



(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)



1-0042043

(51)^{2020.01} H04N 19/186; H04N 19/00

(13) B

(21) 1-2021-08312

(22) 29/05/2020

(86) PCT/CN2020/093499 29/05/2020

(87) WO2020/239110 03/12/2020

(30) 62/855,007 31/05/2019 US; 62/865,390 24/06/2019 US; 62/876,094 19/07/2019 US;
62/905,581 25/09/2019 US

(45) 25/12/2024 441

(43) 25/02/2022 407

(73) HFI Innovation Inc. (TW)

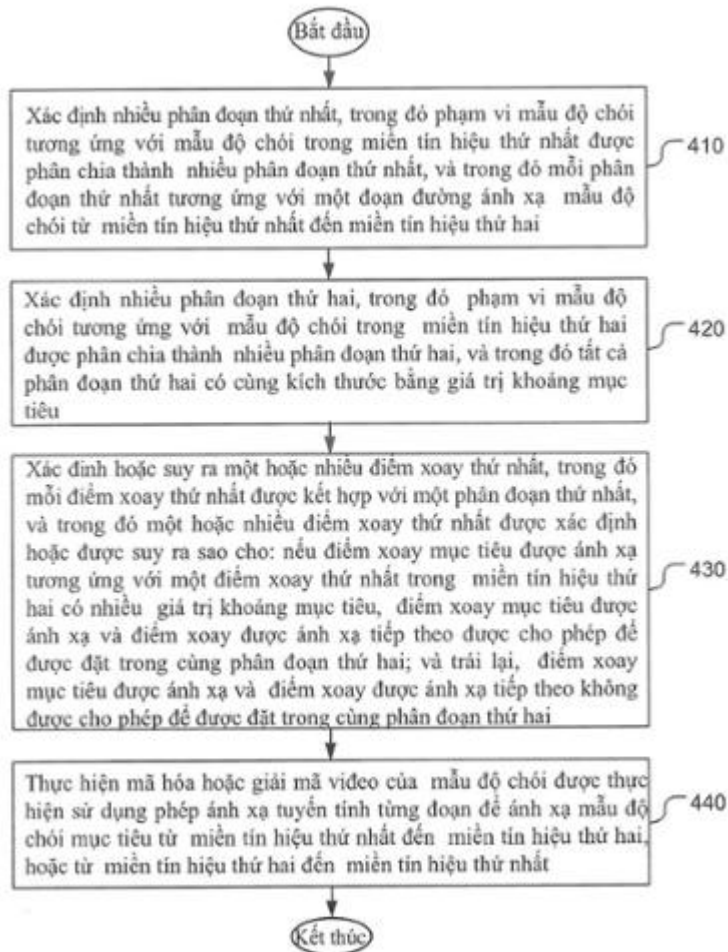
3F.-7, No.5, Taiyuan 1st St., Zhubei City, Hsinchu County 302, Taiwan

(72) LIN, Zhi-Yi (CN); CHUANG, Tzu-Der (CN); CHEN, Ching-Yeh (CN); HSU, Chih-Wei (CN).

(74) Công ty TNHH Trường Xuân (AGELESS CO.,LTD.)

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ LẬP MÃ VIDEO LÀM GIẢM ĐỘ TRỄ CHO
PHÉP CHIA TỈ LỆ ĐỘ SẮC DƯ

(57) Sáng chế đề xuất phương pháp và thiết bị lập mã video. Phép ánh xạ tuyến tính từng đoạn được suy ra để ánh xạ mẫu độ chói mục tiêu từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai, hoặc ngược lại. Quy trình ánh xạ tuyến tính từng đoạn bao gồm xác định điểm xoay thứ nhất để thực hiện mã hóa hoặc giải mã video của mẫu độ chói sử dụng phép ánh xạ tuyến tính từng đoạn. Điểm xoay thứ nhất được xác định sao cho: nếu điểm xoay mục tiêu được ánh xạ tương ứng với một điểm xoay thứ nhất trong miền tín hiệu thứ hai có nhiều giá trị khoảng mục tiêu, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo có thể được đặt trong cùng phân đoạn thứ hai; và trái lại, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo có thể không thể được đặt trong cùng phân đoạn thứ hai.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lập mã video cho dữ liệu video màu, mà phép ánh xạ độ chói được áp dụng cho thành phần độ chói. Cụ thể, sáng chế đề cập đến các công nghệ để đơn giản hóa việc suy ra phép ánh xạ độ chói nghịch đảo.

Tình trạng kỹ thuật của sáng chế

Lập mã video đa năng VVC (Versatile Video Coding - VVC) là tiêu chuẩn lập mã video mới nổi được phát triển bởi nhóm các chuyên gia video chung, nhóm cộng tác được thành lập bởi nhóm các chuyên gia lập mã video 16 nhóm nghiên cứu ITU-T và ISO/IEC JTC1 SC29/WG11 (Moving Picture Experts Group, MPEG). VVC được dựa trên tiêu chuẩn video lập mã video hiệu suất cao HEVC (High Efficient Video Coding - HEVC) có các công cụ lập mã mới và được cải tiến. Ví dụ, quy trình tái định dạng là công cụ lập mã mới được sử dụng trong VTM-4.0 (mô hình thử nghiệm VVC phiên bản 4.0). Quy trình định dạng lại cũng được gọi là phép ánh xạ độ chói và phép chia tỷ lệ độ sắc LMCS (Luma Mapping and Chroma Scaling - LMCS). Khi phép tái định dạng được áp dụng, các mẫu video được lập mã và tái cấu trúc trong miền được tái định dạng trước khi lọc vòng. Các mẫu được tái cấu trúc miền được tái định dạng được biến đổi thành miền gốc bằng cách sử dụng phép tái định dạng nghịch đảo. Các mẫu được tái cấu trúc miền gốc được lọc vòng được lưu trữ trong bộ đệm hình ảnh được giải mã. Đối với chế độ liên, bộ dự đoán được bù chuyển động (motion compensated - MC) được biến đổi thành miền được tái định dạng bằng cách sử dụng phép tái định dạng tiến. FIG. 1 thể hiện ví dụ về quy trình định dạng lại ở phía bộ giải mã.

Như được thể hiện trong FIG. 1, dòng bit được xử lý bởi bộ giải mã phép lập mã số học nhị phân thích ứng với ngữ cảnh CABAC (context-adaptive binary arithmetic coding - CABAC) (tức là, $CABAC^{-1}$), lượng tử hóa nghịch đảo (tức là, Q^{-1}) và phép biến đổi nghịch đảo (T^{-1}) để suy ra phần dư độ chói được tái cấu trúc Y_{res} . Phần dư độ chói được tái cấu trúc được cung cấp cho khối tái cấu trúc độ chói 120 để tạo ra tín hiệu độ chói được tái cấu trúc. Đối với chế độ nội, bộ dự đoán đến từ khối dự đoán nội 130. Đối với chế độ liên, bộ dự đoán đến từ khối bù chuyển động 140. Vì

phép tái định dạng được áp dụng cho tín hiệu độ chói phía bộ mã hóa, phép tái định dạng tiến 150 được áp dụng bộ dự đoán từ khối bù chuyển động 140 trước khi bộ dự đoán được cung cấp cho khối tái cấu trúc 120. Phép tái định dạng nghịch đảo 160 được áp dụng cho tín hiệu độ chói được tái cấu trúc từ khối tái cấu trúc 120 để khôi phục tín hiệu độ chói được tái cấu trúc không được định dạng. Bộ lọc vòng 170 sau đó được áp dụng cho tín hiệu độ chói được tái cấu trúc không được định dạng trước khi tín hiệu được lưu trữ trong bộ đệm hình ảnh được giải mã DPB (decoded picture buffer - DPB) 180.

Khi phép tái định dạng được áp dụng, phép chia tỉ lệ độ sắc dư cũng được áp dụng. Phép chia tỉ lệ độ sắc dư bù cho tín hiệu độ chói tương tác bằng tín hiệu độ sắc, như được thể hiện trong FIG. 2. Trong FIG. 2, phần trên tương ứng với phép giải mã độ chói và phần dưới tương ứng với phép giải mã độ sắc.

Phép chia tỉ lệ độ sắc dư được áp dụng ở cấp độ TU theo các phương trình sau đây ở phía bộ mã hóa và phía bộ giải mã tương ứng:

$$\text{Phía bộ mã hóa : } C_{\text{ResScale}} = C_{\text{Res}} * C_{\text{Scale}} = C_{\text{Res}} / C_{\text{ScaleInv}}(1)$$

$$\text{Phía bộ giải mã : } C_{\text{Res}} = C_{\text{ResScale}} / C_{\text{Scale}} = C_{\text{ResScale}} * C_{\text{ScaleInv}}(2)$$

Theo các phương trình ở trên, C_{Res} là tín hiệu độ sắc dư gốc và C_{ResScale} là tín hiệu độ sắc dư được chia tỉ lệ. C_{Scale} là hệ số chia tỉ lệ được tính toán sử dụng FwdLUT (tức là, bảng tra cứu trước) cho các bộ dự đoán chế độ liên và được biến đổi thành C_{ScaleInv} thuận nghịch của nó để thực hiện phép nhân thay vì phép chia ở phía bộ giải mã, do đó giảm độ phức tạp khi thực hiện. Các hoạt động chia tỷ lệ ở cả phía bộ mã hóa và bộ giải mã được thực hiện với tính toán số nguyên điểm cố định với phương trình như sau:

$$c' = \text{sign}(c) * ((\text{abs}(c) * s + 2^{\text{CSCALE_FP_PREC}-1}) \gg \text{CSCALE_FP_PREC})(3)$$

Trong phương trình ở trên, c là phần dư độ sắc, s là hệ số chia tỉ lệ độ sắc dư từ $c_{\text{ScaleInv}}[\text{pieceIdx}]$, pieceIdx được quyết định bởi giá trị độ chói trung bình tương ứng của TU, và CSCALE_FP_PREC là giá trị hằng số để chỉ định độ chính xác. Để suy ra hệ số chia tỉ lệ, bộ dự đoán của toàn bộ TU được sử dụng. Giá trị của C_{ScaleInv} được tính toán theo các bước sau đây:

(1) nếu là chế độ nội, tính toán giá trị trung bình của các độ chói được dự đoán

nội; nếu là chế độ liên, tính toán giá trị trung bình của độ chói được dựa đoán tái định dạng trước. Nói cách khác, giá trị độ chói trung bình $avgY'_{TU}$ được tính toán trong miền được tái định dạng.

(2) Tìm chỉ số idx , mà $avgY'_{TU}$ thuộc về phép ánh xạ nghịch đảo PWL.

(3) $C_{ScaleInv} = cScaleInv[idx]$

Các bước để suy ra hệ số chia tỉ lệ độ sắc $C_{ScaleInv}$ được thực hiện bằng khối 210 trong FIG. 2. Hệ số chia tỉ lệ độ sắc $C_{ScaleInv}$ được suy ra được sử dụng để chuyển đổi phần dư độ sắc được chia tỷ lệ, mà được tái cấu trúc nhờ phép giải mã CABAC (tức là, $CABAC^{-1}$), lượng tử hóa nghịch đảo (tức là, Q^{-1}) và phép biến đổi nghịch đảo (T^{-1}). Khối tái cấu trúc 220 tái cấu trúc tín hiệu độ sắc bằng việc thêm bổ sung bộ dự đoán cho phần dư độ sắc được tái cấu trúc. Đối với chế độ nội, bộ dự đoán đến từ khối dự đoán nội 230. Đối với chế độ liên, bộ dự đoán đến từ khối bù chuyển động 240. Bộ lọc vòng 270 sau đó được áp dụng cho tín hiệu độ sắc tái cấu trúc trước khi tín hiệu được lưu trữ trong chroma bộ đệm hình ảnh được giải mã (DPB) độ sắc 280.

FIG. 3A và FIG. 3B minh họa ví dụ phép ánh xạ độ chói. Trong FIG. 3A, phép ánh xạ 1:1 được thể hiện mà đầu ra (tức là, độ chói được tái cấu trúc) giống như đầu vào. Vì biểu đồ của mẫu độ chói thường không bằng phẳng, sử dụng phép định hình cường độ có thể giúp cải thiện hiệu suất trong phát hiện tối ưu hóa tốc độ biến dạng RDO (rate-distortion optimization - RDO). Các thống kê về mẫu độ chói được tính toán cho vùng ảnh, như là hình ảnh. Đường cong ánh xạ sau đó được xác định theo các thống kê. Thông thường, đường cong ánh xạ tuyến tính từng đoạn PWL (piece-wise linear - PWL) được sử dụng. FIG. 3B minh họa ví dụ phép ánh xạ tuyến tính từng đoạn (PWL) có 3 phân đoạn, mà hai phân đoạn lân cận có các hệ số góc khác nhau. Đường nét đứt 340 tương ứng với phép ánh xạ 1:1. Nếu phạm vi mẫu từ 0 đến 340 có phương sai không gian lớn hơn và số lần xuất hiện nhỏ hơn, phạm vi đầu vào 0-340 được ánh xạ đến phạm vi đầu ra nhỏ hơn (tức là, 0-170), như được thể hiện trong phân đoạn 310 của FIG. 3B. Nếu phạm vi mẫu từ 340 đến 680 có phương sai không gian nhỏ hơn và số lần xuất hiện lớn hơn, phạm vi đầu vào 340-680 được ánh xạ đến phạm vi đầu ra lớn hơn (tức là, 170-850), như được thể hiện trong phân đoạn 320 của FIG. 3B. Nếu phạm vi mẫu từ 680 đến 1023 có phương sai không gian lớn hơn và số lần xuất hiện nhỏ hơn, phạm vi đầu vào 680-1023 được ánh xạ đến phạm vi đầu ra nhỏ

hơn (tức là, 850-1023), như được thể hiện trong phân đoạn 330 của FIG. 3B. FIG. 3B được dự định minh họa phép ánh xạ PWL đơn giản. Theo thực tế, phép ánh xạ PWL có thể có nhiều hoặc ít phân đoạn.

Phân vùng khối phụ nội (Intra sub-block partition - ISP) và biến đổi khối phụ SBT (sub-block transform - SBT)

Để tạo ra các bộ dự đoán chế độ nội tốt hơn, phân vùng khối phụ nội (ISP) có thể được áp dụng. Khi ISP được áp dụng, thành phần độ chói được phân chia thành nhiều TB phụ. Các TB phụ được tái cấu trúc từng TB một. Đối với từng TU phụ, mẫu được tái cấu trúc của TB phụ lân cận có thể được sử dụng như mẫu được tái cấu trúc lân cận cho dự đoán nội. Đối với thành phần độ sắc TB, sẽ được không được phân chia thành nhiều TB phụ như là độ chói.

Tương tự với ISP, biến đổi khối phụ (SBT) có thể được áp dụng cho chế độ liên. Khi SBT được áp dụng, chỉ một phần của dữ liệu CU được biến đổi. Ví dụ, khối hiện thời có thể được phân chia thành hai phân vùng bởi phép phân tách phương ngang hoặc phép phân tách thẳng đứng. Chỉ một trong các phép phân vùng có thể được sử dụng để lập mã biến đổi. Phần dư của các phân vùng khác được thiết lập để bằng 0. Ví dụ, CU được phân chia thành hai TU hoặc bốn TU. Chỉ một trong các TU có hệ số khác 0.

Báo hiệu các tham số LMCS

Bảng cú pháp của các tham số LMCS được xem xét bởi VVC được thể hiện trong Bảng 1.

Bảng 1.

lmcs_data () {	Bộ mô tả
lmcs_min_bin_idx	ue(v)
lmcs_delta_max_bin_idx	ue(v)
lmcs_delta_cw_prec_minus1	ue(v)

for (i = lmcs_min_bin_idx; i <= LmcsMaxBinIdx; i++) {	
lmcs_delta_abs_cw[i]	u(v)
if (lmcs_delta_abs_cw[i] > 0)	
lmcs_delta_sign_cw_flag[i]	u(1)
}	
}	

Theo bảng cú pháp ở trên, các ngữ nghĩa của các cú pháp được xác định như sau:

lmcs_min_bin_idx chỉ định chỉ số thùng cực tiểu của mô hình tuyến tính từng đoạn PWL cho phép ánh xạ độ chói

lmcs_delta_max_bin_idx chỉ định giá trị delta giữa 15 và chỉ số thùng cực đại LmcsMaxBinIdx được sử dụng trong lmcs. Các giá trị phải nằm trong phạm vi 1 đến 15.

lmcs_delta_cw_prec_minus1 cộng 1 số lượng các bit được sử dụng để đại diện cho cú pháp **lmcs_delta_abs_cw[i]**.

lmcs_delta_abs_cw[i] là giá trị từ mã delta tuyệt đối cho thùng thứ i.

lmcs_delta_sign_cw_flag[i] là dấu hiệu của biến số **lmcsDeltaCW[i]**.

lmcsDeltaCW[i] có thể biến đổi được suy ra như sau:

$$\text{lmcsDeltaCW}[i] = (1 - 2 * \text{lmcs_delta_sign_cw_flag}[i]) * \text{lmcs_delta_abs_cw}[i].$$

Biến số **lmcsCW[i]** với $i = 0 \dots 15$ chỉ định số lượng các từ mã cho từng khoảng trong miền được ánh xạ. Có thể được suy ra như sau:

$$\text{OrgCW} = (1 \ll \text{BitDepthY}) / 16$$

–Đối với $i = 0 \dots \text{lmcs_min_bin_idx} - 1$, **lmcsCW[i]** được thiết lập bằng 0.

–Đối với $i = \text{lmcs_min_bin_idx} \dots \text{LmcsMaxBinIdx}$, áp dụng các điều sau:

$$-lmcsCW[i] = OrgCW + lmcsDeltaCW[i]$$

–Giá trị của $lmcsCW[i]$ sẽ nằm trong phạm vi $(OrgCW \gg 3)$ đến $(OrgCW \ll 3 - 1)$, bao hàm.

–Đối với $i = LmcsMaxBinIdx + 1 \dots 15$, $lmcsCW[i]$ được thiết lập bằng 0.

Để đại diện cho mô hình PWL của đường cong tái định dạng, ba biến số $LmcsPivot[i]$ với $i = 0 \dots 16$, $ScaleCoeff[i]$ với $i = 0 \dots 15$, và $InvScaleCoeff[i]$ với $i = 0 \dots 15$, được suy ra như sau:

```

LmcsPivot[0] = 0;
for( i = 0; i <= 15; i++ ) {
    LmcsPivot[i + 1] = LmcsPivot[i] + lmcsCW[i]
    ScaleCoeff[i] = ( lmcsCW[i] * (1 << SCALE_FP_PREC) + (1 <<
        (Log2(OrgCW) - 1))) >> (Log2(OrgCW))
    if ( lmcsCW[i] == 0 )
        InvScaleCoeff[i] = 0
    else
        InvScaleCoeff[i] = OrgCW * (1 << SCALE_FP_PREC) / lmcsCW[i]
}

```

Theo phép suy ra ở trên, $SCALE_FP_PREC$ là giá trị hằng số để chỉ định độ chính xác.

Theo quy trình LMCS thông thường, phép ánh xạ độ chói nghịch đảo, như là so sánh mẫu để được ánh xạ với nhiều ngưỡng, là khá tốn thời gian. Vì vậy, cần phải phát triển các kỹ thuật để đơn giản hóa quy trình ánh xạ nghịch đảo.

Bản chất kỹ thuật của sáng chế

Phương pháp và thiết bị lập mã video được bộc lộ. Theo một phương pháp của sáng chế, nhiều phân đoạn thứ nhất được xác định, mà phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ nhất được phân chia thành nhiều phân đoạn thứ nhất, và mỗi phân đoạn thứ nhất tương ứng với một đoạn đường ánh xạ mẫu độ chói từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai. Ngoài ra, nhiều phân đoạn thứ hai được xác định, mà phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ hai được phân chia thành nhiều phân đoạn thứ hai, và tất cả phân đoạn thứ hai

có cùng kích thước bằng giá trị khoảng mục tiêu. Một hoặc nhiều điểm xoay thứ nhất được xác định hoặc được suy ra, mà mỗi điểm xoay thứ nhất được kết hợp với một phân đoạn thứ nhất, và trong đó một hoặc nhiều điểm xoay thứ nhất này được xác định hoặc được suy ra sao cho: nếu điểm xoay mục tiêu được ánh xạ tương ứng với một điểm xoay thứ nhất trong miền tín hiệu thứ hai có nhiều giá trị khoảng mục tiêu, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo được cho phép để được đặt trong cùng phân đoạn thứ hai; và trái lại, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo không được cho phép để được đặt trong cùng phân đoạn thứ hai. Mã hóa hoặc giải mã video của mẫu độ chói được thực hiện sử dụng phép ánh xạ tuyến tính từng đoạn để ánh xạ mẫu độ chói mục tiêu từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai, hoặc từ miền tín hiệu thứ hai đến miền tín hiệu thứ nhất.

Theo phương án này, phép ánh xạ tuyến tính từng đoạn được kết hợp với nhiều phân đoạn thứ nhất, nhiều phân đoạn thứ hai và một hoặc nhiều điểm xoay thứ nhất.

Theo phương án này, tất cả phân đoạn thứ nhất có cùng giá trị khoảng thứ nhất. Ví dụ, giá trị khoảng thứ nhất tương ứng với 2 mũ số định trước thứ nhất và số định trước thứ nhất là số nguyên. Hơn nữa, số định trước thứ nhất có thể bằng $\text{bitDepth} - 4$ và bitDepth biểu thị độ sâu bit của mẫu độ chói.

Theo phương án này, giá trị khoảng mục tiêu tương ứng với 2 mũ số định trước thứ hai và số định trước thứ hai là số nguyên. Ví dụ, số định trước thứ hai bằng $\text{bitDepth} - 5$ và bitDepth đại diện độ sâu bit của mẫu độ chói.

Theo phương án này, mỗi phân đoạn thứ hai có nhiều nhất hai điểm xoay được ánh xạ. Theo phương án này, phạm vi mẫu độ chói bằng 0 đến 1023, và giá trị khoảng mục tiêu bằng 32.

Mô tả vắn tắt các hình vẽ

FIG. 1 minh họa sơ đồ khối điển hình của bộ giải mã video kết hợp quy trình định dạng lại độ chói.

FIG. 2 minh họa sơ đồ khối điển hình của bộ giải mã video kết hợp quy trình định dạng lại độ chói và chia tỷ lệ độ sắc.

FIG. 3A minh họa ví dụ 1:1 phép ánh xạ độ chói, mà đầu ra (tức là, độ chói

được tái cấu trúc) giống như đầu vào.

FIG. 3B minh họa ví dụ phép ánh xạ độ chói tuyến tính từng đoạn (PWL) có 3 phân đoạn.

FIG. 4 minh họa sơ đồ về hệ thống lập mã điển hình để xác định hoặc suy ra các điểm xoay của phép ánh xạ độ chói nghịch đảo theo phương án của sáng chế.

Mô tả chi tiết sáng chế

Phần mô tả sau đây là phương thức thực hiện tốt nhất để thực hiện sáng chế. Phần mô tả này được tạo ra nhằm mục đích minh họa các nguyên lý chung của sáng chế và không nên coi là hạn chế sáng chế. Phạm vi của sáng chế được xác định tốt nhất bằng cách tham chiếu đến các yêu cầu bảo hộ kèm theo.

LMCS ánh xạ mẫu độ chói trong miền gốc đến miền được tái định dạng để ước tính dữ liệu tốt hơn. Đường cong ánh xạ được ước tính bởi mô hình tuyến tính từng đoạn. Để biến đổi các giá trị mẫu từ miền gốc đến miền được tái định dạng, bảng tra cứu (look-up-table - LUT) được sử dụng. Số lượng mục của LUT giống như phạm vi động mẫu đầu vào. Ví dụ, nếu các đầu vào 10-bit được sử dụng, 1024 mục LUT được sử dụng. Nếu đầu vào 14-bit được sử dụng, 8192 mục LUT được sử dụng. Theo phép thực hiện phân cứng, loại LUT này có chi phí đắt đỏ. Vì vậy, mô hình tuyến tính từng đoạn có thể được sử dụng. Đầu vào có thể được so sánh với từng đoạn để tìm ra đầu vào thuộc về đoạn nào. Trong từng đoạn, giá trị đầu ra tương ứng có thể được tính toán theo đặc điểm của đoạn này.

Phương pháp 1 - LMCS với số lượng mã hóa

Theo phương án này, số lượng từ mã cho từng thùng trong miền được ánh xạ (ví dụ $lmcsCW[i]$) có thể được suy ra sử dụng số lượng mặc định của các từ mã thay vì sử dụng $OrgCW$, mà chỉ phụ thuộc vào độ sâu bit của dữ liệu đầu vào.

Theo phương pháp được đề xuất, các biến $lmcsCW[i]$ với $i = lmcs_min_bin_idx$ đến $lmcsMaxBinIdx$, được suy ra theo:

$$lmcsCW[i] = default_CW + lmcsDeltaCW[i],$$

Trong phương trình ở trên, $default_CW$ được suy ra ở phía bộ giải mã hoặc được báo hiệu từ phía bộ mã hóa.

Theo phương án này, nếu `default_CW` được suy ra ở phía bộ giải mã, nó có thể được suy ra theo `lmcs_min_bin_idx` và `LmcsMaxBinIdx`. Nếu tổng số các thùng ít hơn `lmcs_min_bin_idx` và số lượng các thùng lớn hơn `LmcsMaxBinIdx` là lớn hơn, `default_CW` có thể được điều chỉnh đến giá trị lớn hơn `OrgCW`.

Ví dụ, nếu tổng số các thùng nhỏ hơn `lmcs_min_bin_idx` và số lượng các thùng lớn hơn `LmcsMaxBinIdx` bằng 2, `default_CW` được suy ra là $\text{default_CW} = \text{OrgCW} + A$, mà là số nguyên dương (ví dụ 1, 2, 3...).

Nếu tổng số các thùng nhỏ hơn `lmcs_min_bin_idx` và số lượng các thùng lớn hơn `LmcsMaxBinIdx` bằng 0, sau đó `default_CW` bằng `OrgCW`.

Theo phương án này, nếu `default_CW` được báo hiệu, hai cú pháp, `default_delta_abs_CW` và `default_delta_sign_CW_flag` được báo hiệu trước khi `lmcs_delta_cw_prec_minus1`.

Biến số `default_delta_abs_CW` đại diện cho sai phân tuyệt đối của `default_CW` và `OrgCW`, và các biến `default_delta_sign_CW_flag` biểu thị xem giá trị `delta` là dương hay âm. `Default_delta_sign_CW_flag` chỉ được báo hiệu nếu `default_delta_abs_CW` lớn hơn 0.

Theo phương án này, nếu `default_CW` được báo hiệu, cú pháp `default_delta_CW` được báo hiệu sau `lmcs_delta_cw_prec_minus1`.

Theo phương án này, nếu `default_CW` được báo hiệu, cú pháp `lmcs_delta_default_cw_prec_minus1` khác được báo hiệu trước khi cú pháp `default_delta_CW`, biểu thị xem bao nhiêu thùng được yêu cầu cho cú pháp `default_delta_CW`.

Biến số `default_delta_CW` đại diện sai phân giữa giữa `default_CW` và `OrgCW`.

Phương pháp 2 - Phép ánh xạ nghịch đảo được đơn giản hóa

Đối với phép ánh xạ độ chói, biến số `InputPivot[ i ]` được xác định, mà `InputPivot[ i ]` đại diện cho các điểm xoay cho mẫu độ chói đầu vào. Biến số `InputPivot[ i ]`, với $i = 0 \dots 16$, được suy ra như sau:

$$\text{InputPivot}[i] = i * \text{OrgCW}.$$

Theo thiết kế ban đầu, các giá trị thang đo nghịch đảo được lưu trữ trong bảng

với 16 mục, và mỗi mục tương ứng với mỗi đoạn của ánh xạ PWL. Vì vậy, khi thực hiện phép ánh xạ nghịch đảo, để biết được đoạn mà mẫu để được ánh xạ thuộc về, mẫu độ chói để được ánh xạ cần được so sánh với điểm xoay của các đoạn trong miền được tái định dạng để đặt đoạn mà mẫu độ chói để được ánh xạ thuộc về. Quy trình này đòi hỏi nhiều nhất 15 phép so sánh cho từng mẫu để được ánh xạ, và phép suy ra là như sau.

```

if ( lumaSample < LmcsPivot[ lmcs_min_bin_idx + 1 ] )

idxYInv = lmcs_min_bin_idx

else if ( lumaSample >= LmcsPivot[ LmcsMaxBinIdx ] )

idxYInv = LmcsMaxBinIdx

else {

for( idxYInv = lmcs_min_bin_idx; idxYInv < LmcsMaxBinIdx; idxYInv++ ) {

if( lumaSample < LmcsPivot [ idxYInv + 1 ] )

break

}

}

```

Theo phép suy ra ở trên, lumaSample là mẫu để được ánh xạ, LmcsPivot[i] với $i = 0 \dots 16$ là điểm xoay của từng đoạn trong miền tái định dạng, lmcs_min_bin_idx là chỉ số thùng cực tiểu được sử dụng trong phép ánh xạ độ chói, LmcsMaxBinIdx là chỉ số thùng cực đại được sử dụng trong phép ánh xạ độ chói, và idxYInv là đoạn mà mẫu độ sắc lumaSample thuộc về.

Phương pháp 2.1-Tái cấu trúc đường cong tái định dạng nghịch đảo

Để đơn giản hóa quy trình ánh xạ nghịch đảo, được đề xuất để tái cấu trúc đường cong ánh xạ nghịch đảo theo đường cong ánh xạ được báo hiệu tiến xạ. Mẫu độ chói trong miền được tái định dạng được chia thành các đoạn cho kích thước bằng $(1 \ll \text{LOG2_INVERSE_INTERVAL})$, và mỗi đoạn có giá trị thang đo nghịch đảo và các điểm xoay đầu vào nghịch đảo của nó. Giá trị thang đo nghịch đảo cho từng đoạn được thực hiện với bảng có kích thước bằng $((1 \ll \text{bitDepth}) \gg \text{LOG2_INVERSE_INTERVAL})$, mà bitDepth là độ sâu bit bên trong. Khi tìm kiếm

đoạn mà mẫu độ chói để được ánh xạ thuộc về, chúng ta chỉ cần dịch chuyển sang bên phải mẫu độ chói để được ánh xạ bởi LOG2_INVERSE_INTERVAL (tức là, $\log_2(\text{INVERSE_INTERVAL})$), sau đó chỉ số của đoạn và điểm xoay đầu vào nghịch đảo có thể được suy ra và do đó giá trị thang đo nghịch đảo và điểm xoay đầu vào nghịch đảo có thể được tìm thấy.

Giá trị thang đo nghịch đảo của từng khoảng trong miền được tái định dạng được suy ra bằng cách phân chia sai phân của mẫu độ chói nghịch đảo thứ nhất và mẫu độ chói nghịch đảo cuối cùng với kích thước bằng khoảng trong miền được tái định dạng.

Theo một ví dụ, các giá trị thang đo nghịch đảo bảng được tái cấu trúc bởi các bước sau:

Bước 1. Suy ra các giá trị thang đo tiến theo dữ liệu LMCS được báo hiệu

Để biểu thị mô hình PWL của đường cong tái định dạng, ba biến số $\text{LmcsPivot}[i]$ với $i = 0 \dots 16$, $\text{ScaleCoeff}[i]$ với $i = 0 \dots 15$, và $\text{InvScaleCoeff_tmp}[i]$ với $i = 0 \dots 15$, được suy ra như sau:

```

LmcsPivot[ 0 ] = 0;
for( i = 0; i <= 15; i++ ) {
    LmcsPivot[ i + 1 ] = LmcsPivot[ i ] + lmcscw[ i ]
    ScaleCoeff[ i ] = ( lmcscw[ i ] * (1 << SCALE_FP_PREC) + (1 <<
(Log2(OrgCW) - 1))) >> (Log2(OrgCW))
    if ( lmcscw[ i ] == 0 )
        InvScaleCoeff_tmp [ i ] = 0
    else
        InvScaleCoeff_tmp [ i ] = OrgCW * (1 << SCALE_FP_PREC) / lmcscw[ i ]
}

```

Bước 2. Tìm giá trị xoay của giá trị xoay được ánh xạ nghịch đảo

Đối với mẫu độ chói bằng bội của $(1 \ll \text{LOG2_INVERSE_INTERVAL})$ trong phạm vi từ 0 đến $(1 \ll \text{BitDepth}) - 1$, biến số $\text{InputPivot_inv}[i \gg \text{LOG2_INVERSE_INTERVAL}]$ được suy ra như sau.

```

if (i < LmcsPivot [lmcsc_min_bin_idx])

```

```

    InputPivot_inv[i >> LOG2_INVERSE_INTERVAL] = (OrgCW) *
    lmcs_min_bin_idx

```

```

    if (i >= LmcsPivot [LmcsMaxBinIdx + 1])

```

```

        InputPivot_inv[i >> LOG2_INVERSE_INTERVAL] = (OrgCW) *
        (LmcsMaxBinIdx + 1)

```

```

    Otherwise,

```

```

    InputPivot_inv[i >> LOG2_INVERSE_INTERVAL] =

```

```

        InputPivot[idxYInv] + InvScaleCoeff_tmp[idxYInv] * (i -
        LmcsPivot[ idxYInv ]) + (1 << ( SCALE_FP_PREC - 1 )) >> SCALE_FP_PREC.

```

Trong phương trình ở trên, idxYInv được suy ra bằng việc thiết lập lumaSample bằng i, và sau đó theo quy trình sau đây:

```

    if ( lumaSample < LmcsPivot[ lmcs_min_bin_idx + 1 ] )

```

```

        idxYInv = lmcs_min_bin_idx

```

```

    else if ( lumaSample >= LmcsPivot[ LmcsMaxBinIdx ] )

```

```

        idxYInv = LmcsMaxBinIdx

```

```

    else {

```

```

        for( idxYInv = lmcs_min_bin_idx; idxYInv < LmcsMaxBinIdx; idxYInv++ ) {

```

```

            if( lumaSample < LmcsPivot [ idxYInv + 1 ] )

```

```

                break;

```

```

        }

```

```

    }

```

Bước 3. Suy ra giá trị thang đo nghịch đảo cho từng đoạn

Đối với đoạn có chỉ số bằng i, $i = 0 \dots (\text{LmcsPivot}[\text{LmcsMaxBinIdx} + 1] \gg \text{LOG2_INVERSE_INTERVAL})$:

```

    If ((i == (LmcsPivot [LmcsMaxBinIdx + 1] >>
    LOG2_INVERSE_INTERVAL)) && (LmcsPivot [LmcsMaxBinIdx + 1] != ( i <<
    LOG2_INVERSE_INTERVAL ))

```

$$\text{InvScaleCoeff}[i] = (\text{InputPivot_inv}[i+1] - \text{InputPivot_inv}[i])$$

$$* (1 \ll \text{SCALE_FP_PREC}) / (\text{LmcsPivot}[\text{LmcsMaxBinIdx} + 1] - (i \ll \text{LOG2_INVERSE_INTERVAL}))$$

Otherwise,

$$\text{InvScaleCoeff}[i] = (\text{InputPivot_inv}[i+1] - \text{InputPivot_inv}[i]) * (1 \ll \text{SCALE_FP_PREC}) / (1 \ll \text{LOG2_INVERSE_INTERVAL}).$$

Trong phương án khác, đối với đoạn có chỉ số bằng i , $i = 0 \dots (((1 \ll \text{bitDepth}) - 1) \gg \text{LOG2_INVERSE_INTERVAL}) - 1$:

$$\text{InvScaleCoeff}[i] = (\text{InputPivot_inv}[i+1] - \text{InputPivot_inv}[i]) * (1 \ll \text{SCALE_FP_PREC}) / (1 \ll \text{LOG2_INVERSE_INTERVAL})$$

Phép ánh xạ nghịch đảo

Đối với phép ánh xạ nghịch đảo, biến số idxYInv được suy ra bởi $(\text{lumaSample} \gg \text{LOG2_INVERSE_INTERVAL})$, mà lumaSample là mẫu để được ánh xạ. Nói cách khác, idxYInv được suy ra bởi right-shifting lumaSample bởi các bit $\text{LOG2_INVERSE_INTERVAL}$. Biến số invSample được suy ra như sau:

$$\text{invSample} = \text{InputPivot_inv}[\text{idxYInv}] + (\text{InvScaleCoeff}[\text{idxYInv}] * (\text{lumaSample} - (\text{idxYInv} \ll \text{LOG2_INVERSE_INTERVAL})) + (1 \ll (\text{SCALE_FP_PREC} - 1))) \gg \text{SCALE_FP_PREC}$$

Mẫu độ chói được ánh xạ nghịch đảo invLumaSample được suy ra như sau:

$$\text{invLumaSample} = \text{Clip1}(\text{invSample}).$$

Trong phương trình ở trên, $\text{Clip1}(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}) - 1, x)$, mà BitDepth là độ sâu bit của dữ liệu cơ bản x . $\text{Clip3}(x, y, z)$ được xác định là:

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{trái lại} \end{cases}.$$

Trường hợp cho phép chia tỷ lệ độ sắc được kích hoạt

Trong trường hợp này, biến số $\text{ChromaScaleCoeff}[i]$, với $i = 0 \dots (((1 \ll \text{bitDepth}) - 1) \gg \text{LOG2_INVERSE_INTERVAL}) - 1$, được suy ra như sau:

if ($\text{InvScaleCoeff}[i] = 0$)

$\text{ChromaScaleCoeff}[i] = (1 \ll \text{CSCALE_FP_PREC})$

else

$\text{ChromaScaleCoeff}[i] = \text{InvScaleCoeff}[i]$.

Trong phương trình ở trên, CSCALE_FP_PREC là giá trị hằng số để chỉ định độ chính xác. Quy trình để tìm kiếm giá trị chia tỷ lệ độ sắc trở thành:

1. Tìm giá trị mẫu độ chói tham chiếu (S)

2. $\text{Idx} = (\text{S} \gg \text{LOG2_INVERSE_INTERVAL})$

3. Hệ số chia tỉ lệ độ sắc (varScale) = $\text{ChromaScaleCoeff}[\text{Idx}]$

Phương pháp 2.-Cấu trúc lên bảng ánh xạ từ piece_reshaped thành piece_original ($\text{MapIdxRtoIdxO}[]$)

Được đề xuất để cấu trúc lên bảng ánh xạ từ pieceIdx_reshaped thành pieceIdx_original theo đường cong ánh xạ được báo hiệu tiền xạ. Mẫu độ chói trong miền được tái định dạng được phân chia thành các khoảng có kích thước bằng $(1 \ll \text{LOG2_INVERSE_INTERVAL})$, và mỗi khoảng có chỉ số đoạn tương ứng trong miền gốc. Kích thước của bảng ánh xạ MapIdxRtoIdxO bằng $((1 \ll \text{bitDepth}) - 1 \gg \text{LOG2_INVERSE_INTERVAL})$, mà bitDepth là độ sâu bit bên trong. Khi tìm kiếm đoạn mà mẫu độ chói để được ánh xạ thuộc về để phép ánh xạ nghịch đảo, chúng ta chỉ cần dịch chuyển sang bên phải mẫu độ chói để được ánh xạ bằng $\text{LOG2_INVERSE_INTERVAL}$, và tìm kiếm chỉ số đoạn tương ứng của nó trong miền gốc mà có thể được tìm thấy bằng cách sử dụng bảng MapIdxRtoIdxO (ví dụ $\text{MapIdxRtoIdxO}[\text{S} \gg \text{LOG2_INVERSE_INTERVAL}]$, mà S là mẫu để được ánh xạ).

Trong phương án khác, cho từng đoạn trong miền được tái định dạng, chỉ số đoạn tương ứng trong miền gốc có thể được suy ra bằng cách sử dụng phần lớn các chỉ số mẫu độ chói' của miền gốc.

Theo phương án này, cho từng đoạn trong miền được tái định dạng, chỉ số đoạn tương ứng trong miền gốc có thể được suy ra bằng cách sử dụng chỉ số của mẫu độ chói thứ nhất miền gốc. Theo một ví dụ, bảng MapIdxRtoIdxO có thể được cấu trúc như dưới đây. Đối với mẫu độ chói bằng $(i \ll \text{LOG2_INVERSE_INTERVAL})$, mà $i = 0 \dots (((1 \ll \text{bitDepth}) - 1) \gg \text{LOG2_INVERSE_INTERVAL})$:

If (i << LOG2_INVERSE_INTERVAL) < LmcsPivot [lmcs_min_bin_idx])

MapIdxRtoIdxO[i] = lmcs_min_bin_idx

If (i << LOG2_INVERSE_INTERVAL) >= LmcsPivot [LmcsMaxBinIdx +1])

MapIdxRtoIdxO[i] = LmcsMaxBinIdx

Otherwise,

MapIdxRtoIdxO[i] = idxYInv

Trong phương trình ở trên, idxYInv được suy ra bằng việc thiết lập lumaSample bằng (i << LOG2_INVERSE_INTERVAL), và theo quy trình sau đây:

```
if ( lumaSample < LmcsPivot[ lmcs_min_bin_idx + 1 ] )
```

```
idxYInv = lmcs_min_bin_idx
```

```
else if ( lumaSample >= LmcsPivot[ LmcsMaxBinIdx ] )
```

```
idxYInv = LmcsMaxBinIdx
```

```
else {
```

```
for( idxYInv = lmcs_min_bin_idx; idxYInv < LmcsMaxBinIdx; idxYInv++ ) {
```

```
if( lumaSample < LmcsPivot [ idxYInv + 1 ] )
```

```
break;
```

```
}
```

```
}
```

Phép ánh xạ nghịch đảo

Biến số idxYInv được suy ra bởi MapIdxRtoIdxO[(lumaSample >> LOG2_INVERSE_INTERVAL)], mà lumaSample là mẫu để được ánh xạ. Biến số invSample được suy ra như sau:

```
invSample = InputPivot[ idxYInv ] + ( InvScaleCoeff[ idxYInv ] *
( lumaSample - LmcsPivot[ idxYInv ] ) + ( 1 << (SCALE_FP_PREC-1) ) ) >>
SCALE_FP_PREC
```

Mẫu độ chói được ánh xạ nghịch đảo invLumaSample được suy ra như sau:

```
invLumaSample = Clip1 ( invSample )
```

Trường hợp phép chia tỷ lệ độ sắc được kích hoạt

Trong trường hợp này, quy trình để tìm kiếm giá trị chia tỷ lệ độ sắc trở thành

1. Tìm giá trị mẫu độ chói tham chiếu (S)

2. $\text{Idx} = \text{MapIdxRtoIdxO}[(S \gg \text{LOG2_INVERSE_INTERVAL})]$

3. Hệ số chia tỷ lệ độ sắc (varScale) = $\text{ChromaScaleCoeff}[\text{Idx}]$

Phương pháp 2.3-So sánh tối đa N lần khi suy ra chỉ số đoạn trong miền gốc

Mẫu độ chói trong miền được tái định dạng được phân chia thành nhiều đoạn với khoảng bằng ($1 \ll \text{LOG2_INVERSE_INTERVAL}$). Đối với từng khoảng, tồn tại N ngưỡng. Bằng cách so sánh mẫu để được ánh xạ với N ngưỡng, chỉ số nghịch đảo có thể được tìm thấy. Để đảm bảo rằng chỉ tồn tại N điểm chuyển trong từng khoảng, có sự tương thích dòng bit mà chỉ tồn tại N điểm chuyển trong từng khoảng. Nếu có nhiều hơn N điểm chuyển trong cùng một khoảng, số lượng của các từ mã mà trong đó khoảng được điều chỉnh.

Khi tìm chỉ số nghịch đảo của mẫu độ chói, chúng ta tìm kiếm đầu tiên chỉ số khoảng trong miền được tái định dạng bởi (mẫu độ chói $\gg \text{LOG2_INVERSE_INTERVAL}$), và sau đó so sánh mẫu độ chói có N ngưỡng mà trong đó khoảng để tìm kiếm chỉ số đoạn chính xác trong miền gốc. Theo phương án này, N bằng 1, 2, 3, 4, ...15.

Phương pháp 3-Phép ánh xạ nghịch đảo được đơn giản hóa – so sánh tối đa N lần khi suy ra chỉ số đoạn trong miền gốc

Để đơn giản hóa quy trình được sử dụng để suy ra chỉ số khoảng, được đề xuất để so sánh tối đa N lần khi tìm chỉ số khoảng. Để đạt được điều này, mẫu độ chói trong miền được tái định dạng được phân chia thành nhiều phân đoạn có kích thước bằng ($1 \ll \text{LOG2_INVERSE_INTERVAL}$). Đối với từng phân đoạn, tồn tại N giá trị ngưỡng ($\text{LmcsSegPivot}[(1 \ll \text{bitDepth}) \gg \text{LOG2_INVERSE_INTERVAL}])$ và một base chỉ số khoảng cơ bản ($\text{idxYBase}[(1 \ll \text{bitDepth}) \gg \text{LOG2_INVERSE_INTERVAL} + 1]$). Theo phương án này, $\text{LOG2_INVERSE_INTERVAL}$ bằng 5 và N bằng 1, tương ứng. các phương án sau đây là để cho trường hợp khi INVERSE_INTERVAL bằng 5 và N bằng 1.

Trong phương án khác, mẫu độ chói trong miền được tái định dạng được phân

chia thành ($1 \leq K$) phân đoạn có kích thước bằng nhau. Đối với từng phân đoạn, tồn tại N giá trị ngưỡng (LmcsSegPivot) và one base chỉ số khoảng (idxYBase).

Theo phương án này, bitDepth bằng 10, K bằng 5 và N bằng 1, tương ứng. Các phương án sau đây là để cho trường hợp khi bitDepth bằng 10, K bằng 5, và N bằng 1.

Trong phương án khác, số các phân đoạn là bội của số các khoảng của đường cong ánh xạ từng đoạn LMCS. Theo phương án này, nếu số lượng của các khoảng của đường cong ánh xạ từng đoạn LMCS bằng X, số các phân đoạn bằng $M \cdot X$, mà M là số nguyên, và có thể là 2, 3, 4, 5, 6, 7..., v.v.

Theo phương án này, M bằng 2, nếu số lượng của các khoảng của đường cong ánh xạ từng đoạn LMCS bằng 16 ($=2^4$), số các phân đoạn bằng 32 ($=2^5$).

Phương pháp 3.1-Phép ánh xạ độ chói nghịch đảo có phân đoạn có kích thước bằng nhau

Đối với từng phân đoạn, chỉ số (idxYSeg), mà được tính toán bằng cách chia mẫu để được ánh xạ cho 32, chỉ định chỉ số của phân đoạn hiện thời. Trong phương án khác, idxYseg được suy ra bằng cách lấy 5 bit quan trọng nhất MSB (most significant bit - MSB) đầu tiên của mẫu độ chói. Trong phương án khác, idxYseg được suy ra bằng cách chia mẫu để được ánh xạ cho ($1 \leq (\text{bitDepth} - K)$). idxYBase [idxYSeg] là chỉ số khoảng của mẫu độ chói được ánh xạ thấp nhất trong phân đoạn với chỉ số bằng idxYSeg. Được suy ra bằng cách so sánh mẫu độ chói được ánh xạ thấp nhất trong phân đoạn hiện thời đến tối đa 15 giá trị LmcsPivot. LmcsSegPivot[idxYSeg] được thiết lập bằng với LmcsPivot[idxYBase [idxYSeg] + 1]. Khi tìm chỉ số khoảng (idxYInv) của mẫu độ chói (lumaSample) bất kỳ, các bước sau được áp dụng.

1. Suy ra idxYSeg bằng cách phân chia lumaSample cho 32.

2. Bằng cách so sánh lumaSample với LmcsSegPivot[idxYSeg], chỉ số khoảng nghịch đảo (idxYInv) có thể được tìm thấy.

```
if ( lumaSample < LmcsPivot[ lmc_min_bin_idx + 1 ] )
```

```
idxYInv = lmc_min_bin_idx
```

```
else if ( lumaSample >= LmcsPivot[ LmcMaxBinIdx ] )
```

```
idxYInv = LmcMaxBinIdx
```

```

else {

idxYInv = (lumaSample >= LmcsSegPivot [idxYSeg])?

idxYBase [idxYSeg] + 1 : idxYBase [idxYSeg]

}

```

Trong phương án khác, khi tìm chỉ số khoảng (tức là, idxYInv) của mẫu độ chói bất kỳ (tức là, lumaSample), các bước sau được áp dụng.

1. Suy ra idxYSeg bằng cách phân chia lumaSample cho 32 hoặc bằng cách lấy 5 MSB đầu tiên của mẫu độ chói.

2. Bằng cách so sánh lumaSample với LmcsSegPivot[idxYSeg], chỉ số khoảng nghịch đảo (idxYInv) có thể được tìm thấy.

$$\text{idxYInv} = (\text{lumaSample} \geq \text{LmcsSegPivot} [\text{idxYSeg}])? \text{idxYBase} [\text{idxYSeg}] + 1 : \text{idxYBase} [\text{idxYSeg}]$$

Khi phép chia tỷ lệ độ sắc được kích hoạt, Bước 1 và Bước 2 được áp dụng để suy ra chỉ số khoảng của mẫu độ chói tham chiếu.

Khi suy ra mẫu độ chói nghịch đảo, các bước sau được áp dụng:

1. Mẫu độ chói nghịch đảo được suy ra như sau:

$$\text{invSample} = \text{InputPivot} [\text{idxYInv}] + (\text{InvScaleCoeff} [\text{idxYInv}] * (\text{lumaSample} - \text{LmcsPivot} [\text{idxYInv}]) + (1 \ll 10)) \gg 11$$

2. Cắt invSample thành giá trị giá trị 10-bit.

Phương pháp 3.2-Phép ánh xạ độ chói nghịch đảo có phân đoạn có kích thước bằng nhau và độ sâu bit được rút ngắn

Theo phương pháp này, độ sâu bit của phép tính toán phép ánh xạ nghịch đảo còn được làm giảm thêm bằng cách hạn chế thuật ngữ (ví dụ, $(\text{lumaSample} - \text{LmcsPivot} [\text{idxYInv}])$) được sử dụng để nhân hệ số chia tỉ lệ trong phạm vi $[-32, 31]$.

Đối với từng phân đoạn, chỉ số (idxYSeg), mà được tính toán bằng cách phân chia mẫu để được ánh xạ cho 32, chỉ định chỉ số của phân đoạn hiện thời. Trong phương án khác, idxYseg được suy ra bằng cách lấy 5 MSB đầu tiên của mẫu độ chói. Trong phương án khác, idxYseg được suy ra bằng cách chia mẫu để được ánh xạ với

$(1 \ll (\text{bitDepth} - K))$. $\text{IdxYBase}[i]$, với $i = 0 \dots ((1 \ll \text{bitDepth}) \gg 5) - 1$, là chỉ số khoảng của mẫu độ chói được ánh xạ thấp nhất trong phân đoạn với chỉ số bằng i . Mỗi một trong $\text{IdxYBase}[i]$ được suy ra bằng cách so sánh mẫu độ chói được ánh xạ thấp nhất trong phân đoạn thứ $(i+1)$ đến tối đa 15 giá trị LmcsPivot . $\text{LmcsSegPivot}[i]$, với $i = 0 \dots ((1 \ll \text{bitDepth}) \gg 5) - 1$ được suy ra như sau. Nếu chỉ có một điểm chuyển trong phân đoạn thứ $(i+1)$, $\text{LmcsSegPivot}[i]$ được thiết lập bằng với $\text{LmcsPivot}[\text{IdxYBase}[i] + 1]$; trái lại, nó được thiết lập bằng $(i+1)*32$ (hoặc bằng $i * (1 \ll (\text{bitDepth} - K))$).

Xem chỉ có một điểm chuyển trong phân đoạn với chỉ số “ i ” có thể được biết đến bằng cách so sánh $\text{IdxYBase}[i]$ và $\text{IdxYBase}[i+1]$. Nếu hai giá trị này là giống nhau, và $\text{LmcsPivot}[\text{LmcsMaxBinIdx} + 1]$ không ở trong phân đoạn hiện thời, hoặc $\text{LmcsPivot}[\text{LmcsMaxBinIdx} + 1]$ ở trong phân đoạn hiện thời và $\text{LmcsPivot}[\text{LmcsMaxBinIdx} + 1]$ bằng $i*32$ (hoặc bằng $i * (1 \ll (\text{bitDepth} - K))$), nó có nghĩa là không có điểm chuyển; trái lại, có một điểm chuyển trong phân đoạn thứ $(i+1)$. $\text{InputSegPivot}[i]$, với $i = 0 \dots ((1 \ll \text{bitDepth}) \gg 5) - 1$ được thiết lập bằng với mẫu độ chói được ánh xạ nghịch đảo của $\text{LmcsSegPivot}[i]$, và phép suy ra tuân theo thiết kế ban đầu. Trong phương án khác, nếu không có điểm chuyển trong phân đoạn hiện thời, $\text{InputSegPivot}[i]$ được thiết lập bằng với mẫu độ chói được ánh xạ nghịch đảo của mẫu độ chói $i*32$, và phép suy ra tuân theo thiết kế ban đầu.

Khi tìm chỉ số khoảng (IdxYInv) của mẫu độ chói bất kỳ (lumaSample), các bước sau được áp dụng.

1. Suy ra IdxYSeg bằng cách phân chia lumaSample cho 32

2. Bằng cách so sánh lumaSample với $\text{LmcsSegPivot}[\text{IdxYSeg}]$, chỉ số khoảng nghịch đảo (IdxYInv) có thể được tìm thấy.

```
if ( lumaSample < LmcsPivot[ lmc_min_bin_idx + 1 ] )
```

```
    IdxYInv = lmc_min_bin_idx
```

```
else if ( lumaSample >= LmcsPivot[ LmcsMaxBinIdx ] )
```

```
    IdxYInv = LmcsMaxBinIdx
```

```
else {
```

```

idxYInv = (lumaSample >= LmcsSegPivot [idxYSeg])? idxYBase [idxYSeg] +
1 :
idxYBase [idxYSeg]
}

```

Trong phương án khác, khi tìm chỉ số khoảng (idxYInv) của mẫu độ chói bất kỳ (lumaSample), các bước sau được áp dụng.

1. Suy ra idxYSeg bằng cách phân chia lumaSample cho 32 hoặc bằng cách lấy 5 MSB của mẫu độ chói

2. Bằng cách so sánh lumaSample với LmcsSegPivot[idxYSeg], chỉ số khoảng nghịch đảo (idxYInv) có thể được tìm thấy.

```

idxYInv = (lumaSample >= LmcsSegPivot [idxYSeg])? idxYBase [idxYSeg] +
1 : idxYBase [idxYSeg]

```

Khi phép chia tỷ lệ độ sắc được kích hoạt, Bước 1 và Bước 2 được áp dụng để suy ra chỉ số khoảng của mẫu độ chói tham chiếu.

Khi suy ra mẫu độ chói nghịch đảo, các bước sau được áp dụng:

1. Mẫu độ chói nghịch đảo được suy ra như sau:

```

idxYSeg = (lumaSample >> 5) hoặc idxYSeg = 5 MSB đầu tiên của
lumaSample

```

```

invSample = InputSegPivot[idxYSeg ] + ( InvScaleCoeff[ idxYInv ] *
( lumaSample – LmcsSegPivot[idxYSeg] ) + ( 1 << 10 ) ) >> 11

```

Trong phương án khác, nếu không có điểm chuyển trong phân đoạn hiện thời, InputSegPivot[i] được thiết lập bằng với mẫu độ chói được ánh xạ nghịch đảo của mẫu độ chói idxYSeg *32, và phép suy ra tuân theo thiết kế ban đầu., phép ánh xạ nghịch đảo có thể được suy ra bởi:

```

invSample = InputSegPivot[idxYSeg ] + ( InvScaleCoeff[ idxYInv ] *
( lumaSample – (idxYSeg << 5) ) + ( 1 << 10 ) ) >> 11

```

2. Cắt invSample thành giá trị 10-bit.

Phương pháp 3.3-Các ràng buộc để đảm bảo rằng chỉ có N điểm chuyển trong một phân đoạn

Để đảm bảo rằng chỉ tồn tại N điểm chuyển trong từng khoảng (hoặc được gọi là từng phân đoạn), có sự tương thích dòng bit hoặc ràng buộc quy chuẩn mà có thể chỉ tồn tại N điểm chuyển trong từng khoảng. Nếu nhiều hơn N điểm chuyển trong cùng một khoảng, số các các từ mã mà trong đó khoảng được điều chỉnh.

Theo phương án này, đối với biến số $LmcsPivot[i]$ với $i = lmc_min_bin_idx \dots LmcsMaxBinIdx$, nó là yêu cầu về tính tương thích dòng bit là $(LmcsPivot[i] \gg 5)$ và $(LmcsPivot[i+1] \gg 5)$ không cần phải giống nhau.

Trong phương án khác, đối với biến số $LmcsPivot[i]$ với $i = lmc_min_bin_idx \dots LmcsMaxBinIdx$, nó là yêu cầu về sự tương thích dòng bit mà $(LmcsPivot[i] \gg LOG2_INVERSE_INTERVAL)$ và $(LmcsPivot[i+1] \gg LOG2_INVERSE_INTERVAL)$ không cần phải giống nhau. Theo phương án này, $LOG2_INVERSE_INTERVAL$ phụ thuộc vào độ sâu bit của mẫu độ chói, và là bằng $bitDepth - 5$. Theo phương án này, $LOG2_INVERSE_INTERVAL$ bằng 5 và N bằng 1.

Việc phát hiện “nhiều hơn một điểm chuyển” cho phân đoạn với $idxYSeg = 0 \dots ((1 \ll bitDepth) \gg 5) - 1$ có thể được thực hiện bởi các bước sau:

Đối với chỉ số khoảng $intervalIdx = 0 \dots 15$

1. $idxLow = LmcsPivot[intervalIdx] \gg 5$

2. $idxHigh = LmcsPivot[intervalIdx + 1] \gg 5$

3. Nếu $idxLow$ bằng $idxHigh$, $LmcsPivot[intervalIdx]$ không bằng $LmcsPivot[intervalIdx+1]$, và $LmcsPivot[intervalIdx]$ không là bội số của 32, sau đó “nhiều hơn một điểm chuyển” được phát hiện.

Khi “nhiều hơn một điểm chuyển” được phát hiện trong khoảng i , $LmcsPivot[i+1]$ cần được sửa đổi như sau.

– $adjVal = (idxHigh + 1) * 32 - LmcsPivot[i+1]$

– $LmcsPivot[i+1] = LmcsPivot[i+1] + adjVal$

– $lmcCW[i] = lmcCW[i] + adjVal$

– Quy trình sau được áp dụng với $j = i+1 \dots 15$ cho đến khi $adjVal$ bằng 0:

if($lmcCW[j] < adjVal$) {

```

lmcscw[j] = 0
adjVal = adjVal - lmcscw[j]
}
else {
    lmcscw[j] = lmcscw[j] - adjVal
    adjVal = 0
}

```

Mặc dù mẫu bên trên là để cho trường hợp khi LOG2_INVERSE_INTERVAL bằng 5 và N bằng 1. Những phương pháp này có thể được mở rộng cho các thiết lập khác nhau của LOG2_INVERSE_INTERVAL và N.

Trong phương án khác, đối với biến số $Lmcspivot[i]$ với $i = lmc_min_bin_idx.. Lmcmaxbinidx$, nó là yêu cầu về sự tương thích dòng bit mà 5 MSB đầu tiên của $Lmcspivot[i]$ và 5 MSB đầu tiên của $Lmcspivot[i + 1]$ không cần phải giống nhau.

Phương pháp 3.4-Các ràng buộc để đảm bảo rằng tất cả mẫu độ chói trong một phân đoạn thuộc về tối đa hai khoảng khác nhau

Để đạt được việc sử dụng chỉ một phép so sánh cho từng mẫu để được ánh xạ khi suy ra chỉ số khoảng, có sự tương thích dòng bit, hoặc bộ mã hóa ràng buộc quy chuẩn hoặc ràng buộc quy chuẩn để đảm bảo rằng tất cả mẫu độ chói trong một phân đoạn thuộc về tối đa hai khoảng khác nhau.

Đối với biến số $Lmcspivot[i]$ với $i = lmc_min_bin_idx.. Lmcmaxbinidx$, nó là yêu cầu về sự tương thích dòng bit mà nếu $Lmcspivot[i]$ không là bội số của 32, sau đó $(Lmcspivot[i] \gg 5)$ và $(Lmcspivot[i + 1] \gg 5)$ không cần phải giống nhau.

Theo ví dụ khác, để đạt được sử dụng chỉ một phép so sánh cho từng mẫu để được ánh xạ khi suy ra chỉ số khoảng, số các các từ mã trong từng khoảng được ràng buộc cho giá trị nhất định. Các phương án sau đây thể hiện sự tương thích dòng bit để đảm bảo rằng trong từng phân đoạn có kích thước bằng $OrgCW/2$, chỉ một điểm chuyển tồn tại.

Theo phương án này, đối với biến số $lmcscw[i]$ với $i = lmc_min_bin_idx.. Lmcmaxbinidx$, nó là yêu cầu về sự tương thích dòng bit mà $lmcscw[i]$ cần lớn

hơn $\text{OrgCW} / 2 - 1$.

Trong phương án khác, đối với biến số $\text{lmcsCW}[i]$ với $i = \text{lmcs_min_bin_idx}.. \text{LmcsMaxBinIdx}$, nó là yêu cầu về sự tương thích dòng bit mà $\text{lmcsCW}[i]$ sẽ lớn hơn $\text{OrgCW} / 2 + 1$.

Phương pháp 4—Phép chia tỷ lệ độ sắc cho CIIP

Theo phương án này, các chế độ khác nhau có thể sử dụng mẫu độ chói tham chiếu trong các vị trí khác nhau.

Trong phương án khác, đối với các khối mà có quy trình dự đoán có thể tham chiếu mẫu được tái cấu trúc lân cận, mẫu độ chói tham chiếu cho phép suy ra giá trị thang đo là từ mẫu được tái cấu trúc lân cận của CU hoặc TU hiện thời. Ví dụ, nếu khối hiện thời là chế độ dự đoán nội, mẫu độ chói được tham chiếu ở mẫu biên dạng tham chiếu trên cùng-bên trái, trên cùng, hoặc bên trái của CU hiện thời, vì vậy, đối với chế độ dự đoán phân vùng phụ nội (ISP), giá trị chia tỷ lệ độ sắc dư được suy ra sử dụng trên cùng-bên trái, trên cùng, hoặc bên trái của mẫu được tái cấu trúc biên dạng hình chữ L của CU/TU hiện thời (không phải là TU phân vùng phụ). Theo ví dụ khác, nếu khối hiện thời là chế độ dự đoán nội, mẫu độ chói được tham chiếu là mẫu biên dạng tham chiếu trên cùng-bên trái của TU hiện thời. Vì vậy, đối với chế độ dự đoán phân vùng phụ nội (ISP), giá trị chia tỷ lệ độ sắc dư được suy ra sử dụng trên cùng-bên trái, trên cùng, hoặc bên trái của mẫu được tái cấu trúc biên dạng hình chữ L của TU hiện thời (phân vùng phụ). Mẫu được tái cấu trúc biên dạng chữ L trên cùng-bên trái có thể là một ví dụ.

Trong phương án khác, nếu khối hiện thời là chế độ dự đoán liên ngoại trừ chế độ CIIP, mẫu độ chói tham chiếu có thể là mẫu dự đoán độ chói trên cùng-bên trái của CU hiện thời hoặc TU. Ví dụ, nếu CU hiện thời là chế độ phép dự đoán liên, nhưng được dự đoán bởi combined Inter Chế độ nội (CIIP) hoặc các phương pháp dự đoán khác mà cần mẫu được tái cấu trúc lân cận, mẫu độ chói được tham chiếu có thể là mẫu được tái cấu trúc biên dạng tham chiếu của CU hoặc TU hiện thời như được mô tả ở bên trên. Ví dụ, mẫu được tái cấu trúc lân cận trái - trên cùng có thể được sử dụng.

Trong phương án khác, nếu khối hiện thời là chế độ dự đoán liên, mẫu độ chói tham chiếu có thể là mẫu dự đoán độ chói trên cùng-bên trái của CU hoặc TU hiện thời. Ví dụ, nếu khối hiện thời là chế độ CIIP, mẫu độ chói tham chiếu có thể là mẫu

dự đoán độ chói trên cùng-bên trái của CU hoặc TU hiện thời.

Trong phương án khác, nếu khối hiện thời là chế độ IBC, quyết định mẫu độ chói tham chiếu giống như chế độ phép dự đoán liên.

Trong phương án khác, nếu khối hiện thời là chế độ IBC, quyết định mẫu độ chói tham chiếu giống như chế độ dự đoán nội.

Bất kỳ phương án nào được đề xuất trước đó có thể được thực hiện trong các bộ mã hóa và/hoặc các bộ giải mã. Ví dụ, bất kỳ phương pháp nào được đề xuất ở trên có thể được thực hiện trong môđun tái định dạng hoặc môđun tái định dạng/chia tỷ lệ dư của bộ mã hóa, và/hoặc môđun tái định dạng hoặc môđun tái định dạng/chia tỷ lệ dư của bộ giải mã. Thay vào đó, bất kỳ phương pháp nào được đề xuất ở trên có thể được thực hiện như mạch được ghép nối với môđun tái định dạng hoặc môđun tái định dạng/chia tỷ lệ dư của bộ mã hóa và/hoặc môđun tái định dạng hoặc môđun tái định dạng/chia tỷ lệ dư của bộ giải mã, để cung cấp thông tin cần thiết bởi môđun tái định dạng hoặc môđun tái định dạng/chia tỷ lệ dư.

Theo một số phương án, các bộ mã hóa video phải tuân theo thiết kế cú pháp trước đó để tạo ra dòng bit hợp pháp, và bộ giải mã video có thể giải mã dòng bit chính xác chỉ khi quy trình phân tích cú pháp được tuân thủ với thiết kế cú pháp bên trên. Khi cú pháp được bỏ qua trong dòng bit, các bộ mã hóa và các bộ giải mã cần thiết lập giá trị cú pháp như giá trị được suy ra đảm bảo các kết quả mã hóa và giải mã được kết nối.

FIG. 4 minh họa sơ đồ hệ thống lập mã điển hình để xác định hoặc suy ra các điểm xoay của phép ánh xạ độ chói nghịch đảo theo phương án của sáng chế. Các bước được thể hiện trong sơ đồ, cũng như các sơ đồ khác trong sáng chế này, có thể được thực hiện như các mã chương trình có thể chạy trên một hoặc nhiều bộ xử lý (ví dụ, một hoặc nhiều CPU) phía bộ mã hóa và/hoặc phía bộ giải mã. Các bước được thể hiện trong sơ đồ cũng có thể được thực hiện dựa trên phần cứng như là một hoặc nhiều thiết bị điện tử hoặc bộ xử lý được sắp xếp để thực hiện các bước trong sơ đồ. Theo phương pháp này, nhiều phân đoạn thứ nhất được xác định trong bước 410, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ nhất được phân chia thành nhiều phân đoạn thứ nhất, và trong đó mỗi phân đoạn thứ nhất tương ứng với một đoạn đường ánh xạ mẫu độ chói từ miền tín hiệu thứ nhất đến miền tín

hiệu thứ hai. Nhiều phân đoạn thứ hai được xác định trong bước 420, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ hai được phân chia thành nhiều phân đoạn thứ hai, và trong đó tất cả phân đoạn thứ hai có cùng kích thước bằng giá trị khoảng mục tiêu. Một hoặc nhiều điểm xoay thứ nhất được xác định hoặc được suy ra trong bước 430, trong đó mỗi điểm xoay thứ nhất được kết hợp với một phân đoạn thứ nhất, và trong đó một hoặc nhiều điểm xoay thứ nhất được xác định hoặc được suy ra sao cho: nếu điểm xoay mục tiêu được ánh xạ tương ứng với một điểm xoay thứ nhất trong miền tín hiệu thứ hai có nhiều giá trị khoảng mục tiêu, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo được cho phép để được đặt trong cùng phân đoạn thứ hai; và trái lại, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo không được cho phép để được đặt trong cùng phân đoạn thứ hai. Mã hóa hoặc giải mã video của mẫu độ chói được thực hiện sử dụng phép ánh xạ tuyến tính từng đoạn để ánh xạ mẫu độ chói mục tiêu từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai, hoặc từ miền tín hiệu thứ hai đến miền tín hiệu thứ nhất trong bước 440.

Sơ đồ được thể hiện nhằm minh họa ví dụ lập mã video theo sáng chế. Người có hiểu biết trong lĩnh vực này có thể biến đổi từng bước, sắp xếp lại các bước, phân tách bước, hoặc kết hợp các bước để thực hiện sáng chế mà không nằm ngoài bản chất của sáng chế. Theo sáng chế, cú pháp và ngữ nghĩa cụ thể được sử dụng để minh họa các ví dụ thực hiện các phương án của sáng chế. Người có hiểu biết trung bình có thể thực hiện sáng chế bằng cách thay thế cú pháp và ngữ nghĩa bằng cú pháp và ngữ nghĩa tương đương mà không nằm ngoài bản chất của sáng chế.

Phần mô tả trên được thể hiện để cho phép người có hiểu biết trung bình trong lĩnh vực này thực hiện sáng chế như được đề xuất về nội dung của ứng dụng cụ thể và yêu cầu của nó. Các biến thể khác nhau cho các phương án được mô tả sẽ rõ ràng cho những người có hiểu biết trung bình trong lĩnh vực này, và các nguyên lý chung được xác định ở đây có thể được áp dụng cho các phương án khác. Vì vậy, sáng chế không nhằm giới hạn các phương án cụ thể được thể hiện và mô tả, nhưng phải được coi là có phạm vi rộng nhất thống nhất với các nguyên lý và các dấu hiệu có tính mới được bộc lộ ở đây. Trong phần mô tả chi tiết bên trên, các chi tiết cụ thể khác nhau được minh họa để giúp hiểu rõ sáng chế. Tuy nhiên, sáng chế sẽ được hiểu bởi người có hiểu biết trung bình trong lĩnh vực này rằng sáng chế có thể được thực hiện

Phương án của sáng chế như được mô tả ở bên trên có thể được thực hiện trong các phần cứng, mã phần mềm khác nhau, hoặc sự kết hợp của cả hai. Ví dụ, phương án của sáng chế có thể là một hoặc nhiều mã được tích hợp vào chip nén video hoặc mã chương trình được tích hợp vào phần mềm nén video để thực hiện việc xử lý được mô tả ở đây. Phương án của sáng chế cũng có thể là mã chương trình để được thực thi trên bộ xử lý tín hiệu số DSP (Digital Signal Processor - DSP) để thực hiện việc xử lý được mô tả ở đây. Sáng chế cũng đề cập đến số các hàm được thực hiện bởi bộ xử lý máy tính, bộ xử lý tín hiệu kỹ thuật số, bộ vi xử lý, hoặc mảng phần tử logic có thể tái lập trình FPGA (field programmable gate array - FPGA). Những bộ xử lý này có thể được tạo cấu hình để thực hiện các tác vụ cụ thể theo sáng chế, bằng các thực thi mã phần mềm hoặc mã sụn có thể đọc được bằng máy tính mà xác định các phương pháp cụ thể được thực hiện bởi sáng chế. Mã phần mềm hoặc mã phần sụn có thể được phát triển theo các ngôn ngữ lập trình khác nhau và các định dạng hoặc kiểu khác nhau. Mã phần mềm cũng có thể được biên dịch cho các nền tảng mục tiêu khác nhau. Tuy nhiên, các định dạng mã khác nhau, các kiểu và ngôn ngữ của mã phần mềm và khác có nghĩa của mã cấu hình để thực hiện các tác vụ theo sáng chế sẽ không nằm ngoài bản chất và phạm vi của sáng chế.

Sáng chế có thể được thực hiện theo các dạng cụ thể khác mà không nằm ngoài bản chất và các đặc điểm cần thiết của nó. Các ví dụ được mô tả là để được xem xét theo tất cả các khía cạnh chỉ nhằm mục đích minh họa và không nhằm giới hạn. Phạm vi của sáng chế vì vậy, được biểu thị bởi các yêu cầu bảo hộ kèm thêm thay vì bởi phần mô tả bên trên. Tất cả các thay đổi mà nằm trong bản chất hoặc phạm vi của các yêu cầu bảo hộ tương đương được coi là nằm trong phạm vi của sáng chế.

Sáng chế yêu cầu hưởng quyền ưu tiên Đơn sáng chế tạm thời Hoa Kỳ số 62/855,007, được nộp vào 31/05/2019, Đơn sáng chế tạm thời Hoa Kỳ số 62/865,390, được nộp vào 24/06/2019, Đơn sáng chế tạm thời Hoa Kỳ số 62/876,094, được nộp vào 19/07/2019 và Đơn sáng chế tạm thời Hoa Kỳ số 62/905,581, được nộp vào 25/09/2019. Các Đơn sáng chế tạm thời Hoa Kỳ được kết hợp ở đây bằng cách tham chiếu đến toàn bộ chúng.

YÊU CẦU BẢO HỘ

1. Phương pháp lập mã video, phương pháp này bao gồm:

xác định nhiều phân đoạn thứ nhất, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ nhất được phân chia thành nhiều phân đoạn thứ nhất, và trong đó mỗi phân đoạn thứ nhất tương ứng với một đoạn đường ánh xạ mẫu độ chói từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai;

xác định nhiều phân đoạn thứ hai, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ hai được phân chia thành nhiều phân đoạn thứ hai, và trong đó tất cả phân đoạn thứ hai có cùng kích thước bằng giá trị khoảng mục tiêu;

xác định hoặc suy ra một hoặc nhiều điểm xoay thứ nhất, trong đó mỗi điểm xoay thứ nhất được kết hợp với một phân đoạn thứ nhất, và trong đó một hoặc nhiều điểm xoay thứ nhất được xác định hoặc được suy ra sao cho: nếu điểm xoay mục tiêu được ánh xạ tương ứng với một điểm xoay thứ nhất trong miền tín hiệu thứ hai có nhiều giá trị khoảng mục tiêu, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo được cho phép để được đặt trong cùng phân đoạn thứ hai; và trái lại, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo không được cho phép để được đặt trong cùng phân đoạn thứ hai; và

thực hiện mã hóa hoặc giải mã video của mẫu độ chói sử dụng phép ánh xạ tuyến tính từng đoạn để ánh xạ mẫu độ chói mục tiêu từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai, hoặc từ miền tín hiệu thứ hai đến miền tín hiệu thứ nhất.

2. Phương pháp theo điểm 1, trong đó phép ánh xạ tuyến tính từng đoạn được kết hợp với nhiều phân đoạn thứ nhất, nhiều phân đoạn thứ hai và một hoặc nhiều điểm xoay thứ nhất.

3. Phương pháp theo điểm 1, trong đó tất cả phân đoạn thứ nhất có cùng giá trị khoảng thứ nhất.

4. Phương pháp theo điểm 3, trong đó giá trị khoảng thứ nhất tương ứng với 2 mũ số định trước thứ nhất và số định trước thứ nhất là số nguyên.

5. Phương pháp theo điểm 4, trong đó số định trước thứ nhất bằng $(\text{bitDepth} - 4)$ và bitDepth đại diện độ sâu bit của mẫu độ chói.

6. Phương pháp theo điểm 1, trong đó giá trị khoảng mục tiêu tương ứng với 2 mũ số

định trước thứ hai và số định trước thứ hai là số nguyên.

7. Phương pháp theo điểm 6, trong đó số định trước thứ hai bằng $(\text{bitDepth} - 5)$ và bitDepth đại diện độ sâu bit của mẫu độ chói.

8. Phương pháp theo điểm 1, trong đó mỗi phân đoạn thứ hai có tối đa hai điểm xoay được ánh xạ.

9. Phương pháp theo điểm 1, trong đó phạm vi mẫu độ chói bằng 0 đến 1023, và giá trị khoảng mục tiêu bằng 32.

10. Thiết bị lập mã video, thiết bị bao gồm một hoặc nhiều mạch điện hoặc bộ xử lý được sắp xếp để:

xác định nhiều phân đoạn thứ nhất, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ nhất được phân chia thành nhiều phân đoạn thứ nhất, và trong đó mỗi phân đoạn thứ nhất tương ứng với một đoạn đường ánh xạ mẫu độ chói từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai;

xác định nhiều phân đoạn thứ hai, trong đó phạm vi mẫu độ chói tương ứng với mẫu độ chói trong miền tín hiệu thứ hai được phân chia thành nhiều phân đoạn thứ hai, và trong đó tất cả phân đoạn thứ hai có cùng kích thước bằng giá trị khoảng mục tiêu;

xác định hoặc suy ra một hoặc nhiều điểm xoay thứ nhất, trong đó mỗi điểm xoay thứ nhất được kết hợp với một phân đoạn thứ nhất, và trong đó một hoặc nhiều điểm xoay thứ nhất được xác định hoặc được suy ra sao cho: nếu điểm xoay mục tiêu được ánh xạ tương ứng với một điểm xoay thứ nhất trong miền tín hiệu thứ hai có nhiều giá trị khoảng mục tiêu, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo được cho phép để được đặt trong cùng phân đoạn thứ hai; và trái lại, điểm xoay mục tiêu được ánh xạ và điểm xoay được ánh xạ tiếp theo không được cho phép để được đặt trong cùng phân đoạn thứ hai; và

thực hiện mã hóa hoặc giải mã video của mẫu độ chói sử dụng phép ánh xạ tuyến tính từng đoạn để ánh xạ mẫu độ chói mục tiêu từ miền tín hiệu thứ nhất đến miền tín hiệu thứ hai, hoặc từ miền tín hiệu thứ hai đến miền tín hiệu thứ nhất.

11. Thiết bị theo điểm 10, trong đó phép ánh xạ tuyến tính từng đoạn được kết hợp với nhiều phân đoạn thứ nhất, nhiều phân đoạn thứ hai và said một hoặc nhiều điểm xoay thứ nhất.

12. Thiết bị theo điểm 10, trong đó tất cả phân đoạn thứ nhất có cùng giá trị khoảng thứ nhất.
13. Thiết bị theo điểm 12, trong đó giá trị khoảng thứ nhất tương ứng với 2 mũ số định trước thứ nhất và số định trước thứ nhất là số nguyên.
14. Thiết bị theo điểm 13, trong đó số định trước thứ nhất bằng $(\text{bitDepth} - 4)$ và bitDepth đại diện độ sâu bit của mẫu độ chói.
15. Thiết bị theo điểm 10, trong đó giá trị khoảng mục tiêu tương ứng với 2 mũ số định trước thứ hai và số định trước thứ hai là số nguyên.
16. Thiết bị theo điểm 15, trong đó số định trước thứ hai bằng $(\text{bitDepth} - 5)$ và bitDepth đại diện độ sâu bit của mẫu độ chói.
17. Thiết bị theo điểm 10, trong đó each phân đoạn thứ hai có tối đa hai điểm xoay được ánh xạ.
18. Thiết bị theo điểm 10, trong đó phạm vi mẫu độ chói bằng 0 đến 1023, và giá trị khoảng mục tiêu bằng 32.

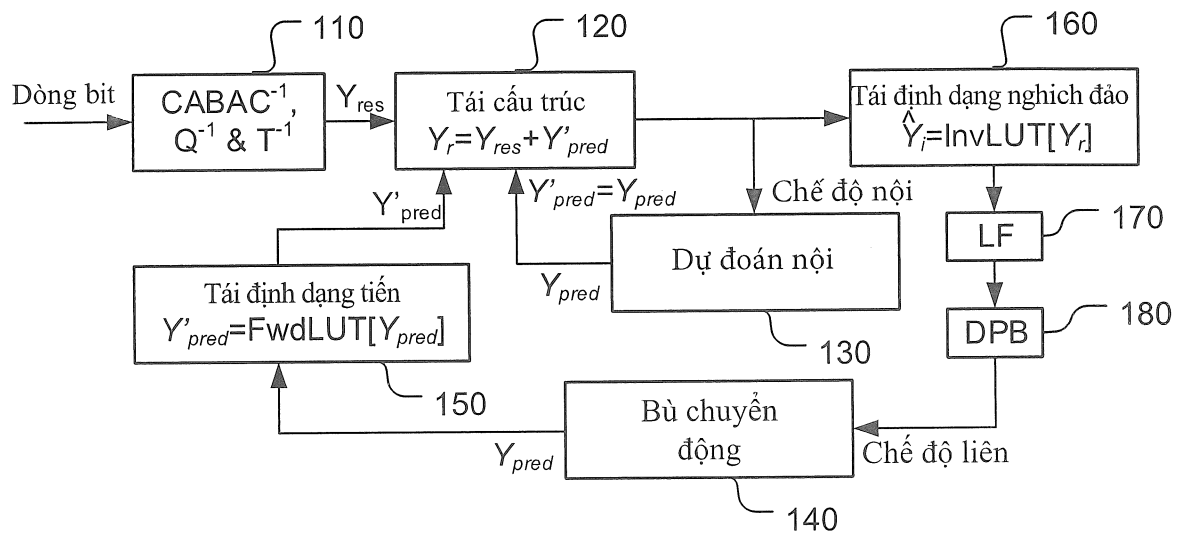


FIG. 1

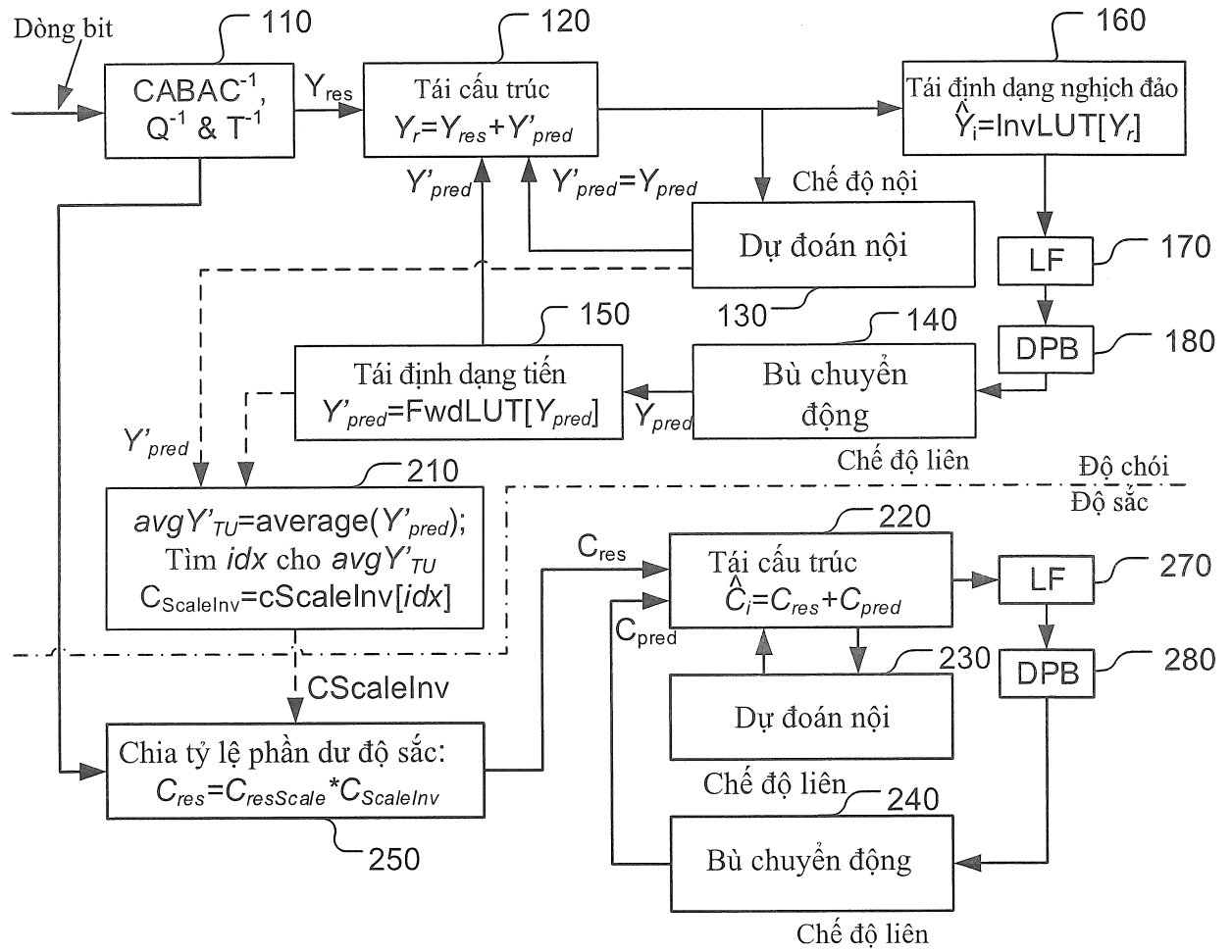


FIG. 2

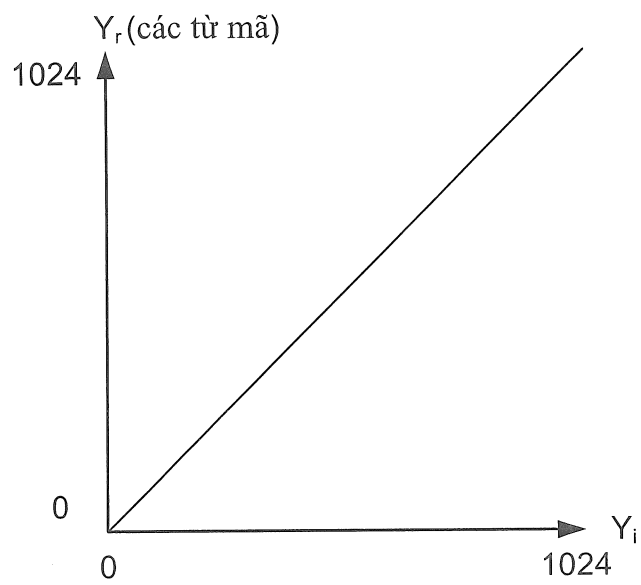


FIG. 3A

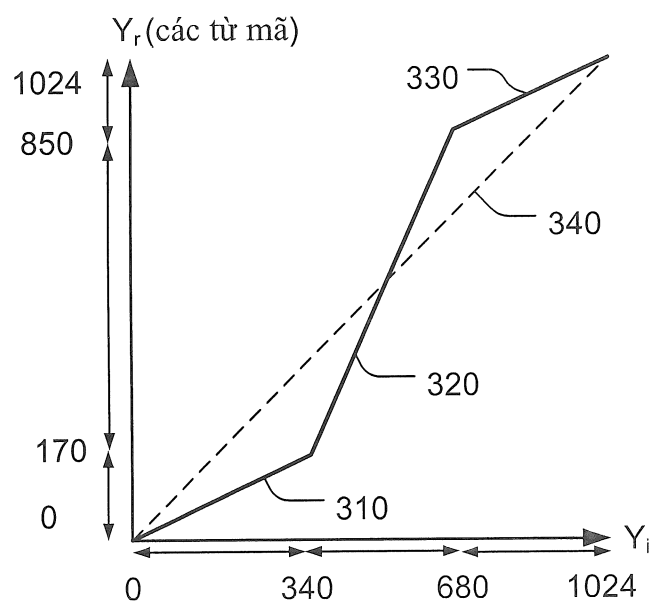


FIG. 3B

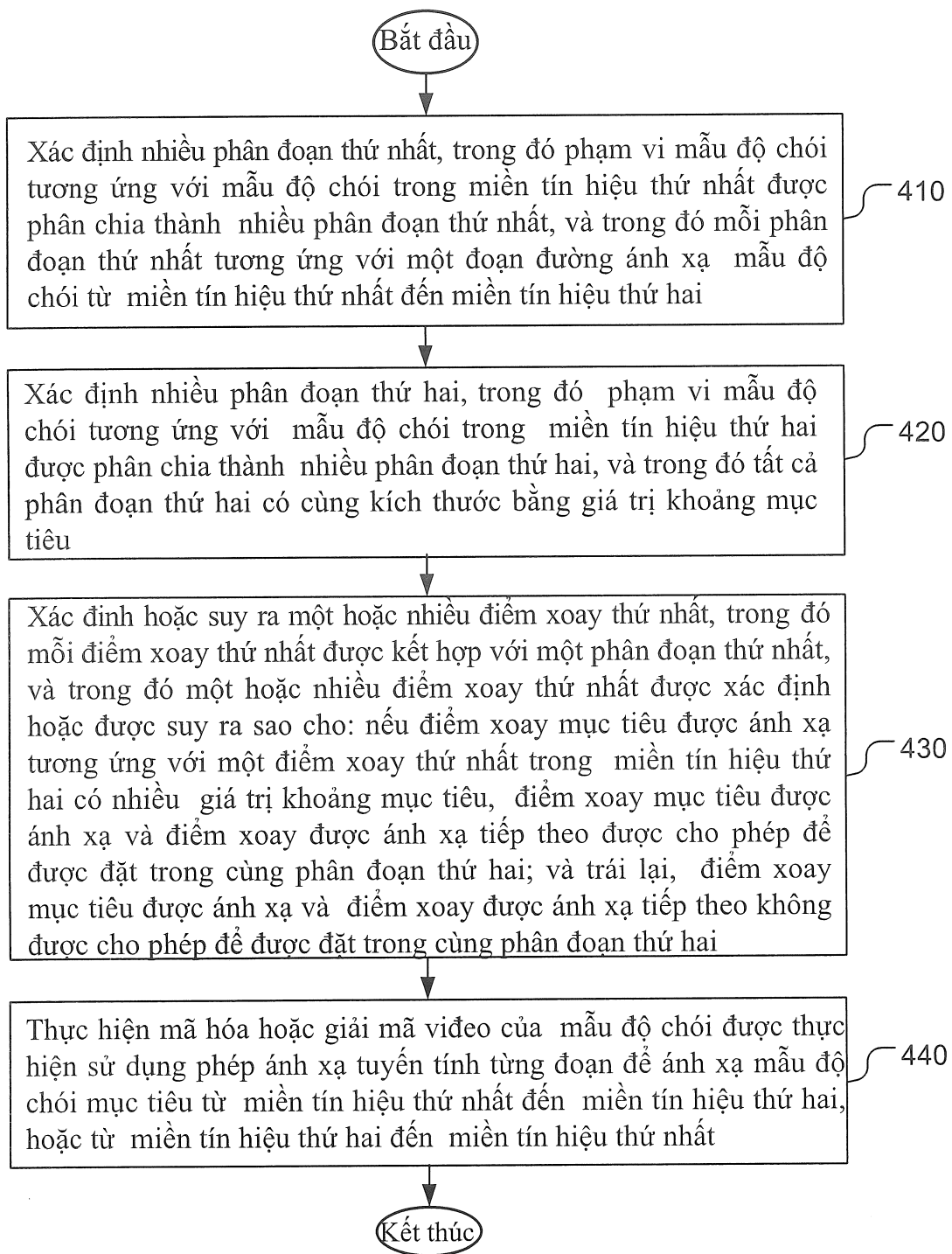


FIG. 4