



(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)



1-0049410

(51)^{2021.01} H04N 19/70

(13) B

(21) 1-2022-04883

(22) 11/12/2020

(86) PCT/CN2020/135802 11/12/2020

(87) WO2021/143413 22/07/2021

(30) 62/960,782 14/01/2020 US; 17/117,757 10/12/2020 US

(45) 25/08/2025 449

(43) 25/10/2022 415A

(73) HFI INNOVATION INC. (TW)

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

(72) CHUANG, Tzu-Der (TW); CHEN, Ching-Yeh (TW).

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

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ LẬP MÃ CHUỖI VIDEO

(21) 1-2022-04883

(57) Sáng chế đề xuất phương pháp và thiết bị lập mã chuỗi video, trong đó chế độ tái tạo mẫu hình ảnh tham chiếu (Reference Picture Resampling: RPR) được bao gồm dưới dạng công cụ lập mã. Theo phương pháp này, luồng bit tương ứng với dữ liệu được mã hóa của chuỗi video được tạo ra hoặc nhận, tại đó luồng bit bao gồm một hoặc nhiều cú pháp liên quan đến chế độ RPR khi chế độ RPR được kích hoạt, và thông tin chia tỷ lệ cho chế độ RPR được suy ra sử dụng thông tin thứ nhất bao gồm một hoặc nhiều cú pháp này và trong đó một hoặc nhiều cú pháp này được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói - chroma}))$. Hình ảnh đích của chuỗi video được mã hóa hoặc được giải mã sử dụng thông tin chia tỷ lệ khi chế độ RPR được kích hoạt cho hình ảnh đích.

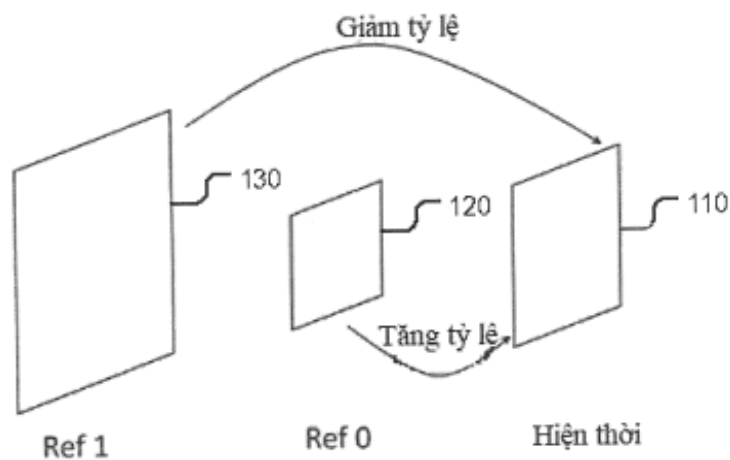


Fig. 1

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lập mã video kết hợp công cụ lập mã tái tạo mẫu hình ảnh tham chiếu (reference picture Resampling-RPR). Cụ thể là, sáng chế đề cập đến phương pháp và thiết bị lập mã chuỗi video.

Tình trạng kỹ thuật của sáng chế

Tiêu chuẩn lập mã video hiệu năng cao (High Efficiency Video Coding: HEVC) đã được phát triển theo dự án video chung của các tổ chức tiêu chuẩn hóa của nhóm chuyên gia lập mã video ITU-T (ITU-T Video Coding Experts Group: VCEG) và nhóm chuyên gia hình ảnh động ISO/IEC (ISO/IEC Moving Picture Experts Group: MPEG), và đặc biệt là với quan hệ đối tác được gọi là nhóm cộng tác chung về lập mã video (Joint Collaborative Team on Video Coding: JCT-VC). Sự phát triển tiêu chuẩn lập mã video mới nổi, được đặt tên là lập mã video đa năng (VVC), đã diễn ra vào năm gần đây như một thể hệ lập mã video tiếp theo ngoài HEVC. VVC hỗ trợ tái tạo cấu trúc hình ảnh tham chiếu (Reference Picture Resampling: RPR) làm công cụ cho các dịch vụ phát trực tuyến tương thích để hỗ trợ bù chuyển động quá trình tạo mẫu lên và gỡ mẫu xuống. Các công nghệ liên quan đến dịch vụ phát trực tuyến tương thích được xem xét như sau.

Tái tạo mẫu hình ảnh tham chiếu (Reference Picture Resampling-RPR)

Trong suốt sự phát triển của VVC, theo “Yêu cầu cho tiêu chuẩn lập mã video tương lai”, “tiêu chuẩn này sẽ hỗ trợ chuyển đổi trình chiếu nhanh trong trường hợp dịch vụ phát trực tuyến tương thích cung cấp nhiều trình chiếu của cùng một nội dung, mỗi trình chiếu có các đặc tính khác nhau (ví dụ, độ phân giải không gian hoặc độ sâu bit mẫu).” Trong truyền thông video thời gian thực, cho phép độ phân giải thay đổi trong chuỗi video được lập mã mà không cần chèn hình ảnh I có thể chỉ không tương thích dữ liệu video vào các điều kiện kênh động hoặc sở thích của người dùng một cách liền mạch, mà còn loại bỏ hiệu ứng vụn (beating) do ảnh I gây ra. Ví dụ giả định về thay đổi độ phân giải tương thích (Adaptive Resolution Change: ARC) với tái tạo mẫu hình ảnh tham chiếu (RPR) như được thể hiện trong Fig. 1, trong đó hình ảnh hiện thời (110) được dự đoán từ hình ảnh tham chiếu (Ref0

120 và Ref1 130) có các kích thước khác nhau. Như được thể hiện trong Fig. 1, hình ảnh tham chiếu Ref0 (120) có độ phân giải thấp hơn so với hình ảnh hiện thời (110). Để sử dụng hình ảnh tham chiếu Ref0 làm tham chiếu, Ref0 phải được chia tỷ lệ tăng lên đến cùng một độ phân giải với hình ảnh hiện thời. Hình ảnh tham chiếu Ref1 (130) có độ phân giải cao hơn so với hình ảnh hiện thời (110). Để sử dụng hình ảnh tham chiếu Ref1 làm tham chiếu, Ref1 phải giảm chia tỷ lệ đến cùng một độ phân giải với hình ảnh hiện thời.

Để hỗ trợ khả năng chia tỷ lệ không gian, kích thước hình ảnh của hình ảnh tham chiếu có thể khác với hình ảnh hiện thời, mà hữu ích để cho các ứng dụng phát trực tuyến. Phương pháp hỗ trợ tái tạo mẫu hình ảnh tham chiếu (RPR), mà cũng được đề cập đến là sự thay đổi độ phân giải tương thích (Adaptive Resolution Change: ARC), đã được nghiên cứu để đưa vào đặc điểm kỹ thuật VVC. Tại cuộc họp JVET lần thứ 14 ở Geneva, một số đóng góp về RPR đã được đưa ra để thảo luận trong cuộc họp.

Khi RPR được sử dụng, tỷ lệ kích thước hình ảnh được suy ra từ chiều rộng và chiều cao hình ảnh tham chiếu và chiều rộng và chiều cao hình ảnh hiện thời. Tỷ lệ kích thước hình ảnh được ràng buộc trong phạm vi $[1/8 \text{ đến } 2]$. Theo cách khác, tỷ lệ kích thước hình ảnh ở giữa bao gồm cả $1/8$ và 2 . Chiều rộng/chiều cao hình ảnh trong các mẫu luma có thể được báo hiệu trong luồng bit, như PPS, và ngữ nghĩa được thể hiện như sau.

pic_width_in_luma_samples chỉ chiều rộng của mỗi hình ảnh được giải mã đề cập đến PPS theo đơn vị các mẫu luma. **pic_width_in_luma_samples** sẽ không bằng 0, sẽ là bội số nguyên của $\text{Max}(8, \text{MinCbSizeY})$, và sẽ nhỏ hơn hoặc bằng **pic_width_max_in_luma_samples**.

Khi **subpics_present_flag** bằng 1 hoặc **ref_pic_resampling_enabled_flag** bằng 0, giá trị của **pic_width_in_luma_samples** sẽ bằng **pic_width_max_in_luma_samples**.

pic_height_in_luma_samples chỉ chiều cao của mỗi hình ảnh được giải mã đề cập đến PPS theo đơn vị của các mẫu luma. **pic_height_in_luma_samples** sẽ không bằng 0 và sẽ là bội số nguyên của $\text{Max}(8, \text{MinCbSizeY})$, và sẽ nhỏ hơn hoặc bằng **pic_height_max_in_luma_samples**.

Khi **subpics_present_flag** bằng 1 hoặc **ref_pic_resampling_enabled_flag** bằng 0, giá

trị của `pic_height_in_luma_samples` shall be bằng `pic_height_max_in_luma_samples`.

Khi kích thước hình ảnh của hình ảnh hiện thời và hình ảnh tham chiếu được chỉ định, ràng buộc sau sẽ được thỏa mãn. Ràng buộc này giới hạn tỷ lệ kích thước hình ảnh của hình ảnh tham chiếu với hình ảnh hiện thời trong phạm vi $[1/8, 2]$.

Đặt `refPicWidthInLumaSamples` và `refPicHeightInLumaSamples` lần lượt là `pic_width_in_luma_samples` và `pic_height_in_luma_samples`, của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu về sự phù hợp luồng bit mà tất cả trong số các điều kiện sau được thỏa mãn:

- `pic_width_in_luma_samples * 2` sẽ lớn hơn hoặc bằng `refPicWidthInLumaSamples`.
- `pic_height_in_luma_samples * 2` sẽ lớn hơn hoặc bằng `refPicHeightInLumaSamples`.
- `pic_width_in_luma_samples` sẽ nhỏ hơn hoặc bằng `refPicWidthInLumaSamples * 8`.
- `pic_height_in_luma_samples` sẽ nhỏ hơn hoặc bằng `refPicHeightInLumaSamples * 8`.

Trong VVC, tỷ lệ chia và sai số chia tỷ lệ cho RPR được suy ra từ thông tin cú pháp được báo hiệu trong PPS. Cú pháp PPS như được thể hiện trong bảng sau.

Bảng 1. Cú pháp RBSP thiết lập thông số hình ảnh cho tỷ lệ chia và sai số chia tỷ lệ

<code>pic_parameter_set_rbsp() {</code>	Biến mô tả
<code>pps_pic_parameter_set_id</code>	<code>ue(v)</code>
<code>pps_seq_parameter_set_id</code>	<code>u(4)</code>
<code>pic_width_in_luma_samples</code>	<code>ue(v)</code>
<code>pic_height_in_luma_samples</code>	<code>ue(v)</code>

conformance_window_flag	u(1)
if(conformance_window_flag) {	
conf_win_left_offset	ue(v)
conf_win_right_offset	ue(v)
conf_win_top_offset	ue(v)
conf_win_bottom_offset	ue(v)
}	
scaling_window_flag	u(1)
if(scaling_window_flag) {	
scaling_win_left_offset	ue(v)
scaling_win_right_offset	ue(v)
scaling_win_top_offset	ue(v)
scaling_win_bottom_offset	ue(v)
}	
...	

Ngữ nghĩa của cú pháp được mô tả như sau.

scaling_window_flag bằng 1 chỉ định rằng thông số sai số cửa sổ chia tỷ lệ được thể hiện trong PPS. **scaling_window_flag** bằng 0 chỉ định rằng thông số sai số cửa sổ chia tỷ lệ không có mặt trong PPS. Khi **ref_pic_resampling_enabled_flag** bằng 0, giá trị của **scaling_window_flag** sẽ bằng 0.

scaling_win_left_offset, **scaling_win_right_offset**, **scaling_win_top_offset**, và

scaling_win_bottom_offset chỉ định sai số, trong các đơn vị của các mẫu luma, mà được áp dụng cho kích thước hình ảnh để tính toán tỷ lệ chia. Khi **scaling_window_flag** bằng 0, giá trị của **scaling_win_left_offset**, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** được suy ra là bằng 0.

Giá trị của **scaling_win_left_offset** + **scaling_win_right_offset** sẽ nhỏ hơn so với **pic_width_in_luma_samples**, và giá trị của **scaling_win_top_offset** + **scaling_win_bottom_offset** sẽ nhỏ hơn so với **pic_height_in_luma_samples**.

Các biến **PicOutputWidthL** và **PicOutputHeightL** được suy ra như sau:

$$\text{PicOutputWidthL} = \text{pic_width_in_luma_samples} - (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}).$$

$$\text{PicOutputHeightL} = \text{pic_height_in_luma_samples} - (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}).$$

fRefWidth được thiết lập bằng **PicOutputWidthL** của hình ảnh tham chiếu **RefPicList[i][j]** trong các mẫu luma và **fRefHeight** được thiết lập bằng **PicOutputHeightL** của hình ảnh tham chiếu **RefPicList[i][j]** trong các mẫu luma.

$$\text{RefPicScale}[i][j][0] = ((\text{fRefWidth} \ll 14) + (\text{PicOutputWidthL} \gg 1)) / \text{PicOutputWidthL}.$$

$$\text{RefPicScale}[i][j][1] = ((\text{fRefHeight} \ll 14) + (\text{PicOutputHeightL} \gg 1)) / \text{PicOutputHeightL}.$$

$$\text{RefPicIsScaled}[i][j] = (\text{RefPicScale}[i][j][0] \neq (1 \ll 14)) \mid (\text{RefPicScale}[i][j][1] \neq (1 \ll 14)).$$

Trong khi RPR thêm tính linh hoạt cho luồng bit video được lập mã, bù chuyển động kết hợp với chia tỷ lệ RPR làm tăng sự phức tạp trong tính toán cũng như băng thông bộ nhớ. Để giảm bớt băng thông bộ nhớ trong trường hợp xấu nhất, sáng chế bộc lộ các phương pháp và thiết bị để hạn chế các tham số liên quan đến RPR.

Bản chất kỹ thuật của sáng chế

Phương pháp và thiết bị lập mã chuỗi video để giảm bớt sự gia tăng băng thông bộ nhớ trong trường hợp xấu nhất liên quan đến bù chuyển động, được bộc lộ, trong đó chế độ tái tạo hình ảnh tham chiếu (Reference Picture Resampling: RPR) được hỗ trợ. Theo phương pháp này, luồng bit tương ứng với dữ liệu được mã hóa của chuỗi video được tạo ra ở phía bộ mã hóa hoặc nhận được ở phía bộ giải mã, trong đó luồng bit bao gồm một hoặc nhiều cú pháp liên quan đến chế độ RPR khi chế độ RPR được kích hoạt, và thông tin chia tỷ lệ cho chế độ RPR được suy ra sử dụng thông tin thứ nhất bao gồm một hoặc nhiều cú pháp, và trong đó một hoặc nhiều cú pháp được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$. Hình ảnh đích của chuỗi video được mã hóa, ở phía bộ mã hóa, hoặc được giải mã ở phía bộ giải mã, sử dụng thông tin chia tỷ lệ khi chế độ RPR được kích hoạt cho hình ảnh đích. Theo một phương án, kích thước khối lập mã tối thiểu cho thành phần chói được chỉ ra bởi cú pháp `MinCbSizeY`.

Theo một phương án, cú pháp bao gồm cú pháp thứ nhất tương ứng với chiều rộng hình ảnh hiện thời trong các mẫu luma và cú pháp thứ hai tương ứng với chiều rộng hình ảnh tối đa trong các mẫu luma. Theo một phương án, cú pháp được ràng buộc để làm cho $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều rộng hình ảnh tối đa} * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời})$.

Theo một phương án, cú pháp bao gồm cú pháp thứ nhất tương ứng với chiều cao hình ảnh hiện thời trong các mẫu luma và cú pháp thứ hai tương ứng với chiều cao hình ảnh tối đa trong các mẫu luma. Theo một phương án, cú pháp được ràng buộc để làm cho $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều cao hình ảnh tối đa} * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời})$.

Theo một phương án, luồng bit tuân thủ yêu cầu tương thích luồng bit mà một hoặc nhiều cú pháp được ràng buộc bằng cách bỏ qua bộ lọc nội suy cho bù chuyển động, và một hoặc nhiều cú pháp được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$.

Theo một phương án, luồng bit tuân thủ yêu cầu tương thích luồng bit mà (chiều rộng

hình ảnh tối đa * chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời) lớn hơn hoặc bằng $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

Theo một phương án, luồng بیت tuân thủ yêu cầu tương thích luồng بیت rằng (chiều cao hình ảnh tối đa * chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời) lớn hơn hoặc bằng $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

Mô tả vắn tắt các hình vẽ

Fig. 1 minh họa ví dụ giả định về sự thay đổi độ phân giải tương thích (Adaptive Resolution Change: ARC) với tái tạo mẫu hình ảnh tham chiếu (Reference Picture Resampling: RPR), trong đó hình ảnh hiện thời được dự đoán từ các hình ảnh tham chiếu (Ref0 và Ref1) với các kích thước khác nhau.

Fig. 2 minh họa sơ đồ khối điển hình về hệ thống kết hợp với các thông số tái tạo mẫu hình ảnh tham chiếu được ràng buộc theo một phương án của sáng chế.

Mô tả chi tiết sáng chế

Mô tả sau là phương án tốt nhất thực hiện sáng chế. Phần mô tả này được tạo ra với mục đích minh họa nguyên tắc chung của sáng chế và không nên giới hạn. Phạm vi của sáng chế được xác định tốt nhất bằng cách tham chiếu đến các yêu cầu bảo hộ đi kèm.

Có thể hiểu một cách dễ dàng rằng các thành phần của sáng chế, như được mô tả và minh họa chung trong các hình ở đây, có thể được sắp xếp và thiết kế theo nhiều dạng cấu hình khác nhau. Do đó, mô tả chi tiết hơn sau đây về các phương án của hệ thống và phương pháp của sáng chế, như được trình bày trong các hình, không nhằm giới hạn phạm vi của sáng chế, như được yêu cầu bảo hộ, mà chỉ là đại diện cho các phương án được chọn của sáng chế.

Trong toàn bộ bản mô tả này, việc tham chiếu đến “một phương án”, “phương án” hoặc tương tự có nghĩa là một tính năng, cấu trúc hoặc đặc tính cụ thể được mô tả liên quan đến phương án này có thể được đưa vào ít nhất một phương án của sáng chế. Do đó, sự xuất hiện của các cụm từ “theo một phương án” hoặc “trong phương án” ở nhiều nơi khác nhau

trong toàn bộ bản mô tả này không nhất thiết phải đề cập đến cùng một phương án.

Hơn nữa, các tính năng, cấu trúc hoặc đặc điểm được mô tả có thể được kết hợp theo bất kỳ cách nào phù hợp theo một hoặc nhiều phương án. Tuy nhiên, người có hiểu biết trung bình trong lĩnh vực này sẽ nhận ra rằng sáng chế có thể được thực hiện mà không có một hoặc nhiều chi tiết cụ thể, hoặc với các phương pháp, thành phần khác, v.v. Trong các trường hợp khác, các cấu trúc hoặc hoạt động đã biết rõ không được hiển thị hoặc được mô tả chi tiết để tránh làm mờ các khía cạnh của sáng chế.

Các phương án được minh họa của sáng chế sẽ được hiểu rõ nhất bằng cách tham chiếu đến các hình vẽ, trong đó các bộ phận giống nhau được chỉ định bằng các chữ số giống nhau. Mô tả sau đây chỉ nhằm mục đích làm ví dụ, và chỉ đơn giản là minh họa cho một số phương án được chọn nhất định của thiết bị và phương pháp phù hợp với sáng chế như được yêu cầu bảo hộ ở đây.

Trong phần mô tả, các số tham chiếu giống nhau trong các hình vẽ và phần mô tả để chỉ các chi tiết tương tự hoặc giống nhau trong các hình chiếu khác nhau.

Theo VVC, tỷ lệ chia hình ảnh tham chiếu suy ra ($\text{RefPicScale}[i][j][0]$, $\text{RefPicScale}[i][j][1]$) được sử dụng để bù chuyển động. RefPicScale được suy ra từ kích thước/chiều rộng/chiều cao của số chia tỷ lệ được chỉ định trong PPS. Nó ảnh hưởng đến bộ lọc nào sẽ được sử dụng trong giai đoạn bù chuyển động và cũng ảnh hưởng đến băng thông bộ nhớ được sử dụng cho bù chuyển động. Ví dụ, đối với khối 16×16 , khi thực hiện bù chuyển động, nó yêu cầu khối tham chiếu $(16 + L - 1) \times (16 + L - 1)$ nếu tỷ lệ chia bằng 1 (ví dụ, $\text{RefPicScale}[i][j][0]$ và $\text{RefPicScale}[i][j][1]$ tất cả bằng 16384), trong đó L chiều dài bộ lọc (filter-tap) của bù chuyển động. Trong SPS, kích thước hình ảnh tối đa của chuỗi được chỉ định. Theo kích thước hình ảnh tối đa, băng thông trường hợp xấu nhất có thể được tính toán và được ràng buộc. Tuy nhiên, khi tỷ lệ chia bằng 2 (ví dụ, $\text{RefPicScale}[i][j][0]$ và $\text{RefPicScale}[i][j][1]$ tất cả đều bằng 32768), nó yêu cầu khối tham chiếu $(32 + L - 1) \times (32 + L - 1)$. Băng thông gần gấp bốn lần khi xem xét ảnh hưởng của độ dài bộ lọc. Băng thông yêu cầu bị ảnh hưởng bởi tỷ lệ chia. Ví dụ, nếu hình ảnh hiện thời có kích thước bằng kích thước hình ảnh tối đa, và tỷ lệ chia của một trong số hình ảnh tham chiếu lớn hơn 1, băng thông yêu cầu của hình ảnh hiện thời này có thể lớn hơn những

gì mà hệ thống mong đợi. Để hạn chế bằng thông trường hợp xấu nhất, vài phương pháp được đề xuất.

Phương pháp-1: Ràng buộc vào tỷ lệ chia

Trong sáng chế này, tỷ lệ chia của (kích thước hình ảnh tham chiếu với kích thước hình ảnh hiện thời) được ràng buộc để không lớn hơn tỷ lệ của (kích thước hình ảnh tối đa trong SPS với kích thước hình ảnh hiện thời). Ví dụ, $\text{scaling_ratio_x} * \text{current_picture_width} * \text{scaling_ratio_y} * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_width} * \text{max_picture_height}$. $\text{Current_picture_width}$ hoặc height có thể là chiều rộng hoặc chiều cao của hình ảnh (ví dụ, được báo hiệu hoặc suy ra trong PPS), hoặc cửa sổ tuần thủ (ví dụ, được báo hiệu hoặc suy ra trong PPS), hoặc cửa sổ chia tỷ lệ (ví dụ, được báo hiệu hoặc suy ra trong PPS). Max_picture_width hoặc $\text{max_picture_height}$ có thể là chiều rộng hoặc chiều cao hình ảnh tối đa của chuỗi hiện thời (ví dụ, được báo hiệu hoặc suy ra trong SPS). Theo một phương án, scaling_ratio_x và scaling_ratio_y có thể là $\text{RefPicIsScaled}[][][0]$ và $\text{RefPicIsScaled}[][][1]$. Ví dụ, $\text{RefPicIsScaled}[][][0] * \text{current_picture_width} * \text{RefPicIsScaled}[][][1] * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_width} * \text{max_picture_height} * 2^K$, hoặc $((\text{RefPicIsScaled}[][][0] * \text{RefPicIsScaled}[][][1] * \text{current_picture_width} * \text{current_picture_height}) >> K)$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_width} * \text{max_picture_height}$. K có thể là 28.

Theo phương án khác, tỷ lệ chia theo chiều ngang và tỷ lệ chia theo chiều dọc có thể được ràng buộc riêng rẽ. Ví dụ, tỷ lệ chia theo chiều ngang và chiều dọc của (kích thước hình ảnh tham chiếu với kích thước hình ảnh hiện thời) được ràng buộc để không lớn hơn tỷ lệ của (chiều rộng hình ảnh tối đa trong SPS với chiều rộng hình ảnh hiện thời) và tỷ lệ của (chiều cao hình ảnh tối đa trong SPS với chiều cao hình ảnh hiện thời) tương ứng. Ví dụ, $\text{scaling_ratio_x} * \text{current_picture_width}$ sẽ nhỏ hơn so với hoặc bằng max_picture_width . $\text{scaling_ratio_y} * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_height}$. $\text{current_picture_width}$ hoặc height có thể là chiều rộng hoặc chiều cao (ví dụ, được báo hiệu hoặc suy ra trong PPS), hoặc cửa sổ tuần thủ (ví dụ, được báo hiệu hoặc suy ra trong PPS), hoặc cửa sổ chia tỷ lệ (ví dụ, được báo hiệu hoặc suy ra trong PPS). max_picture_width hoặc $\text{max_picture_height}$ có thể là chiều rộng hoặc chiều cao hình ảnh tối đa của chuỗi hiện

thời (ví dụ, được báo hiệu hoặc suy ra trong SPS). Theo một phương án, scaling_ratio_x và scaling_ratio_y có thể là $\text{RefPicIsScaled}[][][0]$ và $\text{RefPicIsScaled}[][][1]$. Ví dụ, $\text{RefPicIsScaled}[][][0] * \text{current_picture_width}$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_width} * 2^K$, hoặc $((\text{RefPicIsScaled}[][][0] * \text{current_picture_width}) \gg K)$ sẽ nhỏ hơn so với hoặc bằng max_picture_width . Tương tự, $\text{RefPicIsScaled}[][][1] * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_height} * 2^K$, hoặc $((\text{RefPicIsScaled}[][][1] * \text{current_picture_height}) \gg K)$ sẽ nhỏ hơn so với hoặc bằng $\text{max_picture_height}$. K có thể là 14.

Theo phương án khác, kích thước của cửa sổ chia tỷ lệ, kích thước hình ảnh hiện thời, kích thước hình ảnh tham chiếu, và/hoặc kích thước hình ảnh tối đa trong chuỗi hiện thời được ràng buộc. Ví dụ, đặt $\text{refPicOutputWidthL}$ và $\text{refPicOutputHeightL}$ lần lượt là PicOutputWidthL và PicOutputHeightL , của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu của tuân thủ luồng bit mà tất cả điều kiện sau được thỏa mãn:

- $\text{refPicOutputWidthL} * \text{current_picture_width} * \text{refPicOutputHeightL} * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{PicOutputWidthL} * \text{max_picture_width} * \text{PicOutputHeightL} * \text{max_picture_height}$.

Theo phương án khác, chiều rộng và chiều cao của cửa sổ chia tỷ lệ, chiều rộng và chiều cao của hình ảnh hiện thời, chiều rộng và chiều cao của hình ảnh tham chiếu, và/hoặc chiều rộng và chiều cao tối đa của hình ảnh trong chuỗi hiện thời được ràng buộc riêng rẽ. Ví dụ, let $\text{refPicOutputWidthL}$ và $\text{refPicOutputHeightL}$ là PicOutputWidthL và PicOutputHeightL tương ứng của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu của tuân thủ luồng bit mà tất cả các điều kiện sau được thỏa mãn:

- $\text{refPicOutputWidthL} * \text{current_picture_width}$ sẽ nhỏ hơn so với hoặc bằng $\text{PicOutputWidthL} * \text{max_picture_width}$,
- $\text{refPicOutputHeightL} * \text{current_picture_height}$ sẽ nhỏ hơn so với hoặc bằng $\text{PicOutputHeightL} * \text{max_picture_height}$.

$\text{Current_picture_width}$ và $\text{current_picture_height}$ có thể là chiều rộng và chiều cao

được báo hiệu trong SPS hoặc PPS. Ví dụ, chiều rộng và chiều cao hình ảnh có thể là `pic_width_in_luma_samples` và `pic_height_in_luma_samples`, hoặc có thể là chiều rộng và chiều cao của cửa sổ cắt xén tuân thủ, hoặc có thể là chiều rộng và chiều cao của cửa sổ chia tỷ lệ.

Theo phương án khác, bộ lọc nội suy được xem xét. Bằng thông bộ nhớ MC trường hợp xấu nhất của hình ảnh hiện thời là bằng $\text{CurPicWidth} * \text{CurPicHeight} * \text{WorstCaseBlockBW} / \text{WorstCaseBlockSize}$. Do đó, nếu điều kiện sau được thỏa mãn, bằng thông bộ nhớ MC trường hợp xấu nhất không tăng.

$$\text{CurPicWidth} * \text{CurPicHeight} * (8 * \text{ScalingRatioX} + 7) * (8 * \text{ScalingRatioY} + 7) / (8 * 8) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (8 + 7) * (8 + 7) / (8 * 8).$$

Trong phương trình trên, ScalingRatioX bằng $(\text{refPicOutputWidthL} / \text{PicOutputWidthL})$, ScalingRatioY là bằng $(\text{refPicOutputHeightL} / \text{PicOutputHeightL})$, SpsMaxPicWidth là chiều rộng hình ảnh tối đa được báo hiệu trong SPS, và SpsMaxPicHeight là chiều cao hình ảnh tối đa được báo hiệu trong SPS.

Sau khi đơn giản hóa phương trình, phương trình trên có thể được viết lại như sau.

$$\text{CurPicWidth} * \text{CurPicHeight} * (8 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (8 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * 225 * \text{PicOutputWidthL} * \text{PicOutputHeightL}.$$

Tương tự cho các thành phần chroma (độ chói), ràng buộc như sau.

$$\text{CurPicWidth} * \text{CurPicHeight} * (8 / \text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) * (8 / \text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (8 / \text{SubWidthC} + 3) * (8 / \text{SubHeightC} + 3) * \text{PicOutputWidthL} * \text{PicOutputHeightL}.$$

Trong phương trình trên, $(\text{SubWidthC}, \text{SubHeightC})$ là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Ký tự đề xuất cho kích thước cửa sổ chia tỷ lệ là như sau.

`scaling_win_left_offset`, `scaling_win_right_offset`, `scaling_win_top_offset`, và

scaling_win_bottom_offset chỉ sai số, trong các đơn vị của các mẫu luma, điều này được áp dụng cho kích thước hình ảnh để tính toán tỷ lệ chia. Khi **scaling_window_flag** bằng 0, giá trị của **scaling_win_left_offset**, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** được suy ra bằng 0.

Giá trị của **scaling_win_left_offset + scaling_win_right_offset** sẽ nhỏ hơn so với **pic_width_in_luma_samples**, và giá trị của **scaling_win_top_offset + scaling_win_bottom_offset** sẽ nhỏ hơn so với **pic_height_in_luma_samples**.

Các biến **PicOutputWidthL** và **PicOutputHeightL** được suy ra như sau:

$$\begin{aligned} \text{PicOutputWidthL} &= \text{pic_width_in_luma_samples} - \\ &\quad (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}). \\ \text{PicOutputHeightL} &= \text{pic_height_in_luma_samples} - \\ &\quad (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}). \end{aligned}$$

Theo ví dụ khác, **scaling_win_left_offset**, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** có thể được báo hiệu trong các đơn vị của mẫu chroma. Các biến **PicOutputWidthL** và **PicOutputHeightL** được suy ra như sau:

$$\begin{aligned} \text{PicOutputWidthL} &= \text{pic_width_in_luma_samples} - \\ &\quad \text{SubWidthC} * (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}). \\ \text{PicOutputHeightL} &= \text{pic_height_in_luma_samples} - \\ &\quad \text{SubHeightC} * (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}). \end{aligned}$$

Trong phương trình trên, **SubWidthC** và **SubHeightC** lần lượt chỉ tỷ lệ mẫu của mẫu luma với mẫu chroma theo hướng dọc và ngang.

Theo một phương án, các ràng buộc sau được đặt ra. Khi **scaling_window_flag** bằng 1, đặt **refPicOutputWidthL** và **refPicOutputHeightL** lần lượt là **PicOutputWidthL** và **PicOutputHeightL**, của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu của tuân thủ luồng bit mà các điều kiện sau được thỏa mãn:

$$\begin{aligned} & - \text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (8 * \\ & \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (8 * \text{refPicOutputHeightL} + 7 * \end{aligned}$$

PicOutputHeightL) sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} \text{ height} * 225 * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (8 / \text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) * (8 / \text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} \text{ height} * (8 / \text{SubWidthC} + 3) * (8 / \text{SubHeightC} + 3) * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

Theo phương pháp khác, chúng ta chỉ có thể lấy thành phần chẵn vào phép tính toán. Ký tự đề xuất cho kích thước cửa sổ chia tỷ lệ là như sau.

scaling_win_left_offset, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** chỉ ra sai số, trong các đơn vị của các mẫu luma, mà được áp dụng cho kích thước hình ảnh để tính toán tỷ lệ chia. Khi **scaling_window_flag** bằng 0, giá trị của **scaling_win_left_offset**, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** được suy ra bằng 0.

Giá trị của **scaling_win_left_offset** + **scaling_win_right_offset** sẽ nhỏ hơn so với **pic_width_in_luma_samples**, và giá trị của **scaling_win_top_offset** + **scaling_win_bottom_offset** sẽ nhỏ hơn so với **pic_height_in_luma_samples**.

Các biến **PicOutputWidthL** và **PicOutputHeightL** được suy ra như sau:

$$\text{PicOutputWidthL} = \text{pic_width_in_luma_samples} - (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}).$$

$$\text{PicOutputHeightL} = \text{pic_height_in_luma_samples} - (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}).$$

Theo một phương án, các ràng buộc sau được đặt ra. Khi **scaling_window_flag** bằng 1, đặt **refPicOutputWidthL** và **refPicOutputHeightL** lần lượt là **PicOutputWidthL** và **PicOutputHeightL**, của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu của tuân thủ luồng bit mà điều kiện sau được thỏa mãn:

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (8 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (8 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} * \text{height} * 225 * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

Lưu ý rằng, số 8, 7, và 3 trên có thể được thay thế bằng số khác. Ví dụ, sau khi đơn giản hóa phương trình, có thể được viết như sau.

$$\text{CurPicWidth} * \text{CurPicHeight} * (\text{M} * \text{refPicOutputWidthL} + \text{N} * \text{PicOutputWidthL}) * (\text{O} * \text{refPicOutputHeightL} + \text{P} * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (\text{M} + \text{N}) * (\text{O} + \text{P}) * \text{PicOutputWidthL} * \text{PicOutputHeightL}.$$

Tương tự đối với các thành phần chroma, ràng buộc như sau.

$$\text{CurPicWidth} * \text{CurPicHeight} * (\text{A} / \text{SubWidthC} * \text{refPicOutputWidthL} + \text{B} * \text{PicOutputWidthL}) * (\text{C} / \text{SubHeightC} * \text{refPicOutputHeightL} + \text{D} * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (\text{A} / \text{SubWidthC} + \text{B}) * (\text{C} / \text{SubHeightC} + \text{D}) * \text{PicOutputWidthL} * \text{PicOutputHeightL}.$$

Trong phương trình trên, (SubWidthC, SubHeightC) là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Theo một phương án, M và O có thể là 1, N và P có thể là 0, A và C có thể là 1 hoặc 2, B và D có thể là 0.

Theo một phương án, 16x4 và 4x16 được sử dụng làm kích thước khối để tính toán bằng thông trường hợp xấu nhất. Các ràng buộc có thể được viết như sau.

Sau khi đơn giản hóa phương trình, có thể được viết như sau:

$$\text{CurPicWidth} * \text{CurPicHeight} * (16 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (4 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * 253 * \text{PicOutputWidthL} * \text{PicOutputHeightL},$$

$$\text{CurPicWidth} * \text{CurPicHeight} * (4 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL})$$

$$* (16 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * 253 * \text{PicOutputWidthL} * \text{PicOutputHeightL}.$$

Tương tự đối với các thành phần chroma, ràng buộc như sau:

$$\begin{aligned} & \text{CurPicWidth} * \text{CurPicHeight} * (16/\text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \\ & \text{PicOutputWidthL}) * (4/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \\ & \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (16/\text{SubWidthC}+3) * (4/\text{SubHeightC}+3) * \\ & \text{PicOutputWidthL} * \text{PicOutputHeightL}, \end{aligned}$$

$$\begin{aligned} & \text{CurPicWidth} * \text{CurPicHeight} * (4/\text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \\ & \text{PicOutputWidthL}) * (16/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \\ & \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (4/\text{SubWidthC}+3) * (16/\text{SubHeightC}+3) * \\ & \text{PicOutputWidthL} * \text{PicOutputHeightL}. \end{aligned}$$

Trong các phương trình trên, (SubWidthC, SubHeightC) là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Ký tự đề xuất cho kích thước của số chia tỷ lệ là như sau.

scaling_win_left_offset, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** chỉ sai số, trong các đơn vị của các mẫu luma, mà được áp dụng cho kích thước hình ảnh để tính toán tỷ lệ chia. Khi **scaling_window_flag** bằng 0, giá trị của **scaling_win_left_offset**, **scaling_win_right_offset**, **scaling_win_top_offset**, và **scaling_win_bottom_offset** được suy ra bằng 0.

Giá trị của **scaling_win_left_offset** + **scaling_win_right_offset** sẽ nhỏ hơn so với **pic_width_in_luma_samples**, và giá trị của **scaling_win_top_offset** + **scaling_win_bottom_offset** sẽ nhỏ hơn so với **pic_height_in_luma_samples**.

Các biến **PicOutputWidthL** và **PicOutputHeightL** được suy ra như sau:

$$\begin{aligned} \text{PicOutputWidthL} &= \text{pic_width_in_luma_samples} - \\ & (\text{scaling_win_right_offset} + \text{scaling_win_left_offset}). \end{aligned}$$

$$\begin{aligned} \text{PicOutputHeightL} &= \text{pic_height_in_luma_samples} - \\ & (\text{scaling_win_bottom_offset} + \text{scaling_win_top_offset}). \end{aligned}$$

Theo một phương án, các ràng buộc sau được áp đặt. Khi `scaling_window_flag` bằng 1, đặt `refPicOutputWidthL` và `refPicOutputHeightL` lần lượt là `PicOutputWidthL` và `PicOutputHeightL`, của hình ảnh tham chiếu của hình ảnh hiện thời đề cập đến PPS này. Nó là yêu cầu của tuân thủ luồng bit mà các điều kiện sau được thỏa mãn:

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (16 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (4 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} * \text{height} * 253 * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (4 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) * (16 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} * \text{height} * 253 * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (16 / \text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) * (4 / \text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} * \text{height} * (16 / \text{SubWidthC} + 3) * (4 / \text{SubHeightC} + 3) * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

– $\text{pic_width_in_luma_samples} * \text{pic_height_in_luma_samples} * (4 / \text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) * (16 / \text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL})$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{pic_height_max_in_luma_samples} * \text{height} * (4 / \text{SubWidthC} + 3) * (16 / \text{SubHeightC} + 3) * \text{PicOutputWidthL} * \text{PicOutputHeightL}$.

Ngoài ra, theo ví dụ khác, có thể chỉ lấy thành phần chẵn vào tính toán.

Theo phương án khác, tỷ lệ dung sai có thể được dẫn ra. Ví dụ, T_y có thể là giá trị dung sai cho luma. T_c có thể là giá trị dung sai cho chroma. T_y và T_c có thể là số nguyên hoặc số thực (ví dụ, 1,0, 1,2, 1,1, 1,5, v.v.).

Ví dụ sau khi đơn giản hóa phương trình, ràng buộc có thể được viết lại như sau.

$$\begin{aligned} & \text{CurPicWidth} * \text{CurPicHeight} * (\text{M} * \text{refPicOutputWidthL} + \text{N} * \\ & \text{PicOutputWidthL}) * (\text{O} * \text{refPicOutputHeightL} + \text{P} * \text{PicOutputHeightL}) \leq \\ & \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (\text{M} + \text{N}) * (\text{O} + \text{P}) * \text{PicOutputWidthL} * \\ & \text{PicOutputHeightL} * \text{Ty}. \end{aligned}$$

Tương tự cho các thành phần chroma, ràng buộc như sau.

$$\begin{aligned} & \text{CurPicWidth} * \text{CurPicHeight} * (\text{A} / \text{SubWidthC} * \text{refPicOutputWidthL} + \text{B} * \\ & \text{PicOutputWidthL}) * (\text{C} / \text{SubHeightC} * \text{refPicOutputHeightL} + \text{D} * \text{PicOutputHeightL}) \\ & \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * (\text{A} / \text{SubWidthC} + \text{B}) * (\text{C} / \text{SubHeightC} + \text{D}) * \\ & \text{PicOutputWidthL} * \text{PicOutputHeightL} * \text{Tc}, \end{aligned}$$

Trong phương trình trên, (SubWidthC, SubHeightC) là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Trong một ví dụ, M và O có thể là 1, N và P có thể là 0, A và C có thể là 1 hoặc 2, B và D có thể là 0. Trong một ví dụ, M và O có thể là 8, N và P có thể là 7 hoặc 8, A và C có thể là 8, B và D có thể là 3 hoặc 4. Trong một ví dụ, M có thể là 16 O có thể là 4, N và P có thể là 7 hoặc 8, A có thể là 16 và C có thể là 4, B và D có thể là 3 hoặc 4. Trong một ví dụ, M có thể là 4 O có thể là 16, N và P có thể là 7 hoặc 8, A có thể là 4 và C có thể là 16, B và D có thể là 3 hoặc 4.

Theo phương án khác, các ràng buộc hướng ngang và dọc có thể được tách ra. Ví dụ, sau khi đơn giản hóa phương trình, các ràng buộc có thể được viết lại như sau.

$$\begin{aligned} & \text{CurPicWidth} * (\text{M} * \text{refPicOutputWidthL} + \text{N} * \text{PicOutputWidthL}) \leq \\ & \text{SpsMaxPicWidth} * (\text{M} + \text{N}) * \text{PicOutputWidthL} * \text{Ty}_x. \end{aligned}$$

$$\begin{aligned} & \text{CurPicHeight} * (\text{O} * \text{refPicOutputHeightL} + \text{P} * \text{PicOutputHeightL}) \leq \\ & \text{SpsMaxPicHeight} * (\text{O} + \text{P}) * \text{PicOutputHeightL} * \text{Ty}_y. \end{aligned}$$

Tương tự cho các thành phần chroma, các ràng buộc như sau.

$$\begin{aligned} & \text{CurPicWidth} * (\text{A} / \text{SubWidthC} * \text{refPicOutputWidthL} + \text{B} * \text{PicOutputWidthL}) \\ & \leq \text{SpsMaxPicWidth} * (\text{A} / \text{SubWidthC} + \text{B}) * \text{PicOutputWidthL} * \text{Tc}_x. \end{aligned}$$

$$\text{CurPicHeight} * (\text{C} / \text{SubHeightC} * \text{refPicOutputHeightL} + \text{D} * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (\text{C} / \text{SubHeightC} + \text{D}) * \text{PicOutputHeightL} * \text{Tc}_y.$$

Trong phương trình trên, (SubWidthC, SubHeightC) là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Trong một ví dụ, sau khi đơn giản hóa phương trình, các ràng buộc có thể được viết lại như sau.

$$\text{CurPicWidth} * (16 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (16+7) * \text{PicOutputWidthL} * \text{Ty}_x.$$

$$\text{CurPicHeight} * (4 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (4+7) * \text{PicOutputHeightL} * \text{Ty}_y.$$

$$\text{CurPicWidth} * (4 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (4+7) * \text{PicOutputWidthL} * \text{Ty}_x.$$

$$\text{CurPicHeight} * (16 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (16+7) * \text{PicOutputHeightL} * \text{Ty}_y.$$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * (4 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (4+7) * \text{PicOutputWidthL} * \text{Ty}_x.$$

$$\text{CurPicHeight} * (4 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (4+7) * \text{PicOutputHeightL} * \text{Ty}_y.$$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * (16 * \text{refPicOutputWidthL} + 7 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (16+7) * \text{PicOutputWidthL} * \text{Ty}_x.$$

$$\text{CurPicHeight} * (16 * \text{refPicOutputHeightL} + 7 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (16+7) * \text{PicOutputHeightL} * \text{Ty}_y.$$

Tương tự cho các thành phần chroma, các ràng buộc như sau:

$$\text{CurPicWidth} * (16 / \text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL})$$

$$\leq \text{SpsMaxPicWidth} * (16/\text{SubWidthC}+3) * \text{PicOutputWidthL} * \text{Tc_x}.$$

$$\text{CurPicHeight} * (4/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL})$$

$$\leq \text{SpsMaxPicHeight} * (4/\text{SubHeightC}+3) * \text{PicOutputHeightL} * \text{Tc_y}.$$

$$\text{CurPicWidth} * (4/\text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (4/\text{SubWidthC}+3) * \text{PicOutputWidthL} * \text{Tc_x}.$$

$$\text{CurPicHeight} * (16/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (16/\text{SubHeightC}+3) * \text{PicOutputHeightL} * \text{Tc_y}.$$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * (4/\text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (4/\text{SubWidthC}+3) * \text{PicOutputWidthL} * \text{Tc_x}.$$

$$\text{CurPicHeight} * (4/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (4/\text{SubHeightC}+3) * \text{PicOutputHeightL} * \text{Tc_y}.$$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * (16/\text{SubWidthC} * \text{refPicOutputWidthL} + 3 * \text{PicOutputWidthL}) \leq \text{SpsMaxPicWidth} * (16/\text{SubWidthC}+3) * \text{PicOutputWidthL} * \text{Tc_x}.$$

$$\text{CurPicHeight} * (16/\text{SubHeightC} * \text{refPicOutputHeightL} + 3 * \text{PicOutputHeightL}) \leq \text{SpsMaxPicHeight} * (16/\text{SubHeightC}+3) * \text{PicOutputHeightL} * \text{Tc_y}.$$

Trong phương trình trên, (SubWidthC, SubHeightC) là (2, 2), (2, 1), và (1, 1) lần lượt cho 4:2:0, 4:2:2, và 4:4:4.

Theo phương án khác, chiều dài bộ lọc (filter tap) có thể bỏ qua. Ví dụ, sau khi đơn giản hóa phương trình, ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * \text{CurPicHeight} * \text{refPicOutputWidthL} * \text{refPicOutputHeightL} \leq \text{SpsMaxPicWidth} * \text{SpsMaxPicHeight} * \text{PicOutputWidthL} * \text{PicOutputHeightL} * \text{T}.$$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$\text{CurPicWidth} * \text{refPicOutputWidthL} \leq \text{SpsMaxPicWidth} * \text{PicOutputWidthL} * \text{Tx}.$

$\text{CurPicHeight} * \text{refPicOutputHeightL} \leq \text{SpsMaxPicHeight} * \text{PicOutputHeightL} * \text{Ty}.$

Tuy nhiên, trong trường hợp của hình ảnh tham chiếu là HD đầy đủ (full-HD) (ví dụ, 1920x1080) và hình ảnh hiện thời là kích thước một phần tư của full-HD (ví dụ, 960x544), cửa sổ chia tỷ lệ của hình ảnh tham chiếu là 1920x1080 và cửa sổ chia tỷ lệ của hình ảnh hiện thời nếu 960x544, thì ràng buộc chiều cao không được thỏa mãn. Do đó, ràng buộc nên được nới lỏng một chút. Theo một phương án, chiều rộng hoặc chiều cao hình ảnh hiện thời được trừ đi một số hạng. Ví dụ, số hạng có thể là 8, 16, số nguyên dương, giá trị liên quan đến minCbSizeY hoặc minCbSizeC, giá trị liên quan đến CTU/CU ràng buộc, hoặc có thể là $\text{Max}(8, \text{MinCbSizeY})$, hoặc $\text{Max}(k, \text{MinCbSizeY})$ trong đó k là số nguyên dương. Ví dụ, các ràng buộc có thể được sửa đổi như sau:

– $(\text{pic_width_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputWidthL}$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_width_max_in_luma_samples} * \text{PicOutputWidthL}.$

– $(\text{pic_height_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputHeightL}$ sẽ nhỏ hơn so với hoặc bằng $\text{pic_height_max_in_luma_samples} * \text{PicOutputHeightL}.$

Hoặc

– $\text{PicOutputWidthL} * \text{pic_width_max_in_luma_samples}$ lớn hơn hoặc bằng $\text{refPicOutputWidthL} * (\text{pic_width_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY})).$

– $\text{PicOutputHeightL} * \text{pic_height_max_in_luma_samples}$ lớn hơn hoặc bằng $\text{refPicOutputHeightL} * (\text{pic_height_in_luma_samples} - \text{Max}(8, \text{MinCbSizeY})).$

Theo ví dụ khác, các ràng buộc có thể được viết lại như sau:

$(\text{CurPicWidth} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputWidthL} \leq \text{SpsMaxPicWidth} * \text{PicOutputWidthL}.$

$$(\text{CurPicHeight} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputHeightL} \leq \text{SpsMaxPicHeight} * \text{PicOutputHeightL}.$$

Theo ví dụ khác nữa, các ràng buộc có thể được viết lại như sau:

$$(\text{CurPicWidth} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputWidthL} \leq \text{SpsMaxPicWidth} * \text{PicOutputWidthL} * T_x.$$

$$(\text{CurPicHeight} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputHeightL} \leq \text{SpsMaxPicHeight} * \text{PicOutputHeightL} * T_y.$$

Theo ví dụ khác nữa, ràng buộc có thể được viết lại như sau:

$$(\text{CurPicWidth} - \text{Max}(8, \text{MinCbSizeY})) * (\text{CurPicHeight} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputWidthL} * \text{refPicOutputHeightL} \leq \text{SpsMaxPicWidth} * \text{PicOutputWidthL} * \text{SpsMaxPicHeight} * \text{PicOutputHeightL}.$$

Theo ví dụ khác nữa, ràng buộc có thể được viết lại như sau:

$$(\text{CurPicWidth} - \text{Max}(8, \text{MinCbSizeY})) * (\text{CurPicHeight} - \text{Max}(8, \text{MinCbSizeY})) * \text{refPicOutputWidthL} * \text{refPicOutputHeightL} \leq \text{SpsMaxPicWidth} * \text{PicOutputWidthL} * \text{SpsMaxPicHeight} * \text{PicOutputHeightL} * T.$$

Theo phương án khác, chiều rộng hoặc chiều cao hình ảnh tối đa có thể được cộng số hạng. Ví dụ, số hạng có thể là 8, 16, số nguyên dương, giá trị liên quan đến minCbSizeY/C, giá trị liên quan đến CTU/CU ràng buộc, hoặc có thể là $\text{Max}(8, \text{MinCbSizeY})$, hoặc $\text{Max}(k, \text{MinCbSizeY})$ trong đó k là số nguyên dương. Ví dụ, các ràng buộc có thể được sửa đổi như sau:

– $\text{pic_width_in_luma_samples} * \text{refPicOutputWidthL}$ sẽ nhỏ hơn so với hoặc bằng $(\text{pic_width_max_in_luma_samples} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputWidthL}$.

– $\text{pic_height_in_luma_samples} * \text{refPicOutputHeightL}$ sẽ nhỏ hơn so với hoặc bằng $(\text{pic_height_max_in_luma_samples} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputHeightL}$.

Theo ví dụ khác nữa, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * \text{refPicOutputWidthL} \leq (\text{SpsMaxPicWidth} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputWidthL}.$$

$$\text{CurPicHeight} * \text{refPicOutputHeightL} \leq (\text{SpsMaxPicHeight} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputHeightL}.$$

Theo ví dụ khác nữa, các ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * \text{refPicOutputWidthL} \leq (\text{SpsMaxPicWidth} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputWidthL} * T_x.$$

$$\text{CurPicHeight} * \text{refPicOutputHeightL} \leq (\text{SpsMaxPicHeight} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputHeightL} * T_y.$$

Theo ví dụ khác nữa, ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * \text{CurPicHeight} * \text{refPicOutputWidthL} * \text{refPicOutputHeightL} \leq (\text{SpsMaxPicWidth} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputWidthL} * (\text{SpsMaxPicHeight} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputHeightL}.$$

Theo ví dụ khác nữa, ràng buộc có thể được viết lại như sau:

$$\text{CurPicWidth} * \text{CurPicHeight} * \text{refPicOutputWidthL} * \text{refPicOutputHeightL} \leq (\text{SpsMaxPicWidth} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputWidthL} * (\text{SpsMaxPicHeight} + \text{Max}(8, \text{MinCbSizeY})) * \text{PicOutputHeightL} * T.$$

Theo phương án khác, cả chiều rộng hoặc chiều cao hình ảnh hiện thời có thể trừ số hạng và chiều rộng hoặc chiều cao hình ảnh tối đa có thể cộng số hạng như được đề cập ở trên.

Theo phương án khác, trong các phương pháp nêu trên, chiều rộng hoặc chiều cao của số chia tỷ lệ hoặc hình ảnh có thể được thay thế bằng một số CTU trong chiều rộng/chiều cao của số chia tỷ lệ/hình ảnh.

Phương pháp-2: Vô hiệu hóa khối liên ảnh nhỏ khi tỷ lệ chia lớn hơn ngưỡng.

Đối với bù chuyển động, vì ảnh hưởng của bộ lọc nội suy, khối nhỏ hơn yêu cầu bằng thông lớn hơn mỗi mẫu. Để giảm băng thông bộ nhớ, đề xuất vô hiệu hóa khối liên kết nhỏ khi tỷ lệ chia lớn hơn ngưỡng. Ví dụ, khi tỷ lệ chia lớn hơn so với K, hoặc lớn hơn hoặc bằng K, khối liên kết với chiều cao và/hoặc chiều cao nhỏ hơn so với N được vô hiệu hóa. Theo ví dụ khác, khi tỷ lệ chia lớn hơn so với K, khối liên kết với diện tích nhỏ hơn so với M

được vô hiệu hóa. K có thể là 1, 1,25, 1,5, 1,75, hoặc 2,0. N có thể là 8, 16, hoặc 32. M có thể là 32, 64, 128, 256, 512, hoặc 1024.

Theo một phương án, $\text{RefPicIsScaled}[\text{][][0}]$ và/hoặc $\text{RefPicIsScaled}[\text{][][1}]$ có thể được sử dụng. Ví dụ, khi $\text{RefPicIsScaled}[\text{][][0}]$ và/hoặc $\text{RefPicIsScaled}[\text{][][1}]$ lớn hơn so với K hoặc lớn hơn so với hoặc bằng K, khối liên kết với chiều cao và/hoặc chiều cao nhỏ hơn so với N được vô hiệu hóa. Theo ví dụ khác, khi $\text{RefPicIsScaled}[\text{][][0}]$ và/hoặc $\text{RefPicIsScaled}[\text{][][1}]$ lớn hơn so với K, khối liên kết với diện tích nhỏ hơn so với M được vô hiệu hóa. K có thể là $(1, 1,25, 1,5, 1,75, \text{ hoặc } 2,0) \cdot 2^P$, trong đó P có thể là 14. N có thể là 8, 16, hoặc 32. M có thể là 32, 64, 128, 256, 512, hoặc 1024.

Phương pháp-3 : Không có dự đoán liên ảnh nào bên ngoài cửa sổ chia tỷ lệ hiện thời

Trong phương pháp đề xuất, nếu CU hiện thời ở bên ngoài cửa sổ chia tỷ lệ hình ảnh hiện thời, bộ mã hóa không nên sử dụng chế độ liên ảnh cho CU này. Do đó, luồng bit nên có sự tuân thủ là CU ở bên ngoài cửa sổ chia tỷ lệ tất cả đều không nên là chế độ liên ảnh. Theo phương án khác, cú pháp liên quan có thể được lưu lại. Ví dụ, bất kỳ CU ở bên ngoài cửa sổ chia tỷ lệ sẽ chỉ cần gửi cú pháp liên quan đến chế độ liên ảnh sao cho tất cả chế độ liên ảnh liên quan đến cú pháp có thể được lưu lại. Phương pháp này cũng đảm bảo bằng thông trong trường hợp xấu nhất.

Trong các phương pháp được bộc lộ ở trên, các ràng buộc về chia tỷ lệ cho RPR được minh họa trong các phương trình hoặc công thức khác nhau. Các phương trình hoặc công thức này không nhằm mục đích cung cấp danh sách đầy đủ về cách thực hiện có thể dựa trên các phương án của sáng chế. Như đã biết trong lĩnh vực này, các phương trình hoặc công thức này có thể có nhiều dạng khác nhau bằng cách sắp xếp lại hoặc nhóm lại các thuật ngữ trong các phương trình hoặc công thức này. Các phương án của sáng chế đề cập đến tất cả các phương trình hoặc công thức tương đương.

Bất kỳ phương pháp đề xuất nêu trên có thể được thực hiện trong bộ mã hóa và/hoặc bộ giải mã. Ví dụ, bất kỳ phương pháp đề xuất có thể được thực hiện trong mô đun chia tỷ lệ hoặc bù chuyển động hoặc mô đun xác định thông số của bộ mã hóa và/hoặc mô đun chia tỷ lệ hoặc bù chuyển động hoặc mô đun xác định thông số của bộ giải mã. Theo các khác bất kỳ

phương pháp đề xuất có thể được thực hiện dưới dạng một hoặc nhiều mạch được lắp với môđun chia tỷ lệ hoặc bù chuyển động hoặc môđun xác định thông số của bộ mã hóa và/hoặc môđun chia tỷ lệ hoặc bù chuyển động hoặc môđun xác định thông số của bộ giải mã để cung cấp thông tin được yêu cầu bảo môđun chia tỷ lệ hoặc bù chuyển động hoặc môđun xác định thông số.

Bộ mã hóa video phải tuân theo thiết kế cú pháp nêu trên để tạo ra luồng bit hợp lệ và bộ giải mã video có khả năng giải mã luồng bit chính xác chỉ nếu quá trình phân tích cú pháp tuân theo cú pháp trên. Khi cú pháp được bỏ qua trong luồng bit, bộ mã hóa và bộ giải mã nên thiết lập giá trị cú pháp làm giá trị được suy ra để đảm bảo kết quả mã hóa và giải mã khớp nhau.

Fig. 2 minh họa sơ đồ khối điển hình của hệ thống kết hợp các thông số tái tạo mẫu hình ảnh tham chiếu (RPR) được ràng buộc theo một phương án của sáng chế. Các bước như được thể hiện trong lưu đồ này cũng như lưu đồ khác trong sáng chế này, có thể được thực hiện dưới dạng mã chương trình có thể chạy được trên một hoặc nhiều bộ xử lý (ví dụ, một hoặc nhiều CPU) ở phía bộ mã hóa và/hoặc phía bộ giải mã. Các bước như được thể hiện trong lưu đồ có thể cũng được thực hiện dựa vào phần cứng như một hoặc nhiều thiết bị điện tử hoặc bộ xử lý được sắp xếp để thực hiện các bước trong lưu đồ này. Theo phương pháp này, luồng bit tương ứng với dữ liệu được mã hóa của chuỗi video được tạo ra ở phía bộ mã hóa hoặc được nhận ở phía bộ giải mã trong bước 210, trong đó luồng bit bao gồm một hoặc nhiều cú pháp liên quan đến chế độ RPR khi chế độ RPR được kích hoạt, và thông tin chia tỷ lệ cho chế độ RPR được suy ra sử dụng thông tin thứ nhất bao gồm một hoặc nhiều cú pháp nêu trên và trong đó một hoặc nhiều cú pháp này được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$. Hình ảnh đích của chuỗi video được mã hóa, ở phía bộ mã hóa, hoặc được giải mã ở phía bộ giải mã, sử dụng thông tin chia tỷ lệ khi chế độ RPR được kích hoạt cho hình ảnh đích trong bước 220.

Lưu đồ được trình bày ở trên nhằm mục đích dùng làm ví dụ để minh họa các phương án của sáng chế. Một người có hiểu biết trung bình trong lĩnh vực này có thể thực hiện sáng chế này bằng cách sửa đổi các bước riêng lẻ, tách hoặc kết hợp các bước với việc đi từ bản

chất của sáng chế này.

Mô tả trên được trình bày để cho phép một người có hiểu biết trung bình trong lĩnh vực này có thể thực hành sáng chế này như được cung cấp trong bối cảnh của một ứng dụng cụ thể và yêu cầu của nó. Các sửa đổi khác nhau đối với các phương án được mô tả sẽ rõ ràng đối với những người có hiểu biết trung bình trong lĩnh vực này và các nguyên tắc chung được xác định ở đây có thể được áp dụng cho các phương án khác. Do đó, sáng chế này không nhằm giới hạn trong các phương án cụ thể được trình bày và mô tả, nhưng được dành cho phạm vi rộng nhất phù hợp với các nguyên tắc và các tính năng mới được bộc lộ ở đây. Trong phần mô tả chi tiết ở trên, các chi tiết cụ thể khác nhau được minh họa để cung cấp sự hiểu biết thấu đáo về sáng chế. Tuy nhiên, những người có hiểu biết trung bình trong lĩnh vực này sẽ hiểu rằng sáng chế này có thể thực hiện được.

Phương án của sáng chế như được mô tả ở trên có thể được thực hiện trong các mã phần cứng, phần mềm khác nhau hoặc kết hợp cả hai. Ví dụ, phương án của sáng chế có thể là một hoặc nhiều mạch điện được tích hợp vào chip nén video hoặc mã chương trình được tích hợp vào phần mềm nén video để thực hiện quá trình xử lý được mô tả ở đây. Một phương án của sáng chế cũng có thể là mã chương trình được chạy trên bộ xử lý tín hiệu số (Digital Signal Processor: DSP) để thực hiện quy trình xử lý như mô tả ở đây. Sáng chế cũng bao gồm một số chức năng được thực hiện bằng bộ xử lý máy tính, bộ xử lý tín hiệu số, bộ vi xử lý hoặc mảng phần tử có thể lập trình được (field programmable gate array: FPGA). Các bộ xử lý này có thể được cấu hình để thực hiện các tác vụ cụ thể theo sáng chế, bằng cách thực thi mã phần mềm hoặc mã phần sụn có thể đọc được bằng máy xác định các phương pháp cụ thể được bao hàm bởi sáng chế. Mã phần mềm hoặc mã phần sụn có thể được phát triển bằng 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, kiểu và ngôn ngữ khác nhau của mã phần mềm và các phương tiện cấu hình mã khác để thực hiện các nhiệm vụ phù hợp với sáng chế sẽ không lệch khỏi và phạm vi của sáng chế.

Sáng chế có thể được thể hiện dưới các hình thức cụ thể khác mà không lệch khỏi bản chất hoặc các đặc điểm cơ bản của nó. Các ví dụ được mô tả chỉ được coi là minh họa và

không hạn chế. Do đó, phạm vi của sáng chế được chỉ ra bởi các yêu cầu bảo hộ đi kèm chứ không phải bởi phần mô tả ở trên. Tất cả các thay đổi đi kèm với ý nghĩa và phạm vi tương đương của yêu cầu bảo hộ sẽ nằm trong phạm vi của chúng.

Sáng chế yêu cầu hưởng quyền ưu tiên của đơn sáng chế tạm thời Mỹ số 62/960,782, nộp ngày 14/01/2020 mà được tham khảo kết hợp ở đây.

YÊU CẦU BẢO HỘ

1. Phương pháp lập mã chuỗi video, trong đó chế độ tái tạo mẫu hình ảnh tham chiếu (Reference Picture Resampling: RPR) được hỗ trợ, phương pháp này bao gồm:

tạo ở phía bộ mã hóa, hoặc nhận ở phía bộ giải mã, luồng bit tương ứng với dữ liệu được mã hóa của chuỗi video, trong đó luồng bit bao gồm một hoặc nhiều cú pháp liên quan đến chế độ RPR khi chế độ RPR được kích hoạt, và thông tin chia tỷ lệ đối với chế độ RPR được suy ra sử dụng một hoặc nhiều cú pháp này, và trong đó một hoặc nhiều cú pháp trên được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$; trong đó kích thước khối lập mã tối thiểu cho thành phần chói (luma) được chỉ ra bởi một biến, MinCbSizeY , trong đó một hoặc nhiều cú pháp bao gồm cú pháp thứ nhất tương ứng với chiều rộng hình ảnh hiện thời trong các mẫu luma và cú pháp thứ hai tương ứng với chiều rộng hình ảnh tối đa trong các mẫu luma, trong đó một hoặc nhiều cú pháp được ràng buộc để làm cho $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều rộng hình ảnh tối đa} * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời})$, và

trong đó hình ảnh đích của chuỗi video được mã hóa, ở phía bộ mã hóa, hoặc được giải mã ở phía bộ giải mã, sử dụng thông tin chia tỷ lệ.

2. Phương pháp theo điểm 1, trong đó một hoặc nhiều cú pháp bao gồm cú pháp thứ ba tương ứng với chiều cao hình ảnh hiện thời trong các mẫu luma và cú pháp tương ứng thứ tư với chiều cao hình ảnh tối đa trong các mẫu luma.
3. Phương pháp theo điểm 2, trong đó một hoặc nhiều cú pháp được ràng buộc để làm cho $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều cao hình ảnh tối đa} * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời})$.
4. Phương pháp theo điểm 1, trong đó luồng bit (bitstream) tuân thủ yêu cầu tương thích luồng bit mà một hoặc nhiều cú pháp được ràng buộc bằng cách bỏ qua bộ lọc nội suy đối với bù chuyển động, và một hoặc nhiều cú pháp được ràng buộc bằng cách tính đến giá

trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$.

5. Phương pháp theo điểm 1, trong đó luồng بیت tuân thủ yêu cầu tương thích luồng بیت mà $(\text{chiều rộng hình ảnh tối đa} * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời})$ lớn hơn hoặc bằng $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

6. Phương pháp theo điểm 1, trong đó luồng بیت tuân thủ yêu cầu tương thích luồng بیت mà $(\text{chiều cao hình ảnh tối đa} * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời})$ lớn hơn hoặc bằng $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

7. Thiết bị lập mã chuỗi video, trong đó chế độ tái tạo hình ảnh tham chiếu (RPR) mode được hỗ trợ, thiết bị này bao gồm một hoặc nhiều mạch điện hoặc bộ xử lý được sắp xếp để:

tạo ở phía bộ mã hóa, hoặc nhận ở phía bộ giải mã, luồng بیت tương ứng với dữ liệu được mã hóa của chuỗi video, trong đó luồng بیت bao gồm một hoặc nhiều cú pháp liên quan đến chế độ RPR khi chế độ RPR được kích hoạt, và thông tin chia tỷ lệ cho chế độ RPR được suy ra sử dụng một hoặc nhiều cú pháp và trong đó một hoặc nhiều cú pháp này được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$;

trong đó kích thước khối lập mã tối thiểu cho thành phần chói được chỉ ra bởi một biến, MinCbSizeY , trong đó một hoặc nhiều cú pháp bao gồm cú pháp thứ nhất tương ứng với chiều rộng hình ảnh hiện thời trong các mẫu luma và cú pháp thứ hai tương ứng với chiều rộng hình ảnh tối đa trong các mẫu luma, trong đó một hoặc nhiều cú pháp được ràng buộc để làm cho $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều rộng hình ảnh tối đa} * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời})$, và

trong đó hình ảnh đích của chuỗi video được mã hóa, ở phía bộ mã hóa, hoặc được giải mã ở phía bộ giải mã, sử dụng thông tin chia tỷ lệ.

8. Thiết bị theo điểm 7, trong đó một hoặc nhiều cú pháp bao gồm cú pháp thứ ba tương

ứng với chiều cao hình ảnh hiện thời trong các mẫu luma và cú pháp tương ứng thứ tư với chiều cao hình ảnh tối đa trong các mẫu luma.

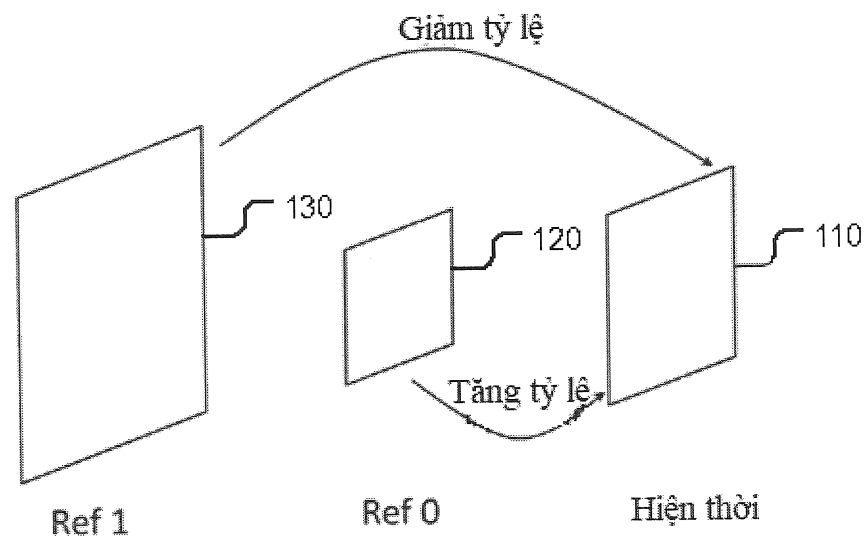
9. Thiết bị theo điểm 8, trong đó một hoặc nhiều cú pháp được ràng buộc để làm cho $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$ nhỏ hơn so với hoặc bằng $(\text{chiều cao hình ảnh tối đa} * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời})$.

10. Thiết bị theo điểm 7, trong đó luồng bit tuân thủ yêu cầu tương thích luồng bit mà một hoặc nhiều cú pháp được ràng buộc bằng cách bỏ qua bộ lọc nội suy cho bù chuyển động, và một hoặc nhiều cú pháp này được ràng buộc bằng cách tính đến giá trị tương ứng với $\text{Max}(8, (\text{kích thước khối lập mã tối thiểu cho thành phần chói}))$.

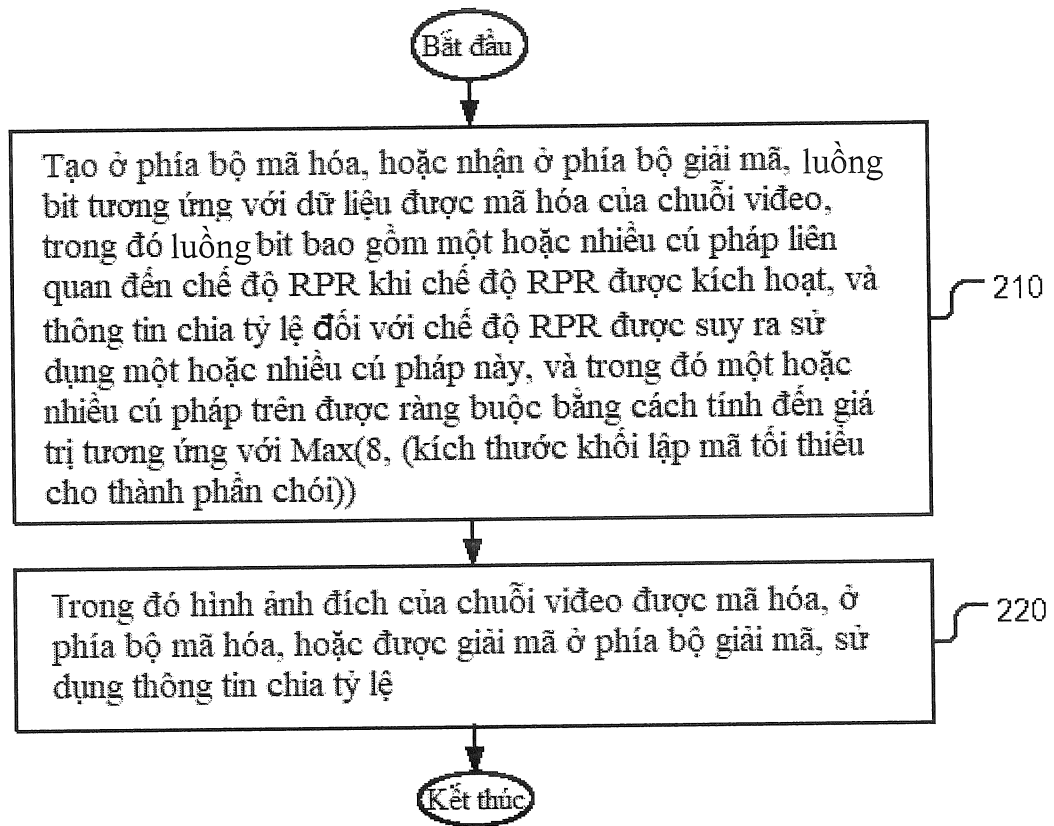
11. Thiết bị theo điểm 7, trong đó luồng bit tuân thủ yêu cầu tương thích luồng bit mà $(\text{chiều rộng hình ảnh tối đa} * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh hiện thời})$ lớn hơn hoặc bằng $((\text{chiều rộng hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều rộng cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

12. Thiết bị theo điểm 7, trong đó luồng bit tuân thủ yêu cầu tương thích luồng bit mà $(\text{chiều cao hình ảnh tối đa} * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh hiện thời})$ lớn hơn hoặc bằng $((\text{chiều cao hình ảnh hiện thời} - \text{Max}(8, \text{MinCbSizeY})) * \text{chiều cao cửa sổ chia tỷ lệ của hình ảnh tham chiếu trong các mẫu luma})$.

1/2

***Fig. 1***

2/2

**Fig. 2**