



(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)



1-0049483

(51)^{2021.01} H04N 7/00; H04N 21/435

(13) B

(21) 1-2022-03080

(22) 17/12/2020

(86) PCT/CN2020/137176 17/12/2020

(87) WO2021/121313 24/06/2021

(30) 62/948,971 17/12/2019 US; 62/954,019 27/12/2019 US; 17/121,927 15/12/2020 US

(45) 25/08/2025 449

(43) 25/08/2022 413A

(73) HFI INNOVATION INC. (CN)

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

(72) CHUANG, Tzu-Der (TW); HSU, Chih-Wei (TW); CHEN, Ching-Yeh (TW); CHEN, Lulin (US).

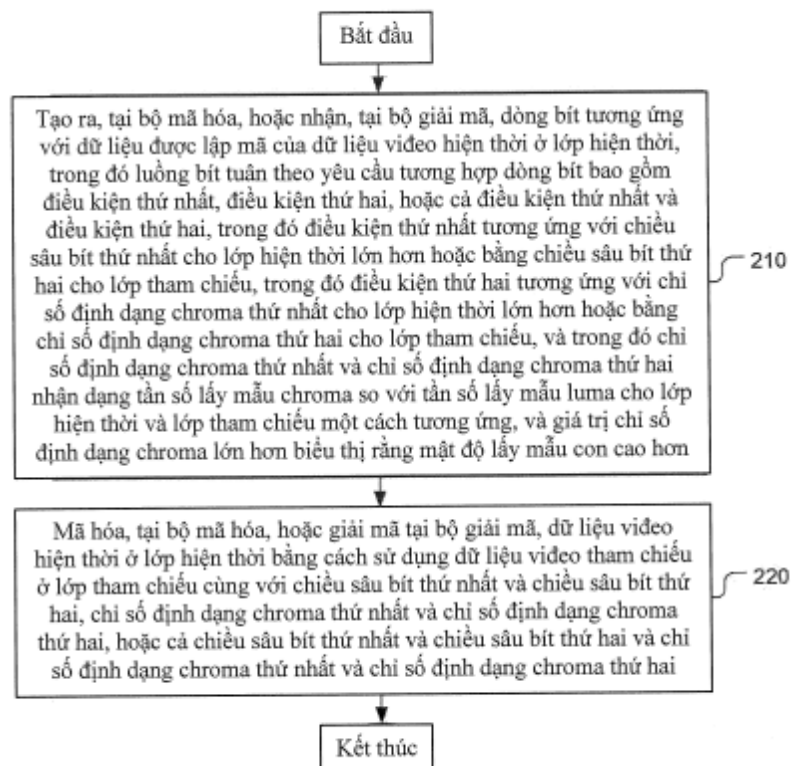
(74) Công ty TNHH Trường Xuân (AGELESS CO.,LTD.)

(54) PHƯƠNG PHÁP LẬP MÃ VIDEO

(21) 1-2022-03080

(57) Sáng chế đề cập đến phương pháp lập mã video. Theo một phương pháp, luồng bit được tạo ra tại bộ mã hóa hoặc được nhận tại bộ giải mã, trong đó luồng bit tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời. Luồng bit tuân theo yêu cầu tương hợp luồng bit tương ứng với cả các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu đều giống nhau và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu đều giống nhau. Dữ liệu video hiện thời ở lớp hiện thời sau đó được mã hóa hoặc được giải mã bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu.

Fig. 2



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lập mã video đa lớp (layer-wise). Cụ thể hơn là, sáng chế đề cập đến phương pháp lập mã video.

Tình trạng kỹ thuật của sáng chế

Theo phiên bản VVC 7 (B. Bross, và các cộng sự, “Lập mã video đa năng (Phiên bản 7)”, Nhóm chuyên gia lập mã video (JVET) của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 16: Geneva, CH, ngày 1 đến 11 tháng 10 2019, Tài liệu: JVET-P2001), lập mã layer-wise được hỗ trợ. Cấu trúc lớp này được định nghĩa trong bộ tham số video (VPS) như được thể hiện ở bảng 1.

Vps_max_layers_minus1 xác định số lượng các lớp video trong cấu trúc VPS. Các cú pháp, vps_all_independent_layers_flag, vps_independent_layer_flag[i], và vps_direct_ref_layer_flag[i][j] xác định độ phụ thuộc tham chiếu dữ liệu liên lớp.

Bảng 1. Bộ tham số video (VPS) trong VVC hỗ trợ cấu trúc lớp

video_parameter_set_rbsp() {	Biến mô tả
vps_video_parameter_set_id	u(4)
vps_max_layers_minus1	u(6)
vps_max_sublayers_minus1	u(3)
if(vps_max_layers_minus1 > 0 && vps_max_sublayers_minus1 > 0)	
vps_all_layers_same_num_sublayers_flag	u(1)
if(vps_max_layers_minus1 > 0)	
vps_all_independent_layers_flag	u(1)
for(i = 0; i <= vps_max_layers_minus1; i++) {	

vps_layer_id[i]	u(6)
if(i > 0 && !vps_all_independent_layers_flag) {	
vps_independent_layer_flag[i]	u(1)
if(!vps_independent_layer_flag[i])	
for(j = 0; j < i; j++)	
vps_direct_ref_layer_flag[i][j]	u(1)
}	
}	
if(vps_max_layers_minus1 > 0) {	
if(vps_all_independent_layers_flag)	
each_layer_is_an_ols_flag	u(1)
if(!each_layer_is_an_ols_flag) {	
if(!vps_all_independent_layers_flag)	
ols_mode_idc	u(2)
if(ols_mode_idc == 2) {	
num_output_layer_sets_minus1	u(8)
for(i = 1; i <= num_output_layer_sets_minus1; i++)	
for(j = 0; j <= vps_max_layers_minus1; j++)	
ols_output_layer_flag[i][j]	u(1)
}	

}	
}	
vps_num_ptls	u(8)
for(i = 0; i < vps_num_ptls; i++) {	
if(i > 0)	
pt_present_flag[i]	u(1)
if(vps_max_sublayers_minus1 > 0 && !vps_all_layers_same_num_sublayers_flag)	
ptl_max_temporal_id[i]	u(3)
}	
while(!byte_aligned())	
vps_ptl_byte_alignment_zero_bit /* equal to 0 */	u(1)
for(i = 0; i < vps_num_ptls; i++)	
profile_tier_level(pt_present_flag[i], ptl_max_temporal_id[i])	
for(i = 0; i < TotalNumOls; i++)	
if(NumLayersInOls[i] > 1 && vps_num_ptls > 1)	
ols_ptl_idx[i]	u(8)
if(!vps_all_independent_layers_flag)	
vps_num_dpb_params	ue(v)
if(vps_num_dpb_params > 0) {	
same_dpb_size_output_or_nonoutput_flag	u(1)

if(vps_max_sublayers_minus1 > 0)	
vps_sublayer_dpb_params_present_flag	u(1)
}	
for(i = 0; i < vps_num_dpb_params; i++) {	
dpb_size_only_flag[i]	u(1)
if(vps_max_sublayers_minus1 > 0 && !vps_all_layers_same_num_sublayers_flag)	
dpb_max_temporal_id[i]	u(3)
dpb_parameters(dpb_size_only_flag[i], dpb_max_temporal_id[i], vps_sublayer_dpb_params_present_flag)	
}	
for(i = 0; i < vps_max_layers_minus1 && vps_num_dpb_params > 1; i++) {	
if(!vps_independent_layer_flag[i])	
layer_output_dpb_params_idx[i]	ue(v)
if(LayerUsedAsRefLayerFlag[i] && !same_dpb_size_output_or_nonoutput_flag)	
layer_nonoutput_dpb_params_idx[i]	ue(v)
}	
vps_general_hrd_params_present_flag	u(1)
if(vps_general_hrd_params_present_flag) {	

general_hrd_parameters()	
if(vps_max_sublayers_minus1 > 0)	
vps_sublayer_cpb_params_present_flag	u(1)
if(TotalNumOlss > 1)	
num_ols_hrd_params_minus1	ue(v)
for(i = 0; i <= num_ols_hrd_params_minus1; i++) {	
if(vps_max_sublayers_minus1 > 0 && !vps_all_layers_same_num_sublayers_flag)	
hrd_max_tid[i]	u(3)
firstSubLayer = vps_sublayer_cpb_params_present_flag ? 0 : hrd_max_tid[i]	
ols_hrd_parameters(firstSubLayer, hrd_max_temporal_id[i])	
}	
if(num_ols_hrd_params_minus1 > 0)	
for(i = 1; i < TotalNumOlss; i++)	
ols_hrd_idx[i]	ue(v)
}	
vps_extension_flag	u(1)
if(vps_extension_flag)	
while(more_rbsp_data())	
vps_extension_data_flag	u(1)

rbsp_trailing_bits()	
}	

Ở mỗi lớp, một hoặc nhiều bộ tham số chuỗi ảnh (SPS) được báo hiệu. SPS chứa nhiều thông tin video, chẳng hạn chiều rộng/chiều cao ảnh lớn nhất, định dạng màu (chroma), kích cỡ CTU, v.v. Dự đoán liên lớp có thể được hỗ trợ trong VVC. Giữa hai lớp khác nhau, các ảnh được khôi phục ở lớp thấp hơn có thể được sử dụng làm các ảnh tham chiếu của lớp cao hơn. Nếu các kích cỡ ảnh khác nhau, phương pháp lấy lại mẫu ảnh tham chiếu được sử dụng để tạo ra các khối dự đoán. Vì vậy, hai lớp có các kích cỡ ảnh khác nhau không là vấn đề khi tham chiếu layer-wise trong VVC. Tuy nhiên, nếu các định dạng chroma của hai lớp khác nhau, thì có thể có nhiều vấn đề đối với dự đoán liên lớp. Ví dụ, nếu lớp thấp hơn là lớp được lập mã đơn sắc và lớp cao hơn có định dạng chroma 420, thì biến dự đoán thành phần chroma không thể được tạo ra bằng cách sử dụng dự đoán liên lớp.

Bản chất kỹ thuật của sáng chế

Phương pháp và thiết bị lập mã video bằng cách sử dụng cơ chế dự đoán đa lớp, được bộc lộ. Theo một phương pháp, luồng bit được tạo ra tại bộ mã hóa hoặc được nhận tại bộ giải mã, trong đó luồng bit tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời. Luồng bit tuân theo yêu cầu tương hợp luồng bit tương ứng với cả các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu đều giống nhau và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu đều giống nhau. Dữ liệu video hiện thời ở lớp hiện thời sau đó được mã hóa hoặc được giải mã bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu.

Theo một phương án, các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu được suy dẫn từ các thành phần cú pháp bit_depth_minus8 trong luồng bit, trong đó các thành phần cú pháp bit_depth_minus8 tương ứng với các giá trị chiều sâu bit trừ 8 cho lớp hiện thời và lớp tham chiếu một cách tương ứng.

Theo một phương án, các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu được xác định theo các thành phần cú pháp chroma_format_idc trong luồng bit, trong đó các thành phần cú pháp chroma_format_idc chroma_format_idc

nhận dạng việc lấy mẫu chroma so với việc lấy mẫu sáng (luma) cho lớp hiện thời và lớp tham chiếu một cách tương ứng.

Theo phương pháp khác, luồng bit được tạo ra tại bộ mã hóa hoặc được nhận tại bộ giải mã, trong đó luồng bit tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời. Luồng bit tuân theo yêu cầu tương hợp luồng bit bao gồm điều kiện thứ nhất, điều kiện thứ hai, hoặc cả điều kiện thứ nhất và điều kiện thứ hai. Điều kiện thứ nhất tương ứng với chiều sâu bit thứ nhất cho lớp hiện thời lớn hơn hoặc bằng chiều sâu bit thứ hai cho lớp tham chiếu. Điều kiện thứ hai tương ứng với chỉ số định dạng chroma thứ nhất cho lớp hiện thời lớn hơn hoặc bằng chỉ số định dạng chroma thứ hai cho lớp tham chiếu, và chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai nhận dạng việc lấy mẫu chroma so với việc lấy mẫu luma cho lớp hiện thời và lớp tham chiếu một cách tương ứng. Giá trị chỉ số định dạng chroma lớn hơn biểu thị rằng mật độ lấy mẫu cao hơn. Dữ liệu video hiện thời ở lớp hiện thời sau đó được mã hóa hoặc được giải mã bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu cùng với chiều sâu bit thứ nhất và chiều sâu bit thứ hai, chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai, hoặc cả chiều sâu bit thứ nhất và chiều sâu bit thứ hai và chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai.

Theo phương pháp khác nữa, dữ liệu đầu vào được nhận, trong đó dữ liệu đầu vào tương ứng với dữ liệu video ở lớp hiện thời tại bộ mã hóa video hoặc dữ liệu đầu vào tương ứng với dữ liệu video được lập mã ở lớp hiện thời tại bộ giải mã video. Bù chuyển động được áp dụng cho dữ liệu video ở lớp hiện thời tại bộ mã hóa hoặc được áp dụng cho dữ liệu video được lập mã ở lớp hiện thời tại bộ giải mã video bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu. Bù chuyển động sử dụng thông tin chứa các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu, các giá trị chiều sâu bit cho cả lớp hiện thời và lớp tham chiếu, hoặc cả các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu và các giá trị chiều sâu bit cho cả lớp hiện thời và lớp tham chiếu.

Theo một phương án, các định dạng chroma hoặc một hoặc nhiều biến bao gồm biến thứ nhất tương ứng với thừa số lấy mẫu con chroma ngang, biến thứ hai tương ứng với thừa số lấy mẫu con chroma đứng, hoặc cả biến thứ nhất và biến thứ hai.

Theo phương án khác, các giá trị chiều sâu بیت được suy dẫn từ các thành phần cú pháp được báo hiệu trong luồng بیت hoặc được phân tích từ luồng بیت, trong đó các thành phần cú pháp tương ứng với các giá trị chiều sâu بیت trừ 8 cho lớp hiện thời và lớp tham chiếu một cách tương ứng.

Theo phương án khác, vị trí mẫu tham chiếu của lớp tham chiếu được tính ở phép bù chuyển động bằng cách sử dụng thừa số lấy mẫu con chroma ngang, thừa số lấy mẫu con chroma đứng, hoặc cả hai thừa số vừa nêu.

Theo phương án khác, vị trí mẫu tham chiếu của lớp tham chiếu được tính ở phép bù chuyển động bằng cách sử dụng tỷ lệ giữa thừa số lấy mẫu con chroma ngang của lớp hiện thời với thừa số lấy mẫu con chroma ngang của lớp tham chiếu, tỷ lệ giữa thừa số lấy mẫu con chroma đứng của lớp hiện thời với thừa số lấy mẫu con chroma đứng của lớp tham chiếu, hoặc cả hai tỷ lệ vừa nêu.

Mô tả vắn tắt các hình vẽ

Fig. 1 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó cả các giá trị chiều sâu بیت và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu là giống nhau;

Fig. 2 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó chiều sâu بیت cho lớp hiện thời lớn hơn hoặc bằng chiều sâu بیت cho lớp tham chiếu, hoặc chỉ số định dạng chroma cho lớp hiện thời lớn hơn hoặc bằng chỉ số định dạng chroma cho lớp tham chiếu; và

Fig. 3 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó bù chuyển động sử dụng thông tin chứa các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu, hoặc các giá trị chiều sâu بیت cho cả lớp hiện thời và lớp tham chiếu.

Mô tả chi tiết sáng chế

Phần mô tả dưới đây là phần mô tả cách thức tốt nhất thực hiện sáng chế. Phần mô tả này được đưa ra nhằm mục đích minh họa các nguyên lý chung của sáng chế

chứ không nhằm giới hạn sáng chế. Phạm vi của sáng chế được xác định đúng nhất bằng cách tham chiếu các điểm yêu cầu bảo hộ đính kèm.

Sẽ dễ dàng được hiểu rằng những thành phần của sáng chế, như được mô tả và được minh họa tổng quát trên các hình vẽ ở bản mô tả này, có thể được sắp xếp và được thiết kế theo rất nhiều dạng cấu hình khác nhau. Do đó, phần mô tả chi tiết hơn về các phương án của các hệ thống và các phương pháp của sáng chế, như được thể hiện trên các hình vẽ, 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.

Tham chiếu toàn bộ bản mô tả này theo “một phương án”, “phương án”, hoặc cách diễn đạt tương tự có nghĩa là một dấu hiệu, cấu trúc, hoặc đặc tính cụ thể nào đó được mô tả theo các phương án có thể thuộc ít nhất một phương án của sáng chế. Do đó, sự xuất hiện của các cụm từ “theo một phương án” hoặc “theo phương án” ở những phần khác nhau trong toàn bộ bản mô tả này không có nghĩa là nhất thiết phải đề cập đến phương án giống nhau.

Ngoài ra, các dấu hiệu, các cấu trúc, hoặc các đặc tính đã được mô tả có thể được kết hợp theo bất kỳ cách thức phù hợp nào theo một hoặc nhiều các phương án. Tuy nhiên, người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương tự này sẽ biết rằng sáng chế có thể được thực hiện mà không có một hoặc nhiều hơn một trong số các phương án chi tiết cụ thể, hoặc được thực hiện với các phương pháp khác, các thành phần khác v.v. Mặt khác, các cấu trúc đã biết rõ, hoặc các cách thức đã biết rõ không được thể hiện hoặc được mô tả chi tiết nhằm tránh làm rối 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 gán các ký tự số giống nhau. Phần mô tả dưới đây được dự định chỉ làm ví dụ, và minh họa giản lược các phương án được chọn nhất định của thiết bị và các phương pháp của sáng chế như được yêu cầu bảo hộ ở đây.

Ở phần mô tả này, các số chỉ dẫn giống nhau xuất hiện trên các hình vẽ và phần mô tả sẽ biểu thị các thành phần tương ứng hoặc giống nhau theo nhiều cách nhìn khác nhau.

Phương pháp-1: Điều kiện ràng buộc của định dạng chroma

Theo sáng chế này, định dạng chroma của các lớp khác nhau sẽ được đặt điều kiện ràng buộc. Theo một phương án, định dạng chroma của hai hoặc nhiều hơn hai lớp mà có độ phụ thuộc (ví dụ một lớp được tham chiếu bởi lớp khác) sẽ giống nhau. Định dạng chroma của lớp cao hơn sẽ giống như định dạng chroma của các lớp tham chiếu thấp hơn. Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của lớp hiện thời sẽ giống như giá trị `chroma_format_idc` của các lớp tham chiếu. Như được biết rõ trong tiêu chuẩn VVC, chỉ số định dạng chroma nhận dạng việc lấy mẫu chroma so với việc lấy mẫu luma. `Chroma_format_idc` có giá trị bằng 0, 1, 2 và 3 tương ứng với định dạng chroma là đơn sắc, 4:2:0, 4:2:2, và 4:4:4 một cách tương ứng. Định dạng 4:2:0 (còn được gọi là 420) tương ứng với phép lấy mẫu con 2:1 ngang và đứng. Định dạng 4:2:2 (còn được gọi là 422) tương ứng với phép lấy mẫu con 2:1 ngang. Định dạng 4:4:4 (còn được gọi là 444) tương ứng với không có phép lấy mẫu con theo phương ngang hoặc đứng nào. Vì vậy, giá trị chỉ số định dạng chroma lớn hơn biểu thị rằng mật độ lấy mẫu con cao hơn. Theo ví dụ khác, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của ảnh hiện thời sẽ giống như giá trị `chroma_format_idc` của các ảnh tham chiếu. Theo phương án khác, các giá trị của các thành phần cú pháp liên quan chiều sâu bit (ví dụ `bit_depth_minus8`, `bit_depth_luma_minus8`, và/hoặc `bit_depth_chroma_minus8`) của lớp hiện thời cũng được ràng buộc điều kiện. Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị của các thành phần cú pháp liên quan chiều sâu bit (ví dụ `bit_depth_minus8`, `bit_depth_luma_minus8`, và/hoặc `bit_depth_chroma_minus8`) của lớp hiện thời sẽ giống như của các lớp tham chiếu.

Theo phương án khác, các giá trị của cờ mặt phẳng màu riêng biệt (ví dụ `separate_color_plane_flag`) của lớp hiện thời cũng được ràng buộc điều kiện. Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị của cờ mặt phẳng màu riêng biệt (ví dụ `separate_color_plane_flag`) của lớp hiện thời sẽ giống như của các lớp tham chiếu. Theo phương án khác, các giá trị của các cờ pha màu (ví dụ `sps_chroma_horizontal_collocated_flag` và `sps_chroma_vertical_collocated_flag`) của lớp hiện thời cũng được ràng buộc điều kiện. Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị của các cờ pha màu (ví dụ `sps_chroma_horizontal_collocated_flag` và `sps_chroma_vertical_collocated_flag`) của lớp hiện thời sẽ giống như của các lớp tham chiếu.

Theo phương án khác, giá trị của chỉ số định dạng chroma (ví dụ `chroma_format_idc`) của lớp cao hơn sẽ lớn hơn hoặc bằng lớp thấp hơn (ví dụ lớp tham chiếu). Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của lớp hiện thời sẽ lớn hơn hoặc bằng giá trị `chroma_format_idc` của các lớp tham chiếu. Theo ví dụ khác, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của ảnh hiện thời sẽ lớn hơn hoặc bằng giá trị `chroma_format_idc` của các ảnh tham chiếu. Vì giá trị `chroma_format_idc` của lớp/ảnh hiện thời lớn hơn giá trị `chroma_format_idc` của ảnh/lớp tham chiếu, số lượng các mẫu chroma của ảnh/lớp tham chiếu nhỏ hơn số lượng các mẫu chroma của lớp/ảnh hiện thời, nên phép bù chuyển động chroma sẽ cần thực hiện phép nội suy ở miền được lấy mẫu con. Ví dụ, nếu định dạng chroma của lớp/ảnh hiện thời là 4:4:4 và định dạng chroma của ảnh/lớp tham chiếu là 4:2:0, thì kích cỡ ảnh chroma của ảnh/lớp tham chiếu được giảm một nửa kích cỡ của ảnh hiện thời theo chiều rộng và chiều cao.

Theo một phương án, các hệ số phóng tỷ lệ của mẫu luma và mẫu chroma được suy dẫn riêng. Các độ lệch của số tỷ lệ (ví dụ `scaling_win_left_offset` và `scaling_win_top_offset`) của mẫu luma và mẫu chroma còn được suy dẫn riêng. Nếu lớp tham chiếu có thành phần màu ít (ví dụ đơn sắc), thì trị số được định trước, được suy dẫn, hoặc được báo hiệu được gán thành các biến dự đoán của các thành phần màu khuyết. Ví dụ, giá trị của $(1 \ll (\text{bit_depth} - 1))$ có thể được sử dụng làm các biến dự đoán của các thành phần màu khuyết.

Theo phương án khác, giá trị của chỉ số định dạng chroma (ví dụ `chroma_format_idc`) của lớp cao hơn sẽ nhỏ hơn hoặc bằng lớp thấp hơn (ví dụ lớp tham chiếu). Ví dụ, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của lớp hiện thời sẽ nhỏ hơn hoặc bằng giá trị `chroma_format_idc` của các lớp tham chiếu. Theo ví dụ khác, có yêu cầu về tương hợp luồng bit trong đó giá trị `chroma_format_idc` của ảnh hiện thời sẽ nhỏ hơn hoặc bằng giá trị `chroma_format_idc` của các ảnh tham chiếu. Vì giá trị `chroma_format_idc` của lớp/ảnh hiện thời nhỏ hơn giá trị `chroma_format_idc` của ảnh/lớp tham chiếu, số lượng các mẫu chroma của ảnh/lớp tham chiếu nhỏ hơn số lượng các mẫu chroma của lớp/ảnh hiện thời. Vì vậy, bù chuyển động chroma sẽ cần thực hiện phép nội suy ở miền được nâng việc lấy mẫu. Ví dụ, nếu định dạng chroma của lớp/ảnh hiện thời là 4:2:0 và định dạng chroma của ảnh/lớp tham chiếu là 4:4:4, kích cỡ ảnh chroma của ảnh/lớp tham

chiều được tăng gấp đôi kích cỡ của ảnh hiện thời theo chiều rộng và chiều cao.

Theo một phương án, các hệ số phóng tỷ lệ của mẫu luma và mẫu chroma được suy dẫn riêng. Các độ lệch của số tỷ lệ (ví dụ `scaling_win_left_offset` và `scaling_win_top_offset`) của mẫu luma và mẫu chroma còn được suy dẫn riêng. Theo phương án khác, khi lớp/ảnh tham chiếu có giá trị của `chroma_format_idc` lớn hơn so với lớp/ảnh hiện thời, thì phép lấy mẫu con mẫu chroma được áp dụng trước hoặc trong lúc hoạt động cho ảnh/lớp tham chiếu để tương hợp với định dạng chroma của lớp/ảnh hiện thời.

Phương pháp-2: Giá trị được suy dẫn của `chroma_format_idc` cho lớp cao hơn

Theo sáng chế này, một hoặc nhiều thành phần cú pháp dự đoán/tham chiếu liên lớp được báo hiệu trong SPS hoặc PPS, chẳng hạn `inter_layer_ref_pics_present_flag`. Nếu thành phần cú pháp này biểu thị rằng dự đoán/tham chiếu liên lớp được sử dụng, thì một hoặc nhiều hành phần cú pháp được bỏ qua và các giá trị của các thành phần cú pháp được suy dẫn. Ví dụ, giá trị `chroma_format_idc` được suy dẫn làm giá trị tương tự của `chroma_format_idc` của lớp tham chiếu. Theo ví dụ khác, các giá trị của các thành phần cú pháp liên quan chiều sâu bit (ví dụ `bit_depth_minus8`, `bit_depth_luma_minus8`, và/hoặc `bit_depth_chroma_minus8`) của lớp hiện thời cũng được suy dẫn làm các giá trị tương tự của các giá trị của lớp tham chiếu nếu dự đoán/tham chiếu liên lớp được sử dụng. Theo ví dụ khác, giá trị của cờ mặt phẳng màu riêng biệt (ví dụ `separate_color_plane_flag`) của lớp hiện thời cũng được suy dẫn làm các giá trị tương tự của lớp tham chiếu nếu dự đoán/tham chiếu liên lớp được sử dụng. Theo ví dụ khác, các giá trị của các cờ pha màu (ví dụ `sps_chroma_horizontal_collocated_flag` và `sps_chroma_vertical_collocated_flag`) của lớp hiện thời cũng được suy dẫn làm các giá trị tương tự của lớp tham chiếu nếu dự đoán/tham chiếu liên lớp được sử dụng.

Các thành phần cú pháp của `chroma_format_idc`, `bit_depth_minus8`,

`bit_depth_luma_minus8`, `bit_depth_chroma_minus8`,
`separate_colour_plane_flag`,

`sps_chroma_horizontal_collocated_flag`, và `sps_chroma_vertical_collocated_flag` được di chuyển đến vị trí sau các thành phần cú pháp dự đoán/tham chiếu liên lớp theo SPS hoặc PPS ở bất kỳ vị trí thích hợp.

Phương pháp-3: Xem xét định dạng chroma khi thực hiện bù chuyển động

Theo sáng chế này, các định dạng chroma của lớp/ảnh hiện thời và ảnh/lớp tham chiếu được xem xét khi thực hiện phép bù chuyển động. Khi thực hiện bù chuyển động, cần xem xét định dạng chroma hoặc biến liên quan đến định dạng chroma (ví dụ SubWidthC và RefSubWidthC) của lớp hiện thời và lớp tham chiếu. Các giá trị SubWidthC và SubHeightC được suy dẫn từ chroma_format_idc và separate_colour_plane_flag được thể hiện ở Bảng 2. Ví dụ, khi thực hiện bù chuyển động, vị trí mẫu tham chiếu của lớp tham chiếu được tính ở phép bù chuyển động bằng cách sử dụng SubWidthC, SubHeightC, hoặc cả SubWidthC và SubHeightC. Theo ví dụ khác, khi thực hiện bù chuyển động, vị trí mẫu tham chiếu của lớp tham chiếu được tính ở phép bù chuyển động bằng cách sử dụng tỷ lệ giữa SubWidthC của lớp hiện thời với SubWidthC của lớp tham chiếu, tỷ lệ giữa SubHeightC của lớp hiện thời với SubHeightC của lớp tham chiếu, hoặc cả hai tỷ lệ vừa nêu.

Bảng 2 – Các giá trị SubWidthC và SubHeightC được suy dẫn từ

chroma_format_idc và separate_colour_plane_flag

chroma_format_idc	separate_colour_plane_flag	Định dạng chroma	SubWidthC	SubHeightC
0	0	Đơn sắc	1	1
1	0	4:2:0	2	2
2	0	4:2:2	2	1
3	0	4:4:4	1	1
3	1	4:4:4	1	1

Dưới đây, chúng tôi minh họa một phần của quá trình giải mã của phép bù chuyển động chroma theo VVC. ScalingRatio [0] và scalingRatio[1] là hệ số phóng tỷ lệ ngang và đứng được suy dẫn từ cửa sổ phóng tỷ lệ luma. SubWidthC và SubHeightC là tỷ lệ lấy mẫu con chroma (liên quan đến các mẫu luma) theo phương

ngang và đứng như được thể hiện ở bảng 2. RefMvCLX là MV chroma.

```

addX = sps_chroma_horizontal_collocated_flag ? 0 : 8 * ( scalingRatio[ 0 ] - ( 1
<< 14 ) )

addY = sps_chroma_vertical_collocated_flag ? 0 : 8 * ( scalingRatio[ 1 ] - ( 1 <<
14 ) )

refxSbc = ( ( ( xSb - scaling_win_left_offset ) / SubWidthC <<
5 ) + refMvCLX[ 0 ] ) *
scalingRatio[ 0 ] + addX

refxC = ( ( Sign( refxSbc ) * ( ( Abs( refxSbc ) + 256 ) >> 9 )
+ xC * ( ( scalingRatio[ 0 ] + 8 ) >> 4 ) ) + fRefLeftOffset / SubWidthC + 16 )
>> 5

refySbc = ( ( ( ySb - scaling_win_top_offset ) / SubHeightC << 5 ) +
refMvCLX[ 1 ] ) * scalingRatio[ 1 ] + addY

refyC = ( ( Sign( refySbc ) * ( ( Abs( refySbc ) + 256 ) >> 9 )
+ yC * ( ( scalingRatio[ 1 ] + 8 )
>> 4 ) ) + fRefTopOffset / SubHeightC + 16 ) >> 5

```

Ở các công thức trên, (refxSbc, refySbc) và (refxC, refyC) là các vị trí chroma được chỉ hướng bằng vectơ chuyển động (refMvLX[0], refMvLX[1]) được xác định ở các đơn vị 1/32-mẫu. Để hỗ trợ thực hiện bù chuyển động với các định dạng chroma khác nhau, RefSubWidthC và RefSubHeightC được suy dẫn sao cho bằng SubWidthC và SubHeightC của ảnh/lớp tham chiếu, một cách tương ứng. Phép bù chuyển động được sửa đổi thành như sau:

```

addX = sps_chroma_horizontal_collocated_flag ? 0 : 8 * ( scalingRatio[ 0 ] - ( 1
<< 14 ) )

addY = sps_chroma_vertical_collocated_flag ? 0 : 8 * ( scalingRatio[ 1 ] - ( 1 <<
14 ) )

refxSbc = ( ( ( xSb - scaling_win_left_offset ) / SubWidthC <<
5 ) + refMvCLX[ 0 ] ) *

```



```

scalingRatio[ 0 ] + addX    (947)

refxC = ( ( Sign( refxSbC ) * ( ( Abs( refxSbC ) + 256 ) >>
9 ) + xC * ( ( scalingRatio[ 0 ] + 8 )
>> 4 ) ) * (SubWidthC/RefSubWidthC) + fRefLeftOffset / RefSubWidthC
+ 16 ) >> 5

refySbC = ( ( ( ySb - scaling_win_top_offset ) / SubHeightC << 5 ) +
refMvCLX[ 1 ] ) *
scalingRatio[ 1 ] + addY

refyC = ( ( Sign( refySbC ) * ( ( Abs( refySbC ) + 256 ) >>
9 ) + yC * ( ( scalingRatio[ 1 ] + 8 )
>> 4 ) ) * (SubHeightC/RefSubHeightC) +
fRefTopOffset / RefSubHeightC + 16 ) >> 5

```

Theo phương án khác, pha màu của ảnh/lớp tham chiếu cũng được xem xét. Lấy RefChromaHorCollocatedFlag và RefChromaVerCollocatedFlag làm SPS_chroma_horizontal_collocated_flag và sps_chroma_vertical_collocated_flag của ảnh/lớp tham chiếu.

```

addXcur = (sps_chroma_horizontal_collocated_flag ? 0 : 16) * scalingRatio[ 0 ] /
SubWidthC

addXref = (RefChromaHorCollocatedFlag ? 0 : 16) * ( 1 << 14 ) /
RefSubWidthC

addX = addXcur - addXref

addYcur = (sps_chroma_vertical_collocated_flag ? 0 : 16) * scalingRatio[ 1 ] /
SubHeightC

addYref = (RefChromaVerCollocatedFlag ? 0 : 16) * ( 1 << 14 ) /
RefSubHeightC

addY = addYcur - addYref

refxSbC = ( ( ( xSb - scaling_win_left_offset ) / SubWidthC <<
5 ) + refMvCLX[ 0 ] ) *

```

```

scalingRatio[ 0 ] + addX

refxC = ( ( Sign( refxSbC ) * ( ( Abs( refxSbC ) + 256 ) >>
          9 ) + xC * ( ( scalingRatio[ 0 ] + 8 )
          >> 4 ) ) * (SubWidthC/RefSubWidthC) + fRefLeftOffset /
          RefSubWidthC + 16 ) >> 5

refySbC = ( ( ( ySb - scaling_win_top_offset ) / SubHeightC << 5 ) +
refMvCLX[ 1 ] ) *
          scalingRatio[ 1 ] + addY

refyC = ( ( Sign( refySbC ) * ( ( Abs( refySbC ) + 256 ) >>
          9 ) + yC * ( ( scalingRatio[ 1 ] + 8 )
          >> 4 ) ) * (SubHeightC/RefSubHeightC) +
          fRefTopOffset / RefSubHeightC + 16 ) >> 5

```

Theo phương án khác, MV luma, refMvLX[] được sử dụng. (refxSbC, refySbC) và (refxC, refyC) được suy dẫn bằng cách sử dụng vị trí mẫu luma tương ứng và sau đó biến đổi nó thành vị trí mẫu chroma ở ảnh/lớp tham chiếu.

Theo sáng chế này, nếu lớp tham chiếu có ít các thành phần màu (ví dụ đơn sắc), thì trị số được định trước, được suy dẫn, hoặc được báo hiệu được gán thành các biến dự đoán của các thành phần màu khuyết. Ví dụ, giá trị của $(1 \ll (\text{bit_depth} - 1))$ có thể được sử dụng làm các biến dự đoán của các thành phần màu khuyết.

Theo phương án khác, nếu chiều sâu bit của hai lớp khác nhau, phép cắt xén chiều sâu bit hoặc phép mở rộng chiều sâu bit được áp dụng. Nếu lớp thấp hơn có chiều sâu bit nhỏ hơn, thì phép mở rộng chiều sâu bit được áp dụng. Các bit 0 được chèn vào sau LSB cho đến khi chiều sâu bit của hai lớp là giống nhau. Nếu lớp thấp hơn có chiều sâu bit lớn hơn, phép cắt xén chiều sâu bit được áp dụng. LSB được loại bỏ cho đến khi chiều sâu bit của hai lớp là giống nhau.

Phương pháp-4: Điều kiện ràng buộc chiều sâu bit cho cấu trúc đa lớp

Theo phương pháp này, để hỗ trợ khả năng mở rộng chiều sâu bit cho cấu trúc đa lớp, đã đề xuất điều kiện ràng buộc luồng bit trong đó chiều sâu bit của lớp cao hơn sẽ lớn hơn hoặc bằng chiều sâu bit của lớp thấp hơn. Ví dụ, có yêu cầu về tương hợp

luồng bit trong đó giá trị của `bit_depth_minus8` của lớp hiện thời sẽ lớn hơn hoặc bằng giá trị của `bit_depth_minus8` của các lớp tham chiếu của lớp hiện thời.

Theo phương pháp này, đã đề xuất xem xét chiều sâu bit của lớp hiện thời và chiều sâu bit của lớp tham chiếu khi thực hiện bù chuyển động.

Vì ảnh tham chiếu có thể có chiều sâu bit nhỏ hơn ảnh hiện thời, nên quá trình giải mã dự đoán liên ảnh cần được sửa đổi. Theo một phương án, tham số dịch chuyển trái và phải ở phép lọc nội suy cần xem xét chiều sâu bit của ảnh tham chiếu và ảnh hiện thời. Ví dụ, khi trực tiếp truy xuất điểm ảnh nguyên mà không thực hiện nội suy, thì phép dịch chuyển trái (`shift3`) cần xem xét chênh lệch chiều sâu bit giữa ảnh hiện thời và giữa ảnh tham chiếu. `Shift3` có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$. Tương tự, để tính gradient ở phép làm mịn dự đoán bằng dòng quang học PROF (Prediction Refinement with Optical Flow - PROF), các mẫu tham chiếu có thể được sử dụng trực tiếp. Nếu chiều sâu bit của ảnh hiện thời và giữa ảnh tham chiếu khác nhau, giá trị dịch chuyển sẽ được sửa đổi. Ví dụ, nó có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$. Khi thực hiện nội suy, mẫu tham chiếu đầu vào có thể được dịch chuyển trái một lượng, ví dụ `shift4`. `Shift4` có thể là chênh lệch chiều sâu bit giữa ảnh hiện thời và giữa ảnh tham chiếu. Các kích cỡ cửa sổ phóng tỷ lệ được đề xuất như dưới đây và được viết dưới dạng in nghiêng. Dưới đây, các chữ số phân mục (ví dụ, 8.5.6.3.2) là các chữ số phân mục theo VVC phiên bản 7.

8.5.6.3.2 Phép lọc nội suy mẫu luma

...

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán
`predSampleLXL`

Các biến `shift1`, `shift2` và `shift3` được suy dẫn như sau:

- *Lấy `RefBitDepth` là `BitDepth` của ảnh tham chiếu.* Biến `shift1` được thiết lập bằng $\text{Min}(4, \text{BitDepth} - 8)$, biến `shift2` được thiết lập bằng 6, biến `shift3` được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$ và biến `shift4` được thiết lập bằng $(\text{BitDepth} - \text{RefBitDepth})$.

- Biến `picW` được thiết lập bằng `pic_width_in_luma_samples` và biến `picH` được thiết lập bằng `pic_height_in_luma_samples`.

...

Giá trị mẫu luma được dự đoán `predSampleLXL` được suy dẫn như sau:

- Nếu cả `xFracL` và `yFracL` bằng 0, và cả `scalingRatio[ 0 ]` và `scalingRatio[ 1 ]` nhỏ hơn 20481, thì giá trị của `predSampleLXL` được suy dẫn như sau:

$$\text{predSampleLXL} = \text{refPicLXL}[\text{xInt}_3][\text{yInt}_3] \ll \text{shift}_3$$

- Nếu không thì, nếu `yFracL` bằng 0 và `scalingRatio[ 1 ]` nhỏ hơn 20481, thì giá trị của `predSampleLXL` được suy dẫn như sau:

$$\begin{aligned} \text{predSampleLXL} = \\ (\sum_{i=0}^7 f_{LH}[\text{xFracL}][i] (\text{refPicLXL}[\text{xInt}_i][\text{yInt}_3] \ll \text{shift}_4)) \gg \text{shift}_1 \end{aligned}$$

- Nếu không thì, nếu `xFracL` bằng 0 và `scalingRatio[ 0 ]` nhỏ hơn 20481, giá trị của `predSampleLXL` được suy dẫn như sau:

$$\begin{aligned} \text{predSampleLXL} = \\ (\sum_{i=0}^7 f_{LV}[\text{yFracL}][i] * (\text{refPicLXL}[\text{xInt}_3][\text{yInt}_i] \ll \text{shift}_4)) \gg \text{shift}_1 \end{aligned}$$

- Nếu không thì, giá trị của `predSampleLXL` được suy dẫn như sau:

- Mảng mẫu `temp[ n ]` với $n = 0..7$, được suy dẫn như sau:

$$\begin{aligned} \text{temp}[n] = \\ (\sum_{i=0}^7 f_{LH}[\text{xFracL}][i] * (\text{refPicLXL}[\text{xInt}_i][\text{yInt}_n] \ll \text{shift}_4)) \gg \text{shift}_1 \end{aligned}$$

- Giá trị mẫu luma được dự đoán `predSampleLXL` được suy dẫn như sau:

$$\text{predSampleLXL} = (\sum_{i=0}^7 f_{LV}[\text{yFracL}][i] * \text{temp}[i]) \gg \text{shift}_2$$

8.5.6.3.3 Phép lọc nội suy mẫu luma

Các đầu vào của phép nội suy vừa nêu là:

- vị trí luma ở các đơn vị mẫu toàn phần (`xIntL`, `yIntL`),
- mảng mẫu tham chiếu luma `refPicLXL`,

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán predSampleLXL .

- Lấy RefBitDepth là BitDepth của ảnh tham chiếu. Biến shift được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$.
- Biến picW được thiết lập bằng $\text{pic_width_in_luma_samples}$ và biến picH được thiết lập bằng $\text{pic_height_in_luma_samples}$.

8.5.6.3.4 Phép nội suy mẫu chroma

Đầu ra của phép nội suy vừa nêu là giá trị mẫu chroma được dự đoán predSampleLXC . Các biến shift1 , shift2 và shift3 được suy dẫn như sau:

- Lấy RefBitDepth là BitDepth của ảnh tham chiếu. Biến shift1 được thiết lập bằng $\text{Min}(4, \text{BitDepth} - 8)$, biến shift2 được thiết lập bằng 6, biến shift3 được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$ và biến shift4 được thiết lập bằng $(\text{BitDepth} - \text{RefBitDepth})$.
- Biến picWC được thiết lập bằng $\text{pic_width_in_luma_samples} / \text{SubWidthC}$ và biến picHC được thiết lập bằng $\text{pic_height_in_luma_samples} / \text{SubHeightC}$.
- ...

Giá trị mẫu chroma được dự đoán predSampleLXC được suy dẫn như sau:

- Nếu cả xFracc và yFracc bằng 0, và cả $\text{scalingRatio}[0]$ và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} = \text{refPicLXC}[\text{xInt}_1][\text{yInt}_1] \ll \text{shift3}$$

- Nếu không thì, nếu yFracc bằng 0 và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} = \left(\sum_{i=0}^3 f_{CH}[\text{xFracc}][i] * (\text{refPicLXC}[\text{xInt}_1][\text{yInt}_1] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Nếu không thì, nếu xFracC bằng 0 và scalingRatio[0] nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} = (\sum_{i=0}^3 \text{fcv}[\text{yFracC}][i] * (\text{refPicLXC}[\text{xInt}_i][\text{yInt}_i] \ll \text{shift4})) \gg \text{shift1}$$

- Nếu không thì, giá trị của predSampleLXC được suy dẫn như sau:

- Mảng mẫu temp[n] với n = 0..3, được suy dẫn như sau:

$$\text{temp}[n] = (\sum_{i=0}^3 \text{fCH}[\text{xFracC}][i] * (\text{refPicLXC}[\text{xInt}_i][\text{yInt}_n] \ll \text{shift4})) \gg \text{shift1}$$

- Giá trị mẫu chroma được dự đoán predSampleLXC được suy dẫn như sau:

$$\begin{aligned} \text{predSampleLXC} \\ = & (\text{fcv}[\text{yFracC}][0] * \text{temp}[0] + \text{fcv}[\text{yFracC}][1] * \text{temp}[1] \\ & + \text{fcv}[\text{yFracC}][2] * \text{temp}[2] + \text{fcv}[\text{yFracC}][3] * \text{temp}[3]) > \\ & > \text{shift2} \end{aligned}$$

Theo phương án khác, tham số dịch chuyển trái và phải ở phép lọc nội suy cần xem xét chiều sâu bit của ảnh tham chiếu và ảnh hiện thời. Ví dụ, khi trực tiếp truy xuất điểm ảnh nguyên mà không thực hiện nội suy, thì phép dịch chuyển trái (shift3) cần xem xét chênh lệch chiều sâu bit giữa ảnh hiện thời và ảnh tham chiếu. Shift3 có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$. Tương tự, để tính gradient ở phép làm mịn dự đoán bằng dòng quang học (PROF), các mẫu tham chiếu được sử dụng trực tiếp. Nếu chiều sâu bit của ảnh hiện thời và giữa ảnh tham chiếu khác nhau, giá trị dịch chuyển sẽ được sửa đổi. Ví dụ, nó có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$. Khi thực hiện nội suy, mẫu tham chiếu đầu vào có thể được dịch chuyển trái một lượng, chẳng hạn shift4. Shift4 có thể là chênh lệch chiều sâu bit giữa ảnh hiện thời và ảnh tham chiếu. Để thực hiện phép dịch chuyển phải ở phép lọc nội suy giai đoạn đầu tiên (ví dụ shift1), có thể xem xét chênh lệch chiều sâu bit giữa ảnh hiện thời và ảnh tham chiếu. Shift1 và shift4 có thể được bù cho nhau. Chỉ một trong số chúng có thể là số nguyên dương khác không. Nếu một trong số chúng là số nguyên dương, thì giá trị còn lại là 0. Hoặc, cả shift1 và shift4 có thể bằng 0. Shift1 có thể được sửa đổi từ $\text{Min}(4, \text{BitDepth} - 8)$ thành $\text{Max}(2 + \text{BitDepth}$

– RefBitDepth , $14 - \text{RefBitDepth}$). Shift4 có thể được sửa đổi thành $\text{Max}(0, \text{Max}(\text{BitDepth} - \text{RefBitDepth} - 4, 8 - \text{RefBitDepth}))$. Ví dụ về câu lệnh được đề xuất cho các kích cỡ cửa sổ phóng tỷ lệ dựa vào phiên bản VVC 7 là như sau và được viết dưới dạng in nghiêng.

8.5.6.3.2 Phép lọc nội suy mẫu luma

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán predSampleLXL . Các biến shift1 , shift2 và shift3 được suy dẫn như sau:

- Lấy RefBitDepth là BitDepth của ảnh tham chiếu. Biến shift1 được thiết lập bằng $\text{Max}(0, \text{Min}(\text{RefBitDepth} + 4 - \text{BitDepth}, \text{RefBitDepth} - 8))$, biến shift2 được thiết lập bằng 6, biến shift3 được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$ và biến shift4 được thiết lập bằng $\text{Max}(0, \text{Max}(\text{BitDepth} - \text{RefBitDepth} - 4, 8 - \text{RefBitDepth}))$.
- Biến picW được thiết lập bằng $\text{pic_width_in_luma_samples}$ và biến picH được thiết lập bằng $\text{pic_height_in_luma_samples}$.

Giá trị mẫu luma được dự đoán predSampleLXL được suy dẫn như sau:

- Nếu cả xFracL và yFracL bằng 0, và cả $\text{scalingRatio}[0]$ và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXL được suy dẫn như sau:

$$\text{predSampleLXL} = \text{refPicLXL}[\text{xInt}_3][\text{yInt}_3] \ll \text{shift3}$$

- Nếu không thì, nếu yFracL bằng 0 và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXL được suy dẫn như sau:

$$\text{predSampleLXL} = \left(\sum_{i=0}^7 f_{LH}[\text{xFracL}][i] * (\text{refPicLXL}[\text{xInt}_i][\text{yInt}_3] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Nếu không thì, nếu xFracL bằng 0 và $\text{scalingRatio}[0]$ nhỏ hơn 20481, giá trị của predSampleLXL được suy dẫn như sau:

$$\text{predSampleLXL} = \left(\sum_{i=0}^7 f_{LV}[\text{yFracL}][i] * (\text{refPicLXL}[\text{xInt}_3][\text{yInt}_i] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Nếu không thì, giá trị của predSampleLXL được suy dẫn như sau:

- Mảng mẫu temp[n] với $n = 0..7$, được suy dẫn như sau:

$$\text{temp}[n] = \left(\sum_{i=0}^7 f_{LH}[x\text{Frac}_L][i] * \right. \\ \left. (\text{refPicLXL}[x\text{Int}_i][y\text{Int}_n] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Giá trị mẫu luma được dự đoán predSampleLXL được suy dẫn như sau:

$$\text{predSampleLXL} = \left(\sum_{i=0}^7 f_{LV}[y\text{Frac}_L][i] * \text{temp}[i] \right) \gg \text{shift2}$$

8.5.6.3.3 Phép lọc nội suy mẫu luma

Các đầu vào của phép nội suy vừa nêu là:

- vị trí luma ở các đơn vị mẫu toàn phần ($x\text{Int}_L, y\text{Int}_L$),
- mảng mẫu tham chiếu luma refPicLXL,

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán predSampleLXL.

Lấy RefBitDepth là BitDepth của ảnh tham chiếu. Biến shift được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$.

Biến picW được thiết lập bằng pic_width_in_luma_samples và biến picH được thiết lập bằng pic_height_in_luma_samples.

8.5.6.3.4 Phép nội suy mẫu chroma

Đầu ra của phép nội suy vừa nêu là giá trị mẫu chroma được dự đoán predSampleLXC. Các biến shift1, shift2 và shift3 được suy dẫn như sau:

- *Lấy RefBitDepth là BitDepth của ảnh tham chiếu.* Biến shift1 được thiết lập bằng $\text{Max}(0, \text{Min}(\text{RefBitDepth} + 4 - \text{BitDepth}, \text{RefBitDepth} - 8))$, biến shift2 được thiết lập bằng 6, biến shift3 được thiết lập bằng $\text{Max}(2 + \text{BitDepth} - \text{RefBitDepth}, 14 - \text{RefBitDepth})$ và biến shift4 được thiết lập bằng $\text{Max}(0, \text{Max}(\text{BitDepth} - \text{RefBitDepth} - 4, 8 - \text{RefBitDepth}))$.
- Biến picWc được thiết lập bằng $\text{pic_width_in_luma_samples} / \text{SubWidthC}$ và biến picHc được thiết lập bằng $\text{pic_height_in_luma_samples} / \text{SubHeightC}$.

Giá trị mẫu chroma được dự đoán predSampleLXC được suy dẫn như sau:

- Nếu cả xFracc và yFracc bằng 0, và cả $\text{scalingRatio}[0]$ và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} = \text{refPicLXC}[\text{xInt}_1][\text{yInt}_1] \ll \text{shift3}$$

- Nếu không thì, nếu yFracc bằng 0 và $\text{scalingRatio}[1]$ nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} =$$

$$\left(\sum_{i=0}^3 \text{fCH}[\text{xFracc}][i] * (\text{refPicLXC}[\text{xInt}_i][\text{yInt}_1] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Nếu không thì, nếu xFracc bằng 0 và $\text{scalingRatio}[0]$ nhỏ hơn 20481, thì giá trị của predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC} =$$

$$\left(\sum_{i=0}^3 \text{fCV}[\text{yFracc}][i] * (\text{refPicLXC}[\text{xInt}_1][\text{yInt}_i] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Nếu không thì, giá trị của predSampleLXC được suy dẫn như sau:
- Mảng mẫu $\text{temp}[n]$ với $n = 0..3$, được suy dẫn như sau:

$$\text{temp}[n] = \left(\sum_{i=0}^3 \text{fCH}[\text{xFracc}][i] * (\text{refPicLXC}[\text{xInt}_i][\text{yInt}_n] \ll \text{shift4}) \right) \gg \text{shift1}$$

- Giá trị mẫu chroma được dự đoán predSampleLXC được suy dẫn như sau:

$$\text{predSampleLXC}$$

$$= (\text{fCV}[\text{yFracc}][0] * \text{temp}[0] + \text{fCV}[\text{yFracc}][1] * \text{temp}[1] + \text{fCV}[\text{yFracc}][2] * \text{temp}[2] + \text{fCV}[\text{yFracc}][3] * \text{temp}[3]) \gg \text{shift2}$$

Chú ý rằng, nếu chiều sâu bit của ảnh hiện thời (BitDepth) nhỏ hơn hoặc bằng 12 bit, thì shift4 luôn bằng 0. Đối với cấu hình mà hỗ trợ chiều sâu bit nhỏ hơn hoặc bằng 12 bit (ví dụ cấu hình Main 10, Main 12, Monochrome 12, Main 4:4:4/4:2:2 10/12 trong HEVC), sửa đổi liên quan đến shift4 có thể được loại bỏ.

Theo phương án khác, tham số dịch chuyển trái và phải ở phép lọc nội suy cần xem

xét chiều sâu bit của ảnh hiện thời. Ví dụ, khi trực tiếp truy xuất điểm ảnh nguyên mà không thực hiện nội suy, phép dịch chuyển trái (shift3) có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2, 14 - \text{RefBitDepth})$. Vì vậy, để tính gradient ở phép làm mịn dự đoán bằng dòng quang học (PROF), các mẫu tham chiếu được sử dụng trực tiếp. Nếu chiều sâu bit của ảnh hiện thời và ảnh tham chiếu là khác nhau, giá trị dịch chuyển sẽ được sửa đổi. Ví dụ, nó có thể được sửa đổi từ $\text{Max}(2, 14 - \text{BitDepth})$ thành $\text{Max}(2, 14 - \text{RefBitDepth})$. Khi thực hiện nội suy, để thực hiện phép dịch chuyển phải ở phép lọc nội suy giai đoạn đầu tiên, chẳng hạn shift1, có thể xem xét chiều sâu bit của ảnh tham chiếu. Shift1 có thể được sửa đổi từ $\text{Min}(4, \text{BitDepth} - 8)$ thành $\text{Min}(4, \text{RefBitDepth} - 8)$. Câu lệnh được đề xuất cho các kích cỡ cửa sổ phóng tỷ lệ là như sau và được viết dưới dạng in nghiêng.

8.5.6.3.2 Phép lọc nội suy mẫu luma

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán predSampleLXL . Các biến shift1, shift2 và shift3 được suy dẫn như sau:

- Lấy RefBitDepth là BitDepth của ảnh tham chiếu.
- Biến shift1 được thiết lập bằng $\text{Min}(4, \text{RefBitDepth} - 8)$, biến shift2 được thiết lập bằng 6 và biến shift3 được thiết lập bằng $\text{Max}(2, 14 - \text{RefBitDepth})$.
- Biến picW được thiết lập bằng $\text{pic_width_in_luma_samples}$ và biến picH được thiết lập bằng $\text{pic_height_in_luma_samples}$.

8.5.6.3.3 Phép truy xuất mẫu nguyên luma

Các đầu vào của phép nội suy vừa nêu là:

- vị trí luma ở các đơn vị mẫu toàn phần ($x_{\text{IntL}}, y_{\text{IntL}}$),
- mảng mẫu tham chiếu luma refPicLXL ,

Đầu ra của phép nội suy vừa nêu là giá trị mẫu luma được dự đoán predSampleLXL .

Lấy RefBitDepth là BitDepth của ảnh tham chiếu.

Biến shift được thiết lập bằng $\text{Max}(2, 14 - \text{RefBitDepth})$.

8.5.6.3.4 Phép nội suy mẫu chroma

Đầu ra của phép nội suy vừa nêu là giá trị mẫu chroma được dự đoán predSampleLXC . Các biến shift1 , shift2 và shift3 được suy dẫn như sau:

- Lấy RefBitDepth là BitDepth của ảnh tham chiếu.

Biến shift1 được thiết lập bằng $\text{Min}(4, \text{RefBitDepth} - 8)$, biến shift2 được thiết lập bằng 6 và biến shift3 được thiết lập bằng $\text{Max}(2, 14 - \text{RefBitDepth})$.

- Biến picWC được thiết lập bằng $\text{pic_width_in_luma_samples} / \text{SubWidthC}$ và biến picHC được thiết lập bằng $\text{pic_height_in_luma_samples} / \text{SubHeightC}$.

Ở phương pháp đã nêu trên, chiều sâu bit hoặc dịch chuyển bit ở phép dự đoán mẫu được tính trọng số sẽ sử dụng chiều sâu bit của ảnh hiện thời thay cho ảnh tham chiếu.

Theo phương án khác, tất cả các mẫu tham chiếu đầu vào sẽ thực hiện mở rộng chiều sâu bit hoặc giảm chiều sâu bit để tương hợp với chiều sâu bit của ảnh hiện thời. Khi thực hiện giảm chiều sâu bit, phép làm tròn hoặc cắt xét trực tiếp có thể được sử dụng. Khi thực hiện mở rộng chiều sâu bit, n bit của các ký tự 0 (ví dụ 2 bit của '00') có thể được chèn vào sau LSB. Theo ví dụ khác, khi thực hiện mở rộng chiều sâu bit, n bit của các ký tự 1 (ví dụ 2 bit của '11') có thể được chèn vào sau LSB. Theo ví dụ khác, khi thực hiện mở rộng chiều sâu bit, một bit của 1 và $n-1$ bit của 0 (ví dụ 2 bit của '11', 4 bit của '1000') có thể được chèn vào sau LSB. Theo ví dụ khác, khi thực hiện mở rộng chiều sâu bit, n bit của các bit được báo hiệu hoặc được định trước hoặc được suy dẫn có thể được chèn vào sau LSB.

Các phương pháp đã nêu trên có thể được kết hợp và được áp dụng toàn phần hoặc một phần.

Bất kỳ phương pháp nào trong số các phương pháp đã đề xuất ở trên có thể được thực hiện trong các bộ mã hóa và/hoặc các bộ giải mã. Ví dụ, bất kỳ phương pháp nào trong số các phương pháp đã đề xuất có thể được thực hiện ở mô đun phóng tỷ lệ hoặc bù chuyển động hoặc mô đun xác định tham số của bộ mã hóa, và/hoặc mô đun phóng tỷ lệ

hoặc bù chuyển động hoặc môđun xác định tham số của bộ giải mã. Ngoài ra, bất kỳ phương pháp nào trong số các phương pháp đã đề xuất có thể được thực hiện dưới dạng vi mạch được tích hợp vào môđun phóng tỷ lệ hoặc bù chuyển động hoặc môđun xác định tham số của bộ mã hóa và/hoặc môđun phóng tỷ lệ hoặc bù chuyển động hoặc môđun xác định tham số của bộ giải mã, để cấp thông tin cần cho môđun phóng tỷ lệ hoặc bù chuyển động hoặc môđun xác định tham số.

Các bộ mã hóa video sẽ tuân theo thiết kế cấu trúc cú pháp đã nêu ở trên để tạo ra luồng bit tương hợp, và các bộ giải mã video có thể giải mã luồng bit một cách đúng đắn chỉ khi phép phân tích được tuân theo thiết kế cấu trúc cú pháp đã nêu ở trên. Khi các thành phần cú pháp được bỏ qua trong luồng bit, các bộ mã hóa và các bộ giải mã sẽ thiết lập các giá trị của các thành phần cú pháp được bỏ qua đó thành các giá trị được suy dẫn để đảm bảo rằng không có sự không tương hợp giữa các kết quả mã hóa và giải mã.

Fig. 1 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó cả các giá trị chiều sâu bit và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu là giống nhau. Các bước được thể hiện trên lưu đồ này, cũng như các lưu đồ khác ở sáng chế này, có thể được thực hiện dưới dạng các mã chương trình có thể xử lý bằng một hoặc nhiều bộ xử lý (ví dụ, một hoặc nhiều CPU) tại bộ mã hóa và/hoặc bộ giải mã. Các bước được thể hiện trên lưu đồ có thể còn được thực hiện dựa vào phần cứng chẳng hạn một hoặc nhiều thiết bị điện tử hoặc bộ xử lý được bố trí để thực hiện các bước trên lưu đồ. Theo phương pháp này, luồng bit được tạo ra tại bộ mã hóa hoặc được nhận tại bộ giải mã ở bước 110, trong đó luồng bit tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời. Luồng bit tuân theo yêu cầu tương hợp luồng bit tương ứng với cả các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu đều giống nhau và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu đều giống nhau. Dữ liệu video hiện thời ở lớp hiện thời sau đó được mã hóa hoặc được giải mã bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu ở bước 120.

Fig. 2 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó chiều sâu bit cho lớp hiện thời lớn hơn hoặc bằng chiều sâu bit cho lớp tham chiếu, hoặc chỉ số định dạng chroma cho lớp hiện thời lớn hơn hoặc bằng chỉ số định dạng chroma cho lớp tham chiếu. Theo phương pháp này, luồng bit được tạo ra tại bộ mã hóa hoặc được nhận tại bộ

giải mã ở bước 210, trong đó luồng بیت tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời. Luồng بیت tuân theo yêu cầu tương hợp luồng بیت bao gồm điều kiện thứ nhất, điều kiện thứ hai, hoặc cả điều kiện thứ nhất và điều kiện thứ hai. Điều kiện thứ nhất tương ứng với chiều sâu بیت thứ nhất cho lớp hiện thời lớn hơn hoặc bằng chiều sâu بیت thứ hai cho lớp tham chiếu. Điều kiện thứ hai tương ứng với chỉ số định dạng chroma thứ nhất cho lớp hiện thời lớn hơn hoặc bằng chỉ số định dạng chroma thứ hai cho lớp tham chiếu, và chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai nhận dạng việc lấy mẫu chroma so với việc lấy mẫu luma cho lớp hiện thời và lớp tham chiếu một cách tương ứng. Giá trị chỉ số định dạng chroma lớn hơn biểu thị rằng mật độ lấy mẫu cao hơn. Ở bước 220, dữ liệu video hiện thời ở lớp hiện thời sau đó được mã hóa hoặc được giải mã bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu cùng với chiều sâu بیت thứ nhất và chiều sâu بیت thứ hai, chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai, hoặc cả chiều sâu بیت thứ nhất và chiều sâu بیت thứ hai và chỉ số định dạng chroma thứ nhất và chỉ số định dạng chroma thứ hai.

Fig. 3 minh họa sơ đồ khối được lấy làm ví dụ của hệ thống tích hợp lập mã video layer-wise được ràng buộc điều kiện theo phương án của sáng chế, trong đó bù chuyển động sử dụng thông tin chứa các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu, hoặc các giá trị chiều sâu بیت cho cả lớp hiện thời và lớp tham chiếu. Theo phương pháp này, dữ liệu đầu vào được nhận ở bước 310, trong đó dữ liệu đầu vào tương ứng với dữ liệu video ở lớp hiện thời tại bộ mã hóa video hoặc dữ liệu đầu vào tương ứng với dữ liệu video được lập mã ở lớp hiện thời tại bộ giải mã video. Bù chuyển động được áp dụng cho dữ liệu video ở lớp hiện thời tại bộ mã hóa hoặc được áp dụng cho dữ liệu video được lập mã ở lớp hiện thời tại bộ giải mã video bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu ở bước 320. Bù chuyển động sử dụng thông tin chứa các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu, các giá trị chiều sâu بیت cho cả lớp hiện thời và lớp tham chiếu, hoặc cả các định dạng chroma hoặc một hoặc nhiều biến liên quan đến định dạng chroma cho cả lớp hiện thời và lớp tham chiếu và các giá trị chiều sâu بیت cho cả lớp hiện thời và lớp tham chiếu.

Các lưu đồ được thể hiện là nhằm minh họa ví dụ về lập mã video theo sáng chế. Người có hiểu biết trung bình trong lĩnh vực kỹ thuật này có thể sửa đổi từng bước, sắp

xếp lại các bước, phân chia bước, hoặc kết hợp các bước để thực hiện sáng chế mà không trệch khỏi nội dung kỹ thuật của sáng chế.

Phần mô tả vừa nêu được đưa ra nhằm cho phép người có hiểu biết trung bình trong cùng lĩnh vực kỹ thuật này thực hiện sáng chế như được trình bày trong phần diễn tả ứng dụng cụ thể và yêu cầu của nó. Nhiều sửa đổi khác nhau cho các phương án đã được mô tả sẽ là điều hiển nhiên đối với những người có hiểu biết trung bình trong cùng lĩnh vực kỹ thuật này, và các nguyên lý chung được định nghĩa ở bản mô tả này có thể được áp dụng cho các phương án khác. Vì vậy, sáng chế không có ý định bị giới hạn ở các phương án cụ thể đã được thể hiện và được mô tả, mà được ý định là tuân theo phạm vi rộng nhất phù hợp với những nguyên lý và dấu hiệu mới được bộc lộ ở bản mô tả này. Ở phần mô tả chi tiết sáng chế ở trên, nhiều phần chi tiết cụ thể khác nhau được minh họa nhằm giúp hiểu toàn bộ sáng chế. Tuy nhiên, sẽ được hiểu bởi những người có hiểu biết trung bình trong cùng lĩnh vực kỹ thuật này rằng sáng chế có thể được thực hiện.

Phương án của sáng chế như được mô tả ở trên có thể được thực hiện theo nhiều lập trình phần cứng, phần mềm khác nhau hoặc kết hợp cả phần cứng và phần mềm. Ví dụ, phương án của sáng chế có thể là một hoặc nhiều vi mạch điện tử 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 đã được mô tả ở bản mô tả này. Phương án của sáng chế có thể còn là mã chương trình được xử lý trên bộ xử lý tín hiệu số DSP (Digital Signal Processor - DSP) để thực hiện quá trình xử lý đã được mô tả ở bản mô tả này. Sáng chế có thể còn liên quan đến số lượng chức năng được thực hiện bởi bộ xử lý máy tính, bộ xử lý tín hiệu số, bộ vi xử lý, hoặc vi mạch tích hợp FPGA (Field Programmable Gate Array-FPGA). Những bộ xử lý này có thể được cấu hình để thực hiện những nhiệm vụ cụ thể theo sáng chế, bằng cách xử lý mã phần mềm hoặc mã vi chương trình có thể đọc được bằng máy mà định ra các phương pháp thực hiện của sáng chế. Mã phần mềm hoặc mã vi chương trình có thể được phát triển theo các ngôn ngữ lập trình khác và các định dạng hoặc cách thức khác. Mã phần mềm có thể còn được tương thích với nhiều nền tảng khác nhau. Tuy nhiên, các định dạng, cách thức và ngôn ngữ mã khác nhau của các mã chương trình và các phương thức cấu hình mã khác để thực hiện những nhiệm vụ theo sáng chế sẽ không trệch khỏi nội dung và phạm vi của sáng chế.

Sáng chế có thể được thực hiện theo các dạng cụ thể khác mà không trệch khỏi nội dung và các đặc tính cơ bản của nó. Các ví dụ đã mô tả được xem xét theo toàn bộ các

khía cạnh chỉ là nhằm minh họa chứ không nhằm giới hạn. Vì vậy, phạm vi của sáng chế được thể hiện theo các điểm yêu cầu bảo hộ chứ không phải phần mô tả đã nêu ở trên. Mọi thay đổi mà nằm trong nội dung ý nghĩa và phạm vi nghĩa tương đương của các điểm yêu cầu bảo hộ đều được xem là thuộc phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Phương pháp lập mã video, trong đó cơ chế dự đoán đa lớp được hỗ trợ, phương pháp này bao gồm các bước:

tạo ra, tại bộ mã hóa, hoặc nhận, tại bộ giải mã, luồng bit tương ứng với dữ liệu được lập mã của dữ liệu video hiện thời ở lớp hiện thời, trong đó luồng bit tuân theo yêu cầu tương hợp luồng bit tương ứng với cả các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu đều giống nhau, và các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu đều giống nhau; và

mã hóa, tại bộ mã hóa, hoặc giải mã tại bộ giải mã, dữ liệu video hiện thời ở lớp hiện thời bằng cách sử dụng dữ liệu video tham chiếu ở lớp tham chiếu.

2. Phương pháp theo điểm 1, trong đó các giá trị chiều sâu bit cho lớp hiện thời và lớp tham chiếu được suy dẫn từ các thành phần cú pháp `bit_depth_minus8` trong luồng bit, và trong đó các thành phần cú pháp `bit_depth_minus8` tương ứng với các giá trị chiều sâu bit trừ 8 cho lớp hiện thời và lớp tham chiếu một cách tương ứng.

3. Phương pháp theo điểm 1, trong đó các giá trị chỉ số định dạng chroma cho lớp hiện thời và lớp tham chiếu được xác định theo các thành phần cú pháp `chroma_format_idc` trong luồng bit, và trong đó các thành phần cú pháp `chroma_format_idc` nhận dạng việc lấy mẫu chroma so với việc lấy mẫu luma cho lớp hiện thời và lớp tham chiếu một cách tương ứng.

Fig. 1

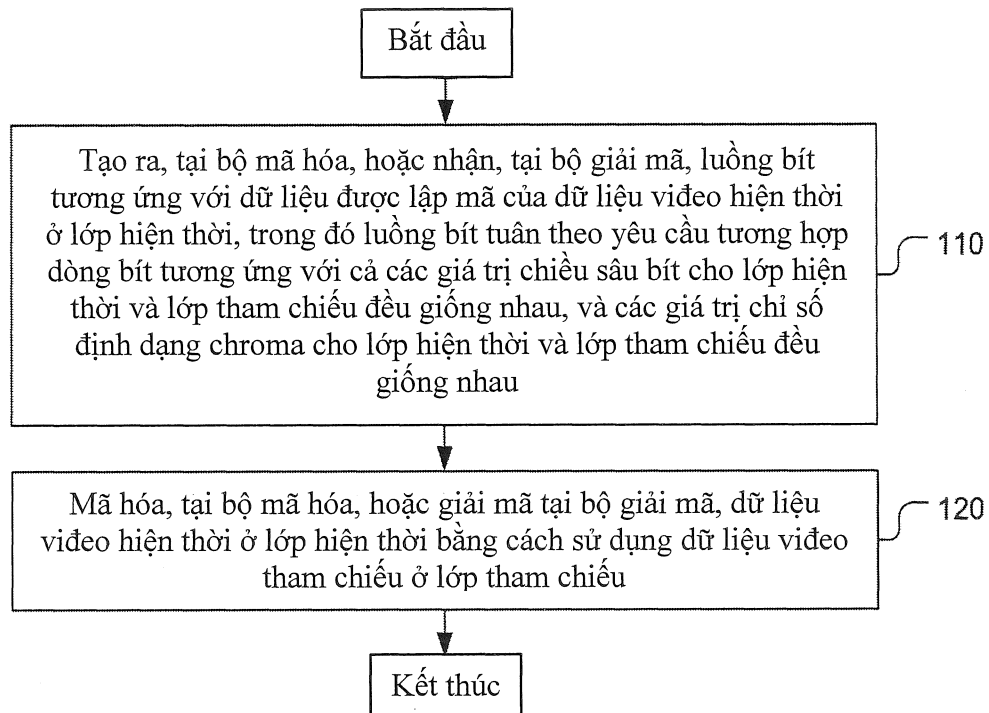


Fig. 2

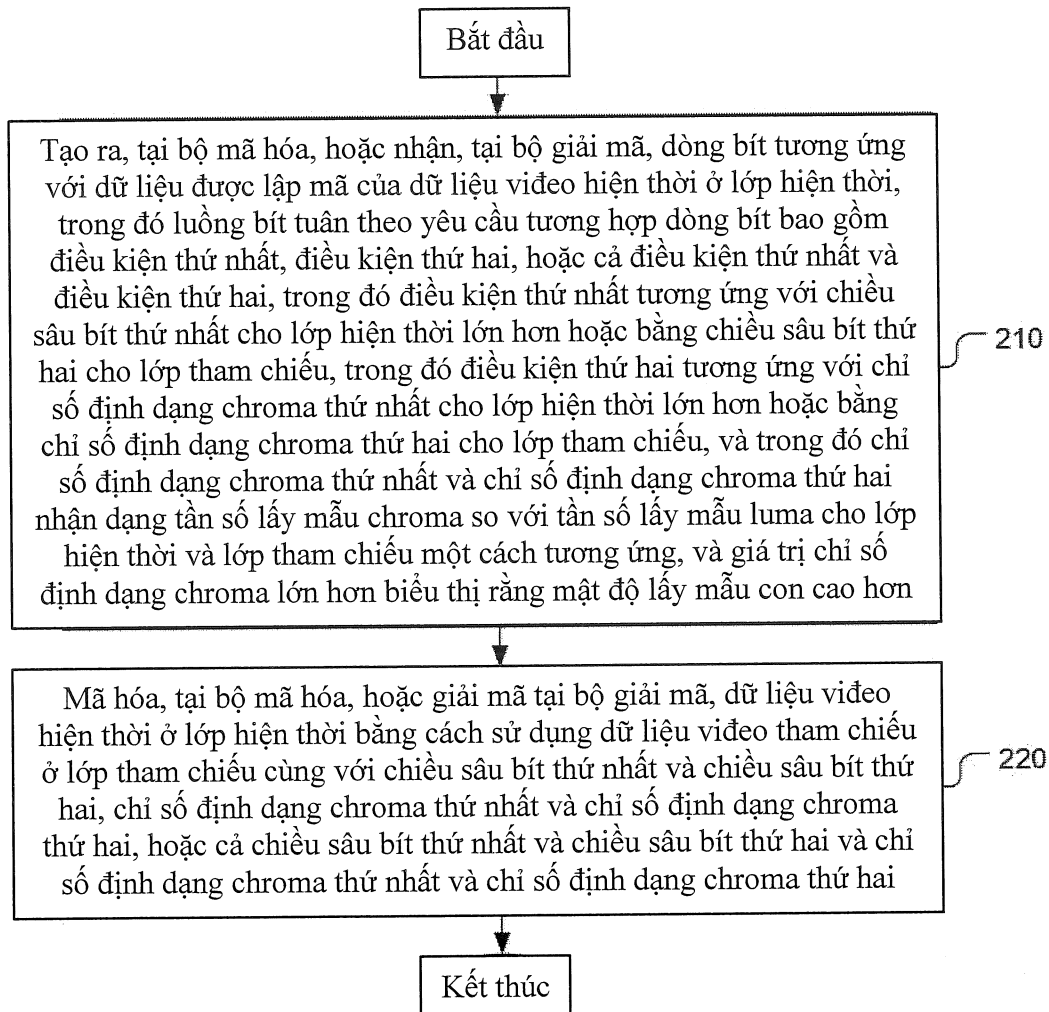


Fig. 3