



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ
1-0044869
(51)⁸ H04N 19/139; H04N 19/52; H04N 19/503 (13) B

(21) 1-2018-01726 (22) 03/03/2016
(86) PCT/KR2016/002161 03/03/2016 (87) WO 2017/052009 A1 30/03/2017
(30) 62/222,791 24/09/2015 US
(45) 25/04/2025 445 (43) 25/07/2018 364A
(71) LG ELECTRONICS INC. (KR)
128, Yeoui-daero, Yeongdeungpo-gu, Seoul 07336, Republic of Korea
(72) JANG, Hyeongmoon (KR); LIM, Jaehyun (KR); PARK, Seungwook (KR); YEA,
Sehoon (KR).
(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP MÃ HÓA ẢNH DỰA VÀO KHOẢNG VECTƠ CHUYỂN ĐỘNG
THÍCH ỨNG VÀ PHƯƠNG PHÁP DỰ BÁO GIỮA CÁC KHUNG

(21) 1-2018-01726

(57) Sáng chế đề cập đến phương pháp mã hóa ảnh được thực hiện bằng thiết bị mã hóa, bao gồm các bước: tạo cờ cho phép AMVR (Adaptive Motion Vector Range – khoảng vector chuyển động thích ứng); thu MVD (Motion Vector Difference – chênh lệch vector chuyển động) đối với khối hiện thời; thu giá trị đại diện MVD của khoảng MVD bao gồm giá trị của MVD trong số các khoảng MVD; tạo MVD được mã hóa tương ứng với giá trị đại diện MVD; và xuất cờ cho phép AMVR và MVD được mã hóa qua dòng bit. Theo sáng chế, lượng bit được cấp phát cho MVD có thể được giảm và do vậy hiệu suất mã hóa tổng có thể được tăng.

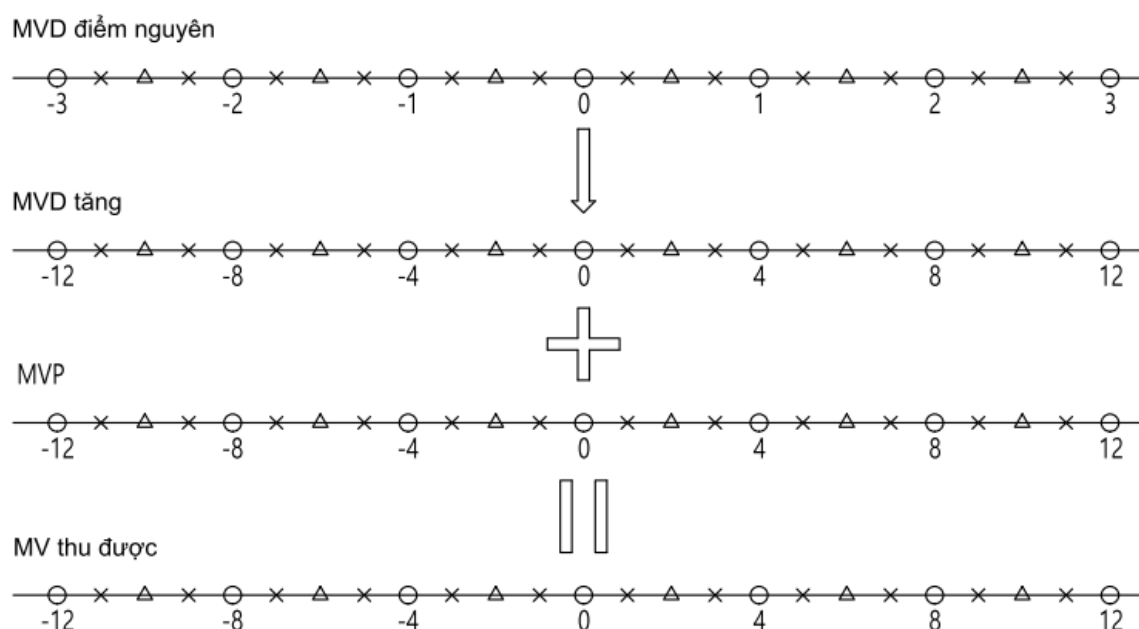


FIG.5

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến kỹ thuật mã hóa ảnh, và cụ thể hơn là, phương pháp và thiết bị mã hóa ảnh dựa vào AMVR (Adaptive Motion Vector Range – khoảng vector chuyển động thích ứng) trong hệ thống mã hóa ảnh.

Tình trạng kỹ thuật của sáng chế

Nhu cầu đối với các ảnh độ phân giải cao, chất lượng cao như các ảnh HD (High Definition - độ phân giải cao) và các ảnh UHD (Ultra High Definition - độ phân giải rất cao) đã tăng trong nhiều lĩnh vực. Do dữ liệu ảnh có độ phân giải cao và chất lượng cao, nên lượng thông tin hoặc bit được truyền sẽ tăng so với dữ liệu ảnh kế thừa. Do đó, khi dữ liệu ảnh được truyền bằng cách sử dụng phương tiện như đường truyền băng rộng hữu tuyến/vô tuyến hoặc dữ liệu ảnh được lưu trữ bằng cách sử dụng phương tiện lưu trữ hiện thời, chi phí truyền và lưu trữ sẽ tăng.

Do vậy, cần có kỹ thuật nén ảnh hiệu suất cao để truyền, lưu trữ, và tái tạo thông tin có độ phân giải cao và các ảnh chất lượng cao một cách hiệu quả.

Bản chất kỹ thuật của sáng chế

Sáng chế đề xuất phương pháp và thiết bị tăng hiệu suất mã hóa video.

Sáng chế cũng đề xuất phương pháp và thiết bị tăng hiệu suất dự báo giữa các khung.

Sáng chế cũng đề xuất phương pháp giảm các bit được cấp phát cho MVD (Motion Vector Difference – chênh lệch vector chuyển động) dựa vào AMVR.

Sáng chế cũng đề xuất phương pháp truyền/thu MVD hiệu quả dựa vào khoảng MVD tuyến tính/không tuyến tính.

Theo một khía cạnh, phương pháp mã hóa ảnh được thực hiện bằng thiết

bị mã hóa bao gồm các bước: tạo cờ cho phép AMVR, thu MVD dùng cho khối hiện thời, thu giá trị đại diện MVD dùng cho khoảng MVD bao gồm giá trị của MVD trong số các khoảng MVD, tạo MVD được mã hóa tương ứng với giá trị đại diện MVD, và xuất cờ cho phép AMVR và MVD được mã hóa qua dòng bit.

Theo khía cạnh khác, phương pháp dự báo giữa các khung được thực hiện bằng thiết bị giải mã bao gồm các bước: thu cờ cho phép AMVR từ dòng bit, thu MVD được mã hóa từ dòng bit, thu giá trị của MVD đại diện tương ứng với giá trị của MVD được mã hóa khi giá trị của cờ cho phép AMVR là 1, thu MVP (Motion Vector Predictor - bộ dự báo vector chuyển động) dùng cho khối hiện thời dựa vào khối lân cận của khối hiện thời, thu MV (Motion Vector - vector chuyển động) dùng cho khối hiện thời dựa vào MVP và MVD đại diện, và tạo mẫu dự báo dùng cho khối hiện thời dựa vào MV.

Theo khía cạnh khác nữa, thiết bị giải mã để thực hiện dự báo giữa các khung được đề xuất. Thiết bị giải mã bao gồm môđun giải mã mà thu cờ cho phép AMVR từ dòng bit và thu MVD được mã hóa từ dòng bit, và môđun dự báo mà thu giá trị của MVD đại diện tương ứng với giá trị của MVD được mã hóa khi giá trị của cờ cho phép AMVR là 1, thu MVP dùng cho khối hiện thời dựa vào khối lân cận của khối hiện thời, thu MV dùng cho khối hiện thời dựa vào MVP và MVD đại diện, và tạo mẫu dự báo dùng cho khối hiện thời dựa vào MV.

Theo một phương án của sáng chế, sự dự báo giữa các khung của khối hiện thời có thể được thực hiện một cách hiệu quả trong khi sử dụng thông tin bổ sung rất ít. Ngoài ra, theo một phương án của sáng chế, lượng bit được cấp phát cho MVD giảm, và hiệu suất mã hóa tổng có thể được tăng.

Mô tả vắn tắt các hình vẽ

FIG.1 là sơ đồ khối minh họa giản lược thiết bị mã hóa video theo một phương án của sáng chế.

FIG.2 là sơ đồ khối minh họa giản lược thiết bị giải mã video theo một

phương án của sáng chế.

FIG.3 là hình vẽ giản lược minh họa một phương án của khối ứng viên mà có thể sử dụng được khi dự báo giữa các khung được thực hiện trên khối hiện thời.

FIG.4 là hình vẽ minh họa các vị trí của các mẫu nguyên và phân số dùng cho nội suy 1/4 mẫu phân số trong dự báo giữa các khung.

FIG.5 là hình vẽ minh họa ví dụ về phương pháp thu MV dựa vào AMVR.

FIG.6 là hình vẽ minh họa phương pháp AVMR có khoảng tuyến tính.

FIG.7 là hình vẽ minh họa phương pháp AVMR có khoảng tuyến tính cân bằng.

FIG.8 là hình vẽ minh họa ví dụ về phương pháp AVMR có khoảng không tuyến tính.

FIG.9 là hình vẽ minh họa ví dụ về việc tạo AMVR không tuyến tính bằng cách phân tách cú pháp tham số khoảng AVMR không tuyến tính theo một phương án của sáng chế.

FIG.10 là hình vẽ minh họa ví dụ về phương pháp AMVR không tuyến tính có độ chính xác MVD thích ứng.

FIG.11 là hình vẽ minh họa phương pháp thu MVD theo độ chính xác MVD thích ứng theo một phương án của sáng chế.

FIG.12 đến FIG.14 là các hình vẽ minh họa phương pháp giải mã các giá trị MVD theo độ chính xác điểm nguyên, độ chính xác nửa điểm, hoặc độ chính xác phần tư điểm theo một phương án của sáng chế.

FIG.15 là hình vẽ giản lược minh họa ví dụ về phương pháp mã hóa video theo một phương án của sáng chế.

FIG.16 là hình vẽ giản lược minh họa ví dụ về phương pháp dự báo giữa các khung theo một phương án của sáng chế.

Mô tả chi tiết sáng chế

Sáng chế có thể được cải biến theo các dạng khác nhau, và các phương

án cụ thể của sáng chế sẽ được mô tả và thể hiện trong hình vẽ. Tuy nhiên, các phương án không nhằm giới hạn sáng chế. Các thuật ngữ được sử dụng trong phần mô tả dưới đây được sử dụng chỉ để mô tả các phương án cụ thể, mà không nhằm giới hạn sáng chế. Việc biểu thị số ít bao gồm việc biểu thị số nhiều, chỉ cần hiểu rõ là khác nhau. Thuật ngữ như “bao gồm” và “có” nhằm chỉ báo là các dấu hiệu, số, bước, thao tác, phần tử, bộ phận, hoặc kết hợp của chúng được sử dụng trong phần mô tả dưới đây là tồn tại và do vậy nên được hiểu là khả năng tồn tại hoặc bổ sung một hoặc nhiều dấu hiệu, số, bước, thao tác, phần tử, bộ phận, hoặc kết hợp khác của chúng không bị loại trừ.

Mặt khác, các phần tử trên hình vẽ được mô tả trong sáng chế được vẽ một cách độc lập để thuận lợi cho việc giải thích các chức năng cụ thể khác nhau trong thiết bị mã hóa/giải mã ảnh và không có nghĩa là các phần tử được thể hiện bởi phần cứng độc lập hoặc phần mềm độc lập. Ví dụ, hai hoặc nhiều phần tử trong số các phần tử này có thể được kết hợp để tạo một phần tử, hoặc một phần tử có thể được chia thành nhiều phần tử. Các phương án trong đó các phần tử được kết hợp và/hoặc được phân tách đều thuộc về sáng chế mà không nằm ngoài khái niệm của sáng chế.

Dưới đây, các phương án ví dụ của sáng chế sẽ được mô tả chi tiết có dựa vào các hình vẽ kèm theo.

FIG.1 là sơ đồ khối minh họa giản lược thiết bị mã hóa video theo một phương án của sáng chế.

Tham chiếu FIG.1, thiết bị mã hóa video 100 bao gồm môđun phân chia hình ảnh 105, môđun dự báo 110, môđun biến đổi 115, môđun lượng tử hóa 120, môđun sắp xếp lại 125, môđun mã hóa entropy 130, môđun giải lượng tử hóa 135, môđun biến đổi ngược 140, môđun lọc 145, và bộ nhớ 150.

Môđun phân chia hình ảnh 105 có thể được tạo cấu hình để tách hình ảnh đầu vào thành ít nhất một khối đơn vị xử lý. Trong kết nối này, khối làm đơn vị xử lý có thể là đơn vị dự báo PU, đơn vị biến đổi TU, hoặc đơn vị mã hóa CU.

Hình ảnh có thể gồm các đơn vị cây mã hóa CTU. Từng CTU có thể được tách thành các CU làm cấu trúc cây tứ phân. CU có thể được tách thành các CU có chiều sâu lớn hơn làm các cấu trúc cây tứ phân. PU và TU có thể thu được từ CU. Ví dụ, PU có thể được phân chia từ CU thành cấu trúc vuông đối xứng hoặc không đối xứng. Hơn thế nữa, TU có thể được tách thành cấu trúc cây tứ phân từ CU.

Môđun dự báo 110 bao gồm môđun dự báo giữa các khung mà thực hiện xử lý dự báo giữa các khung và môđun dự báo trong khung mà thực hiện xử lý dự báo trong khung, như sẽ được mô tả sau. Môđun dự báo 110 thực hiện xử lý dự báo trên các đơn vị xử lý của hình ảnh được chia bởi môđun chia hình ảnh 105 để tạo khối dự báo bao gồm mẫu dự báo hoặc mảng mẫu dự báo. Trong môđun dự báo 110, đơn vị xử lý của hình ảnh có thể là CU, TU, hoặc PU. Môđun dự báo 110 có thể xác định việc dự báo được thực hiện trên đơn vị xử lý tương ứng là dự báo giữa các khung hoặc dự báo trong khung, và có thể xác định các chi tiết cụ thể, ví dụ, chế độ dự báo của các phương pháp dự báo. Đơn vị xử lý được thực hiện xử lý dự báo có thể khác đơn vị xử lý có phương pháp dự báo và các chi tiết cụ thể được xác định. Ví dụ, phương pháp dự báo và chế độ dự báo có thể được xác định theo các đơn vị PU và xử lý dự báo có thể được thực hiện theo các đơn vị TU.

Trong dự báo giữa các khung, quá trình xử lý dự báo có thể được thực hiện trên cơ sở thông tin trên ít nhất một trong số hình ảnh trước và/hoặc hình ảnh sau của hình ảnh hiện thời để tạo khối dự báo. Trong dự báo trong khung, xử lý dự báo có thể được thực hiện trên cơ sở thông tin điểm ảnh của hình ảnh hiện thời để tạo khối dự báo.

Dự báo giữa các khung có thể sử dụng chế độ bỏ qua, chế độ trộn, hoặc AMVP. Trong dự báo giữa các khung, hình ảnh tham chiếu có thể được chọn cho PU, và khối tham chiếu tương ứng với PU có thể được chọn. Khối tham chiếu có thể là điểm ảnh nguyên hoặc đơn vị mẫu, điểm ảnh phân số hoặc đơn vị

mẫu. Khối dự báo sau đó được tạo ra mà tín hiệu dư liên quan đến PU hiện thời được giảm thiểu và kích thước vector chuyển động được giảm thiểu. Các điểm ảnh, điểm, và mẫu có thể được sử dụng thay thế lẫn nhau ở đây.

Khối dự báo có thể được tạo ra làm đơn vị điểm ảnh nguyên, hoặc đơn vị điểm ảnh phân số như $1/2$ đơn vị điểm ảnh hoặc $1/4$ đơn vị điểm ảnh. Trong kết nối này, vector chuyển động có thể cũng được biểu thị làm đơn vị điểm ảnh phân số.

Thông tin như chỉ số của hình ảnh tham chiếu được chọn qua dự báo giữa các khung, chênh lệch vector chuyển động MVD, dự báo vector chuyển động MVP, tín hiệu dư, v.v., có thể được mã hóa entropy và sau đó được truyền đến bộ giải mã. Khi chế độ bỏ qua được áp dụng, khối dự báo có thể được sử dụng làm khối khôi phục, để phần dư có thể không được tạo ra, biến đổi, lượng tử hóa, hoặc truyền.

Khi dự báo trong khung được thực hiện, chế độ dự báo có thể được xác định theo đơn vị PU và xử lý dự báo có thể được thực hiện theo đơn vị PU. Theo cách khác, chế độ dự báo có thể được xác định theo đơn vị PU và dự báo giữa các khung có thể được thực hiện theo đơn vị TU.

Các chế độ dự báo trong dự báo trong khung có thể bao gồm 33 chế độ dự báo theo hướng và ít nhất hai chế độ không theo hướng chẳng hạn. Các chế độ không theo hướng có thể bao gồm chế độ dự báo DC và chế độ phẳng.

Trong dự báo trong khung, khối dự báo có thể được tạo ra sau khi bộ lọc được áp dụng cho mẫu tham chiếu. Tại thời gian này, có thể xác định việc bộ lọc có cần được áp dụng cho mẫu tham chiếu hay không theo chế độ dự báo trong khung và/hoặc kích thước của khối hiện thời.

Các giá trị dư (khối dư hoặc tín hiệu dư) giữa khối dự báo được tạo ra và khối gốc được nhập vào môđun biến đổi 115. Thông tin chế độ dự báo, thông tin vector chuyển động, và tương tự được sử dụng để dự báo được mã hóa cùng với các giá trị dư bằng môđun mã hóa entropy 130 và được truyền đến bộ giải mã.

Môđun biến đổi 115 thực hiện xử lý biến đổi trên khối dư theo đơn vị TU và tạo các hệ số biến đổi.

Khối biến đổi là khối hình chữ nhật của các mẫu và là khối mà cùng biến đổi được áp dụng. Khối biến đổi có thể là TU và có thể có cấu trúc cây tứ phân.

Môđun biến đổi 115 có thể thực hiện xử lý biến đổi theo chế độ dự báo được áp dụng cho khối dư và kích thước của khối.

Ví dụ, khi dự báo trong khung được áp dụng cho khối dư và khối dư có mảng 4x4, khối dư được biến đổi bằng cách sử dụng DST (discrete sine transform – biến đổi sin rời rạc). Theo cách khác, khối dư có thể được biến đổi bằng cách sử dụng DCT (discrete cosine transform – biến đổi cosin rời rạc).

Môđun biến đổi 115 có thể tạo khối biến đổi của các hệ số biến đổi qua hoạt động biến đổi.

Môđun lượng tử hóa 120 có thể lượng tử hóa các giá trị dư, cụ thể là, các hệ số biến đổi, được biến đổi bằng môđun biến đổi 115 và có thể tạo các hệ số lượng tử hóa. Các giá trị được tính bằng môđun lượng tử hóa 120 có thể được cấp đến môđun giải lượng tử hóa 135 và môđun sắp xếp lại 125.

Môđun sắp xếp lại 125 có thể sắp xếp lại các hệ số biến đổi được cấp từ môđun lượng tử hóa 120. Bằng cách sắp xếp lại các hệ số lượng tử hóa, có thể tăng hiệu suất mã hóa trong môđun mã hóa entropy 130.

Môđun sắp xếp lại 125 có thể sắp xếp lại các hệ số biến đổi được lượng tử hóa ở dạng khối hai chiều thành dạng vector một chiều bằng cách sử dụng phương pháp quét hệ số.

Môđun mã hóa entropy 130 có thể được tạo cấu hình để mã hóa entropy ký hiệu theo phân bố xác suất dựa vào các giá trị được lượng tử hóa được sắp xếp lại bằng môđun sắp xếp lại 125 hoặc giá trị tham số mã hóa được tính trong xử lý mã hóa v.v., và sau đó xuất dòng bit. Phương pháp mã hóa entropy là phương pháp thu ký hiệu có các giá trị khác nhau và biểu thị ký hiệu là chuỗi nhị phân mà có thể được giải mã trong khi loại bỏ dư thừa thống kê.

Liên quan đến phương pháp này, ký hiệu nêu trên có nghĩa là phần tử cú pháp, tham số mã hóa, giá trị tín hiệu dư sẽ được mã hóa/giải mã v.v.. Tham số mã hóa là cần thiết để mã hóa và giải mã. Tham số mã hóa có thể chứa thông tin mà có thể được suy ra khi mã hóa hoặc giải mã, cũng như thông tin được mã hóa trong thiết bị mã hóa và được chuyển đến bộ giải mã giống như phần tử cú pháp. Tham số mã hóa là thông tin cần thiết để mã hóa hoặc giải mã ảnh. Tham số mã hóa có thể bao gồm các thông kê hoặc các giá trị như, ví dụ, chế độ dự báo trong khung/giữa các khung, vector dịch chuyển/chuyển động, chỉ số hình ảnh tham chiếu, mẫu khối mã hóa, sự có mặt hoặc không có mặt tín hiệu dư, hệ số biến đổi, hệ số biến đổi được lượng tử hóa, tham số lượng tử hóa, kích thước khối, thông tin phân chia khối, v.v.. Hơn thế nữa, tín hiệu dư có thể có nghĩa là chênh lệch giữa tín hiệu gốc và tín hiệu dự báo. Hơn thế nữa, chênh lệch giữa tín hiệu gốc và tín hiệu dự báo có thể được biến đổi để xác định tín hiệu dư, hoặc chênh lệch giữa tín hiệu gốc và tín hiệu dự báo có thể được biến đổi và lượng tử hóa để xác định tín hiệu dư. Tín hiệu dư có thể được gọi là khối dư trong đơn vị khối và tín hiệu dư có thể được gọi là mẫu dư trong đơn vị mẫu.

Khi mã hóa entropy được áp dụng, các ký hiệu có thể được biểu thị để số lượng nhỏ bit được cấp phát cho ký hiệu có xác suất xuất hiện cao, và số lượng lớn bit được cấp phát cho ký hiệu có xác suất xuất hiện thấp. Điều này có thể giảm kích thước của chuỗi bit dùng cho các ký hiệu sẽ được mã hóa. Do đó, tính năng nén của mã hóa ảnh có thể được tăng qua mã hóa entropy.

Các lược đồ mã hóa như golomb ngoại suy, CAVLC (Context-Adaptive Variable Length Coding – mã hóa chiều dài thay đổi thích ứng ngữ cảnh), và CABAC (Context-Adaptive Binary Arithmetic Coding – mã hóa số học nhị phân thích ứng ngữ cảnh) có thể được sử dụng để mã hóa entropy. Ví dụ, mô đun mã hóa entropy 130 có thể lưu trữ bảng để thực hiện mã hóa entropy, như bảng VLC (variable length coding/code – mã/mã hóa chiều dài thay đổi). Mô đun mã hóa entropy 130 có thể thực hiện mã hóa entropy bằng cách sử dụng bảng VLC

được lưu trữ. Hơn thế nữa, môđun mã hóa entropy 130 thu phương pháp nhị phân hóa của ký hiệu tương ứng và mô hình xác suất của ký hiệu/học tương ứng, và sau đó thực hiện mã hóa entropy bằng cách sử dụng phương pháp nhị phân hóa hoặc mô hình xác suất thu được.

Môđun mã hóa entropy 130 có thể thực hiện thay đổi định trước đối với tập hợp tham số hoặc các cú pháp sẽ được truyền, nếu cần.

Môđun giải lượng tử hóa 135 giải lượng tử hóa các giá trị hệ số biến đổi được lượng tử hóa bằng môđun lượng tử hóa 120. Môđun biến đổi ngược 140 biến đổi ngược các giá trị được giải lượng tử hóa bằng môđun giải lượng tử hóa 135.

Giá trị dư hoặc mẫu dư hoặc mảng mẫu dư được tạo ra bởi môđun giải lượng tử hóa 135 và môđun biến đổi ngược 140, và khối dự báo được dự báo bởi môđun dự báo 110 có thể được kết hợp để tạo khối được khôi phục bao gồm mẫu được khôi phục hoặc mảng mẫu được khôi phục.

Trên FIG.1, khối dư và khối dự báo được bổ sung để tạo khối được khôi phục bằng bộ cộng. Tại thời gian này, bộ cộng có thể được coi là môđun tạo khối được khôi phục để tạo khối được khôi phục.

Môđun lọc 145 áp dụng bộ lọc giải khối, bộ lọc ALF (Adaptive Loop Filter – lọc vòng lặp thích ứng), SAO (Sample Adaptive Offset – độ lệch thích ứng mẫu) cho hình ảnh được khôi phục.

Bộ lọc giải khối loại bỏ sự méo khối được tạo ra tại ranh giới giữa các khối trong hình ảnh được khôi phục. ALF thực hiện xử lý lọc trên cơ sở các giá trị kết quả so sánh của hình ảnh gốc với hình ảnh được khôi phục có các khối được lọc bằng bộ lọc giải khối. ALF có thể được áp dụng chỉ khi hiệu suất cao là cần thiết. SAO khôi phục các khác biệt độ lệch giữa các khối dư có bộ lọc giải khối được áp dụng và hình ảnh gốc và được áp dụng ở dạng độ lệch dải, độ lệch mép, hoặc tương tự.

Mặt khác, môđun lọc 145 có thể không thực hiện thao tác lọc trên khối

được khôi phục được sử dụng trong dự báo giữa các khung.

Bộ nhớ 150 có thể lưu trữ khối được khôi phục hoặc hình ảnh được tính bằng mô đun lọc 145. Khối được khôi phục hoặc hình ảnh được lưu trữ trong bộ nhớ 150 có thể được cấp đến mô đun dự báo 110 mà thực hiện dự báo giữa các khung.

FIG.2 là sơ đồ khối minh họa giản lược bộ giải mã video theo một phương án của sáng chế. Tham chiếu FIG.2, bộ giải mã video 200 có thể bao gồm mô đun giải mã entropy 210, mô đun sắp xếp lại 215, mô đun giải lượng tử hóa 220, mô đun biến đổi ngược 225, mô đun dự báo 230, mô đun lọc 235, và bộ nhớ 240.

Khi dòng bit video được nhập từ thiết bị mã hóa video, dòng bit đầu vào có thể được giải mã trên cơ sở thứ tự trong đó thông tin video được xử lý bằng thiết bị mã hóa video.

Mô đun giải mã entropy 210 có thể giải mã entropy dòng bit đầu vào theo phân bố xác suất để tạo các ký hiệu ở dạng hệ số được lượng tử hóa. Phương pháp giải mã entropy là phương pháp thu dãy các số nhị phân và tạo từng ký hiệu bằng cách sử dụng dãy. Phương pháp giải mã entropy tương tự phương pháp mã hóa entropy được mô tả trên đây.

Ví dụ, khi VLC (Variable Length Coding – mã hóa chiều dài thay đổi) (dưới đây được gọi là 'VLC') như CAVLC được sử dụng để thực hiện mã hóa entropy trong thiết bị mã hóa video, mô đun giải mã entropy 210 có thể thực hiện giải mã bằng cách sử dụng bảng VLC giống như thiết bị mã hóa được sử dụng trong thiết bị mã hóa. Hơn thế nữa, khi CABAC được sử dụng để thực hiện mã hóa entropy trong thiết bị mã hóa video, mô đun giải mã entropy 210 có thể thực hiện giải mã entropy bằng cách sử dụng CABAC.

Cụ thể hơn là, phương pháp giải mã entropy CABAC có thể bao gồm các bước: thu học tương ứng với từng phần tử cú pháp trong dòng bit, xác định mô hình ngữ cảnh bằng cách sử dụng thông tin phần tử cú pháp sẽ được giải mã,

giải mã thông tin của khối lân cận và khối sẽ được giải mã, hoặc thông tin của ký hiệu/học được giải mã trong bước trước, và dự báo xác suất xuất hiện của học theo mô hình ngữ cảnh được xác định và do vậy thực hiện giải mã thuật toán của học để tạo ký hiệu tương ứng với giá trị của từng phần tử cú pháp. Trong kết nối này, sau khi xác định mô hình ngữ cảnh, phương pháp giải mã entropy CABAC có thể còn bao gồm bước cập nhật mô hình ngữ cảnh bằng cách sử dụng thông tin của ký hiệu/học được giải mã để xác định mô hình ngữ cảnh của ký hiệu/học tiếp theo.

Thông tin để tạo khối được dự báo trong số các thông tin được giải mã bằng môđun giải mã entropy 210 có thể được cấp đến môđun dự báo 230, và các giá trị dư, cụ thể là, các hệ số biến đổi được lượng tử hóa, được giải mã entropy bằng môđun giải mã entropy 210 có thể được nhập vào môđun sắp xếp lại 215.

Môđun sắp xếp lại 215 có thể sắp xếp lại thông tin dòng bit, cụ thể là, các hệ số biến đổi được lượng tử hóa, được giải mã entropy bằng môđun giải mã entropy 210 trên cơ sở phương pháp sắp xếp lại trong thiết bị mã hóa video.

Môđun sắp xếp lại 215 có thể khôi phục và sắp xếp lại các hệ số được biểu thị ở dạng vectơ một chiều thành các hệ số ở dạng khối hai chiều. Môđun sắp xếp lại 215 có thể quét các hệ số trên cơ sở chế độ dự báo được áp dụng cho khối hiện thời, khối biến đổi và kích thước của khối biến đổi và có thể tạo mảng các hệ số biến đổi được lượng tử hóa ở dạng khối hai chiều.

Môđun giải lượng tử hóa 220 có thể thực hiện giải lượng tử hóa trên cơ sở các tham số lượng tử hóa được cấp từ thiết bị mã hóa video và các giá trị hệ số của khối được sắp xếp lại.

Môđun biến đổi ngược 225 có thể thực hiện DCT ngược và/hoặc DST ngược đối với DCT và/hoặc DST, mà đã được thực hiện bằng môđun biến đổi của thiết bị mã hóa video, trên kết quả lượng tử hóa từ thiết bị mã hóa video.

Xử lý biến đổi ngược này có thể được thực hiện trên cơ sở đơn vị chuyển hoặc đơn vị phân chia của hình ảnh được xác định bằng thiết bị mã hóa video.

Môđun biến đổi của thiết bị mã hóa video có thể tùy ý thực hiện DCT và/hoặc DST theo các thông tin như phương pháp dự báo, kích thước của khối hiện thời, và hướng dự báo, và môđun biến đổi ngược 225 của bộ giải mã video có thể thực hiện biến đổi ngược trên cơ sở thông tin biến đổi trên biến đổi được thực hiện bằng môđun biến đổi của thiết bị mã hóa video.

Môđun dự báo 230 tạo khối dự báo bao gồm mẫu dự báo hoặc mảng mẫu dự báo dựa vào thông tin liên quan đến tạo khối dự báo được cung cấp bằng môđun giải mã entropy 210 và khối và/hoặc thông tin hình ảnh được giải mã trước từ bộ nhớ 240.

Nếu chế độ dự báo cho PU hiện thời là chế độ dự báo trong khung, môđun dự báo 230 có thể thực hiện dự báo trong khung để tạo khối dự báo dựa vào thông tin điểm ảnh trong hình ảnh hiện thời.

Nếu chế độ dự báo cho PU hiện thời là chế độ dự báo giữa các khung, môđun dự báo 230 có thể được tạo cấu hình để thực hiện dự báo giữa các khung trên PU hiện thời dựa vào thông tin có trong ít nhất một hình ảnh của hình ảnh trước hoặc hình ảnh sau đối với hình ảnh hiện thời. Trong kết nối này, thông tin về thông tin chuyển động cần thiết cho dự báo giữa các khung của PU hiện thời được cung cấp trong thiết bị mã hóa video, như vectơ chuyển động và chỉ số hình ảnh tham chiếu có thể được suy ra qua kiểm tra cờ bỏ qua và cờ trộn thu được từ thiết bị mã hóa.

Môđun dự báo 230 có thể tạo khối dự báo để tín hiệu dự liên quan đến khối hiện thời được giảm thiểu và kích thước vectơ chuyển động được giảm thiểu khi dự báo giữa các khung được thực hiện trên hình ảnh hiện thời.

Mặt khác, phương pháp thu thông tin chuyển động có thể được thay đổi theo chế độ dự báo của khối hiện thời. Chế độ dự báo được áp dụng cho dự báo giữa các khung có thể bao gồm chế độ AMVP, chế độ trộn, và tương tự.

Trong một ví dụ, khi chế độ trộn được áp dụng, thiết bị mã hóa và thiết bị giải mã có thể tạo danh sách ứng viên trộn bằng cách sử dụng vectơ chuyển

động của khối lân cận không gian được khôi phục và/hoặc vector chuyển động tương ứng với khối Col là khối lân cận thời gian. Trong chế độ trộn, vector của khối ứng viên được chọn trong danh sách ứng viên trộn được sử dụng làm vector chuyển động của khối hiện thời. Thiết bị mã hóa có thể truyền chỉ số trộn chỉ báo khối ứng viên có vector chuyển động tối ưu được chọn từ các khối ứng viên có trong danh sách ứng viên trộn đến thiết bị giải mã. Trong trường hợp này, thiết bị giải mã có thể thu vector chuyển động của khối hiện thời bằng cách sử dụng chỉ số trộn.

Trong ví dụ khác, khi chế độ AMVP được áp dụng, thiết bị mã hóa và bộ giải mã tạo danh sách ứng viên dự báo vector chuyển động bằng cách sử dụng vector chuyển động của khối lân cận không gian được khôi phục và/hoặc vector chuyển động tương ứng với khối Col làm khối lân cận thời gian. Cụ thể là, vector chuyển động của khối lân cận không gian được khôi phục và/hoặc vector chuyển động tương ứng với khối Col làm khối lân cận thời gian có thể được sử dụng làm ứng viên vector chuyển động. Thiết bị mã hóa có thể truyền đến bộ giải mã chỉ số vector chuyển động dự báo chỉ báo vector chuyển động tối ưu được chọn từ các ứng viên vector chuyển động có trong danh sách ứng viên dự báo vector chuyển động. Trong kết nối này, bộ giải mã có thể chọn vector chuyển động dự báo cho khối hiện thời từ các ứng viên vector chuyển động có trong danh sách ứng viên vector chuyển động bằng cách sử dụng chỉ số vector chuyển động.

Thiết bị mã hóa có thể thu chênh lệch vector chuyển động MVD giữa vector chuyển động dùng cho khối hiện thời và bộ dự báo vector chuyển động, mã hóa MVD, và truyền MVD được mã hóa đến bộ giải mã. Cụ thể là, MVD có thể là giá trị thu được bằng cách trừ MVP từ MV dùng cho khối hiện thời. Trong kết nối này, bộ giải mã có thể giải mã chênh lệch vector chuyển động thu được, và thu vector chuyển động cho khối hiện thời bằng cách cộng chênh lệch vector chuyển động và bộ dự báo vector chuyển động được giải mã.

Hơn thế nữa, thiết bị mã hóa có thể truyền chỉ số hình ảnh tham chiếu chỉ

báo hình ảnh tham chiếu đến bộ giải mã.

Bộ giải mã có thể dự báo vector chuyển động của khối hiện thời bằng cách sử dụng thông tin chuyển động của khối lân cận và thu vector chuyển động của khối hiện thời bằng cách sử dụng phần dư thu được từ thiết bị mã hóa. Bộ giải mã có thể tạo khối dự báo cho khối hiện thời dựa vào vector chuyển động thu được và thông tin chỉ số hình ảnh tham chiếu thu được từ thiết bị mã hóa.

Trong ví dụ khác, khi chế độ trộn được áp dụng, thiết bị mã hóa và bộ giải mã có thể tạo danh sách ứng viên trộn bằng cách sử dụng thông tin chuyển động của khối lân cận được khôi phục và/hoặc thông tin chuyển động của khối Col. Cụ thể là, khi khối lân cận được khôi phục và/hoặc thông tin chuyển động của khối Col tồn tại, thiết bị mã hóa và bộ giải mã có thể sử dụng khối lân cận được khôi phục và/hoặc thông tin chuyển động của khối Col làm ứng viên trộn cho khối hiện thời.

Thiết bị mã hóa có thể chọn ứng viên trộn mà cung cấp hiệu suất mã hóa tối ưu trong số các ứng viên trộn có trong danh sách ứng viên trộn làm thông tin chuyển động cho khối hiện thời. Trong kết nối này, chỉ số trộn chỉ báo ứng viên trộn được chọn có thể có trong dòng bit mà được truyền đến bộ giải mã. Bộ giải mã có thể chọn một trong số các ứng viên trộn có trong danh sách ứng viên trộn bằng cách sử dụng chỉ số trộn được truyền, và bộ giải mã có thể xác định ứng viên trộn được chọn làm thông tin chuyển động của khối hiện thời. Do đó, khi chế độ trộn được áp dụng, thông tin chuyển động của khối lân cận được khôi phục và/hoặc khối Col có thể được sử dụng làm thông tin chuyển động cho khối hiện thời như bình thường. Bộ giải mã có thể khôi phục khối hiện thời bằng cách cộng khối dự báo và phần dư được truyền từ thiết bị mã hóa với nhau.

Trong AMVP và các chế độ trộn được mô tả trên đây, thông tin chuyển động của khối lân cận được khôi phục và/hoặc thông tin chuyển động của khối Col có thể được sử dụng để thu thông tin chuyển động của khối hiện thời.

Trong chế độ bỏ qua, mà là một trong số các chế độ khác được sử dụng

cho dự báo giữa các hình ảnh, thông tin khối lân cận có thể được sử dụng cho khối hiện thời như bình thường. Do đó, trong trường hợp của chế độ bỏ qua, thiết bị mã hóa không truyền thông tin cú pháp như phần dư đến bộ giải mã thêm vào thông tin chỉ báo thông tin chuyển động của khối nào để sử dụng làm thông tin chuyển động cho khối hiện thời.

Thiết bị mã hóa và bộ giải mã có thể tạo khối dự báo của khối hiện thời bằng cách thực hiện bù chuyển động trên khối hiện thời dựa vào thông tin chuyển động thu được. Trong kết nối này, khối dự báo có thể nói đến khối được bù chuyển động như được tạo bằng cách thực hiện bù chuyển động trên khối hiện thời. Hơn thế nữa, các khối được bù chuyển động có thể tạo thành một ảnh được bù chuyển động.

Khối được khôi phục có thể được tạo ra bằng cách sử dụng khối dự báo được tạo ra bằng môđun dự báo 230 và khối dư được cung cấp bởi môđun biến đổi ngược 225. FIG.2 minh họa việc sử dụng bộ cộng, khối dự báo và khối dư được kết hợp để tạo khối được khôi phục. Trong kết nối này, bộ cộng có thể được coi là môđun riêng rẽ (môđun tạo khối được khôi phục) mà được tạo cấu hình để tạo khối được khôi phục. Trong kết nối này, khối được khôi phục bao gồm mẫu được khôi phục hoặc mảng mẫu được khôi phục như được mô tả trên đây; khối dự báo bao gồm mẫu dự báo hoặc mảng mẫu dự báo; khối dư có thể bao gồm mẫu dư hoặc mảng mẫu dư. Do đó, mẫu được khôi phục hoặc mảng mẫu được khôi phục có thể được coi được tạo ra bằng cách kết hợp mẫu dự báo hoặc mảng mẫu dự báo tương ứng với mẫu dư hoặc mảng mẫu dư tương ứng.

Khi chế độ bỏ qua được sử dụng cho khối, tín hiệu dư có thể không được truyền và khối được dự báo có thể được sử dụng làm khối được khôi phục.

Khối được khôi phục và/hoặc hình ảnh có thể được cấp đến môđun lọc 235. Môđun lọc 235 có thể thực hiện thao tác lọc giải khối, thao tác SAO, và/hoặc thao tác ALF trên khối được khôi phục và/hoặc hình ảnh.

Bộ nhớ 240 có thể lưu trữ hình ảnh hoặc khối được khôi phục để sử dụng

làm hình ảnh tham chiếu hoặc khối tham chiếu và có thể cấp hình ảnh được khôi phục đến môđun xuất.

Các phần tử mà trực tiếp liên quan đến giải mã các ảnh trong số môđun giải mã entropy 210, môđun sắp xếp lại 215, môđun giải lượng tử hóa 220, môđun biến đổi ngược 225, môđun dự báo 230, môđun lọc 235 và bộ nhớ 240 có trong thiết bị giải mã 200, ví dụ, môđun giải mã entropy 210, môđun sắp xếp lại 215, môđun giải lượng tử hóa 220, môđun biến đổi ngược 225, môđun dự báo 230, môđun lọc 235, v.v., có thể được biểu thị là bộ giải mã hoặc đơn vị giải mã khác biệt các phần tử khác.

Ngoài ra, thiết bị giải mã 200 có thể còn bao gồm bộ phân tách không được thể hiện trong hình vẽ mà phân tách thông tin liên quan đến ảnh được mã hóa có trong dòng bit. Bộ phân tách có thể bao gồm môđun giải mã entropy 210, và có thể có trong môđun giải mã entropy 210. Bộ phân tách này có thể cũng được thực hiện làm phần tử của đơn vị giải mã.

FIG.3 là hình vẽ giản lược minh họa một phương án của khối ứng viên mà có thể sử dụng được khi dự báo giữa các khung được thực hiện trên khối hiện thời. Ở đây, khối hiện thời có thể là khối dự báo.

Các môđun dự báo của thiết bị mã hóa và thiết bị giải mã có thể sử dụng khối lân cận được khôi phục tại vị trí định trước quanh khối hiện thời 300 làm khối ứng viên. Ví dụ, trong ví dụ của FIG.3, hai khối A0 310 và A1 320 ở phía trái của khối hiện thời và ba khối B0 330, B1 340 và B2 350 ở phía trên của khối hiện thời có thể được chọn làm các khối ứng viên không gian. Ở đây, A0 310 có thể được tham chiếu đến là khối lân cận dưới bên trái, và A1 320 có thể được tham chiếu đến là khối lân cận trái. Ngoài ra, B0 330 có thể được tham chiếu đến là khối lân cận trên bên phải, B1 340 có thể được tham chiếu đến là khối lân cận trên, và B2 350 có thể được tham chiếu đến là khối lân cận trên bên trái.

Ngoài các khối lân cận không gian, khối Col 360 được mô tả trên đây có thể được sử dụng làm khối ứng viên thời gian. Khối Col 360 có thể được tham

chiều đến là khối dự báo Col (ColPb), và có thể là khối tương ứng với khối hiện thời trong hình ảnh cùng chỗ, mà là một trong số các hình ảnh tham chiếu được khôi phục và có thể là khối tồn tại tại vị trí tương đối nhất định (ví dụ, tại vị trí được dịch theo phương pháp số học theo tiêu chuẩn nào đó từ vị trí mẫu lân cận bên dưới bên phải hoặc vị trí mẫu giữa bên dưới bên phải của khối tồn tại tại vị trí giống như khối hiện thời trong hình ảnh Col).

Cụ thể là, trong chế độ AMVP, MVP tối ưu đối với khối hiện thời được chọn từ danh sách ứng viên MVP bao gồm các ứng viên MVP thu được từ các khối ứng viên. Trong trường hợp này, thiết bị mã hóa thực hiện đánh giá chuyển động để thu được MVP tối ưu từ danh sách ứng viên MVP dựa vào MV thu được của khối hiện thời, và tính MVD thu được bằng cách trừ MVP từ MV. Thiết bị mã hóa mã hóa thông tin chỉ số MVP chỉ báo ứng viên MVP nào trong số các ứng viên MVP có trong danh sách ứng viên MVP là MVP dùng cho khối hiện thời và thông tin MVD chỉ báo giá trị trục x và trục y của MVD thu được, và truyền thông tin được mã hóa đến thiết bị giải mã qua dòng bit.

Thiết bị giải mã có thể thu MVP dùng cho khối hiện thời từ danh sách ứng viên MVP dựa vào thông tin chỉ số MVP và thông tin MVD được truyền từ thiết bị mã hóa, và có thể thu MV của khối hiện thời bằng cách cộng MVD vào MVP thu được. Khối tham chiếu trên hình ảnh tham chiếu thu được dựa vào MV của khối hiện thời, và khối tham chiếu có thể được sử dụng làm khối dự báo cho khối hiện thời. Cụ thể là, các mẫu trong khối tham chiếu có thể được sử dụng làm các mẫu dự báo cho khối hiện thời. Thiết bị giải mã có thể thu thông tin trên các mẫu dự từ thiết bị mã hóa để tạo các mẫu dự. Thông tin trên các mẫu dự có thể bao gồm thông tin trên các hệ số biến đổi. Cụ thể là, ví dụ, thiết bị giải mã có thể thu các hệ số biến đổi từ thiết bị mã hóa qua dòng bit, và đảo các hệ số biến đổi để tạo khối dự (hoặc các mẫu dự). Ở đây, mẫu dự có thể biểu thị chênh lệch giữa mẫu gốc và mẫu dự báo, và khối dự có thể biểu thị chênh lệch giữa khối gốc bao gồm các mẫu gốc và khối dự báo bao gồm các mẫu dự báo.

Vector chuyển động có thể có độ phân giải mẫu nhỏ hơn hoặc bằng các đơn vị nguyên. Ví dụ, vector chuyển động có thể có $1/4$ độ phân giải mẫu đối với thành phần độ sáng. Do vậy, khối tham chiếu mà tương tự khối hiện thời hơn có thể được chỉ báo bằng cách tạo $1/4$ mẫu phân số đơn vị từ mẫu nguyên hoặc mẫu đầy đủ bằng cách nội suy trên hình ảnh tham chiếu, và chọn khối tham chiếu trong vùng bao gồm mẫu phân số.

Mẫu phân số nhỏ hơn hoặc bằng các đơn vị nguyên có thể được tạo ra qua bộ lọc nội suy dựa vào mẫu nguyên. Như được mô tả trên đây, trong trường hợp của mẫu thành phần độ sáng (dưới đây được gọi là mẫu độ sáng), độ phân giải của vector chuyển động là $1/4$ mẫu phân số, và thiết bị mã hóa và thiết bị giải mã có thể tạo thông tin mẫu nhỏ hơn hoặc bằng đơn vị nguyên bằng $1/4$ đơn vị mẫu qua nội suy. Ví dụ, để thực hiện nội suy cho mẫu độ sáng, bộ lọc nội suy 8 nhánh với các hệ số lọc khác nhau có thể được sử dụng.

FIG.4 là hình vẽ minh họa các vị trí của các mẫu nguyên và phân số để nội suy $1/4$ mẫu phân số trong dự báo giữa các khung. Trong số các vị trí của các mẫu được thể hiện trên FIG.4, các vị trí tô bóng (hoặc được đánh dấu bằng chữ hoa) tương ứng với các mẫu nguyên, và các vị trí không được tô bóng (hoặc được đánh dấu bằng chữ thường) tương ứng với các mẫu phân số.

Bảng 1 dưới đây thể hiện ví dụ về các hệ số lọc theo vị trí mẫu. Ví dụ, các hệ số lọc có thể được áp dụng cho các mẫu của các thành phần độ sáng.

Bảng 1

Vị trí mẫu	Hệ số lọc
$1/4$	$\{-1, 4, -10, 58, 17, -5, 1, 0\}$
$2/4$	$\{-1, 4, -11, 40, 40, -11, 4, -1\}$
$3/4$	$\{0, 1, -5, 17, 58, -10, 4, -1\}$

Ví dụ, các mẫu phân số của FIG.4 có thể thu được bằng cách áp dụng bộ lọc 8 nhánh dựa vào các hệ số lọc.

Do vậy, khối tham chiếu dò được trong đơn vị mẫu phân số, và MV của

đơn vị mẫu phân số thu được, để dự báo giữa các khung có thể được thực hiện chính xác hơn nữa. Trong trường hợp này, lượng dữ liệu được cấp phát cho tín hiệu dư có thể được giảm. Trong trường hợp này, tuy nhiên, MVD cũng cần được chỉ báo trong các đơn vị mẫu phân số, và lượng dữ liệu được cấp phát cho MVD tăng tương đối.

Mặt khác, hiệu suất mã hóa có thể được cải thiện bằng cách điều chỉnh thích ứng khoảng hoặc độ phân giải của vector chuyển động. Có thể được tham chiếu đến là AMVR, và MV có thể cũng được xác định bằng đơn vị 1/2 mẫu phân số (hoặc nửa mẫu) hoặc đơn vị mẫu nguyên để giảm thông tin. Trong trường hợp này, đơn vị mẫu phân số hoặc đơn vị mẫu nguyên thống nhất hoặc tương tự có thể được sử dụng trong toàn bộ khoảng, hoặc khoảng của đơn vị mẫu có thể được thiết lập theo cách khác theo cách thích ứng theo vùng. Trường hợp đầu có thể được tham chiếu đến là AMVR tuyến tính, và trường hợp sau có thể được tham chiếu đến là AMVR không tuyến tính.

Phương pháp thu MV dựa vào AMVR là như sau.

FIG.5 là hình vẽ minh họa ví dụ về phương pháp thu MV dựa vào AMVR. Phương pháp được bộc lộ trên FIG.5 có thể được thực hiện bằng thiết bị giải mã.

Tham chiếu FIG.5, thiết bị giải mã thu một MVP từ danh sách ứng viên MVP được tạo thành dựa vào các khối lân cận không gian và khối tương ứng thời gian của khối hiện thời. Ở đây, MVP có thể là MV của một trong số các khối lân cận không gian và khối tương ứng thời gian, và do vậy ban đầu có thể là đơn vị 1/4 điểm. Thiết bị giải mã thu MVP của đơn vị điểm nguyên qua thủ tục làm tròn.

MVD có thể thu được theo đơn vị điểm nguyên, và trong trường hợp này, thiết bị giải mã tăng MVD thu được. Thiết bị giải mã tăng giá trị của MVD để phân biệt giá trị của đơn vị MVD từ đơn vị 1/4 điểm và do vậy thu MVD của đơn vị điểm nguyên. Cụ thể là, giá trị 1 của MVD của đơn vị 1/4 điểm có thể biểu thị 1/4 điểm, và giá trị 4 có thể biểu thị 1 điểm nguyên. Thiết bị giải mã có

thể cho phép giá trị 1 của MVD của đơn vị điểm nguyên biểu thị 1 điểm nguyên và cho phép giá trị 4 biểu thị 4 điểm nguyên qua thủ tục tăng.

Thiết bị giải mã thu MV của đơn vị điểm nguyên dựa vào MVP của đơn vị điểm nguyên và MVD của đơn vị điểm nguyên. Trong trường hợp này, thiết bị giải mã có thể thu MV của đơn vị điểm nguyên bằng cách cộng MVP của đơn vị điểm nguyên và MVD của đơn vị điểm nguyên. Như được mô tả trên đây, điểm ảnh, điểm, và mẫu có thể được sử dụng thay thế lẫn nhau trong bản mô tả này.

Mặt khác, thiết bị mã hóa tìm vị trí mẫu có chi phí RD (Rate-Distortion) tối ưu qua đánh giá chuyển động để xác định MV thứ nhất (thời gian). Trong trường hợp này, MV thứ nhất có thể là MV của đơn vị điểm phân số. Thiết bị mã hóa tạo danh sách ứng viên MVP theo cách giống như thiết bị giải mã và thu MVP thứ nhất. Trong trường hợp này, MVP thứ nhất có thể là đơn vị điểm phân số. Thiết bị mã hóa làm tròn xuống MVP thứ nhất để thu MVP thứ hai của đơn vị điểm nguyên.

Thiết bị mã hóa tạo MVD thứ nhất dựa vào giá trị chênh lệch giữa MV thứ nhất và MVP thứ nhất, và sau đó làm tròn MVD thứ nhất để thu MVD thứ hai của đơn vị điểm nguyên.

Thiết bị mã hóa thu MV thứ hai của đơn vị điểm nguyên dựa vào việc cộng MVP thứ hai và MVD thứ hai.

Thiết bị mã hóa so sánh chi phí RD dựa vào MV thứ nhất và chi phí RD dựa vào MV thứ hai, và chọn chế độ có chi phí RD tốt hơn (cụ thể là, chi phí RD thấp hơn). Trong trường hợp này, chế độ thực hiện dự báo dựa vào MV thứ nhất có thể được tham chiếu đến là chế độ bình thường, và chế độ thực hiện dự báo dựa vào MV thứ hai có thể được tham chiếu đến là chế độ AMVR.

Cụ thể là, AMVR là phương pháp biểu thị MV theo đơn vị điểm nguyên và giảm kích thước tuyệt đối (cụ thể là, lượng bit) của MVD sẽ được mã hóa. MVD có thể được truyền, ví dụ, qua cú pháp như được thể hiện trong Bảng 2

dưới đây. Cú pháp có thể có trong dòng bit sẽ được truyền.

Bảng 2

	Descriptor
mvd_coding(x0, y0, refList) {	
abs_mvd_grater0_flag[0]	ae(v)
abs_mvd_grater0_flag[1]	ae(v)
if(abs_mvd_grater0_flag[0])	
abs_mvd_grater1_flag[0]	ae(v)
if(abs_mvd_grater0_flag[1])	
abs_mvd_grater1_flag[1]	ae(v)
if(abs_mvd_grater0_flag[0]) {	
if(abs_mvd_grater1_flag[0])	
abs_mvd_minus2[0]	ae(v)
mvd_sign_flag[0]	ae(v)
}	
if(abs_mvd_grater0_flag[1]) {	
if(abs_mvd_grater1_flag[1])	
abs_mvd_minus2[1]	ae(v)
mvd_sign_flag[1]	ae(v)
}	
}	

Tham chiếu Bảng 2, abs_mvd_grater0_flag [0/1] chỉ báo việc các giá trị tuyệt đối của thành phần x/thành phần y của chênh lệch vector chuyển động có lớn hơn 0 hay không.

abs_mvd_grater1_flag [0/1] chỉ báo việc các giá trị tuyệt đối của thành phần x/thành phần y của chênh lệch vector chuyển động có lớn hơn 1 hay không. abs_mvd_grater1_flag [0/1] có thể được truyền/thu và phân tách khi abs_mvd_grater0_flag [0/1] là đúng (cụ thể là, khi giá trị của abs_mvd_grater0_flag [0/1] là 1).

Giá trị thu được bằng cách cộng 2 vào abs_mvd_minus2 [0/1] biểu thị giá trị tuyệt đối của thành phần x/thành phần y của chênh lệch vector chuyển động. abs_mvd_minus2 [0/1] có thể được truyền/thu và phân tách khi abs_mvd_grater1_flag [0/1] là đúng (cụ thể là, khi giá trị của abs_mvd_grater1_flag [0/1] là 1).

mvd_sign_flag [0/1] chỉ báo dấu của thành phần x/thành phần y của chênh lệch vector chuyển động. Khi giá trị của mvd_sign_flag [0/1] là 0, thành phần x/thành phần y của chênh lệch vector chuyển động tương ứng có giá trị dương. Khi giá trị của mvd_sign_flag [0/1] là 1, thành phần x/thành phần y của chênh lệch vector chuyển động tương ứng có giá trị âm. mvd_sign_flag [0/1] có

thể được truyền/thu và phân tách khi `abs_mvd_grater0_flag [0/1]` là đúng (cụ thể là, khi giá trị của `abs_mvd_grater0_flag [0/1]` là 1).

Khi AMVR được áp dụng, vì MVD có thể được biểu thị theo đơn vị điểm nguyên bằng cách sử dụng các phân tử cú pháp như được mô tả trên đây, giá trị tuyệt đối của MVD có thể được giảm và do vậy các bit được truyền có thể được giảm.

Ngoài ra, theo một phương án của sáng chế, khoảng MV không tuyến tính có thể được áp dụng để tăng hơn nữa hiệu quả tiết kiệm bit. Theo một phương án của sáng chế, việc nén năng lượng được tăng bằng cách sử dụng đơn vị điểm nguyên hoặc đơn vị trên đây trong khoảng mà sự tương quan của các giá trị được dự báo là cao. Ngoài ra, bằng cách biểu thị thích ứng đến vị trí nửa điểm và phân tử điểm theo khoảng cũng nhưng điểm nguyên trong khoảng mà sự tương quan của các giá trị được dự báo là tương đối thấp, hiệu quả tiết kiệm bit có thể đạt được trong khi duy trì độ chính xác dự báo.

Ví dụ 1

Phương án thứ nhất của sáng chế cung cấp phương pháp có thể thực hiện việc truyền bit hiệu quả hơn bằng cách thực hiện hiệu quả việc nén năng lượng của MVD trong khi duy trì khoảng MVD tuyến tính.

FIG.6 là hình vẽ minh họa phương pháp AVMR có khoảng tuyến tính.

Tham chiếu FIG.6, khoảng MV tuyến tính có thể dựa vào phương pháp làm tròn xuống giá trị MVD của đơn vị phân tử điểm thành bội số của 4 và sau đó giảm giá trị MVD xuống 4. Trong trường hợp này, ví dụ, MVD của đơn vị phân tử điểm có giá trị -4 đến -1 trở thành MVD của đơn vị điểm nguyên có giá trị -1, và MVD của đơn vị phân tử điểm có giá trị 0 đến 3 trở thành MVD của đơn vị điểm nguyên có giá trị 0. Ngoài ra, MVD của đơn vị phân tử điểm có giá trị 4 đến 7 trở thành MVD của đơn vị điểm nguyên có giá trị 1.

Trong khi đó, trong cú pháp liên quan đến MVD như được mô tả trong Bảng 2, thực hiện kiểm tra xem kích thước của giá trị tuyệt đối của MVD có lớn

hơn 0 hay không, và khi kích thước của giá trị tuyệt đối của MVD không lớn hơn 0, `abs_mvd_grater0_flag` 1 bit có giá trị 0 được truyền. Mặt khác, khi kích thước của giá trị tuyệt đối của MVD lớn hơn 0, thực hiện kiểm tra xem kích thước của giá trị tuyệt đối của MVD có lớn hơn 1 hay không. Khi không lớn hơn 1, `abs_mvd_grater1_flag` 1 bit có giá trị 0 được truyền. Mặt khác, khi cả hai trường hợp trên đây không được thỏa mãn (cụ thể là, khi MVD bằng hoặc lớn hơn 2), giá trị thu được bằng cách trừ 2 từ giá trị tuyệt đối của MVD được mã hóa trên cơ sở golomb ngoại suy chính. Vì vậy, khi khoảng được dịch đến số dương như được thể hiện trên FIG.6, có sự không cân bằng là hiệu suất mã hóa thay đổi theo việc giá trị của MVD là dương hoặc âm. Ví dụ, khi giá trị của MVD của đơn vị phân tử điểm là 3, giá trị cần được biểu thị bởi 0 và giá trị 0 được giảm bằng 4 đơn vị cần được chỉ báo. Do vậy, thiết bị mã hóa truyền `abs_mvd_grater0_flag` 1 bit có giá trị 0. Tuy nhiên, nếu giá trị của MVD của đơn vị phân tử điểm là -3, giá trị được biểu thị là -4 và giá trị -3 được giảm bằng 4 đơn vị cần được chỉ báo. Trong trường hợp này, thiết bị mã hóa cần truyền tổng cộng 2 bit của `abs_mvd_grater0_flag` 1 bit có giá trị 1 và `abs_mvd_grater1_flag` 1 bit có giá trị 0. Để giải quyết sự không cân bằng này, khoảng tuyến tính cân bằng có thể được sử dụng như được thể hiện trên FIG.7.

FIG.7 là hình vẽ minh họa phương pháp AVMR có khoảng tuyến tính cân bằng.

Tham chiếu FIG.7, khoảng MVD tuyến tính cân bằng có thể dựa vào phương pháp làm tròn xuống giá trị MVD của đơn vị phân tử điểm cộng thêm 2 thành bội số của 4 và sau đó giảm xuống 4. Cụ thể là, trong trường hợp này, ví dụ, MVD của đơn vị phân tử điểm có giá trị -6 đến -3 được biểu thị bằng giá trị đại diện -4 và trở thành MVD của đơn vị điểm nguyên có giá trị -1, và MVD của đơn vị phân tử điểm có giá trị -2 đến 1 được biểu thị bằng giá trị đại diện 0 và trở thành MVD của đơn vị điểm nguyên có giá trị 0. Ngoài ra, MVD của đơn vị phân tử điểm có giá trị 2 đến 5 có giá trị 2 đến 5 được biểu thị bằng giá trị đại

diện 4 và trở thành MVD của đơn vị điểm nguyên có giá trị 1. Mặt khác, ví dụ, khoảng MVD tuyến tính cân bằng có thể dựa vào phương pháp làm tròn xuống giá trị MVD của đơn vị phần tư điểm cộng thêm 1 thành bội số của 4 và sau đó giảm xuống 4.

Bằng cách điều chỉnh khoảng tuyến tính được sử dụng trong phương pháp AMVR như được mô tả trên đây, có thể cải thiện sự không cân bằng mà hiệu suất mã hóa thay đổi theo việc giá trị MVD là dương hoặc âm.

Ví dụ 2

Phương án thứ hai của sáng chế cung cấp phương pháp giảm lượng bit sẽ được truyền bằng cách áp dụng khoảng MVD không tuyến tính thay vì khoảng MV tuyến tính và tăng tỷ lệ sử dụng của `abs_mvd_grater0_flag` hoặc `abs_mvd_grater1_flag`.

FIG.8 là hình vẽ minh họa ví dụ về phương pháp AVMR có khoảng không tuyến tính.

Tham chiếu FIG.8, trong khoảng MVD không tuyến tính, khoảng mà các giá trị MVD của đơn vị phần tư điểm có thể được giảm xuống -1, 0, 1 dựa vào 8 đơn vị, và khoảng còn lại dựa vào 4 đơn vị. Cụ thể là, vùng giữa có thể dựa vào phương pháp làm tròn xuống giá trị MVD của đơn vị phần tư điểm cộng thêm 4 thành bội số của 8 và sau đó giảm xuống 8. Ví dụ, MVD của đơn vị phần tư điểm có giá trị -12 đến -5 được biểu thị bằng giá trị đại diện -8 và được giảm để trở thành MVD của đơn vị điểm nguyên có giá trị -1, và MVD của đơn vị phần tư điểm có giá trị -4 đến 3 được biểu thị bằng giá trị đại diện 0 và được giảm để trở thành MVD của đơn vị điểm nguyên có giá trị 0. Ngoài ra, MVD của đơn vị phần tư điểm có giá trị 4 đến 11 được biểu thị bằng giá trị đại diện 8 và được giảm xuống MVD của đơn vị điểm nguyên có giá trị 1.

Trong trường hợp tạo khoảng MVD không tuyến tính như được mô tả trên đây, các MVD thuộc về khoảng không tuyến tính được biểu thị bằng các giá trị đại diện của khoảng tương ứng, và các giá trị sau khi thực hiện giảm tương

ứng theo các khoảng được mã hóa và truyền qua dòng bit. Trong trường hợp này, sự nén năng lượng có thể được cải thiện bằng cách tăng tỷ lệ được biểu thị bằng 0 hoặc 1 của giá trị MVD sẽ được mã hóa và truyền.

Mặt khác, AMVR không tuyến tính có thể được chỉ báo qua các phần tử cú pháp `non_linear_amvr_range_flag` như được thể hiện trong Bảng 3 sau.

Bảng 3

<code>sps temporal mvp enabled flag</code>
<code>strong intra smoothing enabled flag</code>
<code>amvr enable flag</code>
<code>if(amvr enable flag)</code>
<code> non_linear_amvr_range_flag</code>
<code> vui_parameters_present_flag</code>
<code> if(vui_parameters_present_flag)</code>
<code> vui_parameters()</code>
<code> sps_extension_flag</code>
<code> if(sps_extension_flag)</code>
<code> while(more_rbsp_data())</code>
<code> sps_extension_data_flag</code>
<code> rbsp_trailing_bits()</code>
<code>}</code>

Tham chiếu Bảng 3, `amvr_enable_flag` chỉ báo việc chế độ AMVR có khả dụng hay không. `non_linear_amvr_range_flag` chỉ báo việc khoảng AMVR không tuyến tính có được áp dụng hay không. `non_linear_amvr_range_flag` có thể được truyền/thu và phân tách khi giá trị của `amvr_enable_flag` là 1. Cú pháp được bộc lộ trong Bảng 3 trên đây có thể có trong, ví dụ, cú pháp SPS (Sequence Parameter Set).

Khi giá trị của `non_linear_amvr_range_flag` là 0, khoảng AMVR không tuyến tính không được áp dụng. Cụ thể là, trong trường hợp này, khoảng AMVR tuyến tính được áp dụng để biểu thị MVD. Mặt khác, khi giá trị của `non_linear_amvr_range_flag` là 1, khoảng AMVR không tuyến tính có thể được áp dụng. Cụ thể là, thiết bị giải mã có thể xác định việc khoảng AMVR không tuyến tính có được áp dụng hay không dựa vào `non_linear_amvr_range_flag`.

Về cơ bản, thiết bị giải mã biết thông tin khoảng không tuyến tính và thông tin giá trị đại diện khoảng, và khi giá trị của `non-linearamvr_range_flag` là 1, khoảng không tuyến tính mặc định định trước có thể được sử dụng khi không

có thông tin bổ sung khác.

Ví dụ 3

Hơn thế nữa, phương án thứ ba của sáng chế cung cấp phương pháp có thể chỉ báo khoảng AMVR không tuyến tính cụ thể được cung cấp ngoài phương án thứ hai của sáng chế được mô tả trên đây.

Như được mô tả trên đây, khi giá trị của `amvr_enable_flag` là 1, có thể chỉ báo việc khoảng AMVR tuyến tính được sử dụng hoặc khoảng AMVR không tuyến tính được sử dụng qua `non_linear_amvr_range_flag`.

Trong phương án thứ hai được mô tả trên đây, giá trị mặc định của khoảng không tuyến tính được thiết lập trước trong thiết bị giải mã, và thiết bị giải mã có thể sử dụng khoảng không tuyến tính mặc định khi giá trị của `non_linear_amvr_range_flag` là 1. Tuy nhiên, trong phương án này, các tham số liên quan đến khoảng không tuyến tính AMVR như được thể hiện trong Bảng 4 sau có thể được truyền để thông báo một cách thích hợp cho thiết bị giải mã về khoảng AMVR không tuyến tính.

Bảng 4

<code>sps temporal mvp enabled flag</code>
<code>strong intra smoothing enabled flag</code>
<code>sps amvr enable flag</code>
<code>if(sps amvr enable flag)</code>
<code>non_linear amvr_range_flag</code>
<code>if(amvr enable flag && non_linear amvr_range flag)</code>
<code>non_linear amvr_range parameters()</code>
<code>vui parameters_present_flag</code>
<code>if(vui parameters present flag)</code>
<code>vui parameters()</code>
<code>sps extension flag</code>
<code>if(sps extension flag)</code>
<code>while(more_rbsp_data())</code>
<code>sps extension data flag</code>
<code>rbp trailing bits()</code>
<code>}</code>

Ở đây, `sps_amvr_enable_flag` chỉ báo việc chế độ AMVR có khả dụng hay không. `sps_amvr_enable_flag` có thể được sử dụng thay thế lẫn nhau với `amvr_enable_flag`. `non_linear_amvr_range_flag` chỉ báo việc khoảng AMVR không tuyến tính có được áp dụng hay không. `non_linear_amvr_range_flag` có

thể được truyền/thu và phân tách khi giá trị của `sps_amvr_enable_flag` là 1.

Trong khi đó, cú pháp `non_linear_amvr_range_paramter` có thể được truyền/thu và phân tách/gọi khi giá trị của `non_linear_amvr_range_flag` là 1 (cụ thể là, khi giá trị của `sps_amvr_enable_flag` là 1 và giá trị của `non_linear_amvr_range_flag` là 1). Cú pháp `non_linear_amvr_range_parameter` biểu thị các tham số liên quan đến khoảng không tuyến tính AMVR. Các tham số liên quan đến khoảng không tuyến tính AMVR có thể được chia thành, ví dụ, ba khoảng để chỉ báo các giá trị khoảng theo từng vị trí và các giá trị đại diện của các khoảng. Các tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm, ví dụ, các phần tử cú pháp sau.

Bảng 5

<code>non_linear_amvr_range_parameters() {</code>
<code>num_non_linear_range_table_candidate</code>
<code>for(rangeTableIdx=0; rangeTableIdx<</code>
<code>num_non_linear_range_table_candidate;rangeTableIdx++){</code>
<code>first_range_value</code>
<code>first_range_representative_value</code>
<code>second_range_value</code>
<code>second_range_representative_value</code>
<code>third_range_value</code>
<code>third_range_representative_value</code>
<code>}</code>
<code>}</code>

Ở đây, `num_non_linear_range_table_candidate` biểu thị số lượng ứng viên bảng khoảng không tuyến tính. Từng ứng viên bảng khoảng không tuyến tính có thể bao gồm `first_range_representative_value`, `second_range_value`, `second_range_representative_value`, `third_range_value`, và `third_range_representative_value`. Thiết bị mã hóa có thể chỉ báo một trong số ứng viên bảng khoảng không tuyến tính qua chỉ số bảng khoảng. Mặt khác, nếu một hoặc số lượng cố định của các bảng khoảng không tuyến tính được sử dụng, `num_non_linear_range_table_candidate` có thể được bỏ qua.

`first_range_value`, `second_range_value`, và `third_range_value` biểu thị giá trị khoảng thứ nhất, giá trị khoảng thứ hai, và giá trị khoảng thứ ba có tâm tại 0 tương ứng. Ở đây, khoảng thứ nhất là khoảng bao gồm 0. Từng

first_range_value, second_range_value, và third_range_value có thể biểu thị giá trị 0 hoặc lớn hơn.

Ngoài ra, từng first_range_representative_value, second_range_representative_value và third_range_representative_value biểu thị các giá trị đại diện của khoảng liên quan.

Ví dụ, khi giá trị của first_range_value là a và giá trị của first_range_representative_value là 0, khoảng (bao gồm 0) của $-a \leq \text{MVD} < a$ trở thành khoảng thứ nhất, và giá trị đại diện của khoảng thứ nhất là 0. Ngoài ra, khi giá trị của second_range_value là b và giá trị của second_range_representative_value là m, $-b \leq \text{MVD} < -a$ và $a \leq \text{MVD} < b$ trở thành các khoảng thứ hai, và các giá trị đại diện của khoảng thứ hai trở thành -m và m tương ứng. Khi giá trị của third_range_value là c và giá trị của third_range_representative_value là n, $-c \leq \text{MVD} < -b$ và $b \leq \text{MVD} < c$ là các khoảng thứ ba, và các giá trị đại diện của khoảng thứ ba là -n và n tương ứng.

FIG.9 là hình vẽ minh họa ví dụ tạo AMVR không tuyến tính bằng phân tách cú pháp tham số khoảng AVMR không tuyến tính theo một phương án của sáng chế. Trên FIG.9, miền số dương được thể hiện chủ yếu và được mô tả để tránh mô tả lặp lại.

Tham chiếu FIG.9A, giá trị của first_range_value là 6, giá trị của first_range_representative_value là 0, giá trị của second_range_value là 10, giá trị của second_range_representative_value là 8, giá trị của third_range_value là 14, và giá trị của third_range_representative_value là 12. Trong trường hợp này, khoảng thứ nhất là $-6 \leq \text{MVD} < 6$, và giá trị đại diện cho khoảng thứ nhất là 0. Khoảng thứ hai của miền số dương là $6 \leq \text{MVD} < 10$, và giá trị đại diện cho khoảng thứ hai là 8. Khoảng thứ ba của miền số dương là $10 \leq \text{MVD} < 14$, và giá trị đại diện cho khoảng thứ ba là 12. Các giá trị đại diện cho các khoảng từ thứ nhất đến thứ ba có thể được giảm để tương ứng với 0, 1, và 2. Trong trường hợp này, khoảng AMVR không tuyến tính có thể thu được như được thể hiện trên

FIG.9A. Mặt khác, các phần được biểu thị bằng “v.v.” (ví dụ, các khoảng 13 đến 17) chỉ báo các khoảng dư ngoài các khoảng từ thứ nhất đến thứ ba, và các khoảng dư có thể được thiết lập theo tiêu chuẩn định trước.

Tham chiếu FIG.9B, giá trị của `first_range_value` là 4, giá trị của `first_range_representative_value` là 0, giá trị của `second_range_value` là 12, giá trị của `second_range_representative_value` là 8, giá trị của `third_range_value` là 16, và giá trị của `third_range_representative_value` là 14. Trong trường hợp này, khoảng thứ nhất là $-4 \leq \text{MVD} < 4$, và giá trị đại diện cho khoảng thứ nhất là 0. Khoảng thứ hai của miền số dương là $4 \leq \text{MVD} < 12$, và giá trị đại diện cho khoảng thứ hai là 8. Khoảng thứ ba của miền dương là $12 \leq \text{MVD} < 16$, và giá trị đại diện cho khoảng thứ ba là 14. Các giá trị đại diện cho các khoảng từ thứ nhất đến thứ ba có thể được giảm để tương ứng với 0, 1, và 2. Trong trường hợp này, khoảng AMVR không tuyến tính có thể thu được như được thể hiện trên FIG.9A.

Khi có các ứng viên bảng khoảng không tuyến tính như được mô tả trên đây, thiết bị mã hóa có thể chỉ báo một trong số các ứng viên dựa vào chỉ số bảng khoảng, và thiết bị giải mã có thể thu và phân tách chỉ số bảng khoảng để chọn một trong số các ứng viên. Ví dụ, chỉ số bảng khoảng có thể được truyền qua cú pháp sau.

Bảng 6

<code>slice segment header() {</code>
<code> first slice segment in pic flag</code>
<code> if(nal unit type >= BLA_W_LP && nal unit type <= RSV_IRAP_VCL23)</code>
<code> no output of prior pics flag</code>
<code> slice pic parameter set id</code>
<code> ...</code>
<code> delta_poc_msb_present_flag[i]</code>
<code> if(delta_poc_msb_present_flag[i])</code>
<code> delta_poc_msb_cycle_lt[i]</code>
<code> }</code>
<code> }</code>
<code> if(sps_temporal_mvp_enabled_flag)</code>
<code> slice_temporal_mvp_enabled_flag</code>
<code> if(sps_non_linear_amvr_range_flag)</code>
<code> slice_non_linear_amvr_range_table_idx</code>
<code> }</code>
<code> ...</code>
<code>}</code>

Ở đây, `slice_non_linear_amvr_range_table_idx` có thể biểu thị chỉ số

bảng khoảng được mô tả trên đây. `slice_non_linear_ambr_range_table_idx` có thể được truyền/thu và phân tách khi giá trị của `non_linear_amvr_range_flag` (hoặc `sps_non_linear_amvr_flag`) được mô tả trong Bảng 4 là 1.

`slice_non_linear_ambr_range_table_idx` có thể được truyền tại đầu đoạn đầu lát. Cụ thể là, phần tử cú pháp `slice_non_linear_ambr_range_table_idx` có thể được truyền qua cú pháp đoạn đầu của phần lát cho lát bao gồm khối hiện thời. Ở đây, khối hiện thời có thể là PU hoặc PB.

Thiết bị giải mã có thể thu MVD dựa vào các giá trị khoảng cho ứng viên bảng khoảng không tuyến tính tương ứng với chỉ số được chỉ báo bằng `slice_non_linear_ambr_range_table_idx`.

Ví dụ 4

Ngoài ra, trong phương án thứ tư của sáng chế, để làm phần mở rộng của phương án thứ ba được mô tả trên đây, đơn vị biểu thị của MVD có thể được thay đổi thích ứng theo khoảng. Trong trường hợp này, đơn vị biểu thị của MVD có thể được thay đổi thích ứng thành đơn vị điểm nguyên, đơn vị nửa điểm, hoặc đơn vị.

FIG.10 là hình vẽ minh họa ví dụ về phương pháp AMVR không tuyến tính có độ chính xác MVD thích ứng.

Tham chiếu FIG.10, khoảng $-8 \leq \text{MVD} < 8$ có độ chính xác điểm nguyên, các khoảng $-18 \leq \text{MVD} < -8$ và khoảng $8 \leq \text{MVD} < 18$ có độ chính xác nửa điểm, và khoảng $-24 \leq \text{MVD} < -18$ và khoảng $18 \leq \text{MVD} < 24$ được thiết lập có độ chính xác phần tư điểm. Trong trường hợp này, thiết bị mã hóa có thể thu MVD với độ chính xác điểm nguyên trong khoảng $-8 \leq \text{MVD} < 8$, có thể thu MVD với độ chính xác nửa điểm trong khoảng $-18 \leq \text{MVD} < -8$ và $8 \leq \text{MVD} < 18$, và có thể thu MVD với độ chính xác phần tư điểm trong khoảng $-24 \leq \text{MVD} < -18$ và khoảng $18 \leq \text{MVD} < 24$. Do vậy, thiết bị mã hóa có thể đánh giá và biểu thị theo đơn vị điểm nguyên khi MV được dự báo theo độ méo và quan hệ tốc độ trong xử lý tối ưu RD đến khoảng mà sự tương quan là cao, để giảm lượng bit của MVD.

Ngược lại, khi MV được dự báo thuộc về khoảng mà sự tương quan là thấp, thiết bị mã hóa có thể đánh giá và biểu thị theo đơn vị nhỏ hơn (ví dụ, đơn vị nửa điểm hoặc đơn vị phần tư điểm), để giảm hơn nữa độ méo.

FIG.11 là hình vẽ minh họa phương pháp thu MVD theo độ chính xác MVD thích ứng theo một phương án của sáng chế. Phương pháp của FIG.11 có thể được thực hiện bằng thiết bị giải mã.

Tham chiếu FIG.11, thiết bị giải mã phân tách MVD (S1100). Ở đây, phân tách MVD bao gồm thu và phân tách cú pháp liên quan đến MVD.

Thiết bị giải mã kiểm tra việc MVD có độ chính xác điểm nguyên hay không dựa vào khoảng mà giá trị được mã hóa của MVD thuộc về (S1110).

Khi MVD có độ chính xác điểm nguyên ở bước S1110, thiết bị giải mã thu giá trị đại diện cho giá trị được mã hóa của MVD có tính đến độ chính xác điểm nguyên (S1120)

Nếu MVD không có độ chính xác điểm nguyên ở bước S1110, thì thiết bị giải mã kiểm tra việc MVD dựa vào khoảng mà giá trị được mã hóa của MVD thuộc về có độ chính xác nửa điểm (S1130)

Nếu MVD có độ chính xác nửa điểm ở bước S1130, thì thiết bị giải mã thu giá trị đại diện cho giá trị được mã hóa của MVD có tính đến độ chính xác nửa điểm (S1140).

Nếu MVD không có độ chính xác nửa điểm ở bước S1130, thì tương ứng với trường hợp mà MVD có độ chính xác phần tư điểm, và thiết bị giải mã thu giá trị đại diện cho giá trị được mã hóa của MVD có tính đến độ chính xác phần tư điểm (S1150).

Phương pháp giải mã được mô tả trên đây có thể được thể hiện, ví dụ, giống như các hình vẽ sau.

FIG.12 đến FIG.14 là các hình vẽ minh họa phương pháp giải mã các giá trị MVD theo độ chính xác điểm nguyên, độ chính xác nửa điểm, hoặc độ chính xác phần tư điểm theo một phương án của sáng chế.

Trên FIG.12 đến FIG.14, khoảng $-8 \leq \text{MVD} < 8$ được thiết lập có độ chính xác điểm nguyên, khoảng $8 \leq \text{MVD} < 18$ được thiết lập có độ chính xác nửa điểm, và khoảng $18 \leq \text{MVD} < 24$ được thiết lập có độ chính xác phần tư điểm. Ngoài ra, trên FIG.12 đến FIG.14, giả định là AMVR không tuyến tính được áp dụng.

Tham chiếu FIG.12, khi giá trị gốc của MVD là 5, giá trị 5 được biểu thị bằng giá trị đại diện 0 sẽ được mã hóa thành giá trị 0, và thiết bị giải mã giải mã giá trị được mã hóa 0 để thu được giá trị 0. Thiết bị giải mã có thể thu MV dùng cho khôi hiện thời bằng cách cộng MVD của 0 vào MVP.

Ngoài ra, tham chiếu FIG.13, khi giá trị gốc của MVD là 11, giá trị 11 được biểu thị bằng giá trị đại diện 10, và 10 được giảm để được mã hóa thành giá trị 1. Ngoài ra, thiết bị giải mã giải mã và tăng giá trị được mã hóa 1 để thu được 10 làm giá trị MVD.

Tham chiếu FIG.14, khi giá trị của MVD là 21, giá trị đại diện của giá trị 21 là 21, và nếu khoảng là 1 đến 19, giá trị 21 được mã hóa thành 6. Dựa vào 0 làm điểm tham chiếu, 0 đến 7 có thể được mã hóa thành giá trị 0, 8 đến 13 có thể được mã hóa thành giá trị 1, và 14 đến 17 có thể được mã hóa thành giá trị 2. Ngoài ra, vì khoảng được thiết lập là 1 từ 19, theo tuần tự, 18 có thể được mã hóa thành giá trị 3, 19 có thể được mã hóa thành giá trị 4, 20 có thể được mã hóa thành giá trị 5, và 21 có thể được mã hóa thành giá trị 6. Trong trường hợp này, thiết bị giải mã có thể giải mã giá trị được mã hóa 6, và có thể thu giá trị của 21 làm giá trị MVD tương ứng với giá trị được mã hóa 6.

Theo cách khác, khi khoảng được thiết lập là 4 làm mặc định sau 19, giá trị đại diện của 21 có thể là 20, và giá trị 20 có thể được mã hóa thành giá trị 3. Trong trường hợp này, thiết bị giải mã có thể giải mã giá trị 3, và thu giá trị của 20 làm giá trị MVD tương ứng với giá trị 3.

Thông tin khoảng cho độ chính xác MVD thích ứng như được mô tả trên đây có thể được chỉ báo dựa vào cú pháp SPS và cú pháp đoạn đầu lát như được mô tả trong phương án thứ ba. Trong trường hợp này, cú pháp SPS có thể bao

gồm các phần tử cú pháp liên quan đến số lượng ứng viên bảng khoảng độ chính xác MVD, và có thể bao gồm thông tin về ít nhất một trong số khoảng độ chính xác điểm nguyên và khoảng độ chính xác nửa điểm cho từng ứng viên bảng khoảng độ chính xác MVD. Chỉ số bảng khoảng độ chính xác chỉ báo một trong số các ứng viên bảng khoảng độ chính xác MVD có thể được chỉ báo qua cú pháp đoạn đầu lát.

Theo cách khác, cú pháp cho các tham số khoảng AMVR không tuyến tính có thể bao gồm thông tin về khoảng độ chính xác điểm nguyên và khoảng độ chính xác nửa điểm như được thể hiện trong Bảng 7 sau.

Bảng 7

<code>non_linear_amvr_range_parameters() {</code>
<code> num_non_linear_range_table_candidate</code>
<code> for(rangeTableIdx=0; rangeTableIdx<</code>
<code>num_non_linear_range_table_candidate;rangeTableIdx++){</code>
<code> first_range_value</code>
<code> first_range_representative_value</code>
<code> second_range_value</code>
<code> second_range_representative_value</code>
<code> third_range_value</code>
<code> third_range_representative_value</code>
<code> integer_pel_precision_range</code>
<code> half_pel_precision_range</code>
<code> }</code>
<code>}</code>

Tham chiếu Bảng 7, `num_non_linear_range_table_candidate` biểu thị số lượng ứng viên bảng khoảng không tuyến tính. Từng ứng viên bảng khoảng không tuyến tính có thể bao gồm các phần tử cú pháp `integer_pel_precision_range` và `half_pel_precision_range`.

Ở đây, `integer_pel_precision_range` biểu thị khoảng độ chính xác điểm nguyên, và `half_pel_precision_range` biểu thị khoảng độ chính xác nửa điểm. Khoảng độ chính xác điểm nguyên và khoảng độ chính xác nửa điểm có thể được sử dụng kết hợp với khoảng thứ nhất đến khoảng thứ ba dựa vào `first_range_value`, `first_range_representative_value`, `second_range_value`, `second_range_representative_value`, `third_range_value`, và `third_range_representative_value` của Bảng 7.

Ví dụ, các giá trị biểu thị từng khoảng độ chính xác có thể được giảm và được truyền. Trong trường hợp này, giá trị được chỉ báo bằng `integer_pel_precision_range` có thể được giảm xuống 4 để được truyền, và giá trị được chỉ báo bằng `half_pel_precision_range` có thể được truyền sau khi giá trị chênh lệch với giá trị được chỉ báo bằng `integer_pel_precision_range` giảm xuống 2. Ví dụ, trên FIG.10, giá trị 8 mà là giá trị được chỉ báo bằng `integer_pel_precision_range` được giảm xuống 4 và được truyền làm giá trị của 2, và giá trị 5 được giảm xuống 2 từ 10 mà là giá trị chênh lệch giữa giá trị 8 và giá trị 18 được biểu thị bằng `integer_pel_precision_range` được truyền. Thiết bị giải mã có thể giải mã từng khoảng độ chính xác qua thủ tục ngược với thủ tục nêu trên đây.

FIG.15 là hình vẽ giản lược minh họa ví dụ về phương pháp mã hóa video theo một phương án của sáng chế. Phương pháp được bộc lộ trên FIG.15 có thể được thực hiện bằng thiết bị mã hóa.

Tham chiếu FIG.15, thiết bị mã hóa tạo cờ cho phép AMVR (S1500). Thiết bị mã hóa xác định việc có thực hiện AMVR để dự báo giữa các khung khối hiện thời hay không. Thiết bị mã hóa có thể xác định thực hiện AMVR khi xác định được là việc thực hiện AMVR là tốt hơn để tối ưu RD, và có thể thiết lập giá trị của cờ cho phép AMVR là 1.

Thiết bị mã hóa thu MVD cho khối hiện thời (S1510). MVD có thể là MVD của đơn vị phần tư điểm. Theo cách khác, khi độ chính xác MVD thích ứng được áp dụng, MVD có thể là đơn vị điểm nguyên hoặc đơn vị nửa điểm theo khoảng độ chính xác MVD.

Thiết bị mã hóa có thể thu MV cho khối hiện thời qua việc đánh giá chuyển động. Thiết bị mã hóa có thể thu MVP cho khối hiện thời dựa vào các khối lân cận không gian và khối lân cận thời gian (hoặc các khối tương ứng thời gian) của khối hiện thời. Thiết bị mã hóa có thể thu MVP cho khối hiện thời dựa vào MV và MVP. Ở đây, MVP có thể là MVP của đơn vị phần tư điểm. Theo

cách khác, MVP có thể là MVP của đơn vị điểm nguyên, thu được bằng cách rounding off MVP thời gian của đơn vị phần tư điểm thành đơn vị điểm nguyên.

Thiết bị mã hóa thu giá trị đại diện MVD dùng cho khoảng MVD bao gồm giá trị MVD trong số các khoảng MVD (S1520), và tạo MVD được mã hóa tương ứng với giá trị đại diện MVD (S1530).

Ví dụ, các khoảng MVD có thể là các khoảng MVD tuyến tính có khoảng bằng nhau. Trong trường hợp này, giá trị đại diện MVD có thể là giá trị thu được bằng cách làm tròn xuống giá trị MVD cộng thêm 2 thành bội số của 4. Trong trường hợp này, MVD được mã hóa có thể là giá trị thu được bằng cách giảm giá trị đại diện MVD xuống 4.

Ví dụ khác, các khoảng MVD có thể là các khoảng MVD không tuyến tính có khoảng không bằng nhau. Trong trường hợp này, các khoảng MVD trong vùng giữa các khoảng MVD không tuyến tính có thể có khoảng tương đối rộng. Trong trường hợp này, các khoảng MVD tương ứng với các giá trị MVD được mã hóa -1, 0, và 1 trong số các khoảng MVD không tuyến tính có thể có khoảng tương đối rộng hơn so với các khoảng MVD dư. Khi giá trị MVD được mã hóa là một trong số -1, 0, và 1, giá trị đại diện MVD có thể là giá trị thu được bằng cách làm tròn xuống giá trị MVD cộng thêm 4 thành bội số của 8. Trong trường hợp này, giá trị MVD được mã hóa có thể là giá trị thu được bằng cách giảm giá trị đại diện MVD xuống 8.

Thiết bị mã hóa xuất cờ cho phép AMVR và MVD được mã hóa qua dòng bit (S1540). Ví dụ, cờ cho phép AMVR có thể được xuất qua dòng bit tại mức SPS. Ngoài ra, MVD được mã hóa có thể được xuất qua dòng bit tại mức PU.

Dòng bit đầu ra có thể được truyền đến thiết bị giải mã qua mạng hoặc vật lưu trữ.

Mặt khác, mặc dù không được thể hiện, thiết bị mã hóa có thể tạo cờ khoảng không tuyến tính AMVR chỉ báo việc các khoảng MVD có khoảng

không bằng nhau hay không. Giá trị 0 của cờ khoảng không tuyến tính AMVR chỉ báo là các khoảng MVD có khoảng bằng nhau và giá trị 1 của cờ khoảng không tuyến tính AMVR có thể chỉ báo là các khoảng MVD có khoảng không bằng nhau. Thiết bị mã hóa có thể xuất cờ AMVR không tuyến tính được tạo ra qua dòng bit.

Khi giá trị của cờ khoảng không tuyến tính AMVR là 1, thiết bị mã hóa có thể tạo và xuất các tham số liên quan đến AMVR không tuyến tính qua dòng bit. Các tham số liên quan đến AMVR không tuyến tính có thể bao gồm thông tin được mô tả trong Bảng 5 và 7 trên đây. Ví dụ, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về kích thước của ba khoảng giữa MVD và các giá trị đại diện của các khoảng tương ứng. Ngoài ra, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về số lượng ứng viên bảng khoảng không tuyến tính. Trong trường hợp này, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về kích thước của ba khoảng giữa MVD tương ứng với từng ứng viên bảng khoảng không tuyến tính và các giá trị đại diện của các khoảng tương ứng. Thiết bị mã hóa có thể tạo chỉ số bảng khoảng chỉ báo một trong số các ứng viên bảng khoảng không tuyến tính, và có thể xuất chỉ số bảng khoảng qua dòng bit. Ví dụ, chỉ số bảng khoảng có thể được xuất qua dòng bit tại mức đoạn đầu lát.

Mặt khác, đơn vị biểu thị của MVD có thể được thay đổi thích ứng theo khoảng sử dụng khoảng độ chính xác MVD thích ứng. Trong trường hợp này, thiết bị mã hóa có thể tạo thông tin độ chính xác MVD chỉ báo việc MVD có độ chính xác điểm nguyên hoặc độ chính xác nửa điểm, và có thể xuất thông tin độ chính xác MVD qua dòng bit. Thông tin độ chính xác MVD có thể bao gồm ít nhất một trong số phần tử cú pháp `integer_pel_precision_range` và phần tử cú pháp `half_pel_precision_range` được mô tả trên đây.

FIG.16 là hình vẽ giản lược minh họa ví dụ về phương pháp dự báo giữa các khung theo một phương án của sáng chế. Phương pháp được bộc lộ trên

FIG.16 có thể được thực hiện bằng thiết bị giải mã.

Tham chiếu FIG.16, thiết bị giải mã phân tách và thu cờ cho phép AMVR từ dòng bit thu được từ thiết bị mã hóa (S1600). Thiết bị giải mã có thể thu dòng bit qua mạng hoặc vật lưu trữ. Cờ cho phép AMVR có thể được phân tách và thu được, ví dụ, tại mức SPS.

Thiết bị giải mã phân tách và thu MVD được mã hóa từ dòng bit (S1610). Ví dụ, MVD được mã hóa có thể được phân tách và thu được tại mức PU.

Khi giá trị của cờ cho phép AMVR là 1, thiết bị giải mã thu giá trị của MVD đại diện tương ứng với giá trị MVD được mã hóa (S1620). Giá trị đại diện MVD là giá trị đại diện dùng cho khoảng MVD bao gồm giá trị của MVD gốc trong số các khoảng MVD.

Ví dụ, các khoảng MVD có thể là các khoảng MVD tuyến tính có khoảng bằng nhau, và giá trị MVD được mã hóa có thể là giá trị thu được bằng cách tăng giá trị đại diện MVD đến 4.

Ví dụ khác, các khoảng MVD có thể là các khoảng MVD không tuyến tính có khoảng không bằng nhau. Trong trường hợp này, các khoảng MVD trong vùng giữa trong số các khoảng MVD không tuyến tính có thể có khoảng tương đối rộng. Trong trường hợp này, các khoảng MVD tương ứng với các giá trị MVD được mã hóa -1, 0, và 1 trong số các khoảng MVD không tuyến tính có thể có khoảng tương đối rộng hơn so với các khoảng MVD dư. Khi giá trị MVD được mã hóa là một trong số -1, 0, và 1, giá trị đại diện MVD có thể là giá trị thu được bằng cách làm tròn xuống giá trị MVD cộng thêm 4 thành bội số của 8. Trong trường hợp này, giá trị đại diện MVD có thể là giá trị thu được bằng cách tăng giá trị MVD được mã hóa đến 8.

MVD gốc có thể là MVD của đơn vị phân tử điểm. Theo cách khác, khi độ chính xác MVD thích ứng được áp dụng, MVD gốc có thể là đơn vị điểm nguyên hoặc đơn vị nửa điểm, theo khoảng độ chính xác MVD.

Thiết bị giải mã thu MVP dùng cho khối hiện thời dựa vào các khối lân

cận không gian và khối lân cận thời gian (hoặc khối tương ứng thời gian) của khối hiện thời (S1630). Ở đây, MVP có thể là MVP của đơn vị phân tử điểm. Theo cách khác, MVP có thể là MVP của đơn vị điểm nguyên, thu được bằng cách làm tròn MVP thời gian của đơn vị phân tử điểm thành đơn vị điểm nguyên.

Thiết bị giải mã thu MV dùng cho khối hiện thời dựa vào MVP và MVD đại diện (S1640). Thiết bị giải mã có thể thu MV dùng cho khối hiện thời bằng cách cộng MVP và MVD đại diện.

Thiết bị giải mã thực hiện dự báo giữa các khung dựa vào MV, và tạo mẫu dự báo (hoặc mảng mẫu dự báo) đối với khối hiện thời (S1650). Thiết bị giải mã có thể tạo mẫu dự báo (hoặc mảng mẫu dự báo) đối với khối hiện thời dựa vào mẫu được khôi phục (hoặc mảng mẫu được khôi phục) trong khối tham chiếu tại vị trí tương đối được chỉ báo bằng MV trên hình ảnh tham chiếu. Thiết bị giải mã có thể thu các hệ số biến đổi liên quan đến tín hiệu dư từ dòng bit. Thiết bị giải mã có thể đảo các hệ số biến đổi, và có thể thu mẫu dư (hoặc mảng mẫu dư) đối với khối hiện thời từ các hệ số biến đổi. Thiết bị giải mã có thể tạo mẫu được khôi phục và hình ảnh được khôi phục dựa vào các mẫu dự báo và các mẫu dư.

Mặt khác, mặc dù không được thể hiện, thiết bị giải mã có thể thu qua dòng bit cờ khoảng không tuyến tính AMVR chỉ báo việc các khoảng MVD có khoảng không bằng nhau hay không. Giá trị 0 của cờ khoảng không tuyến tính AMVR chỉ báo là các khoảng MVD có khoảng bằng nhau và giá trị 1 của cờ khoảng không tuyến tính AMVR có thể chỉ báo là các khoảng MVD có khoảng không bằng nhau. Các tham số liên quan đến AMVR không tuyến tính có thể bao gồm thông tin được mô tả trong Bảng 5 và 7 trên đây. Ví dụ, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về kích thước của ba khoảng giữa MVD và các giá trị đại diện của các khoảng tương ứng. Ngoài ra, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về số lượng ứng viên bảng khoảng không tuyến tính. Trong

trường hợp này, tham số liên quan đến khoảng không tuyến tính AMVR có thể bao gồm thông tin về kích thước của ba khoảng giữa MVD tương ứng với từng ứng viên bảng khoảng không tuyến tính và các giá trị đại diện của các khoảng tương ứng. Thiết bị giải mã có thể thu giá trị đại diện MVD dựa vào tham số liên quan đến khoảng không tuyến tính AMVR và giá trị MVD được mã hóa. Trong trường hợp này, thiết bị giải mã có thể thu chỉ số bảng khoảng từ dòng bit, và có thể chọn một trong số các ứng viên bảng khoảng không tuyến tính dựa vào chỉ số bảng khoảng. Thiết bị giải mã có thể biết kích thước của các khoảng MVD và các giá trị đại diện dựa vào ứng viên được chọn, và thu giá trị đại diện MVD tương ứng với giá trị MVD được mã hóa dựa vào kích thước của các khoảng MVD và các giá trị đại diện.

Mặt khác, việc sử dụng khoảng độ chính xác MVD thích ứng, đơn vị biểu thị của MVD gốc có thể được thay đổi thích ứng theo khoảng độ chính xác. Trong trường hợp này, thiết bị giải mã có thể thu, qua dòng bit, thông tin độ chính xác MVD chỉ báo việc MVD gốc có độ chính xác điểm nguyên hoặc độ chính xác nửa điểm, và có thể thu khoảng MVD và giá trị đại diện MVD tương ứng với giá trị MVD được mã hóa dựa vào thông tin độ chính xác MVD. Thông tin độ chính xác MVD có thể bao gồm ít nhất một trong số phần tử cú pháp `integer_pel_precision_range` và phần tử cú pháp `half_pel_precision_range` được mô tả trên đây.

Theo một phương án của sáng chế được mô tả trên đây, dự báo giữa các khung dùng cho khôi hiện thời có thể được thực hiện một cách hiệu quả trong khi sử dụng thông tin bổ sung rất ít. Ngoài ra, theo một phương án của sáng chế, lượng bit được cấp phát cho MVD có thể được giảm và hiệu suất mã hóa tổng có thể được tăng.

Phần mô tả trên đây chỉ minh họa ý tưởng kỹ thuật của sáng chế. Do đó, người có hiểu biết trung bình trong kỹ thuật có thể thực hiện các cải biến và thay đổi khác nhau đối với phần mô tả trên đây mà không nằm ngoài các đặc trưng

chính của sáng chế. Do vậy, các phương án được bộc lộ ở đây chỉ nhằm minh họa, không giới hạn sáng chế. Phạm vi của sáng chế không bị giới hạn bởi các phương án này. Phạm vi bảo hộ của sáng chế nên được hiểu theo các yêu cầu bảo hộ sau.

Khi các phương án của sáng chế được thực hiện trong phần mềm, phương pháp được mô tả trên đây có thể được thực hiện bằng các môđun (các quá trình xử lý, chức năng, v.v.) mà thực hiện các chức năng được mô tả trên đây. Các môđun này có thể được lưu trữ trong bộ nhớ và được thực hiện bằng bộ xử lý. Bộ nhớ có thể là bộ xử lý trong hoặc ngoài, và bộ nhớ có thể được ghép nối với bộ xử lý bằng cách sử dụng phương tiện đã biết khác. Bộ xử lý có thể gồm ASIC (application-specific integrated circuit – mạch tích hợp dành riêng cho ứng dụng), các chipset khác, mạch logic và/hoặc thiết bị xử lý dữ liệu. Bộ nhớ có thể bao gồm ROM (read-only memory), RAM (random access memory), bộ nhớ tia chớp, thẻ nhớ, vật lưu trữ, và/hoặc thiết bị lưu trữ khác.

YÊU CẦU BẢO HỘ

1. Phương pháp mã hóa ảnh được thực hiện bằng thiết bị mã hóa, phương pháp này bao gồm các bước:

tạo cờ cho phép khoảng vector chuyển động thích ứng (AMVR - Adaptive Motion Vector Range);

thu chênh lệch vector chuyển động (MVD - Motion Vector Difference) dùng cho khối hiện thời;

thu giá trị đại diện MVD dùng cho khoảng MVD gồm giá trị của MVD trong số các khoảng MVD;

tạo MVD được mã hóa tương ứng với giá trị đại diện MVD; và

xuất cờ cho phép AMVR và MVD được mã hóa qua dòng bit,

trong đó phương pháp này còn bao gồm các bước:

tạo cờ khoảng AMVR không tuyến tính mà chỉ báo việc các khoảng MVD có khoảng không bằng nhau hay không; và

xuất cờ khoảng AMVR không tuyến tính qua dòng bit.

2. Phương pháp theo điểm 1, trong đó:

các khoảng MVD là các khoảng MVD tuyến tính có kích thước khoảng bằng nhau,

giá trị đại diện MVD là giá trị thu được bằng cách làm tròn xuống giá trị MVD cộng thêm 2 thành bội số của 4, và

giá trị MVD được mã hóa là giá trị thu được dựa trên sự lấy giá trị đại diện MVD chia cho 4.

3. Phương pháp theo điểm 1, trong đó các khoảng MVD là các khoảng MVD không tuyến tính có các kích thước khoảng không bằng nhau, và mỗi kích thước trong số các khoảng MVD thứ nhất tương ứng với các giá trị MVD được mã hóa -1, 0, và 1 trong số các khoảng MVD không tuyến tính là lớn hơn so với từng kích thước của các khoảng MVD thứ hai trong số các khoảng MVD không tuyến

tính,

trong đó các khoảng MVD thứ hai là phần còn lại của các khoảng MVD không tuyến tính không kể các khoảng MVD thứ nhất.

4. Phương pháp theo điểm 1, trong đó:

các khoảng MVD là các khoảng MVD không tuyến tính có các kích thước khoảng không bằng nhau, và

khi giá trị MVD được mã hóa là một trong số -1, 0, và 1,

giá trị đại diện MVD là giá trị thu được bằng cách làm tròn xuống giá trị MVD cộng thêm 4 thành bội số của 8, và

giá trị MVD được mã hóa là giá trị thu được dựa vào sự lấy giá trị đại diện MVD chia cho 8.

5. Phương pháp theo điểm 1, trong đó phương pháp này còn bao gồm bước xuất tham số liên quan đến khoảng AMVR tuyến tính qua dòng bit khi giá trị của cờ khoảng AMVR không tuyến tính là 1,

trong đó tham số liên quan đến khoảng AMVR không tuyến tính bao gồm thông tin về các kích thước của ba khoảng giữa MVD trong số các khoảng MVD và các giá trị đại diện của các khoảng tương ứng.

6. Phương pháp theo điểm 5, trong đó tham số liên quan đến khoảng AMVR không tuyến tính bao gồm thông tin về số lượng ứng viên bảng khoảng không tuyến tính, và

tham số liên quan đến khoảng AMVR không tuyến tính bao gồm thông tin về kích thước của ba khoảng giữa MVD và các giá trị đại diện của các khoảng tương ứng theo từng ứng viên bảng khoảng không tuyến tính.

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

tạo chỉ số bảng khoảng chỉ báo một trong số các ứng viên bảng khoảng không tuyến tính; và

xuất chỉ số bảng khoảng qua dòng bit.

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

tạo thông tin độ chính xác MVD chỉ báo việc MVD có độ chính xác điểm nguyên hoặc độ chính xác nửa điểm hay không;

xuất thông tin độ chính xác MVD qua dòng bit.

9. Phương pháp dự báo giữa các khung được thực hiện bằng thiết bị giải mã, phương pháp này bao gồm các bước:

thu cờ cho phép khoảng vector chuyển động thích ứng (AMVR - Adaptive Motion Vector Range) từ dòng bit;

thu chênh lệch vector chuyển động (MVD - Motion Vector Difference) được mã hóa từ dòng bit;

thu giá trị của MVD đại diện tương ứng với giá trị của MVD được mã hóa khi giá trị của cờ cho phép AMVR là 1;

thu dự báo vector chuyển động (MVP - Motion Vector Predictor) dùng cho khối hiện thời dựa vào khối lân cận của khối hiện thời;

thu vector chuyển động (MV - Motion Vector) dùng cho khối hiện thời dựa vào MVP và MVD đại diện; và

tạo mẫu dự báo dùng cho khối hiện thời dựa vào MV; và

trong đó phương pháp này còn bao gồm bước:

thu cờ khoảng AMVR không tuyến tính mà chỉ báo việc các khoảng MVD có khoảng không bằng nhau hay không từ dòng bit.

10. Phương pháp theo điểm 9, trong đó giá trị đại diện MVD là giá trị đại diện cho khoảng MVD bao gồm giá trị của MVD gốc trong số các khoảng MVD.

11. Phương pháp theo điểm 10, trong đó các khoảng MVD là các khoảng MVD tuyến tính có kích thước khoảng bằng nhau, và

giá trị MVD được mã hóa là giá trị thu được dựa trên sự lấy giá trị đại diện MVD chia cho 4.

12. Phương pháp theo điểm 10, trong đó các khoảng MVD là các khoảng MVD không tuyến tính có các kích thước khoảng không bằng nhau, và

mỗi kích thước của các khoảng MVD thứ nhất tương ứng với các giá trị

MVD được mã hóa -1, 0, và 1 trong số các khoảng MVD không tuyến tính là lớn hơn so với từng kích thước của các khoảng MVD thứ hai trong số các khoảng MVD không tuyến tính,

trong đó các khoảng MVD thứ hai là phần còn lại của các khoảng MVD không tuyến tính, không kể các khoảng MVD thứ nhất.

13. Phương pháp theo điểm 10, trong đó phương pháp này còn bao gồm bước:

thu tham số liên quan đến khoảng AMVR không tuyến tính từ dòng bit khi giá trị của cờ khoảng AMVR không tuyến tính là 1,

trong đó giá trị đại diện MVD thu được dựa vào tham số liên quan đến khoảng AMVR không tuyến tính và giá trị MVD được mã hóa.

14. Phương pháp theo điểm 13, trong đó phương pháp này còn bao gồm bước thu chỉ số bảng khoảng qua dòng bit,

trong đó:

tham số liên quan đến khoảng AMVR không tuyến tính bao gồm thông tin về số lượng ứng viên bảng khoảng không tuyến tính,

tham số liên quan đến khoảng AMVR không tuyến tính bao gồm thông tin về các kích thước của ba khoảng giữa MVD và các giá trị đại diện của các khoảng tương ứng theo từng ứng viên bảng khoảng không tuyến tính, và

chỉ số bảng khoảng chỉ báo một trong số các ứng viên bảng khoảng không tuyến tính.

15. Phương pháp truyền dữ liệu cho hình ảnh, phương pháp này bao gồm các bước:

thu dòng bit dùng cho hình ảnh, trong đó dòng bit này được tạo thành dựa vào sự tạo ra cờ cho phép khoảng vector chuyển động thích ứng (AMVR - Adaptive Motion Vector Range), thu chênh lệch vector chuyển động (MVD - Motion Vector Difference) cho khối hiện thời, thu giá trị đại diện MVD cho khoảng MVD bao gồm giá trị của MVD trong số nhiều khoảng MVD, tạo MVD được mã hóa tương ứng với giá trị đại diện MVD, và giải mã cờ cho phép

AMVR và MVD đã mã hóa để xuất dòng bit; và
truyền dữ liệu bao gồm dòng bit nêu trên,
trong đó phương pháp này còn bao gồm các bước:
tạo cờ khoảng AMVR không tuyến tính mà chỉ báo việc các khoảng
MVD có khoảng không bằng nhau hay không; và
xuất cờ khoảng AMVR không tuyến tính qua dòng bit.

FIG. 2

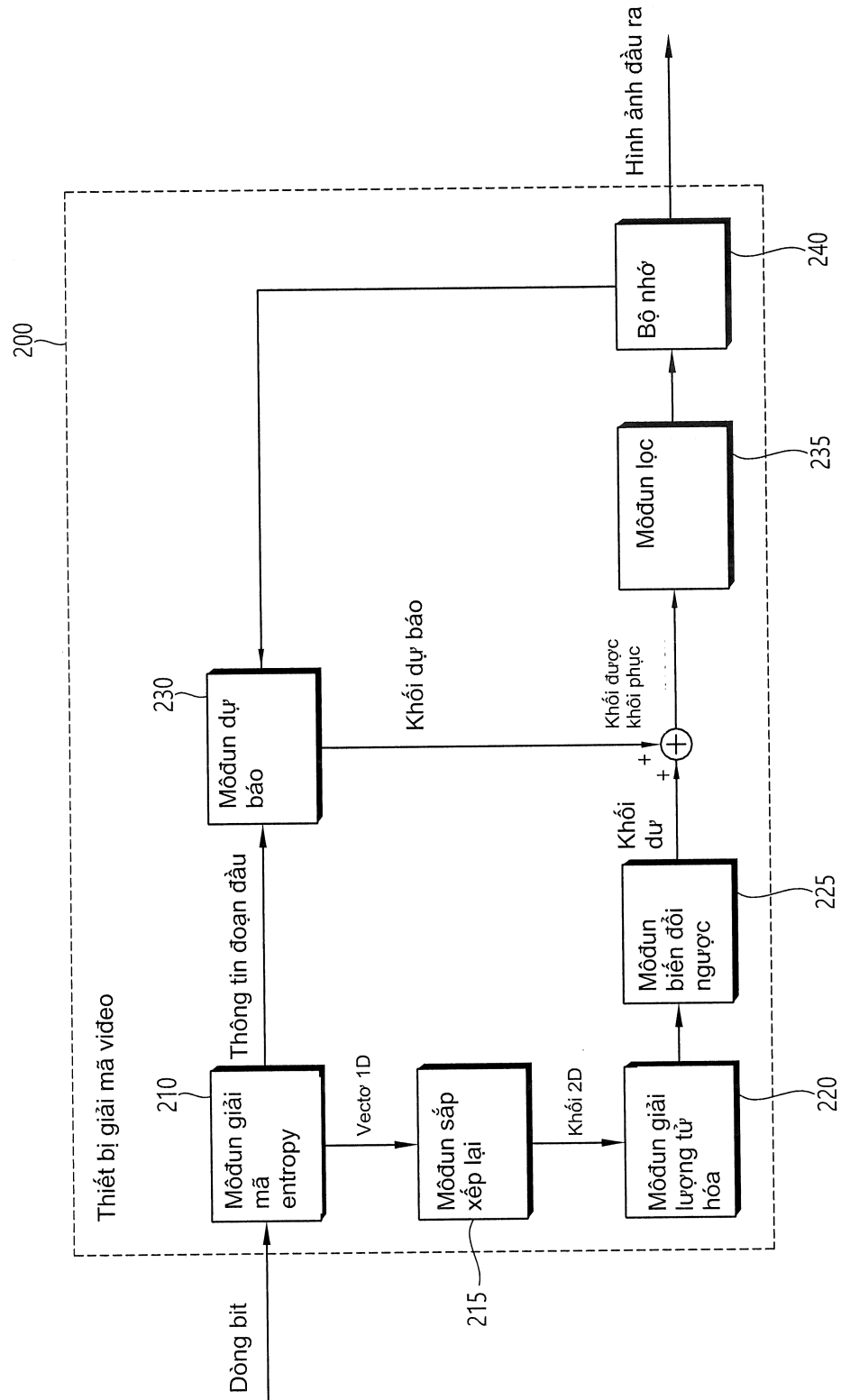


FIG. 3

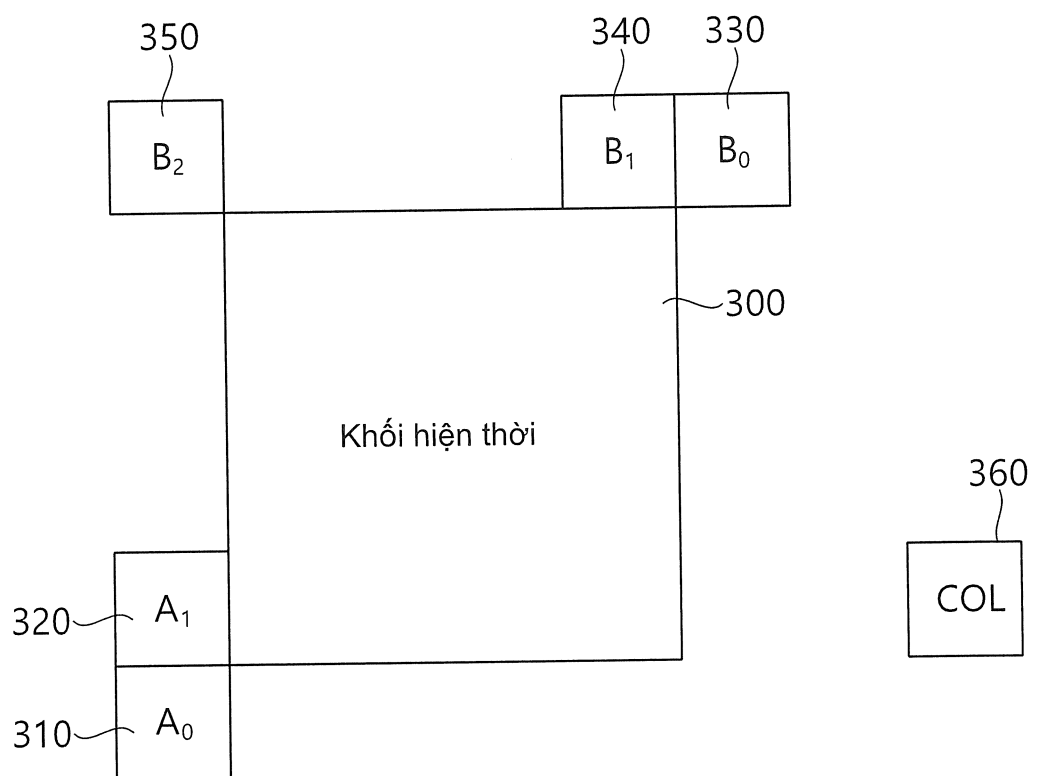


FIG. 4

$A_{-1,-1}$				$A_{0,-1}$	$a_{0,-1}$	$b_{0,-1}$	$c_{0,-1}$	$A_{1,-1}$				$A_{2,-1}$
$A_{-1,0}$				$A_{0,0}$	$a_{0,0}$	$b_{0,0}$	$c_{0,0}$	$A_{1,0}$				$A_{2,0}$
$d_{-1,0}$				$d_{0,0}$	$e_{0,0}$	$f_{0,0}$	$g_{0,0}$	$d_{1,0}$				$d_{2,0}$
$h_{-1,0}$				$h_{0,0}$	$i_{0,0}$	$j_{0,0}$	$k_{0,0}$	$h_{1,0}$				$h_{2,0}$
$n_{-1,0}$				$n_{0,0}$	$p_{0,0}$	$q_{0,0}$	$r_{0,0}$	$n_{1,0}$				$n_{2,0}$
$A_{-1,1}$				$A_{0,1}$	$a_{0,1}$	$b_{0,1}$	$c_{0,1}$	$A_{1,1}$				$A_{2,1}$
$A_{-1,2}$				$A_{0,2}$	$a_{0,2}$	$b_{0,2}$	$c_{0,2}$	$A_{1,2}$				$A_{2,2}$

FIG. 5

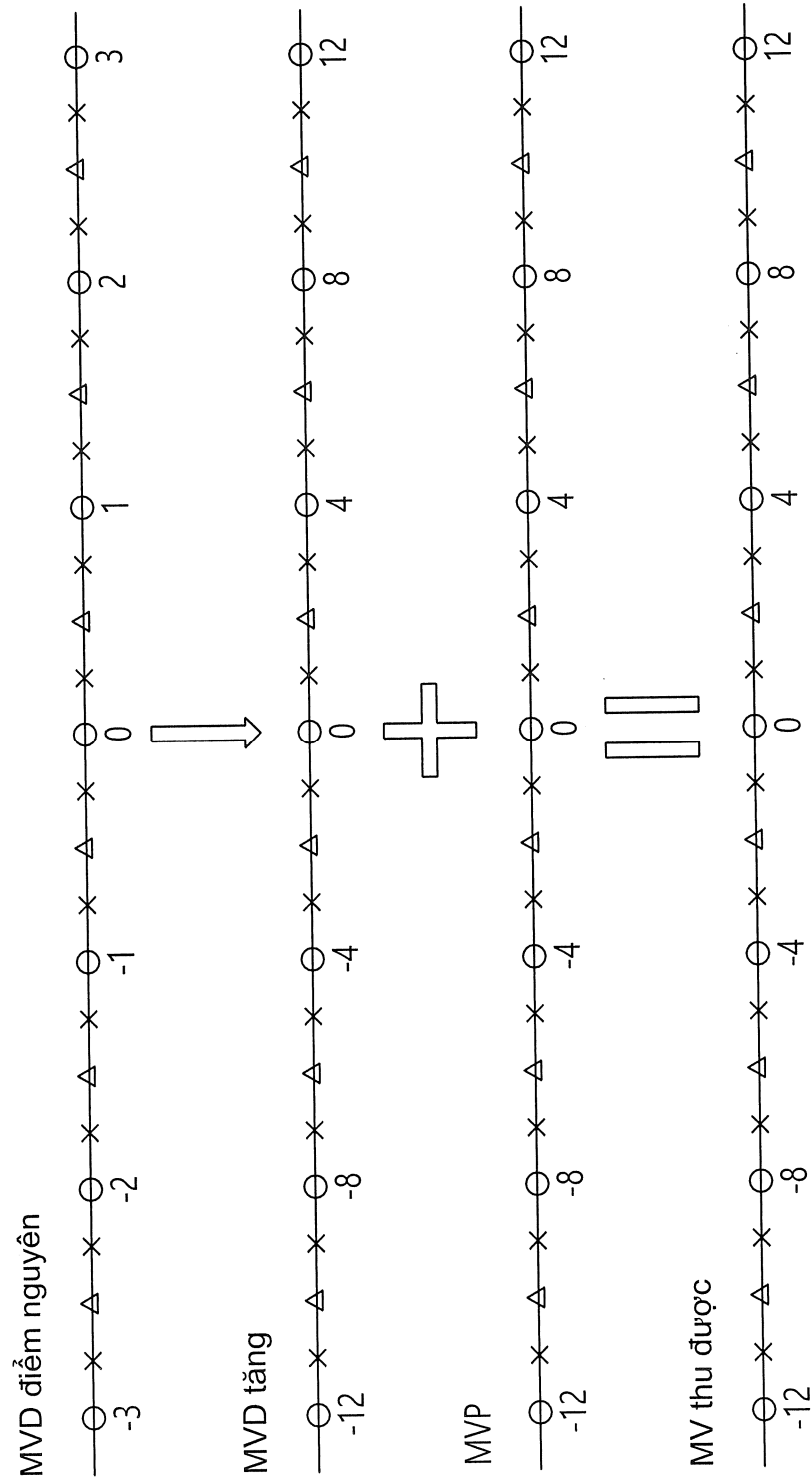


FIG. 6

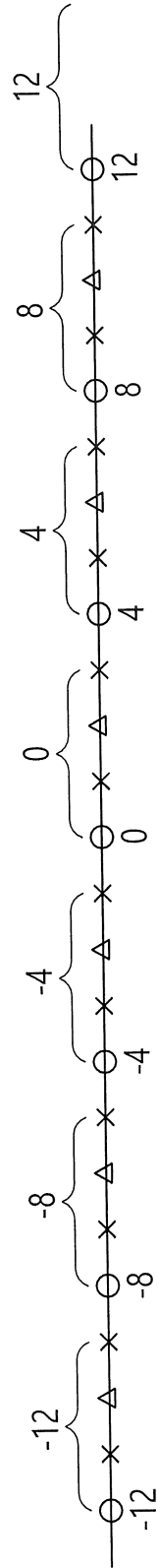


FIG. 7

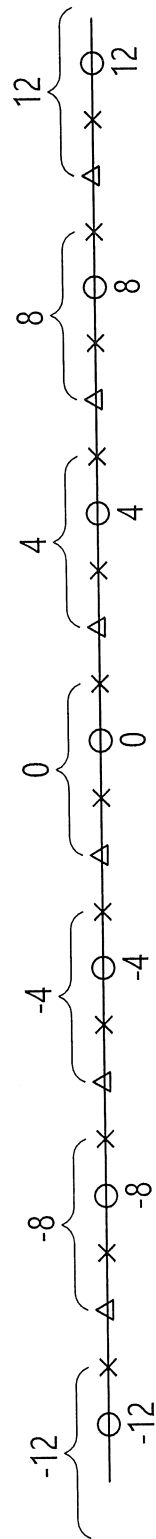


FIG. 8

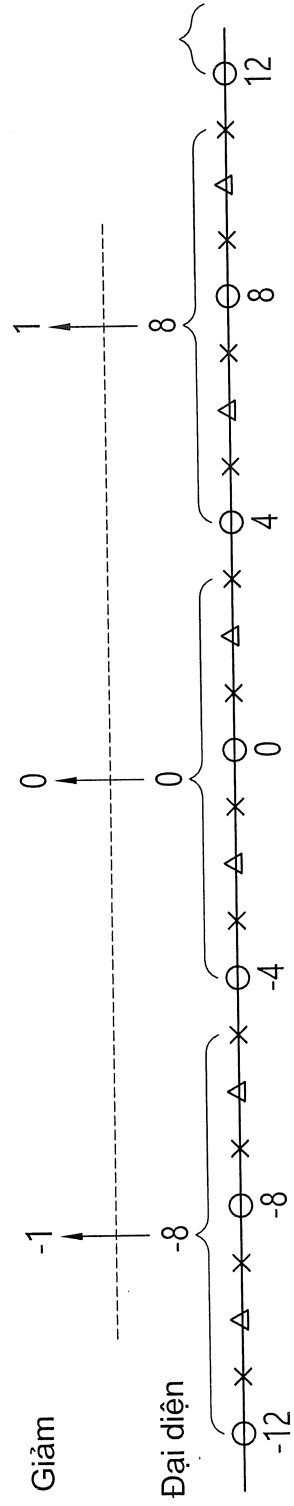
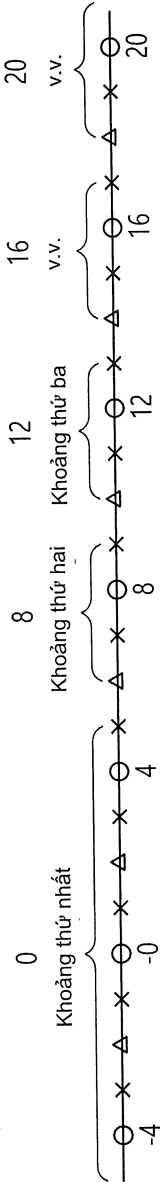


FIG. 9

non_linear_amvr_range_parameters() {
first_range_value (6)
first_range_representative_value (0)
second_range_value (10)
second_range_representative_value (8)
third_range_value (14)
third_range_representative_value (12)
}

(a)



non_linear_amvr_range_parameters() {
first_range_value (4)
first_range_representative_value (0)
second_range_value (12)
second_range_representative_value (8)
third_range_value (16)
third_range_representative_value (14)
}

(b)

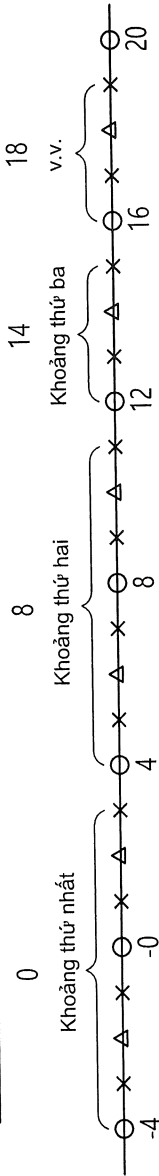


FIG. 10

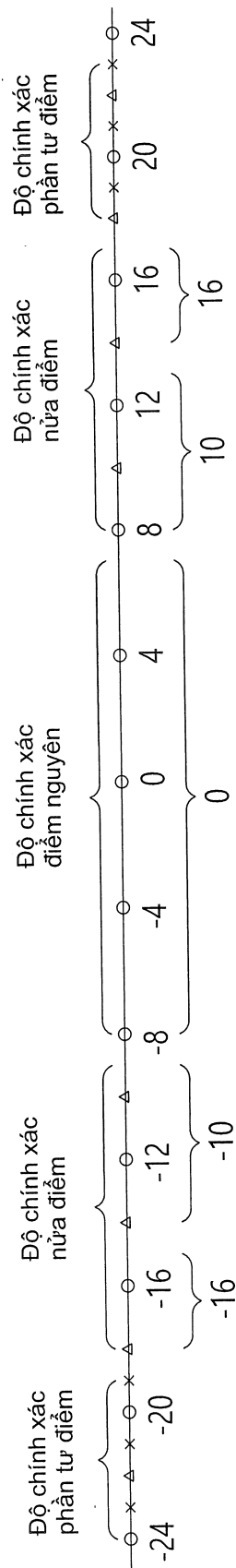


FIG. 11

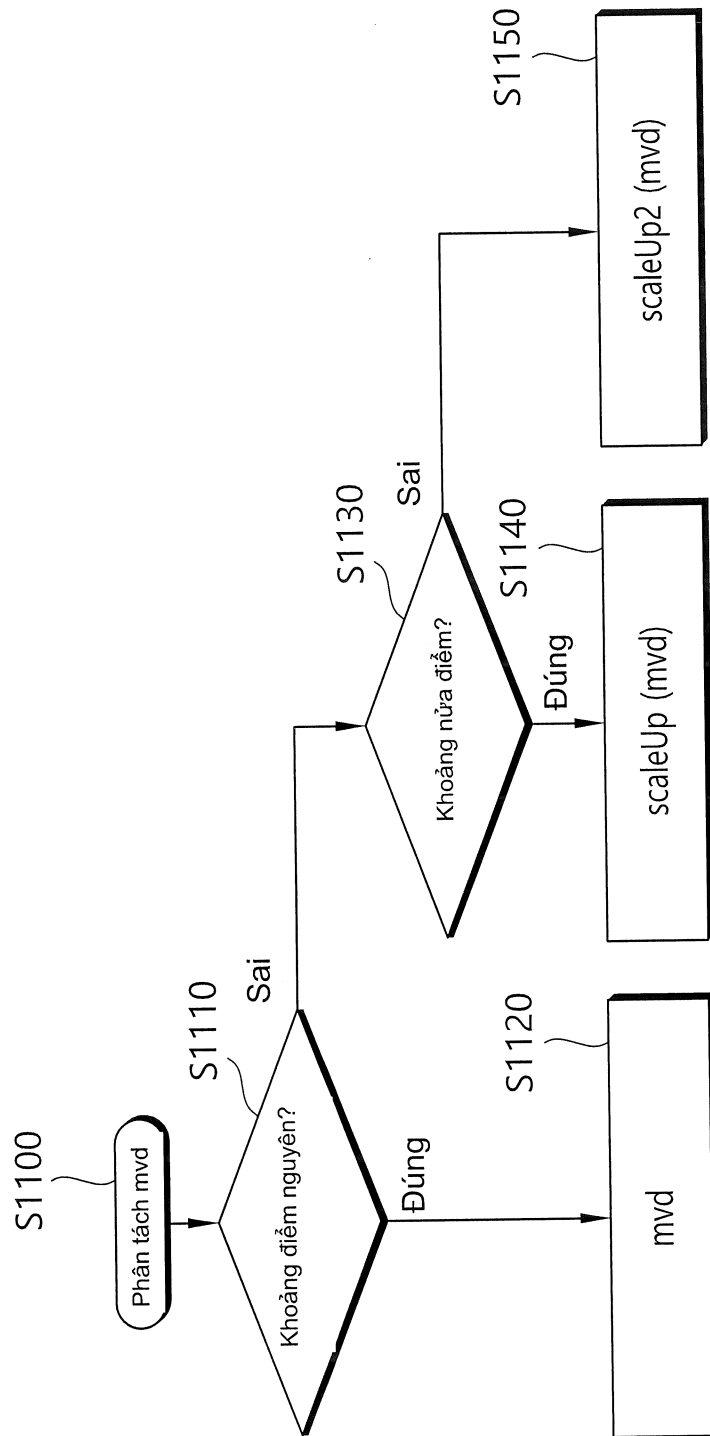


FIG. 12

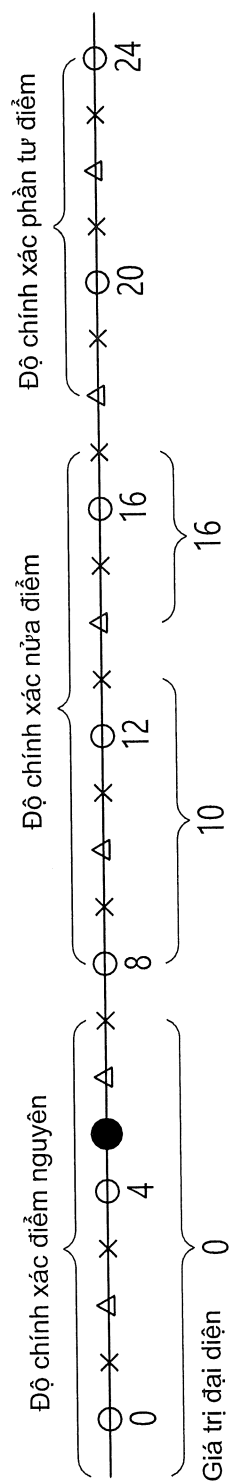


FIG. 13

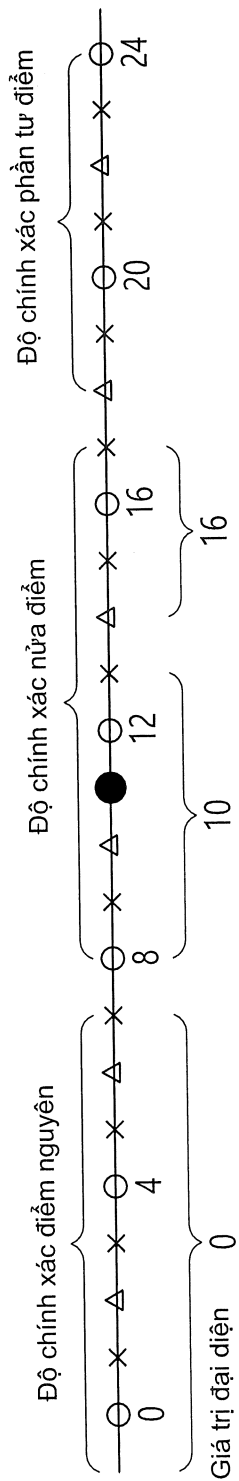


FIG. 14

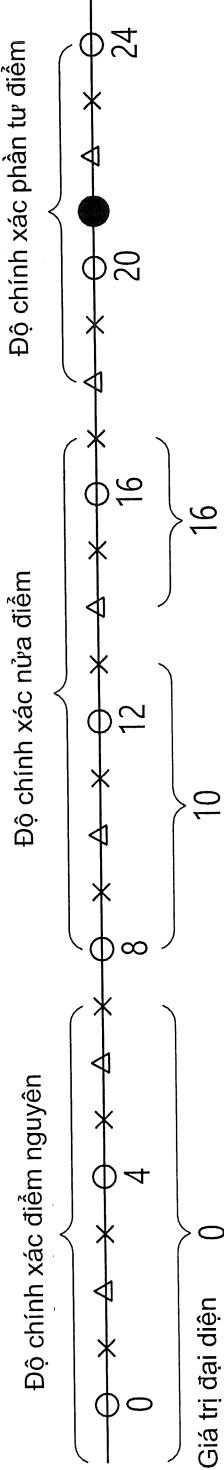


FIG. 15

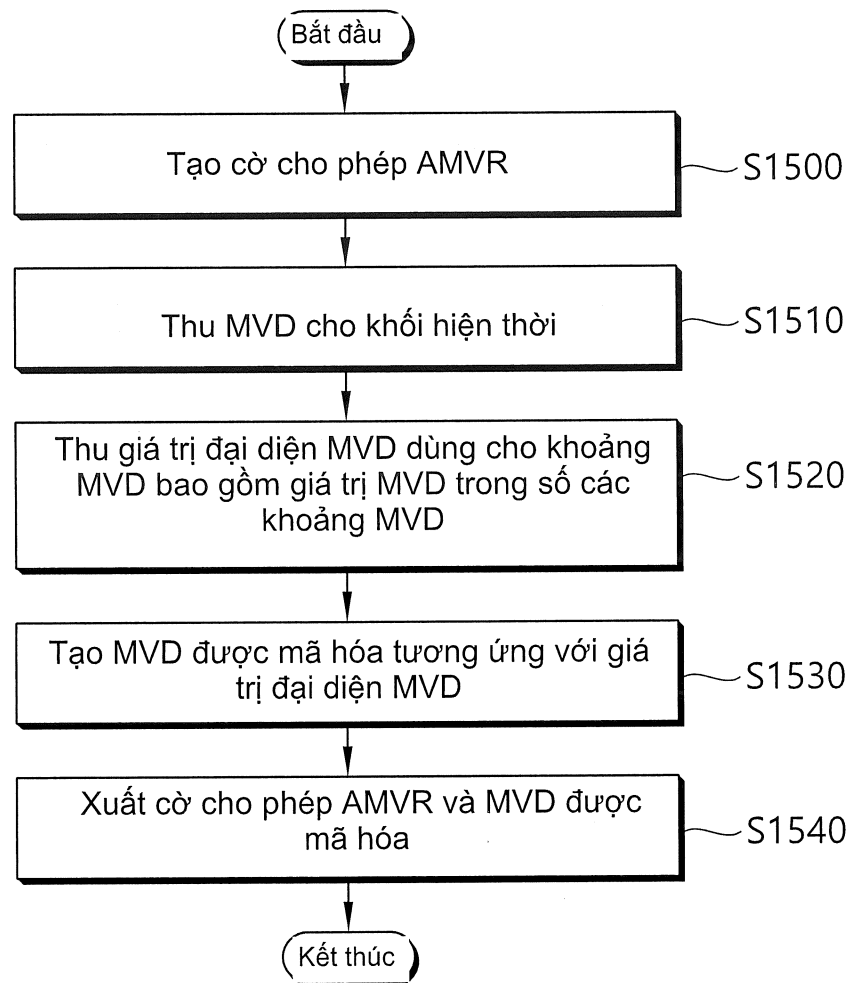


FIG. 16

