặc nhiều khía cạnh và phương án bất kỳ trong số các khía cạnh và các phương án được mô tả ở đây có thể được thi hành một cách thuận tiện nhờ sử dụng hệ mạch điện tử kỹ thuật số, hệ mạch tích hợp, các mạch tích hợp chuyên dụng (application specific integrated circuit, ASIC) được thiết kế đặc biệt, các mảng cổng lập trình được dạng trường (field programmable gate array, FPGA), phần cứng, phần sụn, phần mềm máy tính, và/hoặc những sự kết hợp của chúng, như được thực hiện và/hoặc thi hành trong một hoặc nhiều máy (ví dụ, một hoặc nhiều thiết bị tính toán mà chúng được sử dụng làm thiết bị tính toán người dùng cho tài liệu điện tử, một hoặc nhiều thiết bị máy chủ, chẳng hạn như máy chủ tài liệu, v.v.) được lập trình theo các nội dung bộc lộ của bản mô tả này, như sẽ trở nên rõ ràng đối với những người có trình độ trung bình trong lĩnh vực máy tính. Các khía cạnh hoặc các dấu hiệu khác nhau này có thể bao gồm cách thi hành trong một hoặc nhiều chương trình và/hoặc phần mềm máy tính mà chúng là có thể thực thi được và/hoặc có thể diễn dịch được trên hệ thống lập trình được bao gồm ít nhất một bộ xử lý lập trình được, mà nó có thể là chuyên dụng hoặc đa năng, được ghép nối để nhận dữ liệu và các lệnh từ, và để truyền

dữ liệu và các lệnh tới, hệ thống lưu trữ, ít nhất một thiết bị đầu vào, và ít nhất một thiết bị đầu ra. Việc tạo mã phần mềm thích hợp hoàn toàn có thể được chuẩn bị bởi các lập trình viên có trình độ dựa trên các nội dung bộc lộ của sáng chế, như sẽ trở nên rõ ràng đối với những người có trình độ trung bình trong lĩnh vực phần mềm. Các khía cạnh và các cách thi hành nêu trên, sử dụng phần mềm và/hoặc các môđun phần mềm, cũng có thể bao gồm phần cứng thích hợp để hỗ trợ trong việc thi hành các lệnh có thể thực thi được bằng máy của phần mềm và/hoặc môđun phần mềm này.

Phần mềm này có thể là sản phẩm chương trình máy tính mà nó sử dụng phương tiện lưu trữ đọc được bằng máy. Phương tiện lưu trữ đọc được bằng máy có thể là phương tiện bất kỳ mà nó có khả năng lưu trữ và/hoặc mã hóa chuỗi các lệnh để thực thi bởi máy (ví dụ, thiết bị tính toán) và khiến cho máy này thực hiện phương pháp và/hoặc phương án bất kỳ trong số các phương pháp và/hoặc các phương án được mô tả ở đây. Các ví dụ về phương tiện lưu trữ đọc được bằng máy bao gồm, nhưng không bị giới hạn ở, đĩa từ, đĩa quang (ví dụ, CD, CD-R, DVD, DVD-R, v.v.), đĩa từ quang, thiết bị nhớ chỉ đọc (read-only memory, “ROM”), thiết bị nhớ truy cập ngẫu nhiên (random access memory, “RAM”), thẻ từ, thẻ quang, thiết bị nhớ trạng thái rắn, EPROM, EEPROM, các thiết bị logic lập trình được (Programmable Logic Device, PLD), và/hoặc những sự kết hợp bất kỳ của chúng. Phương tiện đọc được bằng máy, như được sử dụng ở đây, được dự tính bao gồm một phương tiện đơn cũng như một tập hợp các phương tiện tách biệt về vật lý, ví dụ chẳng hạn như một tập hợp của các đĩa compắc hoặc một hoặc nhiều ổ đĩa cứng kết hợp với bộ nhớ máy tính. Như được sử dụng ở đây, phương tiện lưu trữ đọc được bằng máy không bao gồm các dạng chuyển tiếp của hoạt động truyền tín hiệu.

Phần mềm này cũng có thể bao gồm thông tin (ví dụ, dữ liệu) được mang dưới dạng tín hiệu dữ liệu trên bộ mang dữ liệu, chẳng hạn như sóng mang. Ví dụ, thông tin có thể thực thi bằng máy có thể được đưa vào dưới dạng tín hiệu mang dữ liệu được thực hiện trong bộ mang dữ liệu trong đó tín hiệu này mã hóa chuỗi lệnh, hoặc một phần của nó, để thực thi bởi máy (ví dụ, thiết bị tính toán) và thông tin liên quan bất kỳ (ví dụ, các cấu trúc dữ liệu và dữ liệu) mà nó khiến cho máy này thực hiện phương pháp và/hoặc phương án bất kỳ trong số các phương pháp và/hoặc các phương án được mô tả ở đây.

Các ví dụ về thiết bị tính toán bao gồm, nhưng không bị giới hạn ở, thiết bị đọc sách điện tử, trạm máy tính, máy tính đầu cuối, máy tính chủ, thiết bị cầm tay (ví dụ, máy tính bảng, điện thoại thông minh, v.v.), ứng dụng web, bộ định tuyến mạng, bộ chuyển mạch mạng, cầu nối mạng, máy bất kỳ có khả năng thực thi chuỗi lệnh mà chúng chỉ rõ hoạt động cần được thực hiện bởi máy đó, và những sự kết hợp bất kỳ của chúng. Trong một ví dụ, thiết bị tính toán có thể bao gồm và/hoặc được trang bị trong ki ốt.

Fig.9 thể hiện phép biểu diễn dạng biểu đồ của một phương án về thiết bị tính toán dưới dạng làm ví dụ của hệ thống máy tính 900 mà tập lệnh để khiến cho hệ thống điều khiển thực hiện một hoặc nhiều khía cạnh và/hoặc phương pháp bất kỳ trong số các khía cạnh và/hoặc các phương pháp của sáng chế có thể được thực thi bên trong nó. Việc nhiều thiết bị tính toán có thể được sử dụng để thi hành tập lệnh được tạo cấu hình đặc biệt để khiến để khiến cho một hoặc nhiều thiết bị trong số các thiết bị này thực hiện một hoặc nhiều khía cạnh và/hoặc phương pháp bất kỳ trong số các khía cạnh và/hoặc các phương pháp của sáng chế cũng đã được dự liệu. Hệ thống máy tính 900 bao gồm bộ xử lý 904 và bộ nhớ 908 mà chúng truyền thông với nhau, và với các thành phần khác, thông qua bus 912. Bus 912 có thể bao gồm loại bất kỳ trong một số loại cấu trúc bus bao gồm, nhưng không bị giới hạn ở, bus bộ nhớ, bộ điều khiển bộ nhớ, bus ngoại vi, bus cục bộ, và những sự kết hợp bất kỳ của chúng, sử dụng kiến trúc bất kỳ trong số nhiều kiến trúc bus khác nhau.

Bộ nhớ 908 có thể bao gồm các thành phần khác nhau (ví dụ, các phương tiện đọc được bằng máy) bao gồm, nhưng không bị giới hạn ở, thành phần nhớ truy nhập ngẫu nhiên, thành phần chỉ đọc, và những sự kết hợp bất kỳ của chúng. Trong một ví dụ, hệ thống vào/ra cơ bản 916 (basic input/output system, BIOS), bao gồm các đoạn chương trình cơ bản mà chúng giúp truyền thông tin giữa các phần tử bên trong hệ thống máy tính 900, chẳng hạn như khi khởi động, có thể được lưu trữ trong bộ nhớ 908. Bộ nhớ 908 cũng có thể bao gồm (ví dụ, được lưu trữ trên một hoặc nhiều phương tiện đọc được bằng máy) các lệnh (ví dụ, phần mềm) 920 thực hiện một hoặc nhiều khía cạnh và/hoặc phương pháp bất kỳ trong số các khía cạnh và/hoặc các phương pháp của sáng chế. Trong một ví dụ khác, bộ nhớ 908 có thể còn bao gồm số lượng bất kỳ của các môđun chương trình bao gồm, nhưng không bị

giới hạn ở, hệ điều hành, một hoặc nhiều chương trình ứng dụng, các môđun chương trình khác, dữ liệu chương trình, và những sự kết hợp bất kỳ của chúng.

Hệ thống máy tính 900 cũng có thể bao gồm thiết bị lưu trữ 924. Các ví dụ về thiết bị lưu trữ (ví dụ, thiết bị lưu trữ 924) bao gồm, nhưng không bị giới hạn ở, ổ đĩa cứng, ổ đĩa từ, ổ đĩa quang kết hợp với phương tiện quang, thiết bị nhớ trạng thái rắn, và những sự kết hợp bất kỳ của chúng. Thiết bị lưu trữ 924 có thể được kết nối với bus 912 bởi giao diện thích hợp (không được thể hiện). Các giao diện ví dụ bao gồm, nhưng không bị giới hạn ở, SCSI, gắn kèm công nghệ tiên tiến (advanced technology attachment, ATA), ATA nối tiếp, bus nối tiếp vạn năng (universal serial bus, USB), IEEE 1394 (FIREWIRE), và những sự kết hợp bất kỳ của chúng. Trong một ví dụ, thiết bị lưu trữ 924 (hoặc một hoặc nhiều thành phần của nó) có thể được giao tiếp theo cách tháo ra được với hệ thống máy tính 900 (ví dụ, thông qua bộ kết nối cổng ngoài (không được thể hiện)). Cụ thể, thiết bị lưu trữ 924 và phương tiện đọc được bằng máy được liên kết 928 có thể cung cấp bộ lưu trữ bất khả biến và/hoặc khả biến của các lệnh đọc được bằng máy, các cấu trúc dữ liệu, các môđun chương trình, và/hoặc dữ liệu khác cho hệ thống máy tính 900. Trong một ví dụ, phần mềm 920 có thể nằm, hoàn toàn hoặc một phần, bên trong phương tiện đọc được bằng máy 928. Trong một ví dụ khác, phần mềm 920 có thể nằm, hoàn toàn hoặc một phần, bên trong bộ xử lý 904.

Hệ thống máy tính 900 cũng có thể bao gồm thiết bị đầu vào 932. Trong một ví dụ, người dùng của hệ thống máy tính 900 có thể nhập các mệnh lệnh và/hoặc thông tin khác vào trong hệ thống máy tính 900 thông qua thiết bị đầu vào 932. Các ví dụ về thiết bị đầu vào 932 bao gồm, nhưng không bị giới hạn ở, thiết bị đầu vào chữ và số (ví dụ, bàn phím), thiết bị trò, cần điều khiển, tay cầm điều khiển trò chơi, thiết bị đầu vào âm thanh (ví dụ, micrô, hệ thống đáp ứng giọng nói, v.v.), thiết bị điều khiển con trỏ (ví dụ, con chuột), tấm chạm, máy quét quang học, thiết bị chụp video (ví dụ, camera tĩnh, camera video), màn hình cảm ứng chạm, và những sự kết hợp bất kỳ của chúng. Thiết bị đầu vào 932 có thể được giao tiếp với bus 912 thông qua giao diện bất kỳ trong số nhiều giao diện khác nhau (không được thể hiện) bao gồm, nhưng không bị giới hạn ở, giao diện nối tiếp, giao diện song song, cổng trò chơi, giao diện USB, giao diện FIREWIRE, giao diện trực tiếp với bus

912, và những sự kết hợp bất kỳ của chúng. Thiết bị đầu vào 932 có thể bao gồm giao diện màn hình chạm mà nó có thể là một phần của hoặc tách biệt khỏi bộ hiển thị 936, được mô tả thêm bên dưới. Thiết bị đầu vào 932 có thể được sử dụng làm thiết bị chọn của người dùng để chọn một hoặc nhiều biểu diễn đồ họa trong giao diện đồ họa như được mô tả trên đây.

Người dùng cũng có thể nhập các mệnh lệnh và/hoặc thông tin khác vào hệ thống máy tính 900 thông qua thiết bị lưu trữ 924 (ví dụ, ổ đĩa tháo ra được, ổ đĩa tác động nhanh, v.v.) và/hoặc thiết bị giao diện mạng 940. Thiết bị giao diện mạng, chẳng hạn như thiết bị giao diện mạng 940, có thể được sử dụng để kết nối hệ thống máy tính 900 với một hoặc nhiều mạng trong số các loại mạng khác nhau, chẳng hạn như mạng 944, và một hoặc nhiều thiết bị ở xa 948 được kết nối vào đó. Các ví dụ về thiết bị giao diện mạng bao gồm, nhưng không bị giới hạn ở, các giao diện mạng (ví dụ, các giao diện mạng di động, các LAN), môđem, và sự kết hợp bất kỳ của chúng. Các ví dụ về mạng bao gồm, nhưng không bị giới hạn ở, mạng diện rộng (ví dụ, Internet, mạng doanh nghiệp), mạng cục bộ (ví dụ, mạng được liên kết với văn phòng, tòa nhà, khuôn viên hoặc không gian địa lý tương đối nhỏ khác), mạng điện thoại, mạng dữ liệu được liên kết với nhà cung cấp điện thoại/thoại (ví dụ, mạng dữ liệu và/hoặc thoại của nhà cung cấp truyền thông di động), kết nối trực tiếp giữa hai thiết bị tính toán, và những sự kết hợp bất kỳ của chúng. Mạng, chẳng hạn như mạng 944, có thể sử dụng chế độ truyền thông có dây và/hoặc không dây. Nhìn chung, cấu trúc liên kết mạng bất kỳ có thể được sử dụng. Thông tin (ví dụ, dữ liệu, phần mềm 920, v.v.) có thể được truyền thông tới và/hoặc từ hệ thống máy tính 900 thông qua thiết bị giao diện mạng 940.

Hệ thống máy tính 900 có thể còn bao gồm bộ thích ứng hiển thị video 952 để truyền thông ảnh có thể hiển thị được tới thiết bị hiển thị, chẳng hạn như thiết bị hiển thị 936. Các ví dụ về thiết bị hiển thị bao gồm, nhưng không bị giới hạn ở, bộ hiển thị tinh thể lỏng (liquid crystal display, LCD), đèn tia âm cực (cathode ray tube, CRT), bộ hiển thị plasma, bộ hiển thị điốt phát sáng (light emitting diode, LED), và những sự kết hợp bất kỳ của chúng. Bộ thích ứng hiển thị 952 và thiết bị hiển thị 936 có thể được sử dụng kết hợp với bộ xử lý 904 để cung cấp các biểu diễn đồ họa của các khía cạnh của sáng chế. Ngoài thiết

bị hiển thị, hệ thống máy tính 900 có thể bao gồm một hoặc nhiều thiết bị đầu ra ngoại vi khác bao gồm, nhưng không bị giới hạn ở, loa âm thanh, máy in, và những sự kết hợp bất kỳ của chúng. Các thiết bị đầu ra ngoại vi này có thể được kết nối với bus 912 thông qua giao diện ngoại vi 956. Các ví dụ về giao diện ngoại vi bao gồm, nhưng không bị giới hạn ở, cổng nối tiếp, kết nối USB, kết nối FIREWIRE, kết nối song song, và những sự kết hợp bất kỳ của chúng.

Phần nêu trên là phần mô tả chi tiết của các phương án minh họa của sáng chế. Các cải biến và những sự bổ sung khác nhau có thể được thực hiện mà không tách rời khỏi nguyên lý và phạm vi của sáng chế. Các dấu hiệu của mỗi phương án trong số các phương án khác nhau đã được mô tả trên đây có thể được kết hợp với các dấu hiệu của các phương án khác đã được mô tả theo cách thích hợp để cung cấp nhiều cách kết hợp về dấu hiệu trong các phương án mới có liên quan. Hơn thế nữa, tuy phần nêu trên mô tả một số phương án riêng biệt, những gì đã được mô tả ở đây chỉ đơn thuần mang tính minh họa về việc ứng dụng các nguyên lý của sáng chế. Thêm vào đó, mặc dù các phương pháp cụ thể ở đây có thể được minh họa và/hoặc được mô tả là được thực hiện theo một thứ tự nhất định, việc sắp xếp thứ tự này có khả năng biến thiên cao trong phạm vi trình độ trung bình để đạt được các phương án như được bộc lộ ở đây. Theo đó, phần mô tả này chỉ nhằm để lấy làm ví dụ, và theo cách khác không giới hạn phạm vi của sáng chế.

Trong các phần mô tả trên đây và trong các điểm yêu cầu bảo hộ, các cụm từ như “ít nhất một” hoặc “một hoặc nhiều” có thể xuất hiện theo sau bởi danh sách liên hợp của các phần tử hoặc các dấu hiệu. Thuật ngữ “và/hoặc” cũng có thể xuất hiện trong danh sách của hai hay nhiều phần tử hoặc dấu hiệu. Trừ khi được phủ nhận theo cách khác theo cách không tường minh hoặc tường minh bởi ngữ cảnh mà nó được sử dụng trong đó, cụm từ như vậy nhằm đề cập tới riêng phần tử hoặc dấu hiệu bất kỳ trong số các phần tử hoặc các dấu hiệu được liệt kê, hoặc phần tử hoặc dấu hiệu bất kỳ trong số các phần tử hoặc các dấu hiệu đã nêu kết hợp với phần tử hoặc dấu hiệu bất kỳ trong số các phần tử hoặc các dấu hiệu đã nêu khác. Ví dụ, mỗi cụm từ trong số “ít nhất một thành phần trong số A và B;” “một hoặc nhiều thành phần trong số A và B;” và “A và/hoặc B” nhằm đề cập tới “chỉ riêng A, chỉ riêng B, hoặc A và B cùng với nhau.” Cách diễn giải tương tự cũng là dành cho các

danh sách bao gồm ba hay hơn ba khoản mục. Ví dụ, mỗi cụm từ trong số “ít nhất một thành phần trong số A, B, và C;” “một hoặc nhiều thành phần trong số A, B, và C;” và “A, B, và/hoặc C” nhằm đề cập tới “chỉ riêng A, chỉ riêng B, chỉ riêng C, A và B cùng với nhau, A và C cùng với nhau, B và C cùng với nhau, hoặc A và B và C cùng với nhau.” Thêm vào đó, việc sử dụng thuật ngữ “dựa trên” trên đây và trong các điểm yêu cầu bảo hộ là để có nghĩa là, “dựa ít nhất một phần trên,” sao cho dấu hiệu hoặc phần tử không được nêu cũng có thể được cho phép.

Đối tượng được mô tả ở đây có thể được thực hiện trong các hệ thống, các thiết bị, các phương pháp, và/hoặc các vật phẩm phụ thuộc vào cấu hình mong muốn. Các cách thi hành được nêu trong phần mô tả trên đây không thể hiện tất cả các cách thi hành nhất quán với đối tượng được mô tả ở đây. Thay vào đó, chúng chỉ đơn thuần là một số ví dụ nhất quán với các khía cạnh liên quan đến đối tượng được mô tả. Mặc dù một vài biến thể đã được mô tả một cách chi tiết trên đây, có thể có các cải biến hoặc các bổ sung khác. Cụ thể là, thêm các dấu hiệu và/hoặc các biến thể khác có thể được cung cấp bổ sung cho những gì đã được nêu ra ở đây. Ví dụ, các cách thi hành được mô tả trên đây có thể được hướng tới các tổ hợp và các tổ hợp con khác nhau của các dấu hiệu đã được bộc lộ và/hoặc các tổ hợp và các tổ hợp con của một số dấu hiệu khác đã được bộc lộ trên đây. Thêm vào đó, các luồng logic được thể hiện trên các hình vẽ kèm theo và/hoặc đã được mô tả ở đây không nhất thiết yêu cầu thứ tự cụ thể được thể hiện, hoặc thứ tự tuần tự, để đạt được các kết quả mong muốn. Các cách thi hành khác có thể nằm trong phạm vi của các điểm yêu cầu bảo hộ sau đây.

YÊU CẦU BẢO HỘ

1. Bộ giải mã được tạo cấu hình để:

nhận dòng bit được tạo ra bởi bộ mã hóa và bao gồm thông tin báo hiệu và hình ảnh được tạo mã thứ nhất sẽ được sử dụng làm hình ảnh tham chiếu và hình ảnh được tạo mã thứ hai, hình ảnh được tạo mã thứ hai này bao gồm khối thứ nhất và khối thứ hai, thông tin báo hiệu này bao gồm tập tham số chuỗi (sequence parameter set, SPS) chỉ ra rằng chế độ được cho phép thực hiện trong đó một phần của hình ảnh tham chiếu có thể được cải biến;

giải mã và lưu trữ hình ảnh thứ nhất làm hình ảnh tham chiếu;

nhờ sử dụng chế độ này, cải biến một phần của hình ảnh tham chiếu này chỉ để đáp ứng lại thông tin được tạo ra bởi bộ mã hóa và được chứa trong dòng bit;

sử dụng phần được cải biến này của hình ảnh tham chiếu làm bộ dự đoán cho khối thứ nhất của hình ảnh thứ hai;

giải mã khối thứ nhất bằng cách cộng các điểm ảnh phần dư vào bộ dự đoán này; và

giải mã khối thứ hai của hình ảnh thứ hai mà không sử dụng phần được cải biến của hình ảnh tham chiếu bất kỳ.

2. Bộ giải mã theo điểm 1, trong đó khối thứ nhất và khối thứ hai là các đơn vị tạo mã $N \times N$.

3. Bộ giải mã theo điểm 1, trong đó khối thứ hai của hình ảnh thứ hai là khối được tạo mã nội.

1/9

100

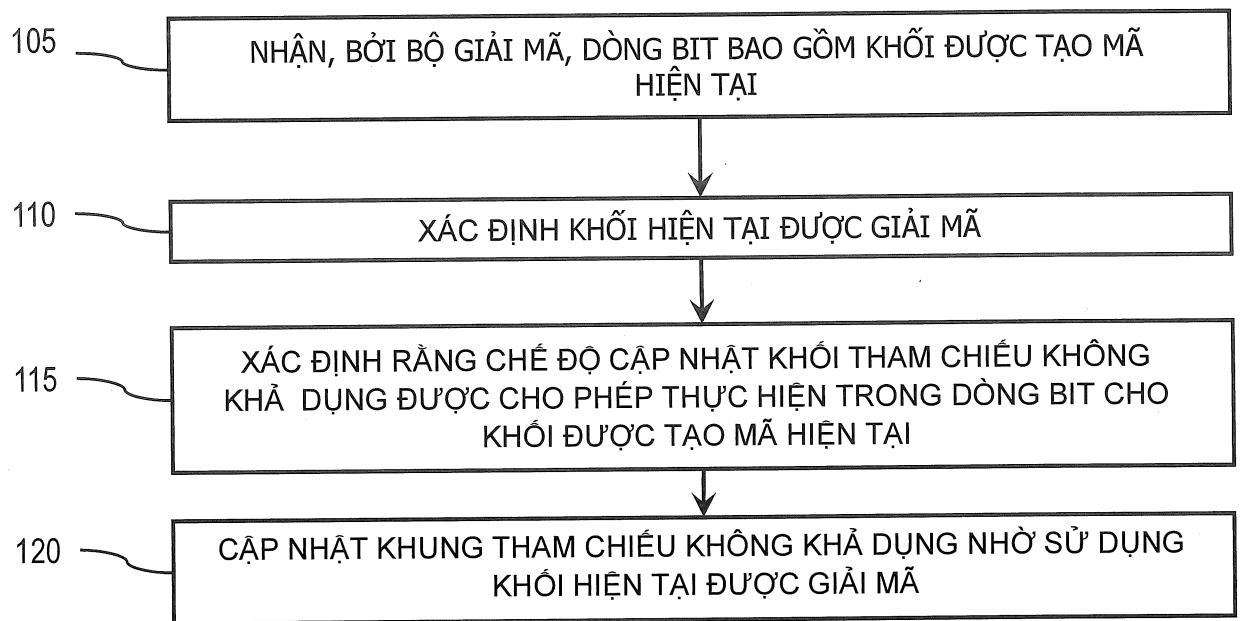
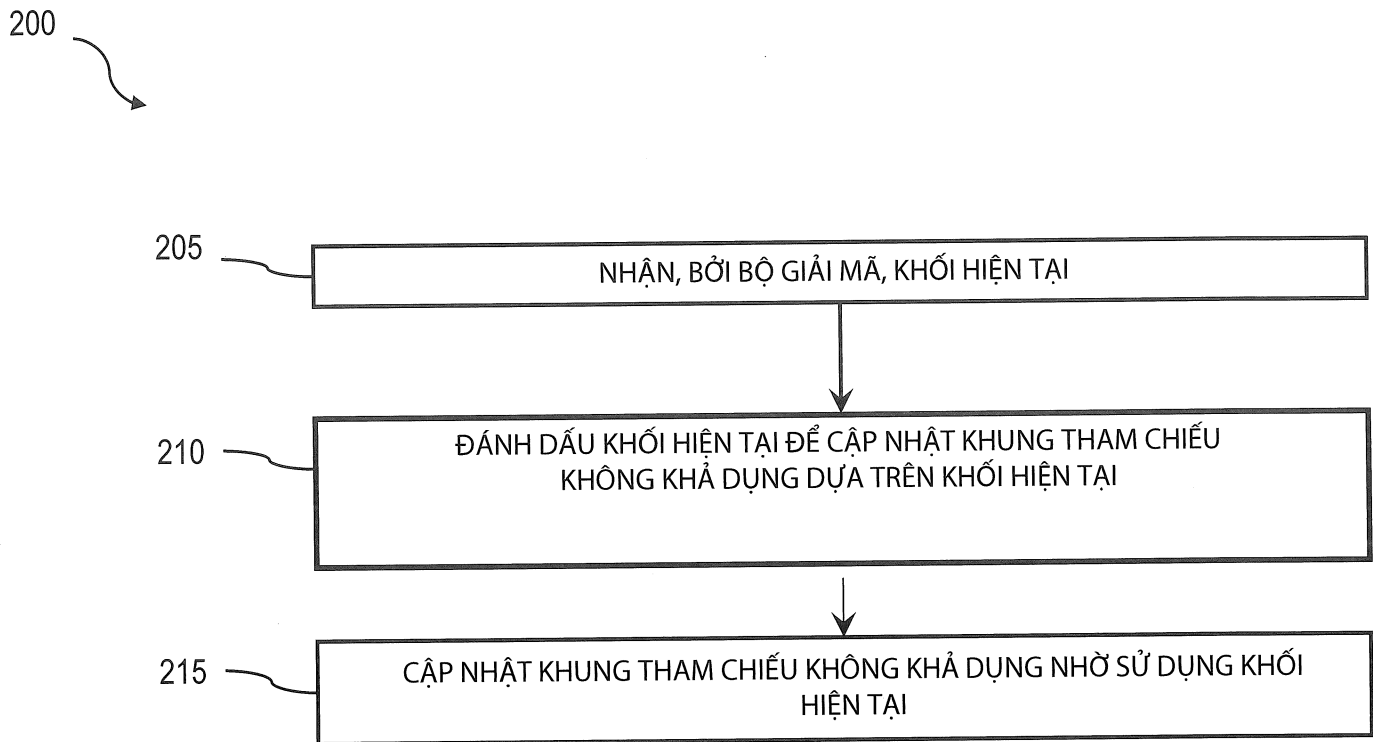


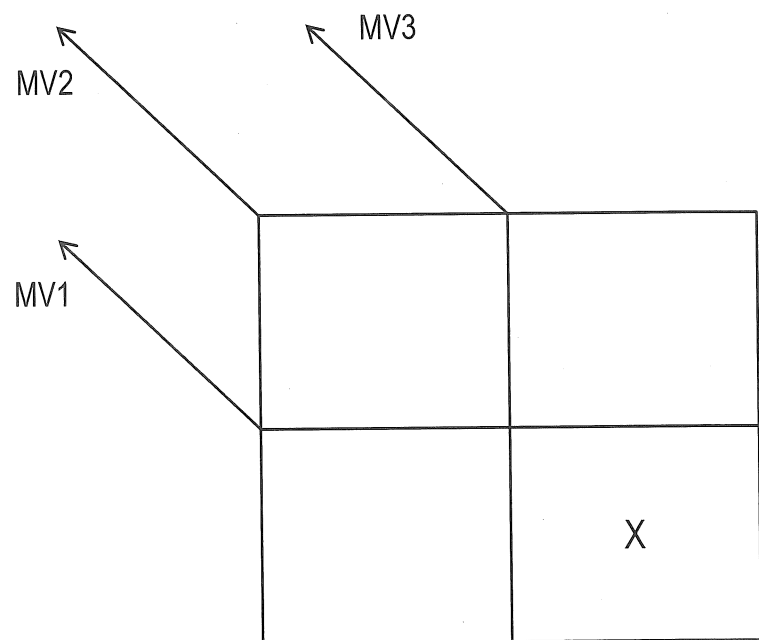
FIG. 1

2/9

**FIG. 2**

3/9

300

**FIG. 3**

4/9

400


L	L	L	L
L	X	X	L
L	X	X	
L			

FIG. 4

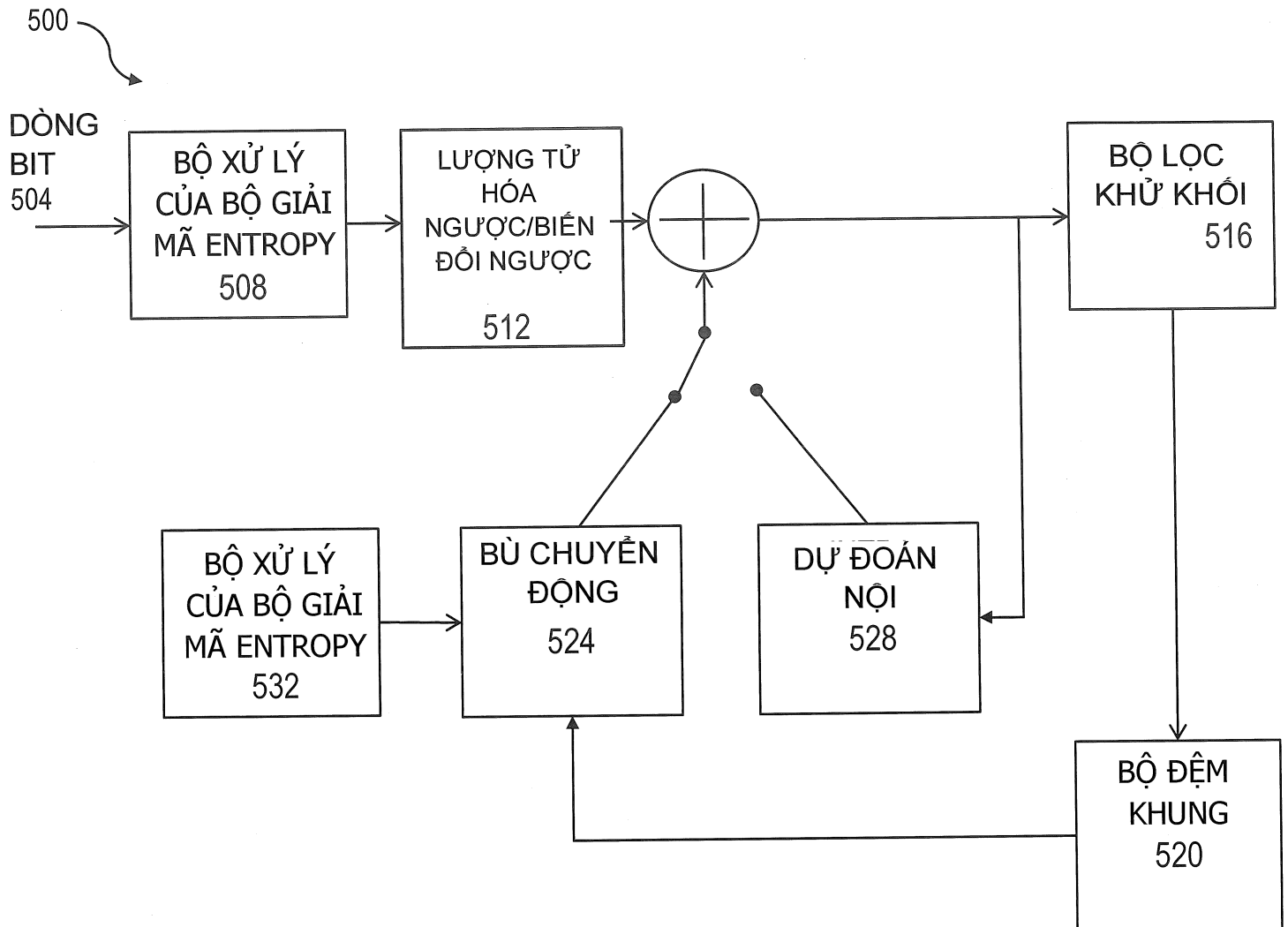


FIG. 5

6/9

600

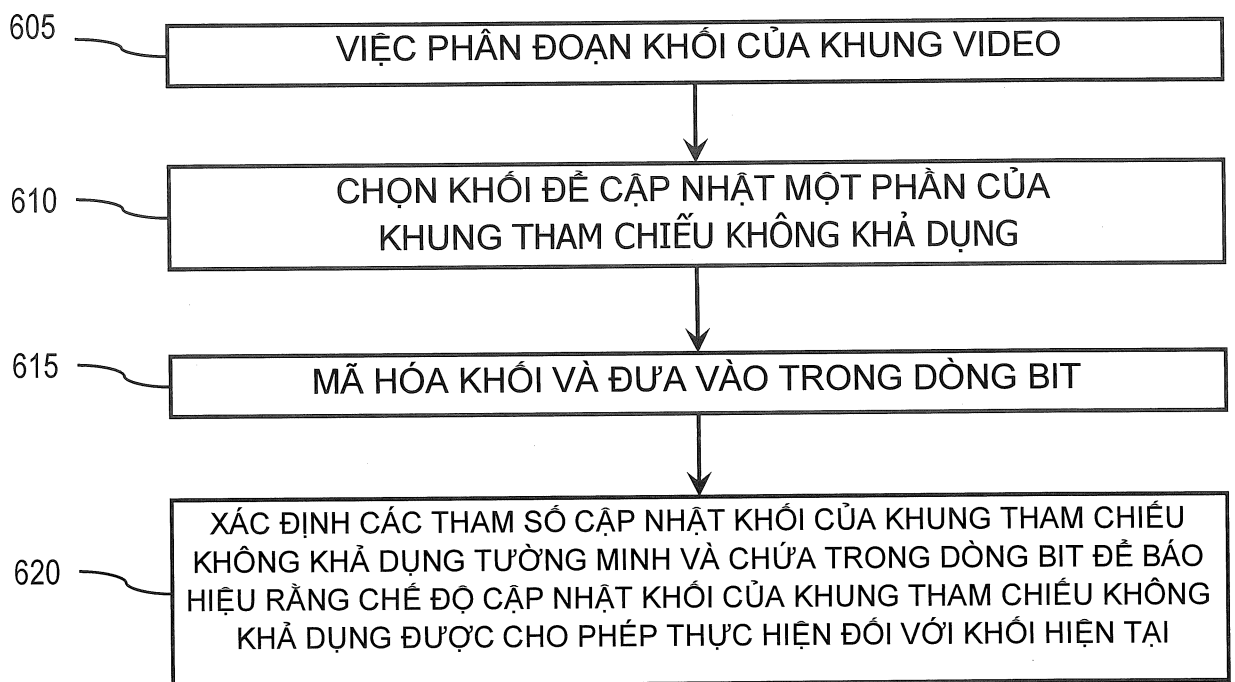
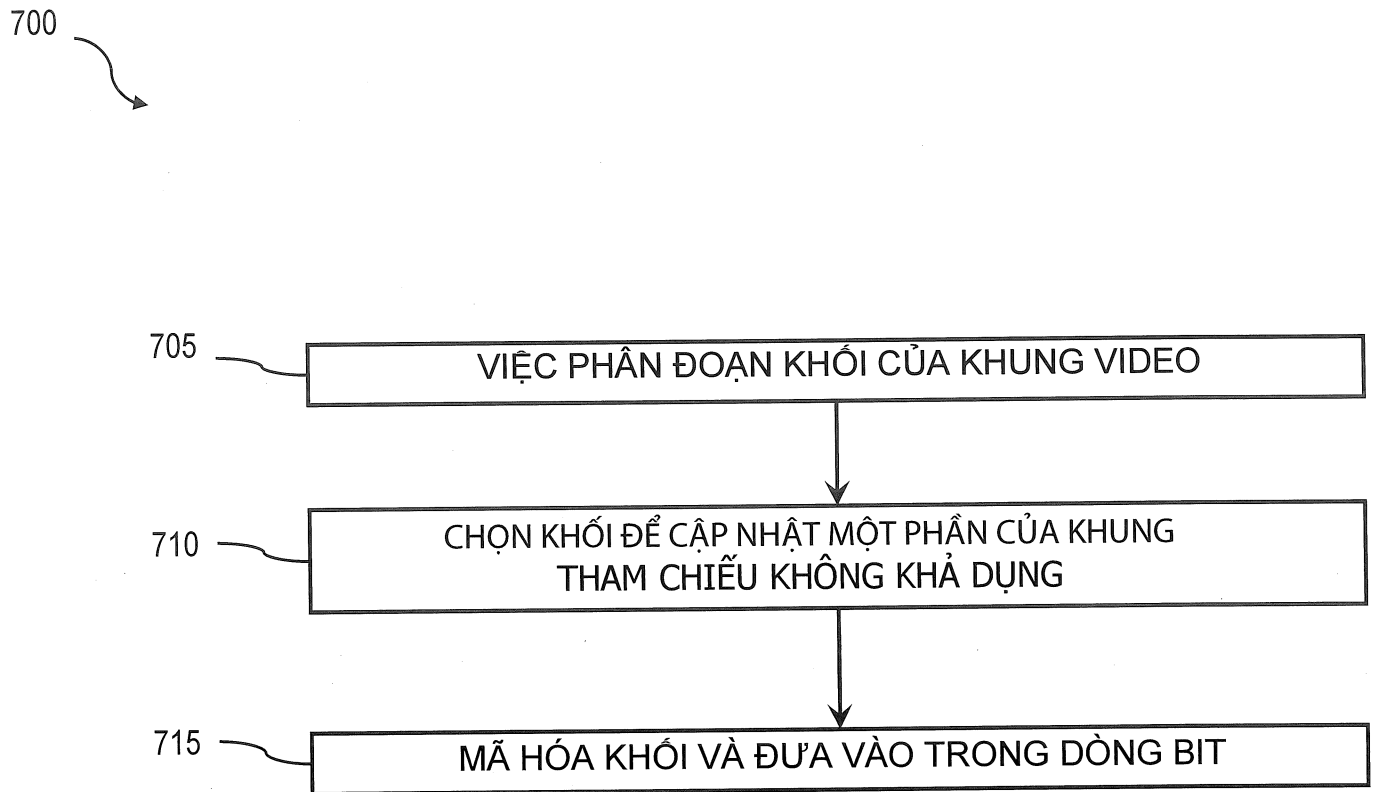


FIG. 6

7/9

**FIG. 7**

8/9

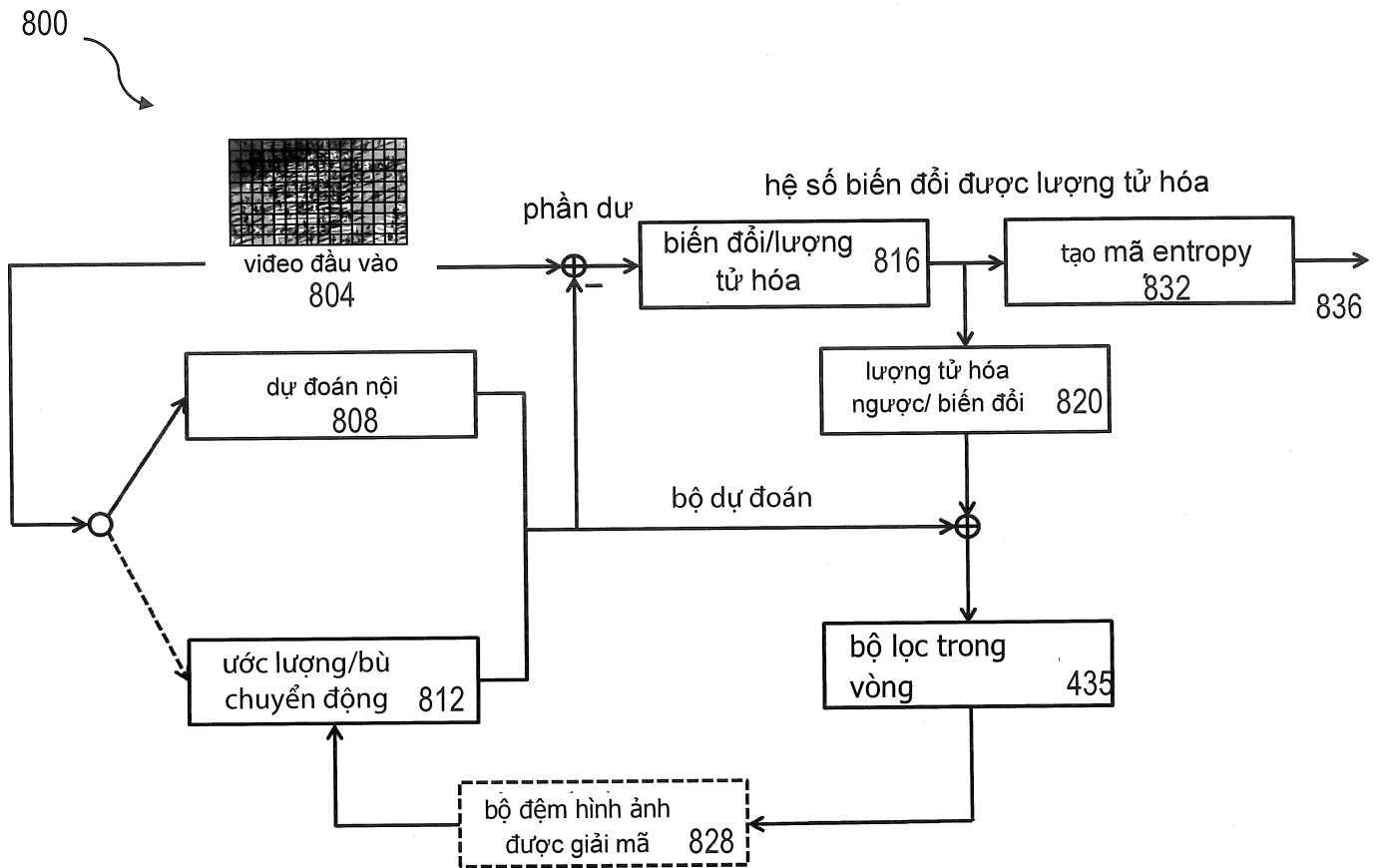


FIG. 8

9/9

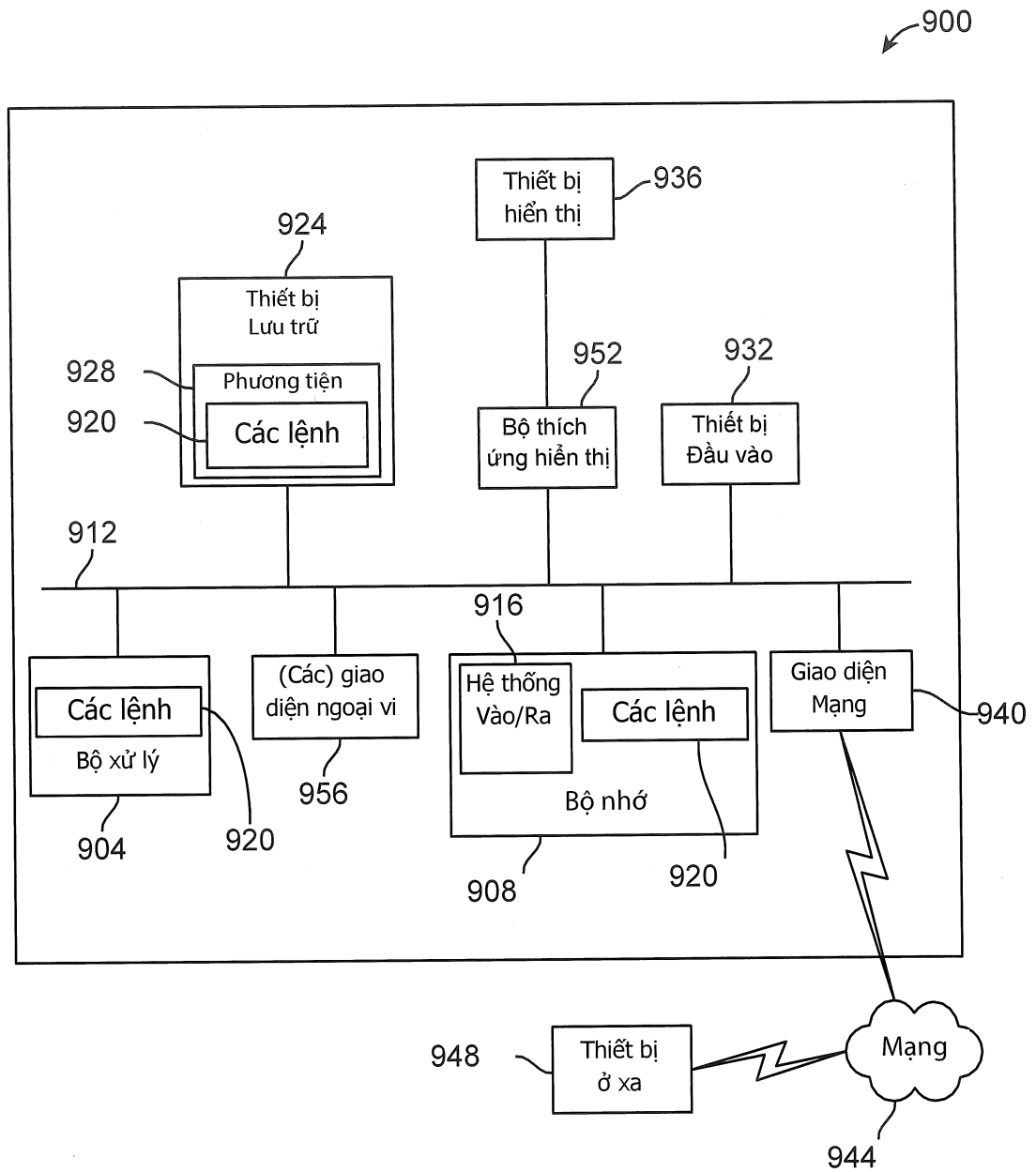


FIG. 9