



(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)



1-0052062

(51)^{2020.01} H04N 19/503; H04N 19/119; H04N 19/122; H04N 19/129; H04N 19/70;
H04N 19/18; H04N 19/50; H04N 19/61;
H04N 19/103; H04N 19/176

(13) B

(21) 1-2021-02116

(22) 20/09/2019

(86) PCT/KR2019/012291 20/09/2019

(87) WO2020/060328 A1 26/03/2020

(30) 10-2018-0114345 21/09/2018 KR; 10-2018-0114346 21/09/2018 KR; 10-2018-0114347 21/09/2018 KR; 10-2018-0114350 21/09/2018 KR; 10-2019-0022754 26/02/2019 KR

(45) 25/09/2025 450

(43) 25/06/2021 399A

(73) GUANGDONG OPPO MOBILE TELECOMMUNICATIONS CORP., LTD. (CN)
No.18, Haibin Road, Wusha, Chang'an, Dongguan, Guangdong 523860, China

(72) LEE, Bae Keun (KR).

(74) Công ty TNHH Dịch vụ Sở hữu trí tuệ KENFOX (KENFOX IP SERVICE CO.,LTD.)

(54) PHƯƠNG PHÁP GIẢI MÃ VIDEO, PHƯƠNG PHÁP MÃ HÓA VIDEO, THIẾT BỊ GIẢI MÃ VIDEO, THIẾT BỊ MÃ HÓA VIDEO VÀ PHƯƠNG TIỆN LƯU TRỮ CÓ THỂ ĐỌC ĐƯỢC BẰNG MÁY TÍNH

(21) 1-2021-02116

(57) Sáng chế đề cập đến phương pháp giải mã video, phương pháp mã hóa video, thiết bị giải mã video và thiết bị mã hóa video, trong đó phương pháp giải mã video này bao gồm các bước: tạo ra danh mục ứng viên hợp nhất của khối hiện tại; định rõ bất kỳ một trong số nhiều ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất; dựa trên vector hạt affin thứ nhất và vector hạt affin thứ hai của ứng viên hợp nhất được định rõ, đem lại vector hạt affin thứ nhất và vector hạt affin thứ hai của khối hiện tại; sử dụng vector hạt affin thứ nhất và vector hạt affin thứ hai của khối hiện tại, đem lại vector affin liên quan đến khối con trong khối hiện tại; và dựa trên vector affin, thực hiện dự báo bù chuyển động liên quan đến khối con.

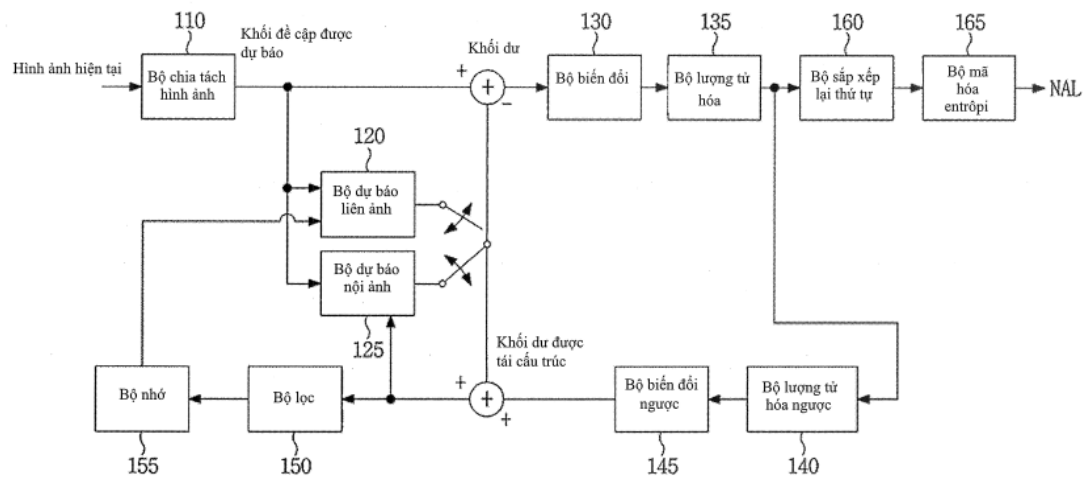


FIG. 1

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp giải mã video, phương pháp mã hóa video, thiết bị giải mã video và thiết bị mã hóa video.

Tình trạng kỹ thuật của sáng chế

Do xu hướng sử dụng băng hiển thị ngày càng trở nên lớn hơn, các dịch vụ video độ nét cao được yêu cầu. Vấn đề lớn nhất của các dịch vụ video độ nét cao là khối lượng dữ liệu tăng đáng kể, và để giải quyết vấn đề này, nghiên cứu được tích cực thực hiện để cải thiện tốc độ nén video. Theo một ví dụ đại diện, nhóm hợp tác chung về mã hóa video (Joint Collaborative Team on Video Coding - JCT-VC) đã được thành lập trong nhóm các chuyên gia mã hóa video (Video Coding Experts Group - VCEG) thuộc ban tiêu chuẩn hoá viễn thông của liên minh viễn thông quốc tế (International Telecommunication Union-Telecommunication - ITU-T) và nhóm các chuyên gia hình ảnh chuyển động (Motion Picture Experts Group - MPEG) vào năm 2009. JCT-VC đã đề xuất mã hóa video hiệu suất cao (High Efficiency Video Coding - HEVC), đây là tiêu chuẩn nén video có hiệu năng nén cao gấp khoảng hai lần hiệu năng nén của H.264/AVC, và đã được phê chuẩn vào ngày 25/01/2013. Với sự phát triển nhanh chóng trong các dịch vụ video độ nét cao, hiệu năng của HEVC dần bộc lộ các nhược điểm của tiêu chuẩn này.

Bản chất kỹ thuật của sáng chế

Vấn đề kỹ thuật

Mục đích của sáng chế này là đề xuất phương pháp và thiết bị mã hóa hoặc giải mã hệ số còn lại khi mã hóa hoặc giải mã tín hiệu video.

Mục đích của sáng chế này là đề xuất phương pháp và thiết bị mã hóa hoặc giải mã hệ số còn lại sử dụng cờ để so sánh kích cỡ của hệ số còn lại với trị số ngưỡng khi mã hóa hoặc giải mã tín hiệu video.

Mục đích của sáng chế này là đề xuất phương pháp và thiết bị mã hóa hoặc giải mã hệ số còn lại sử dụng cờ biểu thị việc liệu hệ số còn lại là số chẵn hoặc số lẻ khi mã

hóa hoặc giải mã tín hiệu video.

Các vấn đề kỹ thuật được ý định được giải quyết bởi sáng chế không bị giới hạn ở các vấn đề nêu trên, và các vấn đề kỹ thuật khác mà không được mô tả ở đây nên được hiểu rõ bởi người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng từ phần mô tả sau đây.

Giải pháp kỹ thuật

Theo một khía cạnh rộng, sáng chế đề xuất phương pháp giải mã video, phương pháp bao gồm: phân tách cờ khác không biểu thị việc liệu hệ số còn lại là khác không từ dòng bit; phân tách thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại từ dòng bit khi cờ khác không biểu thị rằng hệ số còn lại là không khác không; và xác định trị số tuyệt đối của hệ số còn lại dựa trên thông tin trị số tuyệt đối, trong đó thông tin trị số tuyệt đối bao gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn lại là lớn hơn trị số thứ nhất, và cờ chẵn lẻ còn được phân tách từ dòng bit chỉ khi hệ số còn lại là lớn hơn trị số thứ nhất.

Theo một khía cạnh rộng khác, sáng chế đề xuất phương pháp mã hóa video, phương pháp bao gồm: mã hóa cờ khác không biểu thị việc liệu hệ số còn lại là khác không; và mã hóa thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại khi hệ số còn lại là không khác không, trong đó thông tin trị số tuyệt đối gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn lại là lớn hơn trị số thứ nhất, và cờ chẵn lẻ cho hệ số còn lại còn được mã hóa chỉ khi hệ số còn lại là lớn hơn trị số thứ nhất.

Theo một số phương án, cờ chẵn lẻ có thể biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ.

Theo một số phương án, khi hệ số còn lại là lớn hơn trị số thứ nhất, cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm 1 lớn hơn trị số thứ hai còn có thể được phân tách.

Theo một số phương án, khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ hai, hệ số còn lại có thể được xác định là $2N$ hoặc $2N+1$ phụ thuộc vào trị số của cờ chẵn lẻ.

Theo một số phương án, khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai, cờ so sánh hệ số còn lại được điều chỉnh thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba còn có thể được phân tách.

Theo một số phương án, khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai,

thông tin trị số còn lại còn có thể được phân tách, và thông tin trị số còn lại có thể là trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn lại được điều chỉnh.

Các đặc tính được tóm lược trên đây liên quan đến sáng chế này chỉ là các khía cạnh dùng làm ví dụ của phần mô tả chi tiết của sáng chế này được mô tả dưới đây và không giới hạn phạm vi của sáng chế này.

[Ưu điểm của sáng chế]

Theo sáng chế này, có thể mã hóa hoặc giải mã một cách hiệu quả hệ số còn lại.

Theo sáng chế này, có thể mã hóa hoặc giải mã một cách hiệu quả hệ số còn lại bằng cách sử dụng cờ để so sánh kích cỡ của hệ số còn lại với trị số ngưỡng.

Theo sáng chế này, có thể mã hóa hoặc giải mã một cách hiệu quả hệ số còn lại bằng cách sử dụng cờ biểu thị việc liệu hệ số còn lại là số chẵn hoặc số lẻ.

Ưu điểm của sáng chế của sáng chế không bị giới hạn ở các tác dụng nêu trên, và còn các ưu điểm khác của sáng chế mà không được mô tả ở đây nên được hiểu rõ bởi người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng từ phần mô tả sau đây.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ khối của bộ mã hóa video theo một phương án của sáng chế này.

Fig.2 là sơ đồ khối của bộ giải mã video theo một phương án của sáng chế này.

Fig.3 là sơ đồ thể hiện đơn vị cây mã hóa cơ bản theo một phương án của sáng chế này.

Fig.4 là sơ đồ thể hiện các kiểu chia tách khác nhau của khối mã hóa.

Fig.5 là sơ đồ minh họa khía cạnh chia tách của đơn vị cây mã hóa.

Fig.6 thể hiện ví dụ trong đó khối nhỏ hơn đơn vị cây mã hóa của kích cỡ được xác định trước xảy ra tại mép hình ảnh.

Fig.7 là sơ đồ thể hiện ví dụ trong đó chia tách cây tứ phân được thực hiện trên khối mép ranh giới không điển hình.

Fig.8 là sơ đồ thể hiện ví dụ trong đó chia tách cây tứ phân được thực hiện trên khối gần kề mép hình ảnh.

Fig.9 là sơ đồ thể hiện khía cạnh chia tách của khối gần kề mép hình ảnh.

Fig.10 là sơ đồ thể hiện khía cạnh mã hóa của khối gần kề mép hình ảnh.

Fig.11 là lưu đồ của phương pháp dự báo liên ảnh theo một phương án của sáng chế này.

Fig.12 là lưu đồ của quy trình suy ra thông tin chuyển động của khối hiện tại trong

phương thức hợp nhất.

Fig.13 là sơ đồ minh họa các khối ứng viên được sử dụng để suy ra ứng viên hợp nhất.

Fig.14 là sơ đồ thể hiện các vị trí của các mẫu tham chiếu.

Fig.15 là sơ đồ minh họa các khối ứng viên được sử dụng để suy ra ứng viên hợp nhất.

Fig.16 là sơ đồ thể hiện ví dụ trong đó vị trí của mẫu tham chiếu bị thay đổi.

Fig.17 là sơ đồ thể hiện ví dụ trong đó vị trí của mẫu tham chiếu bị thay đổi.

Fig.18 là sơ đồ để mô tả khía cạnh cập nhật của bảng thông tin chuyển động liên vùng.

Fig.19 là sơ đồ thể hiện khía cạnh cập nhật của bảng ứng viên hợp nhất liên vùng.

Fig.20 là sơ đồ thể hiện ví dụ trong đó chỉ số của ứng viên hợp nhất liên vùng được cập nhật được lưu trữ trước.

Fig.21 là sơ đồ thể hiện vị trí của khối con đại diện.

Fig.22 thể hiện ví dụ trong đó bảng thông tin chuyển động liên vùng được tạo ra cho mỗi phương thức dự báo liên ảnh.

Fig.23 là sơ đồ thể hiện ví dụ trong đó ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động dài hạn được thêm vào danh mục ứng viên hợp nhất.

Fig.24 là sơ đồ thể hiện ví dụ trong đó việc kiểm tra dư thừa được thực hiện trên chỉ một số ứng viên hợp nhất.

Fig.25 là sơ đồ thể hiện ví dụ trong đó việc kiểm tra dư thừa trên ứng viên hợp nhất cụ thể bị bỏ qua.

Fig.26 là sơ đồ thể hiện ví dụ trong đó khối ứng viên được đưa vào khu vực hợp nhất song song giống như khu vực của khối hiện tại được đặt là không sẵn có là ứng viên hợp nhất.

Fig.27 là sơ đồ thể hiện bảng thông tin chuyển động tạm thời.

Fig.28 là sơ đồ thể hiện ví dụ về việc hợp nhất bảng thông tin chuyển động tạm thời với bảng thông tin chuyển động liên vùng.

Fig.29 là lưu đồ của phương pháp dự báo nội ảnh theo một phương án của sáng chế này.

Fig.30 là sơ đồ thể hiện các mẫu tham chiếu được đưa vào mỗi đường mẫu tham chiếu.

Fig.31 là sơ đồ thể hiện các phương thức dự báo nội ảnh.

Fig.32 và Fig.33 là các sơ đồ thể hiện ví dụ của mảng một chiều trong đó các mẫu tham chiếu được bố trí trong một đường.

Fig.34 là sơ đồ minh họa các góc được tạo ra giữa đường thẳng song song với trục x và các phương thức dự báo nội ảnh định hướng.

Fig.35 là sơ đồ thể hiện khía cạnh trong đó mẫu dự báo thu được khi khôi hiện tại không phải là hình vuông.

Fig.36 là sơ đồ thể hiện các phương thức dự báo nội ảnh góc rộng.

Fig.37 là sơ đồ thể hiện khía cạnh ứng dụng của PDPC.

Fig.38 và Fig.39 là các sơ đồ thể hiện khối con cần phải chịu sự biến đổi thứ hai.

Fig.40 là sơ đồ để mô tả ví dụ trong đó kiểu biến đổi của khối hiện tại được xác định.

Fig.41 là lưu đồ thể hiện phương pháp mã hóa hệ số còn lại.

Fig.42 và Fig.43 là các sơ đồ thể hiện thứ tự sắp xếp của các hệ số còn lại cho mỗi thứ tự quét.

Fig.44 thể hiện ví dụ trong đó vị trí của hệ số khác không cuối cùng được mã hóa.

Fig.45 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Fig.46 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Fig.47 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Fig.48 là lưu đồ thể hiện quy trình xác định cường độ tạo khối.

Fig.49 thể hiện các ứng viên bộ lọc được xác định trước.

Mô tả chi tiết sáng chế

Trong phần dưới đây, các phương án của sáng chế này sẽ được mô tả chi tiết kết hợp với các hình vẽ kèm theo.

Video được mã hóa và được giải mã trong các đơn vị của các khối. Theo một ví dụ, quy trình mã hóa và giải mã chẳng hạn như biến đổi, lượng tử hóa, dự báo, lọc trong vòng lặp, hoặc tái cấu trúc có thể được thực hiện trên khối mã hóa, khối biến đổi, hoặc khối dự báo.

Trong phần dưới đây, khối cần được mã hóa hoặc được giải mã sẽ được gọi là “khối hiện tại”. Ví dụ như, khối hiện tại có thể biểu thị khối mã hóa, khối biến đổi, hoặc khối dự báo phụ thuộc vào bước hiện tại để xử lý mã hóa hoặc giải mã.

Ngoài ra, thuật ngữ “bộ phận” được sử dụng ở đây có thể được hiểu là biểu thị bộ phận cơ bản để thực hiện quy trình mã hóa và giải mã cụ thể, và thuật ngữ “khối” có thể

được hiểu là biểu thị mảng mẫu của kích cỡ được xác định trước. Nếu không được định rõ, các thuật ngữ “khối” và “bộ phận” có thể được sử dụng hoán đổi cho nhau. Theo một ví dụ, trong các phương án sau đây, khối mã hóa và bộ phận mã hóa có thể được hiểu là có các ý nghĩa tương đương.

Fig.1 là sơ đồ khối của bộ mã hóa video theo một phương án của sáng chế này.

Như được thể hiện trên Fig.1, thiết bị mã hóa video 100 có thể gồm bộ chia tách hình ảnh 110, các bộ dự báo 120 và 125, bộ biến đổi 130, bộ lượng tử hóa 135, bộ sắp xếp lại thứ tự 160, bộ mã hóa entropi 165, bộ lượng tử hóa ngược 140, bộ biến đổi ngược 145, bộ lọc 150, và bộ nhớ 155.

Các phần tử trên Fig.1 được thể hiện độc lập để thể hiện các chức năng đặc trưng khác nhau trong thiết bị mã hóa video, và mỗi trong số các phần tử không có nghĩa là cần được tạo cấu hình trong bộ phận phân cứng riêng biệt hoặc là một bộ phận phần mềm. Nói cách khác, các phần tử được sắp xếp một cách độc lập để thuận tiện cho việc mô tả. Để thực hiện các chức năng, ít nhất hai các phần tử có thể được kết hợp thành một phần tử, hoặc một phần tử có thể được chia thành nhiều phần tử. Trong trường hợp này, một phương án cho sự kết hợp của các phần tử và một phương án để phân chia phần tử được bao hàm trong phạm vi của sáng chế này mà không nằm ngoài bản chất của sáng chế này.

Ngoài ra, một số trong số các phần tử có thể không là các phần tử thiết yếu để thực hiện các chức năng thiết yếu trong sáng chế này và có thể chỉ là các phần tử tùy chọn để cải thiện hiệu năng. Sáng chế này có thể được thực hiện bằng cách chỉ gồm các phần tử cần thiết để thực hiện bản chất của sáng chế này hơn là các phần tử được sử dụng chỉ để cải thiện hiệu năng. Thậm chí kết cấu chỉ gồm các phần tử thiết yếu hơn là các phần tử tùy chọn được sử dụng chỉ để cải thiện hiệu năng được bao hàm trong phạm vi của sáng chế này.

Bộ chia tách hình ảnh 110 có thể chia tách hình ảnh đầu vào thành ít nhất một bộ phận xử lý. Trong trường hợp này, bộ phận xử lý có thể là bộ phận dự báo (Prediction Unit - PU), bộ phận biến đổi (Transform Unit - TU), hoặc bộ phận mã hóa (Coding Unit - CU). Bộ chia tách hình ảnh 110 có thể chia tách một hình ảnh thành nhiều tổ hợp các bộ phận mã hóa, bộ phận dự báo, và bộ phận biến đổi và có thể chọn một tổ hợp của các bộ phận mã hóa, các bộ phận dự báo, và các bộ phận biến đổi theo tiêu chuẩn được xác định trước (ví dụ như, hàm chi phí) để mã hóa hình ảnh.

Ví dụ như, một hình ảnh có thể được chia tách thành nhiều bộ phận mã hóa. Cấu trúc cây đệ quy, chẳng hạn như cấu trúc cây tứ phân, có thể được sử dụng để chia tách

hình ảnh thành các bộ phận mã hóa. Bộ phận mã hóa được phân tách thành các bộ phận mã hóa khác bằng cách sử dụng một hình ảnh hoặc bộ phận mã hóa lớn nhất làm nghiệm có thể có một số nút con tương ứng với số lượng các bộ phận mã hóa bị chia tách. Bộ phận mã hóa không còn bị chia tách nữa do giới hạn được xác định trước có tác dụng là nút lá. Tức là, khi giả sử rằng chỉ có thể chia tách hình vuông cho một bộ phận mã hóa, một bộ phận mã hóa có thể được chia tách lên đến bốn bộ phận mã hóa khác.

Trong các phương án sau đây của sáng chế này, bộ phận mã hóa có thể đề cập đến bộ phận được tạo cấu hình để thực hiện mã hóa hoặc bộ phận được tạo cấu hình để thực hiện giải mã.

Một bộ phận mã hóa có thể được chia tách thành ít nhất một hoặc nhiều bộ phận dự báo có cùng kích cỡ có dạng hình vuông hoặc hình chữ nhật và có thể được chia tách thành các bộ phận dự báo sao cho một trong số các bộ phận dự báo khác với một bộ phận dự báo khác về hình dạng và/hoặc kích cỡ.

Khi bộ phận dự báo phải chịu dự báo nội ảnh dựa trên bộ phận mã hóa được tạo ra và bộ phận mã hóa không phải là bộ phận mã hóa tối thiểu, dự báo nội ảnh có thể được thực hiện mà không chia tách bộ phận mã hóa thành nhiều bộ phận dự báo $N \times N$.

Các bộ dự báo 120 và 125 có thể gồm bộ phận dự báo liên ảnh 120 được tạo cấu hình để thực hiện dự báo liên ảnh và bộ dự báo nội ảnh 125 được tạo cấu hình để thực hiện dự báo nội ảnh. Các bộ dự báo 120 và 125 có thể xác định việc liệu thực hiện dự báo nội ảnh hay sử dụng dự báo liên ảnh trên các bộ phận dự báo và có thể xác định thông tin chi tiết (ví dụ như, phương thức dự báo nội ảnh, vectơ chuyển động, hình ảnh tham chiếu, và nội dung tương tự) tương ứng với mỗi phương pháp dự báo. Trong trường hợp này, bộ phận xử lý trong đó việc dự báo được thực hiện có thể là khác nhau từ bộ phận xử lý trong đó phương pháp dự báo và các chi tiết cụ thể được xác định. Ví dụ như, phương pháp dự báo, phương thức dự báo, và phương thức tương tự có thể được xác định bởi bộ phận dự báo, và việc dự báo có thể được thực hiện bởi bộ phận biến đổi. Trị số còn lại (khối còn lại) giữa được tạo ra khỏi dự báo và khối gốc có thể được nhập vào bộ biến đổi 130. Ngoài ra, thông tin vectơ chuyển động, thông tin phương thức dự báo, và thông tin tương tự, được sử dụng cho việc dự báo, ngoài trị số còn lại có thể được mã hóa bởi bộ mã hóa entropi 165 và được chuyển giao đến bộ giải mã. Khi phương thức mã hóa cụ thể được sử dụng, khối gốc có thể được mã hóa nguyên vẹn và được truyền đến bộ giải mã mà không có các bộ dự báo 120 và 125 tạo ra khỏi dự báo.

Bộ phận dự báo liên ảnh 120 có thể dự báo bộ phận dự báo dựa trên thông tin về ít

nhất một trong số hình ảnh đứng trước hình ảnh hiện tại hoặc hình ảnh theo sau hình ảnh hiện tại, và trong một số trường hợp, bộ phận dự báo có thể được dự báo dựa trên thông tin về khu vực nào đó của hình ảnh hiện tại mà ở đó việc mã hóa được hoàn thành. Bộ phận dự báo liên ảnh 120 có thể gồm bộ nội suy hình ảnh tham chiếu, bộ dự báo chuyển động, và bộ bù chuyển động.

Bộ nội suy hình ảnh tham chiếu có thể nhận thông tin hình ảnh tham chiếu từ bộ nhớ 155 và có thể tạo ra thông tin về các điểm ảnh nhỏ hơn hoặc bằng các điểm ảnh số nguyên từ hình ảnh tham chiếu. Trong trường hợp các điểm ảnh độ sáng, bộ lọc nội suy 8 nhánh dựa trên DCT có các hệ số bộ lọc khác nhau có thể được sử dụng để tạo ra thông tin về các điểm ảnh nhỏ hơn hoặc bằng các điểm ảnh số nguyên trong các đơn vị của $1/4$ các điểm ảnh. Trong trường hợp các tín hiệu màu, bộ lọc nội suy 4 nhánh dựa trên DCT có các hệ số bộ lọc khác nhau có thể được sử dụng để tạo ra thông tin về các điểm ảnh nhỏ hơn hoặc bằng các điểm ảnh số nguyên trong các đơn vị của $1/8$ các điểm ảnh.

Bộ dự báo chuyển động có thể thực hiện việc dự báo chuyển động dựa trên hình ảnh tham chiếu được nội suy bởi bộ nội suy hình ảnh tham chiếu. Với tư cách là phương pháp tính toán vectơ chuyển động, các phương pháp khác nhau chẳng hạn như thuật toán ghép khối dựa trên tìm kiếm đầy đủ (Full search-based Block Matching Algorithm - FBMA), thuật toán tìm kiếm ba bước (Three-Step Search - TSS), và thuật toán tìm kiếm ba bước mới (New Three-step Search - NTS) có thể được sử dụng. Vectơ chuyển động có thể có trị số vectơ chuyển động trong các đơn vị của $1/2$ hoặc $1/4$ điểm ảnh dựa trên các điểm ảnh được nội suy. Bộ phận dự báo chuyển động có thể dự báo bộ phận dự báo hiện tại bằng cách sử dụng phương pháp dự báo chuyển động theo các cách khác nhau. Với tư cách là phương pháp dự báo chuyển động, các phương pháp khác nhau chẳng hạn như phương pháp nhảy, phương pháp hợp nhất, phương pháp dự báo vectơ chuyển động tiên tiến (Advanced Movement Vector Prediction - AMVP), và phương pháp sao chép nội khối có thể được sử dụng.

Bộ dự báo nội ảnh 125 có thể tạo ra bộ phận dự báo dựa trên thông tin về điểm ảnh tham chiếu gần khối hiện tại, đây là thông tin điểm ảnh trong hình ảnh hiện tại. Khi khối gần kề của bộ phận dự báo hiện tại là khối phải chịu dự báo liên ảnh và do đó điểm ảnh tham chiếu là điểm ảnh phải chịu dự báo liên ảnh, thông tin điểm ảnh tham chiếu của khối gần kề phải chịu việc dự báo nội ảnh có thể được sử dụng thay cho điểm ảnh tham chiếu được đưa vào khối phải chịu dự báo liên ảnh. Tức là, khi điểm ảnh tham chiếu không sẵn có, ít nhất một điểm ảnh tham chiếu sẵn có có thể được sử dụng thay cho

thông tin điểm ảnh tham chiếu không sẵn có.

Các phương thức dự báo trong dự báo nội ảnh có thể gồm phương thức dự báo định hướng mà sử dụng thông tin điểm ảnh tham chiếu phụ thuộc vào hướng dự báo và phương thức không định hướng mà không sử dụng thông tin định hướng khi thực hiện việc dự báo. Phương thức dự báo thông tin độ sáng và phương thức dự báo thông tin màu có thể là khác nhau, và thông tin phương thức dự báo nội ảnh được sử dụng để dự báo thông tin độ sáng, hoặc thông tin tín hiệu độ sáng được dự báo có thể được sử dụng để dự báo thông tin màu.

Khi dự báo nội ảnh được thực hiện và bộ phận dự báo có kích cỡ bằng bộ biến đổi, dự báo nội ảnh có thể được thực hiện trên bộ phận dự báo dựa trên các điểm ảnh được bố trí ở bên trái của bộ phận dự báo, các điểm ảnh được bố trí trên góc trên-trái của bộ phận dự báo, và các điểm ảnh được bố trí trên đỉnh của bộ phận dự báo. Tuy nhiên, khi dự báo nội ảnh được thực hiện và bộ phận dự báo có kích cỡ khác với bộ biến đổi, dự báo nội ảnh có thể được thực hiện bằng cách sử dụng điểm ảnh tham chiếu dựa trên bộ biến đổi. Ngoài ra, dự báo nội ảnh sử dụng $N \times N$ chia tách chỉ cho bộ phận mã hóa tối thiểu có thể được sử dụng.

Theo phương pháp dự báo nội ảnh, khối dự báo có thể được tạo ra sau khi ứng dụng bộ lọc trơn nội ảnh thích ứng (Adaptive Intra Smoothing - AIS) cho điểm ảnh tham chiếu phụ thuộc vào phương thức dự báo. Loại của bộ lọc AIS được ứng dụng cho điểm ảnh tham chiếu có thể thay đổi. Để thực hiện phương pháp dự báo nội ảnh, phương thức dự báo nội ảnh của bộ phận dự báo hiện tại có thể được dự báo từ phương thức dự báo nội ảnh của bộ phận dự báo gần bộ phận dự báo hiện tại. Khi phương thức dự báo của bộ phận dự báo hiện tại được dự báo sử dụng thông tin phương thức được dự báo từ bộ phận dự báo gần, thông tin biểu thị rằng bộ phận dự báo hiện tại và bộ phận dự báo gần có cùng phương thức dự báo có thể được truyền sử dụng thông tin cờ được xác định trước khi phương thức dự báo nội ảnh của bộ phận dự báo hiện tại là giống như phương thức dự báo nội ảnh của bộ phận dự báo gần, và mã hóa entropi có thể được thực hiện để mã hóa thông tin phương thức dự báo của khối hiện tại khi phương thức dự báo của bộ phận dự báo hiện tại khác với phương thức dự báo của bộ phận dự báo gần.

Ngoài ra, khối còn lại gồm thông tin về trị số còn lại, là hiệu số giữa bộ phận dự báo phải chịu dự báo và khối gốc của bộ phận dự báo, có thể được tạo ra dựa trên các bộ phận dự báo được tạo ra bởi các bộ dự báo 120 và 125. Khối còn lại được tạo ra có thể được nhập vào bộ biến đổi 130.

Bộ biến đổi 130 có thể biến đổi khối còn lại gồm thông tin về trị số còn lại giữa khối gốc và các bộ phận dự báo được tạo ra bởi các bộ dự báo 120 và 125 bằng cách sử dụng phương pháp biến đổi chẳng hạn như biến đổi cosin rời rạc (Discrete Cosine Transform - DCT), biến đổi sin rời rạc (Discrete Sine Transform - DST), và KLT. Việc liệu có ứng dụng DCT, DST, hoặc KLT để biến đổi khối còn lại có thể được xác định dựa trên thông tin phương thức dự báo nội ảnh của bộ phận dự báo được sử dụng để tạo ra khối còn lại.

Bộ lượng tử hóa 135 có thể lượng tử hóa các trị số được biến đổi thành miền tần số bởi bộ biến đổi 130. Các hệ số lượng tử hóa có thể thay đổi phụ thuộc vào khối hoặc tầm quan trọng của hình ảnh. Các trị số được tính toán bởi bộ lượng tử hóa 135 có thể được cung cấp cho bộ lượng tử hóa ngược 140 và bộ sắp xếp lại thứ tự 160.

Bộ sắp xếp lại thứ tự 160 có thể thực hiện việc sắp xếp lại thứ tự các trị số hệ số trên các trị số còn lại được lượng tử hóa.

Bộ sắp xếp lại thứ tự 160 có thể thay đổi các hệ số từ dạng khối hai chiều sang dạng vector một chiều thông qua phương pháp quét hệ số. Ví dụ như, bộ sắp xếp lại thứ tự 160 có thể quét các hệ số DC và thậm chí các hệ số tần số cao sử dụng phương pháp quét hình chữ chi để thay đổi các hệ số sang dạng vector một chiều. Phụ thuộc vào phương thức dự báo nội ảnh và kích cỡ của bộ biến đổi, quét dọc trong đó các hệ số kiểu khối hai chiều được quét theo hướng cột hoặc quét ngang trong đó các hệ số kiểu khối hai chiều được quét theo hướng hàng có thể được sử dụng thay vì quét hình chữ chi. Tức là, bộ sắp xếp lại thứ tự có thể xác định phương pháp quét cần được sử dụng trong số quét hình chữ chi, quét dọc, và quét ngang phụ thuộc vào kích cỡ của bộ biến đổi và phương thức dự báo nội ảnh.

Bộ mã hóa entropi 165 có thể thực hiện mã hóa entropi dựa trên các trị số được tính toán bởi bộ sắp xếp lại thứ tự 160. Mã hóa entropi có thể được sử dụng, ví dụ như, các phương pháp mã hóa khác nhau chẳng hạn như mã hóa Golomb hàm mũ, mã hóa độ dài có thể thay đổi thích ứng ngữ cảnh (Context-Adaptive Variable Length Coding - CAVLC), và mã hóa số học nhị phân thích ứng ngữ cảnh (Context-Adaptive Binary Arithmetic Coding - CABAC).

Bộ mã hóa entropi 165 có thể mã hóa nhiều thông tin chẳng hạn như trị số còn lại thông tin hệ số và thông tin kiểu khối của bộ phận mã hóa, thông tin phương thức dự báo, thông tin bộ phận chia tách, thông tin bộ phận dự báo, thông tin bộ phận truyền, thông tin vector chuyển động, thông tin khung tham chiếu, thông tin nội suy khối, và thông tin lọc

từ bộ sắp xếp lại thứ tự 160 và các bộ dự báo 120 và 125.

Bộ mã hóa entropi 165 có thể thực hiện mã hóa entropi trên các trị số hệ số của bộ phận mã hóa được nhập từ bộ sắp xếp lại thứ tự 160.

Bộ lượng tử hóa ngược 140 có thể lượng tử hóa ngược các trị số được lượng tử hóa bởi bộ lượng tử hóa 135, và bộ biến đổi ngược 145 có thể biến đổi ngược các trị số được biến đổi bởi bộ biến đổi 130. Trị số còn lại được tạo ra bởi bộ lượng tử hóa ngược 140 và bộ biến đổi ngược 145 có thể được kết hợp với bộ phận dự báo được dự báo thông qua bộ ước lượng chuyển động, bộ bù chuyển động, và bộ dự báo nội ảnh của các bộ dự báo 120 và 125 để tạo ra khối được tái cấu trúc.

Bộ lọc 150 có thể ứng dụng ít nhất một trong số bộ lọc giải khối, bộ hiệu chỉnh độ lệch, và bộ lọc lọc vòng lặp thích ứng (Adaptive Loop Filtering - ALF).

Bộ lọc giải khối có thể loại bỏ sự méo khối đã xảy ra do đường biên giữa các khối từ hình ảnh được tái cấu trúc. Để xác định việc liệu có thực hiện giải khối, việc liệu có ứng dụng bộ lọc giải khối cho khối hiện tại có thể được xác định dựa trên các điểm ảnh được đưa vào một số hàng hoặc cột trong khối. Khi bộ lọc giải khối được ứng dụng cho khối, bộ lọc khỏe hoặc bộ lọc yếu có thể được ứng dụng phụ thuộc vào cường độ giải khối bộ lọc được yêu cầu. Ngoài ra, khi bộ lọc giải khối được ứng dụng, lọc đứng và lọc ngang có thể được thực hiện sao cho lọc ngang và lọc đứng được xử lý song song.

Bộ hiệu chỉnh độ lệch có thể điều chỉnh độ lệch từ hình ảnh gốc trong hình ảnh phải chịu giải khối trong các đơn vị của các điểm ảnh. Phương pháp phân loại các điểm ảnh được đưa vào hình ảnh thành một số khu vực nhất định, xác định khu vực cần phải chịu độ lệch, và ứng dụng độ lệch cho khu vực được xác định hoặc phương pháp ứng dụng độ lệch xét đến thông tin cạnh biên của mỗi điểm ảnh có thể được sử dụng để thực hiện hiệu chỉnh độ lệch trên hình ảnh cụ thể.

ALF có thể được thực hiện dựa trên các trị số thu được bằng cách so sánh hình ảnh được tái cấu trúc được lọc và hình ảnh gốc. Bằng cách phân loại các điểm ảnh được đưa vào hình ảnh thành các nhóm được xác định trước và xác định bộ lọc cần được ứng dụng cho mỗi trong số các nhóm, lọc vi sai có thể được thực hiện cho mỗi nhóm. Thông tin về việc liệu có ứng dụng ALF có thể được truyền cho mỗi bộ phận mã hóa (Coding Unit - CU), và hình dạng và lọc các hệ số của bộ lọc AFL cần được ứng dụng có thể thay đổi phụ thuộc vào khối. Ngoài ra, bộ lọc ALF ở cùng một dạng (dạng cố định) có thể được ứng dụng bất kể đặc điểm của khối cần được ứng dụng.

Bộ nhớ 155 có thể lưu trữ hình ảnh hoặc khối được tái cấu trúc được tính toán

thông qua bộ lọc 150. Hình ảnh được lưu trữ hoặc khôi phục tái cấu trúc có thể được cung cấp cho các bộ dự báo 120 và 125 khi dự báo liên ảnh được thực hiện.

Fig.2 là sơ đồ khối của bộ giải mã video theo một phương án của sáng chế này.

Như được thể hiện trên Fig.2, bộ giải mã video 200 có thể gồm bộ giải mã entropi 210, bộ sắp xếp lại thứ tự 215, bộ lượng tử hóa ngược 220, bộ biến đổi ngược 225, các bộ dự báo 230 và 235, bộ lọc 240, và bộ nhớ 245.

Khi dòng bit video được nhập vào từ bộ mã hóa video, dòng bit đầu vào có thể được giải mã theo quy trình ngược với quy trình của bộ mã hóa video.

Bộ giải mã entropi 210 có thể thực hiện giải mã entropi theo quy trình ngược với quy trình trong đó bộ mã hóa entropi của bộ mã hóa video thực hiện mã hóa entropi. Ví dụ như, các phương pháp khác nhau chẳng hạn như mã hóa Golomb hàm mũ, mã hóa độ dài có thể thay đổi thích ứng ngữ cảnh (Context-Adaptive Variable Length Coding - CAVLC), và mã hóa số học nhị phân thích ứng ngữ cảnh (Context-Adaptive Binary Arithmetic Coding - CABAC) có thể được ứng dụng cho tương ứng với phương pháp được thực hiện bởi bộ mã hóa video.

Bộ giải mã entropi 210 có thể giải mã thông tin liên quan đến dự báo nội ảnh và dự báo liên ảnh được thực hiện bởi bộ mã hóa.

Bộ sắp xếp lại thứ tự 215 có thể thực hiện việc sắp xếp lại trên dòng bit phải chịu giải mã entropi bởi bộ giải mã entropi 210 dựa trên phương pháp sắp xếp lại thứ tự được sử dụng bởi bộ mã hóa. Bộ sắp xếp lại thứ tự 215 có thể tái cấu trúc các hệ số được biểu hiện dưới dạng vector một chiều thành các hệ số kiểu khối hai chiều để sắp xếp lại thứ tự các hệ số kiểu khối hai chiều. Bộ sắp xếp lại thứ tự 215 có thể nhận thông tin liên quan đến quét hệ số được thực hiện bởi bộ mã hóa và thực hiện việc sắp xếp lại thứ tự thông qua phương pháp quét ngược dựa trên thứ tự quét của bộ mã hóa.

Bộ lượng tử hóa ngược 220 có thể thực hiện lượng tử hóa ngược dựa trên khối được sắp xếp lại thứ tự các trị số hệ số và các thông số lượng tử hóa được cung cấp bởi bộ mã hóa.

Bộ biến đổi ngược 225 có thể thực hiện DCT ngược, DST ngược, và KLT ngược theo kết quả lượng tử hóa được thực hiện bởi bộ mã hóa video, trong đó DCT ngược, DST ngược, và KLT ngược là việc đảo ngược các phép biến đổi đã được thực hiện bởi bộ biến đổi, tức là, DCT, DST, và KLT. Biến đổi ngược có thể được thực hiện dựa trên bộ phận truyền được xác định bởi bộ mã hóa video. Trong bộ biến đổi ngược 225 của bộ giải mã video, các kỹ thuật biến đổi (ví dụ như, DCT, DST, và KLT) có thể được thực

hiện một cách chọn lọc phụ thuộc vào nhiều thông tin chẳng hạn như phương pháp dự báo, kích cỡ của khối hiện tại, và hướng dự báo.

Các bộ dự báo 230 và 235 có thể tạo ra khối dự báo dựa trên thông tin liên quan đến tạo ra khối dự báo, là khối được cung cấp bởi bộ giải mã entropi 210, và thông tin về hình ảnh hoặc khối được giải mã trước, được cung cấp bởi bộ nhớ 245.

Như được mô tả trên đây, khi dự báo nội ảnh được thực hiện theo phương thức giống như phương thức của bộ mã hóa video và bộ phận dự báo có kích cỡ bằng bộ biến đổi, dự báo nội ảnh có thể được thực hiện trên bộ phận dự báo dựa trên các điểm ảnh được bố trí ở bên trái của bộ phận dự báo, các điểm ảnh được bố trí trên góc trên-trái của bộ phận dự báo, và các điểm ảnh được bố trí trên đỉnh của bộ phận dự báo. Mặt khác, khi dự báo nội ảnh được thực hiện và bộ phận dự báo có kích cỡ khác với bộ biến đổi, dự báo nội ảnh có thể được thực hiện sử dụng điểm ảnh tham chiếu dựa trên bộ biến đổi. Ngoài ra, dự báo nội ảnh sử dụng $N \times N$ chia tách chỉ cho bộ phận mã hóa tối thiểu có thể được sử dụng.

Các bộ dự báo 230 và 235 có thể gồm bộ xác định bộ phận dự báo, bộ phận dự báo liên ảnh, và bộ dự báo nội ảnh. Bộ xác định bộ phận dự báo có thể nhận nhiều thông tin chẳng hạn như thông tin bộ phận dự báo, thông tin phương thức dự báo cho phương pháp dự báo nội ảnh, và thông tin liên quan đến dự báo chuyển động cho phương pháp dự báo liên ảnh từ bộ giải mã entropi 210, phân loại bộ phận dự báo trong bộ phận mã hóa hiện tại, và xác định việc liệu bộ phận dự báo thực hiện dự báo liên ảnh hoặc dự báo nội ảnh. Bằng cách sử dụng thông tin cần thiết cho dự báo liên ảnh của bộ phận dự báo hiện tại được cung cấp bởi bộ mã hóa video, bộ phận dự báo liên ảnh 230 có thể thực hiện dự báo liên ảnh trên bộ phận dự báo hiện tại dựa trên thông tin được đưa vào ít nhất một trong số hình ảnh đứng trước hình ảnh hiện tại gồm bộ phận dự báo hiện tại hoặc hình ảnh theo sau hình ảnh hiện tại. Theo lựa chọn, dự báo liên ảnh có thể được thực hiện dựa trên thông tin về một số khu vực được tái cấu trúc trước trong hình ảnh hiện tại gồm bộ phận dự báo hiện tại.

Bộ phận dự báo liên ảnh 230 có thể xác định việc liệu phương pháp dự báo chuyển động cho bộ phận dự báo được đưa vào bộ phận mã hóa tương ứng là phương thức nhảy, phương thức hợp nhất, phương thức AMVP, hoặc phương thức sao chép nội khối dựa trên bộ phận mã hóa để thực hiện dự báo liên ảnh.

Bộ dự báo nội ảnh 235 có thể tạo ra khối dự báo dựa trên thông tin về các điểm ảnh trong hình ảnh hiện tại. Khi bộ phận dự báo là bộ phận dự báo phải chịu dự báo nội

ảnh, bộ dự báo nội ảnh 235 có thể thực hiện dự báo nội ảnh dựa trên thông tin phương thức dự báo nội ảnh của bộ phận dự báo được cung cấp bởi bộ mã hóa video. Bộ dự báo nội ảnh 235 có thể gồm bộ lọc AIS, bộ nội suy điểm ảnh tham chiếu, và bộ lọc DC. Bộ lọc AIS, là một phần mà thực hiện việc lọc trên điểm ảnh tham chiếu của khối hiện tại, có thể xác định việc liệu có ứng dụng việc lọc phụ thuộc vào phương thức dự báo của bộ phận dự báo hiện tại. Bộ lọc AIS có thể thực hiện việc lọc AIS trên điểm ảnh tham chiếu của khối hiện tại sử dụng thông tin lọc AIS và phương thức dự báo của bộ phận dự báo được cung cấp bởi bộ mã hóa video. Khi phương thức dự báo của khối hiện tại là phương thức trong đó việc lọc AKhông được thực hiện, bộ lọc AIS có thể không được ứng dụng.

Khi phương thức dự báo của bộ phận dự báo là phương thức dự báo trong đó dự báo nội ảnh được thực hiện dựa trên trị số điểm ảnh thu được bằng cách nội suy điểm ảnh tham chiếu, bộ nội suy điểm ảnh tham chiếu có thể nội suy điểm ảnh tham chiếu để tạo ra điểm ảnh tham chiếu trong đơn vị điểm ảnh nhỏ hơn hoặc bằng số nguyên. Khi phương thức dự báo của bộ phận dự báo hiện tại là phương thức dự báo trong đó khối dự báo được tạo ra mà không nội suy điểm ảnh tham chiếu, điểm ảnh tham chiếu có thể không được nội suy. Khi phương thức dự báo của khối hiện tại là phương thức DC, bộ lọc DC có thể tạo ra khối dự báo thông qua lọc.

Hình ảnh hoặc khối được tái cấu trúc có thể được cung cấp cho bộ lọc 240. Bộ lọc 240 có thể gồm bộ lọc giải khối, bộ hiệu chỉnh độ lệch, và bộ lọc ALF.

Bộ lọc 240 có thể nhận thông tin về việc liệu bộ lọc giải khối được ứng dụng cho hình ảnh hoặc khối tương ứng hoặc thông tin về việc liệu bộ lọc khỏe hoặc bộ lọc yếu được ứng dụng khi bộ lọc giải khối được ứng dụng từ bộ mã hóa video. Bộ lọc giải khối của bộ giải mã video có thể nhận thông tin liên quan đến bộ lọc giải khối, được cung cấp bởi bộ mã hóa video, và bộ giải mã video có thể thực hiện giải khối bộ lọc trên khối tương ứng.

Bộ hiệu chỉnh độ lệch có thể thực hiện hiệu chỉnh độ lệch trên hình ảnh được tái cấu trúc dựa trên loại hiệu chỉnh độ lệch, thông tin trị số độ lệch, và thông tin tương tự được ứng dụng cho hình ảnh khi mã hóa.

ALF có thể được ứng dụng cho bộ phận mã hóa dựa trên thông tin về việc liệu có ứng dụng ALF, ALF thông tin hệ số, và thông tin tương tự được cung cấp từ bộ mã hóa. Thông tin ALF có thể được cung cấp bằng cách được đưa vào tập hợp thông số cụ thể.

Bộ nhớ 245 có thể lưu trữ hình ảnh hoặc khối được tái cấu trúc sao cho hình ảnh hoặc khối có thể được sử dụng là hình ảnh tham chiếu hoặc khối tham chiếu và cũng có

thể cung cấp hình ảnh được tái cấu trúc cho bộ phận đầu ra.

Fig.3 là sơ đồ thể hiện đơn vị cây mã hóa cơ bản theo một phương án của sáng chế này.

Bộ phận mã hóa có kích cỡ lớn nhất có thể được xác định là khối cây mã hóa. Một hình ảnh được chia tách thành nhiều đơn vị cây mã hóa (Coding Tree Unit - CTU). Đơn vị cây mã hóa, là bộ phận mã hóa có kích cỡ lớn nhất, có thể được gọi là bộ phận mã hóa lớn nhất (Largest Coding Unit - LCU). Fig.3 thể hiện ví dụ trong đó một hình ảnh được chia tách thành nhiều các đơn vị cây mã hóa.

Kích cỡ của đơn vị cây mã hóa có thể được xác định ở cấp độ hình ảnh hoặc cấp độ trình tự. Để thực hiện được điều này, thông tin biểu thị kích cỡ của đơn vị cây mã hóa có thể được báo hiệu thông qua tập hợp thông số hình ảnh hoặc tập hợp thông số trình tự.

Ví dụ như, kích cỡ của đơn vị cây mã hóa cho toàn bộ hình ảnh theo trình tự có thể được đặt thành 128×128 . Theo lựa chọn, một trong số 128×128 hoặc 256×256 có thể được xác định là kích cỡ của đơn vị cây mã hóa ở cấp độ hình ảnh. Theo một ví dụ, kích cỡ của đơn vị cây mã hóa ở hình ảnh thứ nhất có thể được đặt thành 128×128 , và kích cỡ của đơn vị cây mã hóa ở hình ảnh thứ hai có thể được đặt thành 256×256 .

Đơn vị cây mã hóa có thể được chia tách để tạo ra khối mã hóa. Khối mã hóa biểu thị đơn vị cơ bản để xử lý mã hóa hoặc giải mã. Theo một ví dụ, dự báo hoặc biến đổi có thể được thực hiện cho mỗi khối mã hóa, hoặc phương thức dự báo có thể được xác định cho mỗi khối mã hóa. Ở đây, phương thức dự báo biểu thị phương pháp tạo ra hình ảnh dự báo. Theo một ví dụ, phương thức dự báo có thể gồm dự báo nội ảnh, dự báo liên ảnh, tham chiếu hình ảnh hiện tại (Current Picture Referencing - CPR) (hoặc sao chép nội khối (Intra-Block Copy - IBC)), hoặc việc dự báo kết hợp. Khối dự báo cho khối mã hóa có thể được tạo ra sử dụng ít nhất một phương thức dự báo trong số dự báo nội ảnh, dự báo liên ảnh, tham chiếu hình ảnh hiện tại, hoặc việc dự báo kết hợp cho khối mã hóa.

Thông tin biểu thị phương thức dự báo của khối hiện tại có thể được báo hiệu trong dòng bit. Theo một ví dụ, thông tin có thể là cờ 1 bit biểu thị việc liệu phương thức dự báo là phương thức nội ảnh hoặc phương thức liên ảnh. Tham chiếu hình ảnh hiện tại hoặc việc dự báo kết hợp có thể là sẵn có chỉ khi nội dung được xác định là phương thức dự báo của khối hiện tại là phương thức liên ảnh.

Tham chiếu hình ảnh hiện tại là để đặt hình ảnh hiện tại là hình ảnh tham chiếu và thu được khối dự báo của khối hiện tại từ khu vực của hình ảnh hiện tại mà ở đó việc mã hóa hoặc giải mã được hoàn thành. Ở đây, hình ảnh hiện tại đề cập đến hình ảnh gồm

khối hiện tại. Thông tin biểu thị rằng tham chiếu hình ảnh hiện tại được ứng dụng cho khối hiện tại có thể được báo hiệu trong dòng bit. Theo một ví dụ, thông tin có thể là cờ 1 bit. Nội dung có thể được xác định là phương thức dự báo của khối hiện tại là tham chiếu hình ảnh hiện tại khi cờ là đúng, và nội dung có thể được xác định là phương thức dự báo của khối hiện tại là dự báo liên ảnh khi cờ là sai.

Theo lựa chọn, phương thức dự báo của khối hiện tại có thể được xác định dựa trên chỉ số hình ảnh tham chiếu. Theo một ví dụ, khi chỉ số hình ảnh tham chiếu biểu thị hình ảnh hiện tại, nội dung có thể được xác định là phương thức dự báo của khối hiện tại là tham chiếu hình ảnh hiện tại. Khi chỉ số hình ảnh tham chiếu biểu thị hình ảnh không phải là hình ảnh hiện tại, nội dung có thể được xác định là phương thức dự báo của khối hiện tại là dự báo liên ảnh. Tức là, tham chiếu hình ảnh hiện tại là phương pháp dự báo mà sử dụng thông tin của khu vực của hình ảnh hiện tại mà ở đó việc mã hóa hoặc giải mã được hoàn thành, và dự báo liên ảnh là phương pháp dự báo mà sử dụng thông tin của một hình ảnh khác mà ở đó việc mã hóa hoặc giải mã được hoàn thành.

Việc dự báo kết hợp biểu thị phương thức mã hóa thu được bằng cách kết hợp hai hoặc nhiều hơn hai trong số dự báo nội ảnh, dự báo liên ảnh, và tham chiếu hình ảnh hiện tại. Theo một ví dụ, khi việc dự báo kết hợp được ứng dụng, khối dự báo thứ nhất có thể được tạo ra dựa trên một trong số dự báo nội ảnh, dự báo liên ảnh, hoặc tham chiếu hình ảnh hiện tại, và khối dự báo thứ hai có thể được tạo ra dựa trên nhau. Khi khối dự báo thứ nhất và khối dự báo thứ hai được tạo ra, khối dự báo cuối cùng có thể được tạo ra thông qua phép toán lấy trung bình hoặc phép tính tổng có trọng số giữa khối dự báo thứ nhất và khối dự báo thứ hai. Thông tin biểu thị việc liệu việc dự báo kết hợp được ứng dụng có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit.

Fig.4 là sơ đồ thể hiện các kiểu chia tách khác nhau của khối mã hóa.

Khối mã hóa có thể được chia tách thành nhiều khối mã hóa dựa trên chia tách cây tứ phân, chia tách cây nhị phân, hoặc chia tách cây tam phân. Khối mã hóa thu được thông qua chia tách có thể được chia tách lại thành nhiều khối mã hóa dựa trên chia tách cây tứ phân, chia tách cây nhị phân, hoặc chia tách cây tam phân.

Chia tách cây tứ phân biểu thị kỹ thuật chia tách để chia tách khối hiện tại thành bốn khối. Do chia tách cây tứ phân, khối hiện tại có thể được chia tách thành bốn phần hình vuông (xem “SPLIT_QT” trên Fig.4A).

Chia tách cây nhị phân biểu thị kỹ thuật chia tách để chia tách khối hiện tại thành hai khối. Phân chia khối hiện tại thành hai khối theo hướng dọc (tức là, dùng đường dọc

ngang qua khối hiện tại) có thể được gọi là chia tách cây nhị phân dọc, và phân chia khối hiện tại thành hai khối theo hướng ngang (tức là, dùng đường ngang ngang qua khối hiện tại) có thể được gọi là chia tách cây nhị phân ngang. Do chia tách cây nhị phân, khối hiện tại có thể được chia tách thành hai phần không phải hình vuông. Trên Fig.4B, “SPLIT_BT_VER” thể hiện kết quả chia tách cây nhị phân dọc. Trên Fig.4C, “SPLIT_BT_HOR” thể hiện kết quả chia tách cây nhị phân ngang.

Chia tách cây tam phân biểu thị kỹ thuật chia tách để chia tách khối hiện tại thành ba khối. Phân chia khối hiện tại thành ba khối theo hướng dọc (tức là, sử dụng hai đường dọc ngang qua khối hiện tại) có thể được gọi là chia tách cây tam phân dọc, và phân chia khối hiện tại thành ba khối theo hướng ngang (tức là, dùng hai đường ngang ngang qua khối hiện tại) có thể được gọi là chia tách cây tam phân ngang. Do chia tách cây tam phân, khối hiện tại có thể được chia tách thành ba phần không phải hình vuông. Trong trường hợp này, độ rộng và độ cao của phần phân chia được bố trí tại trung tâm của khối hiện tại có thể bằng hai lần độ rộng và độ cao của các phần kia. Trên Fig.4D, “SPLIT_TT_VER” thể hiện kết quả chia tách cây tam phân dọc. Trên Fig.4E, “SPLIT_TT_HOR” thể hiện kết quả chia tách cây tam phân ngang.

Số lần đơn vị cây mã hóa bị chia tách có thể được xác định là độ sâu chia tách (độ sâu phân chia). Độ sâu chia tách tối đa của đơn vị cây mã hóa có thể được xác định ở cấp độ trình tự hoặc cấp độ hình ảnh. Do đó, độ sâu chia tách tối đa của đơn vị cây mã hóa có thể thay đổi phụ thuộc vào trình tự hoặc hình ảnh.

Theo lựa chọn, độ sâu chia tách tối đa có thể được xác định riêng cho mỗi kỹ thuật chia tách. Theo một ví dụ, độ sâu chia tách tối đa được phép cho chia tách cây tứ phân có thể là khác nhau từ độ sâu chia tách tối đa được phép cho chia tách cây nhị phân và/hoặc chia tách cây tam phân.

Bộ mã hóa có thể báo hiệu thông tin biểu thị ít nhất một trong số kiểu chia tách hoặc độ sâu chia tách của khối hiện tại trong dòng bit. Bộ giải mã có thể xác định kiểu chia tách và độ sâu chia tách của đơn vị cây mã hóa dựa trên thông tin được phân tách từ dòng bit.

Fig.5 là sơ đồ minh họa khía cạnh chia tách của đơn vị cây mã hóa.

Chia tách khối mã hóa sử dụng kỹ thuật chia tách chẳng hạn như chia tách cây tứ phân, chia tách cây nhị phân, và/hoặc chia tách cây tam phân có thể được gọi là chia tách nhiều cây.

Các khối mã hóa được tạo ra bằng cách ứng dụng chia tách nhiều cây cho khối mã

hóa có thể được gọi là các khối mã hóa dưới. Khi độ sâu chia tách của khối mã hóa là k , độ sâu chia tách của các khối mã hóa dưới được đặt thành $k+1$.

Ngược lại, liên quan đến các khối mã hóa với độ sâu chia tách của $k+1$, khối mã hóa với độ sâu chia tách của k có thể được gọi là khối mã hóa trên.

Kiểu chia tách của khối mã hóa hiện tại có thể được xác định dựa trên ít nhất một trong số kiểu chia tách của khối mã hóa trên hoặc kiểu chia tách của khối mã hóa lân cận. Ở đây, khối mã hóa lân cận gần kề khối mã hóa hiện tại và có thể gồm ít nhất một trong số khối lân cận được bố trí phía trên khối mã hóa hiện tại, khối lân cận được bố trí về phía trái của khối mã hóa hiện tại, hoặc khối lân cận gần kề góc trên-trái của khối mã hóa hiện tại. Ở đây, kiểu chia tách có thể gồm ít nhất một trong số sự có mặt của chia tách cây tứ phân, sự có mặt của chia tách cây nhị phân, hướng chia tách cây nhị phân, sự có mặt của chia tách cây tam phân, hoặc hướng chia tách cây tam phân.

Để xác định kiểu chia tách của khối mã hóa, thông tin biểu thị việc liệu khối mã hóa bị chia tách có thể được báo hiệu trong dòng bit. Thông tin là cờ 1 bit “split_cu_flag,” và cờ là đúng biểu thị rằng khối mã hóa được chia tách bởi kỹ thuật chia tách nhiều cây.

Khi split_cu_flag là đúng, thông tin biểu thị việc liệu khối mã hóa bị chia tách thông qua chia tách cây tứ phân có thể được báo hiệu trong dòng bit. Thông tin là cờ 1 bit “split_qt_flag,” và khi cờ là đúng, khối mã hóa có thể được chia tách thành bốn khối.

Ví dụ như, nội dung được thể hiện trên Fig.5 là bốn khối mã hóa với độ sâu chia tách là một được tạo ra do đơn vị cây mã hóa bị chia tách thông qua chia tách cây tứ phân. Nội dung cũng được thể hiện là chia tách cây tứ phân được ứng dụng lại cho khối mã hóa thứ nhất và khối mã hóa thứ tư trong số bốn khối mã hóa được tạo ra bằng cách thực hiện chia tách cây tứ phân. Kết quả là, bốn khối mã hóa với độ sâu chia tách là hai có thể được tạo ra.

Ngoài ra, bằng cách ứng dụng lại chia tách cây tứ phân cho khối mã hóa với độ sâu chia tách là hai, khối mã hóa với độ sâu chia tách là ba có thể được tạo ra.

Khi chia tách cây tứ phân không được ứng dụng cho khối mã hóa, việc liệu có thực hiện chia tách cây nhị phân hoặc chia tách cây tam phân trên khối mã hóa có thể được xác định xem xét ít nhất một trong số kích cỡ của khối mã hóa, việc liệu khối mã hóa được bố trí tại mép hình ảnh, độ sâu chia tách tối đa, hoặc khía cạnh chia tách của khối lân cận. Khi nội dung được xác định là chia tách cây nhị phân hoặc chia tách cây tam phân được thực hiện trên khối mã hóa, thông tin biểu thị hướng chia tách có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit mtt_split_cu_vertical_flag.

Dựa trên cờ, việc liệu hướng chia tách là dọc hoặc ngang có thể được xác định. Ngoài ra, thông tin biểu thị hình thức nào trong số chia tách cây nhị phân và chia tách cây tam phân được ứng dụng cho khối mã hóa có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit `mtt_split_cu_binary_flag`. Dựa trên cờ, việc liệu chia tách cây nhị phân hoặc chia tách cây tam phân được ứng dụng cho khối mã hóa có thể được xác định.

Ví dụ như, nội dung được thể hiện trên Fig.5 là chia tách cây nhị phân dọc được ứng dụng cho khối mã hóa với độ sâu chia tách là một, chia tách cây tam phân dọc được ứng dụng cho khối mã hóa trái trong số các khối mã hóa được tạo ra do chia tách cây nhị phân dọc, và chia tách cây nhị phân dọc được ứng dụng cho khối mã hóa phải.

Do hình ảnh được chia tách thành các đơn vị cây mã hóa, khối nhỏ hơn kích cỡ đặt sẵn có thể có mặt ở khu vực gần kề mép phải hoặc mép dưới của hình ảnh. Khi giả sử rằng khối là đơn vị cây mã hóa, đơn vị cây mã hóa nhỏ hơn kích cỡ đặt sẵn có thể được tạo ra tại mép phải hoặc mép dưới của hình ảnh. Trong trường hợp này, kích cỡ của đơn vị cây mã hóa có thể được xác định dựa trên thông tin được báo hiệu thông qua tập hợp thông số trình tự hoặc tập hợp thông số hình ảnh.

Fig.6 thể hiện ví dụ trong đó khối nhỏ hơn đơn vị cây mã hóa của kích cỡ được xác định trước xảy ra tại mép hình ảnh.

Khi hình ảnh 1292×1080 được chia tách thành 128×128 các đơn vị cây mã hóa, khối nhỏ hơn 128×128 có mặt tại mép phải và mép dưới của hình ảnh trong ví dụ được thể hiện trên Fig.6. Trong các phương án sau đây, khối nhỏ hơn đơn vị cây mã hóa của kích cỡ đặt sẵn xảy ra tại mép hình ảnh được gọi là khối mép ranh giới không điển hình.

Chỉ phương pháp chia tách được xác định trước có thể được phép cho khối mép ranh giới không điển hình. Ở đây, phương pháp chia tách được xác định trước có thể gồm ít nhất một trong số chia tách cây tứ phân, chia tách cây tam phân, hoặc chia tách cây nhị phân.

Ví dụ như, chỉ chia tách cây tứ phân có thể được phép cho khối mép ranh giới không điển hình. Trong trường hợp này, chia tách cây tứ phân có thể được lặp lại cho đến khi khối tương ứng với mép hình ảnh trở thành kích cỡ chia tách cây tứ phân tối thiểu. Ở đây, kích cỡ chia tách cây tứ phân tối thiểu có thể được xác định trước trong bộ mã hóa và bộ giải mã. Theo lựa chọn, thông tin biểu thị kích cỡ chia tách cây tứ phân tối thiểu có thể được báo hiệu trong dòng bit.

Fig.7 là sơ đồ thể hiện ví dụ trong đó chia tách cây tứ phân được thực hiện trên khối mép ranh giới không điển hình. Để thuận tiện cho việc mô tả, giả sử rằng kích cỡ

chia tách cây tứ phân tối thiểu là 4×4 .

Khối mép ranh giới không điển hình có thể được chia tách dựa trên khối hình vuông. Khối hình vuông có thể được suy ra dựa trên trị số lớn hơn giữa độ rộng và độ cao của khối mép ranh giới không điển hình. Theo một ví dụ, lũy thừa của 2 lớn hơn trị số tham chiếu và gần nhất với trị số tham chiếu có thể được coi là độ dài của một mặt của khối hình vuông. Theo một ví dụ, khối 12×20 được thể hiện trên Fig.7 có thể được xem xét để thuộc về khối 32×32 , và kết quả chia tách cho khối 32×32 có thể được ứng dụng cho khối 12×20 .

Khi chia tách cây tứ phân được thực hiện trên khối 12×20 , khối có thể được chia tách thành khối 12×16 và khối 12×4 . Khi chia tách cây tứ phân được thực hiện trên mỗi trong số các khối, khối 12×16 được chia tách thành hai khối 8×8 và hai khối 4×8 , và khối 12×4 được chia tách thành khối 8×4 và khối 4×4 .

Chia tách cây tứ phân có thể được thực hiện lại trên khối 4×8 được bố trí lại tại mép hình ảnh, và do đó hai khối 4×4 có thể được tạo ra. Tương tự, chia tách cây tứ phân có thể được thực hiện lại trên khối 8×8 được bố trí tại mép hình ảnh, và do đó hai khối 4×4 có thể được tạo ra.

Theo lựa chọn, khi ít nhất một trong số độ rộng hoặc độ cao của khối nhỏ hơn hoặc bằng kích cỡ chia tách cây tứ phân tối thiểu, chia tách cây nhị phân có thể được thực hiện. Ở đây, kích cỡ chia tách cây tứ phân tối thiểu có thể biểu thị độ rộng chia tách cây tứ phân tối thiểu hoặc độ cao chia tách cây tứ phân tối thiểu. Theo một ví dụ, khi kích cỡ chia tách cây tứ phân tối thiểu là 4×4 , độ rộng chia tách cây tứ phân tối thiểu và độ cao chia tách cây tứ phân tối thiểu có thể là 4.

Trong trường hợp này, chia tách cây nhị phân dọc có thể được thực hiện khi độ rộng của khối nhỏ hơn hoặc bằng kích cỡ chia tách cây tứ phân tối thiểu, và chia tách cây nhị phân ngang có thể được thực hiện khi độ cao của khối nhỏ hơn hoặc bằng độ cao chia tách cây tứ phân tối thiểu.

Ngược lại, chia tách cây tứ phân có thể được thực hiện khi độ rộng hoặc độ cao của khối lớn hơn kích cỡ chia tách cây tứ phân tối thiểu. Theo một ví dụ, khi vị trí trên-phải và vị trí dưới-trái của khối chệch từ hình ảnh và độ rộng hoặc độ cao của khối lớn hơn kích cỡ chia tách cây tứ phân tối thiểu, chia tách cây tứ phân có thể được ứng dụng cho khối tương ứng.

Fig.8 là sơ đồ thể hiện ví dụ trong đó chia tách cây tứ phân được thực hiện trên khối gần kề cả mép phải và mép dưới của hình ảnh. Để thuận tiện cho việc mô tả, giả sử

rằng kích cỡ chia tách cây tứ phân tối thiểu là 4×4 .

Trong ví dụ được thể hiện trên Fig.8, khi chia tách cây tứ phân được thực hiện trên khối 32×32 gồm khối 12×20 , bốn khối 16×16 được tạo ra. Trong số các khối được tạo ra, chia tách cây tứ phân có thể được thực hiện lại cho hai khối 16×16 gồm dữ liệu kết cấu. Kết quả là, tọa độ x và tọa độ y có thể trích từ mép hình ảnh, và khối 8×8 gồm dữ liệu kết cấu với kích cỡ là 4×4 có thể được tạo ra. Do độ rộng và độ cao của khối 8×8 lớn hơn kích cỡ chia tách cây tứ phân tối thiểu, chia tách cây tứ phân có thể được thực hiện trên khối tương ứng.

Khi chia tách cây tứ phân được thực hiện trên khối 12×20 , khối 12×20 có thể được chia tách thành khối 12×16 và khối 12×4 . Khi chia tách cây tứ phân được thực hiện lại trên các khối, khối 12×16 được chia tách thành hai khối 8×8 và hai khối 4×8 , và khối 12×4 được chia tách thành khối 8×4 và khối 4×4 .

Do độ rộng của khối 4×8 được bố trí tại mép hình ảnh bằng kích cỡ chia tách cây tứ phân tối thiểu, chia tách cây nhị phân có thể được thực hiện trên khối 4×8 . Cụ thể, chia tách cây nhị phân dọc có thể được thực hiện dựa trên khối hình vuông (tức là, 8×8) gồm khối 4×8 .

Ngoài ra, do độ rộng của khối 8×4 được bố trí tại mép hình ảnh bằng kích cỡ chia tách cây tứ phân tối thiểu, chia tách cây nhị phân có thể được thực hiện trên khối 8×4 . Cụ thể, chia tách cây nhị phân ngang có thể được thực hiện dựa trên khối hình vuông (tức là, 8×8) gồm khối 4×8 .

Do sự chia tách, khối 8×4 , khối 4×4 , và khối 4×8 có thể được bố trí gần kề mép hình ảnh.

Theo lựa chọn, khi ít nhất một trong số độ rộng hoặc độ cao của khối nhỏ hơn hoặc bằng trị số ngưỡng, chia tách cây nhị phân có thể được thực hiện. Nếu không, chia tách cây tứ phân có thể được thực hiện. Ở đây, trị số ngưỡng có thể được suy ra dựa trên kích cỡ chia tách cây tứ phân tối thiểu. Theo một ví dụ, khi kích cỡ chia tách cây tứ phân tối thiểu là `minQTsize`, trị số ngưỡng có thể được đặt thành "`minQTsize << 1`." Theo lựa chọn, thông tin để xác định trị số ngưỡng có thể được báo hiệu riêng biệt trong dòng bit.

Fig.9 là sơ đồ thể hiện khía cạnh chia tách của khối gần kề mép hình ảnh. Để thuận tiện cho việc mô tả, giả sử rằng kích cỡ chia tách cây tứ phân tối thiểu là 4×4 . Trị số ngưỡng có thể được đặt thành 8.

Trước hết, chia tách cây tứ phân có thể được thực hiện trên khối 12×20 . Kết quả là, khối có thể được chia tách thành khối 12×16 và khối 12×4 . Do độ rộng và độ cao của

khối 12×16 lớn hơn trị số ngưỡng, chia tách cây tứ phân có thể được ứng dụng cho khối. Do đó, khối có thể được chia tách thành hai khối 8×8 và hai khối 4×8 .

Độ rộng của khối 12×4 lớn hơn trị số ngưỡng. Do đó, chia tách cây tứ phân có thể được ứng dụng cho khối 12×4 . Kết quả là, khối có thể được chia tách thành khối 8×4 và khối 4×4 .

Sau đó, do các độ rộng và độ cao của khối 4×8 và khối 8×4 được bố trí tại mép hình ảnh nhỏ hơn hoặc bằng trị số ngưỡng, chia tách cây nhị phân có thể được ứng dụng cho các khối.

Không giống như ví dụ nêu trên, khi ít nhất một trong số độ rộng hoặc độ cao của khối lớn hơn trị số ngưỡng, chia tách cây nhị phân có thể được thực hiện. Nếu không, chia tách cây tứ phân có thể được thực hiện.

Theo lựa chọn, chỉ chia tách cây tứ phân hoặc chia tách cây nhị phân có thể được ứng dụng cho khối mép ranh giới không điển hình. Theo một ví dụ, chia tách cây tứ phân có thể được lặp lại cho đến khi khối được bố trí tại mép hình ảnh có kích cỡ tối thiểu, hoặc chia tách cây nhị phân có thể được lặp lại cho đến khi khối được bố trí tại mép hình ảnh có kích cỡ tối thiểu.

Khối mép ranh giới không điển hình có thể được xác định là bộ phận mã hóa. Phương thức nhảy có thể được ứng dụng cố định cho khối mép ranh giới không điển hình, hoặc tất cả các hệ số biến đổi có thể đặt thành 0. Do đó, trị số của cờ khối được mã hóa (Coded Block Flag - CBF) biểu thị việc liệu khối mép ranh giới không điển hình có các hệ số biến đổi khác không có thể được đặt thành 0. Bộ phận mã hóa được mã hóa trong phương thức nhảy hoặc bộ phận mã hóa có hệ số biến đổi được đặt thành 0 có thể được gọi là bộ phận mã hóa không ranh giới.

Theo lựa chọn, bằng cách so sánh ít nhất một trong số độ rộng hoặc độ cao của khối mã hóa được tạo ra bằng cách chia tách khối mép ranh giới không điển hình với trị số ngưỡng, việc liệu để đặt bộ phận mã hóa tương ứng như bộ phận mã hóa không ranh giới có thể được xác định. Theo một ví dụ, khối mã hóa trong đó ít nhất một trong số độ rộng hoặc độ cao của khối mã hóa nhỏ hơn trị số ngưỡng có thể được mã hóa trong phương thức nhảy, hoặc các hệ số biến đổi có thể được đặt thành 0.

Fig.10 là sơ đồ thể hiện khía cạnh mã hóa của khối gần kề đường viền hình ảnh. Giả sử rằng trị số ngưỡng là 8.

Khối mã hóa trong đó ít nhất một trong số độ rộng hoặc độ cao nhỏ hơn trị số ngưỡng trong số các khối mã hóa được tạo ra bằng cách chia tách khối mép ranh giới

không điền hình có thể được đặt là bộ phận mã hóa không ranh giới.

Ví dụ như, trong ví dụ được thể hiện trên Fig.10, khối mã hóa 4×16 , khối mã hóa 8×4 , và khối mã hóa 4×4 có thể được đặt là các bộ phận mã hóa không ranh giới. Do đó, các khối có thể được mã hóa trong phương thức nhảy, hoặc các hệ số biến đổi của các khối có thể được đặt thành 0.

Việc liệu phương thức nhảy hoặc hệ số biến đổi được đặt thành 0 có thể là selectively được ứng dụng cho khối mã hóa có độ rộng và độ cao lớn hơn hoặc bằng trị số ngưỡng. Để thực hiện được điều này, cờ biểu thị việc liệu phương thức nhảy được ứng dụng cho khối mã hóa hoặc cờ biểu thị việc liệu hệ số biến đổi được đặt thành 0 có thể được mã hóa và được báo hiệu.

Theo lựa chọn, chỉ các bộ phận mã hóa được tạo ra bằng cách chia tách cây nhị phân có thể được phép được đặt là các bộ phận mã hóa không ranh giới. Theo lựa chọn, chỉ các bộ phận mã hóa được tạo ra bằng cách chia tách cây tứ phân có thể được phép được đặt là các bộ phận mã hóa không ranh giới.

Dự báo liên ảnh là phương thức dự báo trong đó khối hiện tại được dự báo sử dụng thông tin về hình ảnh đứng trước. Theo một ví dụ, khối được đưa vào hình ảnh đứng trước và được đặt ở cùng vị trí với vị trí của khối hiện tại (dưới đây được gọi là khối được sắp xếp) có thể được đặt là khối dự báo của khối hiện tại. Khối dự báo được tạo ra dựa trên khối được đặt tại cùng một vị trí như vị trí của khối hiện tại sẽ được gọi là khối dự báo được sắp xếp.

Trong khi đó, khi vật có mặt trong hình ảnh đứng trước được chuyển đến vị trí khác trong hình ảnh hiện tại, khối hiện tại có thể được dự báo một cách hiệu quả sử dụng chuyển động của vật. Ví dụ như, khi hướng chuyển động và kích cỡ của vật có thể được tìm thấy bằng cách so sánh hình ảnh đứng trước với hình ảnh hiện tại, khối dự báo (hoặc hình ảnh dự báo) của khối hiện tại có thể được tạo ra xem xét thông tin chuyển động của vật. Trong phần dưới đây, khối dự báo được tạo ra sử dụng thông tin chuyển động có thể được gọi là khối dự báo chuyển động.

Khối còn lại có thể được tạo ra bằng cách trừ khối dự báo từ khối hiện tại. Lúc này, khi chuyển động của vật có mặt, có thể giảm năng lượng của khối còn lại, và theo đó có thể cải thiện hiệu năng nén của khối còn lại bằng cách sử dụng khối dự báo chuyển động thay vì khối dự báo được sắp xếp.

Như được mô tả trên đây, tạo ra khối dự báo sử dụng thông tin chuyển động có thể được gọi là dự báo bù chuyển động. Trong dự báo liên ảnh, khối dự báo thường có

thể được tạo ra dựa trên dự báo bù chuyển động.

Thông tin chuyển động có thể gồm ít nhất một trong số vectơ chuyển động, chỉ số hình ảnh tham chiếu, hướng dự báo, hoặc chỉ số trọng số hai hướng. Vectơ chuyển động biểu thị hướng chuyển động và kích cỡ của vật. Chỉ số hình ảnh tham chiếu định rõ hình ảnh tham chiếu của khối hiện tại trong số hình ảnh tham chiếu được đưa vào danh mục hình ảnh tham chiếu. Hướng dự báo biểu thị một trong số dự báo L0 đơn hướng, dự báo L1 đơn hướng, hoặc dự báo hai hướng (dự báo L0 và dự báo L1). Ít nhất một trong số thông tin chuyển động hướng L0 hoặc thông tin chuyển động hướng L1 có thể được sử dụng phụ thuộc vào hướng dự báo của khối hiện tại. Chỉ số trọng số hai hướng định rõ trọng số được ứng dụng cho khối dự báo L0 và trọng số được ứng dụng cho khối dự báo L1.

Fig.11 là lưu đồ của phương pháp dự báo liên ảnh theo một phương án của sáng chế này.

Như được thể hiện trên Fig.11, phương pháp dự báo liên ảnh gồm xác định phương thức dự báo liên ảnh của khối hiện tại (S1101), thông tin chuyển động thu được của khối hiện tại theo phương thức dự báo liên ảnh được xác định (S1102), và thực hiện dự báo bù chuyển động trên khối hiện tại dựa trên thông tin chuyển động thu được (S1103).

Ở đây, phương thức dự báo liên ảnh, là phương thức thể hiện các kỹ thuật khác nhau để xác định thông tin chuyển động của khối hiện tại, có thể gồm phương thức dự báo liên ảnh sử dụng thông tin chuyển động tịnh tiến và phương thức dự báo liên ảnh sử dụng thông tin chuyển động affin. Theo một ví dụ, phương thức dự báo liên ảnh sử dụng thông tin chuyển động tịnh tiến có thể gồm phương thức hợp nhất và vectơ chuyển động phương thức dự báo, và phương thức dự báo liên ảnh sử dụng thông tin chuyển động affin có thể gồm phương thức hợp nhất affin và phương thức dự báo vectơ chuyển động affin. Thông tin chuyển động của khối hiện tại có thể được xác định dựa trên thông tin được phân tách từ dòng bit hoặc các khối lân cận khối hiện tại theo phương thức dự báo liên ảnh.

Phương pháp dự báo liên ảnh sử dụng thông tin chuyển động tịnh tiến sẽ được mô tả chi tiết dưới đây.

Thông tin chuyển động của khối hiện tại có thể được suy ra từ thông tin chuyển động của khối không phải là khối hiện tại. Ở đây, khối kia có thể là khối được mã hóa hoặc được giải mã thông qua dự báo liên ảnh sớm hơn khối hiện tại. Đặt thông tin

chuyển động của khối hiện tại là giống như thông tin chuyển động của khối kia có thể được xác định là phương thức hợp nhất. Ngoài ra, đặt vector chuyển động của khối kia là trị số dự báo của vector chuyển động của khối hiện tại có thể được xác định là vector chuyển động phương thức dự báo.

Fig.12 là lưu đồ của quy trình suy ra thông tin chuyển động của khối hiện tại trong phương thức hợp nhất.

Ứng viên hợp nhất của khối hiện tại có thể được suy ra (S1201). Ứng viên hợp nhất của khối hiện tại có thể được suy ra từ khối được mã hóa hoặc được giải mã thông qua dự báo liên ảnh sớm hơn khối hiện tại.

Fig.13 là sơ đồ minh họa các khối ứng viên được sử dụng để suy ra ứng viên hợp nhất.

Các khối ứng viên có thể gồm ít nhất một trong số các khối lân cận gồm mẫu gần kề khối hiện tại hoặc các khối không lân cận gồm mẫu không gần kề khối hiện tại. Trong phần dưới đây, các mẫu để xác định các khối ứng viên được xác định là các mẫu tham chiếu. Ngoài ra, mẫu tham chiếu gần kề khối hiện tại sẽ được gọi là mẫu tham chiếu lân cận, và mẫu tham chiếu không gần kề khối hiện tại sẽ được gọi là mẫu tham chiếu không lân cận.

Mẫu tham chiếu lân cận có thể được đưa vào cột lân cận cột cực trái của khối hiện tại hoặc hàng lân cận hàng trên của khối hiện tại. Theo một ví dụ, khi hệ tọa độ của mẫu trên-trái của khối hiện tại là $(0, 0)$, ít nhất một trong số khối gồm mẫu tham chiếu được bố trí tại $(-1, H-1)$, khối gồm mẫu tham chiếu được bố trí tại $(W-1, -1)$, khối gồm mẫu tham chiếu được bố trí tại $(W, -1)$, khối gồm mẫu tham chiếu được bố trí tại $(-1, H)$, hoặc khối gồm mẫu tham chiếu được bố trí tại $(-1, -1)$ có thể được sử dụng là khối ứng viên. Như được thể hiện trên hình vẽ, các khối lân cận với các chỉ số là 0 đến 4 có thể được sử dụng làm các khối ứng viên.

Mẫu tham chiếu không lân cận biểu thị mẫu trong đó ít nhất một trong số khoảng cách trục x hoặc khoảng cách trục y từ mẫu tham chiếu gần kề khối hiện tại có trị số được xác định trước. Theo một ví dụ, ít nhất một trong số khối gồm mẫu tham chiếu trong đó khoảng cách trục x từ mẫu tham chiếu trái có trị số được xác định trước, khối gồm mẫu không lân cận trong đó khoảng cách trục y từ mẫu tham chiếu trên có trị số được xác định trước, hoặc khối gồm mẫu không lân cận trong đó khoảng cách trục x và khoảng cách trục y từ mẫu tham chiếu trên-trái có trị số được xác định trước có thể được sử dụng là khối ứng viên. Trị số được xác định trước có thể là số tự nhiên chẳng hạn như

4, 8, 12, và 16. Như được thể hiện trên hình vẽ, ít nhất một trong số các khối với các chỉ số của 5 đến 26 có thể được sử dụng là khối ứng viên.

Mẫu không được bố trí trên cùng một đường dọc, đường ngang, hoặc đường xiên như đường của mẫu tham chiếu lân cận có thể được đặt là mẫu tham chiếu không lân cận.

Fig.14 là sơ đồ thể hiện các vị trí của các mẫu tham chiếu.

Như trong ví dụ được thể hiện trên Fig.14, hệ tọa độ x của các mẫu tham chiếu không lân cận trên có thể được đặt là khác với hệ tọa độ x của các mẫu tham chiếu lân cận trên. Theo một ví dụ, khi vị trí của mẫu tham chiếu lân cận trên là (W-1, -1), vị trí của mẫu tham chiếu không lân cận trên được đặt nằm cách xa từ mẫu tham chiếu lân cận trên một khoảng N trong trục y được đặt thành ((W/2)-1, -1-N), và vị trí của mẫu tham chiếu không lân cận trên được đặt nằm cách xa từ mẫu tham chiếu lân cận trên trong trục y một khoảng 2N có thể được đặt thành (0, -1-2N). Tức là, vị trí của mẫu tham chiếu không gần kề có thể được xác định dựa trên vị trí của mẫu tham chiếu gần kề và khoảng cách từ mẫu tham chiếu gần kề.

Trong phần dưới đây, trong số các khối ứng viên, khối ứng viên gồm mẫu tham chiếu lân cận sẽ được gọi là khối lân cận, và khối gồm mẫu tham chiếu không lân cận sẽ được gọi là khối không lân cận.

Khi khoảng cách giữa khối hiện tại và khối ứng viên lớn hơn hoặc bằng trị số ngưỡng, khối ứng viên có thể được đặt là không sẵn có là ứng viên hợp nhất. Trị số ngưỡng có thể được xác định dựa trên kích cỡ của đơn vị cây mã hóa. Theo một ví dụ, trị số ngưỡng có thể được đặt thành độ cao của đơn vị cây mã hóa (ctu_height) hoặc độ cao của đơn vị cây mã hóa cộng hoặc trừ độ lệch (ví dụ như, $\text{ctu_height} \pm N$). Độ lệch N, là trị số được xác định trước trong bộ mã hóa hoặc bộ giải mã, có thể được đặt thành 4, 8, 16, 32, hoặc ctu_height.

Khi khoảng cách giữa tọa độ y của khối hiện tại và tọa độ y của mẫu được đưa vào khối ứng viên lớn hơn trị số ngưỡng, nội dung có thể được xác định là khối ứng viên không sẵn có là ứng viên hợp nhất.

Theo lựa chọn, khối ứng viên không thuộc về đơn vị cây mã hóa giống như khối hiện tại có thể được đặt là không sẵn có là ứng viên hợp nhất. Theo một ví dụ, khi mẫu tham chiếu chệch từ mép trên của đơn vị cây mã hóa mà khối hiện tại thuộc về, khối ứng viên gồm mẫu tham chiếu có thể được đặt là không sẵn có là ứng viên hợp nhất.

Khi mép trên của khối hiện tại là gần kề mép trên của đơn vị cây mã hóa, nội dung có thể được xác định là nhiều khối ứng viên không sẵn có là các ứng viên hợp nhất,

và hiệu suất mã hóa hoặc giải mã của khối hiện tại có thể được giảm. Để giải quyết vấn đề như vậy, các khối ứng viên có thể được đặt sao cho số lượng của các khối ứng viên được bố trí về phía trái của khối hiện tại là lớn hơn số lượng của các khối ứng viên được bố trí phía trên khối hiện tại.

Fig.15 là sơ đồ minh họa các khối ứng viên được sử dụng để suy ra ứng viên hợp nhất.

Như trong ví dụ được thể hiện trên Fig.15, các khối trên thuộc về N khối trên các cột của khối hiện tại và các khối trái thuộc về M cột khối trái của khối hiện tại có thể được đặt là các khối ứng viên. Trong trường hợp này, bằng cách đặt M lớn hơn N, số lượng các khối ứng viên trái có thể được đặt là lớn hơn số lượng các khối ứng viên trên.

Ví dụ như, có thể thực hiện việc đặt sao cho hiệu số giữa tọa độ y của mẫu tham chiếu trong khối hiện tại và tọa độ y của khối trên sẵn có là khối ứng viên không vượt quá N lần độ cao của khối hiện tại. Ngoài ra, có thể thực hiện việc đặt sao cho hiệu số giữa tọa độ x của mẫu tham chiếu trong khối hiện tại và tọa độ x của khối trái sẵn có là khối ứng viên không vượt quá M lần độ rộng của khối hiện tại.

Ví dụ như, nội dung được thể hiện trên Fig.15 là các khối thuộc về hai cột khối trên của khối hiện tại và các khối thuộc về năm cột khối trái của khối hiện tại được đặt là các khối ứng viên.

Theo một ví dụ khác, khi khối ứng viên không thuộc về đơn vị cây mã hóa mẫu là khối hiện tại, ứng viên hợp nhất có thể được suy ra sử dụng khối thuộc về đơn vị cây mã hóa giống như khối hiện tại hoặc khối gồm mẫu tham chiếu gần kề mép của đơn vị cây mã hóa thay vì khối ứng viên.

Fig.16 là sơ đồ thể hiện ví dụ trong đó vị trí của mẫu tham chiếu bị thay đổi.

Khi mẫu tham chiếu được đưa vào đơn vị cây mã hóa khác với mẫu tham chiếu của khối hiện tại và không gần kề mép của đơn vị cây mã hóa, khối ứng viên có thể được xác định sử dụng mẫu tham chiếu gần kề mép của đơn vị cây mã hóa thay vì mẫu tham chiếu.

Ví dụ như, trong các ví dụ được thể hiện trên Fig.16A và Fig.16B, khi mép trên của khối hiện tại là tiếp giáp với mép trên của đơn vị cây mã hóa, các mẫu tham chiếu được bố trí phía trên khối hiện tại thuộc về đơn vị cây mã hóa khác với that của khối hiện tại. Mẫu tham chiếu không gần kề mép trên của đơn vị cây mã hóa trong số các mẫu tham chiếu thuộc về đơn vị cây mã hóa khác với mẫu tham chiếu của khối hiện tại có thể bị thay thế bằng mẫu gần kề mép trên của đơn vị cây mã hóa.

Ví dụ như, mẫu tham chiếu tại vị trí #6 có thể bị thay thế bằng mẫu tại vị trí #6' được đặt tại mép trên của đơn vị cây mã hóa, như trong ví dụ được thể hiện trên Fig.16A, và mẫu tham chiếu tại vị trí #15 có thể bị thay thế bằng mẫu tại vị trí #15' được đặt tại mép trên của đơn vị cây mã hóa, như trong ví dụ được thể hiện trên Fig.16B. Trong trường hợp này, tọa độ y của mẫu thay thế có thể bị thay đổi thành tọa độ của vị trí gần kề đơn vị cây mã hóa, và tọa độ x của mẫu thay thế có thể được đặt là giống như vị trí của mẫu tham chiếu. Theo một ví dụ, mẫu tại vị trí #6' có thể có cùng tọa độ x như mẫu tại vị trí #6, và mẫu tại vị trí #15' có thể có cùng tọa độ x như mẫu tại vị trí #15.

Theo lựa chọn, trị số thu được bằng cách cộng hoặc trừ độ lệch vào hoặc từ tọa độ x của mẫu tham chiếu có thể được đặt là tọa độ x của mẫu thay thế. Theo một ví dụ, khi mẫu tham chiếu không lân cận và mẫu tham chiếu lân cận được bố trí phía trên khối hiện tại có cùng tọa độ x, trị số thu được bằng cách cộng hoặc trừ độ lệch vào hoặc từ tọa độ x của mẫu tham chiếu có thể được đặt là tọa độ x của mẫu thay thế. Điều này nhằm ngăn mẫu thay thế thay thế mẫu tham chiếu không lân cận từ được đặt tại cùng một vị trí như một mẫu tham chiếu không lân cận hoặc mẫu tham chiếu lân cận khác.

Fig.17 là sơ đồ thể hiện ví dụ trong đó vị trí của mẫu tham chiếu bị thay đổi.

Khi mẫu tham chiếu mà được đưa vào đơn vị cây mã hóa khác với mẫu tham chiếu của khối hiện tại và không gần kề mép của đơn vị cây mã hóa bị thay thế bằng mẫu được bố trí tại mép của đơn vị cây mã hóa, trị số thu được bằng cách cộng hoặc trừ độ lệch vào hoặc từ tọa độ x của mẫu tham chiếu có thể được đặt là tọa độ x của mẫu thay thế.

Ví dụ như, trong ví dụ được thể hiện trên Fig.17, mẫu tham chiếu tại vị trí #6 và mẫu tham chiếu tại vị trí #15 có thể bị thay thế bằng mẫu tại vị trí #6' và mẫu tại vị trí #15', các mẫu này có tọa độ y giống như hàng gần kề mép trên của đơn vị cây mã hóa. Trong trường hợp này, tọa độ x của mẫu tại vị trí #6' có thể được đặt thành trị số thu được bằng cách trừ W/2 từ tọa độ x của mẫu tham chiếu tại vị trí #6, và tọa độ x của mẫu tại vị trí #15' có thể được đặt thành trị số thu được bằng cách trừ W-1 từ tọa độ x của mẫu tham chiếu tại vị trí #15.

Không giống như các ví dụ được thể hiện trên Fig.16 và Fig.17, tọa độ y của hàng được bố trí phía trên hàng đỉnh của khối hiện tại hoặc tọa độ y của mép trên của bộ phận cây mã hóa có thể được đặt thành tọa độ y của mẫu thay thế.

Mặc dù không được thể hiện, mẫu thay thế mẫu tham chiếu có thể được xác định dựa trên mép trái của đơn vị cây mã hóa. Theo một ví dụ, khi mẫu tham chiếu không

được đưa vào đơn vị cây mã hóa giống như đơn vị cây mã hóa của khối hiện tại và không gần kề mép trái của đơn vị cây mã hóa, mẫu tham chiếu có thể bị thay thế bằng mẫu gần kề mép trái của đơn vị cây mã hóa. Trong trường hợp này, mẫu thay thế có thể có tọa độ y giống như mẫu tham chiếu và có thể có tọa độ x thu được bằng cách cộng hoặc trừ độ lệch vào hoặc từ tọa độ y của mẫu tham chiếu.

Sau đó, khối gồm mẫu thay thế có thể được đặt là khối ứng viên, và ứng viên hợp nhất của khối hiện tại có thể được suy ra dựa trên khối ứng viên.

Ứng viên hợp nhất có thể được suy ra từ khối lân cận thời gian được đưa vào hình ảnh khác với hình ảnh của khối hiện tại. Theo một ví dụ, ứng viên hợp nhất có thể được suy ra từ khối được sắp xếp được đưa vào hình ảnh được sắp xếp.

Thông tin chuyển động của ứng viên hợp nhất có thể được đặt là giống như thông tin chuyển động của khối ứng viên. Theo một ví dụ, ít nhất một trong số vectơ chuyển động, chỉ số hình ảnh tham chiếu, hướng dự báo, hoặc chỉ số trọng số hai hướng của khối ứng viên có thể được đặt là thông tin chuyển động của ứng viên hợp nhất.

Danh mục ứng viên hợp nhất gồm các ứng viên hợp nhất có thể được tạo ra (S1202). Các ứng viên hợp nhất có thể được phân loại thành các ứng viên hợp nhất gần kề được suy ra từ các khối lân cận gần kề khối hiện tại và các ứng viên hợp nhất không gần kề được suy ra từ các khối không lân cận.

Các chỉ số có thể được gán cho các ứng viên hợp nhất của danh mục ứng viên hợp nhất theo thứ tự được xác định trước. Theo một ví dụ, chỉ số được gán cho ứng viên hợp nhất gần kề có thể có trị số nhỏ hơn chỉ số được gán cho ứng viên hợp nhất không gần kề. Theo lựa chọn, chỉ số có thể được gán cho mỗi trong số các ứng viên hợp nhất dựa trên các chỉ số của các khối được thể hiện trên Fig.13 hoặc Fig.15.

Khi nhiều ứng viên hợp nhất được đưa vào ứng viên hợp nhất, ít nhất một trong số nhiều ứng viên hợp nhất có thể được chọn (S1203). Trong trường hợp này, thông tin biểu thị việc liệu thông tin chuyển động của khối hiện tại được suy ra từ ứng viên hợp nhất gần kề có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit. Theo một ví dụ, phần tử cú pháp `isAdjacentMergeFlag`, điều này biểu thị việc liệu thông tin chuyển động của khối hiện tại được suy ra từ ứng viên hợp nhất gần kề, có thể được báo hiệu trong dòng bit. Khi trị số của phần tử cú pháp `isAdjacentMergeFlag` là 1, thông tin chuyển động của khối hiện tại có thể được suy ra dựa trên ứng viên hợp nhất gần kề. Mặt khác, khi trị số của phần tử cú pháp `isAdjacentMergeFlag` là 0, thông tin chuyển động của khối hiện tại có thể được suy ra dựa trên ứng viên hợp nhất không gần kề.

Bảng 1 thể hiện bảng cú pháp gồm phần tử cú pháp isAdjacentMergeFlag.

[Bảng 1]

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Bộ mô tả
if(slice_type != I) {	
pred_mode_flag	ae(v)
}	
if(CuPredMode[x0][y0] == MODE_INTRA) {	
if(treeType == SINGLE_TREE treeType ==	
DUAL_TREE_LUMA) {	
intra_luma_mpm_flag[x0][y0]	
if(intra_luma_mpm_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ae(v)
else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
if(treeType == SINGLE_TREE treeType ==	
DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ae(v)
} else {	
if(cu_skip_flag[x0][y0]) {	
if(MaxNumMergeCand > 1){	
isAdjacentMergeflag	ae(v)
if(isAdjacentMergeflag){	
merge_idx[x0][y0]	ae(v)
} else{	
NA_merge_idx[x0][y0]	ae(v)
}	
}	
} else { /* MODE_INTER*/	
merge_flag[x0][y0]	ae(v)
if(merge_flag[x0][y0]){	
if(MaxNumMergeCand > 1){	

isAdjacentMergeflag	ae(v)
if (isAdjcanetMergeflag){	
merge_idx[x0][y0]	ae(v)
} else{	
NA_merge_idx[x0][y0]	ae(v)
}	
}	
}	
if(CuPredMode[x0][y0] != MODE_INTRA)	
cu_cbf	ae(v)
if(cu_cbf) {	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	

Thông tin để định rõ một trong số nhiều ứng viên hợp nhất có thể được báo hiệu trong dòng bit. Theo một ví dụ, thông tin biểu thị chỉ số của một trong số các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất có thể được báo hiệu trong dòng bit.

Khi isAdjacentMergeflag là 1, phần tử cú pháp merge_idx, phần tử cú pháp này là để định rõ một trong số các ứng viên hợp nhất gần kề, có thể được báo hiệu. Trị số cực đại của phần tử cú pháp merge_idx có thể được đặt thành trị số thu được bằng cách trừ one từ số lượng của các ứng viên hợp nhất gần kề.

Khi isAdjacentMergeflag là 0, phần tử cú pháp NA_merge_idx, phần tử cú pháp này dùng để định rõ một trong số các ứng viên hợp nhất không gần kề, có thể được báo hiệu. Phần tử cú pháp NA_merge_idx thể hiện trị số thu được bằng cách trừ số lượng của các ứng viên hợp nhất gần kề từ chỉ số của ứng viên hợp nhất không gần kề. Bộ giải mã có thể chọn ứng viên hợp nhất không gần kề bằng cách thêm số lượng của các ứng viên hợp nhất gần kề vào chỉ số được định rõ bởi NA_merge_idx.

Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất nhỏ hơn trị số lớn nhất, ứng viên hợp nhất được đưa vào bảng thông tin chuyển động liên vùng có thể được thêm vào danh mục ứng viên hợp nhất. Bảng thông tin chuyển động liên vùng có thể gồm ứng viên hợp nhất được suy ra dựa trên khối được mã hóa hoặc được giải mã sớm hơn khối hiện tại.

Bảng thông tin chuyển động liên vùng gồm ứng viên hợp nhất được suy ra từ khối

được mã hóa hoặc được giải mã dựa trên dự báo liên ảnh trong hình ảnh hiện tại. Theo một ví dụ, thông tin chuyển động của ứng viên hợp nhất được đưa vào bảng thông tin chuyển động liên vùng có thể được đặt là giống như thông tin chuyển động của khối được mã hóa hoặc được giải mã dựa trên dự báo liên ảnh. Ở đây, thông tin chuyển động có thể gồm ít nhất một trong số vector chuyển động, chỉ số hình ảnh tham chiếu, hướng dự báo, hoặc chỉ số trọng số hai hướng.

Đề thuận tiện cho việc mô tả, ứng viên hợp nhất được đưa vào bảng thông tin chuyển động liên vùng sẽ được gọi là ứng viên hợp nhất liên vùng.

Số lượng tối đa của các ứng viên hợp nhất có thể được đưa vào bảng thông tin chuyển động liên vùng có thể được xác định trước trong bộ mã hóa và bộ giải mã. Theo một ví dụ, số lượng tối đa của các ứng viên hợp nhất có thể được đưa vào bảng thông tin chuyển động liên vùng có thể là 1, 2, 3, 4, 5, 6, 7, 8, hoặc nhiều hơn (ví dụ như, 16).

Theo lựa chọn, thông tin biểu thị số lượng tối đa của các ứng viên hợp nhất của bảng thông tin chuyển động liên vùng có thể được báo hiệu trong dòng bit. Thông tin có thể được báo hiệu ở cấp độ trình tự, hình ảnh, hoặc phiên.

Theo lựa chọn, số lượng tối đa của các ứng viên hợp nhất của bảng thông tin chuyển động liên vùng có thể được xác định phụ thuộc vào kích cỡ của hình ảnh, kích cỡ của phiên, hoặc kích cỡ của đơn vị cây mã hóa.

Bảng thông tin chuyển động liên vùng có thể được khởi tạo trong các đơn vị của các hình ảnh, phiên, gạch, đơn vị cây mã hóa, hoặc đường đơn vị cây mã hóa (các hàng và các cột). Theo một ví dụ, khi phiên được khởi tạo, bảng thông tin chuyển động liên vùng cũng được khởi tạo và có thể gồm ứng viên không hợp nhất.

Theo lựa chọn, thông tin biểu thị việc liệu có nên khởi tạo bảng thông tin chuyển động liên vùng có thể được báo hiệu trong dòng bit. Thông tin có thể được báo hiệu ở cấp độ phiên, ngôi, gạch, hoặc khối. Bảng thông tin chuyển động liên vùng được tạo cấu trúc trước có thể được sử dụng cho đến khi thông tin biểu thị sự khởi tạo bảng thông tin chuyển động liên vùng.

Theo lựa chọn, thông tin liên quan đến ứng viên hợp nhất liên vùng ban đầu có thể được báo hiệu thông qua tập hợp thông số hình ảnh hoặc tiêu đề phiên. Thậm chí mặc dù phiên được khởi tạo, bảng thông tin chuyển động liên vùng có thể gồm ứng viên hợp nhất liên vùng ban đầu. Do đó, ứng viên hợp nhất liên vùng có thể được sử dụng cho khối cần được mã hóa hoặc được giải mã trước tiên trong phiên.

Các khối có thể được mã hóa hoặc được giải mã theo thứ tự mã hóa hoặc giải mã,

và khối được mã hóa hoặc được giải mã dựa trên dự báo liên ảnh có thể được đặt tuần tự là các ứng viên hợp nhất liên vùng theo thứ tự mã hóa hoặc giải mã.

Fig.18 là sơ đồ để mô tả khía cạnh cập nhật của bảng thông tin chuyển động liên vùng.

Khi dự báo liên ảnh được thực hiện trên khối hiện tại (S1801), ứng viên hợp nhất liên vùng có thể được suy ra dựa trên khối hiện tại (S1802). Thông tin chuyển động của ứng viên hợp nhất liên vùng có thể được đặt là giống như thông tin chuyển động của khối hiện tại.

Khi bảng thông tin chuyển động liên vùng là trống rỗng (S1803), ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại có thể được thêm vào bảng thông tin chuyển động liên vùng (S1804).

Khi bảng thông tin chuyển động liên vùng đã bao gồm ứng viên hợp nhất liên vùng (S1803), việc kiểm tra dư thừa có thể được thực hiện trên thông tin chuyển động của khối hiện tại (hoặc ứng viên hợp nhất liên vùng được suy ra dựa trên thông tin chuyển động) (S1805). Kiểm tra dư thừa dùng để xác định việc liệu thông tin chuyển động của khối hiện tại là giống như thông tin chuyển động của ứng viên hợp nhất liên vùng được lưu trữ trước trong bảng thông tin chuyển động liên vùng. Kiểm tra dư thừa có thể được thực hiện trên tất cả các ứng viên hợp nhất liên vùng được lưu trữ trước trong bảng thông tin chuyển động liên vùng. Theo lựa chọn, việc kiểm tra dư thừa có thể được thực hiện trên các ứng viên hợp nhất liên vùng với các chỉ số lớn hơn hoặc bằng trị số ngưỡng hoặc các chỉ số nhỏ hơn hoặc bằng trị số ngưỡng trong số các ứng viên hợp nhất liên vùng được lưu trữ trước trong bảng thông tin chuyển động liên vùng.

Khi không dự báo liên ảnh ứng viên hợp nhất nào có thông tin chuyển động giống như khối hiện tại được đưa vào, ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại có thể được thêm vào bảng thông tin chuyển động liên vùng (S1808). Việc liệu các ứng viên hợp nhất dự báo liên ảnh có giống nhau hay không có thể được xác định dựa trên việc liệu các ứng viên hợp nhất dự báo liên ảnh có cùng thông tin chuyển động (ví dụ như, vector chuyển động và/hoặc chỉ số hình ảnh tham chiếu).

Trong trường hợp này, khi số lượng tối đa của các ứng viên hợp nhất liên vùng được lưu trữ trước trong bảng thông tin chuyển động liên vùng (S1806), ứng viên hợp nhất liên vùng cũ nhất bị xóa bỏ (S1807), và ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại có thể được thêm vào bảng thông tin chuyển động liên vùng (S1808).

Các ứng viên hợp nhất liên vùng có thể được nhận dạng bởi các chỉ số của chúng. Khi ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại được thêm vào bảng thông tin chuyển động liên vùng, chỉ số thấp nhất (ví dụ như, 0) được gán cho ứng viên hợp nhất liên vùng, và các chỉ số của các ứng viên hợp nhất liên vùng được lưu trữ trước có thể được tăng lên một. Trong trường hợp này, khi số lượng tối đa của các ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước trong bảng thông tin chuyển động liên vùng, ứng viên hợp nhất liên vùng với chỉ số lớn nhất bị loại bỏ.

Theo lựa chọn, khi ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại được thêm vào bảng thông tin chuyển động liên vùng, chỉ số lớn nhất có thể được gán cho ứng viên hợp nhất liên vùng. Theo một ví dụ, khi số lượng các ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước trong bảng thông tin chuyển động liên vùng là nhỏ hơn trị số lớn nhất, chỉ số có cùng trị số như số lượng của các ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước có thể được gán cho ứng viên hợp nhất liên vùng. Theo lựa chọn, khi số lượng các ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước trong bảng thông tin chuyển động liên vùng bằng trị số lớn nhất, chỉ số có trị số lớn nhất trừ 1 có thể được gán cho ứng viên hợp nhất liên vùng. Ngoài ra, ứng viên hợp nhất liên vùng với chỉ số nhỏ nhất bị loại bỏ, và các chỉ số của các ứng viên hợp nhất liên vùng được lưu trữ trước kia bị giảm đi 1.

Fig.19 là sơ đồ thể hiện khía cạnh cập nhật của bảng ứng viên hợp nhất liên vùng.

Do ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại được thêm vào bảng ứng viên hợp nhất liên vùng, giả sử rằng chỉ số lớn nhất được gán cho ứng viên hợp nhất liên vùng. Ngoài ra, giả sử rằng số lượng tối đa của các ứng viên hợp nhất liên vùng được lưu trữ trong bảng ứng viên hợp nhất liên vùng.

Khi ứng viên hợp nhất liên vùng $HmvpCand[n+1]$, được suy ra từ khối hiện tại, được thêm vào bảng ứng viên hợp nhất liên vùng $HmvpCandList$, ứng viên hợp nhất liên vùng $HmvpCand[0]$, ứng viên này có chỉ số nhỏ nhất trong số các ứng viên hợp nhất liên vùng được lưu trữ trước, có thể bị xóa bỏ, và các chỉ số của các ứng viên hợp nhất liên vùng khác có thể bị giảm đi 1. Ngoài ra, chỉ số của ứng viên hợp nhất liên vùng $HmvpCand[n+1]$, được suy ra từ khối hiện tại, có thể được đặt là trị số lớn nhất (n trong ví dụ được thể hiện trên Fig.19).

Khi ứng viên hợp nhất liên vùng giống với ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại được lưu trữ trước (S1805), ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại có thể không được thêm vào bảng thông tin chuyển động

liên vùng (S1809).

Theo lựa chọn, trong khi ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại được thêm vào bảng thông tin chuyển động liên vùng, ứng viên hợp nhất liên vùng được lưu trữ trước giống với ứng viên hợp nhất liên vùng có thể bị loại bỏ. Điều này dẫn đến tác dụng giống như việc cập nhật các chỉ số của các ứng viên hợp nhất liên vùng được lưu trữ trước.

Fig.20 là sơ đồ thể hiện ví dụ trong đó chỉ số của ứng viên hợp nhất liên vùng được cập nhật được lưu trữ trước.

Khi chỉ số của ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước giống với ứng viên hợp nhất liên vùng mvCand, được suy ra dựa trên khối hiện tại, là hIdx, ứng viên hợp nhất dự báo liên ảnh được lưu trữ trước có thể bị xóa bỏ, và các chỉ số của các ứng viên hợp nhất dự báo liên ảnh lớn hơn hIdx có thể bị giảm đi 1. Theo một ví dụ, nội dung được thể hiện trên Fig.20 là HmvpCand[2], chỉ số này giống với mvCand, bị xóa bỏ khỏi bảng thông tin chuyển động liên vùng HmvpCandList, và các chỉ số HmvpCand[3] đến HmvpCand[n] bị giảm đi 1.

Ngoài ra, ứng viên hợp nhất liên vùng mvCand, được suy ra dựa trên khối hiện tại, có thể được thêm vào phần cuối của bảng thông tin chuyển động liên vùng.

Theo lựa chọn, chỉ số được gán cho ứng viên hợp nhất liên vùng được lưu trữ trước giống với ứng viên hợp nhất liên vùng được suy ra dựa trên khối hiện tại có thể được cập nhật. Ví dụ như, chỉ số của ứng viên hợp nhất liên vùng được lưu trữ trước có thể bị thay đổi thành trị số nhỏ nhất hoặc trị số lớn nhất.

Có thể thực hiện việc đặt sao cho thông tin chuyển động của các khối được đưa vào khu vực được xác định trước không được thêm vào bảng thông tin chuyển động liên vùng. Theo một ví dụ, ứng viên hợp nhất liên vùng được suy ra dựa trên thông tin chuyển động của khối được đưa vào khu vực hợp nhất song song có thể không được thêm vào bảng thông tin chuyển động liên vùng. Do thứ tự mã hóa hoặc giải mã không được xác định cho khối được đưa vào khu vực hợp nhất song song, không thích hợp để sử dụng thông tin chuyển động của một trong số các khối để thực hiện dự báo liên ảnh trên các khối khác. Do đó, các ứng viên hợp nhất liên vùng được suy ra dựa trên khối được đưa vào khu vực hợp nhất song song có thể không được thêm vào bảng thông tin chuyển động liên vùng.

Khi dự báo bù chuyển động được thực hiện trong các đơn vị của các khối con, ứng viên hợp nhất liên vùng có thể được suy ra dựa trên thông tin chuyển động của khối

con đại diện trong số nhiều khối con được đưa vào khối hiện tại. Theo một ví dụ, khi ứng viên hợp nhất khối con được sử dụng cho khối hiện tại, ứng viên hợp nhất liên vùng có thể được suy ra dựa trên thông tin chuyển động của khối con đại diện trong số các khối con.

Các vector chuyển động của các khối con có thể được suy ra theo thứ tự sau đây. Trước hết, một ứng viên hợp nhất có thể được chọn từ trong số các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất của khối hiện tại, và vector dịch chuyển ban đầu $shVector$ có thể được suy ra dựa trên vector chuyển động của ứng viên hợp nhất được chọn. Khi đó, khối con dịch chuyển có mẫu tham chiếu được bố trí tại $(xColSb, yColSb)$ có thể được suy ra bằng cách thêm vector dịch chuyển ban đầu đến vị trí (xSb, ySb) của mẫu tham chiếu (ví dụ như, mẫu trên-trái hoặc mẫu giữa) của mỗi khối con trong khối mã hóa. Phương trình 1 dưới đây biểu thị phương trình để suy ra khối con dịch chuyển.

[Phương trình 1]

$$(xColSb, yColSb) = (xSb + shVector[0] \gg 4, ySb + shVector[1] \gg 4)$$

Khi đó, vector chuyển động của khối được sắp xếp tương ứng với vị trí trung tâm của khối con gồm $(xColSb, yColSb)$ có thể được đặt là vector chuyển động của khối con gồm (xSb, ySb) .

Khối con đại diện có thể đề cập đến khối con gồm mẫu trên-trái hoặc mẫu trung tâm của khối hiện tại.

Fig.21 là sơ đồ thể hiện vị trí của khối con đại diện.

Fig.21A thể hiện ví dụ trong đó khối con được bố trí trên góc trên-trái của khối hiện tại, và Fig.21B thể hiện ví dụ trong đó khối con được bố trí tại trung tâm của khối hiện tại được đặt là khối con đại diện. Khi dự báo bù chuyển động được thực hiện trong các đơn vị của các khối con, ứng viên hợp nhất liên vùng của khối hiện tại có thể được suy ra dựa trên vector chuyển động của khối con gồm mẫu trên-trái của khối hiện tại hoặc khối con gồm mẫu trung tâm của khối hiện tại.

Việc liệu có sử dụng khối hiện tại làm ứng viên hợp nhất liên vùng cũng có thể được xác định dựa trên phương thức dự báo liên ảnh của khối hiện tại. Theo một ví dụ, khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affine có thể được đặt là không sẵn có làm ứng viên hợp nhất liên vùng. Do đó, mặc dù khối hiện tại được mã hóa hoặc được giải mã thông qua dự báo liên ảnh, bảng thông tin chuyển động dự báo liên ảnh có thể không được cập nhật dựa trên khối hiện tại khi phương thức dự báo liên ảnh của khối hiện tại là phương thức dự báo affine.

Theo lựa chọn, ứng viên hợp nhất liên vùng có thể được suy ra dựa trên khối con vector của ít nhất một khối con được đưa vào khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affin. Theo một ví dụ, ứng viên hợp nhất liên vùng có thể được suy ra bằng cách sử dụng khối con được bố trí trên góc trên-trái của khối hiện tại, khối con được bố trí tại trung tâm của khối hiện tại, hoặc khối con được bố trí trên góc trên-phải của khối hiện tại. Theo lựa chọn, trị số trung bình của các vector khối con của nhiều khối con có thể được đặt là vector chuyển động của ứng viên hợp nhất liên vùng.

Theo lựa chọn, ứng viên hợp nhất liên vùng có thể được suy ra dựa trên trị số trung bình của các vector hạt affin của khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affin. Theo một ví dụ, trị số trung bình của ít nhất một trong số vector hạt affin thứ nhất, vector hạt affin thứ hai, hoặc vector hạt affin thứ ba của khối hiện tại có thể được đặt là vector chuyển động của ứng viên hợp nhất liên vùng.

Theo lựa chọn, bảng thông tin chuyển động liên vùng có thể được tạo cấu hình cho mỗi phương thức dự báo liên ảnh. Theo một ví dụ, ít nhất một trong số bảng thông tin chuyển động liên vùng cho khối được mã hóa hoặc được giải mã bằng sao chép nội khối, bảng thông tin chuyển động liên vùng cho khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động tịnh tiến, hoặc bảng thông tin chuyển động liên vùng cho khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affin có thể được xác định. Một trong số nhiều bảng thông tin chuyển động liên vùng có thể được chọn theo phương thức dự báo liên ảnh của khối hiện tại.

Fig.22 thể hiện ví dụ trong đó bảng thông tin chuyển động liên vùng được tạo ra cho mỗi phương thức dự báo liên ảnh.

Khi khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động không affin, ứng viên hợp nhất liên vùng mvCand, được suy ra dựa trên khối, có thể được thêm vào bảng thông tin chuyển động không affin liên vùng HmvpCandList. Mặt khác, khi khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affin, ứng viên hợp nhất liên vùng mvAfCand, được suy ra dựa trên khối, có thể được thêm vào bảng thông tin chuyển động affin liên vùng HmvpAfCandList.

Các vector hạt affin của khối có thể được lưu trữ trong ứng viên hợp nhất liên vùng được suy ra từ khối được mã hóa hoặc được giải mã dựa trên mô hình chuyển động affin. Do đó, ứng viên hợp nhất liên vùng có thể được sử dụng là ứng viên hợp nhất để suy ra vector hạt affin của khối hiện tại.

Bảng thông tin chuyển động liên vùng bổ sung có thể được xác định ngoài bảng

thông tin chuyển động liên vùng được mô tả trên đây ra. Ngoài bảng thông tin chuyển động liên vùng được mô tả trên đây (dưới đây được gọi là bảng thông tin chuyển động liên vùng thứ nhất), bảng thông tin chuyển động dài hạn (dưới đây được gọi là bảng thông tin chuyển động liên vùng thứ hai) có thể được xác định. Ở đây, bảng thông tin chuyển động dài hạn gồm các ứng viên hợp nhất dài hạn.

Khi cả bảng thông tin chuyển động liên vùng thứ nhất và bảng thông tin chuyển động liên vùng thứ hai là trống rỗng, các ứng viên hợp nhất liên vùng có thể được ưu tiên thêm vào bảng thông tin chuyển động liên vùng thứ hai. Cho đến khi số lượng của các ứng viên hợp nhất liên vùng được phép cho bảng thông tin chuyển động liên vùng thứ hai đạt được số lượng tối đa, các ứng viên hợp nhất liên vùng không thể được thêm vào bảng thông tin chuyển động liên vùng thứ nhất.

Theo lựa chọn, một dự báo liên ảnh ứng viên hợp nhất có thể được thêm vào cả bảng thông tin chuyển động liên vùng thứ nhất và bảng thông tin chuyển động liên vùng thứ hai.

Trong trường hợp này, bảng thông tin chuyển động liên vùng thứ hai mà ở đó cấu hình được hoàn thành không thể thực hiện việc cập nhật được nữa. Theo lựa chọn, khi khu vực được giải mã lớn hơn hoặc bằng tỷ lệ được xác định trước của phiên, bảng thông tin chuyển động liên vùng thứ hai có thể được cập nhật. Theo lựa chọn, bảng thông tin chuyển động liên vùng thứ hai có thể được cập nhật mọi N đường đơn vị cây mã hóa.

Mặt khác, bảng thông tin chuyển động liên vùng thứ nhất có thể được cập nhật bất cứ khi nào khối được mã hóa hoặc được giải mã thông qua dự báo liên ảnh được tạo ra. Tuy nhiên, có thể thực hiện việc đặt sao cho các ứng viên hợp nhất liên vùng được thêm vào bảng thông tin chuyển động liên vùng thứ hai không được sử dụng để cập nhật bảng thông tin chuyển động liên vùng thứ nhất.

Thông tin để chọn một trong số bảng thông tin chuyển động liên vùng thứ nhất hoặc bảng thông tin chuyển động liên vùng thứ hai có thể được báo hiệu trong dòng bit. Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất nhỏ hơn trị số lớn nhất, các ứng viên hợp nhất được đưa vào bảng thông tin chuyển động liên vùng được biểu thị bởi thông tin có thể được thêm vào danh mục ứng viên hợp nhất.

Theo lựa chọn, bảng thông tin chuyển động liên vùng có thể được chọn dựa trên kích cỡ, hình dạng, hoặc phương thức dự báo liên ảnh của khối hiện tại, sự có mặt của dự báo hai hướng, sự có mặt của lọc vector chuyển động, hoặc sự có mặt của chia tách hình tam giác.

Theo lựa chọn, khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất nhỏ hơn số lượng hợp nhất tối đa thậm chí mặc dù các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng thứ nhất được thêm vào danh mục ứng viên hợp nhất, các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng thứ hai có thể được thêm vào danh mục ứng viên hợp nhất.

Fig.23 là sơ đồ thể hiện ví dụ trong đó ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động dài hạn được thêm vào danh mục ứng viên hợp nhất.

Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất nhỏ hơn số lượng tối đa, các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng thứ nhất HmvpCandList có thể được thêm vào danh mục ứng viên hợp nhất. Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất nhỏ hơn số lượng tối đa thậm chí mặc dù các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng thứ nhất được thêm vào danh mục ứng viên hợp nhất, các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động dài hạn HmvpLTCandList có thể được thêm vào danh mục ứng viên hợp nhất.

Bảng 2 thể hiện quy trình thêm các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động dài hạn vào danh mục ứng viên hợp nhất.

[Bảng 2]

Đối với mỗi ứng viên trong HMVPCandList với chỉ số HMVPLTIdx = 1..numHMVPLTCand, các bước được sắp thứ tự sau đây được lặp lại cho đến khi combStop bằng true

- sameMovement được đặt thành FALSE
- Nếu hmvpStop bằng FALSE và numCurrMergec nhỏ hơn (MaxNumMergeCand-1), hmvpLT được đặt thành TRUE
- Nếu HMVPLTCandList[NumLTHmvp-HMVPLTIdx] có các vector chuyển động giống nhau và các chỉ số tham chiếu giống nhau với bất kỳ mergeCandList[i] nào với I là 0.. numOrigMergeCand-1 HasBeenPruned[i] bằng false, sameMovement được đặt thành true
- Nếu sameMovement bằng false, mergeCandList[numCurrMergeCand++] được đặt thành HMVPLTCandList[NumLTHmvp-HMVPLTIdx]
- Nếu numCurrMergeCand bằng (MaxNumMergeCand-1), hmvpLTStop được đặt thành TRUE

Có thể thực hiện việc đặt sao cho ứng viên hợp nhất liên vùng gồm thêm thông tin cũng như thông tin chuyển động. Theo một ví dụ, kích cỡ, hình dạng, hoặc thông tin phân chia của khối có thể được lưu trữ thêm cho ứng viên hợp nhất liên vùng. Khi tạo cấu hình danh mục ứng viên hợp nhất của khối hiện tại, chỉ các ứng viên hợp nhất dự báo liên ảnh có cùng hoặc kích cỡ, hình dạng tương tự, hoặc thông tin phân chia như khối hiện tại trong số các ứng viên hợp nhất dự báo liên ảnh có thể được sử dụng, hoặc các ứng viên hợp nhất dự báo liên ảnh có cùng hoặc kích cỡ, hình dạng tương tự, hoặc thông tin phân chia như khối hiện tại có thể được ưu tiên thêm vào danh mục ứng viên hợp nhất.

Theo lựa chọn, bảng thông tin chuyển động liên vùng có thể được tạo ra cho mỗi kích cỡ, hình dạng, hoặc thông tin phân chia của khối. Danh mục ứng viên hợp nhất của khối hiện tại có thể được tạo ra sử dụng bảng thông tin chuyển động liên vùng phù hợp với hình dạng, kích cỡ, hoặc thông tin phân chia của khối hiện tại trong số nhiều bảng thông tin chuyển động liên vùng.

Theo lựa chọn, bảng thông tin chuyển động liên vùng có thể được tạo ra cho mỗi độ phân giải của vector chuyển động. Theo một ví dụ, khi vector chuyển động của khối hiện tại có độ phân giải của 1/4 điểm ảnh, ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại có thể được thêm vào danh mục thông tin chuyển động 1/4 điểm ảnh liên vùng. Khi vector chuyển động của khối hiện tại có độ phân giải của 1 điểm ảnh số nguyên, ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại có thể được thêm vào danh mục thông tin chuyển động điểm ảnh số nguyên liên vùng. Khi vector chuyển động của khối hiện tại có độ phân giải của 4 điểm ảnh số nguyên, ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại có thể được thêm vào danh mục thông tin chuyển động 4 điểm ảnh số nguyên liên vùng. Một trong số nhiều danh mục thông tin chuyển động liên vùng có thể được chọn theo độ phân giải vector chuyển động của khối cần được mã hóa hoặc được giải mã.

Khi phương pháp mã hóa vector độ lệch hợp nhất được ứng dụng cho khối hiện tại, ứng viên hợp nhất liên vùng được suy ra từ khối hiện tại có thể được thêm vào danh mục thông tin chuyển động độ lệch hợp nhất liên vùng `HmvpHMVDCandList` thay vì được thêm vào bảng thông tin chuyển động liên vùng `HmvpCandList`. Trong trường hợp này, ứng viên hợp nhất liên vùng có thể gồm thông tin độ lệch vector chuyển động của khối hiện tại. `HmvpHMVDCandList` có thể được sử dụng để suy ra độ lệch của khối được ứng dụng phương pháp mã hóa vector độ lệch hợp nhất.

Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp

nhất của khối hiện tại là không bằng trị số lớn nhất, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng có thể được thêm vào danh mục ứng viên hợp nhất. Quy trình thêm có thể theo thứ tự tăng dần hoặc giảm dần dựa trên các chỉ số. Theo một ví dụ, ứng viên hợp nhất liên vùng với chỉ số lớn nhất có thể được ưu tiên thêm vào danh mục ứng viên hợp nhất.

Khi ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng cần được thêm vào danh mục ứng viên hợp nhất, kiểm tra dư thừa giữa ứng viên hợp nhất liên vùng và các ứng viên hợp nhất được lưu trữ trước trong danh mục ứng viên hợp nhất có thể được thực hiện.

Ví dụ như, bảng 3 thể hiện quy trình thêm ứng viên hợp nhất liên vùng vào danh mục ứng viên hợp nhất.

[Bảng 3]

Đối với mỗi ứng viên trong HMVPCandList với chỉ số HMVPIdx = 1..
numCheckedHMVPCand, các bước được sắp thứ tự sau đây được lặp lại cho
đến khi combStop bằng true

- sameMovement được đặt thành false
- Nếu HMVPCandList[NumHmvp-HMVPIdx] có các vector chuyển động
giống nhau và các chỉ số tham chiếu giống nhau với bất kỳ mergeCandList[i]
nào với I là 0..
numOrigMergeCand-1 và HasBeenPruned[i] bằng false,
sameMovement được đặt thành true
- Nếu sameMovement bằng false, mergeCandList[numCurrMergeCand++]
được đặt thành HMVPCandList[NumHmvp-HMVPIdx]
- Nếu numCurrMergeCand bằng (MaxNumMergeCand-1), hmvpStop được đặt
thành TRUE

Kiểm tra dư thừa có thể được thực hiện trên chỉ một số trong số các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng. Theo một ví dụ, việc kiểm tra dư thừa có thể được thực hiện trên chỉ các ứng viên hợp nhất liên vùng với các chỉ số lớn hơn hoặc bằng trị số ngưỡng hoặc các chỉ số nhỏ hơn hoặc bằng trị số ngưỡng.

Theo lựa chọn, việc kiểm tra dư thừa có thể được thực hiện trên chỉ một số trong số các ứng viên hợp nhất được lưu trữ trước trong danh mục ứng viên hợp nhất. Theo một ví dụ, việc kiểm tra dư thừa có thể được thực hiện chỉ trên ứng viên hợp nhất với chỉ số lớn hơn hoặc bằng trị số ngưỡng hoặc chỉ số nhỏ hơn hoặc bằng trị số ngưỡng hoặc

ứng viên hợp nhất được suy ra từ khối ở vị trí cụ thể. Ở đây, vị trí cụ thể có thể gồm ít nhất một trong số khối lân cận trái, khối lân cận trên, khối lân cận trên-phải, hoặc khối lân cận dưới-trái của khối hiện tại.

Fig.24 là sơ đồ thể hiện ví dụ trong đó việc kiểm tra dư thừa được thực hiện trên chỉ một số ứng viên hợp nhất.

Khi ứng viên hợp nhất liên vùng $HmvpCand[j]$ cần được thêm vào danh mục ứng viên hợp nhất, việc kiểm tra dư thừa có thể được thực hiện trên ứng viên hợp nhất liên vùng liên quan đến hai ứng viên hợp nhất $mergeCandList[NumMerge-2]$ và $mergeCandList[NumMerge-1]$, các ứng viên này có các chỉ số lớn nhất. Ở đây, $NumMerge$ có thể thể hiện số lượng các ứng viên hợp nhất không gian và các ứng viên hợp nhất thời gian sẵn có.

Khi ứng viên hợp nhất giống với ứng viên hợp nhất liên vùng thứ nhất được tìm thấy, kiểm tra dư thừa trên ứng viên hợp nhất giống với ứng viên hợp nhất liên vùng thứ nhất có thể bị bỏ qua trong khi kiểm tra dư thừa được thực hiện trên ứng viên hợp nhất liên vùng thứ hai.

Fig.25 là sơ đồ thể hiện ví dụ trong đó việc kiểm tra dư thừa trên ứng viên hợp nhất cụ thể bị bỏ qua.

Khi ứng viên hợp nhất liên vùng $HmvpCand[i]$ với chỉ số của i cần được thêm vào danh mục ứng viên hợp nhất, kiểm tra dư thừa giữa ứng viên hợp nhất liên vùng và các ứng viên hợp nhất được lưu trữ trước trong danh mục ứng viên hợp nhất có thể được thực hiện. Trong trường hợp này, khi ứng viên hợp nhất $mergeCandList[j]$ giống với ứng viên hợp nhất liên vùng $HmvpCand[i]$ được tìm thấy, ứng viên hợp nhất liên vùng $HmvpCand[i]$ có thể không được thêm vào danh mục ứng viên hợp nhất, và kiểm tra dư thừa giữa ứng viên hợp nhất liên vùng $HmvpCand[i-1]$, ứng viên này chỉ số của $i-1$, và các ứng viên hợp nhất có thể được thực hiện. Trong trường hợp này, kiểm tra dư thừa giữa ứng viên hợp nhất liên vùng $HmvpCand[i-1]$ và ứng viên hợp nhất $mergeCandList[j]$ có thể bị bỏ qua.

Ví dụ như, trong ví dụ được thể hiện trên Fig.25, nội dung được xác định là $HmvpCand[i]$ và $mergeCandList[2]$ là giống nhau. Do đó, $HmvpCand[i]$ có thể không được thêm vào danh mục ứng viên hợp nhất, và kiểm tra dư thừa có thể được thực hiện trên $HmvpCand[i-1]$. Trong trường hợp này, kiểm tra dư thừa giữa $HmvpCand[i-1]$ và $mergeCandList[2]$ có thể bị bỏ qua.

Khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp

nhất của khối hiện tại là nhỏ hơn số lượng tối đa, ít nhất một trong số ứng viên hợp nhất từng đôi hoặc ứng viên hợp nhất số không còn có thể được đưa vào ngoài ứng viên hợp nhất liên vùng ra. Ứng viên hợp nhất từng đôi đề cập đến ứng viên hợp nhất với vector chuyển động bằng trị số trung bình của các vector chuyển động của hai hoặc nhiều hơn hai ứng viên hợp nhất, và ứng viên hợp nhất số không đề cập đến ứng viên hợp nhất với vector chuyển động của 0.

Ứng viên hợp nhất có thể được thêm vào danh mục ứng viên hợp nhất của khối hiện tại theo thứ tự sau đây.

Ứng viên hợp nhất không gian – Ứng viên hợp nhất thời gian – Ứng viên hợp nhất liên vùng – (Ứng viên hợp nhất affin liên vùng) – Ứng viên hợp nhất từng đôi – Ứng viên hợp nhất số không.

Ứng viên hợp nhất không gian đề cập đến ứng viên hợp nhất được suy ra từ ít nhất một trong số khối lân cận hoặc khối không lân cận, và ứng viên hợp nhất thời gian đề cập đến ứng viên hợp nhất được suy ra từ hình ảnh tham chiếu đứng trước. Ứng viên hợp nhất affin liên vùng đề cập đến ứng viên hợp nhất liên vùng được suy ra từ khối được mã hóa hoặc được giải mã bởi mô hình chuyển động affin.

Bảng thông tin chuyển động liên vùng có thể được sử dụng ngay cả trong vector chuyển động phương thức dự báo. Theo một ví dụ, khi số lượng ứng viên dự báo các vector chuyển động được đưa vào danh mục ứng viên dự báo vector chuyển động của khối hiện tại là nhỏ hơn số lượng tối đa, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng có thể được đặt là ứng viên dự báo vector chuyển động cho khối hiện tại. Cụ thể, vector chuyển động của ứng viên hợp nhất liên vùng có thể được đặt là ứng viên dự báo vector chuyển động.

Khi một trong số ứng viên dự báo các vector chuyển động được đưa vào danh mục ứng viên dự báo vector chuyển động của khối hiện tại được chọn, ứng viên được chọn có thể được đặt là bộ dự báo vector chuyển động của khối hiện tại. Sau đó, vector chuyển động của khối hiện tại có thể thu được bằng cách giải mã trị số còn lại vector chuyển động của khối hiện tại và sau đó thêm bộ dự báo vector chuyển động và vector chuyển động trị số còn lại.

Danh mục ứng viên dự báo vector chuyển động của khối hiện tại có thể được tạo cấu hình theo thứ tự sau đây.

Ứng viên dự báo vector chuyển động không gian – Ứng viên dự báo vector chuyển động thời gian – Ứng viên hợp nhất khu vực giải mã liên ảnh – (Ứng viên hợp nhất affin

khu vực giải mã liên ảnh) – Ứng viên dự báo vector chuyển động số không

Ứng viên dự báo vector chuyển động không gian đề cập đến ứng viên dự báo vector chuyển động được suy ra từ ít nhất một trong số khối lân cận hoặc khối không lân cận, và ứng viên dự báo vector chuyển động thời gian đề cập đến ứng viên dự báo vector chuyển động được suy ra từ hình ảnh tham chiếu đứng trước. Ứng viên hợp nhất affin liên vùng đề cập đến ứng viên dự báo vector chuyển động liên vùng được suy ra từ khối được mã hóa hoặc được giải mã bởi mô hình chuyển động affin. Ứng viên dự báo vector chuyển động số không thể hiện ứng viên có vector chuyển động của 0.

Khu vực xử lý hợp nhất lớn hơn khối mã hóa có thể được xác định. Các khối mã hóa được đưa vào khu vực xử lý hợp nhất có thể được xử lý song song mà không được mã hóa hoặc được giải mã theo tuần tự. Ở đây, mã hóa hoặc giải mã không theo tuần tự nghĩa là thứ tự mã hóa hoặc giải mã không được xác định. Do đó, quy trình mã hóa hoặc giải mã khối được đưa vào khu vực xử lý hợp nhất có thể được xử lý độc lập. Theo lựa chọn, khối được đưa vào khu vực xử lý hợp nhất có thể chia sẻ các ứng viên hợp nhất. Ở đây, các ứng viên hợp nhất có thể được suy ra dựa trên khu vực xử lý hợp nhất.

Theo các đặc điểm được mô tả trên đây, khu vực xử lý hợp nhất có thể được gọi là khu vực xử lý song song, vùng hợp nhất dùng chung (Shared Merge Region - SMR), hoặc vùng ước tính hợp nhất (Merge Estimation Region - MER).

Ứng viên hợp nhất của khối hiện tại có thể được suy ra dựa trên khối mã hóa. Tuy nhiên, khi khối hiện tại được đưa vào khu vực hợp nhất song song lớn hơn khối hiện tại, khối ứng viên được đưa vào khu vực hợp nhất song song giống như khu vực của khối hiện tại có thể được đặt là không sẵn có là ứng viên hợp nhất.

Fig.26 là sơ đồ thể hiện ví dụ trong đó khối ứng viên được đưa vào khu vực hợp nhất song song giống như khu vực của khối hiện tại được đặt là không sẵn có là ứng viên hợp nhất.

Trong ví dụ được thể hiện trên Fig.26A, khi CU5 được mã hóa hoặc được giải mã, các khối gồm các mẫu tham chiếu gần kề CU5 có thể được đặt là các khối ứng viên. Trong trường hợp này, các khối ứng viên X3 và X4, được đưa vào khu vực hợp nhất song song giống như khu vực của CU5, có thể được đặt là không sẵn có là các ứng viên hợp nhất. Mặt khác, các khối ứng viên X0, X1, và X2, các khối này không được đưa vào khu vực hợp nhất song song giống như khu vực của CU5, có thể được đặt sẵn có là các ứng viên hợp nhất.

Trong ví dụ được thể hiện trên Fig.26B, khi CU8 được mã hóa hoặc được giải mã,

các khối gồm các mẫu tham chiếu gần kề CU8 có thể được đặt là các khối ứng viên. Trong trường hợp này, các khối ứng viên X6, X7, và X8, được đưa vào khu vực hợp nhất song song giống như khu vực của CU8, có thể được đặt là không sẵn có là các ứng viên hợp nhất. Mặt khác, các khối ứng viên X5 và X9, các khối này không được đưa vào cùng một vùng hợp nhất giống như của CU8, có thể được đặt sẵn có là các ứng viên hợp nhất.

Khu vực hợp nhất song song có thể là hình vuông hoặc không phải hình vuông. Thông tin để xác định khu vực hợp nhất song song có thể được báo hiệu trong dòng bit. Thông tin có thể gồm ít nhất một trong số thông tin biểu thị hình dạng của khu vực hợp nhất song song hoặc thông tin biểu thị kích cỡ của khu vực hợp nhất song song. Khi khu vực hợp nhất song song không phải là hình vuông, ít nhất một trong số thông tin biểu thị kích cỡ của khu vực hợp nhất song song, thông tin biểu thị độ rộng và/hoặc độ cao của khu vực hợp nhất song song, hoặc thông tin biểu thị tỷ lệ giữa độ rộng và độ cao của khu vực hợp nhất song song có thể được báo hiệu trong dòng bit.

Kích cỡ của khu vực hợp nhất song song có thể được xác định dựa trên ít nhất một trong số thông tin được báo hiệu trong dòng bit, độ phân giải hình ảnh, kích cỡ phiên, hoặc kích cỡ ngói.

Khi dự báo bù chuyển động được thực hiện trên khối được đưa vào khu vực hợp nhất song song, ứng viên hợp nhất liên vùng được suy ra dựa trên thông tin chuyển động của khối phải chịu dự báo bù chuyển động có thể được thêm vào bảng thông tin chuyển động liên vùng.

Tuy nhiên, khi ứng viên hợp nhất liên vùng được suy ra từ khối được đưa vào khu vực hợp nhất song song được thêm vào bảng thông tin chuyển động liên vùng, ứng viên hợp nhất liên vùng được suy ra từ khối có thể được sử dụng khi một khối khác được đưa vào khu vực hợp nhất song song thực tế được mã hóa hoặc được giải mã muộn hơn khối. Tức là, mặc dù tính phụ thuộc giữa các khối phải bị loại trừ khi các khối mã hóa hoặc giải mã được đưa vào khu vực hợp nhất song song, bù dự báo chuyển động có thể được thực hiện sử dụng thông tin chuyển động của khối kia được đưa vào khu vực hợp nhất song song. Để giải quyết vấn đề như vậy, thậm chí mặc dù mã hóa hoặc giải mã của khối được đưa vào khu vực hợp nhất song song được hoàn thành, thông tin chuyển động của khối trong đó mã hóa hoặc giải mã được hoàn thành có thể không được thêm vào bảng thông tin chuyển động liên vùng.

Theo lựa chọn, khi dự báo bù chuyển động được thực hiện trên các khối được đưa vào khu vực hợp nhất song song, ứng viên hợp nhất liên vùng được suy ra từ các khối có

thể được thêm vào bảng thông tin chuyển động liên vùng theo thứ tự được xác định trước. Ở đây, thứ tự được xác định trước có thể được xác định theo thứ tự quét của các khối mã hóa trong đơn vị cây mã hóa hoặc khu vực hợp nhất song song. Thứ tự quét có thể là ít nhất một trong số quét mảnh, quét ngang, quét dọc, hoặc quét hình chữ chi. Theo lựa chọn, thứ tự được xác định trước có thể được xác định dựa trên thông tin chuyển động của mỗi trong số các khối hoặc số lượng của các khối có cùng thông tin chuyển động.

Theo lựa chọn, ứng viên hợp nhất liên vùng gồm thông tin chuyển động một chiều có thể được thêm vào danh sách hợp nhất liên vùng sớm hơn ứng viên hợp nhất liên vùng gồm thông tin chuyển động hai chiều. Ngược lại, ứng viên hợp nhất liên vùng gồm thông tin chuyển động hai chiều có thể được thêm vào danh sách ứng viên hợp nhất liên vùng sớm hơn ứng viên hợp nhất liên vùng gồm thông tin chuyển động một chiều.

Theo lựa chọn, các ứng viên hợp nhất liên vùng có thể được thêm vào bảng thông tin chuyển động liên vùng theo thứ tự tăng dần hoặc giảm dần của tần suất sử dụng trong khu vực hợp nhất song song hoặc đơn vị cây mã hóa.

Khi khối hiện tại được đưa vào khu vực hợp nhất song song và số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất của khối hiện tại là nhỏ hơn số lượng tối đa, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng có thể được thêm vào danh mục ứng viên hợp nhất. Trong trường hợp này, có thể thực hiện việc đặt sao cho ứng viên hợp nhất liên vùng được suy ra từ khối được đưa vào khu vực hợp nhất song song giống như khu vực của khối hiện tại được thêm vào danh mục ứng viên hợp nhất của khối hiện tại.

Theo lựa chọn, có thể thực hiện việc đặt sao cho khi khối hiện tại được đưa vào khu vực hợp nhất song song, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng không được sử dụng. Tức là, ngay cả khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất của khối hiện tại là nhỏ hơn số lượng tối đa, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng có thể không được thêm vào danh mục ứng viên hợp nhất.

Bảng thông tin chuyển động liên vùng có thể được tạo cấu hình cho khu vực hợp nhất song song hoặc đơn vị cây mã hóa. Bảng thông tin chuyển động liên vùng này có tác dụng lưu trữ tạm thời thông tin chuyển động của các khối được đưa vào khu vực hợp nhất song song. Để phân biệt bảng thông tin chuyển động liên vùng chung từ bảng thông tin chuyển động liên vùng cho khu vực hợp nhất song song hoặc đơn vị cây mã hóa, bảng thông tin chuyển động liên vùng cho khu vực hợp nhất song song hoặc đơn vị cây

mã hóa sẽ được gọi là bảng thông tin chuyển động tạm thời. Ngoài ra, ứng viên hợp nhất liên vùng được lưu trữ trong bảng thông tin chuyển động tạm thời sẽ được gọi là ứng viên hợp nhất tạm thời.

Fig.27 là sơ đồ thể hiện bảng thông tin chuyển động tạm thời.

Bảng thông tin chuyển động tạm thời có thể được tạo cấu hình cho đơn vị cây mã hóa hoặc khu vực hợp nhất song song. Khi dự báo bù chuyển động được thực hiện trên khối hiện tại được đưa vào đơn vị cây mã hóa hoặc khu vực hợp nhất song song, thông tin chuyển động của khối có thể không được thêm vào bảng thông tin chuyển động dự báo liên ảnh HmvpCandList. Thay vào đó, ứng viên hợp nhất tạm thời được suy ra từ khối có thể được thêm vào bảng thông tin chuyển động tạm thời HmvpMERCandList. Tức là, ứng viên hợp nhất tạm thời được thêm vào bảng thông tin chuyển động tạm thời có thể không được thêm vào bảng thông tin chuyển động liên vùng. Do đó, bảng thông tin chuyển động liên vùng có thể không gồm ứng viên hợp nhất liên vùng được suy ra dựa trên thông tin chuyển động của các khối được đưa vào khu vực hợp nhất song song hoặc đơn vị cây mã hóa gồm khối hiện tại.

Số lượng tối đa của các ứng viên hợp nhất có thể được đưa vào bảng thông tin chuyển động tạm thời có thể được đặt là giống như vị trí của bảng thông tin chuyển động liên vùng. Theo lựa chọn, số lượng tối đa của các ứng viên hợp nhất có thể được đưa vào bảng thông tin chuyển động tạm thời có thể được xác định theo kích cỡ của đơn vị cây mã hóa hoặc khu vực hợp nhất song song.

Khối hiện tại được đưa vào đơn vị cây mã hóa hoặc khu vực hợp nhất song song có thể được đặt không sử dụng bảng thông tin chuyển động tạm thời cho đơn vị cây mã hóa tương ứng hoặc khu vực hợp nhất song song tương ứng. Tức là, khi số lượng của các ứng viên hợp nhất được đưa vào danh mục ứng viên hợp nhất của khối hiện tại là nhỏ hơn trị số lớn nhất, ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng có thể được thêm vào danh mục ứng viên hợp nhất, và ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời có thể không được thêm vào danh mục ứng viên hợp nhất. Do đó, thông tin chuyển động của các khối khác được đưa vào đơn vị cây mã hóa hoặc khu vực hợp nhất song song giống như đơn vị hoặc khu vực của khối hiện tại có thể không được sử dụng cho dự báo bù chuyển động của khối hiện tại.

Khi việc mã hóa hoặc giải mã tất cả các khối được đưa vào đơn vị cây mã hóa hoặc khu vực hợp nhất song song được hoàn thành, bảng thông tin chuyển động liên

vùng và bảng thông tin chuyển động tạm thời có thể được hợp nhất.

Fig.28 là sơ đồ thể hiện ví dụ về việc hợp nhất bảng thông tin chuyển động tạm thời với bảng thông tin chuyển động liên vùng.

Khi việc mã hóa hoặc giải mã tất cả các khối được đưa vào đơn vị cây mã hóa hoặc khu vực hợp nhất song song được hoàn thành, bảng thông tin chuyển động liên vùng có thể được cập nhật với ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời trong ví dụ được thể hiện trên Fig.28.

Trong trường hợp này, ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời có thể được thêm vào bảng thông tin chuyển động liên vùng theo thứ tự trong đó các ứng viên hợp nhất tạm thời đã được đưa vào bảng thông tin chuyển động tạm thời (tức là, theo thứ tự tăng dần hoặc giảm dần của các trị số chỉ số).

Theo một ví dụ khác, ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời có thể được thêm vào bảng thông tin chuyển động liên vùng theo thứ tự được xác định trước.

Ở đây, thứ tự được xác định trước có thể được xác định theo thứ tự quét của các khối mã hóa trong đơn vị cây mã hóa hoặc khu vực hợp nhất song song. Thứ tự quét có thể là ít nhất một trong số quét mảnh, quét ngang, quét dọc, hoặc quét hình chữ chi. Theo lựa chọn, thứ tự được xác định trước có thể được xác định dựa trên thông tin chuyển động của mỗi trong số các khối hoặc số lượng của các khối có cùng thông tin chuyển động.

Theo lựa chọn, ứng viên hợp nhất tạm thời gồm thông tin chuyển động một chiều có thể được thêm vào danh sách hợp nhất liên vùng sớm hơn ứng viên hợp nhất tạm thời gồm thông tin chuyển động hai chiều. Ngược lại, ứng viên hợp nhất tạm thời gồm thông tin chuyển động hai chiều có thể được thêm vào danh sách ứng viên hợp nhất liên vùng sớm hơn ứng viên hợp nhất tạm thời gồm thông tin chuyển động một chiều.

Theo lựa chọn, các ứng viên hợp nhất tạm thời có thể được thêm vào bảng thông tin chuyển động liên vùng theo thứ tự tăng dần hoặc giảm dần của tần suất sử dụng trong khu vực hợp nhất song song hoặc đơn vị cây mã hóa.

Khi ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời được thêm vào bảng thông tin chuyển động liên vùng, kiểm tra dư thừa cho ứng viên hợp nhất tạm thời có thể được thực hiện. Theo một ví dụ, khi ứng viên hợp nhất liên vùng giống với ứng viên hợp nhất tạm thời được đưa vào bảng thông tin chuyển động tạm thời được lưu trữ trước trong bảng thông tin chuyển động liên vùng, ứng viên hợp

nhất tạm thời có thể không được thêm vào bảng thông tin chuyển động liên vùng. Trong trường hợp này, việc kiểm tra dư thừa có thể được thực hiện trên chỉ một số trong số các ứng viên hợp nhất liên vùng được đưa vào bảng thông tin chuyển động liên vùng. Theo một ví dụ, việc kiểm tra dư thừa có thể được thực hiện chỉ trên các ứng viên hợp nhất dự báo liên ảnh với các chỉ số lớn hơn hoặc bằng trị số ngưỡng hoặc các chỉ số nhỏ hơn hoặc bằng trị số ngưỡng. Theo một ví dụ, khi ứng viên hợp nhất tạm thời giống với ứng viên hợp nhất liên vùng với chỉ số lớn hơn hoặc bằng trị số được xác định trước, ứng viên hợp nhất tạm thời có thể không được thêm vào bảng thông tin chuyển động liên vùng.

Dự báo nội ảnh là để dự báo khối hiện tại sử dụng mẫu được tái cấu trúc mà ở đó việc mã hóa hoặc giải mã được hoàn thành và mẫu này nằm gần khối hiện tại. Trong trường hợp này, mẫu được tái cấu trúc trước khi bộ lọc trong vòng lặp được ứng dụng có thể được sử dụng để dự báo nội ảnh của khối hiện tại.

Kỹ thuật dự báo nội ảnh gồm dự báo nội ảnh dựa trên ma trận và dự báo nội ảnh bình thường xét đến hướng so với mẫu được tái cấu trúc gần đó. Thông tin biểu thị kỹ thuật dự báo nội ảnh của khối hiện tại có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit. Theo lựa chọn, kỹ thuật dự báo nội ảnh của khối hiện tại có thể được xác định dựa trên ít nhất một trong số vị trí, kích cỡ, hoặc hình dạng của khối hiện tại hoặc kỹ thuật dự báo nội ảnh của khối lân cận. Theo một ví dụ, khi khối hiện tại có mặt qua ranh giới hình ảnh, có thể thực hiện việc đặt sao cho dự báo nội ảnh dựa trên ma trận không được ứng dụng cho khối hiện tại.

Dự báo nội ảnh dựa trên ma trận có thể là phương pháp thu được khối dự báo của khối hiện tại dựa trên sản phẩm ma trận giữa ma trận được lưu trữ trước trong bộ mã hóa và bộ giải mã và các mẫu được tái cấu trúc gần khối hiện tại. Thông tin để định rõ một trong số nhiều được lưu trữ trước các ma trận có thể được báo hiệu trong dòng bit. Bộ giải mã có thể xác định ma trận cho dự báo nội ảnh của khối hiện tại dựa trên thông tin và kích cỡ của khối hiện tại.

Dự báo nội ảnh bình thường là phương pháp thu được khối dự báo cho khối hiện tại dựa trên phương thức dự báo nội ảnh không định hướng hoặc phương thức dự báo định hướng nội khung. Quy trình thực hiện dự báo nội ảnh dựa trên dự báo nội ảnh bình thường sẽ được mô tả chi tiết dưới đây kết hợp với các hình vẽ.

Fig.29 là lưu đồ của phương pháp dự báo nội ảnh theo một phương án của sáng chế này.

Đường mẫu tham chiếu của khối hiện tại có thể được xác định (S2901). Đường mẫu tham chiếu đề cập đến tập hợp các mẫu tham chiếu được đưa vào đường thứ k nằm cách xa so với phía đỉnh và/hoặc phía trái của khối hiện tại. Các mẫu tham chiếu có thể được suy ra từ các mẫu được tái cấu trúc mà ở đó việc mã hóa hoặc giải mã được hoàn thành và các mẫu này gần khối hiện tại.

Thông tin chỉ số để nhận dạng đường mẫu tham chiếu của khối hiện tại trong số nhiều đường mẫu tham chiếu có thể được báo hiệu trong dòng bit. Nhiều đường mẫu tham chiếu có thể gồm ít nhất một trong số đường thứ nhất, đường thứ hai, đường thứ ba, hoặc đường thứ tư từ phía đỉnh và/hoặc phía trái của khối hiện tại. Bảng 4 thể hiện các chỉ số được gán cho các đường mẫu tham chiếu. Bảng 4 giả định rằng đường thứ nhất, đường thứ hai, và đường thứ tư được sử dụng là các ứng viên đường mẫu tham chiếu.

[Bảng 4]

Chỉ số	Đường mẫu tham chiếu
0	Đường mẫu tham chiếu thứ nhất
1	Đường mẫu tham chiếu thứ hai
2	Đường mẫu tham chiếu thứ tư

Đường mẫu tham chiếu của khối hiện tại có thể được xác định dựa trên ít nhất một trong số vị trí, kích cỡ, hoặc hình dạng của khối hiện tại hoặc phương thức dự báo của khối lân cận. Theo một ví dụ, khi khối hiện tại là tiếp giáp với mép của hình ảnh, ngôi, phiên, hoặc đơn vị cây mã hóa, đường mẫu tham chiếu thứ nhất có thể được xác định là đường mẫu tham chiếu của khối hiện tại.

Đường mẫu tham chiếu có thể gồm các mẫu tham chiếu trên được bố trí phía trên khối hiện tại và mẫu tham chiếu trái được bố trí về phía trái của khối hiện tại. Các mẫu tham chiếu trên và các mẫu tham chiếu trái có thể được suy ra từ các mẫu được tái cấu trúc gần khối hiện tại. Các mẫu được tái cấu trúc có thể ở trạng thái trước khi bộ lọc trong vòng lặp được ứng dụng.

Fig.30 là sơ đồ thể hiện các mẫu tham chiếu được đưa vào mỗi đường mẫu tham chiếu.

Mẫu dự báo có thể thu được sử dụng ít nhất một mẫu tham chiếu thuộc về đường mẫu tham chiếu theo phương thức dự báo nội ảnh của khối hiện tại.

Tiếp theo, phương thức dự báo nội ảnh của khối hiện tại có thể được xác định (S2902). Ít nhất một trong số phương thức dự báo nội ảnh không định hướng hoặc

phương thức dự báo định hướng nội khung có thể được xác định là phương thức dự báo nội ảnh của khối hiện tại. Phương thức dự báo nội ảnh không định hướng gồm phương thức hai chiều và phương thức DC, và dự báo nội ảnh định hướng gồm 33 hoặc 65 phương thức từ hướng dưới-trái xiên đến hướng trên-phải xiên.

Fig.31 là sơ đồ thể hiện các phương thức dự báo nội ảnh.

Fig.31A thể hiện 35 các phương thức dự báo nội ảnh, và Fig.31B thể hiện 67 các phương thức dự báo nội ảnh.

Số lượng lớn hơn của các phương thức dự báo nội ảnh hoặc số lượng nhỏ hơn của các phương thức dự báo nội ảnh có thể được xác định than được thể hiện trên Fig.31.

Phương thức đúng nhất (Most Probable Mode - MPM) có thể được đặt dựa trên phương thức dự báo nội ảnh của khối lân cận khối hiện tại. Ở đây, khối lân cận có thể gồm khối lân cận trái gần kề phía trái của khối hiện tại và khối lân cận trên gần kề đỉnh của khối hiện tại. Khi hệ tọa độ của mẫu trên-trái của khối hiện tại là $(0, 0)$, khối lân cận trái có thể gồm mẫu được bố trí tại $(-1, 0)$, $(-1, H-1)$, hoặc $(-1, (H-1)/2)$. Ở đây, H thể hiện độ cao của khối hiện tại. Khối lân cận trên có thể gồm mẫu được bố trí tại $(0, -1)$, $(W-1, -1)$, hoặc $((W-1)/2, -1)$. Ở đây, W thể hiện độ rộng của khối hiện tại.

Khi khối lân cận được mã hóa thông qua dự báo nội ảnh bình thường, MPM có thể được suy ra dựa trên phương thức dự báo nội ảnh của khối lân cận. Cụ thể, phương thức dự báo nội ảnh của khối lân cận trái có thể được đặt bằng cách sử dụng `candIntraPredModeA` biến thiên, và phương thức dự báo nội ảnh của khối lân cận trên có thể được đặt bằng cách sử dụng `candIntraPredModeB` biến thiên.

Trong trường hợp này, khi khối lân cận không sẵn có (ví dụ như, khi khối lân cận chưa được mã hóa hoặc được giải mã hoặc khi vị trí của khối lân cận chệch từ mép hình ảnh), khi khối lân cận được mã hóa thông qua dự báo nội ảnh dựa trên ma trận, khi khối lân cận được mã hóa thông qua dự báo liên ảnh, hoặc khi khối lân cận được đưa vào đơn vị cây mã hóa khác với đơn vị cây mã hóa của khối hiện tại, `candIntraPredModeX` biến thiên (ở đây, X là A hoặc B), được suy ra dựa trên phương thức dự báo nội ảnh của khối lân cận có thể được đặt là phương thức mặc định. Ở đây, phương thức mặc định có thể gồm ít nhất một trong số phương thức hai chiều, phương thức DC, phương thức dọc hoặc Phương thức ngang.

Theo lựa chọn, khi khối lân cận được mã hóa thông qua dự báo nội ảnh dựa trên ma trận, phương thức dự báo nội ảnh tương ứng với trị số chỉ số để định rõ một trong số nhiều ma trận có thể được đặt là `candIntraPredModeX`. Để thực hiện được điều này, bảng

tra cứu biểu thị mối quan hệ ánh xạ giữa các phương thức dự báo nội ảnh và các trị số chỉ số để định rõ các ma trận có thể được lưu trữ trước trong bộ mã hóa và bộ giải mã.

Các MPM có thể được suy ra dựa trên `candIntraPredModeA` biến thiên và `candIntraPredModeB` biến thiên. Số lượng của các MPM được đưa vào danh mục MPM có thể được đặt sẵn trong bộ mã hóa và bộ giải mã. Theo một ví dụ, số lượng của các MPM có thể là ba, bốn, năm, hoặc sáu. Theo lựa chọn, thông tin biểu thị số lượng của các MPM có thể được báo hiệu trong dòng bit. Theo lựa chọn, số lượng của các MPM có thể được xác định dựa trên ít nhất một trong số phương thức dự báo của khối lân cận hoặc kích cỡ hoặc hình dạng của khối hiện tại.

Các phương án sau đây giả định rằng số lượng của các MPM là ba, và ba MPM được gọi là `MPM[0]`, `MPM[1]`, và `MPM[2]`. Khi số lượng của các MPM lớn hơn ba, các MPM có thể được tạo cấu hình để gồm ba MPM mà sẽ được mô tả trong các phương án sau đây.

Khi `candIntraPredA` và `candIntraPredB` là giống nhau và `candIntraPredA` là Hai chiều hoặc phương thức DC, `MPM[0]` và `MPM[1]` có thể được đặt tương ứng cho phương thức Hai chiều và phương thức DC. `MPM[2]` có thể được đặt cho phương thức dự báo nội ảnh dọc, phương thức dự báo nội ảnh ngang, hoặc phương thức dự báo nội ảnh xiên. Phương thức dự báo nội ảnh xiên có thể là phương thức dự báo nội ảnh dưới-trái xiên, phương thức dự báo nội ảnh trên-trái, hoặc phương thức dự báo nội ảnh trên-phải.

Khi `candIntraPredA` và `candIntraPredB` là giống nhau và `candIntraPredA` là phương thức dự báo định hướng nội khung, `MPM[0]` có thể được đặt là giống với `candIntraPredA`. `MPM[1]` và `MPM[2]` có thể được đặt cho các phương thức dự báo nội ảnh tương tự với `candIntraPredA`. Phương thức dự báo nội ảnh tương tự với `candIntraPredA` có thể là phương thức dự báo nội ảnh trong đó trị số chênh lệch chỉ số liên quan đến `candIntraPredA` là ± 1 hoặc ± 2 . Phép toán môđun (%) và độ lệch có thể được sử dụng để suy ra phương thức dự báo nội ảnh tương tự với `candIntraPredA`.

Khi `candIntraPredA` và `candIntraPredB` là khác nhau, `MPM[0]` có thể được đặt là giống với `candIntraPredA`, và `MPM[1]` có thể được đặt là giống với `candIntraPredB`. Trong trường hợp này, khi cả `candIntraPredA` và `candIntraPredB` là các phương thức dự báo nội ảnh không định hướng, `MPM[2]` có thể được đặt là phương thức dự báo nội ảnh dọc, phương thức dự báo nội ảnh ngang, hoặc phương thức dự báo nội ảnh xiên. Theo lựa chọn, khi ít nhất một trong số `candIntraPredA` và `candIntraPredB` là phương thức dự

báo định hướng nội khung, MPM[2] có thể được đặt là phương thức dự báo nội ảnh được suy ra bằng cách cộng hoặc trừ độ lệch vào hoặc từ Hai chiều, DC, hoặc trị số lớn hơn giữa $candIntraPredA$ và $candIntraPredB$. Ở đây, độ lệch có thể là 1 hoặc 2.

Danh mục MPM gồm nhiều MPM được tạo ra, và thông tin biểu thị việc liệu MPM có giống với phương thức dự báo nội ảnh của khối hiện tại được đưa vào danh mục MPM có thể được báo hiệu trong dòng bit. Thông tin là cờ 1 bit, và cờ có thể được gọi là cờ MPM. Khi cờ MPM biểu thị rằng MPM giống với khối hiện tại được đưa vào danh mục MPM, thông tin chỉ số để nhận dạng một trong số các MPM có thể được báo hiệu trong dòng bit. MPM được định rõ bởi thông tin chỉ số có thể được đặt là phương thức dự báo nội ảnh của khối hiện tại. Khi cờ MPM biểu thị rằng MPM giống với khối hiện tại không được đưa vào danh mục MPM, thông tin phương thức còn lại biểu thị một trong số các phương thức dự báo nội ảnh còn lại không phải là các MPM có thể được báo hiệu trong dòng bit. Thông tin phương thức còn lại biểu thị trị số chỉ số tương ứng với phương thức dự báo nội ảnh của khối hiện tại khi các chỉ số được gán lại cho các phương thức dự báo nội ảnh còn lại không phải là các MPM. Bộ giải mã có thể phân loại các MPM theo thứ tự tăng lên và so sánh các MPM với thông tin phương thức còn lại để xác định phương thức dự báo nội ảnh của khối hiện tại. Theo một ví dụ, khi thông tin phương thức còn lại là nhỏ hơn hoặc bằng MPM, phương thức dự báo nội ảnh của khối hiện tại có thể được suy ra bằng cách thêm một vào thông tin phương thức còn lại.

Thay vì đặt MPM là phương thức mặc định, thông tin biểu thị việc liệu phương thức dự báo nội ảnh của khối hiện tại là phương thức mặc định có thể được báo hiệu trong dòng bit. Thông tin có thể là cờ 1 bit, và cờ có thể được gọi là cờ phương thức mặc định. Cờ phương thức mặc định có thể được báo hiệu chỉ khi cờ MPM biểu thị rằng MPM giống với khối hiện tại được đưa vào danh mục MPM. Như được mô tả trên đây, phương thức mặc định có thể gồm ít nhất một trong số phương thức hai chiều, DC, phương thức dọc hoặc phương thức ngang. Theo một ví dụ, khi phương thức hai chiều được đặt là phương thức mặc định, cờ phương thức mặc định có thể biểu thị việc liệu phương thức dự báo nội ảnh của khối hiện tại là phương thức hai chiều. Khi cờ phương thức mặc định biểu thị rằng phương thức dự báo nội ảnh của khối hiện tại là not phương thức mặc định, một trong số các MPM được biểu thị bởi thông tin chỉ số có thể được đặt là phương thức dự báo nội ảnh của khối hiện tại.

Khi nhiều phương thức dự báo nội ảnh được đặt là các phương thức mặc định, chỉ số thông tin biểu thị một trong số các phương thức mặc định còn có thể được báo hiệu.

Phương thức dự báo nội ảnh của khối hiện tại có thể được đặt là phương thức mặc định được biểu thị bởi thông tin chỉ số.

Khi chỉ số của đường mẫu tham chiếu của khối hiện tại không phải là không, có thể thực hiện việc đặt sao cho phương thức mặc định không được sử dụng. Do đó, khi chỉ số của đường mẫu tham chiếu không phải là không, cờ phương thức mặc định có thể không được báo hiệu, và trị số của cờ phương thức mặc định có thể đặt thành trị số được xác định trước (tức là, sai).

Khi phương thức dự báo nội ảnh của khối hiện tại được xác định, các mẫu dự báo cho khối hiện tại có thể thu được dựa trên phương thức dự báo nội ảnh được xác định (S2903).

Khi Phương thức DC được chọn, các mẫu dự báo cho khối hiện tại có thể được tạo ra dựa trên trị số trung bình của các mẫu tham chiếu. Cụ thể, các trị số của tất cả các mẫu trong khối dự báo có thể được tạo ra dựa trên trị số trung bình của các mẫu tham chiếu. Trị số trung bình có thể được suy ra sử dụng ít nhất một trong số các mẫu tham chiếu trên được bố trí phía trên khối hiện tại và mẫu tham chiếu trái được bố trí về phía trái của khối hiện tại.

Số lượng hoặc phạm vi của các mẫu tham chiếu được sử dụng để suy ra trị số trung bình có thể thay đổi phụ thuộc vào hình dạng của khối hiện tại. Theo một ví dụ, khi khối hiện tại là khối không phải hình vuông trong đó độ rộng lớn hơn độ cao, trị số trung bình có thể được tính toán chỉ bằng cách sử dụng các mẫu tham chiếu trên. Mặt khác, khi khối hiện tại là khối không phải hình vuông trong đó độ rộng nhỏ hơn độ cao, trị số trung bình có thể được tính toán chỉ bằng cách sử dụng các mẫu tham chiếu trái. Tức là, khi độ rộng và độ cao của khối hiện tại là khác nhau, trị số trung bình có thể được tính toán chỉ bằng cách sử dụng các mẫu tham chiếu gần kề kích cỡ dài hơn giữa độ rộng và độ cao. Theo lựa chọn, việc liệu có tính toán trị số trung bình bằng cách sử dụng các mẫu tham chiếu trên hoặc để tính toán trị số trung bình bằng cách sử dụng các mẫu tham chiếu trái có thể được xác định dựa trên tỷ lệ độ rộng so với độ cao của khối hiện tại.

Khi phương thức Hai chiều được chọn, mẫu dự báo có thể thu được sử dụng mẫu dự báo ngang và mẫu dự báo dọc. Ở đây, mẫu dự báo ngang thu được dựa trên mẫu tham chiếu trái và mẫu tham chiếu phải được bố trí trên cùng một đường ngang như mẫu dự báo, và mẫu dự báo dọc thu được dựa trên mẫu tham chiếu trên và mẫu tham chiếu dưới được bố trí trên cùng một đường dọc như mẫu dự báo. Ở đây, mẫu tham chiếu phải có thể được tạo ra bằng cách sao chép mẫu tham chiếu gần kề góc trên-phải của khối hiện

tại, và mẫu tham chiếu dưới có thể được tạo ra bằng cách sao chép mẫu tham chiếu gần kề góc dưới-trái của khối hiện tại. Mẫu dự báo ngang có thể thu được dựa trên phép tính tổng có trọng số của mẫu tham chiếu trái và mẫu tham chiếu phải, và mẫu dự báo dọc có thể thu được dựa trên phép tính tổng có trọng số của mẫu tham chiếu trên và mẫu tham chiếu dưới. Trong trường hợp này, trọng số được gán cho mỗi mẫu tham chiếu có thể được xác định phụ thuộc vào vị trí của mẫu dự báo. Mẫu dự báo có thể thu được dựa trên phép toán lấy trung bình hoặc phép tính tổng có trọng số của mẫu dự báo ngang và mẫu dự báo dọc. Khi phép tính tổng có trọng số được thực hiện, các trọng số được gán cho mẫu dự báo ngang và mẫu dự báo dọc có thể được xác định dựa trên vị trí của mẫu dự báo.

Khi phương thức dự báo định hướng được chọn, thông số biểu thị hướng dự báo (hoặc góc dự báo) của phương thức dự báo định hướng được chọn có thể được xác định. bảng 5 dưới đây thể hiện thông số nội hướng intraPredAng cho mỗi phương thức dự báo nội ảnh.

[Bảng 5]

PredModeIntraIntra PredAng	1-	232	326	421	517	613	79
PredModeIntraIntra PredAng	85	92	100	11-2	12-5	13-9	14-13
PredModeIntraIntra PredAng	15-17	16-21	17-26	18-32	19-26	20-21	21-17
PredModeIntraIntra PredAng	22-13	23-9	24-5	25-2	260	272	285
PredModeIntraIntra PredAng	299	3013	3117	3221	3326	3432	

Bảng 5 thể hiện thông số nội hướng cho mỗi phương thức dự báo nội ảnh với chỉ số giữa 2 và 34 khi 35 các phương thức dự báo nội ảnh được xác định. Khi nhiều hơn 33 các phương thức dự báo nội ảnh định hướng được xác định, bảng 5 có thể được chia nhỏ để đặt các thông số nội hướng cho mỗi phương thức dự báo định hướng nội khung.

Sau khi sắp xếp các mẫu tham chiếu trên và các mẫu tham chiếu trái của khối hiện tại trong một đường, mẫu dự báo có thể thu được dựa trên trị số của thông số nội hướng. Trong trường hợp này, khi trị số của thông số nội hướng là âm, các mẫu tham chiếu trái

và các mẫu tham chiếu trên có thể được sắp xếp trong một đường.

Fig.32 và Fig.33 là các sơ đồ thể hiện ví dụ của mảng một chiều trong đó các mẫu tham chiếu được bố trí trong một đường.

Fig.32 thể hiện ví dụ của mảng dọc một chiều trong đó các mẫu tham chiếu được sắp xếp theo hướng dọc, và Fig.33 thể hiện ví dụ của mảng ngang một chiều trong đó các mẫu tham chiếu được sắp xếp theo hướng ngang. Các phương án trên Fig.32 và Fig.33 sẽ được mô tả dựa trên giả định rằng 35 phương thức dự báo nội ảnh được xác định.

Mảng ngang một chiều trong đó các mẫu tham chiếu trên được xoay ngược chiều kim đồng hồ có thể được ứng dụng khi chỉ số phương thức dự báo nội ảnh là giữa 11 và 18, và mảng dọc một chiều trong đó mẫu tham chiếu trái được xoay theo chiều kim đồng hồ có thể là khi chỉ số phương thức dự báo nội ảnh là giữa 19 và 25. Khi các mẫu tham chiếu được bố trí trong một đường, góc phương thức dự báo nội ảnh có thể được xem xét.

Thông số xác định mẫu tham chiếu có thể được xác định dựa trên thông số nội hướng. Thông số xác định mẫu tham chiếu có thể gồm chỉ số mẫu tham chiếu để định rõ mẫu tham chiếu và thông số trọng số để xác định trọng số cần được ứng dụng cho mẫu tham chiếu.

Chỉ số mẫu tham chiếu i_{Idx} và thông số trọng số i_{fact} có thể thu được thông qua Các phương trình 2 và 3 dưới đây.

[Phương trình 2]

$$i_{Idx} = (y+1) * P_{ang} / 32$$

[Phương trình 3]

$$i_{fact} = [(y+1) * P_{ang}] \& 31$$

Trong các phương trình 2 và 3, P_{ang} thể hiện thông số nội hướng. Mẫu tham chiếu được định rõ bởi chỉ số mẫu tham chiếu i_{Idx} tương ứng với điểm ảnh số nguyên.

Để suy ra mẫu dự báo, ít nhất một mẫu tham chiếu có thể được định rõ. Cụ thể, vị trí của mẫu tham chiếu cần được sử dụng để suy ra mẫu dự báo có thể được định rõ xem xét độ nghiêng của phương thức dự báo. Theo một ví dụ, mẫu tham chiếu cần được sử dụng để suy ra mẫu dự báo có thể được định rõ sử dụng chỉ số mẫu tham chiếu i_{Idx} .

Trong trường hợp này, khi độ nghiêng của phương thức dự báo nội ảnh không thể được biểu hiện với một mẫu tham chiếu, mẫu dự báo có thể được tạo ra bằng cách nội suy nhiều mẫu tham chiếu. Theo một ví dụ, khi độ nghiêng của phương thức dự báo nội ảnh là trị số giữa độ nghiêng giữa mẫu dự báo và mẫu tham chiếu thứ nhất và độ nghiêng

giữa mẫu dự báo và mẫu tham chiếu thứ hai, mẫu dự báo có thể thu được bằng cách nội suy mẫu tham chiếu thứ nhất và mẫu tham chiếu thứ hai. Tức là, đường góc phù hợp với góc dự báo nội khung không đi qua mẫu tham chiếu được bố trí tại điểm ảnh số nguyên, mẫu dự báo có thể thu được bằng cách nội suy các mẫu tham chiếu gần kề phía trái, phía phải, phía đỉnh, hoặc phía đáy của vị trí mà đường góc đi qua.

Phương trình 4 dưới đây thể hiện ví dụ của thu được mẫu dự báo dựa trên các mẫu tham chiếu.

[Phương trình 4]

$$P(x,y)=((32-i_{fact})/32)*Ref_1D(x+iIdx+1)+(i_{fact}/32)*Ref_1D(x+iIdx+2)$$

Trong phương trình 4, P thể hiện mẫu dự báo, và Ref_1D thể hiện một trong số các mẫu tham chiếu được sắp xếp dưới dạng một chiều. Trong trường hợp này, vị trí của mẫu tham chiếu có thể được xác định bởi chỉ số mẫu tham chiếu iIdx và vị trí (x, y) của mẫu dự báo.

Khi độ nghiêng của phương thức dự báo nội ảnh có thể được biểu hiện với một mẫu tham chiếu, thông số trọng số i_{fact} có thể được đặt thành số không. Do đó, Phương trình 4 có thể được đơn giản hóa thành Phương trình 5 dưới đây.

[Phương trình 5]

$$P(x,y)=Ref_1D(x+iIdx+1)$$

Dự báo nội ảnh có thể được thực hiện trên khối hiện tại dựa trên nhiều phương thức dự báo nội ảnh. Theo một ví dụ, phương thức dự báo nội ảnh có thể được suy ra cho mỗi mẫu dự báo, và mẫu dự báo có thể được suy ra dựa trên phương thức dự báo nội ảnh được gán cho mẫu dự báo tương ứng.

Theo lựa chọn, phương thức dự báo nội ảnh có thể được suy ra cho mỗi khu vực, và dự báo nội ảnh có thể được thực hiện trên khu vực tương ứng dựa trên phương thức dự báo nội ảnh được gán cho khu vực tương ứng. Ở đây, khu vực có thể gồm ít nhất một mẫu. Ít nhất một trong số kích cỡ hoặc hình dạng của khu vực có thể được xác định theo cách thích ứng dựa trên ít nhất một trong số kích cỡ, hình dạng, hoặc phương thức dự báo nội ảnh của khối hiện tại. Theo lựa chọn, ít nhất một trong số kích cỡ hoặc hình dạng của khu vực có thể được xác định trước trong bộ mã hóa và bộ giải mã độc lập với kích cỡ hoặc hình dạng của khối hiện tại.

Theo lựa chọn, dự báo nội ảnh có thể được thực hiện nhiều lần, và mẫu dự báo cuối cùng có thể được suy ra dựa trên phép toán lấy trung bình hoặc phép tính tổng có

trọng số của nhiều các mẫu dự báo thu được bằng cách thực hiện dự báo nội ảnh nhiều lần. Theo một ví dụ, mẫu dự báo thứ nhất có thể thu được bằng cách thực hiện dự báo nội ảnh dựa trên phương thức dự báo nội ảnh thứ nhất, và mẫu dự báo thứ hai có thể thu được bằng cách thực hiện dự báo nội ảnh dựa trên phương thức dự báo nội ảnh thứ hai. Sau đó, mẫu dự báo cuối cùng có thể thu được dựa trên phép toán lấy trung bình hoặc phép tính tổng có trọng số giữa mẫu dự báo thứ nhất và mẫu dự báo thứ hai. Trong trường hợp này, các trọng số được gán cho mẫu dự báo thứ nhất và mẫu dự báo thứ hai có thể được xác định xem xét ít nhất một trong số việc liệu phương thức dự báo nội ảnh thứ nhất là phương thức dự báo không định hướng hoặc phương thức dự báo định hướng, việc liệu phương thức dự báo nội ảnh thứ hai là phương thức dự báo không định hướng hoặc phương thức dự báo định hướng, hoặc phương thức dự báo nội ảnh của khối lân cận.

Nhiều phương thức dự báo nội ảnh có thể là sự kết hợp của phương thức dự báo nội ảnh không định hướng và phương thức dự báo định hướng nội khung, sự kết hợp của các phương thức dự báo nội ảnh định hướng, hoặc sự kết hợp của các phương thức dự báo nội ảnh không định hướng.

Fig.34 là sơ đồ minh họa các góc được tạo ra giữa đường thẳng song song với trục x và các phương thức dự báo nội ảnh định hướng.

Trong ví dụ được thể hiện trên Fig.34, các phương thức dự báo định hướng có thể nằm trong khoảng từ hướng dưới-trái xiên đến hướng trên-phải xiên. Khi phần mô tả sau đây giả định góc được tạo ra giữa trục x và phương thức dự báo định hướng, các phương thức dự báo định hướng nằm trong khoảng giữa 45 độ (theo hướng dưới-trái xiên) và -135 độ (theo hướng trên-phải xiên).

Khi khối hiện tại không phải là hình vuông, mẫu dự báo có thể được suy ra sử dụng mẫu tham chiếu nằm cách xa so với mẫu dự báo trong số các mẫu tham chiếu được bố trí trên đường góc phù hợp với góc dự báo nội ảnh thay vì mẫu tham chiếu gần với mẫu dự báo phụ thuộc vào phương thức dự báo nội ảnh của khối hiện tại.

Fig.35 là sơ đồ thể hiện khía cạnh trong đó mẫu dự báo thu được khi khối hiện tại không phải là hình vuông.

Ví dụ như, như trong ví dụ được thể hiện trên Fig.35A, giả sử rằng khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao và phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo định hướng nội khung có góc giữa 0 độ và 45 độ. Trong trường hợp này, khi suy ra mẫu dự báo A gần cột phải của khối hiện tại, mẫu tham chiếu trái L nằm cách xa so với mẫu dự báo hơn là mẫu tham chiếu trên T

gần với mẫu dự báo trong số các mẫu tham chiếu được bố trí trên đường góc phù hợp với góc có thể được sử dụng.

Theo một ví dụ khác, như trong ví dụ được thể hiện trên Fig.35B, giả sử rằng khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng và phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo định hướng nội khung có góc giữa -90 độ và -135 độ. Trong trường hợp này, khi suy ra mẫu dự báo A gần hàng đáy của khối hiện tại, mẫu tham chiếu trên T nằm cách xa so với mẫu dự báo trong số các mẫu tham chiếu được bố trí trên đường góc phù hợp với góc có thể được sử dụng thay vì mẫu tham chiếu trái L gần với mẫu dự báo.

Để giải quyết vấn đề như vậy, khi khối hiện tại không phải là hình vuông, phương thức dự báo nội ảnh của khối hiện tại có thể bị thay thế bằng phương thức dự báo nội ảnh theo hướng đối diện. Do đó, các phương thức dự báo định hướng có góc lớn hơn hoặc nhỏ hơn các phương thức dự báo định hướng được thể hiện trên Fig.31 có thể được sử dụng cho khối không phải hình vuông. Phương thức dự báo định hướng nội khung như vậy có thể được xác định là phương thức dự báo nội ảnh góc rộng. Phương thức dự báo nội ảnh góc rộng thể hiện phương thức dự báo định hướng nội khung mà không nằm trong khoảng giữa 45 độ và -135 độ.

Fig.36 là sơ đồ thể hiện các phương thức dự báo nội ảnh góc rộng.

Trong ví dụ được thể hiện trên Fig.36, các phương thức dự báo nội ảnh với chỉ số giữa -1 và -14 và các phương thức dự báo nội ảnh với chỉ số giữa 67 và 80 biểu thị các phương thức dự báo nội ảnh góc rộng.

Trên Fig.36, 14 các phương thức dự báo nội ảnh góc rộng có góc lớn hơn 45 độ (-1 đến -14) và 14 phương thức dự báo nội ảnh góc rộng có góc nhỏ hơn -135 độ (67 đến 80) được minh họa, nhưng số lượng lớn hơn hoặc nhỏ hơn của các phương thức dự báo nội ảnh góc rộng có thể được xác định.

Khi phương thức dự báo nội ảnh góc rộng được sử dụng, độ dài của các mẫu tham chiếu trên có thể được đặt thành $2W+1$, và độ dài của các mẫu tham chiếu trái có thể được đặt thành $2H+1$.

Khi phương thức dự báo nội ảnh góc rộng được sử dụng, mẫu A được thể hiện trên Fig.35A có thể được dự báo bằng cách sử dụng mẫu tham chiếu T, và mẫu A được thể hiện trên Fig.35B có thể được dự báo bằng cách sử dụng mẫu tham chiếu L.

Tổng số $67+N$ các phương thức dự báo nội ảnh có thể được sử dụng bằng cách thêm N phương thức dự báo nội ảnh góc rộng vào các phương thức dự báo nội ảnh hiện

có. Theo một ví dụ, bảng 6 thể hiện các thông số nội hướng của các phương thức dự báo nội ảnh khi 20 phương thức dự báo nội ảnh góc rộng được xác định.

[Bảng 6]

PredModeIntra	-10	-9	-8	-7	-6	-5	-4	-3	-2
intraPredAngle	114	93	79	68	60	54	49	45	39
PredModeIntra	-1	2	3	4	5	6	7	8	9
intraPredAngle	35	32	29	26	23	21	19	17	15
PredModeIntra	10	11	12	13	14	15	16	17	18
intraPredAngle	13	11	9	7	5	3	2	1	0
PredModeIntra	19	20	21	22	23	24	25	26	27
intraPredAngle	-1	-2	-3	-5	-7	-9	-11	-13	-15
PredModeIntra	28	29	30	31	32	33	34	35	36
intraPredAngle	-17	-19	-21	-23	-26	-29	-32	-29	-26
PredModeIntra	37	38	39	40	41	42	43	44	45
intraPredAngle	-23	-21	-19	-17	-15	-13	-11	-9	-7
PredModeIntra	46	47	48	49	50	51	52	53	54
intraPredAngle	-5	-3	-2	-1	0	1	2	3	5
PredModeIntra	55	56	57	58	59	60	61	62	63
intraPredAngle	7	9	11	13	15	17	19	21	23
PredModeIntra	64	65	66	67	68	69	70	71	72
intraPredAngle	26	29	32	35	39	45	49	54	60
PredModeIntra	73	74	75	76					
intraPredAngle	68	79	93	114					

Các thông số nội hướng có thể được đặt khác nhau dựa trên ít nhất một trong số kích cỡ, hình dạng, hoặc đường mẫu tham chiếu của khối hiện tại. Theo một ví dụ, thông số nội hướng cho phương thức dự báo nội ảnh khi khối hiện tại cụ thể là hình vuông có thể là khác với thông số nội hướng khi khối hiện tại không phải là hình vuông. Theo một ví dụ, thông số nội hướng intraPredAngle của phương thức dự báo nội ảnh #15 có thể có trị số lớn hơn khi khối hiện tại là hình vuông so với khi khối hiện tại không phải là hình vuông.

Theo lựa chọn, thông số nội hướng intraPredAngle của phương thức dự báo nội

ảnh #75 có thể có trị số lớn hơn khi chỉ số của đường mẫu tham chiếu của khối hiện tại là 1 hoặc lớn hơn khi chỉ số của đường mẫu tham chiếu của khối hiện tại là 0.

Khi khối hiện tại không phải là hình vuông và phương thức dự báo nội ảnh của khối hiện tại thu được ở công đoạn S2902 nằm trong dải biến đổi, phương thức dự báo nội ảnh của khối hiện tại có thể được biến đổi thành phương thức dự báo nội ảnh góc rộng. Dải biến đổi có thể được xác định dựa trên ít nhất một trong số kích cỡ, hình dạng, hoặc tỷ lệ của khối hiện tại. Ở đây, tỷ lệ có thể biểu thị tỷ lệ giữa độ rộng và độ cao của khối hiện tại.

Khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, dải biến đổi có thể được đặt trong dải từ chỉ số phương thức dự báo nội ảnh của hướng trên-phải xiên (ví dụ như, 66) đến chỉ số phương thức dự báo nội ảnh của hướng trên-phải xiên trừ N. Ở đây, N có thể được xác định dựa trên tỷ lệ của khối hiện tại. Khi phương thức dự báo nội ảnh của khối hiện tại nằm trong dải biến đổi, phương thức dự báo nội ảnh có thể được biến đổi thành phương thức dự báo nội ảnh góc rộng. Phép biến đổi có thể là trừ trị số được xác định trước từ phương thức dự báo nội ảnh, và trị số được xác định trước có thể là tổng số (ví dụ như, 67) của các phương thức dự báo nội ảnh trừ các phương thức dự báo nội ảnh góc rộng.

Theo phương án nêu trên, các phương thức dự báo nội ảnh #66 đến #53 có thể được biến đổi tương ứng thành các phương thức dự báo nội ảnh góc rộng #-1 đến #-14.

Khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng, dải biến đổi có thể được đặt trong dải từ chỉ số phương thức dự báo nội ảnh của hướng xiên dưới-trái (ví dụ như, 2) đến chỉ số phương thức dự báo nội ảnh của hướng xiên dưới-trái plus M. Ở đây, M có thể được xác định dựa trên tỷ lệ của khối hiện tại. Khi phương thức dự báo nội ảnh của khối hiện tại nằm trong dải biến đổi, phương thức dự báo nội ảnh có thể được biến đổi thành phương thức dự báo nội ảnh góc rộng. Phép biến đổi có thể là cộng trị số được xác định trước vào phương thức dự báo nội ảnh, và trị số được xác định trước có thể là tổng số (ví dụ như, 65) của các phương thức dự báo nội ảnh định hướng loại trừ các phương thức dự báo nội ảnh góc rộng.

Theo phương án nêu trên, các phương thức dự báo nội ảnh #2 đến #15 có thể được biến đổi tương ứng thành các phương thức dự báo nội ảnh góc rộng #67 đến #80.

Các phương thức dự báo nội ảnh nằm trong dải biến đổi sẽ được gọi là các phương thức dự báo nội ảnh thay thế góc rộng.

Dải biến đổi có thể được xác định dựa trên tỷ lệ của khối hiện tại. Theo một ví dụ,

bảng 7 và bảng 8 thể hiện dải biến đổi khi 35 các phương thức dự báo nội ảnh trừ các phương thức dự báo nội ảnh góc rộng được xác định và dải biến đổi khi 67 phương thức dự báo nội ảnh được xác định.

[Bảng 7]

Điều kiện	Các phương thức Dự báo nội ảnh bị thay thế
$W/H = 2$	Các phương thức 2, 3, 4
$W/H > 2$	Các phương thức 2, 3, 4, 5, 6
$W/H = 1$	Không có
$H/W = 1/2$	Các phương thức 32, 33, 34
$H/W < 1/2$	Các phương thức 30, 31, 32, 33, 34

[Bảng 8]

Điều kiện	Các phương thức Dự báo nội ảnh bị thay thế
$W/H = 2$	Các phương thức 2, 3, 4, 5, 6, 7
$W/H > 2$	Các phương thức 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
$W/H = 1$	Không có
$H/W = 1/2$	Các phương thức 61, 62, 63, 64, 65, 66
$H/W < 1/2$	Các phương thức 57, 58, 59, 60, 61, 62, 63, 64, 65, 66

Trong các ví dụ được thể hiện trong bảng 7 và bảng 8, số lượng các phương thức dự báo nội ảnh thay thế góc rộng nằm trong dải biến đổi có thể khác nhau phụ thuộc vào tỷ lệ của khối hiện tại.

Khi các phương thức dự báo nội ảnh góc rộng được sử dụng ngoài các phương thức dự báo nội ảnh hiện có ra, các tài nguyên cần thiết cho việc mã hóa các phương thức dự báo nội ảnh góc rộng có thể được tăng lên, và do đó hiệu suất mã hóa có thể được giảm. Do đó, bằng cách mã hóa các phương thức dự báo nội ảnh thay thế cho các phương thức dự báo nội ảnh góc rộng, thay vì mã hóa các phương thức dự báo nội ảnh góc rộng theo thực tế, có thể cải thiện hiệu suất mã hóa.

Ví dụ như, khi khối hiện tại được mã hóa trong phương thức dự báo nội ảnh góc rộng #67, phương thức dự báo nội ảnh góc rộng #2, là phương thức dự báo nội ảnh góc rộng thay thế cho #67, có thể được mã hóa cho phương thức dự báo nội ảnh của khối hiện tại. Ngoài ra, khi khối hiện tại được mã hóa trong phương thức dự báo nội ảnh góc rộng #-1, phương thức dự báo nội ảnh góc rộng #66, là phương thức dự báo nội ảnh góc

rộng thay thế cho #-1, có thể được mã hóa cho phương thức dự báo nội ảnh của khối hiện tại.

Bộ giải mã có thể giải mã phương thức dự báo nội ảnh của khối hiện tại và có thể xác định việc liệu phương thức dự báo nội ảnh được giải mã nằm trong dải biến đổi. Khi phương thức dự báo nội ảnh được giải mã là phương thức dự báo nội ảnh góc rộng thay thế, phương thức dự báo nội ảnh có thể được biến đổi thành phương thức dự báo nội ảnh góc rộng.

Theo lựa chọn, khi khối hiện tại được mã hóa trong phương thức dự báo nội ảnh góc rộng, phương thức dự báo nội ảnh góc rộng có thể được mã hóa theo hoàn cảnh thực tế.

Việc mã hóa phương thức dự báo nội ảnh có thể được thực hiện dựa trên danh mục MPM được mô tả trên đây. Cụ thể, khi khối lân cận được mã hóa trong phương thức dự báo nội ảnh góc rộng, MPM có thể được đặt dựa trên phương thức dự báo nội ảnh góc rộng thay thế tương ứng với phương thức dự báo nội ảnh góc rộng. Theo một ví dụ, khi khối lân cận được mã hóa trong phương thức dự báo nội ảnh góc rộng, `candIntraPredX` biến thiên (X là A hoặc B) có thể được đặt là phương thức dự báo nội ảnh góc rộng thay thế.

Khi khối dự báo được tạo ra do thực hiện dự báo nội ảnh, các mẫu dự báo có thể được cập nhật dựa trên các vị trí của các mẫu dự báo được đưa vào khối dự báo. Phương pháp cập nhật có thể được gọi là phương pháp dự báo-có trọng số-nội ảnh-dựa trên-vị trí-mẫu (hoặc tổ hợp dự báo phụ thuộc vào vị trí (Position Dependent Prediction Combination - PDPC)).

Việc liệu có sử dụng PDPC hay không có thể được xác định xem xét phương thức dự báo nội ảnh của khối hiện tại, đường mẫu tham chiếu của khối hiện tại, kích cỡ của khối hiện tại, hoặc thành phần màu. Theo một ví dụ, PDPC có thể được sử dụng khi phương thức dự báo nội ảnh của khối hiện tại là ít nhất một trong số phương thức có trị số chỉ số nhỏ hơn Hai chiều, DC, Dọc, và Ngang hoặc phương thức có trị số chỉ số lớn hơn Phương thức ngang. Theo lựa chọn, PDPC có thể chỉ được sử dụng khi ít nhất một trong số độ rộng và độ cao của khối hiện tại là lớn hơn 4. Theo lựa chọn, PDPC có thể chỉ được sử dụng khi chỉ số của đường hình ảnh tham chiếu của khối hiện tại là 0. Theo lựa chọn, PDPC có thể chỉ được sử dụng khi chỉ số của đường hình ảnh tham chiếu của khối hiện tại là lớn hơn hoặc bằng trị số được xác định trước. Theo lựa chọn, PDPC có thể chỉ được sử dụng cho thành phần độ sáng. Theo lựa chọn, việc liệu có sử dụng PDPC

có thể được xác định theo việc liệu hai hoặc nhiều hơn hai trong số các điều kiện nêu trên được thỏa mãn.

Theo một ví dụ khác, thông tin biểu thị việc liệu PDPC được ứng dụng có thể được báo hiệu trong dòng bit.

Khi mẫu dự báo thu được thông qua mẫu dự báo nội ảnh, mẫu tham chiếu được sử dụng để hiệu chỉnh mẫu dự báo có thể được xác định dựa trên vị trí của mẫu dự báo thu được. Để thuận tiện cho việc mô tả, trong các phương án sau đây, mẫu tham chiếu được sử dụng để hiệu chỉnh mẫu dự báo sẽ được gọi là mẫu tham chiếu PDPC. Ngoài ra, mẫu dự báo thu được thông qua dự báo nội ảnh sẽ được gọi là mẫu dự báo thứ nhất, và mẫu dự báo thu được bằng cách chỉnh sửa mẫu dự báo thứ nhất sẽ được gọi là mẫu dự báo thứ hai.

Fig.37 là sơ đồ thể hiện khía cạnh ứng dụng của PDPC.

Mẫu dự báo thứ nhất có thể được điều chỉnh bằng cách sử dụng ít nhất một mẫu tham chiếu PDPC. Mẫu tham chiếu PDPC có thể gồm ít nhất một trong số mẫu tham chiếu gần kề góc trên-trái của khối hiện tại, mẫu tham chiếu trên được bố trí trên đỉnh của khối hiện tại, hoặc mẫu tham chiếu trái được bố trí ở bên trái của khối hiện tại.

Ít nhất một trong số các mẫu tham chiếu thuộc về đường mẫu tham chiếu của khối hiện tại có thể được đặt là mẫu tham chiếu PDPC. Theo lựa chọn, bất kể đường mẫu tham chiếu của khối hiện tại, ít nhất một mẫu tham chiếu thuộc về đường mẫu tham chiếu với chỉ số 0 có thể được đặt là mẫu tham chiếu PDPC. Theo một ví dụ, mặc dù mẫu dự báo thứ nhất thu được sử dụng các mẫu tham chiếu được đưa vào đường mẫu tham chiếu với chỉ số 1 hoặc chỉ số 2, mẫu dự báo thứ hai có thể thu được sử dụng các mẫu tham chiếu được đưa vào đường mẫu tham chiếu với chỉ số 0.

Số lượng hoặc các vị trí của mẫu tham chiếu PDPCs được sử dụng để hiệu chỉnh mẫu dự báo thứ nhất có thể được xác định xem xét ít nhất một trong số các phương thức dự báo nội ảnh của khối hiện tại, kích cỡ của khối hiện tại, hình dạng của khối hiện tại, hoặc vị trí của mẫu dự báo thứ nhất.

Ví dụ như, khi phương thức dự báo nội ảnh của khối hiện tại là Hai chiều hoặc Phương thức DC, mẫu dự báo thứ hai có thể thu được sử dụng mẫu tham chiếu trên và mẫu tham chiếu trái. Trong trường hợp này, mẫu tham chiếu trên có thể là mẫu tham chiếu vuông góc với mẫu dự báo thứ nhất (ví dụ như, mẫu tham chiếu có cùng tọa độ x), và mẫu tham chiếu trái có thể là mẫu tham chiếu song song với mẫu dự báo thứ nhất (ví dụ như, mẫu tham chiếu có cùng tọa độ y).

Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh ngang, mẫu dự báo thứ hai có thể thu được sử dụng mẫu tham chiếu trên. Trong trường hợp này, mẫu tham chiếu trên có thể là mẫu tham chiếu vuông góc với mẫu dự báo thứ nhất.

Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh dọc, mẫu dự báo thứ hai có thể thu được sử dụng mẫu tham chiếu trái. Trong trường hợp này, mẫu tham chiếu trái có thể là mẫu tham chiếu song song với mẫu dự báo thứ nhất.

Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh dưới-trái xiên hoặc phương thức dự báo nội ảnh trên-phải xiên, mẫu dự báo thứ hai có thể thu được dựa trên mẫu tham chiếu trên-trái, mẫu tham chiếu trên, và mẫu tham chiếu trái. Mẫu tham chiếu trên-trái có thể là mẫu tham chiếu gần kề góc trên-trái của khối hiện tại (ví dụ như, mẫu tham chiếu được bố trí tại $(-1, -1)$). Mẫu tham chiếu trên có thể là mẫu tham chiếu được bố trí theo hướng xiên phía trên và phía bên phải của mẫu dự báo thứ nhất, và mẫu tham chiếu trái có thể là mẫu tham chiếu được bố trí theo hướng xiên dưới đây và về phía trái của mẫu dự báo thứ nhất.

Tóm lại, khi vị trí của mẫu dự báo thứ nhất là (x, y) , $R(-1, -1)$ có thể được đặt là mẫu tham chiếu trên-trái, và $R(x+y+1, -1)$ hoặc $R(x, -1)$ có thể được đặt là mẫu tham chiếu trên. Ngoài ra, $R(-1, x+y+1)$ hoặc $R(-1, y)$ có thể được đặt là mẫu tham chiếu trái.

Theo một ví dụ khác, vị trí của mẫu tham chiếu trái hoặc mẫu tham chiếu trên có thể được xác định xem xét ít nhất một trong số hình dạng của khối hiện tại hoặc việc liệu phương thức nội ảnh góc rộng có được ứng dụng hay không.

Cụ thể, khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh góc rộng, mẫu tham chiếu được đặt cách nhau bởi độ lệch từ mẫu tham chiếu được bố trí nằm xiên từ mẫu dự báo thứ nhất có thể được đặt là mẫu tham chiếu PDPC. Theo một ví dụ, mẫu tham chiếu trên $R(x+y+k+1, -1)$ và mẫu tham chiếu trái $R(-1, x+y-k+1)$ có thể được đặt là các mẫu tham chiếu PDPC.

Trong trường hợp này, độ lệch k có thể được xác định dựa trên phương thức dự báo nội ảnh góc rộng. Phương trình 6 và phương trình 7 thể hiện ví dụ của việc suy ra độ lệch dựa trên phương thức dự báo nội ảnh góc rộng.

[Phương trình 6]

$$k = CurrIntraMode - 66$$

nếu $(CurrIntraMode > 66)$

[Phương trình 7]

$$k = -CurrIntraMode$$

nếu ($CurrIntraMode < 0$)

Mẫu dự báo thứ hai có thể được xác định dựa trên phép tính tổng có trọng số giữa mẫu dự báo thứ nhất và các mẫu tham chiếu PDPC. Theo một ví dụ, mẫu dự báo thứ hai có thể thu được dựa trên Phương trình 8 dưới đây.

[Phương trình 8]

$$pred(x,y) = (xL * R_L + wT * R_T - wTL * R_{TL} + (64 - wL - wT + wTL) * pred(x,y) + 32) >> 6$$

Trong phương trình 8, R_L thể hiện mẫu tham chiếu trái, R_T thể hiện mẫu tham chiếu trên, và R_{TL} thể hiện mẫu tham chiếu trên-trái. $pred(x,y)$ thể hiện mẫu dự báo được bố trí tại (x,y) . wL thể hiện trọng số được gán cho mẫu tham chiếu trái, wT thể hiện trọng số được gán cho mẫu tham chiếu trên, và wTL thể hiện trọng số được gán cho mẫu tham chiếu trên-trái. Trọng số được gán cho mẫu dự báo thứ nhất có thể được suy ra bằng cách trừ các trọng số được gán cho các mẫu tham chiếu từ trị số lớn nhất. Để thuận tiện cho việc mô tả, trọng số được gán cho mẫu tham chiếu PDPC sẽ được gọi là trọng số PDPC.

Trọng số được gán cho mỗi mẫu tham chiếu có thể được xác định dựa trên ít nhất một trong số phương thức dự báo nội ảnh của khối hiện tại hoặc vị trí của mẫu dự báo thứ nhất.

Ví dụ như, ít nhất một trong số wL , wT , hoặc wTL có thể là tỷ lệ trực tiếp hoặc tỷ lệ nghịch với ít nhất một trong số tọa độ x hoặc tọa độ y của mẫu dự báo. Theo lựa chọn, ít nhất một trong số wL , wT , hoặc wTL có thể là tỷ lệ trực tiếp hoặc tỷ lệ nghịch với ít nhất một trong số độ rộng hoặc độ cao của khối hiện tại.

Khi phương thức dự báo nội ảnh của khối hiện tại là các trọng số DC, PDPC có thể được xác định bởi Phương trình 9 dưới đây.

[Phương trình 9]

$$\begin{aligned} wT &= 32 >> ((y << 1) >> shift) \\ wL &= 32 >> ((x << 1) >> shift) \\ wTL &= (wL >> 4) + (wT >> 4) \end{aligned}$$

Trong phương trình 9, x và y thể hiện vị trí của mẫu dự báo thứ nhất.

Trong phương trình 9, độ dịch (shift) có thể biến đổi, được sử dụng cho phép tính độ dịch bit, có thể được suy ra dựa trên độ rộng (width) hoặc độ cao (height) của khối hiện tại. Theo một ví dụ, độ dịch (shift) có thể biến đổi có thể được suy ra dựa trên

Phương trình 10 hoặc 11 dưới đây.

[Phương trình 10]

$$shift = (\log_2(width) - 2 + \log_2(height) - 2 + 2) \gg 2$$

[Phương trình 11]

$$shift = ((\log_2(nTbW) + \log_2(nTbH) - 2) \gg 2)$$

Theo lựa chọn, độ dịch có thể biến đổi có thể được suy ra xem xét thông số nội hướng của khối hiện tại.

Số lượng hoặc các loại thông số được sử dụng để suy ra độ dịch có thể biến đổi có thể được xác định khác nhau phụ thuộc vào phương thức dự báo nội ảnh của khối hiện tại. Theo một ví dụ, khi phương thức dự báo nội ảnh của khối hiện tại là Hai chiều, DC, Dọc, hoặc Ngang, độ dịch có thể biến đổi có thể được suy ra sử dụng độ rộng và độ cao của khối hiện tại như trong ví dụ được thể hiện trong Phương trình 10 hoặc Phương trình 11. Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh có chỉ số lớn hơn phương thức dự báo nội ảnh dọc, độ dịch có thể biến đổi có thể được suy ra bằng cách sử dụng thông số nội hướng và độ cao của khối hiện tại. Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh có chỉ số nhỏ hơn phương thức dự báo nội ảnh ngang, độ dịch có thể biến đổi có thể được suy ra bằng cách sử dụng thông số nội hướng và độ rộng của khối hiện tại.

Khi phương thức dự báo nội ảnh của khối hiện tại là Hai chiều, trị số của wTL có thể được đặt thành 0. wL và wT có thể được suy ra dựa trên Phương trình 12 dưới đây.

[Phương trình 12]

$$wT[y] = 32 \gg ((y \ll 1) \gg nScale)$$

$$wL[x] = 32 \gg ((x \ll 1) \gg nScale)$$

Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh ngang, wT có thể được đặt thành 0, và wTL và wL có thể được đặt giống nhau. Mặt khác, khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh dọc, wL có thể được đặt thành 0, và wTL và wL có thể được đặt giống nhau.

Khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh lên trên và sang phải có trị số chỉ số lớn hơn phương thức dự báo nội ảnh dọc, các trọng số PDPC có thể được suy ra bằng cách sử dụng Phương trình 13 dưới đây.

[Phương trình 13]

$$\begin{aligned}
 wT &= 16 \gg ((y \ll 1) \gg \text{shift}) \\
 wL &= 16 \gg ((x \ll 1) \gg \text{shift}) \\
 wTL &= 0
 \end{aligned}$$

Mặt khác, khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh xuống dưới và sang trái có trị số chỉ số nhỏ hơn phương thức dự báo nội ảnh ngang, các trọng số PDPC có thể được suy ra bằng cách sử dụng Phương trình 14 dưới đây.

[Phương trình 14]

$$\begin{aligned}
 wT16 &\gg ((y \ll 1) \gg \text{shift}) \\
 wL &= 16 \gg ((x \ll 1) \gg \text{shift}) \\
 wTL &= 0
 \end{aligned}$$

Như trong các phương án được mô tả trên đây, các trọng số PDPC có thể được xác định dựa trên các vị trí x và y của mẫu dự báo.

Theo một ví dụ khác, trọng số được gán cho mỗi mẫu tham chiếu PDPC có thể được xác định trong các đơn vị của các khối con. Các mẫu dự báo được đưa vào khối con chia sẻ các trọng số PDPC giống nhau.

Kích cỡ của khối con, là đơn vị cơ bản để xác định các trọng số, có thể được xác định trước trong bộ mã hóa và bộ giải mã. Theo một ví dụ, các trọng số có thể được xác định cho các khối con 2×2 hoặc 4×4 .

Theo lựa chọn, kích cỡ, hình dạng, hoặc số lượng các khối con có thể được xác định phụ thuộc vào kích cỡ hoặc hình dạng của khối hiện tại. Theo một ví dụ, khối mã hóa có thể được chia tách thành bốn khối con bất kể kích cỡ của khối mã hóa. Theo lựa chọn, khối mã hóa có thể được chia tách thành bốn hoặc mười sáu các khối con phụ thuộc vào kích cỡ của khối mã hóa.

Theo lựa chọn, kích cỡ, hình dạng, hoặc số lượng các khối con có thể được xác định phụ thuộc vào phương thức dự báo nội ảnh của khối hiện tại. Theo một ví dụ, N cột (hoặc N hàng) có thể được đặt là một khối con khi phương thức dự báo nội ảnh của khối hiện tại là ngang, và N hàng (hoặc N cột) có thể được đặt là một khối con khi phương thức dự báo nội ảnh của khối hiện tại là dọc.

Các phương trình 15 đến 17 thể hiện ví dụ của xác định các trọng số PDPC cho khối con 2×2 . Phương trình 15 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức DC.

[Phương trình 15]

$$\begin{aligned}
wT &= 32 \gg (((y \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wL &= 32 \gg (((x \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wTL &= (wL \gg 4) + (wT \gg 4)
\end{aligned}$$

Trong phương trình 15, K có thể là trị số được xác định dựa trên phương thức dự báo nội ảnh hoặc kích cỡ của khối con.

Phương trình 16 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh lên trên và sang phải có trị số chỉ số lớn hơn phương thức dự báo nội ảnh dọc.

[Phương trình 16]

$$\begin{aligned}
wT &= 16 \gg (((y \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wL &= 16 \gg (((x \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wTL &= 0
\end{aligned}$$

Phương trình 17 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh xuống dưới và sang trái có trị số chỉ số nhỏ hơn phương thức dự báo nội ảnh ngang.

[Phương trình 17]

$$\begin{aligned}
wT &= 16 \gg (((y \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wL &= 16 \gg (((x \ll \log_2 K) \gg \log_2 K) \ll 1) \gg \text{shift}) \\
wTL &= 0
\end{aligned}$$

Trong các phương trình 15 đến phương trình 17, x và y thể hiện vị trí của mẫu tham chiếu trong khối con. Mẫu tham chiếu có thể là một trong số mẫu được bố trí trên góc trên-trái của khối con, mẫu được bố trí tại trung tâm của khối con, hoặc mẫu được bố trí trên góc dưới-phải của khối con.

Các phương trình 18 đến 20 thể hiện ví dụ của xác định các trọng số PDPC cho khối con 4×4. Phương trình 18 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức DC.

[Phương trình 18]

$$\begin{aligned}
wT &= 32 \gg (((y \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
wL &= 32 \gg (((x \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
wTL &= (wL \gg 4) + (wT \gg 4)
\end{aligned}$$

Phương trình 19 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh lên trên và sang phải có trị số chỉ số lớn hơn phương thức dự báo nội ảnh dọc.

[Phương trình 19]

$$\begin{aligned}
 wT &= 16 \gg (((y \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
 wL &= 16 \gg (((x \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
 wTL &= 0
 \end{aligned}$$

Phương trình 20 minh họa rằng phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh xuống dưới và sang trái có trị số chỉ số nhỏ hơn phương thức dự báo nội ảnh ngang.

[Phương trình 20]

$$\begin{aligned}
 wT &= 16 \gg (((y \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
 wL &= 16 \gg (((x \ll 2) \gg 2) \ll 1) \gg \text{shift}) \\
 wTL &= 0
 \end{aligned}$$

Trong các phương án nêu trên, các trọng số PDPC được mô tả được xác định xem xét các vị trí của các mẫu dự báo trong bao gồm khối con hoặc mẫu dự báo thứ nhất. Các trọng số PDPC có thể được xác định khi xem xét thêm khối hiện tại.

Ví dụ như, trong trường hợp Phương thức DC, phương pháp suy ra các trọng số PDPC có thể thay đổi tùy thuộc vào việc liệu khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao hoặc hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng.

Phương trình 21 thể hiện ví dụ của việc suy ra các trọng số PDPC khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, và Phương trình 22 thể hiện ví dụ của việc suy ra các trọng số PDPC khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng.

[Phương trình 21]

$$\begin{aligned}
 wT &= 32 \gg ((y \ll 1) \gg \text{shift}) \\
 wL &= 32 \gg (x \gg \text{shift}) \\
 wTL &= (wL \gg 4) + (wT \gg 4)
 \end{aligned}$$

[Phương trình 22]

$$\begin{aligned}
 wT &\gg (y \gg \text{shift}) \\
 wL &= 32 \gg ((x \ll 1) \gg \text{shift}) \\
 wTL &= (xL \gg 4) + (wT \gg 4)
 \end{aligned}$$

Khi khối hiện tại không phải là hình vuông, khối hiện tại có thể được dự báo bằng cách sử dụng phương thức dự báo nội ảnh góc rộng. Ngay cả khi dự báo nội ảnh góc rộng được ứng dụng, mẫu dự báo thứ nhất có thể được cập nhật bằng cách ứng dụng PDPC.

Khi dự báo nội ảnh góc rộng được ứng dụng cho khối hiện tại, các trọng số PDPC có thể được xác định xem xét hình dạng của khối mã hóa.

Ví dụ như, khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, mẫu tham chiếu trên được bố trí về phía phải của và phía trên mẫu dự báo thứ nhất có thể là gần hơn mẫu dự báo thứ nhất so với mẫu tham chiếu trái được bố trí về phía trái của và bên dưới mẫu dự báo thứ nhất phụ thuộc vào vị trí của mẫu dự báo thứ nhất. Do đó, khi mẫu dự báo thứ nhất được hiệu chỉnh, trọng số được ứng dụng cho mẫu tham chiếu trên có thể được đặt thành có trị số lớn hơn trọng số được ứng dụng cho mẫu tham chiếu trái.

Mặt khác, khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng, mẫu tham chiếu trái được bố trí về phía trái của và bên dưới mẫu dự báo thứ nhất có thể là gần hơn mẫu dự báo thứ nhất so với mẫu tham chiếu trên được bố trí về phía phải của và phía trên mẫu dự báo thứ nhất phụ thuộc vào vị trí của mẫu dự báo thứ nhất. Do đó, khi mẫu dự báo thứ nhất được hiệu chỉnh, trọng số được ứng dụng cho mẫu tham chiếu trái có thể được đặt thành có trị số lớn hơn trọng số được ứng dụng cho mẫu tham chiếu trên.

Phương trình 23 thể hiện ví dụ của việc suy ra các trọng số PDPC khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh góc rộng với chỉ số lớn hơn 66.

[Phương trình 23]

$$\begin{aligned}wT &= 16 \gg (y \gg \text{shift}) \\wL &= 16 \gg ((x \ll 1) \gg \text{shift}) \\wTL &= 0\end{aligned}$$

Phương trình 24 thể hiện ví dụ của việc suy ra các trọng số PDPC khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh góc rộng với chỉ số nhỏ hơn 0.

[Phương trình 24]

$$\begin{aligned}wT &= 16 \gg ((y \ll 1) \gg \text{shift}) \\wL &= 16 \gg (x \gg \text{shift}) \\wTL &= 0\end{aligned}$$

Các trọng số PDPC có thể được xác định dựa trên tỷ lệ của khối hiện tại. Tỷ lệ của khối hiện tại biểu thị tỷ lệ độ rộng và độ cao của khối hiện tại và có thể được xác định bằng cách sử dụng Phương trình 25 dưới đây.

[Phương trình 25]

$$whRatio = CUwidth / CUheight$$

Phương pháp suy ra các trọng số PDPC có thể được xác định có thể thay đổi được

phụ thuộc vào phương thức dự báo nội ảnh của khối hiện tại.

Ví dụ như, Phương trình 26 và phương trình 27 thể hiện ví dụ của việc suy ra các trọng số PDPC khi phương thức dự báo nội ảnh của khối hiện tại là DC. Cụ thể, Phương trình 26 là ví dụ trong đó khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, và Phương trình 27 là ví dụ trong đó khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng.

[Phương trình 26]

$$\begin{aligned}wT &= 32 \gg ((y \ll 1) \gg \text{shift}) \\wL &= 32 \gg (((x \ll 1) \gg whRatio) \gg \text{shift}) \\wTL &= (wL \gg 4) + (wT \gg 4)\end{aligned}$$

[Phương trình 27]

$$\begin{aligned}wT &= 32 \gg (((y \ll 1) \gg 1/whRatio) \gg \text{shift}) \\wL &= 32 \gg ((x \ll 1) \gg \text{shift}) \\wTL &= (wL \gg 4) + (wT \gg 4)\end{aligned}$$

Phương trình 28 thể hiện ví dụ của việc suy ra các trọng số PDPC khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh góc rộng với chỉ số lớn hơn 66.

[Phương trình 28]

$$\begin{aligned}wT &= 16 \gg (((y \ll 1) \gg 1/whRatio) \gg \text{shift}) \\wL &= 16 \gg ((x \ll 1) \gg \text{shift}) \\wTL &= 0\end{aligned}$$

Phương trình 29 thể hiện ví dụ của việc suy ra các trọng số PDPC khi phương thức dự báo nội ảnh của khối hiện tại là phương thức dự báo nội ảnh góc rộng với chỉ số nhỏ hơn 0.

[Phương trình 29]

$$\begin{aligned}wT &= 16 \gg ((y \ll 1) \gg \text{shift}) \\wL &= 16 \gg (((x \ll 1) \gg whRatio) \gg \text{shift}) \\wTL &= 0\end{aligned}$$

Khối còn lại có thể được suy ra bằng cách trừ phần dự báo từ khối gốc. Trong trường hợp này, khi khối còn lại bị thay đổi thành miền tần số, tính chất chủ quan của video không giảm đáng kể thậm chí nếu các thành phần tần số cao bị loại bỏ từ các thành phần tần số. Do đó, khi các trị số của các thành phần tần số cao bị thay đổi thành các trị số nhỏ hoặc khi các trị số của các thành phần tần số cao được đặt thành 0, có thể tăng lên hiệu suất nén mà không gây ra hiện tượng méo trực quan đáng kể. Xét đến các đặc điểm nêu trên, khối hiện tại có thể được biến đổi để phân giải khối còn lại thành hai thành

phần tần số hai chiều. Việc biến đổi có thể được thực hiện sử dụng kỹ thuật biến đổi chẳng hạn như biến đổi cosin rời rạc (Discrete Cosine Transform - DCT) hoặc biến đổi sin rời rạc (Discrete Sine Transform - DST).

DCT đang sử dụng phép biến đổi cosin để phân giải (hoặc biến đổi) khối còn lại thành hai thành phần tần số hai chiều, và DST đang sử dụng phép biến đổi sin để phân giải (hoặc biến đổi) khối còn lại thành hai thành phần tần số hai chiều. Do sự biến đổi của khối còn lại, các thành phần tần số có thể được biểu hiện với mẫu hình cơ bản. Theo một ví dụ, khi DCT được thực hiện trên khối $N \times N$, N^2 thành phần mẫu hình cơ bản có thể thu được. Kích cỡ của mỗi trong số thành phần mẫu hình cơ bản được đưa vào khối $N \times N$ có thể thu được thông qua phép biến đổi. Theo kỹ thuật biến đổi được sử dụng, kích cỡ của thành phần mẫu hình cơ bản có thể được gọi là hệ số DCT hoặc hệ số DST.

Kỹ thuật biến đổi DCT chủ yếu được sử dụng để biến đổi khối trong đó nhiều thành phần tần số thấp khác không được phân bố. Kỹ thuật biến đổi DST chủ yếu được sử dụng cho khối trong đó nhiều các thành phần tần số cao được phân bố.

Khối còn lại có thể bị biến đổi bằng cách sử dụng kỹ thuật biến đổi không phải là DCT hoặc DST.

Biến đổi khối còn lại thành hai thành phần tần số hai chiều sẽ được gọi là biến đổi hai chiều. Ngoài ra, kích cỡ của thành phần mẫu hình cơ bản thu được thông qua kết quả biến đổi sẽ được gọi là hệ số biến đổi. Theo một ví dụ, hệ số biến đổi có thể đề cập đến hệ số DCT hoặc hệ số DST. Khi cả sự biến đổi thứ nhất và sự biến đổi thứ hai cần được mô tả dưới đây được ứng dụng, hệ số biến đổi có thể đề cập đến kích cỡ của thành phần mẫu hình cơ bản được tạo ra do sự biến đổi thứ hai.

Kỹ thuật biến đổi có thể được xác định trong các đơn vị của các khối. Kỹ thuật biến đổi có thể được xác định dựa trên ít nhất một trong số phương thức dự báo của khối hiện tại hoặc kích cỡ của khối hiện tại. Theo một ví dụ, khi khối hiện tại được mã hóa trong phương thức dự báo nội ảnh và kích cỡ của khối hiện tại là nhỏ hơn $N \times N$, việc biến đổi có thể được thực hiện sử dụng kỹ thuật biến đổi DST. Mặt khác, khi điều kiện nêu trên không được thỏa mãn, việc biến đổi có thể được thực hiện sử dụng kỹ thuật biến đổi DCT.

Biến đổi hai chiều có thể không được thực hiện trên một số trong số các khối còn lại. Không thực hiện biến đổi hai chiều có thể được gọi là nhảy biến đổi. Khi nhảy biến đổi được ứng dụng, lượng tử hóa có thể được ứng dụng cho các trị số còn lại mà trên đó việc biến đổi không được thực hiện.

Sau khi khôi hiện tại được biến đổi bằng cách sử dụng DCT hoặc DST, khôi hiện tại bị biến đổi có thể được biến đổi lại. Trong trường hợp này, sự biến đổi dựa trên DCT hoặc DST có thể được xác định là sự biến đổi thứ nhất, và sự biến đổi lại của khôi phải chịu sự biến đổi thứ nhất có thể được xác định là sự biến đổi thứ hai.

Sự biến đổi thứ nhất có thể được thực hiện sử dụng một trong số nhiều ứng viên các lõi biến đổi. Theo một ví dụ, sự biến đổi thứ nhất có thể được thực hiện sử dụng một trong số DCT2, DCT8, hoặc DCT7.

Các lõi biến đổi khác nhau có thể được sử dụng theo hướng ngang và hướng dọc. Thông tin biểu thị sự kết hợp của lõi biến đổi dọc và lõi biến đổi ngang có thể được báo hiệu trong dòng bit.

Sự biến đổi thứ nhất và sự biến đổi thứ hai có thể được thực hiện trong các bộ phận khác nhau. Theo một ví dụ, sự biến đổi thứ nhất có thể được thực hiện trên khối 8×8 , và sự biến đổi thứ hai có thể được thực hiện trên khối con 4×4 của khối 8×8 được biến đổi. Trong trường hợp này, hệ số biến đổi của các khu vực còn lại mà ở đó sự biến đổi thứ hai không được thực hiện có thể được đặt thành 0.

Theo lựa chọn, sự biến đổi thứ nhất có thể được thực hiện trên khối 4×4 , và sự biến đổi thứ hai có thể được thực hiện trên khu vực của khối 8×8 gồm khối 4×4 được biến đổi.

Thông tin biểu thị việc liệu có thực hiện sự biến đổi thứ hai có thể được báo hiệu trong dòng bit.

Theo lựa chọn, việc liệu có thực hiện sự biến đổi thứ hai có thể được xác định dựa trên việc liệu lõi biến đổi ngang và lõi biến đổi dọc có giống nhau hay không. Theo một ví dụ, sự biến đổi thứ hai có thể được thực hiện chỉ khi lõi biến đổi ngang và lõi biến đổi dọc là giống nhau. Theo lựa chọn, sự biến đổi thứ hai có thể được thực hiện chỉ khi lõi biến đổi ngang và lõi biến đổi dọc là khác nhau.

Theo lựa chọn, sự biến đổi thứ hai có thể được phép chỉ khi biến đổi ngang và biến đổi dọc sử dụng lõi biến đổi được xác định trước. Theo một ví dụ, khi lõi biến đổi DCT2 được sử dụng cho biến đổi ngang và biến đổi dọc, sự biến đổi thứ hai có thể được phép.

Theo lựa chọn, việc liệu có thực hiện sự biến đổi thứ hai có thể được xác định dựa trên số lượng các hệ số biến đổi khác không của khối hiện tại. Theo một ví dụ, có thể đặt không sử dụng sự biến đổi thứ hai khi hệ số biến đổi khác không của khối hiện tại là nhỏ hơn hoặc bằng trị số ngưỡng, và có thể đặt để sử dụng sự biến đổi thứ hai khi hệ số biến

đổi khác không của khối hiện tại là lớn hơn trị số ngưỡng. Có thể đặt để sử dụng sự biến đổi thứ hai chỉ khi khối hiện tại được mã hóa thông qua dự báo nội ảnh.

Kích cỡ hoặc hình dạng của khối con cần phải chịu sự biến đổi thứ hai có thể được xác định dựa trên hình dạng của khối hiện tại.

Fig.38 và Fig.39 là các sơ đồ thể hiện khối con cần phải chịu sự biến đổi thứ hai.

Khi khối hiện tại là hình vuông, sự biến đổi thứ hai có thể được thực hiện trên khối con $N \times N$ được bố trí tại góc trên-trái của khối hiện tại sau khi sự biến đổi thứ nhất được thực hiện. Theo một ví dụ, khi khối hiện tại là khối mã hóa 8×8 , sự biến đổi thứ hai có thể được thực hiện trên khối con 4×4 được bố trí tại góc trên-trái của khối hiện tại sau khi sự biến đổi thứ nhất được thực hiện trên khối hiện tại (xem Fig.38).

Khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng gấp bốn hoặc nhiều hơn bốn lần độ cao, sự biến đổi thứ hai có thể được thực hiện trên khối con $kN \times 4kN$ được bố trí tại góc trên-trái của khối hiện tại sau khi sự biến đổi thứ nhất được thực hiện. Theo một ví dụ, khi khối hiện tại có hình dạng không phải hình vuông với kích cỡ là 16×4 , sự biến đổi thứ nhất có thể được thực hiện trên khối hiện tại, và sau đó sự biến đổi thứ hai có thể được thực hiện trên khối con 2×8 được bố trí tại góc trên-trái của khối hiện tại (xem Fig.39A).

Khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao gấp bốn hoặc nhiều hơn bốn lần độ rộng, sự biến đổi thứ hai có thể được thực hiện trên khối con $4kN \times kN$ được bố trí tại góc trên-trái của khối hiện tại sau khi sự biến đổi thứ nhất được thực hiện. Theo một ví dụ, khi khối hiện tại có hình dạng không phải hình vuông với kích cỡ là 16×4 , sự biến đổi thứ nhất có thể được thực hiện trên khối hiện tại, và sau đó sự biến đổi thứ hai có thể được thực hiện trên khối con 2×8 được bố trí tại góc trên-trái của khối hiện tại (xem Fig.39B).

Bộ giải mã có thể thực hiện biến đổi ngược của sự biến đổi thứ hai (biến đổi ngược thứ hai) và có thể thực hiện biến đổi ngược của sự biến đổi thứ nhất (biến đổi ngược thứ nhất) theo kết quả của biến đổi ngược thứ hai. Do thực hiện biến đổi ngược thứ hai và biến đổi ngược thứ nhất, các tín hiệu còn lại cho khối hiện tại có thể thu được.

Lượng tử hóa là để giảm năng lượng của khối, và các quy trình lượng tử hóa gồm chia hệ số biến đổi bởi trị số hằng số cụ thể. Trị số hằng số có thể được suy ra bởi thông số lượng tử hóa, và thông số lượng tử hóa có thể được xác định là trị số giữa 1 và 63.

Khi việc biến đổi và lượng tử hóa được thực hiện bởi bộ mã hóa, bộ giải mã có thể thu được khối còn lại thông qua lượng tử hóa ngược và biến đổi ngược. Bộ giải mã

có thể thu được khối được tái cấu trúc cho khối hiện tại bằng cách thêm khối dự báo và khối còn lại.

Thông tin biểu thị kiểu biến đổi của khối hiện tại có thể được báo hiệu trong dòng bit. Thông tin có thể là thông tin chỉ số `tu_mts_idx`, điều này biểu thị một trong số nhiều tổ hợp của kiểu biến đổi ngang và kiểu biến đổi dọc.

Lỗi biến đổi dọc và lỗi biến đổi ngang có thể được xác định dựa trên ứng viên kiểu biến đổi được định rõ bởi thông tin chỉ số `tu_mts_idx`. Bảng 9 và bảng 10 thể hiện các tổ hợp kiểu biến đổi tương ứng với `tu_mts_idx`.

[Bảng 9]

tu_mts_idx	Kiểu biến đổi	
	Dọc	Ngang
0	SKIP	SKIP
1	DCT-II	DCT-II
2	DST-VII	DST-VII
3	DCT-VIII	DST-VII
4	DST-VII	DCT-VIII
5	DCT-VIII	DCT-VIII

[Bảng 10]

tu_mts_idx	Kiểu biến đổi	
	Dọc	Ngang
0	DCT-II	DCT-II
1	SKIP	SKIP
2	DST-VII	DST-VII
3	DCT-VIII	DST-VII
4	DST-VII	DCT-VIII
5	DCT-VIII	DCT-VIII

Nội dung có thể được xác định là kiểu biến đổi là một trong số DCT2, DST7, DCT8, hoặc nhảy biến đổi. Theo lựa chọn, các ứng viên kết hợp kiểu biến đổi có thể được xây dựng chỉ sử dụng các lỗi biến đổi loại trừ nhảy biến đổi.

Khi bảng 9 được sử dụng và `tu_mts_idx` là 0, nhảy biến đổi có thể được ứng dụng

theo hướng ngang và hướng dọc. Khi `tu_mts_idx` là 1, DCT2 có thể được ứng dụng theo hướng ngang và hướng dọc. Khi `tu_mts_idx` là 3, DCT8 có thể được ứng dụng theo hướng ngang, và DCT7 có thể được ứng dụng theo hướng dọc.

Khi bảng 10 được sử dụng và `tu_mts_idx` là 0, DCT2 có thể được ứng dụng theo hướng ngang và hướng dọc. Khi `tu_mts_idx` là 1, nhảy biến đổi có thể được ứng dụng theo hướng ngang và hướng dọc. Khi `tu_mts_idx` là 3, DCT8 có thể được ứng dụng theo hướng ngang, và DCT7 có thể được ứng dụng theo hướng dọc.

Việc liệu để mã hóa thông tin chỉ số có thể được xác định dựa trên ít nhất một trong số kích cỡ hoặc hình dạng của khối hiện tại hoặc số lượng các hệ số khác không. Theo một ví dụ, khi số lượng các hệ số khác không là nhỏ hơn hoặc bằng trị số ngưỡng, thông tin chỉ số có thể không được báo hiệu, và kiểu biến đổi mặc định có thể được ứng dụng cho khối hiện tại. Ở đây, kiểu biến đổi mặc định có thể là DST7. Theo lựa chọn, phương thức mặc định có thể thay đổi phụ thuộc vào kích cỡ, hình dạng, hoặc phương thức dự báo nội ảnh của khối hiện tại.

Trị số ngưỡng có thể được xác định dựa trên kích cỡ hoặc hình dạng của khối hiện tại. Theo một ví dụ, trị số ngưỡng có thể được đặt thành 2 khi kích cỡ của khối hiện tại là nhỏ hơn hoặc bằng 32×32 , và trị số ngưỡng có thể được đặt thành 4 khi khối hiện tại là lớn hơn 32×32 (ví dụ như, khi khối hiện tại là khối mã hóa 32×64 hoặc 64×32).

Nhiều bảng tra cứu có thể được lưu trữ trước trong bộ mã hóa hoặc bộ giải mã. Ít nhất một trong số các trị số chỉ số được gán cho các ứng viên kết hợp kiểu biến đổi, các loại ứng viên kết hợp kiểu biến đổi, hoặc số lượng các ứng viên kết hợp kiểu biến đổi có thể khác nhau cho mỗi trong số nhiều bảng tra cứu.

Bảng tra cứu cho khối hiện tại có thể được chọn dựa trên ít nhất một trong số kích cỡ, hình dạng, phương thức dự báo, hoặc phương thức dự báo nội ảnh của khối hiện tại, việc liệu có ứng dụng phép biến đổi phụ, hoặc việc liệu có ứng dụng nhảy biến đổi cho khối lân cận.

Ví dụ như, bảng tra cứu của bảng 9 có thể được sử dụng khi kích cỡ của khối hiện tại là nhỏ hơn hoặc bằng 4×4 hoặc khi khối hiện tại được mã hóa thông qua dự báo liên ảnh, và bảng tra cứu của bảng 10 có thể được sử dụng khi kích cỡ của khối hiện tại là lớn hơn 4×4 hoặc khi khối hiện tại được mã hóa thông qua dự báo nội ảnh.

Theo lựa chọn, thông tin biểu thị một trong số nhiều bảng tra cứu có thể được báo hiệu trong dòng bit. Bộ giải mã có thể chọn bảng tra cứu cho khối hiện tại dựa trên thông tin.

Theo một ví dụ khác, chỉ số được gán cho tổ hợp ứng viên kiểu biến đổi có thể được xác định theo cách thích ứng dựa trên ít nhất một trong số kích cỡ, hình dạng, phương thức dự báo, hoặc phương thức dự báo nội ảnh của khối hiện tại, việc liệu có ứng dụng phép biến đổi phụ, hoặc việc liệu có ứng dụng nhảy biến đổi cho khối lân cận. Theo một ví dụ, chỉ số được gán cho nhảy biến đổi khi kích cỡ của khối hiện tại là 4×4 có thể có trị số nhỏ hơn chỉ số được gán cho nhảy biến đổi khi kích cỡ của khối hiện tại là lớn hơn 4×4 . Cụ thể, chỉ số 0 có thể được gán cho nhảy biến đổi khi kích cỡ của khối hiện tại là 4×4 , và chỉ số lớn hơn 0 (ví dụ như, chỉ số 1) có thể được gán cho nhảy biến đổi khi khối hiện tại là lớn hơn 4×4 và nhỏ hơn hoặc bằng 16×16 . Khi khối hiện tại là lớn hơn 16×16 , trị số lớn nhất (ví dụ như, 5) có thể được gán cho chỉ số của nhảy biến đổi.

Theo lựa chọn, khi khối hiện tại được mã hóa thông qua dự báo liên ảnh, chỉ số 0 có thể được ứng dụng cho nhảy biến đổi. Khi khối hiện tại được mã hóa thông qua dự báo nội ảnh, chỉ số lớn hơn 0 (ví dụ như, chỉ số 1) có thể được gán cho nhảy biến đổi.

Theo lựa chọn, khi khối hiện tại là khối 4×4 được mã hóa thông qua dự báo liên ảnh, chỉ số 0 có thể được gán cho nhảy biến đổi. Mặt khác, khi khối hiện tại là không được mã hóa thông qua dự báo liên ảnh hoặc khi khối hiện tại là lớn hơn 4×4 , chỉ số lớn hơn 0 (ví dụ như, chỉ số 1) có thể được gán cho nhảy biến đổi.

Các ứng viên kết hợp kiểu biến đổi mà khác với các ứng viên kết hợp kiểu biến đổi được liệt kê trong bảng 9 và bảng 10 có thể được xác định và được sử dụng. Theo một ví dụ, nhảy biến đổi có thể được ứng dụng cho một trong số biến đổi ngang hoặc biến đổi dọc, và tổ hợp ứng viên kiểu biến đổi được ứng dụng lỗi biến đổi, chẳng hạn như DCT7, DCT8, hoặc DST2, có thể được sử dụng cho kiểu biến đổi kia. Trong trường hợp này, việc liệu có sử dụng nhảy biến đổi cho ứng viên kiểu biến đổi ngang hoặc ứng viên kiểu biến đổi dọc có thể được xác định dựa trên ít nhất một trong số kích cỡ (ví dụ như, độ rộng và/hoặc độ cao), hình dạng, phương thức dự báo, hoặc phương thức dự báo nội ảnh của khối hiện tại.

Theo lựa chọn, thông tin biểu thị việc liệu ứng viên kiểu biến đổi cụ thể sẵn có có thể được báo hiệu trong dòng bit. Theo một ví dụ, cờ biểu thị việc liệu có sử dụng nhảy biến đổi cho hướng ngang và hướng dọc là ứng viên kiểu biến đổi có thể được báo hiệu. Việc liệu ứng viên kết hợp kiểu biến đổi cụ thể trong số nhiều ứng viên kết hợp kiểu biến đổi được đưa vào có thể được xác định theo cờ.

Theo lựa chọn, việc liệu ứng viên kiểu biến đổi cụ thể được ứng dụng cho khối hiện tại có thể được báo hiệu trong dòng bit. Theo một ví dụ, cờ `cu_mts_flag`, điều này

biểu thị việc liệu có ứng dụng DCT2 theo hướng ngang và hướng dọc, có thể được báo hiệu. Khi trị số của `cu_mts_flag` là 1, DCT2 có thể được đặt là lõi biến đổi cho hướng dọc và hướng ngang. Khi trị số của `cu_mts_flag` là 0, DCT8 hoặc DST7 có thể được đặt là lõi biến đổi cho hướng dọc và hướng ngang. Theo lựa chọn, khi trị số của `cu_mts_flag` là 0, thông tin `tu_mts_idx`, thông tin này định rõ một trong số nhiều ứng viên kết hợp kiểu biến đổi, có thể được báo hiệu.

Khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao hoặc hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng, việc mã hóa `cu_mts_flag` có thể bị bỏ qua, và trị số của `cu_mts_flag` có thể được coi là không.

Số lượng các ứng viên kết hợp kiểu biến đổi sẵn có có thể được đặt khác nhau phụ thuộc vào kích cỡ, hình dạng, hoặc phương thức dự báo nội ảnh của khối hiện tại. Theo một ví dụ, ba hoặc nhiều hơn ba ứng viên kết hợp kiểu biến đổi có thể được sử dụng khi khối hiện tại là hình vuông, và hai ứng viên kết hợp kiểu biến đổi có thể được sử dụng khi khối hiện tại không phải là hình vuông. Theo lựa chọn, khi khối hiện tại là hình vuông, chỉ các ứng viên kết hợp kiểu biến đổi có kiểu biến đổi ngang và kiểu biến đổi dọc khác với nhau trong số các ứng viên kết hợp kiểu biến đổi có thể được sử dụng.

Khi khối hiện tại có thể sử dụng ba hoặc nhiều hơn ba ứng viên kết hợp kiểu biến đổi, thông tin chỉ số `tu_mts_idx`, điều này biểu thị một trong số các ứng viên kết hợp kiểu biến đổi, có thể được báo hiệu. Mặt khác, khi khối hiện tại có thể sử dụng hai ứng viên kết hợp kiểu biến đổi, cờ `mts_flag`, điều này biểu thị một trong số các ứng viên kết hợp kiểu biến đổi, có thể được báo hiệu. Bảng 11 dưới đây thể hiện khía cạnh mã hóa của thông tin để định rõ các ứng viên kết hợp kiểu biến đổi theo hình dạng của khối hiện tại.

[Bảng 11]

<code>residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {</code>	Bộ mô tả
<code>...</code>	
<code>if(cu_mts_flag[x0][y0] && (cIdx == 0) && !transform_skip_flag[x0][y0][cIdx] && ((CuPredMode[x0][y0] == MODE_INTRA && numSigCoeff > 2) (CuPredMode[x0][y0] == MODE_INTER)) {</code>	
<code>if(cbHeight == cbWidth) {</code>	
<code>mts_idx[x0][y0]</code>	<code>ae(v)</code>

} else {	
mts_flag[x0][y0]	u(1)
}	
}	

Các chỉ số của các ứng viên kết hợp kiểu biến đổi có thể được bố trí lại (được sắp xếp lại thứ tự) theo hình dạng của khối hiện tại. Theo một ví dụ, các chỉ số được gán cho các ứng viên kết hợp kiểu biến đổi khi khối hiện tại là hình vuông có thể là khác nhau từ các chỉ số được gán cho các ứng viên kết hợp kiểu biến đổi khi khối hiện tại không phải là hình vuông. Theo một ví dụ, sự kết hợp kiểu biến đổi có thể được chọn dựa trên bảng 12 khi khối hiện tại là hình vuông, và sự kết hợp kiểu biến đổi có thể được chọn dựa trên bảng 13 khi khối hiện tại không phải là hình vuông.

[Bảng 12]

mts_idx	Nội khung		Liên khung	
	Lỗi biến đổi ngang	Lỗi biến đổi dọc	Lỗi biến đổi ngang	Lỗi biến đổi dọc
0	DST7	DST7	DCT8	DCT8
1	DCT8	DST7	DST7	DCT8
2	DST7	DCT8	DCT8	DST7
3	DCT8	DCT8	DST7	DST7

[Bảng 13]

mts_idx	Nội khung		Liên khung	
	Lỗi biến đổi ngang	Lỗi biến đổi dọc	Lỗi biến đổi ngang	Lỗi biến đổi dọc
0	DST8	DST7	DCT7	DCT8
1	DCT7	DST8	DST8	DCT7
2	DST7	DCT7	DCT7	DST7
3	DCT8	DCT8	DST7	DST7

Kiểu biến đổi có thể được xác định dựa trên số lượng các hệ số khác không ngang của khối hiện tại hoặc số lượng các hệ số khác không dọc của khối hiện tại. Ở đây, số lượng các hệ số khác không ngang biểu thị số lượng các hệ số khác không được đưa vào

$1 \times N$ (ở đây, N là độ rộng của khối hiện tại), và số lượng các hệ số khác không dọc biểu thị số lượng các hệ số khác không được đưa vào $N \times 1$ (ở đây, N là độ cao của khối hiện tại). Kiểu biến đổi thứ nhất có thể được ứng dụng theo hướng ngang khi trị số cực đại của các hệ số khác không ngang là nhỏ hơn hoặc bằng trị số ngưỡng, và kiểu biến đổi thứ hai có thể được ứng dụng theo hướng ngang khi trị số cực đại của các hệ số khác không ngang lớn hơn trị số ngưỡng. Kiểu biến đổi thứ nhất có thể được ứng dụng theo hướng dọc khi trị số cực đại của các hệ số khác không dọc nhỏ hơn hoặc bằng trị số ngưỡng, và kiểu biến đổi thứ hai có thể được ứng dụng theo hướng dọc khi trị số cực đại của các hệ số khác không dọc lớn hơn trị số ngưỡng.

Fig.40 là sơ đồ để mô tả ví dụ trong đó kiểu biến đổi của khối hiện tại được xác định.

Ví dụ như, khi khối hiện tại được mã hóa thông qua dự báo nội ảnh và trị số cực đại của các hệ số khác không ngang của khối hiện tại là nhỏ hơn hoặc bằng 2 (xem Fig.40A), DST7 có thể được xác định là kiểu biến đổi ngang.

Khi khối hiện tại được mã hóa thông qua dự báo nội ảnh và trị số cực đại của các hệ số khác không dọc của khối hiện tại là lớn hơn 2 (xem Fig.40B), DCT2 hoặc DCT8 có thể được xác định là kiểu biến đổi dọc.

Hệ số còn lại có thể được mã hóa trong các bộ phận biến đổi hoặc trong các bộ phận biến đổi con. Ở đây, hệ số còn lại đề cập đến hệ số biến đổi được tạo ra thông qua phép biến đổi, hệ số nhảy biến đổi được tạo ra thông qua nhảy biến đổi, hoặc hệ số được lượng tử hóa được tạo ra bằng cách lượng tử hóa hệ số hoặc hệ số biến đổi.

Bộ biến đổi có thể đề cập đến khối phải chịu sự biến đổi thứ nhất hoặc sự biến đổi thứ hai. Bộ biến đổi con đề cập đến khối nhỏ hơn bộ biến đổi. Theo một ví dụ, bộ biến đổi con có thể là khối 4×4 , 2×8 , hoặc 8×2 .

Ít nhất một trong số kích cỡ hoặc hình dạng của bộ biến đổi con có thể được xác định dựa trên kích cỡ hoặc hình dạng của khối hiện tại. Theo một ví dụ, bộ biến đổi con cũng được đặt để có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao (ví dụ như, 8×2) khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, và bộ biến đổi con cũng có thể được đặt để có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng (ví dụ như, 2×8) khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng. Khi khối hiện tại là hình vuông, bộ biến đổi con cũng có thể được đặt để có hình dạng là hình vuông (ví dụ như, 4×4).

Khi nhiều bộ phận biến đổi con được đưa vào trong khối hiện tại, các bộ phận

biến đổi con có thể được mã hóa hoặc được giải mã theo tuần tự. Hệ số còn lại có thể được mã hóa bằng cách sử dụng mã hóa entropi chẳng hạn như mã hóa số học. Phương pháp mã hóa hoặc giải mã hệ số còn lại sẽ được mô tả chi tiết dưới đây kết hợp với hình vẽ.

Fig.41 là lưu đồ thể hiện phương pháp mã hóa hệ số còn lại.

Trong phương án này, giả sử rằng khối hiện tại gồm một hoặc nhiều bộ phận biến đổi con. Ngoài ra, giả sử rằng bộ biến đổi con có kích cỡ là 4×4 . Tuy nhiên, phương án này có thể được ứng dụng theo điều kiện thực tế ngay cả khi kích cỡ của bộ biến đổi con hoặc hình dạng của bộ biến đổi con khác với nội dung được mô tả trên đây.

Việc liệu hệ số khác không có mặt trong khối hiện tại có thể được xác định (S4101). Hệ số khác không biểu thị hệ số còn lại có trị số tuyệt đối lớn hơn 0. Thông tin biểu thị việc liệu hệ số khác không có mặt trong khối hiện tại có thể được mã hóa và được báo hiệu. Theo một ví dụ, thông tin có thể là cờ 1 bit và có thể là cờ khối được mã hóa (Coded Block Flag - CBF).

Khi hệ số khác không có mặt trong khối hiện tại, việc liệu hệ số khác không có mặt cho mỗi bộ biến đổi con có thể được xác định (S4102). Thông tin biểu thị việc liệu hệ số khác không có mặt cho mỗi bộ biến đổi con có thể được mã hóa và được báo hiệu. Theo một ví dụ, thông tin có thể là cờ 1 bit và có thể là cờ khối con được mã hóa (Coded Sub-Block Flag - CSBF). Các bộ phận biến đổi con có thể được mã hóa trong thứ tự quét được chọn trước.

Khi hệ số khác không có mặt trong bộ biến đổi con, các hệ số còn lại trong bộ biến đổi con có thể được sắp xếp dưới dạng một chiều để mã hóa các hệ số còn lại của bộ biến đổi con (S4103). Các hệ số còn lại có thể được sắp xếp dưới dạng một chiều theo thứ tự quét được chọn trước.

Thứ tự quét có thể gồm ít nhất một trong số quét chéo, quét ngang, quét dọc hoặc các phương thức quét ngược của các hoạt động quét này.

Fig.42 và Fig.43 là các sơ đồ thể hiện thứ tự sắp xếp của các hệ số còn lại cho mỗi thứ tự quét.

Fig.42A đến Fig.42C thể hiện quét chéo, quét ngang, và quét dọc, và Fig.43A đến Fig.43C thể hiện các phương thức quét ngược của các hoạt động quét này.

Các hệ số còn lại có thể được sắp xếp dưới dạng một chiều theo thứ tự quét được chọn.

Thứ tự quét có thể được xác định xem xét ít nhất một trong số kích cỡ, hình dạng,

hoặc phương thức dự báo nội ảnh của khối hiện tại, lỗi biến đổi được sử dụng cho phép biến đổi chính, hoặc việc liệu có ứng dụng phép biến đổi phụ. Theo một ví dụ, khi khối hiện tại có hình dạng không phải hình vuông trong đó độ rộng lớn hơn độ cao, hệ số còn lại có thể được mã hóa bằng cách sử dụng quét ngang ngược. Mặt khác, khi khối hiện tại có hình dạng không phải hình vuông trong đó độ cao lớn hơn độ rộng, hệ số còn lại có thể được mã hóa bằng cách sử dụng quét dọc ngược.

Theo lựa chọn, tối ưu hóa tỷ lệ biến dạng (Rate Distortion Optimization - RDO) có thể được tính toán cho mỗi trong số nhiều thứ tự quét, và thứ tự quét có RDO thấp nhất có thể được xác định là thứ tự quét của khối hiện tại. Trong trường hợp này, thông tin biểu thị thứ tự quét của khối hiện tại có thể được mã hóa và được báo hiệu.

Ứng viên thứ tự quét có thể được sử dụng khi hệ số nhảy biến đổi được mã hóa có thể là khác nhau từ ứng viên thứ tự quét có thể được sử dụng khi hệ số biến đổi được mã hóa. Theo một ví dụ, sự đảo ngược ứng viên thứ tự quét có thể được sử dụng khi hệ số biến đổi được mã hóa có thể được đặt như ứng viên thứ tự quét có thể được sử dụng để mã hóa hệ số nhảy biến đổi.

Ví dụ như, khi hệ số biến đổi được mã hóa bằng cách sử dụng một trong số quét chéo ngược, quét ngang ngược, hoặc quét dọc ngược, hệ số nhảy biến đổi có thể được mã hóa bằng cách sử dụng một trong số quét chéo, quét ngang, hoặc quét dọc.

Tiếp theo, vị trí của hệ số khác không được quét lần cuối trong khối biến đổi có thể được mã hóa (S4103). Vị trí trục x và vị trí trục y của hệ số khác không cuối cùng có thể được mã hóa.

Fig.44 thể hiện ví dụ trong đó vị trí của hệ số khác không cuối cùng được mã hóa.

Trong ví dụ được thể hiện trên Fig.44, khi việc quét chéo được ứng dụng, hệ số còn lại được bố trí ở góc trên-phải của khối biến đổi có thể được đặt là hệ số khác không cuối cùng. Tọa độ x LastX của hệ số còn lại là 3, và tọa độ y LastY của hệ số còn lại là 0. LastX có thể được mã hóa trong khi được tách biệt thành `last_sig_coeff_x_prefix`, là phần tiền tố, và `last_sig_coeff_x_suffix`, là phần hậu tố. LastY cũng có thể được mã hóa trong khi được tách biệt thành `last_sig_coeff_y_prefix`, là phần tiền tố, và `last_sig_coeff_y_suffix`, là phần hậu tố.

Khi vị trí của hệ số khác không cuối cùng được xác định, thông tin biểu thị việc liệu các hệ số còn lại sớm hơn hệ số khác không cuối cùng theo thứ tự quét là các hệ số khác không có thể được mã hóa (S4104). Thông tin có thể là cờ 1 bit `sig_coeff_flag`. Trị số của cờ hệ số khác không `sig_coeff_flag` có thể được đặt thành 0 khi hệ số còn lại là 0,

và trị số của cờ hệ số khác không `sig_coeff_flag` có thể được đặt thành 1 khi hệ số còn lại là 1. Việc mã hóa cờ hệ số khác không `sig_coeff_flag` cho hệ số khác không cuối cùng có thể bị bỏ qua. Khi việc mã hóa cờ hệ số khác không `sig_coeff_flag` bị bỏ qua, hệ số còn lại tương ứng được coi là khác không.

Thông tin liên quan đến vị trí của hệ số khác không cuối cùng cho hệ số nhảy biến đổi có thể không được mã hóa. Trong trường hợp này, cờ hệ số khác không `sig_coeff_flag` cho tất cả các hệ số nhảy biến đổi trong khối có thể được mã hóa.

Theo lựa chọn, thông tin biểu thị việc liệu thông tin liên quan đến vị trí của hệ số khác không cuối cùng được mã hóa có thể được mã hóa và được báo hiệu. Thông tin có thể là cờ 1 bit. Bộ giải mã có thể xác định việc liệu có giải mã thông tin liên quan đến vị trí của hệ số khác không cuối cùng theo trị số của cờ. Khi vị trí của hệ số khác không cuối cùng không được giải mã, cờ hệ số khác không có thể được giải mã cho tất cả các hệ số nhảy biến đổi. Mặt khác, khi vị trí của hệ số khác không cuối cùng được giải mã, thông tin biểu thị việc liệu các hệ số nhảy biến đổi được quét sớm hơn hệ số khác không cuối cùng là các hệ số khác không có thể được mã hóa.

Thành phần DC của hệ số biến đổi hoặc hệ số nhảy biến đổi có thể không được đặt thành không. Ở đây, thành phần DC có thể biểu thị mẫu được quét cuối cùng hoặc mẫu được bố trí trên góc trên trái của khối. Đối với thành phần DC, việc mã hóa cờ hệ số khác không `sig_coeff_flag` có thể bị bỏ qua. Khi việc mã hóa cờ hệ số khác không `sig_coeff_flag` bị bỏ qua, hệ số còn lại tương ứng được coi là khác không.

Khi hệ số còn lại là khác không, thông tin biểu thị trị số tuyệt đối của hệ số còn lại và thông tin biểu thị ký hiệu của hệ số còn lại có thể được mã hóa (S4105). Quy trình mã hóa thông tin biểu thị trị số tuyệt đối của hệ số còn lại sẽ được mô tả chi tiết kết hợp với Fig.45.

Việc mã hóa hệ số còn lại có thể được thực hiện tuần tự trên các bộ phận biến đổi con được đưa vào khối hiện tại và tất cả các hệ số còn lại sau khi hệ số khác không cuối cùng (S4106 và S4107).

Fig.45 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Khi trị số tuyệt đối của hệ số còn lại là lớn hơn 0, thông tin biểu thị việc liệu hệ số còn lại hoặc hệ số còn lại là số chẵn hoặc số lẻ có thể được mã hóa (S4501). Ở đây, hệ số còn lại có thể được xác định là trị số tuyệt đối của hệ số còn lại trừ 1. Thông tin có thể là cờ chẵn lẻ 1 bit. Theo một ví dụ, trị số của phần tử cú pháp `par_level_flag` là 0 nghĩa là hệ số còn lại hoặc hệ số còn lại là số chẵn, và trị số của phần tử cú pháp `par_level_flag` là

1 nghĩa là hệ số còn lại hoặc hệ số còn lại là số lẻ. Trị số của phần tử cú pháp `par_level_flag` có thể được xác định bằng cách sử dụng Phương trình 30 dưới đây.

[Phương trình 30]

$$\text{par_level_flag} = (\text{abs}(\text{Tcoeff}) - 1) \% 2$$

Trong phương trình 30, `Tcoeff` thể hiện hệ số còn lại, và `abs()` thể hiện hàm trị số tuyệt đối.

Hệ số còn lại được điều chỉnh có thể được suy ra bằng cách chia hệ số còn lại hoặc hệ số còn lại cho 2 hoặc bằng cách ứng dụng phép toán dịch chuyển bit cho hệ số còn lại hoặc hệ số còn lại (S4502). Cụ thể, thương số thu được bằng cách chia hệ số còn lại cho 2 có thể được đặt là hệ số còn lại được điều chỉnh, hoặc trị số thu được bằng cách dịch chuyển hệ số còn lại dịch sang phải thêm 1 có thể được đặt là hệ số còn lại được điều chỉnh.

Ví dụ như, hệ số còn lại được điều chỉnh có thể được suy ra bằng cách sử dụng Phương trình 31 dưới đây.

[Phương trình 31]

$$\text{ReRemLevel} = \text{RemLevel} \gg 1$$

Trong phương trình 31, `ReRemLevel` thể hiện hệ số còn lại được điều chỉnh, và `RemLevel` thể hiện hệ số còn lại.

Tiếp theo, thông tin biểu thị kích cỡ của hệ số còn lại được điều chỉnh có thể được mã hóa. Thông tin biểu thị kích cỡ của hệ số còn lại được điều chỉnh có thể gồm thông tin biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn N. Ở đây, N là số tự nhiên chẳng hạn như 1, 2, 3, và 4.

Ví dụ như, thông tin biểu thị việc liệu trị số của hệ số còn lại được điều chỉnh là lớn hơn 1 có thể được mã hóa. Thông tin có thể là cờ 1 bit `rem_abs_gt1_flag`.

Khi hệ số còn lại là 0 hoặc 1, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Tức là, khi hệ số còn lại là 1 hoặc 2, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 0 khi hệ số còn lại là 0, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 1.

Khi hệ số còn lại được điều chỉnh là 2 hoặc nhiều hơn (S4504), trị số của `rem_abs_gt1_flag` có thể được đặt thành 1, và thông tin biểu thị việc liệu trị số của hệ số còn lại được điều chỉnh là lớn hơn 2 có thể được mã hóa (S4505). Thông tin có thể là cờ 1 bit `rem_abs_gt2_flag`.

Khi hệ số còn lại là 1 hoặc 2, trị số của `rem_abs_gt2_flag` có thể được đặt thành 0.

Tức là, khi hệ số còn lại là 3 hoặc 4, trị số của `rem_abs_gt2_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 2 khi hệ số còn lại là 0, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 3.

Khi hệ số còn lại được điều chỉnh là lớn hơn 2, thông tin trị số còn lại biểu thị hệ số còn lại được điều chỉnh trừ 2 có thể được mã hóa (S4506). Tức là, trị số thu được bằng cách trừ 5 từ trị số tuyệt đối của hệ số còn lại và sau đó chia kết quả của việc trừ đi 2 có thể được mã hóa là thông tin trị số còn lại.

Mặc dù không được thể hiện, hệ số còn lại có thể được mã hóa còn sử dụng cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 3 (ví dụ như, `rem_abs_gt3_flag`) hoặc cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 4 (ví dụ như, `rem_abs_gt4_flag`). Trong trường hợp này, hệ số còn lại được điều chỉnh trừ trị số lớn nhất có thể được đặt là trị số còn lại. Trị số lớn nhất thể hiện maximum N được ứng dụng cho `rem_abs_gtN_flag`.

Thay vì các cờ để so sánh trị số của hệ số còn lại được điều chỉnh với trị số cụ thể, các cờ để so sánh trị số tuyệt đối của hệ số được điều chỉnh hoặc hệ số còn lại với trị số cụ thể có thể được sử dụng. Theo một ví dụ, `gr2_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 2, có thể được sử dụng thay vì `rem_abs_gt1_flag`, và `gr4_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 4, có thể được sử dụng thay vì `rem_abs_gt2_flag`.

Bảng 14 thể hiện một cách đơn giản quy trình mã hóa các hệ số còn lại sử dụng các phần tử cú pháp.

[Bảng 14]

```

Sig_coeff_flag = (abs(TCoeff) != 0) ? 1 : 0
Mã hóa Sig_coeff_flag
nếu (Sig_coeff_flag) {
    RemLevel = abs(Tcoeff) - 1
    Parity_level_flag = RemLevel % 2
    Mã hóa parity_level_flag
    ReRemLevel = RemLevel >> 1
    Rem_abs_gt1_flag = (ReRemLevel > 1) ? 1 : 0
    Mã hóa rem_abs_gt1_flag
    nếu (Rem_abs_gt1_flag == 1) {
        Rem_abs_gt2_flag = absLevel > 4 ? 1 : 0
    }
}

```

```

nếu (Rem_abs_gt1_flag == 1) {
    Mã hóa rem_abs_gt2_flag
    Rem = (absLevel - 5) >> 1
    Mã hóa Rem
}
}
Sign_Coeff = TCoeff > 0 ? 0 : 1
Mã hóa Sign_Coeff
}

```

Trong ví dụ nêu trên, hệ số còn lại có thể được mã hóa bằng cách sử dụng `par_level_flag` và ít nhất một flag `rem_abs_gtN_flag` (ở đây, N là số tự nhiên chẳng hạn như 1, 2, 3, và 4).

`rem_abs_gtN_flag` thể hiện việc liệu hệ số còn lại có lớn hơn $2N$ hay không. Đối với hệ số còn lại $2N-1$ hoặc $2N$, `rem_abs_gt(N-1)_flag` được đặt là đúng, và `rem_abs_gtN_flag` được đặt là sai. Ngoài ra, trị số của `par_level_flag` có thể được đặt thành 0 cho hệ số còn lại $2N-1$, và trị số của `par_level_flag` có thể được đặt thành 1 cho hệ số còn lại $2N$. Tức là, các hệ số còn lại nhỏ hơn hoặc bằng $2N$ có thể được mã hóa bằng cách sử dụng `rem_abs_gtN_flag` và `par_level_flag`.

Đối với các hệ số còn lại lớn hơn hoặc bằng $2MAX+1$, thông tin trị số còn lại biểu thị trị số thu được bằng cách chia hiệu số với $2MAX$ by 2 có thể được mã hóa. Ở đây, MAX thể hiện trị số cực đại của N. Theo một ví dụ, khi `rem_abs_gt1_flag` và `rem_abs_gt2_flag` được sử dụng, MAX có thể là 2.

Bộ giải mã cũng có thể giải mã các hệ số còn lại theo thứ tự được thể hiện trên Fig.45. Cụ thể, bộ giải mã có thể xác định vị trí của hệ số khác không cuối cùng và có thể giải mã `sig_coeff_flag` cho mỗi hệ số còn lại mà được quét trước khi hệ số khác không cuối cùng.

Khi `sig_coeff_flag` là đúng, bộ giải mã có thể giải mã `par_level_flag` của hệ số còn lại tương ứng. Ngoài ra, bộ giải mã có thể giải mã `rem_abs_gt1_flag` của hệ số còn lại tương ứng. Trong trường hợp này, `rem_abs_gtN_flag` còn có thể được giải mã phụ thuộc vào trị số của `rem_abs_gt(N-1)_flag`. Theo một ví dụ, khi trị số của `rem_abs_gt(N-1)_flag` là 1, bộ giải mã có thể giải mã `rem_abs_gtN_flag`. Ví dụ như, khi trị số của `rem_abs_gt1_flag` cho hệ số còn lại là 1, bộ giải mã còn có thể phân tách `rem_abs_gt2_flag` cho hệ số còn lại. Khi trị số của `rem_abs_gt(MAX)_flag` là 1, bộ giải

mã có thể giải mã thông tin trị số còn lại.

Khi trị số của `rem_abs_gtN_flag` của hệ số còn lại là 0, trị số của hệ số còn lại có thể được xác định là $2N-1$ hoặc $2N$ phụ thuộc vào trị số của `par_level_flag`. Cụ thể, hệ số còn lại có thể được đặt thành $2N-1$ khi `par_level_flag` là 0, và hệ số còn lại có thể được đặt thành $2N$ khi `par_level_flag` là 1.

Ví dụ như, khi `rem_abs_gt1_flag` là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 1 hoặc 2 phụ thuộc vào trị số của `par_level_flag`. Cụ thể, trị số tuyệt đối của hệ số còn lại là 1 khi trị số của `par_level_flag` là 0, và trị số tuyệt đối của hệ số còn lại là 2 khi trị số của `par_level_flag` là 1.

Ví dụ như, khi `rem_abs_gt2_flag` là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 3 hoặc 4 phụ thuộc vào trị số của `par_level_flag`. Cụ thể, trị số tuyệt đối của hệ số còn lại là 3 khi trị số của `par_level_flag` là 0, và trị số tuyệt đối của hệ số còn lại là 4 khi trị số của `par_level_flag` là 1.

Khi thông tin trị số còn lại được giải mã, hệ số còn lại có thể được đặt thành $2(MAX+R)-1$ hoặc $2(MAX+R)$ phụ thuộc vào trị số của `par_level_flag`. Ở đây, R thể hiện trị số được biểu thị bởi thông tin trị số còn lại. Theo một ví dụ, hệ số còn lại có thể được đặt thành $2(MAX+R)-1$ khi `par_level_flag` là 0, và hệ số còn lại có thể được đặt thành $2(MAX+R)$ khi `par_level_flag` là 1. Theo một ví dụ, khi MAX là 2, hệ số còn lại có thể được suy ra bằng cách sử dụng Phương trình 32 dưới đây.

[Phương trình 32]

$$parity_level_flag + Rem \ll 1 + 5$$

Khi hệ số còn lại được mã hóa trong ví dụ được thể hiện trên Fig.45, cờ chặn lẻ có thể được mã hóa cho mỗi hệ số khác không. Theo một ví dụ, ngay cả khi hệ số còn lại là 1, `par_level_flag` và `rem_abs_gt1_flag` nên được mã hóa. Phương pháp mã hóa nêu trên có thể làm cho số lượng bit cần thiết để mã hóa các hệ số còn lại có trị số tuyệt đối của 1 được tăng lên. Để giải quyết vấn đề, thông tin biểu thị việc liệu hệ số còn lại là lớn hơn 1 có thể được mã hóa trước tiên, và cờ chặn lẻ có thể được mã hóa khi hệ số còn lại là 1.

Fig.46 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Đối với hệ số còn lại khác không, thông tin `gr1_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 1, có thể được mã hóa (S4601). `gr1_flag` có thể được đặt thành 0 khi hệ số còn lại là 1, và `gr1_flag` có thể được đặt thành 1 khi hệ số còn lại là lớn hơn 1.

Khi trị số tuyệt đối của hệ số còn lại là lớn hơn 1, cờ chặn lẻ biểu thị việc liệu hệ

số còn lại hoặc hệ số còn lại là số chẵn hoặc số lẻ có thể được mã hóa (S4602 và S4603). Ở đây, hệ số còn lại có thể được đặt thành hệ số còn lại trừ 2. Theo một ví dụ, `par_level_flag` có thể được suy ra bằng cách sử dụng Phương trình 33 dưới đây.

[Phương trình 33]

$$\text{par_level_flag} = (\text{abs}(T_{\text{coeff}}) - 2) \% 2$$

Hệ số còn lại được điều chỉnh có thể được suy ra bằng cách chia hệ số còn lại hoặc hệ số còn lại cho 2 hoặc bằng cách dịch chuyển bit hệ số còn lại hoặc hệ số còn lại dịch sang phải thêm 1, và sau đó thông tin biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 1 có thể được mã hóa (S4604). Theo một ví dụ, đối với hệ số còn lại có `gr1_flag` là 1, `rem_abs_gt1_flag`, điều này biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 1, có thể được mã hóa.

Khi hệ số còn lại là 0 hoặc 1, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Tức là, khi hệ số còn lại là 2 hoặc 3, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 0 khi hệ số còn lại được điều chỉnh là 0, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 1.

Khi hệ số còn lại được điều chỉnh là 2 hoặc nhiều hơn (S4605), trị số của `rem_abs_gt1_flag` có thể được đặt thành 1, và thông tin biểu thị việc liệu trị số của hệ số còn lại được điều chỉnh là lớn hơn 2 có thể được mã hóa (S4606). Theo một ví dụ, đối với hệ số còn lại có `rem_abs_gt1_flag` là 1, `rem_abs_gt2_flag`, điều này biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 2, có thể được mã hóa.

Khi hệ số còn lại là 2 hoặc 3, trị số của `rem_abs_gt2_flag` có thể được đặt thành 1. Tức là, khi hệ số còn lại là 4 hoặc 5, trị số của `rem_abs_gt2_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 0 khi hệ số còn lại là 2, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 3.

Khi hệ số còn lại được điều chỉnh là lớn hơn 2 (S4607), thông tin trị số còn lại biểu thị hệ số còn lại được điều chỉnh trừ 2 có thể được mã hóa (S4608). Tức là, trị số thu được bằng cách trừ 6 từ trị số tuyệt đối của hệ số còn lại và sau đó chia kết quả của việc trừ đi 2 có thể được mã hóa là thông tin trị số còn lại.

Hệ số còn lại có thể được mã hóa còn sử dụng cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 3 (ví dụ như, `rem_abs_gt3_flag`) hoặc cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 4 (ví dụ như, `rem_abs_gt4_flag`). Trong trường hợp này, hệ số còn lại được điều chỉnh trừ trị số lớn nhất có thể được đặt là trị số còn lại. Trị số lớn nhất thể hiện N lớn nhất được ứng dụng cho `rem_abs_gtN_flag`.

Thay vì các cờ đề so sánh trị số của hệ số còn lại được điều chỉnh với trị số cụ thể, các cờ đề so sánh trị số tuyệt đối của hệ số được điều chỉnh hoặc hệ số còn lại với trị số cụ thể có thể được sử dụng. Theo một ví dụ, `gr3_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 3, có thể được sử dụng thay vì `rem_abs_gt1_flag`, và `gr5_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 5, có thể được sử dụng thay vì `rem_abs_gt2_flag`.

Bảng 15 thể hiện một cách đơn giản quy trình mã hóa các hệ số còn lại sử dụng các phân tử cú pháp.

[Bảng 15]

```

Sig_coeff_flag = (abs(TCcoeff) != 0) ? 1 : 0
Mã hóa Sig_coeff_flag
nếu (Sig_coeff_flag) {
    Gr1_flag = abs(TCcoeff) > 1 ? 1 : 0
    Mã hóa Gr1_flag
    nếu (Gr1_flag) {
        RemLevel = abs(Tcoeff) - 2
        Parity_level_flag = RemLevel % 2
        Mã hóa Parity_level_flag
        ReRemLevel = RemLevel >> 1
        Rem_abs_gt1_flag = (ReRemLevel > 1) ? 1 : 0
        Mã hóa Rem_abs_gt1_flag
        nếu (Rem_abs_gt1_flag == 1) {
            Rem_abs_gt2_flag = absLevel > 5 ? 1 : 0
            nếu (Rem_abs_gt1_flag == 1) {
                Mã hóa Rem_abs_gt2_flag
                Rem = ( absLevel - 6 ) >> 1
                Mã hóa Rem
            }
        }
    }
    Sign_Coeff = TCcoeff > 0 ? 0 : 1
    Mã hóa Sign_Coeff
}

```

Bộ giải mã cũng có thể giải mã các hệ số còn lại theo thứ tự được thể hiện trên Fig.46. Cụ thể, bộ giải mã có thể xác định vị trí của hệ số khác không cuối cùng và có thể giải mã `sig_coeff_flag` cho mỗi hệ số còn lại mà được quét trước khi hệ số khác không cuối cùng.

Khi `sig_coeff_flag` là đúng, bộ giải mã có thể giải mã `gr1_flag` của hệ số còn lại tương ứng. Bộ giải mã có thể xác định rằng trị số tuyệt đối của hệ số còn lại tương ứng là 1 khi `gr1_flag` là 0 và có thể giải mã `par_level_flag` của hệ số còn lại tương ứng khi `gr1_flag` là 1. Sau đó, bộ giải mã có thể giải mã `rem_abs_gt1_flag` của hệ số còn lại tương ứng. Trong trường hợp này, `rem_abs_gtN_flag` còn có thể được giải mã phụ thuộc vào trị số của `rem_abs_gt(N-1)_flag`. Theo một ví dụ, khi trị số của `rem_abs_gt(N-1)_flag` là 1, bộ giải mã có thể giải mã `rem_abs_gtN_flag`. Ví dụ như, khi trị số của `rem_abs_gt1_flag` cho hệ số còn lại là 1, bộ giải mã còn có thể phân tách `rem_abs_gt2_flag` cho hệ số còn lại. Khi trị số của `rem_abs_gt(MAX)_flag` là 1, bộ giải mã có thể giải mã thông tin trị số còn lại.

Khi trị số của `rem_abs_gtN_flag` của hệ số còn lại là 0, trị số của hệ số còn lại có thể được xác định là $2N$ hoặc $2N+1$ phụ thuộc vào trị số của `par_level_flag`. Cụ thể, hệ số còn lại có thể được đặt thành $2N$ khi `par_level_flag` là 0, và hệ số còn lại có thể được đặt thành $2N+1$ khi `par_level_flag` là 1.

Ví dụ như, khi `rem_abs_gt1_flag` là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 2 hoặc 3 phụ thuộc vào trị số của `par_level_flag`. Cụ thể, trị số tuyệt đối của hệ số còn lại là 2 khi trị số của `par_level_flag` là 0, và trị số tuyệt đối của hệ số còn lại là 3 khi trị số của `par_level_flag` là 1.

Ví dụ như, khi `rem_abs_gt2_flag` là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 4 hoặc 5 phụ thuộc vào trị số của `par_level_flag`. Cụ thể, trị số tuyệt đối của hệ số còn lại là 4 khi trị số của `par_level_flag` là 0, và trị số tuyệt đối của hệ số còn lại là 5 khi trị số của `par_level_flag` là 1.

Khi thông tin trị số còn lại được giải mã, hệ số còn lại có thể được đặt thành $2(MAX+R)$ hoặc $2(MAX+R)+1$ phụ thuộc vào trị số của `par_level_flag`. Ở đây, R thể hiện trị số của thông tin trị số còn lại. Theo một ví dụ, hệ số còn lại có thể được đặt thành $2(MAX+R)$ khi `par_level_flag` là 0, và hệ số còn lại có thể được đặt thành $2(MAX+R)+1$ khi `par_level_flag` là 1.

Theo một ví dụ khác, cờ chặn lẻ có thể được mã hóa chỉ khi trị số của hệ số còn lại là lớn hơn 2. Theo một ví dụ, sau khi thông tin biểu thị việc liệu trị số của hệ số còn

lại là lớn hơn 1 và thông tin biểu thị việc liệu trị số của hệ số còn lại là lớn hơn 2 được mã hóa, cờ chắn lẻ cho hệ số còn lại tương ứng có thể được mã hóa khi nội dung được xác định là trị số của hệ số còn lại là lớn hơn 2.

Fig.47 là lưu đồ của quy trình mã hóa trị số tuyệt đối của hệ số còn lại.

Đối với hệ số còn lại khác không, thông tin `gr1_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 1, có thể được mã hóa (S4701). `gr1_flag` có thể được đặt thành 0 khi hệ số còn lại là 1, và `gr1_flag` có thể được đặt thành 1 khi hệ số còn lại là lớn hơn 1.

Đối với hệ số còn lại có trị số tuyệt đối lớn hơn 1, thông tin `gr2_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 2, có thể được mã hóa (S4702 và S4703). `gr2_flag` có thể được đặt thành 0 khi hệ số còn lại là 2, và `gr2_flag` có thể được đặt thành 1 khi hệ số còn lại là lớn hơn 2.

Khi trị số tuyệt đối lớn hơn 2, cờ chắn lẻ biểu thị việc liệu hệ số còn lại hoặc hệ số còn lại là số chẵn hoặc số lẻ có thể được mã hóa (S4704 và S4705). Ở đây, hệ số còn lại có thể được đặt thành hệ số còn lại trừ 3. Theo một ví dụ, `par_level_flag` có thể được suy ra bằng cách sử dụng Phương trình 34 dưới đây.

[Phương trình 34]

$$par_level_flag = (abs(Tcoeff) - 3) \% 2$$

Hệ số còn lại được điều chỉnh có thể được suy ra bằng cách chia hệ số còn lại hoặc hệ số còn lại cho 2 hoặc bằng cách dịch chuyển bit hệ số còn lại hoặc hệ số còn lại dịch sang phải thêm 1, và sau đó thông tin biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 1 có thể được mã hóa. Theo một ví dụ, đối với hệ số còn lại có `gr1_flag` là 1, `rem_abs_gt1_flag`, điều này biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 1, có thể được mã hóa (S4706).

Khi hệ số còn lại là 0 hoặc 1, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Tức là, khi hệ số còn lại là 3 hoặc 4, trị số của `rem_abs_gt1_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 0 khi hệ số còn lại được điều chỉnh là 0, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 1.

Khi hệ số còn lại được điều chỉnh là 2 hoặc nhiều hơn (S4707), trị số của `rem_abs_gt1_flag` có thể được đặt thành 1, và thông tin biểu thị việc liệu trị số của hệ số còn lại được điều chỉnh là lớn hơn 2 có thể được mã hóa (S4708). Theo một ví dụ, đối với hệ số còn lại có `rem_abs_gt1_flag` là 1, `rem_abs_gt2_flag`, điều này biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 2, có thể được mã hóa.

Khi hệ số còn lại là 2 hoặc 3, trị số của `rem_abs_gt2_flag` có thể được đặt thành 1. Tức là, khi hệ số còn lại là 5 hoặc 6, trị số của `rem_abs_gt2_flag` có thể được đặt thành 0. Trong trường hợp này, `par_level_flag` có thể được đặt thành 0 khi hệ số còn lại là 2, và `par_level_flag` có thể được đặt thành 1 khi hệ số còn lại là 3.

Khi hệ số còn lại được điều chỉnh là lớn hơn 2, trị số còn lại bằng hệ số còn lại được điều chỉnh trừ 2 có thể được mã hóa (S4709 và S4710). Tức là, trị số thu được bằng cách trừ 7 từ trị số tuyệt đối của hệ số còn lại và sau đó chia kết quả của việc trừ đi 2 có thể được mã hóa là trị số còn lại.

Hệ số còn lại có thể được mã hóa còn sử dụng cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 3 (ví dụ như, `rem_abs_gt3_flag`) hoặc cờ biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn 4 (ví dụ như, `rem_abs_gt4_flag`). Trong trường hợp này, hệ số còn lại được điều chỉnh trừ trị số lớn nhất có thể được đặt là trị số còn lại. Trị số lớn nhất thể hiện trị số N lớn nhất được ứng dụng cho `rem_abs_gtN_flag`.

Thay vì các cờ để so sánh trị số của hệ số còn lại được điều chỉnh với trị số cụ thể, các cờ để so sánh trị số tuyệt đối của hệ số được điều chỉnh hoặc hệ số còn lại với trị số cụ thể có thể được sử dụng. Theo một ví dụ, `gr4_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 4, có thể được sử dụng thay vì `rem_abs_gt1_flag`, và `gr6_flag`, điều này biểu thị việc liệu trị số tuyệt đối của hệ số còn lại là lớn hơn 6, có thể được sử dụng thay vì `rem_abs_gt2_flag`.

Bộ giải mã cũng có thể giải mã các hệ số còn lại theo cùng một thứ tự như được thể hiện trên Fig.47. Cụ thể, bộ giải mã có thể xác định vị trí của hệ số khác không cuối cùng và có thể giải mã `sig_coeff_flag` cho mỗi hệ số còn lại mà được quét trước khi hệ số khác không cuối cùng.

Khi `sig_coeff_flag` là đúng, bộ giải mã có thể giải mã `gr1_flag` của hệ số còn lại tương ứng. Bộ giải mã có thể xác định rằng trị số tuyệt đối của hệ số còn lại tương ứng là 1 khi `gr1_flag` là 0 và có thể giải mã `gr2_flag` khi `gr1_flag` là 1. Bộ giải mã có thể xác định rằng trị số tuyệt đối của hệ số còn lại tương ứng là 2 khi `gr2_flag` là 0 và có thể giải mã `par_level_flag` của hệ số còn lại tương ứng khi `gr2_flag` là 1. Sau đó, bộ giải mã có thể giải mã `rem_abs_gt1_flag` của hệ số còn lại tương ứng. Trong trường hợp này, `rem_abs_gtN_flag` còn có thể được giải mã phụ thuộc vào trị số của `rem_abs_gt(N-1)_flag`. Theo một ví dụ, khi trị số của `rem_abs_gt(N-1)_flag` là 1, bộ giải mã có thể giải mã `rem_abs_gtN_flag`. Ví dụ như, khi trị số của `rem_abs_gt1_flag` cho hệ số còn lại là 1, bộ giải mã còn có thể phân tách `rem_abs_gt2_flag` cho hệ số còn lại. Khi trị số của

rem_abs_gt(MAX)_flag là 1, bộ giải mã có thể giải mã thông tin trị số còn lại.

Khi trị số của rem_abs_gtN_flag của hệ số còn lại là 0, trị số của hệ số còn lại có thể được xác định là $2N+1$ hoặc $2(N+1)$ phụ thuộc vào trị số của par_level_flag. Cụ thể, hệ số còn lại có thể được đặt thành $2N+1$ khi par_level_flag là 0, và hệ số còn lại có thể được đặt thành $2(N+1)$ khi par_level_flag là 1.

Ví dụ như, khi rem_abs_gt1_flag là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 3 hoặc 4 phụ thuộc vào trị số của par_level_flag. Cụ thể, trị số tuyệt đối của hệ số còn lại là 3 khi trị số của par_level_flag là 0, và trị số tuyệt đối của hệ số còn lại là 4 khi trị số của par_level_flag là 1.

Ví dụ như, khi rem_abs_gt2_flag là 0, trị số tuyệt đối của hệ số còn lại có thể được đặt thành 5 hoặc 6 phụ thuộc vào trị số của par_level_flag. Cụ thể, trị số tuyệt đối của hệ số còn lại là 5 khi trị số của par_level_flag là 0, và trị số tuyệt đối của hệ số còn lại là 6 khi trị số của par_level_flag là 1.

Khi thông tin trị số còn lại được giải mã, hệ số còn lại có thể được đặt thành $2(MAX+R)$ hoặc $2(MAX+R)+1$ phụ thuộc vào trị số của par_level_flag. Ở đây, R thể hiện trị số của thông tin trị số còn lại. Theo một ví dụ, hệ số còn lại có thể được đặt thành $2(MAX+R)$ khi par_level_flag là 0, và hệ số còn lại có thể được đặt thành $2(MAX+R)+1$ khi par_level_flag là 1.

Ít nhất một trong số số lượng hoặc các loại cờ so sánh để so sánh các hệ số còn lại được điều chỉnh với trị số cụ thể dựa trên ít nhất một trong số kích cỡ hoặc hình dạng của khối hiện tại, sự có mặt của nhảy biến đổi, lỗi biến đổi, số lượng các hệ số khác không, hoặc vị trí của hệ số khác không cuối cùng có thể được xác định. Theo một ví dụ, khi hệ số biến đổi được mã hóa, chỉ rem_abs_gt1_flag có thể được sử dụng. Mặt khác, khi hệ số nhảy biến đổi được mã hóa, rem_abs_gt1_flag và rem_abs_gt2_flag có thể được sử dụng.

Theo lựa chọn, trong bộ biến đổi con 4×4 , số lượng flags rem_abs_gt1_flag có thể được đặt thành lên đến 8, và số lượng flags rem_abs_gt2_flag có thể được đặt thành lên đến 1. Theo lựa chọn, khi số lượng cờ các hệ số khác không là $(16-N)$, số lượng flags rem_abs_gt1_flag có thể được đặt thành lên đến $8+(N/2)$, và số lượng flags rem_abs_gt2_flag có thể được đặt thành lên đến $1+(N-(N/2))$.

Khi khối được tái cấu trúc của khối hiện tại thu được, tổn thất thông tin xảy ra trong quá trình lượng tử hóa và mã hóa có thể được giảm thông qua việc lọc trong vòng lặp. Bộ lọc trong vòng lặp có thể gồm ít nhất một trong số bộ lọc giải khối, bộ lọc độ lệch thích ứng mẫu (Sample Adaptive Offset Filter - SAO), hoặc bộ lọc vòng lặp thích

ứng (Adaptive Loop Filter - ALF). Khối được tái cấu trúc trước khi bộ lọc trong vòng lặp được ứng dụng sẽ được gọi là khối được tái cấu trúc thứ nhất, và khối được tái cấu trúc sau khi bộ lọc trong vòng lặp được ứng dụng sẽ được gọi là khối được tái cấu trúc thứ hai.

Khối được tái cấu trúc thứ hai có thể thu được bằng cách ứng dụng ít nhất một trong số bộ lọc giải khối, SAO, hoặc ALF cho khối được tái cấu trúc thứ nhất. Trong trường hợp này, SAO hoặc ALF có thể được ứng dụng sau khi bộ lọc giải khối được ứng dụng.

Bộ lọc giải khối là để giảm nhẹ việc làm giảm chất lượng video (thành phần lạ khối lớn) xảy ra tại mép khối khi lượng tử hóa được thực hiện trong các đơn vị của các khối. Để ứng dụng bộ lọc giải khối, cường độ tạo khối (Blocking Strength - BS) giữa khối được tái cấu trúc thứ nhất và khối được tái cấu trúc lân cận có thể được xác định.

Fig.48 là lưu đồ thể hiện quy trình xác định cường độ tạo khối.

Trong ví dụ được thể hiện trên Fig.48, P thể hiện khối được tái cấu trúc thứ nhất, và Q thể hiện khối được tái cấu trúc lân cận. Ở đây, khối được tái cấu trúc lân cận có thể là khối lân cận phía trái hoặc phía đỉnh của khối hiện tại.

Trong ví dụ được thể hiện trên Fig.48, điều được thể hiện là cường độ tạo khối được xác định xem xét các phương thức dự báo của P và Q, việc liệu các hệ số biến đổi khác không được đưa vào, việc liệu dự báo liên ảnh được thực hiện sử dụng cùng một hình ảnh tham chiếu, hoặc việc liệu trị số chênh lệch giữa các vector chuyển động lớn hơn hoặc bằng trị số ngưỡng.

Việc liệu có ứng dụng bộ lọc giải khối có thể được xác định dựa trên cường độ tạo khối. Theo một ví dụ, việc lọc có thể không được thực hiện khi cường độ tạo khối là 0.

SAO là để giảm bớt thành phần lạ rung lắc xảy ra khi lượng tử hóa được thực hiện trong miền tần số. SAO có thể được thực hiện bằng cách cộng hoặc trừ độ lệch được xác định xem xét mẫu hình của khối được tái cấu trúc thứ nhất. Phương pháp xác định độ lệch gồm độ lệch cạnh biên (Edge Offset - EO) hoặc độ lệch băng thông (Band Offset - BO). EO biểu thị phương pháp xác định độ lệch của mẫu hiện tại theo mẫu hình của các điểm ảnh gần đó. BO biểu thị phương pháp ứng dụng độ lệch chung cho tập hợp các điểm ảnh có các trị số độ sáng tương tự trong một khu vực. Cụ thể, độ sáng điểm ảnh được chia thành 32 phần bằng nhau, và các điểm ảnh với các trị số độ sáng tương tự có thể được đặt là một tập hợp. Theo một ví dụ, bốn băng thông gần kề trong số 32 băng thông có thể được đặt là một nhóm, và cùng một trị số độ lệch có thể được ứng dụng cho

các mẫu thuộc về bốn băng thông.

ALF là phương pháp tạo ra khối được tái cấu trúc thứ hai bằng cách ứng dụng bộ lọc có kích cỡ hoặc hình dạng được xác định trước cho khối được tái cấu trúc thứ nhất hoặc cho khối được tái cấu trúc được ứng dụng bộ lọc giải khối. Phương trình 35 dưới đây thể hiện ví dụ ứng dụng của ALF.

[Phương trình 35]

$$R'(i, j) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \sum_{l=-\frac{N}{2}}^{\frac{N}{2}} f(k, l) \cdot R(i + k, j + l)$$

Một trong số các ứng viên bộ lọc được xác định trước có thể được chọn trong các đơn vị của các hình ảnh, các đơn vị cây mã hóa, các khối mã hóa, các khối dự báo, hoặc các khối biến đổi. Các ứng viên bộ lọc có thể là khác nhau về kích cỡ hoặc hình dạng.

Fig.49 thể hiện các ứng viên bộ lọc được xác định trước.

Trong ví dụ được thể hiện trên Fig.49, ít nhất một trong số hình thoi 5×5, hình thoi 7×7, hoặc hình thoi 9×9 có thể được chọn.

Chỉ hình thoi 5×5 có thể được sử dụng cho thành phần màu.

Việc ứng dụng các phương án đã được mô tả tập trung vào quy trình giải mã hoặc quy trình mã hóa cho quy trình giải mã hoặc quy trình mã hóa nằm trong phạm vi của sáng chế này. Việc thay đổi thứ tự được xác định trước, trong đó các phương án đã được mô tả, sang thứ tự khác với thứ tự được mô tả trên đây cũng nằm trong phạm vi của sáng chế này.

Mặc dù các phương án nêu trên đã được mô tả dựa trên một loạt các bước hoặc các lưu đồ, các bước hoặc các lưu đồ không làm giới hạn thứ tự chuỗi thời gian của sáng chế và có thể được thực hiện đồng thời hoặc theo các thứ tự khác nhau theo yêu cầu. Ngoài ra, mỗi trong số các thành phần (ví dụ như, các đơn vị, các môđun, v.v.) cấu thành sơ đồ khối trong các phương án được mô tả trên đây có thể được thực hiện bởi thiết bị phần cứng hoặc phần mềm, và nhiều thành phần có thể được kết hợp và được thực hiện bởi thiết bị phần cứng đơn lẻ hoặc phần mềm. Các phương án nêu trên có thể được thực hiện dưới dạng các lệnh chương trình có thể được thực hiện thông qua các phần tử máy tính khác nhau và được ghi theo phương tiện ghi có thể đọc được bằng máy tính. Phương tiện ghi có thể đọc được bằng máy tính có thể gồm lệnh chương trình, tệp dữ liệu, cấu trúc dữ liệu, và phương tiện tương tự độc lập hoặc kết hợp với nhau. Các ví dụ của phương tiện ghi có thể đọc được bằng máy tính gồm phương tiện từ tính chẳng hạn như

đĩa cứng, đĩa mềm, và băng từ tính, phương tiện quang chẳng hạn như bộ nhớ chỉ đọc đĩa nén (Compact Disc-Read Only Memory - CD-ROM) và đĩa đa dụng số (Digital Versatile Disc - DVD), phương tiện từ quang chẳng hạn như đĩa mềm quang học, và các thiết bị phần cứng chẳng hạn như ROM, bộ nhớ truy cập ngẫu nhiên (Random Access Memory - RAM), và bộ nhớ cực nhanh, được thiết kế đặc biệt để lưu trữ và thực hiện các lệnh chương trình. Thiết bị phần cứng có thể được tạo cấu hình để vận hành với vai trò là một hoặc nhiều môđun phần mềm để thực hiện các hoạt động của sáng chế này, và ngược lại.

Khả năng ứng dụng trong công nghiệp

Sáng chế có thể được ứng dụng cho thiết bị điện tử để mã hóa hoặc giải mã video.

YÊU CẦU BẢO HỘ

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

phân tách cờ khác không biểu thị việc liệu hệ số còn lại là khác không từ dòng bit;
phân tách thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại từ
dòng bit khi cờ khác không biểu thị rằng hệ số còn lại là khác không; và

xác định trị số tuyệt đối của hệ số còn lại dựa trên thông tin trị số tuyệt đối,
trong đó

thông tin trị số tuyệt đối bao gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số
còn lại là lớn hơn trị số thứ nhất,

cờ chắn lẻ còn được phân tách từ dòng bit chỉ khi hệ số còn lại là lớn hơn trị số
thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ, và

cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại
được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm
1 lớn hơn trị số thứ hai còn được phân tách khi hệ số còn lại là lớn hơn trị số thứ nhất,

trong đó trị số thứ hai là N , hệ số còn lại được xác định là $2N$ hoặc $2N+1$ phụ
thuộc vào cờ chắn lẻ khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ
hai.

2. Phương pháp theo điểm 1, trong đó cờ so sánh hệ số còn lại được điều chỉnh
thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba còn được
phân tách khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

3. Phương pháp theo điểm 1, trong đó:

thông tin trị số còn lại còn được phân tách khi hệ số còn lại được điều chỉnh là lớn
hơn trị số thứ hai, và

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn
lại được điều chỉnh.

4. Phương pháp mã hóa video bao gồm các bước:

mã hóa cờ khác không biểu thị việc liệu hệ số còn lại là khác không; và

mã hóa thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại khi
hệ số còn lại là khác không,

trong đó

thông tin trị số tuyệt đối gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn
lại là lớn hơn trị số thứ nhất,

cờ chắn lẻ cho hệ số còn lại còn được mã hóa chỉ khi hệ số còn lại là lớn hơn trị số

thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ, và
 cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại
 được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm
 1 lớn hơn trị số thứ hai còn được mã hóa khi hệ số còn lại là lớn hơn trị số thứ nhất,

trong đó hệ số thứ hai là N , hệ số còn lại được xác định là $2N$ hoặc $2N+1$ phụ
 thuộc vào cờ chắn lẻ khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ
 hai.

5. Phương pháp theo điểm 4, trong đó cờ so sánh hệ số còn lại được điều chỉnh
 thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba còn được
 mã hóa khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

6. Phương pháp theo điểm 4, trong đó:

thông tin trị số còn lại còn được mã hóa khi hệ số còn lại được điều chỉnh là lớn
 hơn trị số thứ hai, và

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn
 lại được điều chỉnh.

7. Thiết bị giải mã video, thiết bị này bao gồm:

phương tiện phân tách cờ khác không biểu thị việc liệu hệ số còn lại là khác
 không từ dòng bit;

phương tiện phân tách thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ
 số còn lại từ dòng bit khi cờ khác không biểu thị rằng hệ số còn lại là khác không; và

phương tiện xác định trị số tuyệt đối của hệ số còn lại dựa trên thông tin trị số
 tuyệt đối,

trong đó

thông tin trị số tuyệt đối bao gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số
 còn lại là lớn hơn trị số thứ nhất, và

cờ chắn lẻ còn được phân tách từ dòng bit chỉ khi hệ số còn lại là lớn hơn trị số
 thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ,

trong đó thiết bị giải mã video còn bao gồm:

phương tiện phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ nhất
 biểu thị việc liệu hệ số còn lại được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ
 số còn lại dịch sang phải thêm 1 lớn hơn trị số thứ hai khi hệ số còn lại là lớn hơn trị số
 thứ nhất, và

phương tiện xác định hệ số còn lại là $2N$ hoặc $2N+1$ phụ thuộc vào cờ chắn lẻ khi

hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ hai, trong đó trị số thứ hai là N .

8. Thiết bị giải mã video theo điểm 7, thiết bị này còn bao gồm:

phương tiện phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

9. Thiết bị giải mã video theo điểm 7, thiết bị này còn bao gồm:

phương tiện phân tách thêm thông tin trị số còn lại khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai, trong đó

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn lại được điều chỉnh.

10. Thiết bị mã hóa video, thiết bị này bao gồm:

phương tiện mã hóa cờ khác không biểu thị việc liệu hệ số còn lại là khác không; và

phương tiện mã hóa thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại khi hệ số còn lại là khác không,

trong đó

thông tin trị số tuyệt đối gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn lại là lớn hơn trị số thứ nhất, và

cờ chắn lẻ cho hệ số còn lại còn được mã hóa chỉ khi hệ số còn lại là lớn hơn trị số thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ,

trong đó thiết bị mã hóa video bao gồm:

phương tiện phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm 1 lớn hơn trị số thứ hai khi hệ số còn lại là lớn hơn trị số thứ nhất, và

phương tiện xác định hệ số còn lại là $2N$ hoặc $2N+1$ phụ thuộc vào cờ chắn lẻ khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ hai, trong đó trị số thứ hai là N .

11. Thiết bị mã hóa video theo điểm 10, thiết bị này còn bao gồm:

phương tiện phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

12. Thiết bị mã hóa video theo điểm 10, thiết bị này còn bao gồm:

phương tiện phân tách thêm thông tin trị số còn lại khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai, trong đó

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn lại được điều chỉnh.

13. Thiết bị giải mã video, thiết bị này bao gồm:

bộ xử lý và bộ nhớ, trong đó bộ nhớ được tạo cấu hình để lưu trữ chương trình máy tính có thể thực thi trên bộ xử lý; và bộ xử lý được tạo cấu hình để:

phân tách cờ khác không biểu thị việc liệu hệ số còn lại là khác không từ dòng bit;

phân tách thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại từ dòng bit khi cờ khác không biểu thị rằng hệ số còn lại là khác không; và

xác định trị số tuyệt đối của hệ số còn lại dựa trên thông tin trị số tuyệt đối,

trong đó

thông tin trị số tuyệt đối bao gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn lại là lớn hơn trị số thứ nhất,

cờ chắn lẻ còn được phân tách từ dòng bit chỉ khi hệ số còn lại là lớn hơn trị số thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ,

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

phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm 1 lớn hơn trị số thứ hai khi hệ số còn lại là lớn hơn trị số thứ nhất; và

xác định hệ số còn lại là $2N$ hoặc $2N+1$ phụ thuộc vào cờ chắn lẻ khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ hai, trong đó trị số thứ hai là N .

14. Thiết bị theo điểm 13, trong đó bộ xử lý còn được tạo cấu hình để:

phân tách thêm cờ so sánh hệ số còn lại được điều chỉnh thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

15. Thiết bị theo điểm 13, trong đó bộ xử lý còn được tạo cấu hình để:

phân tách thêm thông tin trị số còn lại khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai, trong đó

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn lại được điều chỉnh.

16. Thiết bị mã hóa video, thiết bị này bao gồm: bộ xử lý và bộ nhớ, trong đó bộ

nhớ được tạo cấu hình để lưu trữ chương trình máy tính có thể thực thi trên bộ xử lý, và bộ xử lý được tạo cấu hình để:

mã hóa cờ khác không biểu thị việc liệu hệ số còn lại là khác không; và

mã hóa thông tin trị số tuyệt đối để xác định trị số tuyệt đối của hệ số còn lại khi hệ số còn lại là khác không,

trong đó

thông tin trị số tuyệt đối gồm cờ so sánh hệ số còn lại biểu thị việc liệu hệ số còn lại là lớn hơn trị số thứ nhất, và

cờ chắn lẻ cho hệ số còn lại còn được mã hóa chỉ khi hệ số còn lại là lớn hơn trị số thứ nhất; và cờ chắn lẻ biểu thị việc liệu trị số của hệ số còn lại là số chẵn hoặc số lẻ,

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

mã hóa thêm cờ so sánh hệ số còn lại được điều chỉnh thứ nhất biểu thị việc liệu hệ số còn lại được điều chỉnh được suy ra bằng cách dịch chuyển bit hệ số còn lại dịch sang phải thêm 1 lớn hơn trị số thứ hai khi hệ số còn lại là lớn hơn trị số thứ nhất, và

xác định hệ số còn lại là $2N$ hoặc $2N+1$ phụ thuộc vào cờ chắn lẻ khi hệ số còn lại được điều chỉnh là nhỏ hơn hoặc bằng trị số thứ hai, trong đó trị số thứ hai là N .

17. Thiết bị theo điểm 16, trong đó bộ xử lý còn được tạo cấu hình để:

mã hóa thêm cờ so sánh hệ số còn lại được điều chỉnh thứ hai biểu thị việc liệu hệ số còn lại được điều chỉnh là lớn hơn trị số thứ ba khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai.

18. Thiết bị theo điểm 16, trong đó bộ xử lý còn được tạo cấu hình để:

mã hóa thêm thông tin trị số còn lại khi hệ số còn lại được điều chỉnh là lớn hơn trị số thứ hai, trong đó

thông tin trị số còn lại gồm trị số thu được bằng cách trừ trị số thứ hai từ hệ số còn lại được điều chỉnh.

19. Phương tiện lưu trữ có thể đọc được bằng máy tính có chứa trên đó lệnh có thể thực thi trên bộ xử lý, mà khi được thực thi bởi bộ xử lý, khiến bộ xử lý thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ điểm 1 đến điểm 3.

20. Phương tiện lưu trữ có thể đọc được bằng máy tính có chứa trên đó lệnh có thể thực thi trên bộ xử lý, mà khi được thực thi bởi bộ xử lý, khiến bộ xử lý thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ điểm 4 đến điểm 6.

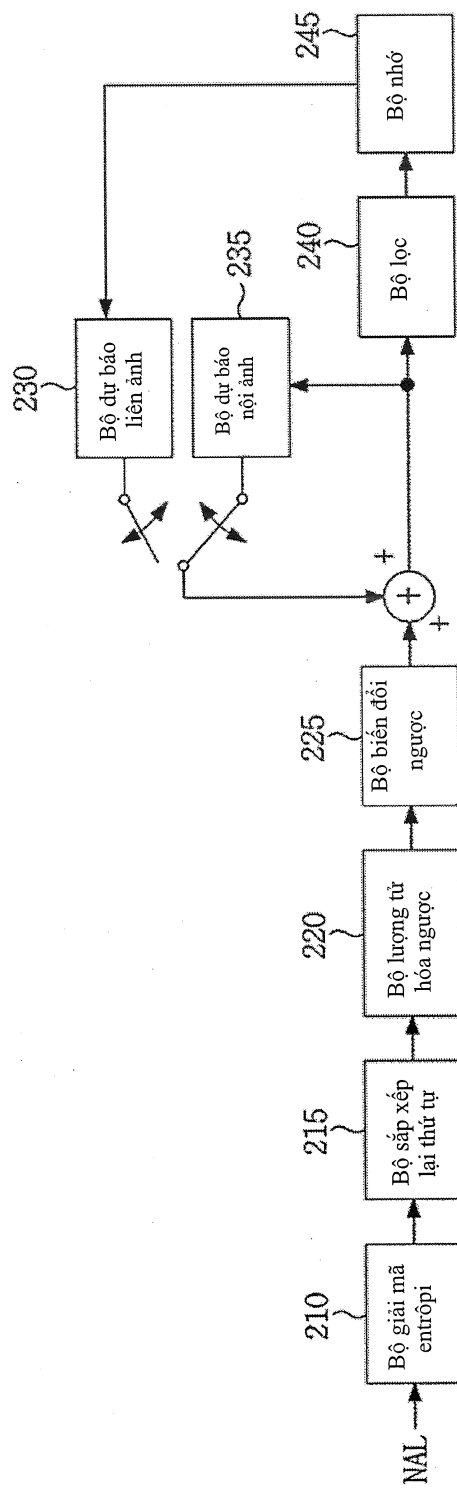
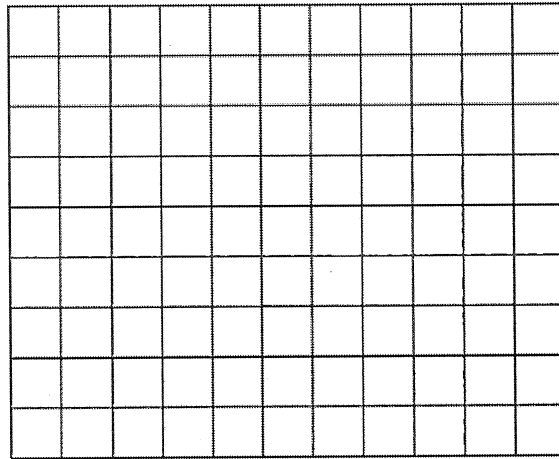
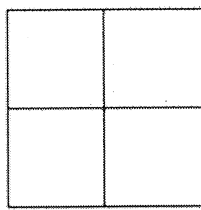
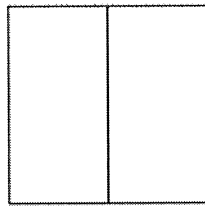
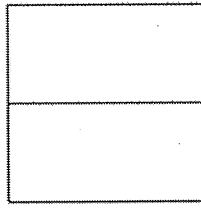
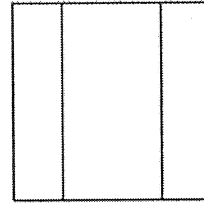
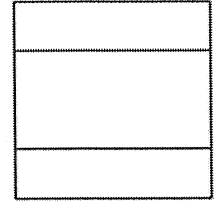


FIG. 2

**FIG. 3****SPLIT_QT****SPLIT_BT_VER****SPLIT_BT_HOR****SPLIT_TT_VER****SPLIT_TT_HOR****FIG. 4**

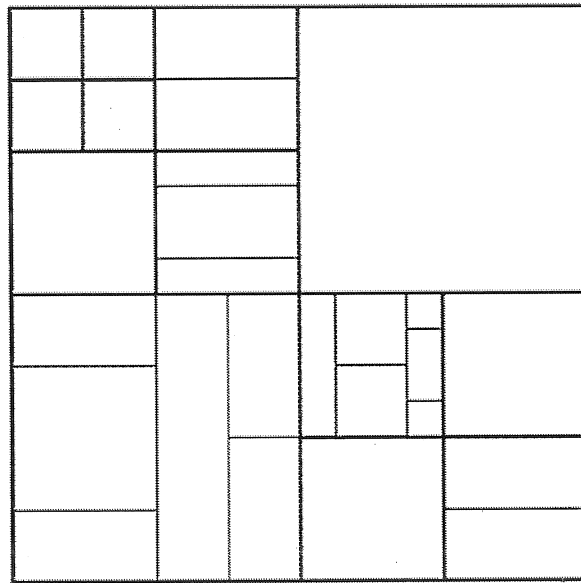


FIG. 5

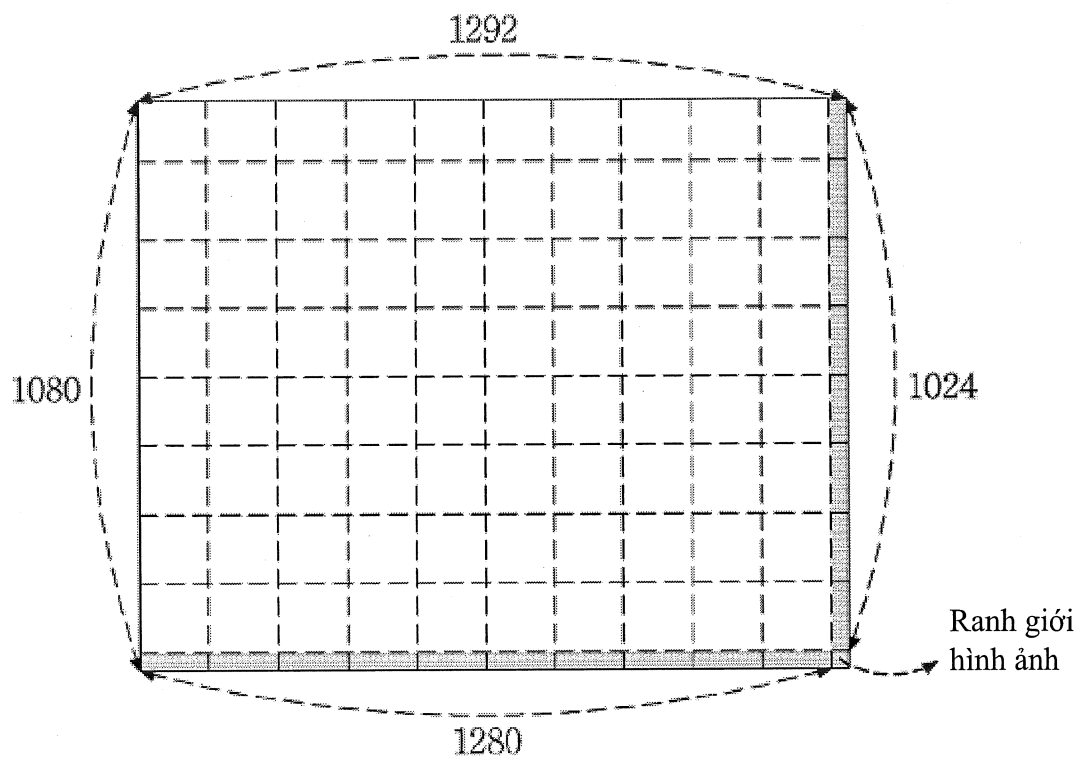


FIG. 6

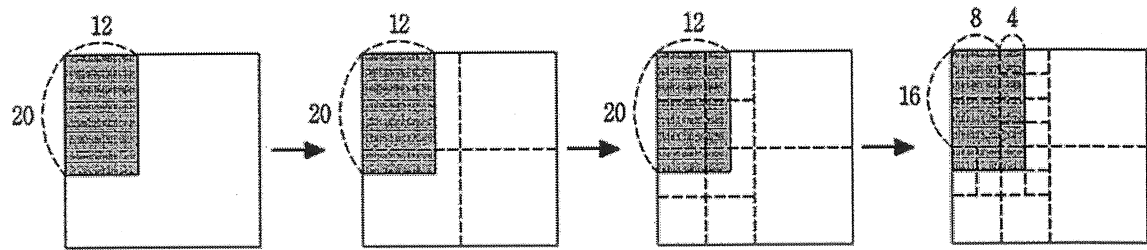


FIG. 7

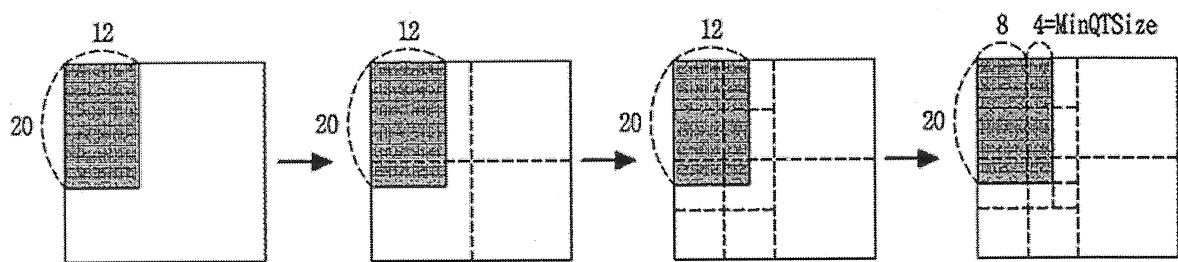


FIG. 8

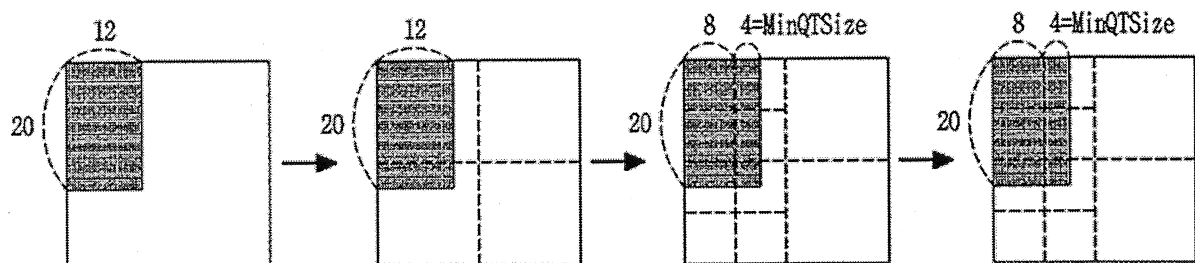


FIG. 9

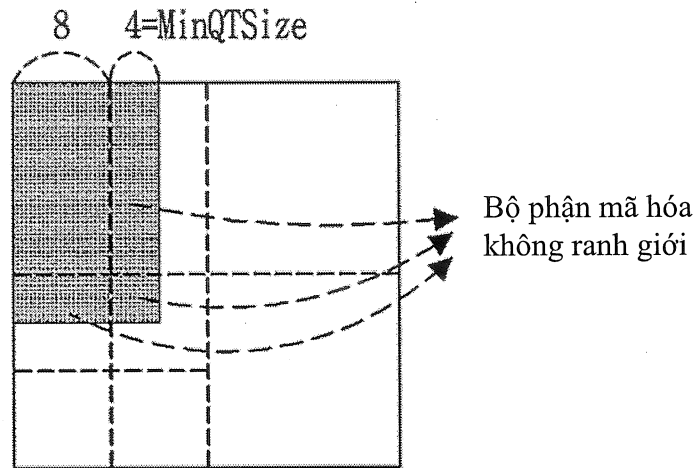


FIG. 10

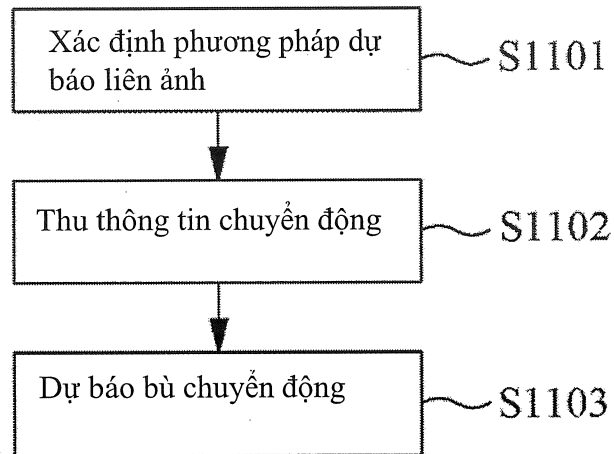


FIG. 11

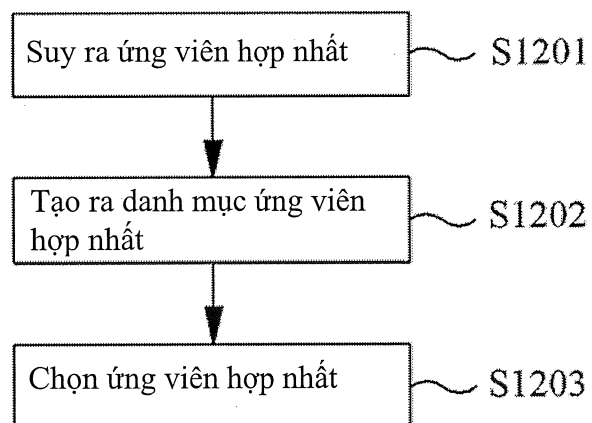
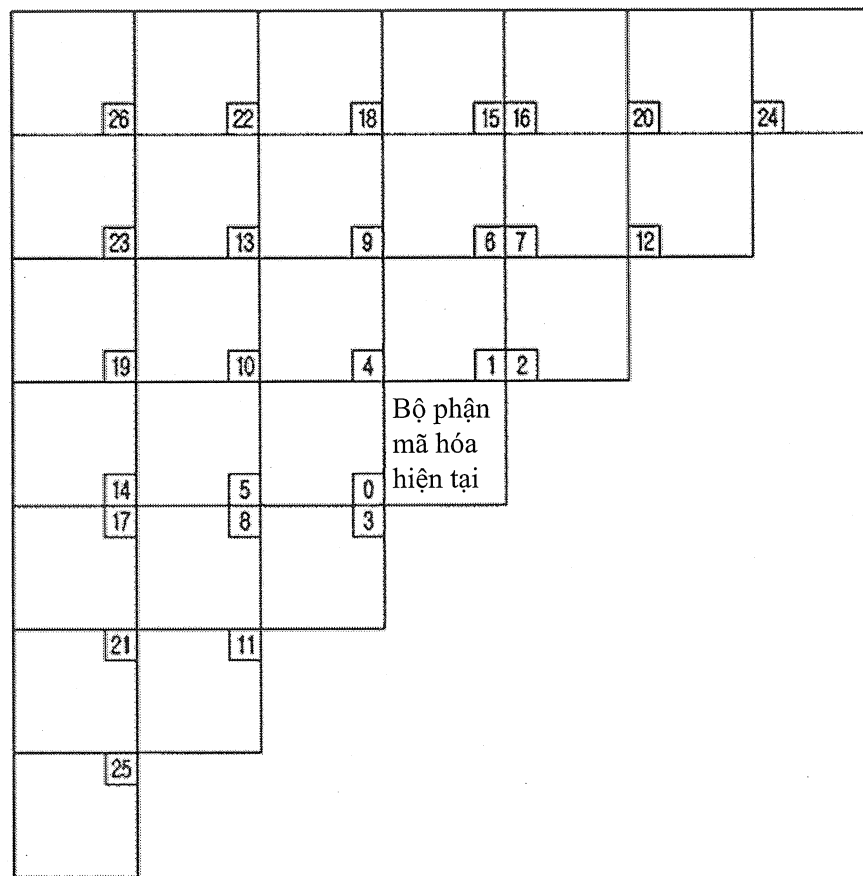


FIG. 12

**FIG. 13**

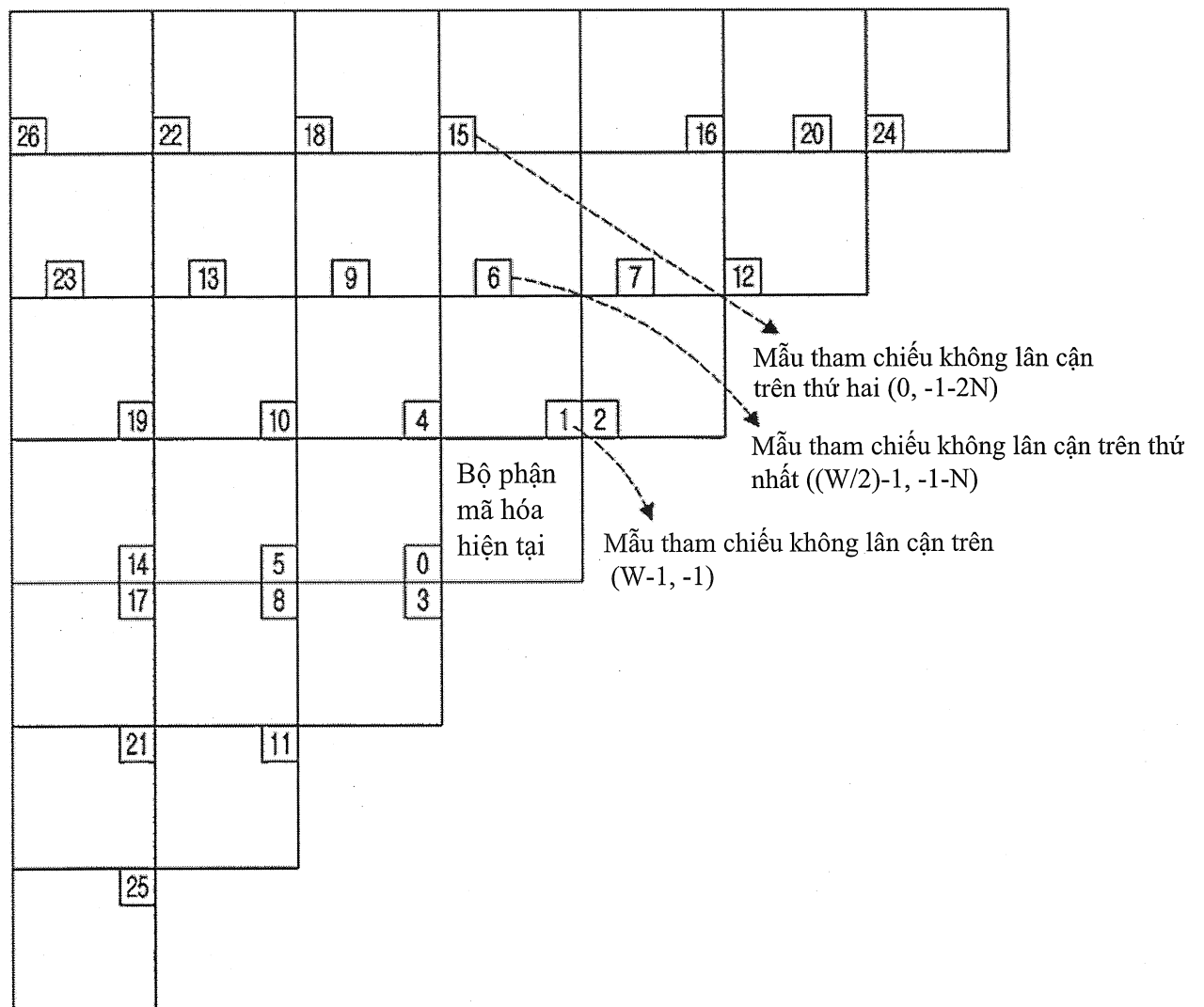


FIG. 14

	28		22		16		13		9		6	7		12
	29		23		18		10		4		1	2		
										Bộ phận mã hóa hiện tại				
	21		15		14		5		0					
	30		24		17		8		3					
	31		25		19		11							
	32		26		20									
	33		27											
	34													

FIG. 15

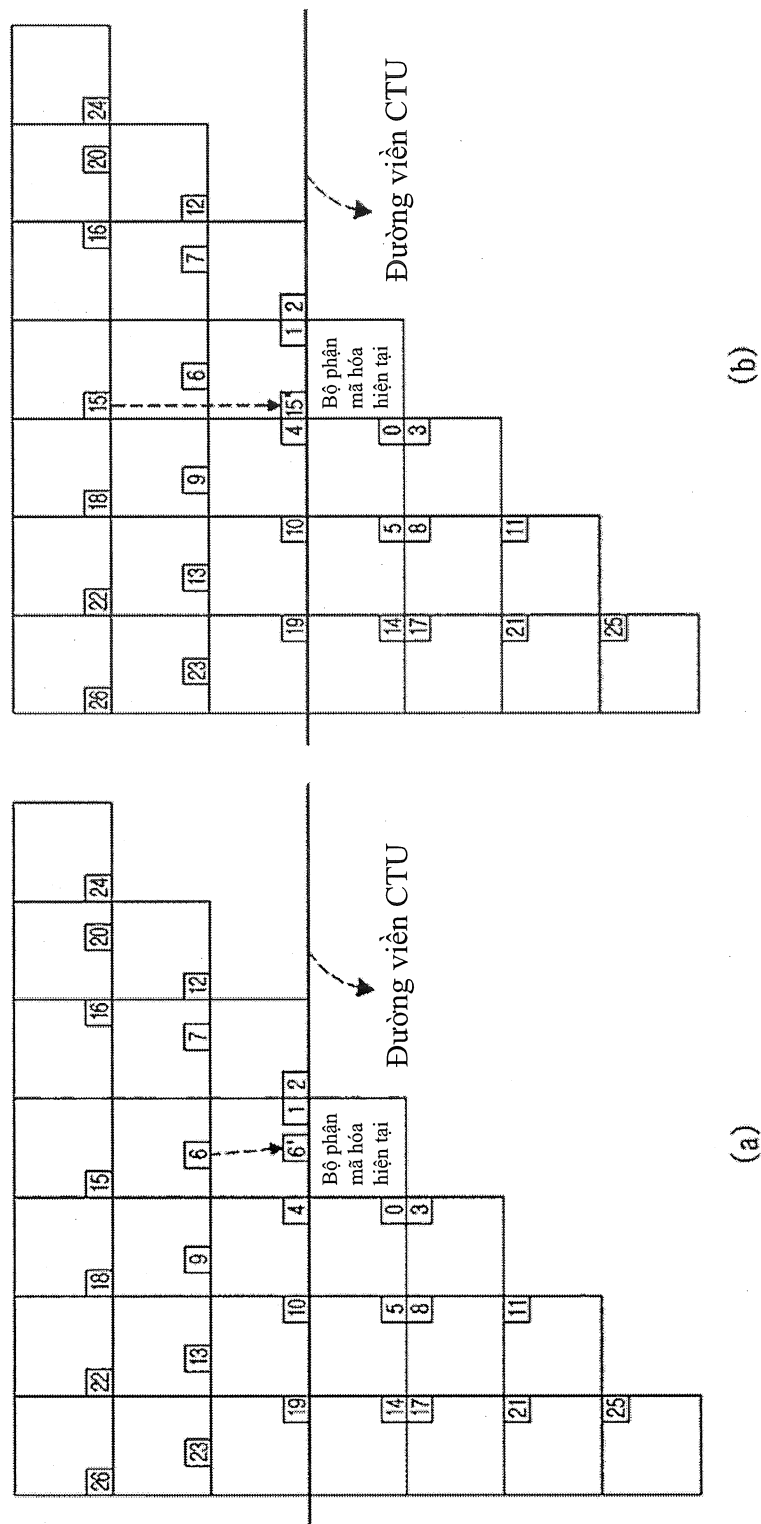


FIG. 16

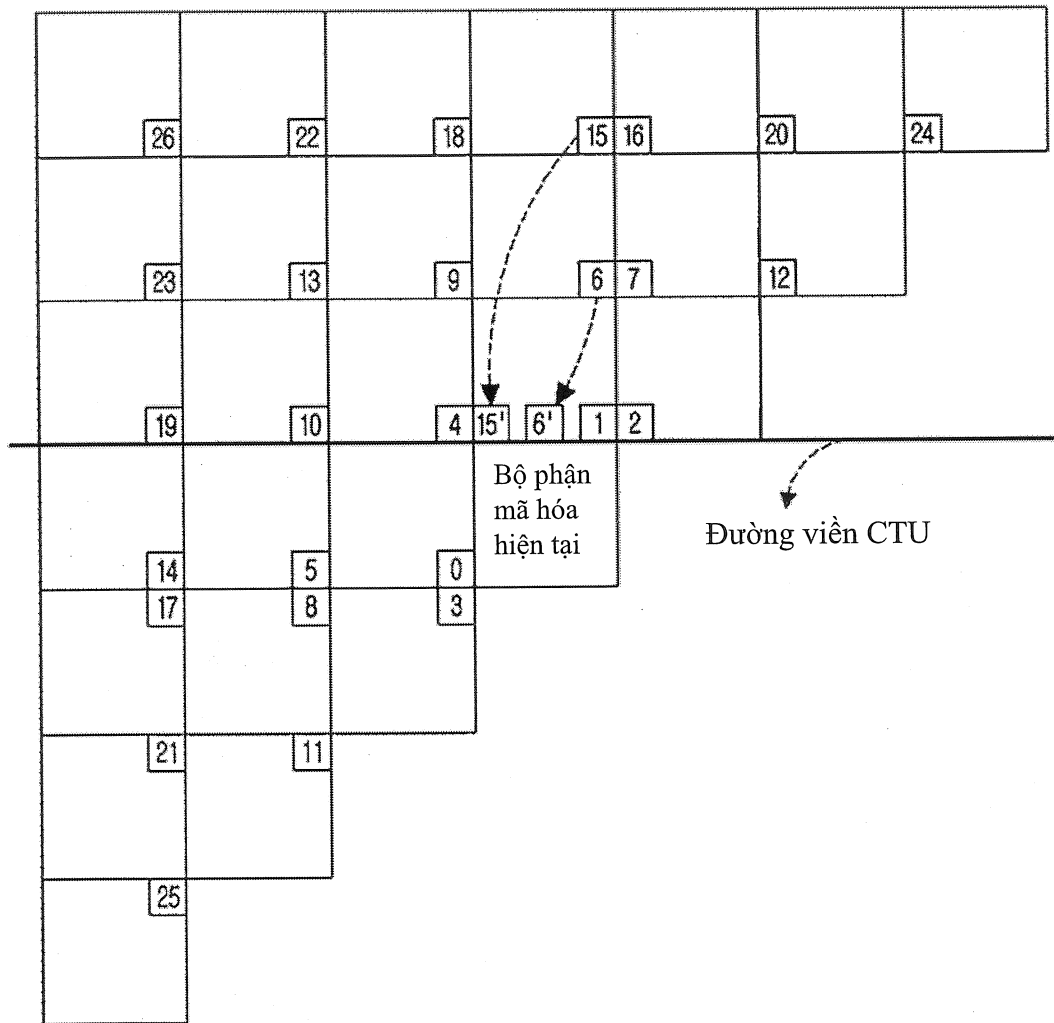


FIG. 17

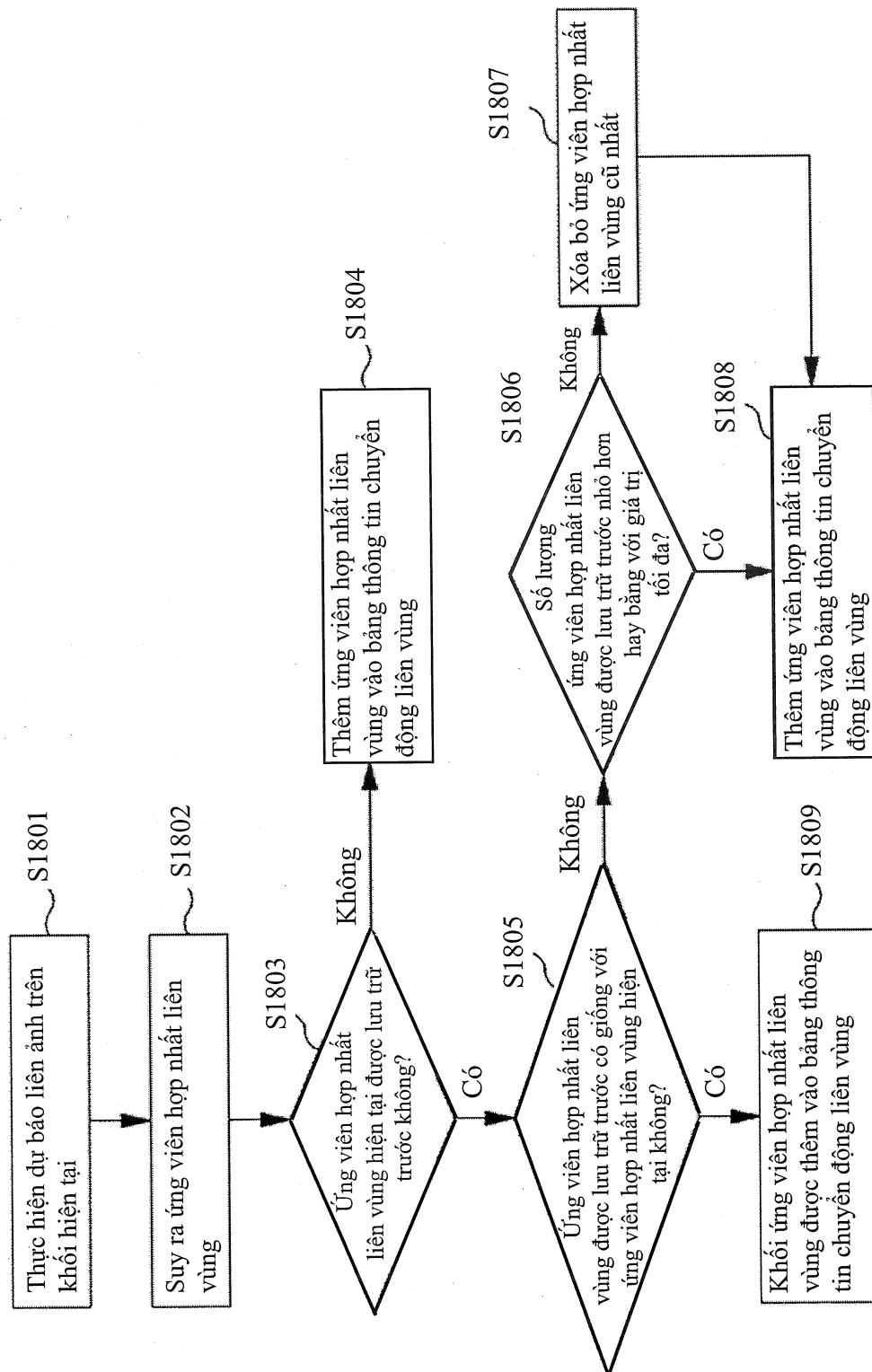


FIG. 18

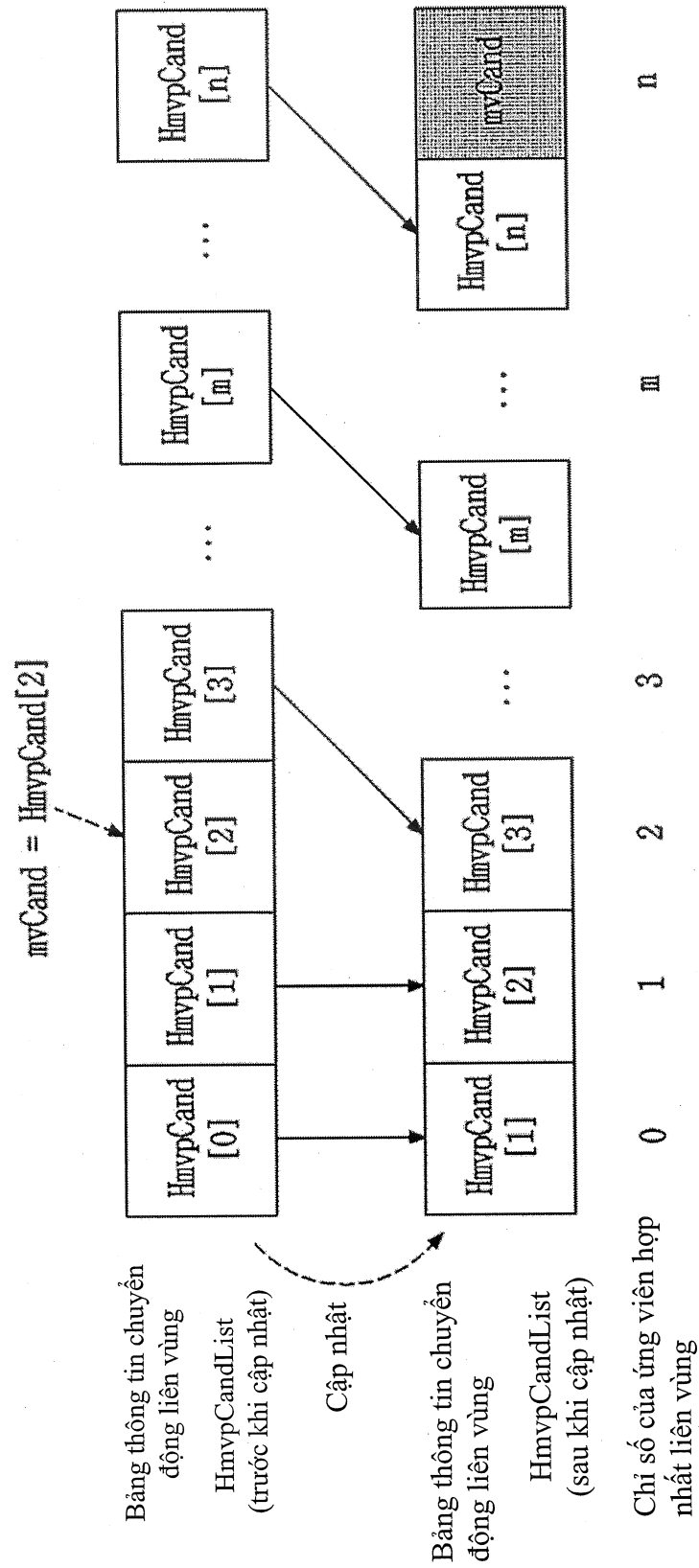
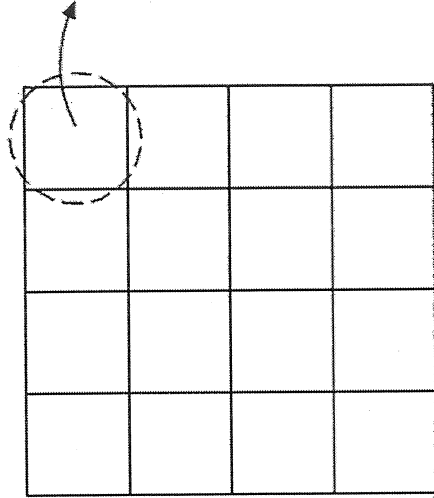


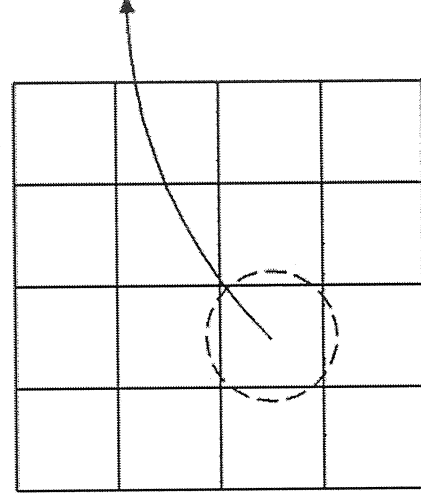
FIG. 20

Khối con đại diện



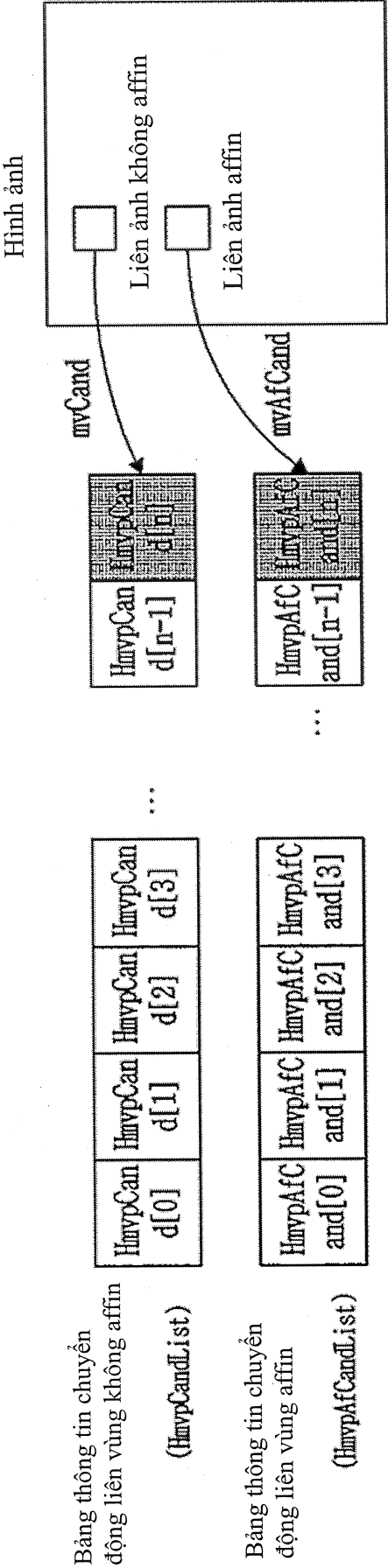
(a)

Khối con đại diện



(b)

FIG. 21



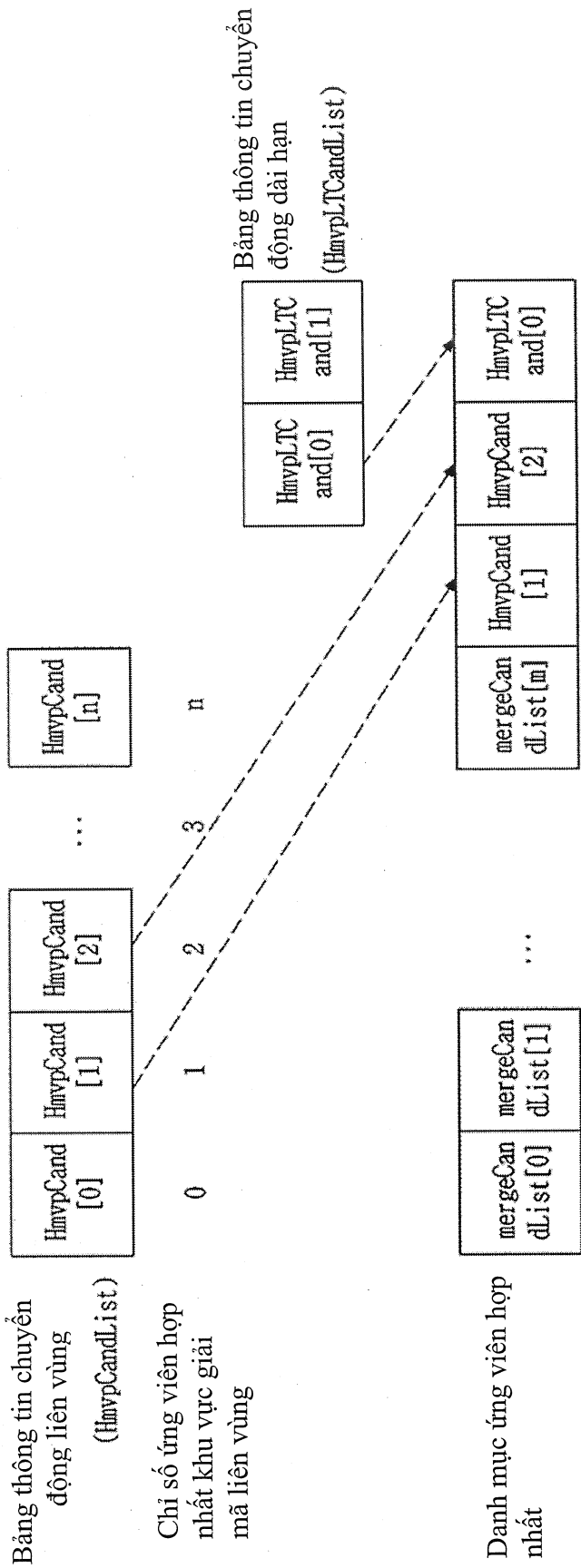


FIG. 23

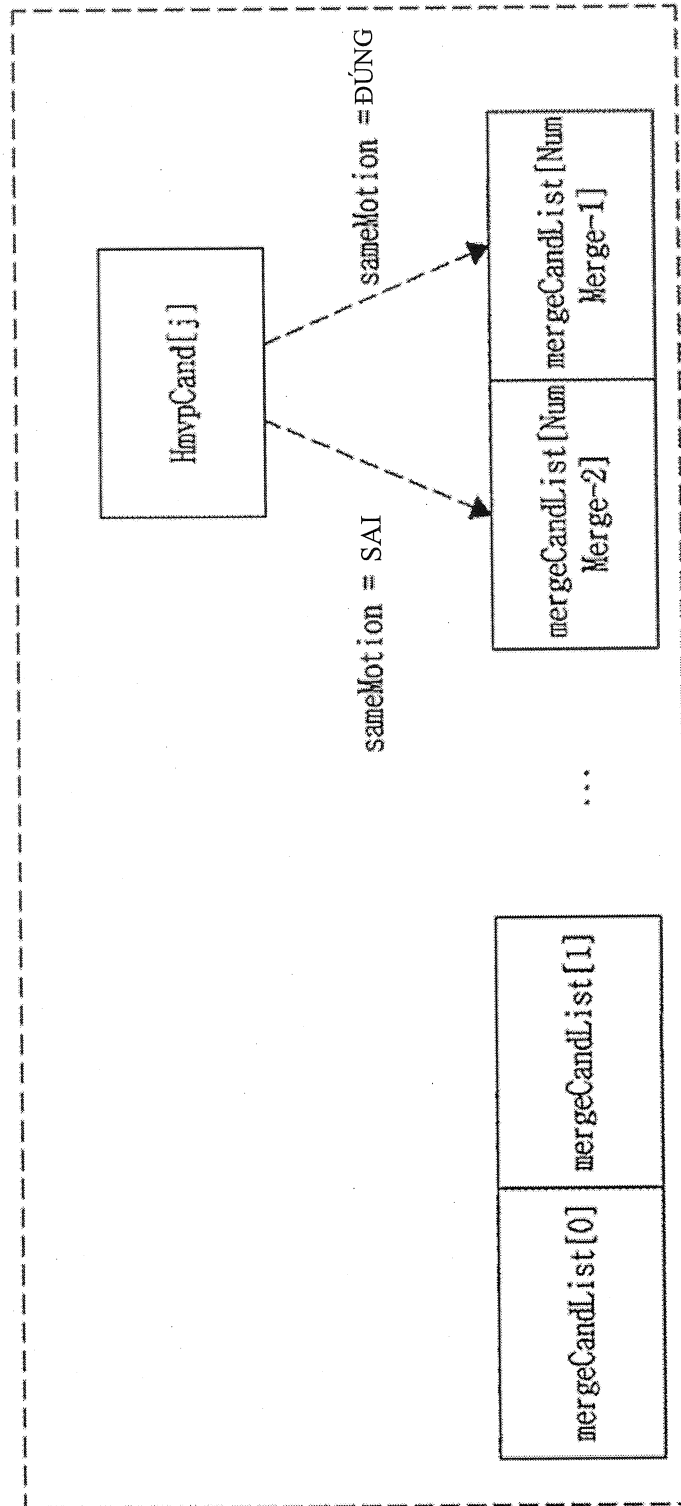


FIG. 24

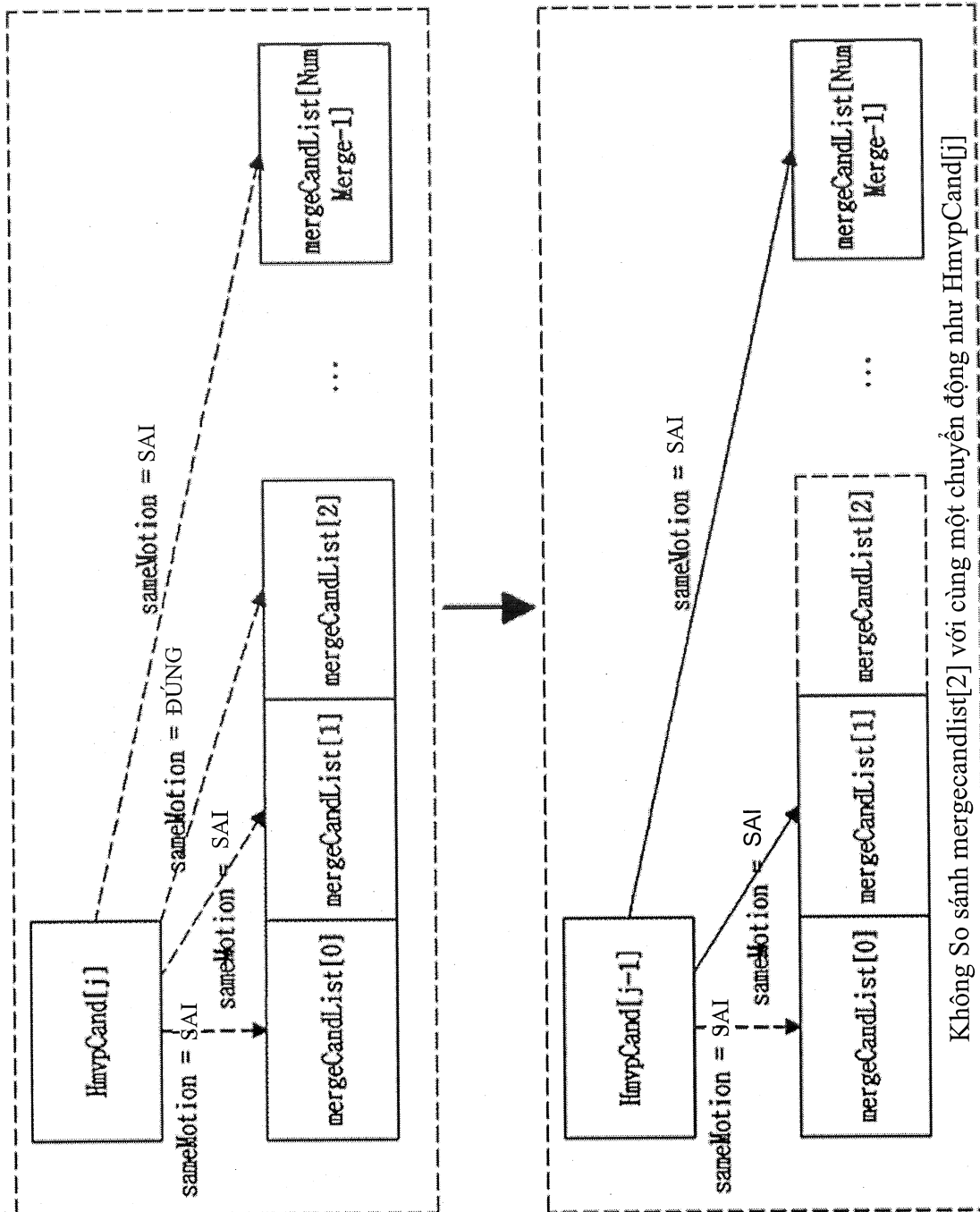


FIG. 25

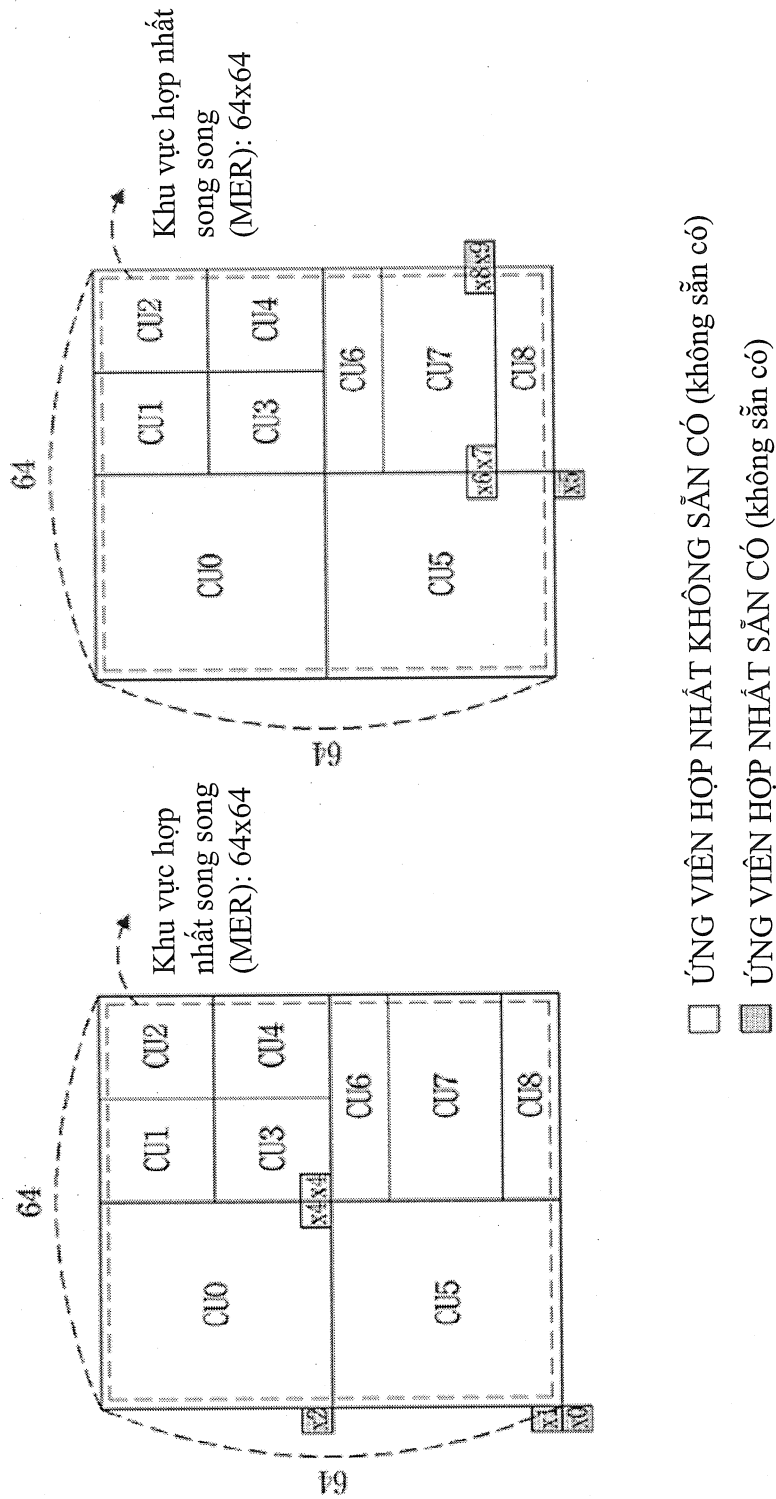


FIG. 26

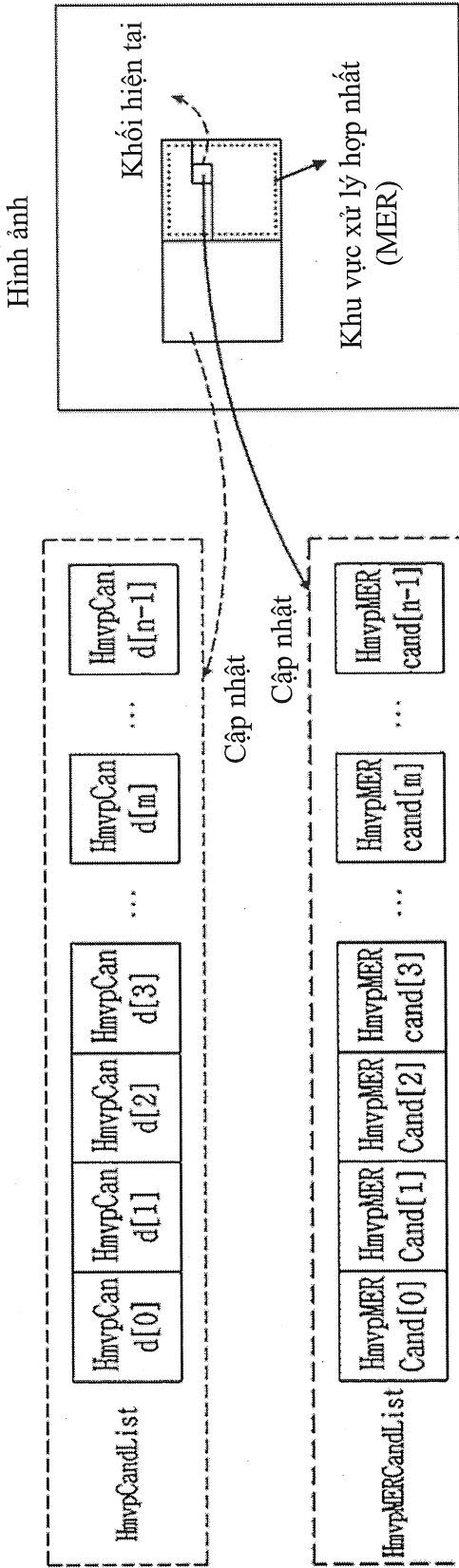


FIG. 27

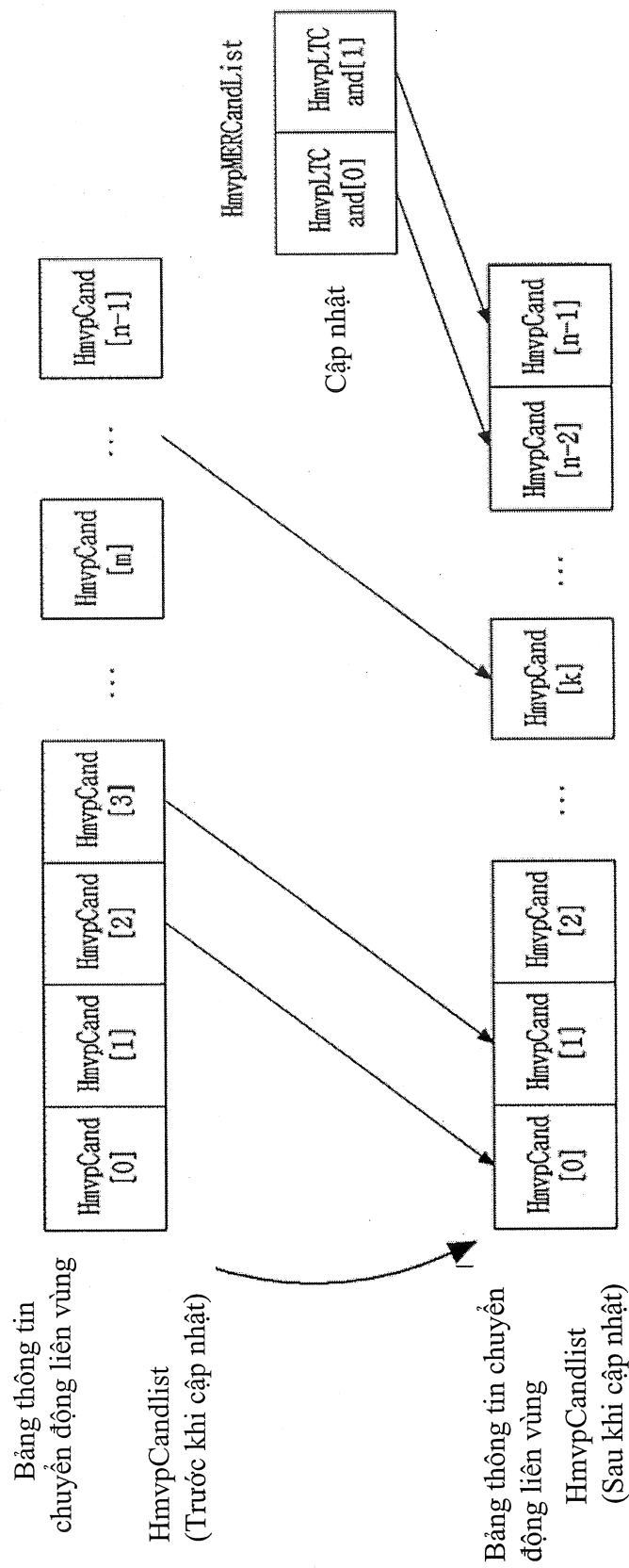
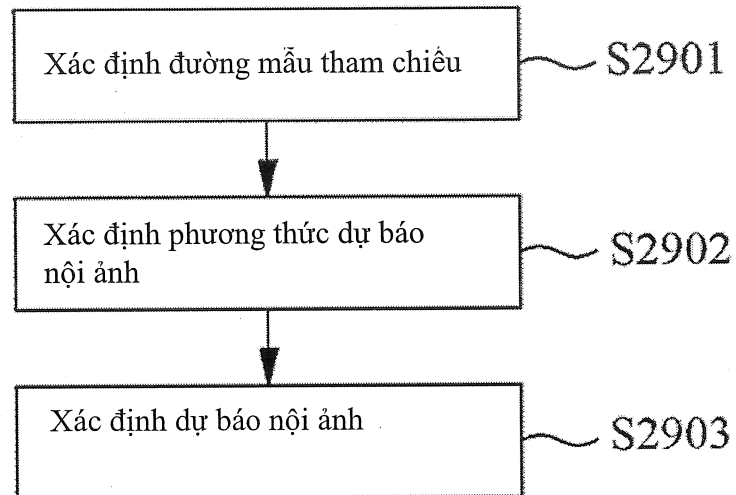
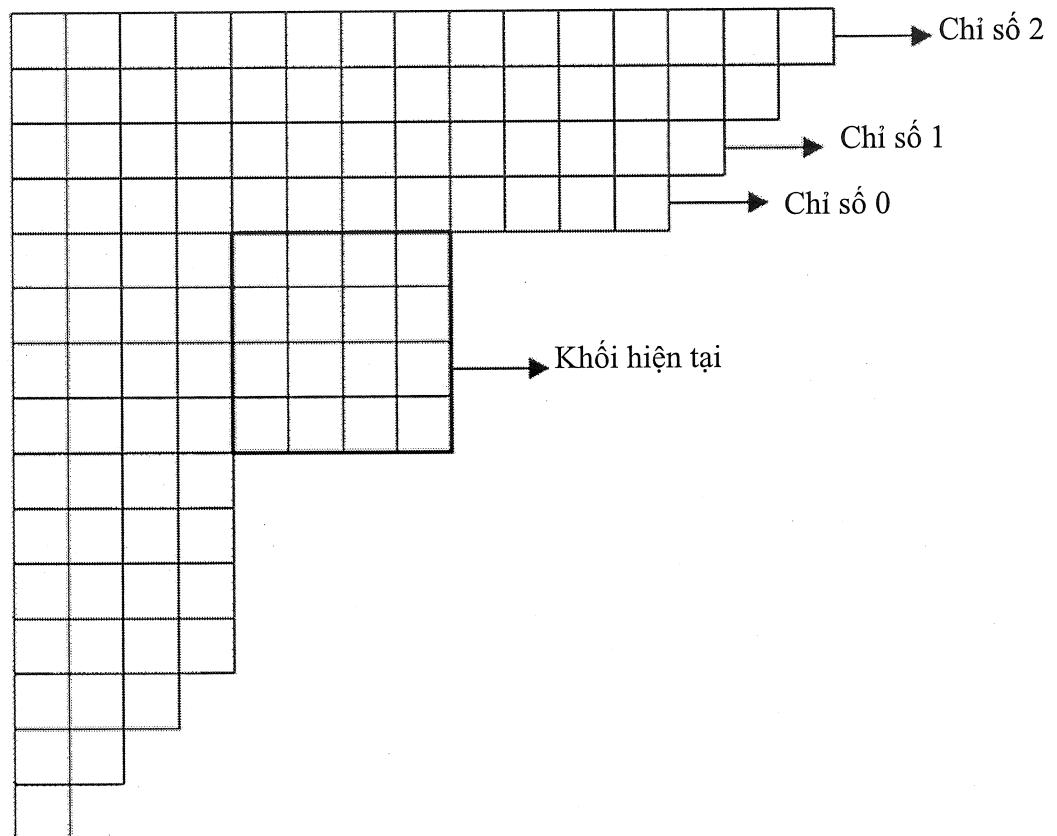


FIG. 28

**FIG. 29****FIG. 30**

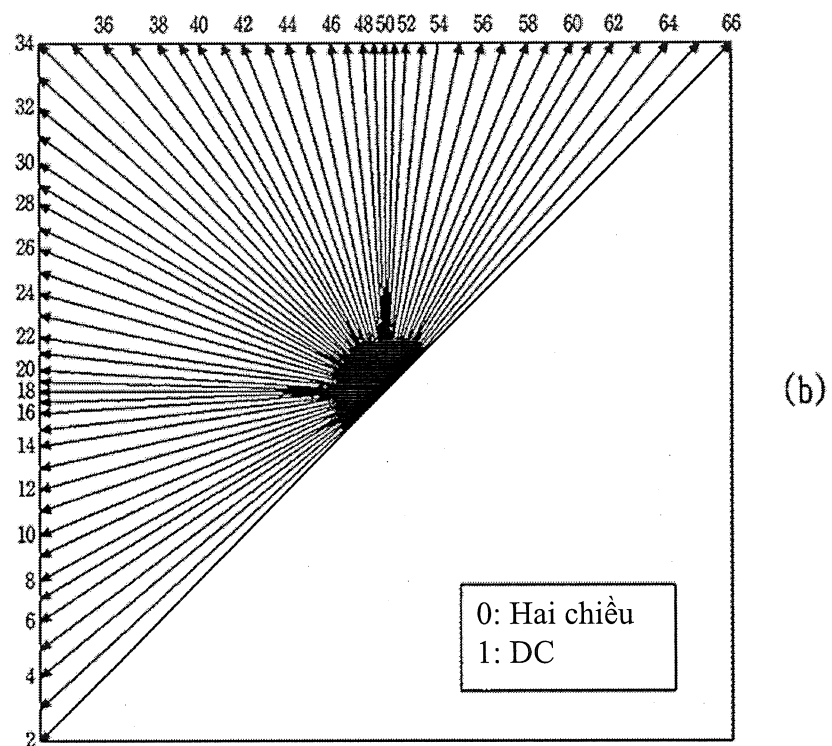
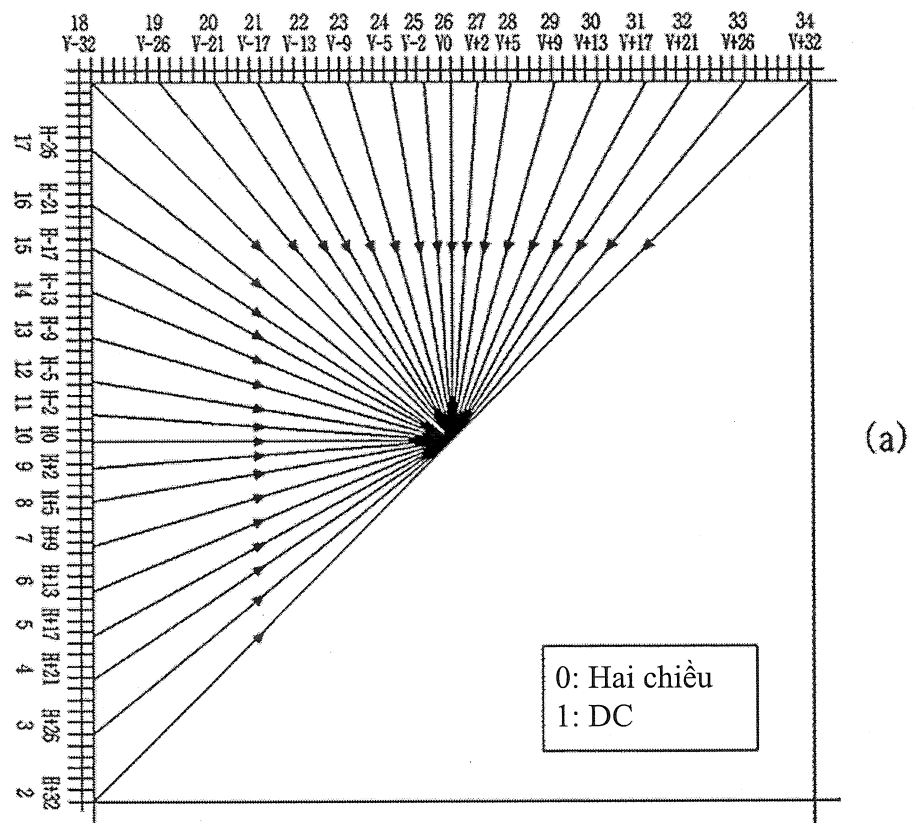


FIG. 31

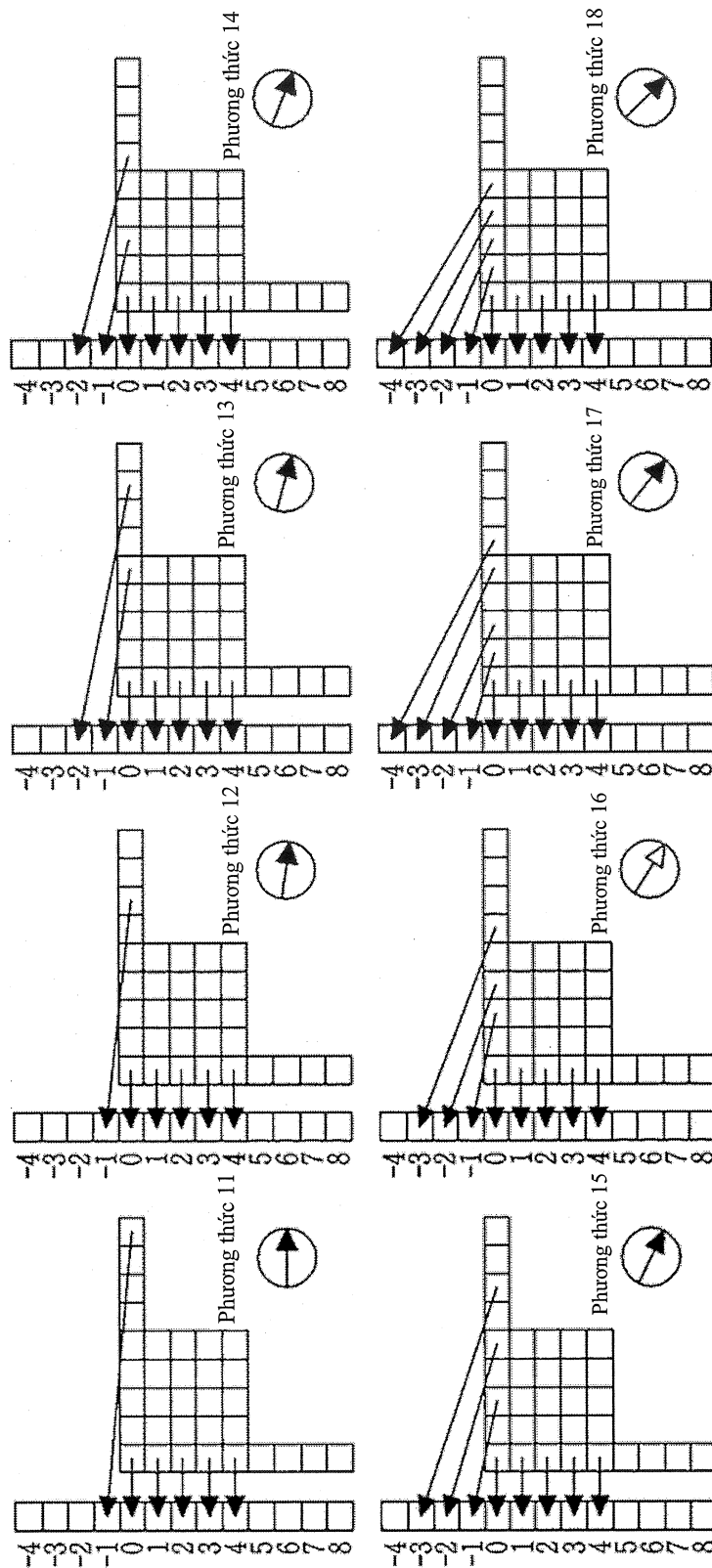


FIG. 32

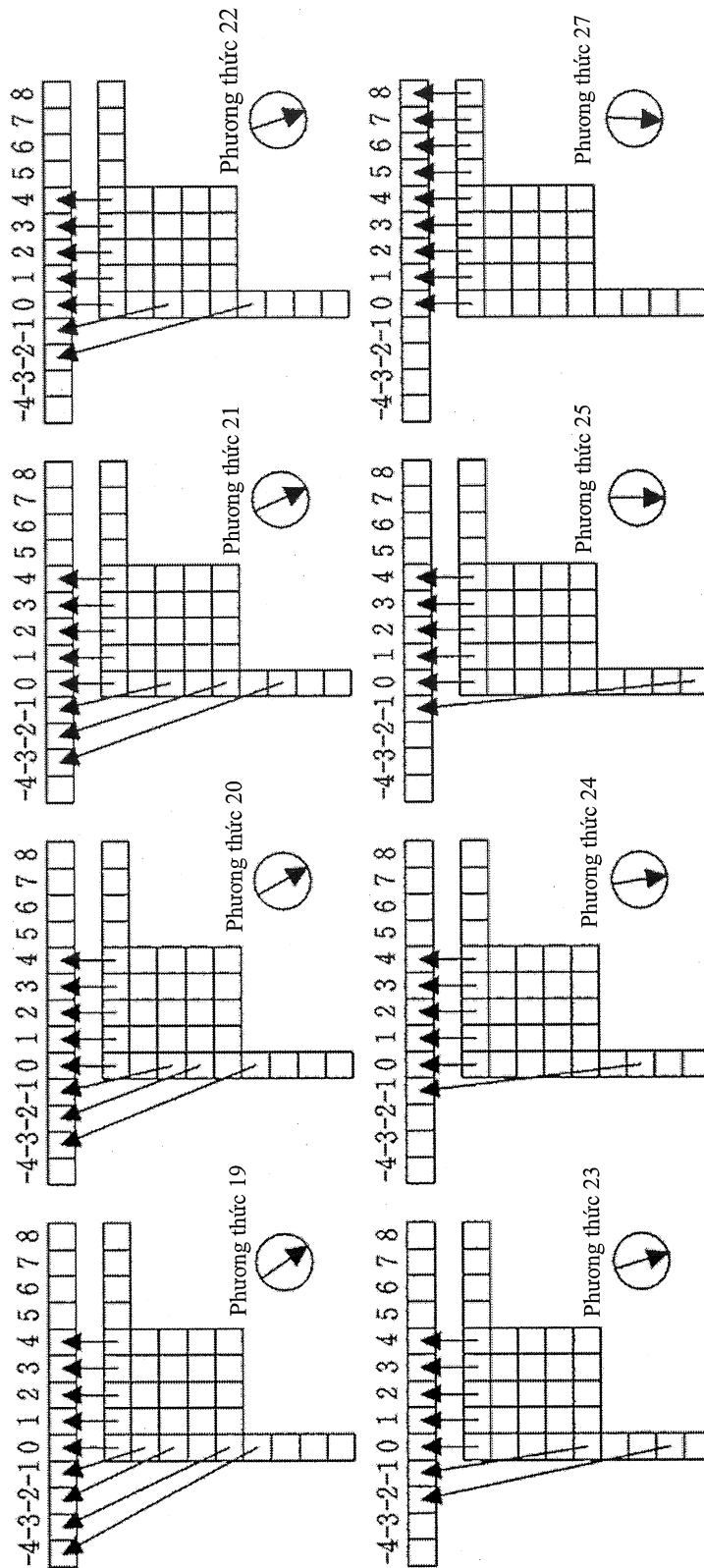


FIG. 33

Phương thức 34

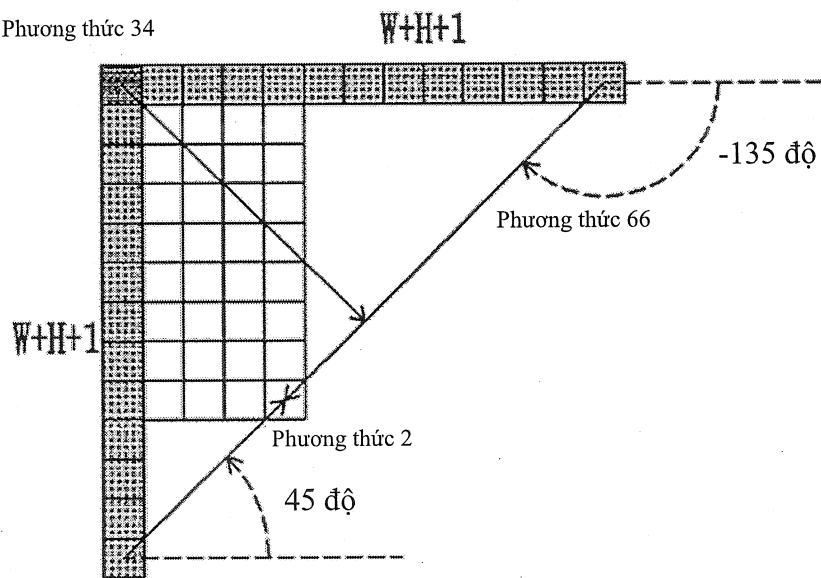


FIG. 34

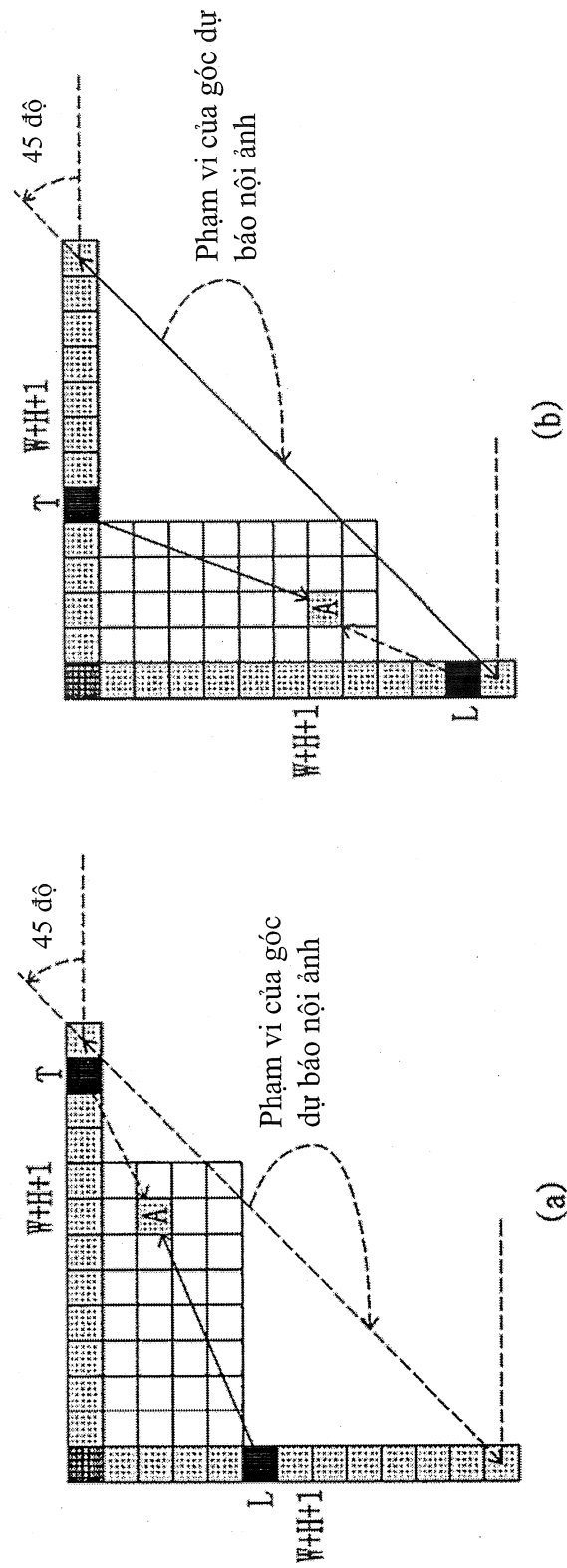


FIG. 35

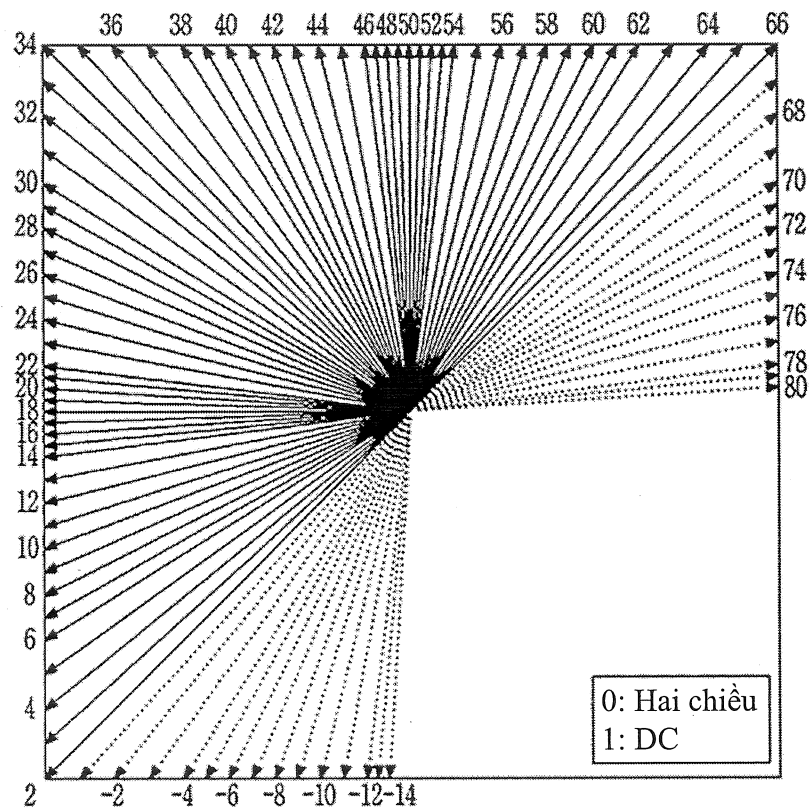


FIG. 36

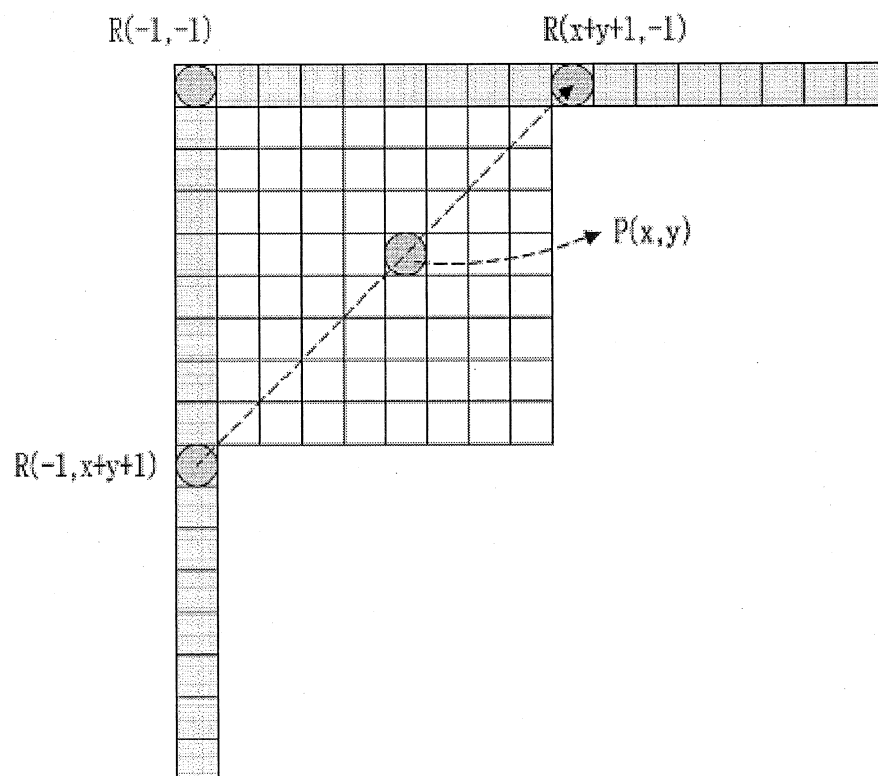


FIG. 37

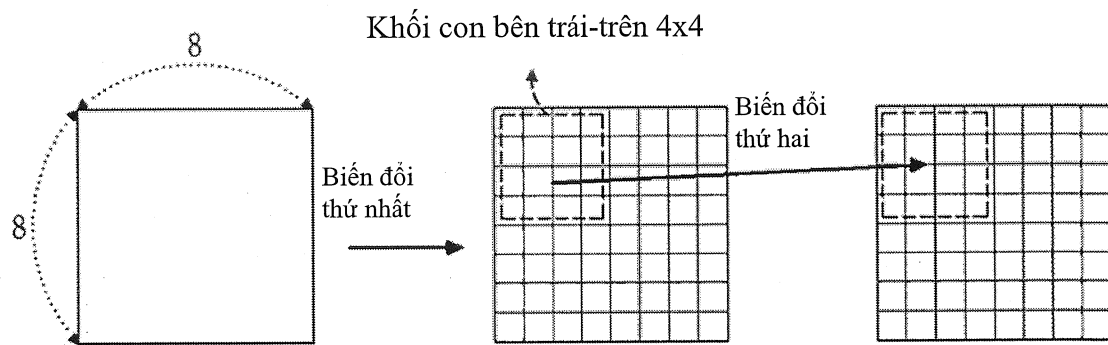


FIG. 38

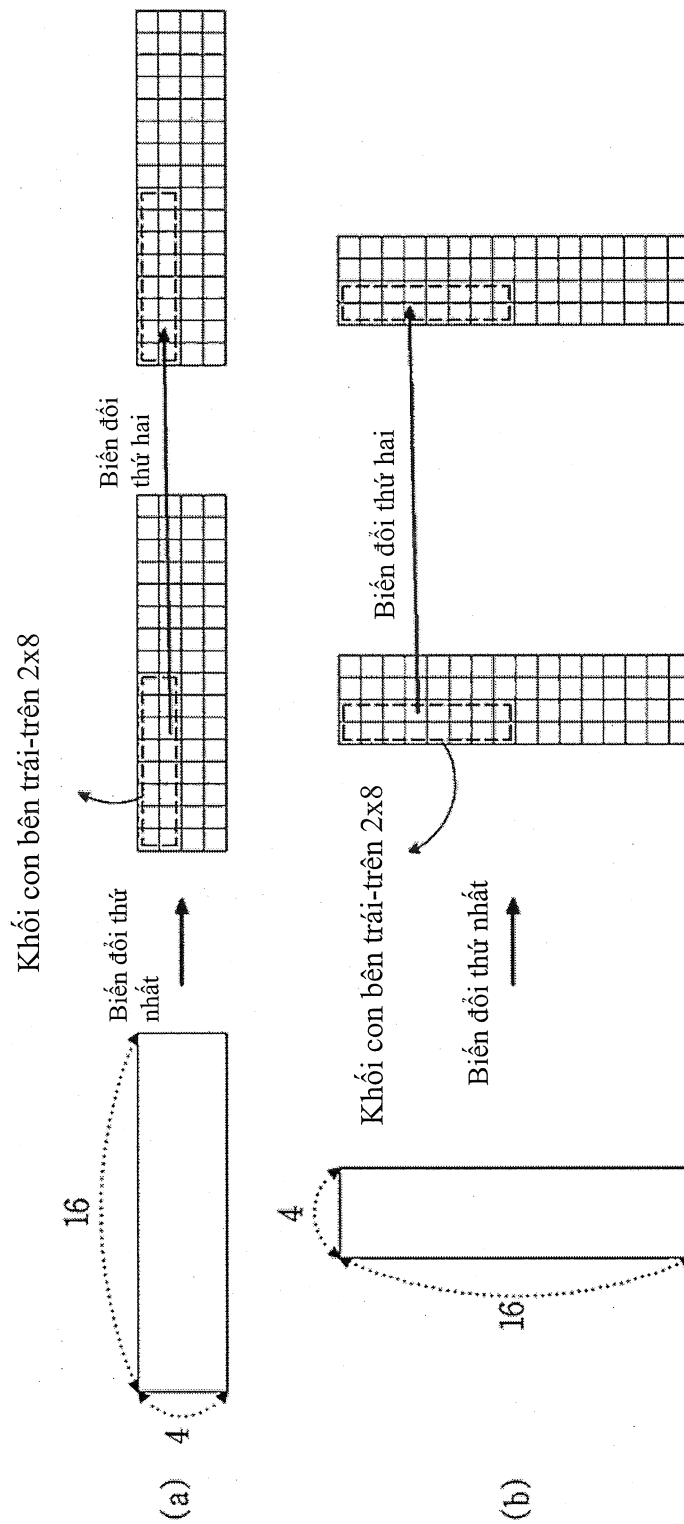


FIG. 39

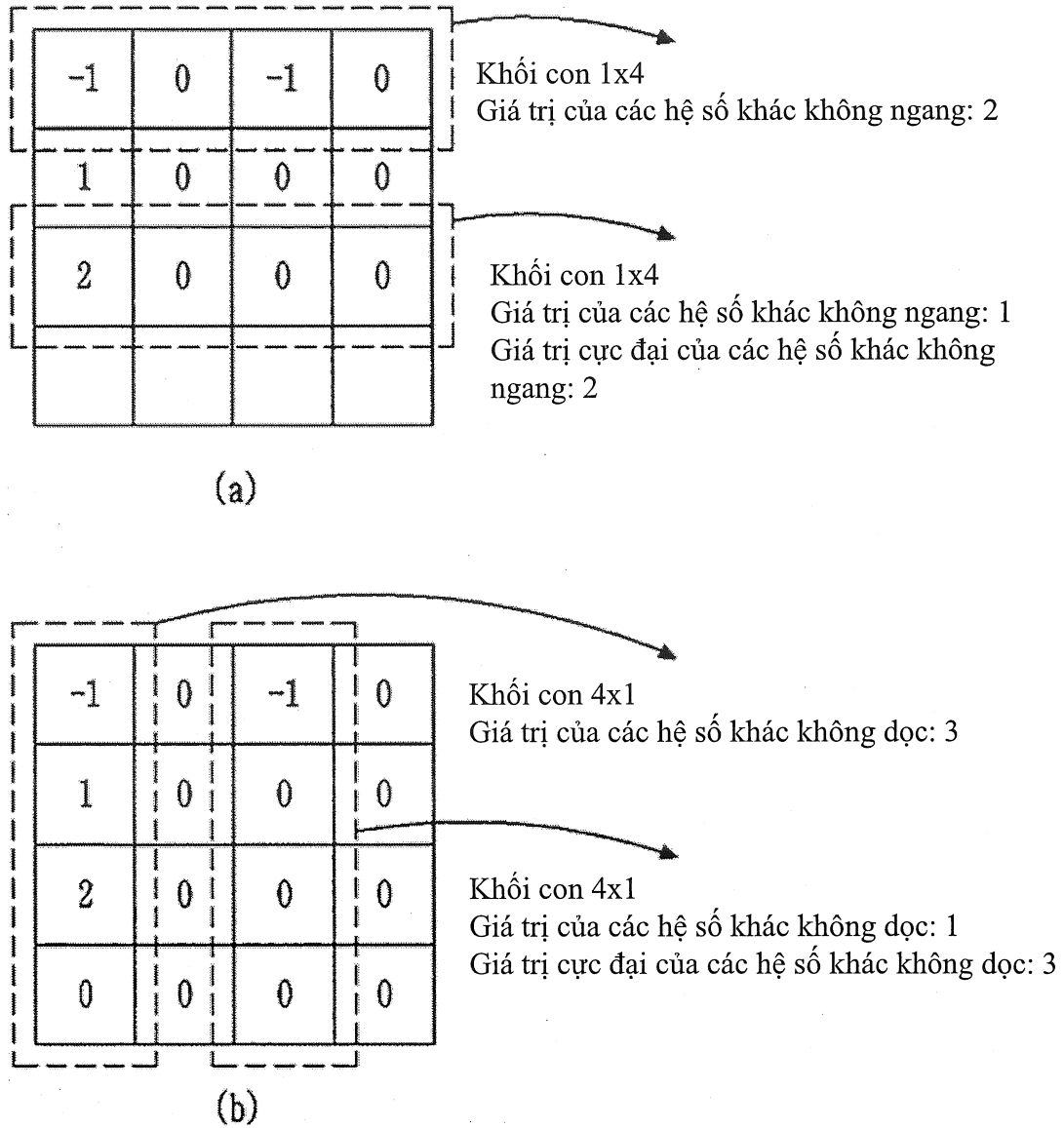


FIG. 40

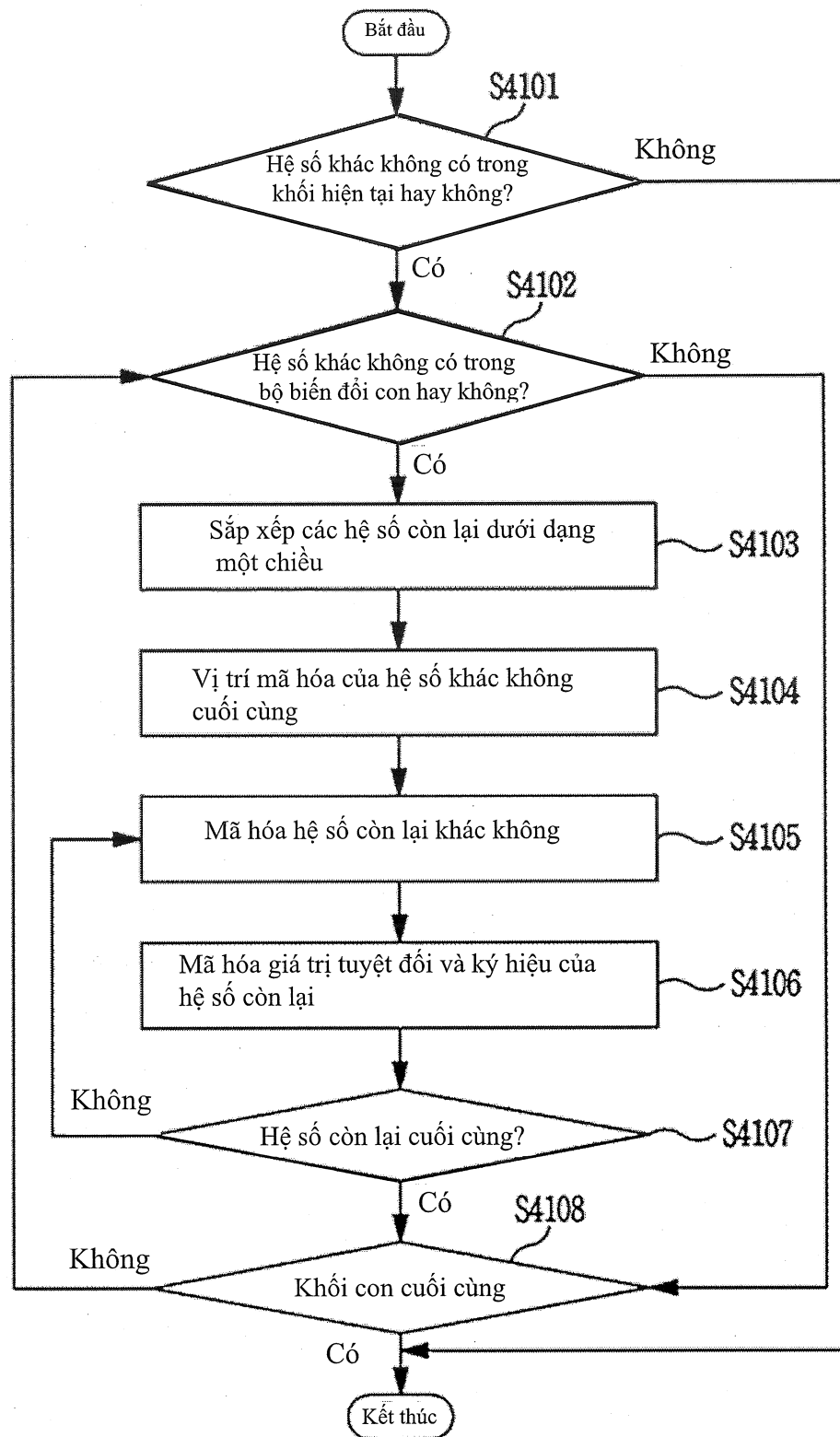


FIG. 41

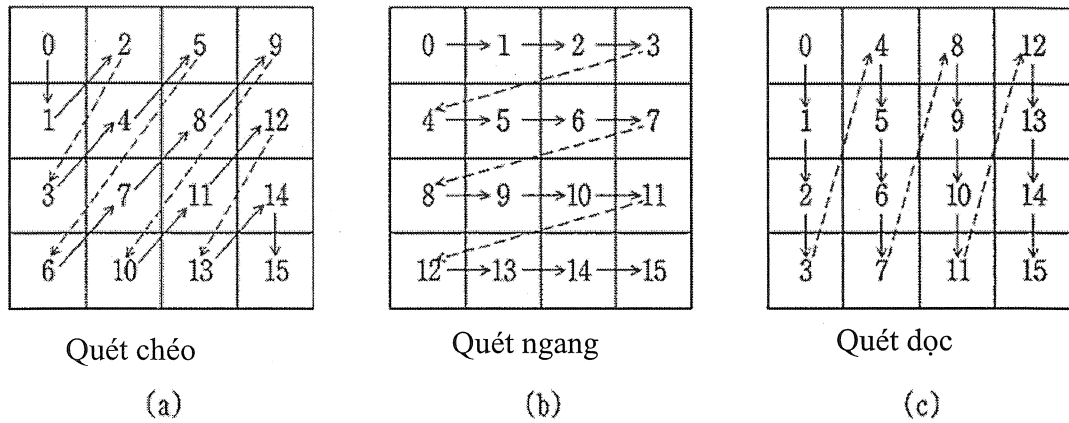


FIG. 42

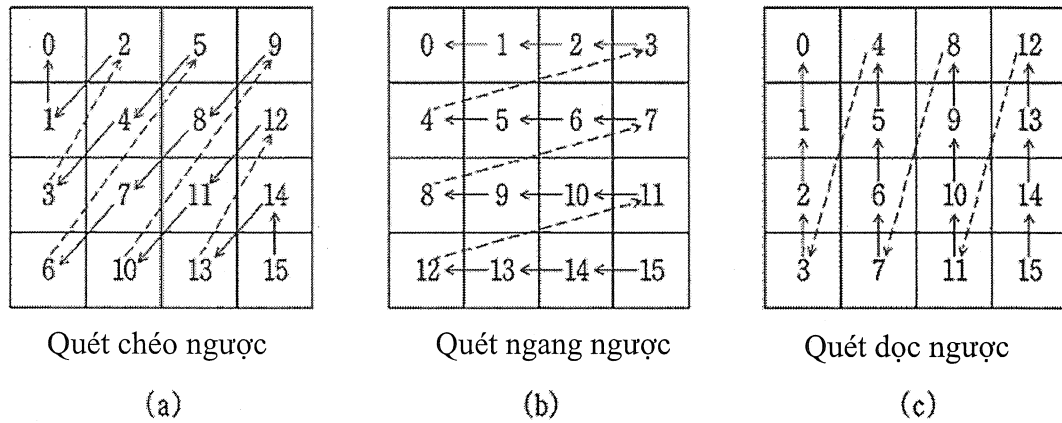


FIG. 43

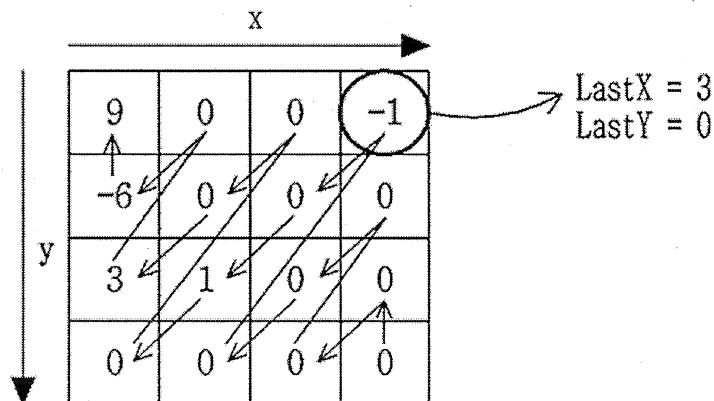


FIG. 44

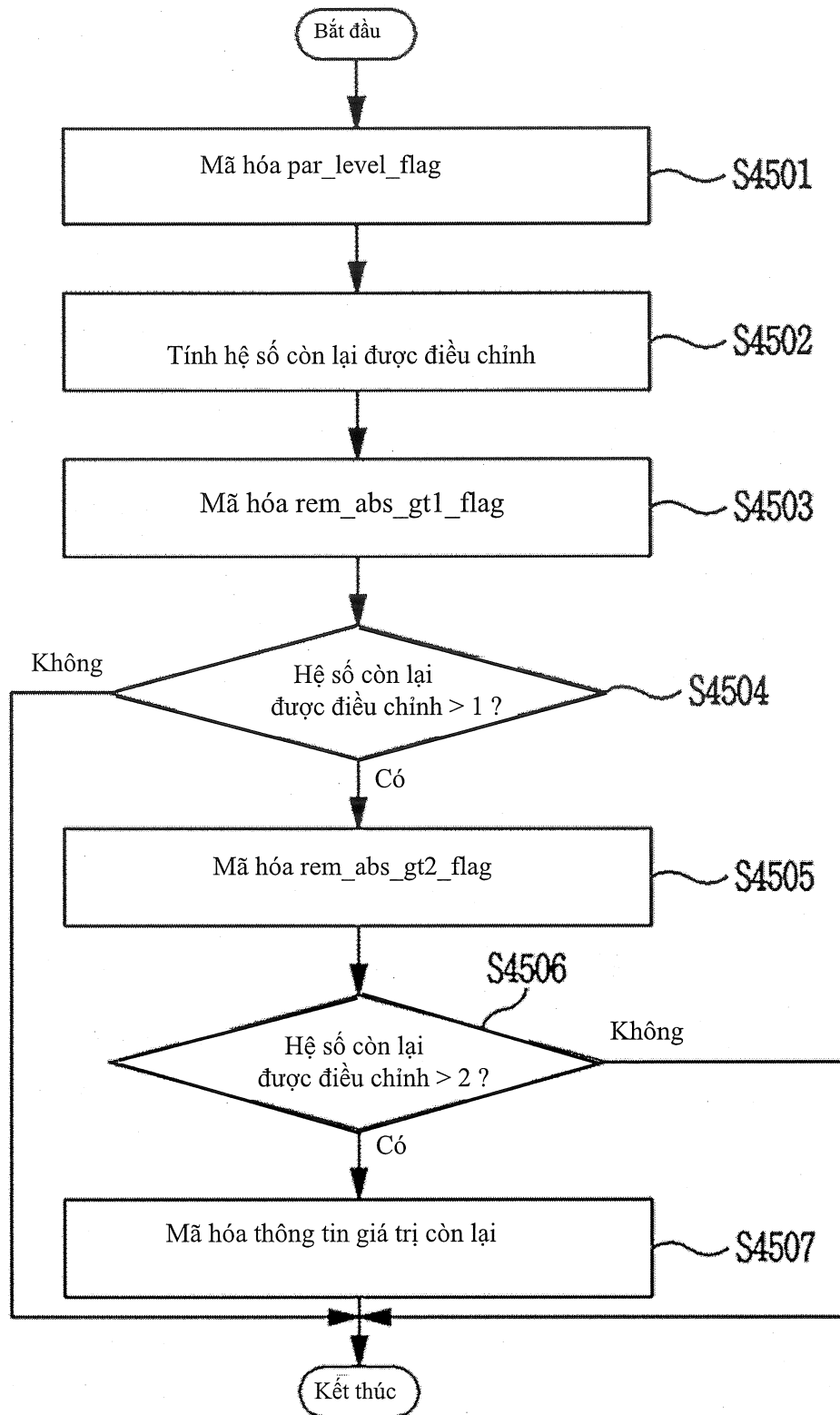


FIG. 45

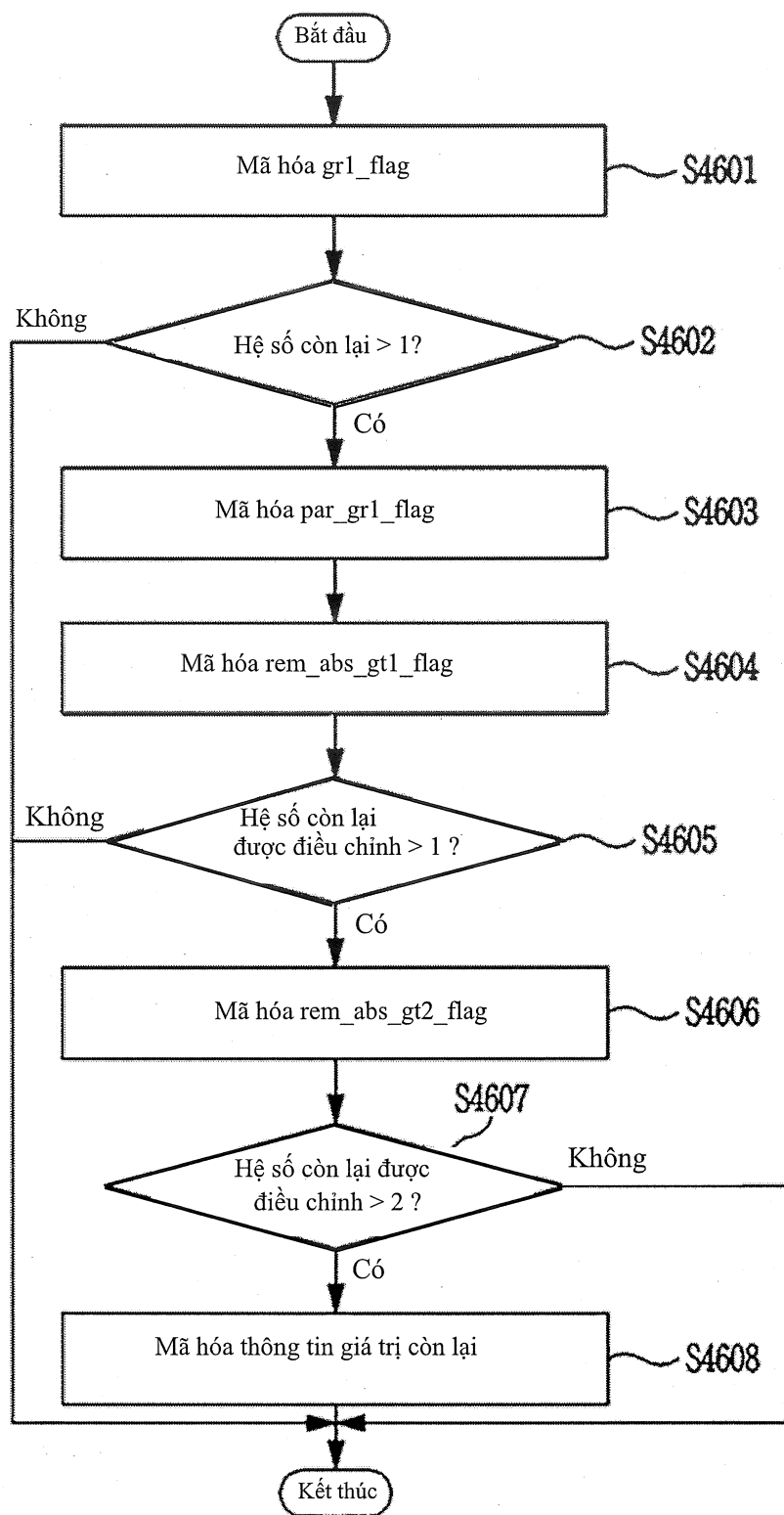


FIG. 46

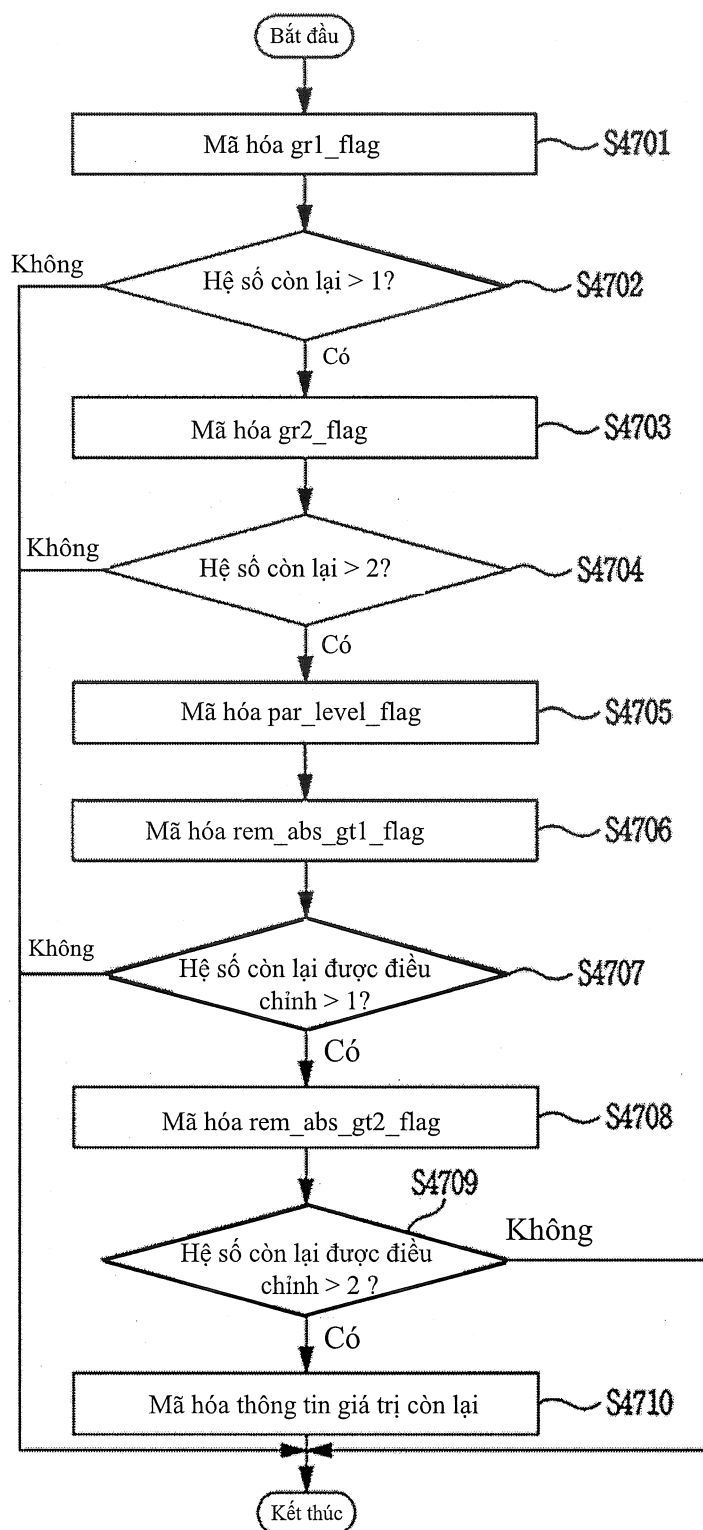


FIG. 47

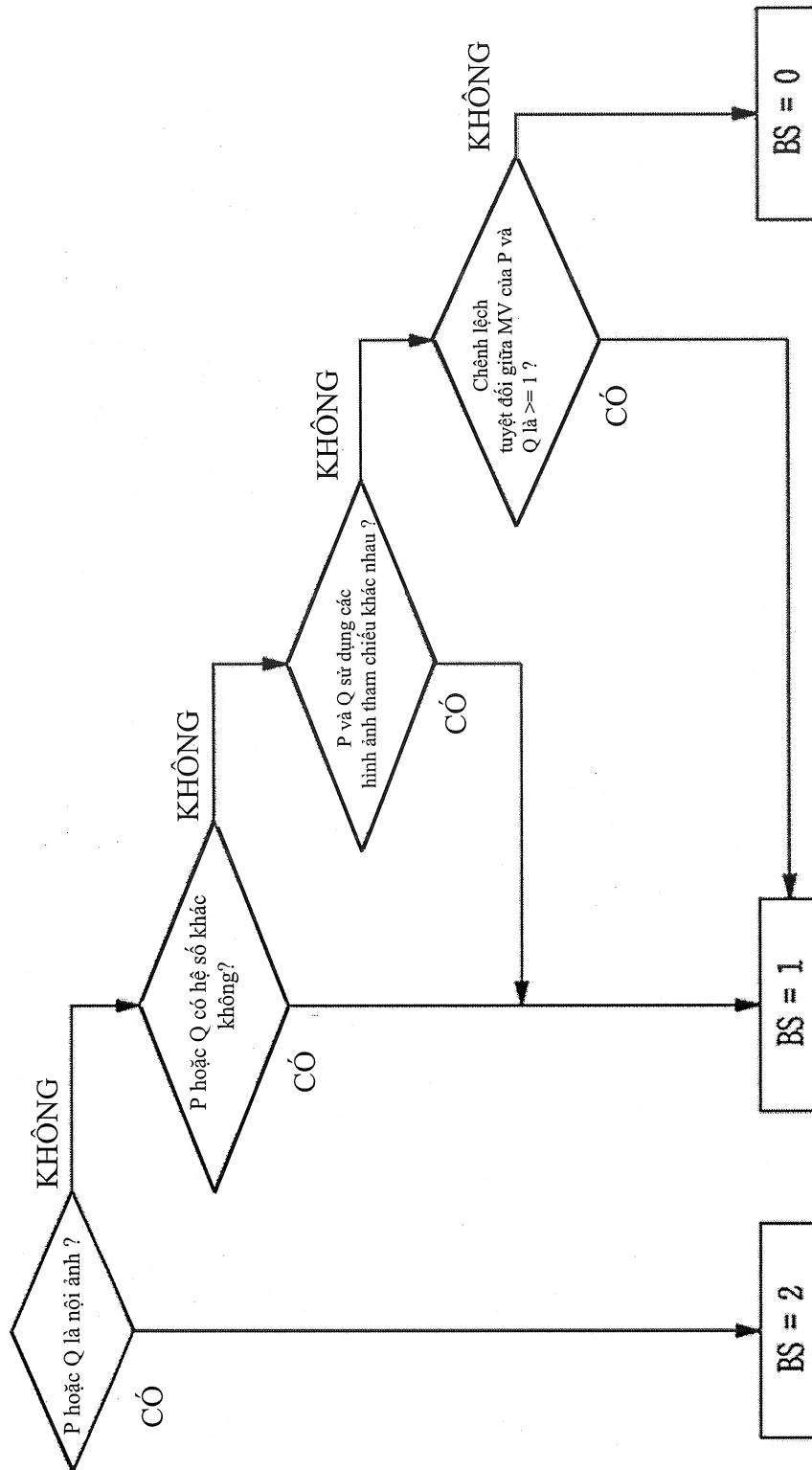


FIG. 48

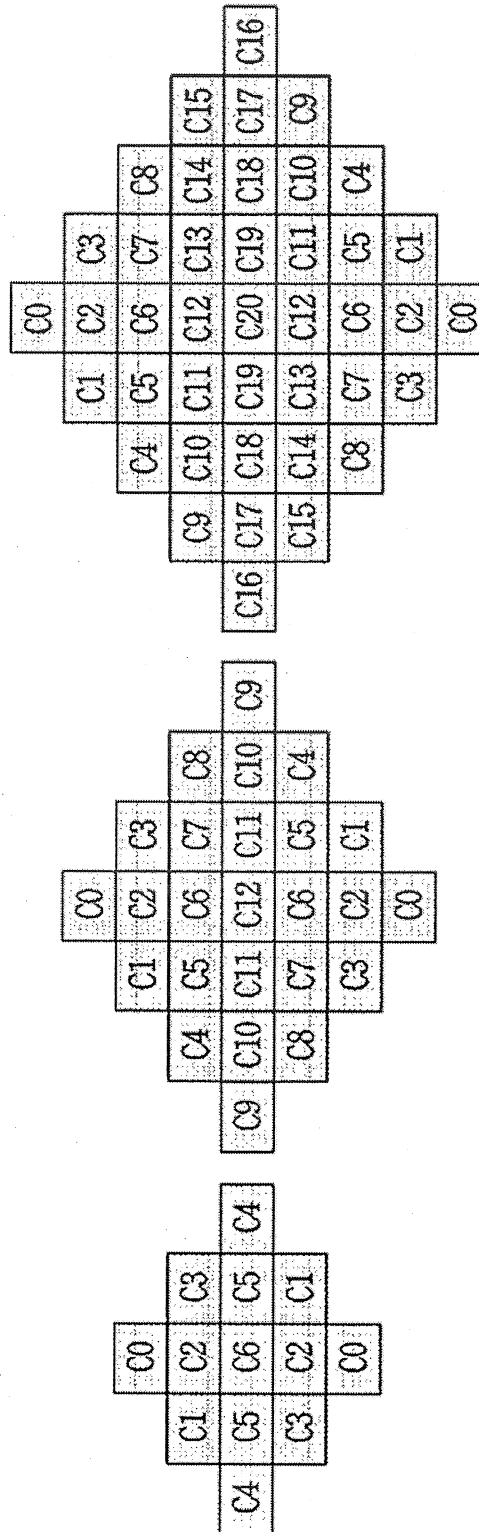


FIG. 49