



(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)



1-0053923

(51)^{2020.01} H04N 19/117

(13) B

(21) 1-2022-00957

(22) 22/07/2020

(86) PCT/CN2020/103523 22/07/2020

(87) WO2021/013178 28/01/2021

(30) 62/878,379 25/07/2019 US

(45) 25/11/2025 452

(43) 25/05/2022 410A

(73) HFI INNOVATION INC. (TW)

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

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

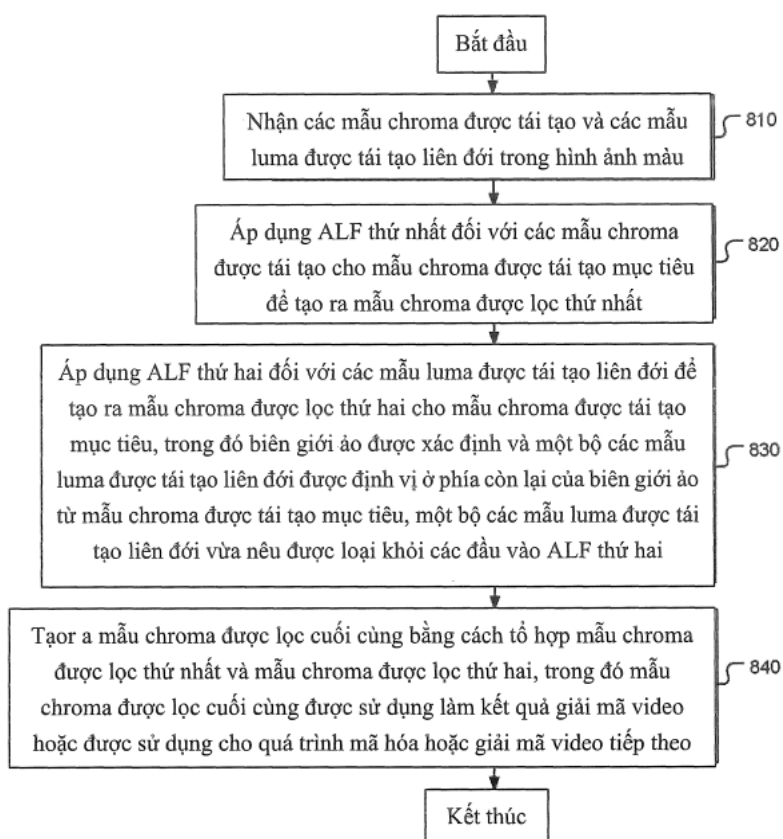
(74) Công ty TNHH Trường Xuân (AGELESS CO.,LTD.)

(54) PHƯƠNG PHÁP LẬP MÃ VIDEO

(21)1-2022-00957

(57) Sáng chế đề cập đến các phương pháp lập mã video. Theo phương pháp này, nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn. Theo phương pháp khác, hình dạng CC-ALF (phép lọc vòng lặp thích ứng thành phần chéo) được thay đổi ở khắp biên giới ảo. Theo phương pháp khác nữa, CC-ALF và ALF luma xử lý các mẫu không hiện có theo cùng cách thức.

Fig. 8



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phép lọc vòng lặp thích ứng ALF (Adaptive Loop Filter - ALF) trong hệ thống lập mã video. Cụ thể hơn là, sáng chế đề cập đến phương pháp lập mã video.

Tình trạng kỹ thuật của sáng chế

Ước lượng chuyển động là kỹ thuật lập mã liên khung hình hiệu quả nhằm khai thác sự sai lệch thời gian (sự dư thời gian) trong các chuỗi video. Lập mã liên khung hình được bù chuyển động đã được sử dụng rộng rãi trong nhiều tiêu chuẩn lập mã video quốc tế khác nhau. Ước lượng chuyển động được sử dụng trong nhiều tiêu chuẩn lập mã khác nhau thường là kỹ thuật dựa vào khối block, trong đó thông tin chuyển động chẳng hạn cơ chế lập mã và vectơ chuyển động được xác định cho từng macroblock hoặc khối block tương tự. Ngoài ra, lập mã nội ảnh cũng được ứng dụng một cách thích ứng, trong đó hình ảnh được xử lý mà không tham chiếu đến bất kỳ hình ảnh nào khác. Các phần dư được dự đoán nội ảnh hoặc được dự đoán liên ảnh thường được xử lý tiếp bằng phép biến đổi, phép lượng tử hóa, và lập mã entropy để tạo ra dòng bit video được nén. Trong quá trình mã hóa, các giả tạo lập mã bị sinh ra, nhất là trong quá trình lượng tử hóa. Để làm giảm các giả tạo lập mã, quá trình xử lý phụ được áp dụng cho video tái tạo nhằm nâng cao chất lượng hình ảnh trong các hệ thống lập mã mới hơn. Quá trình xử lý phụ vừa nêu thường được tích hợp trong phép toán vòng lặp để cho bộ mã hóa và bộ giải mã có thể suy diễn ra các hình ảnh tham chiếu giống nhau nhằm giúp cho hiệu suất hệ thống được cải thiện.

Fig. 1A minh họa hệ thống lập mã video nội ảnh/liên ảnh thích ứng được lấy làm ví dụ mà tích hợp phép xử lý vòng lặp kín có phép lọc vòng lặp thích ứng (ALF). Đối với dự đoán liên ảnh, ước lượng chuyển động (ME)/bù chuyển động (MC) 112 được sử dụng để tạo ra dữ liệu dự đoán dựa vào dữ liệu video từ một hình ảnh hoặc nhiều hình ảnh khác. Chuyển mạch 114 chọn dữ liệu dự đoán nội ảnh 110 hoặc dự đoán liên ảnh và dữ liệu dự đoán được chọn đó được cấp đến bộ cộng 116 để tạo ra các sai số dự đoán, hay còn gọi là các phần dư. Sai số dự đoán sau đó được xử lý bằng phép biến đổi T (Transformation - T) 118 và tiếp theo được xử lý bằng phép lượng tử hóa Q (Quantization - Q) 120. Các phần dư được biến đổi và được lượng tử hóa sau đó được

lập mã bằng bộ mã hóa Entropy 122 để tạo ra dòng bit video tương ứng với dữ liệu video được nén. Dòng bit liên quan đến các hệ số biến đổi sau đó được chèn vào các thông tin phụ chẳng hạn chuyển động, cơ chế, và các thông tin khác liên quan đến vùng hình ảnh. Thông tin phụ có thể cũng được lập mã entropy để giảm băng thông mong muốn. Theo đó, dữ liệu liên quan đến thông tin phụ được cấp đến bộ mã hóa Entropy 122 như được thể hiện trên Fig. 1A. Khi cơ chế dự đoán liên ảnh được sử dụng, một hình ảnh hoặc nhiều hình ảnh tham chiếu cũng phải được tái tạo tại bộ mã hóa. Do vậy, các phần dư được biến đổi và được lượng tử hóa đó được xử lý bằng phép lượng tử hóa ngược IQ (Inverse Quantization - IQ) 124 và phép biến đổi ngược IT (Inverse Transformation - IT) 126 để khôi phục các phần dư. Các phần dư sau đó được cộng trở lại vào dữ liệu dự đoán 136 tại bộ tái tạo REC (Reconstruction - REC) 128 để tái tạo dữ liệu video. Dữ liệu video tái tạo có thể được lưu ở bộ đệm hình ảnh tham chiếu 134 và được sử dụng để dự đoán các khung hình khác.

Như được thể hiện trên Fig. 1A, dữ liệu video đầu vào trải qua một loạt quá trình xử lý trong hệ thống mã hóa. Dữ liệu video tái tạo từ REC 128 có thể bị nhiều suy yếu do loạt quá trình xử lý vừa nêu. Theo đó, nhiều phép xử lý vòng lặp kín được áp dụng cho dữ liệu video tái tạo trước khi dữ liệu video tái tạo đó được lưu ở bộ đệm hình ảnh tham chiếu 134 để nâng cao chất lượng video. Ở các hệ thống lập mã video, nhiều phép lọc vòng lặp kín khác nhau chẳng hạn phép lọc tách khối DF (Deblocking Filter - DF) 130, phép bù thích ứng mẫu SAO (Sample Adaptive Offset - SAO) 131 và phép lọc vòng lặp thích ứng (ALF) 132 được sử dụng nhằm nâng cao chất lượng hình ảnh.

Bộ giải mã tương ứng dùng cho bộ mã hóa của Fig. 1A được thể hiện trên Fig. 1B. Dòng bit video được giải mã bằng bộ giải mã Entropy 142 để khôi phục các phần dư được biến đổi và được lượng tử hóa. Tại bộ giải mã, chỉ phép bù chuyển động (MC) 113 được thực hiện thay cho ME/MC. Quá trình giải mã vừa nêu là tương tự với vòng lặp tái tạo tại bộ mã hóa. Các phần dư được biến đổi và được lượng tử hóa đã được tái tạo ở trên, thông tin SAO/ALF và thông tin hệ thống khác được sử dụng để tái tạo dữ liệu video. Video tái tạo được xử lý tiếp bằng DF 130, SAO 131 và ALF 132 để tạo ra video được giải mã được cải thiện cuối cùng.

Xử lý ALF thành phần chéo

Theo JVET-O0636 (Kiran Misra, và các cộng sự, “Phép lọc vòng lặp thích ứng

thành phần chéo cho chroma”, Nhóm chuyên gia lập mã video chung JVET (Joint Video Experts Team - JVET) của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 15: Gothenburg, SE, từ ngày 3 đến 12 tháng 7 năm 2019, Tài liệu: JVET-O0636), Phép lọc vòng lặp thích ứng thành phần chéo CC-ALF (Cross-Component Adaptive Loop Filter - CC-ALF). CC-ALF sử dụng các trị số mẫu luma để lọc từng thành phần chroma. Fig. 2A minh họa cấu trúc của CC-ALF tương ứng với các phép lọc vòng lặp khác theo JVET-O0636. Trên Fig. 2A, xử lý ALF được thực hiện được thực hiện sau phép SAO (210, 212 và 214) tương ứng. Ở hệ thống lập mã video thông thường, chỉ ALF Luma 220 và ALF Chroma 230 được sử dụng. Theo JVET-O0636, hai phép lọc ALF phụ, tức là, CC ALF Cb 222 và CC ALF Cr 224 được sử dụng được sử dụng để tạo ra các tín hiệu điều chỉnh để chèn vào Cb và Cr được xử lý ALF bằng cách sử dụng các bộ cộng 240 và 242 một cách tương ứng.

CC-ALF thực thi bằng cách áp dụng phép lọc dạng hình thoi, đường thẳng đối với kênh luma cho từng thành phần chroma như được thể hiện trên Fig. 2B theo JVET-O0636. Các hệ số lọc được truyền theo APS, được phóng tỷ lệ với thừa số 2^{10} , và được làm tròn cho biểu diễn điểm cố định. Ứng dụng của các bộ lọc được điều khiển theo kích cỡ khối biến thiên và được báo hiệu bằng cờ được lập mã nội dung nhận được cho từng khối của các mẫu. Kích cỡ khối cùng với cờ kích hoạt CC-ALF được nhận tại phân mức slice cho từng thành phần chroma. Cú pháp và các ngữ nghĩa cho CC-ALF cũng được bộc lộ trong JVET-O0636.

Các biên giới ảo giúp giảm việc sử dụng bộ đệm dòng

Để loại bỏ các yêu cầu về bộ đệm dòng của SAO và ALF, chúng tôi đưa ra khái niệm biên giới ảo VB (Virtual Boundary - VB). Như được thể hiện trên Fig. 3, các VB được dịch chuyển hướng lên trên so với các biên giới đơn vị lập mã lớn nhất LCU nằm ngang (Largest Coding Unit - LCU) một khoảng N điểm ảnh. Đối với từng LCU, SAO và ALF có thể xử lý các điểm ảnh phía trên VB trước khi LCU bên dưới đến nhưng không thể xử lý các điểm ảnh phía dưới VB cho đến khi LCU bên dưới đến, điều mà bị gây ra bởi DF. Khi xét đến DF trong HM-4.0, không gian giữa VB và biên giới LCU nằm ngang được thiết lập làm ba điểm ảnh cho luma (tức là N=3 trên Fig. 3) và một điểm ảnh cho chroma (tức là N=1 trên Fig. 3). SAO và ALF được sửa đổi để đảm bảo sao cho mỗi điểm ảnh sẽ được xử lý ở một phía của VB không yêu cầu bất kỳ dữ liệu nào truy cập từ phía còn lại của VB trừ khi dữ liệu đó có thể luôn hiện hữu theo thời gian mà

không sử dụng bất kỳ bộ đệm dòng phụ nào.

Theo hội nghị lần thứ 11 của JVET, ALF có phép thích ứng dựa vào khối 4x4 được sử dụng trong dự thảo tiêu chuẩn lập mã video thế hệ tiếp theo, lập mã video đa năng VVC (Versatile Video Coding - VVC). Đối với một khối 4x4, một cửa sổ 8x8 được yêu cầu để tính toán đặc tính khối và dấu hiệu lọc là một phép lọc hình thoi 7x7 cho luma và một phép lọc hình thoi 5x5 cho chroma. Vì vậy, bộ đệm dòng yêu cầu mà không có các biên giới ảo là 7 bộ đệm dòng luma và 3 bộ đệm dòng chroma ở trên đỉnh của 4 bộ đệm dòng DF luma và 2 bộ đệm dòng DF chroma. Ý tưởng của các biên giới ảo có thể còn được áp dụng trong VVC để giảm việc sử dụng bộ đệm dòng. Ở phần mô tả nêu trên, đơn vị lập mã lớn nhất (LCU) được sửa tên thành đơn vị cây lập mã CTU (Coding Tree Unit - CTU) trong VVC. Tuy nhiên, một số sửa đổi được đề xuất để phù hợp với những đặc tính của thiết kế ALF hiện tại.

Các thuật toán ALF phi tuyến tính

Theo mô hình kiểm nghiệm lập mã video đa năng phiên bản 3.0 VTM3.0 (Versatile Video Coding Test Model Ver. 3.0 – VTM3.0), quá trình lọc của phép lọc vòng lặp thích ứng, được thực hiện như sau:

$$O(x, y) = \sum_{(i,j)} w(i, j) \cdot I(x + i, y + j), (1)$$

Ở công thức nêu trên, các mẫu $I(x + i, y + j)$ là các mẫu đầu vào, $O(x, y)$ là mẫu ra được lọc (tức là kết quả lọc), và $w(i, j)$ biểu thị các hệ số lọc. Vì ALF được áp dụng cho các mẫu được tái tạo, các mẫu $I(x + i, y + j)$ tương ứng với các mẫu được tái tạo. Điểm ảnh được tái tạo chính giữa tại vị trí chính giữa của xử lý ALF tương ứng với $(i, j) = (0, 0)$, tức là, $I(i, j)$. Trong thực tế, theo VTM3.0, nó được thực hiện bằng cách sử dụng thuật toán số nguyên cho các tính toán chính xác điểm cố định:

$$O(x, y) = \left(\sum_{i=-\frac{L}{2}}^{\frac{L}{2}} \sum_{j=-\frac{L}{2}}^{\frac{L}{2}} w(i, j) \cdot I(x + i, y + j) + 64 \right) \gg 7, (2)$$

Ở công thức nêu trên, L biểu thị độ dài lọc, và trong đó $w(i, j)$ là các hệ số lọc theo độ chính xác điểm cố định.

Công thức (1) có thể có thể được viết lại, mà không ảnh hưởng đến hiệu suất lập mã, theo biểu thức dưới đây:

$$O(x, y) = I(x, y) + \sum_{(i,j) \neq (0,0)} w(i, j) \cdot (I(x + i, y + j) - I(x, y)), (3)$$

Ở công thức nêu trên, $w(i, j)$ là các hệ số lọc giống như công thức (1) ngoại trừ $w(0, 0)$, bằng 1 trong công thức (3) trong khi nó bằng $(1 - \sum_{(i,j) \neq (0,0)} w(i, j))$ trong công thức (1).

Bằng cách sử dụng hệ số lọc của công thức (3), chúng ta có thể dễ dàng đưa sự phi tuyến tính vào để giúp cho ALF hiệu quả hơn bằng cách sử dụng hàm cắt đơn nhằm giảm tác động của các trị số mẫu liên kề ($I(x + i, y + j)$) khi chúng quá sai lệch với trị số mẫu hiện thời ($I(x, y)$) được lọc.

Theo JVET-M0385 ((J. Taquet, và các cộng sự, “*Phép lọc vòng lặp thích ứng phi tuyến tính*”, Nhóm chuyên gia lập mã video chung JVET của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 13: Marrakech, MA, từ 9 đến 18 tháng 1 năm 2019, Tài liệu: JVET- M0385)), ALF phi tuyến tính được bộc lộ. Tài liệu này đưa ra thuật toán cắt thích ứng đối với các trị số mẫu đầu vào của phép lọc vòng lặp thích ứng trong phần mềm kiểm nghiệm VTM3.0. Mục đích của việc cắt xén thích ứng này là nhằm đưa các đặc tính phi tuyến để hạn chế sự sai lệch giữa trị số mẫu đầu vào sẽ được lọc và các trị số mẫu đầu vào liên kề khác của phép lọc.

Theo JVET-M0385, thuật toán lọc ALF được sửa đổi như sau:

$$O'(x, y) = I(x, y) + \sum_{(i,j) \neq (0,0)} w(i, j) \cdot K(I(x + i, y + j) - I(x, y), k(i, j)), (4)$$

Ở công thức nêu trên, $O'(x, y)$ tương ứng với kết quả phép lọc ALF sửa đổi, $I(x, y)$ tương ứng với mẫu trước khi ALF, $K(d, b) = \min(b, \max(-b, d))$ là hàm cắt, và $k(i, j)$ là các tham số cắt, mà phụ thuộc vào hệ số lọc tại (i, j) . Bộ mã hóa thực hiện phép tối ưu hóa để tìm ra $k(i, j)$ tốt nhất.

Khi triển khai theo JVET-M0385, các tham số cắt $k(i, j)$ được định rõ cho từng phép lọc ALF, trong đó một trị số cắt được báo hiệu cho từng hệ số lọc. Điều đó có nghĩa là 12 trị số cắt được báo hiệu trong dòng bit cho từng phép lọc Luma và 6 trị số cắt cho phép lọc Chroma.

Để hạn chế tổn thất báo hiệu và độ phức tạp mã hóa, các trị số cắt được giới hạn thành bộ nhỏ gồm các trị số khả dĩ theo JVET-M0385. Ngoài ra, chỉ 4 trị số khả dĩ được sử dụng cho các slice liên ảnh và 3 trị số khả dĩ được sử dụng cho các slice nội ảnh.

Vì phương sai của các sai lệch cục bộ thường cao hơn đối với Luma so với Chroma, nên hai bộ khác nhau được sử dụng cho các phép lọc Luma và Chroma. Ngoài

ra, trị số mẫu lớn nhất được bao gồm trong từng bộ, để cho phép cắt xén có thể được hủy bỏ nếu nó không cần thiết.

Bộ các trị số cắt được đề xuất theo JVET-M0385 được nêu trong Bảng 1.

Bảng 1: Các trị số cắt cho phép

	INTRA (Nội ảnh)	INTER (Liên ảnh)
LUMA	{ 10, 102, 1024 }	{ 6, 32, 181, 1024 }
CHROMA	{ 4, 24, 1024 }	{ 4, 25, 161, 1024 }

Các trị số cắt được mã hóa ở tiêu đề slice bằng cách sử dụng mã hóa Golomb tương ứng với hệ số của trị số cắt trong bộ vừa nêu.

Giảm yêu cầu bộ đệm cho ALF bằng cách sử dụng biên giới ảo

Theo JVET-M0301 ((A. Kotra, và các cộng sự, “*Non-CE: Giảm bộ đệm dòng phép lọc vòng lặp*”, Nhóm chuyên gia lập mã video chung JVET của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, Hội nghị lần thứ 13: Marrakech, MA, từ ngày 9 đến 18 tháng 1 năm 2019, Tài liệu: JVET- M0301), cơ chế giảm yêu cầu bộ đệm dòng của ALF (phép lọc vòng lặp thích ứng) được bộc lộ. Tài liệu này sử dụng ý tưởng của các biên giới ảo (các VB) mà được dịch chuyển hướng lên trên so với các biên giới CTU nằm ngang một khoảng “N” mẫu. Phân loại khối ALF sửa đổi và phép lọc ALF sửa đổi được áp dụng cho các mẫu gần với biên giới ảo để giảm số lượng bộ đệm dòng yêu cầu. Phân loại khối ALF sửa đổi chỉ sử dụng các mẫu mà ở phía trên VB để phân loại khối 4 x 4 cho trước, mà ở phía trên VB. Tương tự, để phân loại khối 4 x 4 phía dưới VB, các mẫu thuộc những dòng phía dưới VB được sử dụng. Phép lọc ALF sửa đổi sử dụng sự kết hợp giữa phép hủy có điều kiện và các phiên bản cắt xén của phép lọc ALF gốc.

Phép lọc ALF cắt xén tại các biên giới ảo

Phiên bản cắt xén của các phép lọc được sử dụng để lọc các mẫu Luma thuộc những dòng gần với các biên giới ảo. Những phiên bản cắt xén này của các phép lọc có thể được sử dụng cho cả trường hợp N=4 và N=6. Tương tự, phiên bản cắt xén của các phép lọc cũng được sử dụng cho phép lọc ALF chroma.

Đối với ALF cắt xén, các hệ số phải được chuẩn hóa, tức là, tổng của các hệ số còn

lại phải bằng với tổng của các hệ số ban đầu. Fig. 4A đến Fig. 4F minh họa các ví dụ về chuẩn hóa hệ số cho phép xử lý lọc ALF luma cắt xén tại biên giới ảo. Fig. 4A minh họa hệ số ALF sửa đổi cho trường hợp trong đó điểm ảnh trên cùng (tức là, điểm ảnh c_0) nằm ngoài biên giới ảo. Hệ số c_2 được sửa đổi thành c_2' , trong đó $c_2' = c_2 + c_0$. Fig. 4B minh họa trường hợp trong đó điểm ảnh dưới cùng (tức là, c_0) nằm ngoài biên giới ảo. Fig. 4C minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng trên cùng (tức là, các điểm ảnh c_0 , c_1 , c_2 và c_3) nằm ngoài biên giới ảo. Vì vậy, hệ số c_5 , c_6 và c_7 được sửa đổi thành c_5' , c_6' và c_7' , trong đó $c_5' = c_5 + c_1$, $c_6' = c_6 + c_2 + c_0$, và $c_7' = c_7 + c_3$. Fig. 4D minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng dưới cùng (tức là, các điểm ảnh c_0 , c_1 , c_2 và c_3) nằm ngoài biên giới ảo. Fig. 4E minh họa hệ số ALF sửa đổi cho trường hợp trong đó ba hàng trên cùng (tức là, các điểm ảnh từ c_0 đến c_8) nằm ngoài biên giới ảo. Vì vậy, hệ số c_{10} , c_{11} và c_{12} được sửa đổi thành c_{10}' , c_{11}' và c_{12}' , trong đó $c_{10}' = c_{10} + c_4 + c_8$, $c_{11}' = c_{11} + c_5 + c_7 + c_1 + c_3$ và $c_{12}' = c_{12} + 2 \cdot c_6 + 2 \cdot c_2 + 2 \cdot c_0$. Fig. 4F minh họa hệ số ALF sửa đổi cho trường hợp trong đó ba hàng dưới cùng (tức là, các điểm ảnh từ c_0 đến c_8) nằm ngoài biên giới ảo.

Fig. 5A đến 5D minh họa các ví dụ về phép xử lý lọc ALF chroma cắt xén tại biên giới ảo. Fig. 5A minh họa hệ số ALF sửa đổi cho trường hợp trong đó điểm ảnh trên cùng (tức là, điểm ảnh c_0) nằm ngoài biên giới ảo. Hệ số c_2 được sửa đổi thành c_2' , trong đó $c_2' = c_2 + c_0$. Fig. 5B minh họa trường hợp trong đó điểm ảnh dưới cùng (tức là, c_0) nằm ngoài biên giới ảo. Fig. 5C minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng trên cùng (tức là, các điểm ảnh c_0 , c_1 , c_2 và c_3) nằm ngoài biên giới ảo. Vì vậy, hệ số c_5 và c_6 được sửa đổi thành c_5' và c_6' , trong đó $c_5' = c_5 + c_1 + c_3$ và $c_6' = c_6 + 2 \cdot c_2 + 2 \cdot c_0$. Fig. 5D minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng dưới cùng (tức là, các điểm ảnh c_0 , c_1 , c_2 và c_3) nằm ngoài biên giới ảo.

Biên giới ảo tại biên giới slice/brick/tile và biên giới ảo video 360

Theo JVET-O0016 (M. Zou, và các cộng sự, “Báo cáo JVET AHG: Các nghiên cứu triển khai (AHG16)”, Nhóm chuyên gia lập mã video chung JVET của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, hội nghị lần thứ 15: Gothenburg, SE, từ ngày 3 đến 12 tháng 7 năm 2019, Tài liệu: JVET- O0016), vấn đề bộ đệm dòng ALF dọc biên giới slice/brick/tile và biên giới ảo video 360 được báo cáo. Video 360 còn được gọi là video 360 độ hoặc videop thực tế ảo VR 360 độ (Virtual Reality - VR). Biên giới ảo ALF được sử dụng để loại bỏ yêu cầu bộ đệm dòng ALF, trong đó biên giới ảo được định vị

tại 4 dòng luma phía trên biên giới CTU dưới cùng ngoại trừ các CTU tại hàng CTU cuối cùng của hình ảnh. Ở bản mô tả dự thảo theo mô hình kiểm nghiệm VVC 6 (VTM6), biên giới ảo ALF không xuất hiện cho CTU nếu biên giới CTU dưới cùng của nó cũng là biên giới slice/brick/tile (có phép lọc vòng lặp khắp các biên giới vừa nêu được hủy bỏ) hoặc biên giới ảo video 360. Đối với video 360, hình ảnh 360 có thể chứa một hoặc nhiều “biên giới ảo” tương ứng với các viên liên tục giữa các bề mặt. Vì biên giới ảo ALF được hủy bỏ theo tiêu chuẩn VVC hiện tại, nên nó có những hệ quả sau:

- Đối với các bộ giải mã, việc không biết trước các biên giới slice/brick/tile (tức là những ứng dụng độ trễ thấp) cho thấy rằng độ đệm dòng ALF cần được khôi phục lại. Để xử lý phép lọc ALF các dòng dưới cùng của CTU hiện thời (ví dụ các dòng luma 5, 6 và 7 phía trên biên giới CTU dưới cùng), các bộ giải mã cần biết liệu biên giới CTU dưới cùng hiện thời có trùng với các dạng biên giới khác. Tuy nhiên, thông tin này là không biết cho đến khi slice/brick/tile tiếp theo được giải mã.
- Ngay cả khi các biên giới slice/brick/tile đã biết trước (ví dụ giải mã CABAC được tách rời hoàn toàn tại phân mức khung hình), các bộ giải mã cần chạy các phép lọc vòng lặp (phép giải khối, SAO và ALF) với tốc độ lúc nào cũng 68 dòng trên từng bộ xử lý giải mã video VDPU 64x64 (Video Decoder Processing Unit - VDPU) (với mức dự phòng khoảng 6%) để tránh sử dụng bộ đệm dòng ALF.

Theo JVET-O0625 (H. Liu, và các cộng sự, “Non-CE5: Phương pháp đệm các mẫu tại các biên giới biến đổi theo ALF”, Nhóm chuyên gia lập mã video chung JVET của ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11, hội nghị lần thứ 15: Gothenburg, SE, từ ngày 3 đến 12 tháng 7 năm 2019, Tài liệu: JVET- O0016), hai phương pháp được đề xuất để giải quyết vấn đề nêu trên:

1. Biên giới ảo ALF có thể ứng dụng cho các CU có biên giới dưới cùng là slice/brick/biên giới ảo mà không phụ thuộc việc sử dụng các mẫu khắp các biên giới.

2. Theo phương pháp đệm 2-chiều đối xứng, khi mẫu ở tại hai biên giới, ví dụ, một biên giới ở phía trên và một biên giới ở phía dưới, số lượng các mẫu được đệm được xác định theo biên giới gần hơn như được thể hiện trên Fig. 6A và Fig. 6B. Mặt khác, khi suy dẫn thông tin phân loại, chỉ 4 dòng giữa các biên giới phía trên và phía dưới trên Fig. 6A và Fig. B được sử dụng. Fig. 6A minh họa ví dụ trong đó ba hàng nằm dưới

slice/brick/biên giới ảo hoặc biên giới ảo 360 độ (610) đối với mẫu hiện thời (tức là, c12) sẽ được lọc. Phép đệm đối xứng sẽ đệm ba hàng phía dưới slice/brick/biên giới ảo hoặc biên giới ảo 360 độ (610) và ba hàng tương ứng phía trên, trong đó các mẫu được đệm được thể hiện ở các ô được kẻ được xọc nghiêng. Fig. 6B minh họa ví dụ hai hàng nằm dưới slice/brick/biên giới ảo hoặc biên giới ảo 360 độ (610) đối với mẫu hiện thời (tức là, c12) sẽ được lọc. Phép đệm đối xứng sẽ đệm hai hàng phía dưới slice/brick/biên giới ảo hoặc biên giới ảo 360 độ (620) và hai hàng tương ứng phía trên, trong đó các mẫu được đệm được thể hiện ở các ô được kẻ được xọc nghiêng.

Ở sáng chế này, nhiều vấn đề liên quan đến biên giới ALF khác nhau được giải quyết và các giải pháp nhằm nâng cao hiệu suất, nhằm hợp nhất nhiều hệ thống hoặc nhằm đơn giản hóa phép xử lý biên giới ALF được bộc lộ.

Bản chất kỹ thuật của sáng chế

Các phương pháp và thiết bị dùng để xử lý phép lọc vòng lặp video tái tạo được bộc lộ. Theo phương pháp này, các điểm ảnh tái tạo được nhận. Phép lọc vòng lặp mong muốn được áp dụng cho các mẫu được tái tạo, trong đó nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn vừa nêu được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn. Các mẫu tái tạo đã lọc được cấp làm kết quả giải mã video hoặc cấp cho quá trình mã hóa hoặc giải mã video tiếp theo.

Theo một phương án, phép lọc vòng lặp mong muốn tương ứng với phép lọc giải khối. Theo phương án khác, phép lọc vòng lặp mong muốn tương ứng với phép lọc vòng lặp thích ứng, phép bù thích ứng mẫu hoặc kết hợp của cả hai phép vừa nêu.

Theo một phương án, biên giới mong muốn tương ứng với biên giới slice, biên giới tile hoặc cả biên giới slice và tile. Theo phương án khác, biên giới mong muốn tương ứng với biên giới tile, biên giới hình ảnh, hoặc biên giới ảo 360.

Theo phương pháp khác, các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu được nhận. ALF (Phép lọc vòng lặp thích ứng) thứ nhất được áp dụng đối với các mẫu chroma được tái tạo cho mẫu chroma được tái tạo mục tiêu để tạo ra mẫu chroma được lọc thứ nhất. ALF thứ hai được áp dụng đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ hai cho mẫu chroma được tái tạo mục tiêu, trong đó biên giới ảo được xác định và một bộ các mẫu luma được tái

tạo liên đới được định vị ở phía còn lại của biên giới ảo từ mẫu chroma được tái tạo mục tiêu tương ứng, một bộ các mẫu luma được tái tạo liên đới vừa nêu được loại khỏi các đầu vào ALF thứ hai. Mẫu chroma được lọc cuối cùng được tạo ra bằng cách tổ hợp mẫu chroma được lọc thứ nhất và mẫu chroma được lọc thứ hai, trong đó mẫu chroma được lọc cuối cùng được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã video tiếp theo.

Theo một phương án, ALF thứ hai được cấu hình để loại bỏ các đầu vào tương ứng với một hoặc nhiều mẫu luma được tái tạo liên đới đối xứng được định vị tại các vị trí đối xứng của một bộ các mẫu luma được tái tạo liên đới vừa nêu. Theo phương án khác, ALF thứ hai tương ứng với phép lọc đối xứng được cắt xén mà không có các đầu vào của một bộ các mẫu luma được tái tạo liên đới vừa nêu. Theo phương án khác, hình dạng lọc sửa đổi của ALF thứ hai được tạo ra, trong đó hình dạng lọc sửa đổi vừa nêu được sửa đổi từ hình dạng lọc mặc định.

Theo phương pháp khác nữa, các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu được nhận. ALF luma được áp dụng cho các mẫu luma được tái tạo liên đới để tạo ra mẫu luma được lọc. ALF thứ nhất được áp dụng đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ nhất cho mẫu chroma được tái tạo mục tiêu, trong đó nếu phép lọc vòng lặp ở khắp biên giới mong muốn được hủy bỏ, các mẫu luma được tái tạo liên đới tương ứng được tạo ra cho ALF thứ nhất theo cách thức tương tự như cho xử lý ALF luma, và mẫu chroma được lọc thứ nhất sau đó được kết hợp với mẫu chroma được lọc thứ hai được suy dẫn dựa vào các mẫu chroma được tái tạo để tạo ra mẫu chroma được lọc cuối cùng. Mẫu chroma được lọc cuối cùng và mẫu chroma được lọc được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã video tiếp theo.

Theo một phương án, biên giới mong muốn tương ứng với biên giới slice, biên giới tile, biên giới brick hoặc biên giới hình ảnh. Theo phương án khác, biên giới mong muốn tương ứng với biên giới ảo 360.

Theo một phương án, nếu một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng không hiện có, thì cả ALF thứ nhất và ALF luma vừa nêu sử dụng phép đệm lặp để tạo ra một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng vừa nêu. Theo phương án khác, nếu một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng không hiện có,

thì cả ALF thứ nhất và ALF luma vừa nêu được hủy bỏ. Một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng vừa nêu không hiện có khi các mẫu luma được tái tạo liên đới tương ứng này nằm ngoài biên giới. Theo phương án nữa, cả ALF thứ nhất và ALF luma vừa nêu tạo ra các mẫu luma được tái tạo liên đới tương ứng theo phép xử lý biên giới ảo ALF, trong đó một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng và một hoặc nhiều mẫu luma được tái tạo liên đới đối xứng tại các vị trí đối xứng của một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng vừa nêu được loại khỏi ALF thứ nhất và ALF luma vừa nêu.

Mô tả vắn tắt các hình vẽ

Fig. 1A minh họa hệ thống mã hóa video nội ảnh/liên ảnh thích ứng được lấy làm ví dụ có tích hợp phép xử lý vòng lặp kín DF, SAO và ALF;

Fig. 1B minh họa hệ thống giải mã video nội ảnh/liên ảnh thích ứng được lấy làm ví dụ có tích hợp phép xử lý vòng lặp kín DF, SAO và ALF;

Fig. 2A minh họa cấu trúc của CC-ALF tương ứng với các phép lọc vòng lặp khác theo JVET-O0636 trong đó xử lý ALF được thực hiện sau phép SAO tương ứng;

Fig. 2B minh họa phép lọc dạng hình thoi được áp dụng đối với kênh luma cho từng thành phần chroma theo JVET-O0636;

Fig. 3 minh họa ví dụ về các VB được dịch chuyển lên trên so với các biên giới LCU nằm ngang một khoảng N điểm ảnh;

Fig. 4A đến Fig. 4F minh họa các ví dụ về chuẩn hóa hệ số cho phép xử lý lọc ALF luma cắt xén tại biên giới ảo, trong đó Fig. 4A minh họa hệ số ALF sửa đổi cho trường hợp trong đó điểm ảnh trên cùng nằm ngoài biên giới ảo; Fig. 4B minh họa trường hợp trong đó điểm ảnh dưới cùng nằm ngoài biên giới ảo; Fig. 4C minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng trên cùng nằm ngoài biên giới ảo; Fig. 4D minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng dưới cùng nằm ngoài biên giới ảo; Fig. 4E minh họa hệ số ALF sửa đổi cho trường hợp trong đó ba hàng trên cùng nằm ngoài biên giới ảo; và Fig. 4F minh họa hệ số ALF sửa đổi cho trường hợp trong đó ba hàng dưới cùng nằm ngoài biên giới ảo;

Fig. 5A đến 5D minh họa các ví dụ về phép xử lý lọc ALF chroma cắt xén tại biên giới ảo, trong đó Fig. 5A minh họa hệ số ALF sửa đổi cho trường hợp trong đó điểm ảnh

trên cùng nằm ngoài biên giới ảo; Fig. 5B minh họa trường hợp trong đó điểm ảnh dưới cùng nằm ngoài biên giới ảo; Fig. 5C minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng trên cùng nằm ngoài biên giới ảo; và Fig. 5D minh họa hệ số ALF sửa đổi cho trường hợp trong đó hai hàng dưới cùng nằm ngoài biên giới ảo;

Fig. 6A và Fig. 6B minh họa các ví dụ về phương pháp đệm 2-chiều đối xứng, trong đó Fig. 6A minh họa ví dụ trong đó ba hàng nằm dưới slice/brick/biên giới ảo hoặc biên giới ảo 360 độ; và Fig. 6B minh họa ví dụ hai hàng nằm dưới slice/brick/biên giới ảo hoặc biên giới ảo 360 độ;

Fig. 7 minh họa lưu đồ của phép lọc vòng lặp video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn;

Fig. 8 minh họa lưu đồ của phép xử lý ALF video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó hình dạng CC-ALF được thay đổi khắp biên giới ảo; và

Fig. 9 minh họa lưu đồ của phép xử lý ALF video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó CC-ALF và ALF luma xử lý các mẫu không hiện có theo cùng cách thức.

Mô tả chi tiết sáng chế

Phần mô tả dưới đây là phương án tốt nhất thực hiện sáng chế. Phần mô tả đó được trình bày nhằm mục đích minh họa những nguyên lý chung của sáng chế và không được hiểu là nhằm mục đích giới hạn sáng chế. Phạm vi của sáng chế được xác định tốt nhất bằng cách tham chiếu đến các điểm yêu cầu bảo hộ đính kèm.

Phương pháp 1: Hủy bỏ ALF khi lọc vòng lặp khắp các biên giới được hủy bỏ

Để áp dụng phương pháp thứ nhất của JVET-O0625 và cải thiện việc sử dụng các mẫu ở khắp các biên giới (tức là, phép lọc vòng lặp khắp các biên giới được hủy bỏ), một số phương pháp được đề xuất. Theo một phương án, ALF được hủy bỏ đối với hàng các khối 4x4 cuối cùng trong slice hoặc brick khi lọc vòng lặp khắp các biên giới không được phép. Theo phương án khác, ý tưởng vừa nêu được mở rộng để xử lý các biên giới thẳng đứng, mà có thể là biên giới thẳng đứng của slice/brick/tile/hình ảnh. Theo

phương án khác, các hàng của khối 4×4 dưới cùng và trên cùng ở slice/brick/tile không được lọc bằng ALF khi lọc vòng lặp khắp các biên giới không được phép. Theo phương án khác, các cột của các khối 4×4 ngoài cùng bên trái và ngoài cùng bên phải ở slice/brick/tile cũng không được lọc bằng ALF khi lọc vòng lặp khắp các biên giới không được phép. Theo phương án khác, các hàng của các khối 4×4 dưới cùng và trên cùng ở hình ảnh không được lọc bằng ALF. Theo phương án khác, các khối 4×4 ngoài cùng bên trái và ngoài cùng bên phải ở hình ảnh không được lọc bằng ALF.

Theo một phương án, khi lọc vòng lặp khắp các biên giới được hủy bỏ, các phép lọc vòng lặp kín khác nhau được hủy bỏ đối với những biên giới có phép xử lý vòng lặp kín đã hủy bỏ. Ví dụ, khi lọc vòng lặp khắp các biên giới cho một slice được hủy bỏ, thì chỉ các biên giới slice biên trái và trên cùng không được lọc bằng phép lọc giải khối (DF) và bù thích ứng mẫu (SAO). Theo đó, ALF cũng không được áp dụng tại các biên giới slice biên trái và trên cùng. Theo ví dụ khác, khi lọc vòng lặp khắp các biên giới cho một slice được hủy bỏ, thì toàn bộ các biên giới slice, bao gồm các biên giới dưới cùng, bên phải, bên trái, trên cùng không được lọc bằng phép lọc giải khối và bù thích ứng mẫu. Theo đó, ALF cũng không được áp dụng ở tất cả các biên giới slice.

Theo phương án khác, phương pháp đã đề xuất có thể được áp dụng cho các biên giới ảo 360. Đối với một số biên giới ảo 360, các mẫu liền kề trong một hình ảnh được biểu diễn theo miền không gian 2D (ví dụ, các mẫu liền kề ở suốt biên giới liên tục giữa hai bề mặt) là các mẫu liền kề không thực (tức là, không liền kề về mặt không gian) trong miền không gian 3D. Vì vậy, các mẫu liền kề không được sử dụng cho phép lọc. Trong trường hợp này, ALF có thể được hủy bỏ để tránh sử dụng các mẫu liền kề không đúng. Các biên giới ảo 360 có thể là các biên giới nằm ngang hoặc thẳng đứng. Theo đó, nếu nó là biên giới nằm ngang, hai hàng của khối 4×4 tương ứng với các hàng của khối 4×4 phía trên và phía dưới biên giới đó một cách tương ứng không được lọc bằng ALF. Nếu nó là biên giới thẳng đứng, thì hai cột của khối 4×4 gần nhất với biên giới đó không được lọc bằng ALF.

Khối 4×4 được nêu ở trên có thể là khối $M \times N$ trong phép xử lý ALF, trong đó M và N có thể là số nguyên bất kỳ.

Phương pháp 2: Rút gọn CC-ALF thành ALF nhiều đầu vào

Trong CC-ALF, một phép lọc phụ được chèn vào cho thành phần chroma và đầu ra

được lọc sẽ là tổng trọng số của phép lọc chroma gốc ban đầu và phép lọc phụ. Theo một phương án, chỉ một thẻ lọc phụ có đầu vào từ mẫu luma tương ứng được chèn vào phép lọc chroma gốc ban đầu thay cho một phép lọc phụ có các đầu vào từ các mẫu luma tương ứng. Theo phương án khác, thuật toán cắt phi tuyến tính có thể cũng được áp dụng cho thẻ lọc vừa nêu. Theo một phương án, hệ số cắt có thể được báo hiệu cho thẻ lọc đó. Theo phương án khác, thuật toán cắt phi tuyến tính không được áp dụng cho thẻ lọc vừa nêu. Vì vậy, hệ số cắt không được báo hiệu cho thẻ lọc đó.

Theo một phương án, một cờ được báo hiệu theo các tham số lọc ALF để biểu thị liệu thẻ lọc đó được sử dụng hay không. Mẫu luma tương ứng được quyết định theo định dạng chroma. Ví dụ, nếu định dạng video là YUV444, thì mẫu luma tương ứng có cùng vị trí không gian với mẫu chroma sẽ được xử lý. Nếu định dạng video là YUV422 hoặc YUV420, thì mẫu luma tương ứng có thể có vị trí không gian được định trước trong số các mẫu luma tương ứng của mẫu chroma sẽ được xử lý. Ví dụ, đối với YUV420, có bốn mẫu luma tương ứng cho một mẫu chroma. Đầu vào của thẻ lọc phụ từ thành phần luma cho thành phần chroma có thể là một trong số bốn mẫu luma. Việc chọn vừa nêu có thể được định trước trong tiêu chuẩn lập mã video. Theo phương án khác, mẫu luma tương ứng là tổng trọng số của tất cả các mẫu luma tương ứng. Ví dụ, giá trị trung bình của bốn mẫu luma cho YUV420, giá trị trung bình của hai mẫu luma bên trái cho YUV420, hoặc giá trị trung bình của hai mẫu luma cho YUV422.

Theo phương án khác, độ chính xác của các hệ số lọc trong phép lọc phụ trong CC-ALF tương thích với thiết kế gốc ban đầu.

Theo phương án khác, phương pháp đã đề xuất được áp dụng cho ALF luma. Theo đó, hai thẻ lọc phụ có đầu vào từ chroma được thêm vào cho ALF luma. Theo một phương án, chỉ một thẻ lọc phụ được thêm vào cho ALF luma và đầu vào của thẻ lọc phụ là tổng trọng số của các thành phần chroma.

Phương pháp 3: CC-ALF có xử lý biên giới ảo

Theo ALF, để loại bỏ yêu cầu bộ đệm dòng, biên giới ảo được đưa ra. Vị trí của các biên giới ảo là 4 dòng luma phía trên các biên giới hàng CTU và hai hàng chroma phía trên các biên giới hàng CTU. Khi CC-ALF được áp dụng, nó cần tránh sử dụng các mẫu Luma ở khắp các biên giới ảo. Ví dụ, ở thiết kế gốc ban đầu, phép lọc hình thoi 5x6, như được thể hiện trên Fig. 2B, được sử dụng để suy dẫn tín hiệu được lọc màu chéo cho

thành phần chroma. Phép lọc hình thoi 5×6 ở ví dụ này có thể được xét làm hình dạng lọc mặc định. Nhưng mặt khác, phép lọc hình thoi 5×6 cũng có thể được sử dụng làm ví dụ của CC-ALF, nên sáng chế không bị giới hạn chỉ ở một hình dạng lọc cụ thể nào. Nếu các mẫu luma liền kề của mẫu luma tương ứng trong dấu hiệu lọc được định vị ở phía còn lại của các biên giới ảo, phép lọc hoặc hình dạng lọc cần được thay đổi để tránh sử dụng các mẫu được định vị ở phía còn lại của biên giới ảo. Theo một phương án, hình dạng lọc được giảm thành dạng lọc hình thoi 5×4 khi chỉ một hàng luma được định vị ở phía còn lại của biên giới ảo. Nói cách khác, một hàng luma ở phía còn lại của biên giới ảo và một hàng luma được định vị tại các vị trí đối xứng tương ứng của CC-ALF được loại bỏ phép lọc. Nếu có hai hàng luma được định vị ở phía còn lại của biên giới ảo, thì hình dạng lọc giảm thành dạng lọc hình thoi 5×2 . Nói cách khác, hai hàng luma ở phía còn lại của biên giới ảo và hai hàng luma được định vị tại các vị trí đối xứng tương ứng của CC-ALF được loại phép lọc. Theo phương án khác, phương pháp đệm được sử dụng để tạo ra để tạo ra các mẫu được định vị ở phía còn lại của biên giới ảo thay vì thay đổi hình dạng lọc. Phương pháp đệm vừa nêu có thể là mô hình lặp lại, mô hình đối xứng chẵn, mô hình đối xứng lẻ, hoặc phương pháp được sử dụng ở phép xử lý biên giới ảo ALF. Theo phương án khác, CC-ALF được hủy bỏ khi có ít nhất một mẫu luma được định vị ở phía còn lại của biên giới ảo.

Phương pháp 4: CC-ALF có xử lý biên giới slice/brick/tile/hình ảnh/biên giới 360

Tại các biên giới slice/brick/tile, khi lọc vòng lặp khắp các biên giới được hủy bỏ, phương pháp được sử dụng để tạo ra các mẫu luma tương ứng theo CC-ALF cần giống với phương pháp được sử dụng để tạo ra các mẫu luma tương ứng trong ALF luma. Theo một phương án, khi lọc vòng lặp ở khắp (ngoài) các biên giới slice/brick/tile được hủy bỏ, ALF có VB (biên giới ảo) được sử dụng để tạo ra các mẫu luma tương ứng trong ALF luma. Theo đó, ALF có VB được sử dụng để tạo ra các mẫu luma tương ứng trong CC-ALF. Theo một phương án, khi lọc vòng lặp ở khắp các biên giới slice/brick/tile được hủy bỏ, ALF luma được hủy bỏ đối với các mẫu nếu ALF luma cần các mẫu không hiện có mà nằm ngoài biên giới. Theo đó, CC-ALF cũng được hủy bỏ đối với các mẫu chroma tương ứng. Theo một phương án, khi lọc vòng lặp ở khắp các biên giới slice/brick/tile được hủy bỏ, phép đệm lặp được sử dụng để tạo ra các mẫu luma tương ứng trong ALF luma. Theo đó, phép đệm lặp cũng được sử dụng để tạo ra các mẫu luma tương ứng trong CC-ALF. Nói cách khác, phép đệm được áp dụng để thay thế các mẫu

luma tương ứng nằm ngoài các biên giới trong ALF luma và CC-ALF. Các mẫu luma tương ứng được thay thế là các mẫu không hiện có. Phương pháp đã đề xuất cũng có thể được áp dụng tại các biên giới ảnh hoặc biên giới ảo 360.

Bất kỳ phương pháp nào trong số các phương pháp đã được đề xuất ở trên có thể được thực hiện ở các bộ mã hóa và/hoặc các bộ giải mã. Ví dụ, bất kỳ phương pháp nào trong số các phương pháp đã được đề xuất ở trên có thể được thực hiện ở môđun lọc vòng kín của bộ mã hóa và/hoặc bộ giải mã. Ngoài ra, bất kỳ phương pháp nào trong số các phương pháp được đề xuất ở trên có thể được thực hiện dưới dạng mạch được tích hợp vào môđun lọc vòng kín của bộ mã hóa và/hoặc bộ giải mã.

Fig. 7 minh họa lưu đồ của phép lọc vòng lặp video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn. Các bước được thể hiện trên lưu đồ 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. 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 thiết bị xử lý được bố trí để thực hiện các bước trong lưu đồ. Theo phương pháp này, các điểm ảnh tái tạo được nhận ở bước 710. Phép lọc vòng lặp mong muốn được áp dụng cho các mẫu được tái tạo ở bước 720, trong đó nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn vừa nêu được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn. Các mẫu tái tạo được lọc sau đó được cấp làm kết quả giải mã video hoặc được cấp cho quá trình mã hóa hoặc giải mã video tiếp theo ở bước 730.

Fig. 8 minh họa lưu đồ của phép xử lý ALF video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó hình dạng CC-ALF được thay đổi khắp biên giới ảo. Theo phương pháp này, các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu được nhận ở bước 810. ALF thứ nhất được áp dụng đối với các mẫu chroma được tái tạo cho mẫu chroma được tái tạo mục tiêu để tạo ra mẫu chroma được lọc thứ nhất ở bước 820. ALF thứ hai được áp dụng đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ hai cho mẫu chroma được tái tạo mục tiêu ở bước 820, trong đó biên giới ảo được xác định và một bộ các mẫu luma được tái tạo liên đới được định vị ở phía còn lại của biên giới ảo từ mẫu chroma được tái tạo mục

tiêu, một bộ các mẫu luma được tái tạo liên đới vừa nêu được loại khỏi các đầu vào ALF thứ hai. Mẫu chroma được lọc cuối cùng được tạo ra bằng cách tổ hợp mẫu chroma được lọc thứ nhất và mẫu chroma được lọc thứ hai ở bước 840, trong đó mẫu chroma được lọc cuối cùng được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã video tiếp theo.

Fig. 9 minh họa lưu đồ của phép xử lý ALF video tái tạo được lấy làm ví dụ theo phương án của sáng chế, trong đó CC-ALF và ALF luma xử lý các mẫu không hiện có theo cùng cách thức. Theo phương pháp này, các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu được nhận ở bước 910. ALF luma được áp dụng cho các mẫu luma được tái tạo liên đới để tạo ra mẫu luma được lọc ở bước 920. ALF thứ nhất được áp dụng đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ nhất cho mẫu chroma được tái tạo mục tiêu ở bước 930, trong đó nếu phép lọc vòng lặp ở khắp biên giới mong muốn được hủy bỏ, các mẫu luma được tái tạo liên đới tương ứng được tạo ra cho ALF thứ nhất theo cách thức tương tự như cho xử lý ALF luma, trong đó mẫu chroma được lọc thứ nhất được kết hợp với mẫu chroma được lọc thứ hai được suy dẫn dựa vào các mẫu chroma được tái tạo để tạo ra mẫu chroma được lọc cuối cùng; và trong đó mẫu chroma được lọc cuối cùng và mẫu chroma được lọc được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã video tiếp theo. Cần được chú ý là, các mẫu không hiện có vừa nêu là các mẫu luma được tái tạo liên đới nằm ngoài biên giới mong muốn.

Các lưu đồ được thể hiện nhằm mục đích minh họa ví dụ về lập mã video theo sáng chế. Người có chuyên môn 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 lệch khỏi nội dung của sáng chế. Ở bản mô tả này, cú pháp cú pháp và các ngữ nghĩa cụ thể được sử dụng để minh họa các ví dụ thực hiện các phương án của sáng chế. Người có chuyên môn có thể thực hiện sáng chế bằng cách thay thế cú pháp và các ngữ nghĩa này bằng cú pháp và các ngữ nghĩa tương đương khác mà không lệch khỏi nội dung của sáng chế.

Phần mô tả đã nêu được trình bày nhằm giúp người bình thường trong lĩnh vực kỹ thuật này có thể thực hiện sáng chế như được nêu theo nội dung của ứng dụng nhất định và yêu cầu của nó. Nhiều sửa đổi khác nhau đối với các phương án được mô tả sẽ là điều dễ hiểu đối với những người có chuyên môn trong lĩnh vực kỹ thuật này, và những

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 dự định sẽ được giới hạn chỉ ở các phương án cụ thể đã được thể hiện và được mô tả, mà sẽ được tuân theo phạm vi rộng nhất phù hợp với những nguyên lý và những dấu hiệu mới được bộc lộ ở bản mô tả này. Ở phần mô tả chi tiết sáng chế nêu trên, nhiều phần chi tiết cụ thể khác nhau được minh họa để giúp hiểu toàn diện sáng chế. Tuy nhiên, sẽ được hiểu bởi những người có chuyên môn trong lĩnh vực kỹ thuật này đó là 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 mã phần cứng, phần mềm khác nhau, hoặc kết hợp cả mã phần cứng và mã phần mềm. Ví dụ, phương án của sáng chế có thể là một hoặc mạch được tích hợp thành chip nén video hoặc mã chương trình được tích hợp thành phần mềm nén video để thực hiện quá trình xử lý được mô tả ở đây. Phương án của sáng chế có thể còn là mã chương trình được xử lý trên bộ xử lý tín hiệu số (DSP) để thực hiện quá trình xử lý được mô tả ở đây. Sáng chế có thể còn sử dụng số lượng các hàm 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 đoạn mã phần mềm hoặc đoạn mã vi chương trình có thể đọc được bằng máy mà định ra các phương pháp cụ thể được thực hiện bởi sáng chế. Mã phần mềm hoặc mã vi chương trình có thể được phát triển theo các ngôn ngữ lập trình khác nhau và các định dạng hoặc kiểu khác nhau. Mã phần mềm có thể cũng được tương thích theo các nền tảng mục tiêu khác nhau. Tuy nhiên, các định dạng, kiểu, ngôn ngữ lập trình khác nhau của các mã phần mềm và các phương tiện cấu hình nên lập trình 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 hoặc các dấu hiệu cơ bản của sáng chế. Các ví dụ được mô tả được xét theo toàn bộ các khía cạnh chỉ nhằm mục đích minh họa chứ không nhằm mục đích giới hạn. Vì vậy, phạm vi của sáng chế được thể hiện bằng các điểm yêu cầu bảo hộ đính kèm chứ không phải phần mô tả ở trên. Mọi thay đổi mà thuộc nội dung và phạm vi tương đương của các điểm yêu cầu bảo hộ đều được xem là thuộc phạm vi bảo hộ của sáng chế.

Sáng chế này hưởng quyền ưu tiên của đơn xin cấp bằng sáng chế Hoa Kỳ số 62/878,379 nộp ngày 25/07/2019. Nội dung của chúng được tham khảo kết hợp ở đây.

YÊU CẦU BẢO HỘ

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

nhận các mẫu được tái tạo;

áp dụng phép lọc vòng lặp mong muốn đối với các mẫu được tái tạo, trong đó nếu lọc vòng lặp không được phép cho biên giới mong muốn, thì phép lọc vòng lặp mong muốn vừa nêu được hủy bỏ ở phía trên cùng, dưới cùng, bên trái và bên phải của biên giới mong muốn; và

cấp các mẫu tái tạo được lọc làm kết quả giải mã video hoặc cho quá trình mã hóa hoặc giải mã video tiếp theo.

2. Phương pháp theo điểm 1, trong đó phép lọc vòng lặp mong muốn tương ứng với phép lọc giải khối, phép lọc vòng lặp thích ứng, bù thích ứng mẫu hoặc kết hợp của cả phép lọc vòng lặp thích ứng và phép bù thích ứng mẫu.

3. Phương pháp theo điểm 1, trong đó biên giới mong muốn tương ứng với biên giới slice, biên giới tile hoặc cả biên giới slice và tile.

4. Phương pháp theo điểm 1, trong đó biên giới mong muốn tương ứng với biên giới tile, biên giới hình ảnh, hoặc biên giới ảo 360.

5. Phương pháp lập mã video, phương pháp này bao gồm các bước:

nhận các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu;

áp dụng ALF (Phép lọc vòng lặp thích ứng) thứ nhất đối với các mẫu chroma được tái tạo cho mẫu chroma được tái tạo mục tiêu để tạo ra mẫu chroma được lọc thứ nhất;

áp dụng ALF thứ hai đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ hai cho mẫu chroma được tái tạo mục tiêu, trong đó biên giới ảo được xác định và một bộ các mẫu luma được tái tạo liên đới được định vị ở phía còn lại của biên giới ảo từ mẫu chroma được tái tạo mục tiêu, một bộ các mẫu luma được tái tạo liên đới vừa nêu được loại khỏi các đầu vào ALF thứ hai; và

tạo ra mẫu chroma được lọc cuối cùng bằng cách tổ hợp mẫu chroma được lọc thứ nhất và mẫu chroma được lọc thứ hai, trong đó mẫu chroma được lọc cuối cùng được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã

video tiếp theo.

6. Phương pháp theo điểm 5, trong đó ALF thứ hai được cấu hình để loại bỏ các đầu vào tương ứng với một hoặc nhiều mẫu luma được tái tạo liên đới đối xứng được định vị tại các vị trí đối xứng của một bộ các mẫu luma được tái tạo liên đới vừa nêu.

7. Phương pháp theo điểm 5, trong đó ALF thứ hai tương ứng với phép lọc đối xứng được cắt xén mà không có các đầu vào của tương ứng với một bộ các mẫu luma được tái tạo liên đới vừa nêu.

8. Phương pháp theo điểm 5, trong đó hình dạng lọc sửa đổi của ALF thứ hai được tạo ra, trong đó hình dạng lọc sửa đổi vừa nêu được sửa đổi từ hình dạng lọc mặc định.

9. Phương pháp lập mã video, phương pháp này bao gồm các bước:

nhận các mẫu chroma được tái tạo và các mẫu luma được tái tạo liên đới trong hình ảnh màu;

áp dụng ALF luma (Phép lọc vòng lặp thích ứng) đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu luma được lọc;

áp dụng ALF thứ nhất đối với các mẫu luma được tái tạo liên đới để tạo ra mẫu chroma được lọc thứ nhất cho mẫu chroma được tái tạo mục tiêu, trong đó nếu phép lọc vòng lặp ở khắp biên giới mong muốn được hủy bỏ, các mẫu luma được tái tạo liên đới tương ứng được tạo ra cho ALF thứ nhất theo cách thức tương tự như cho xử lý ALF luma, và trong đó mẫu chroma được lọc thứ nhất được kết hợp với mẫu chroma được lọc thứ hai được suy dẫn dựa vào các mẫu chroma được tái tạo để tạo ra mẫu chroma được lọc cuối cùng; và

trong đó mẫu chroma được lọc cuối cùng và mẫu chroma được lọc được sử dụng làm kết quả giải mã video hoặc được sử dụng cho quá trình mã hóa hoặc giải mã video tiếp theo.

10. Phương pháp theo điểm 9, trong đó biên giới mong muốn tương ứng với biên giới slice, biên giới tile, biên giới brick hoặc biên giới hình ảnh.

11. Phương pháp theo điểm 9, trong đó nếu một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng không hiện có, thì cả ALF thứ nhất và ALF luma vừa nêu áp dụng phép đệm lặp để tạo ra một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng vừa nêu.

12. Phương pháp theo điểm 9, trong đó nếu một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng không hiện có, thì cả ALF thứ nhất và ALF luma vừa nêu được hủy bỏ.

13. Phương pháp theo điểm 9, trong đó cả ALF thứ nhất và ALF luma tạo ra các mẫu luma được tái tạo liên đới tương ứng theo phép xử lý biên giới ảo ALF, trong đó một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng và một hoặc nhiều mẫu luma được tái tạo liên đới đối xứng tại các vị trí đối xứng của một hoặc nhiều mẫu luma được tái tạo liên đới tương ứng vừa nêu được loại khỏi ALF thứ nhất và ALF luma.

1/9

Fig. 1A

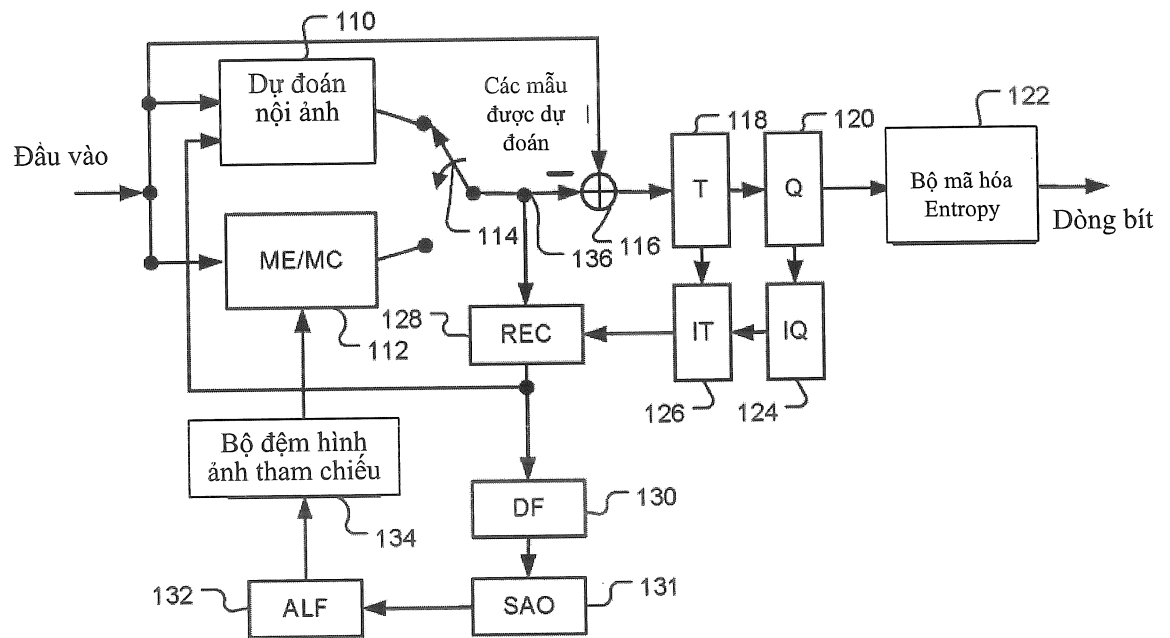
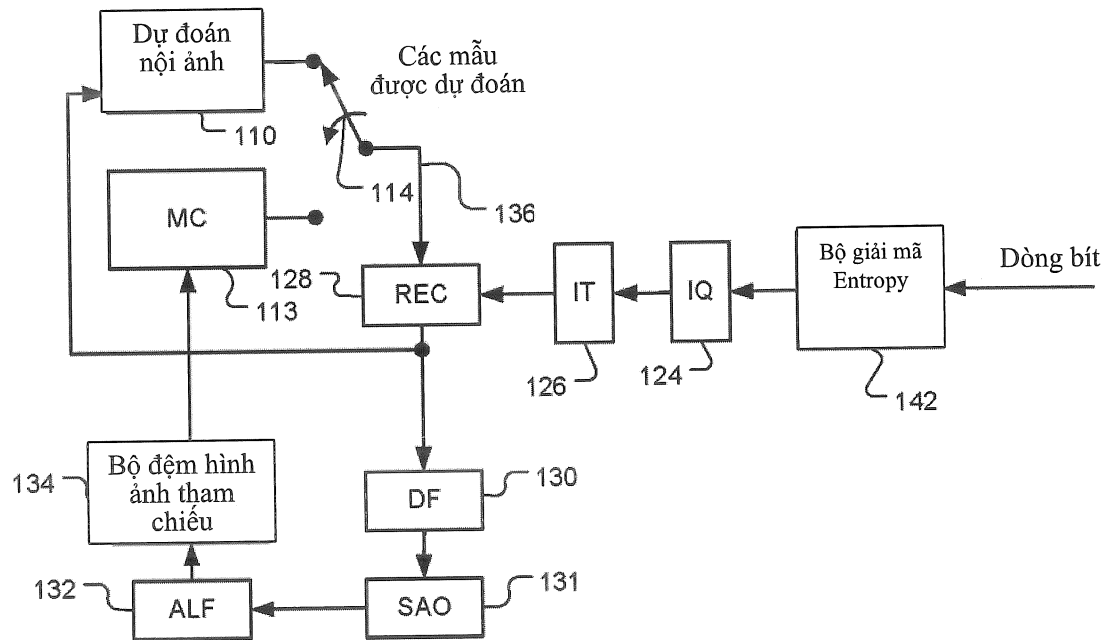


Fig. 1B



3/9

Fig. 2A

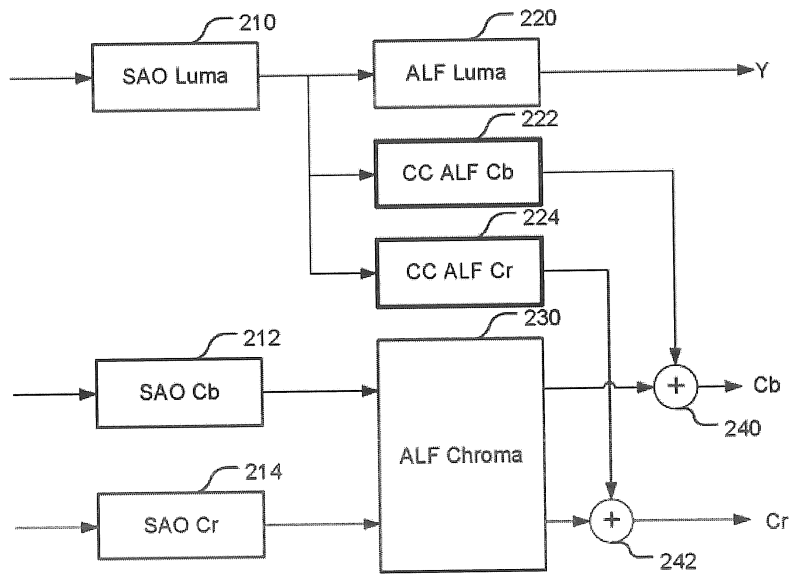
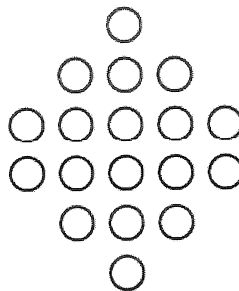
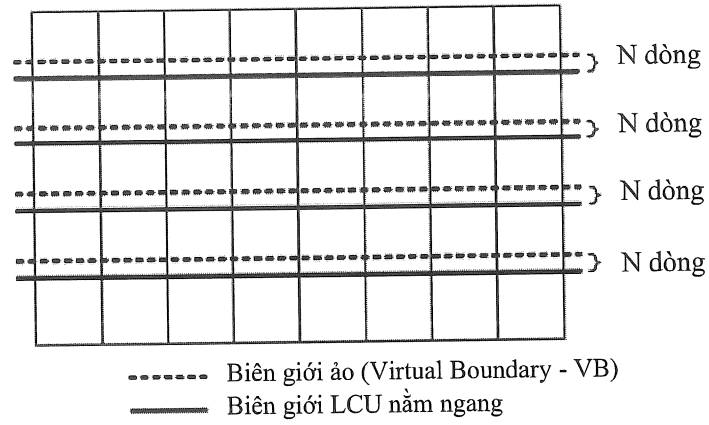


Fig. 2B



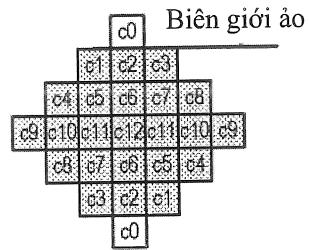
4/9

Fig. 3



5/9

Fig. 4A



$$c2' = c2 + c0$$

Fig. 4B

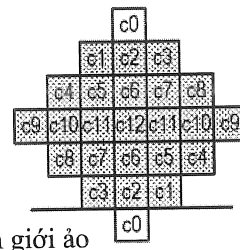
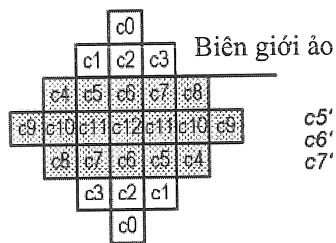


Fig. 4C



$$\begin{aligned} c5' &= c5 + c1 \\ c6' &= c6 + c2 + c0 \\ c7' &= c7 + c3 \end{aligned}$$

Fig. 4D

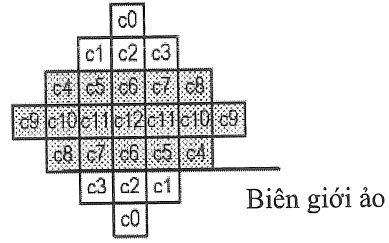
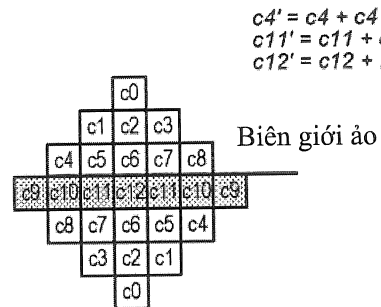


Fig. 4E



$$\begin{aligned} c4' &= c4 + c4 + c8 \\ c11' &= c11 + c5 + c7 + c1 + c3 \\ c12' &= c12 + 2*c6 + 2*c2 + 2*c0 \end{aligned}$$

Fig. 4E

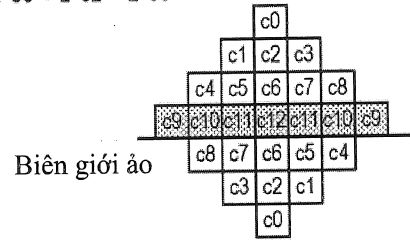


Fig. 5A

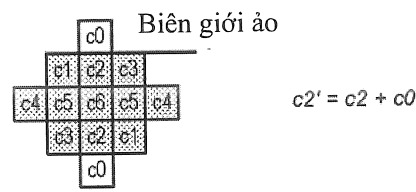


Fig. 5B

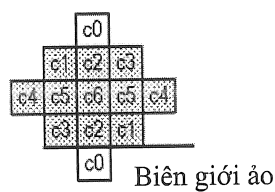


Fig. 5C

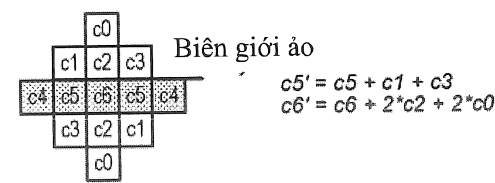


Fig. 5D

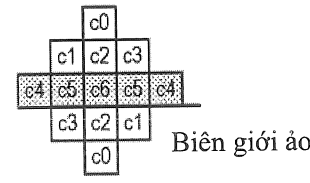


Fig. 6A

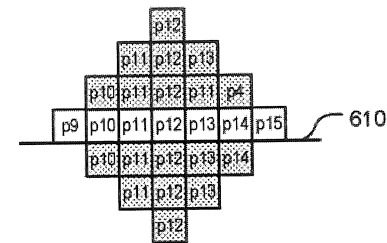
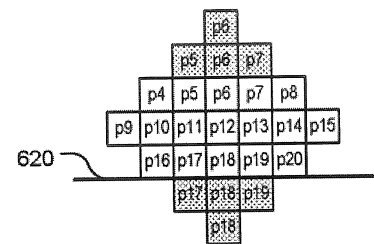
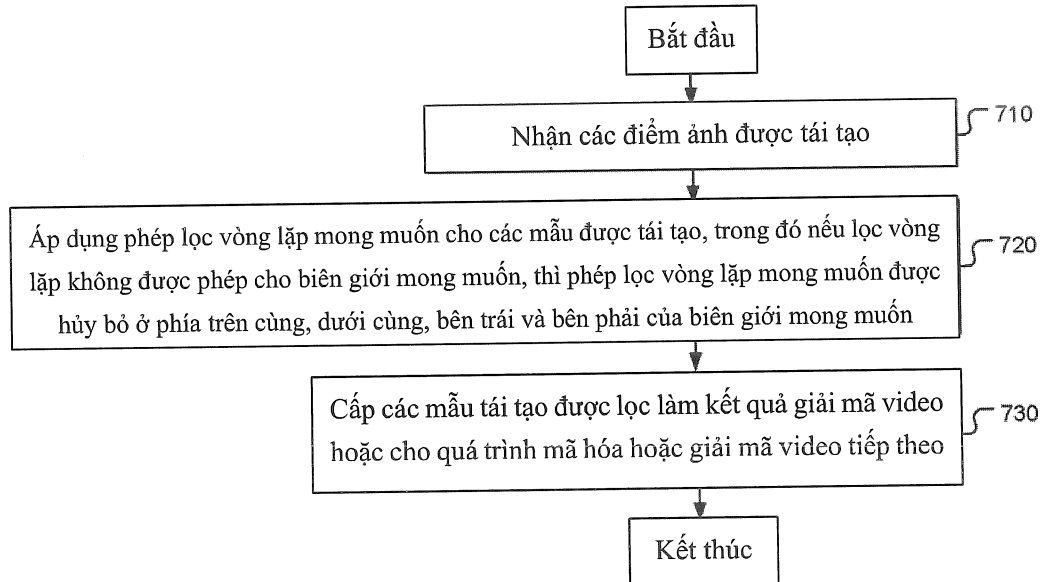


Fig. 6B



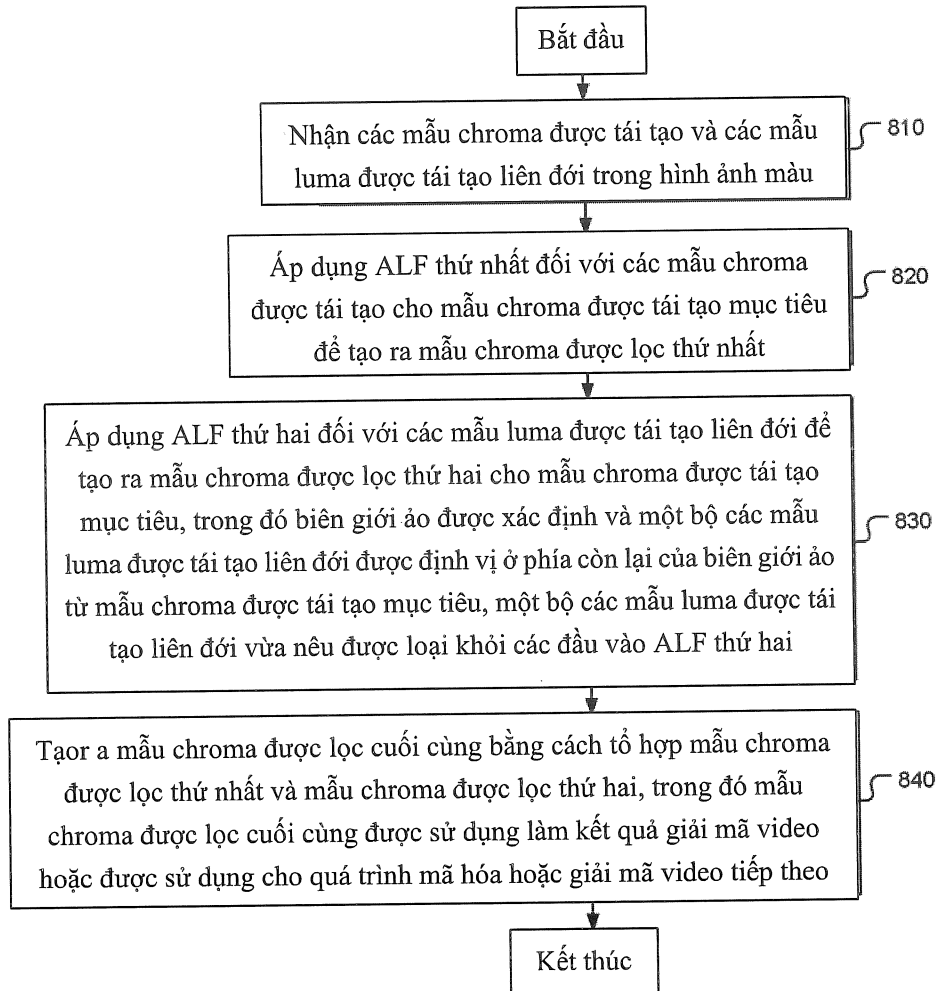
7/9

Fig. 7



8/9

Fig. 8



9/9

Fig. 9

