



- (12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ
1-0053411
(51)^{2020.01} H04N 19/573; H04N 19/119; H04N (13) B
19/176; H04N 19/105; H04N 19/132
-

- (21) 1-2021-05717 (22) 16/03/2020
(86) PCT/US2020/023027 16/03/2020 (87) WO2020/190896 24/09/2020
(30) 62/819,408 15/03/2019 US
(45) 25/11/2025 452 (43) 25/11/2021 404A
(73) BEIJING DAJIA INTERNET INFORMATION TECHNOLOGY CO., LTD. (CN)
Room 101D1-7, 1st Floor, Building 1, No. 6, Shangdi West Road, Haidian District,
Beijing, Beijing 100085, China
(72) XIU, Xiaoyu (CN); CHEN, Yi-Wen (TW); WANG, Xianglin (US).
(74) Công ty Luật TNHH T&G (TGVN)
-

- (54) PHƯƠNG PHÁP ĐIỀU KHIỂN ĐỘ RỘNG BIT CỦA LUỒNG QUANG HỌC HAI
CHIỀU, THIẾT BỊ ĐIỆN TOÁN VÀ PHƯƠNG TIỆN LƯU TRỮ ĐỌC ĐƯỢC BỞI
MÁY TÍNH KHÔNG CHUYỂN TIẾP

(21) 1-2021-05717

(57) Sáng chế đề cập đến phương pháp điều khiển độ rộng bit của luồng quang học hai chiều (Bi-Directional Optical Flow, BDOF) để tạo mã tín hiệu video. Phương pháp gồm các bước thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối video, thu nhận các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$, thu nhận các mẫu dự đoán thứ hai $I^{(1)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$, điều khiển các độ rộng bit bên trong của BDOF bằng cách dẫn xuất các độ rộng bit bên trong của các thông số trung gian, thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối video dựa trên BDOF đang được áp dụng cho khối video dựa trên các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$, và thu nhận các mẫu dự đoán đôi của khối video dựa trên các sự tinh chỉnh chuyển động. Sáng chế cũng đề cập đến thiết bị điện toán.

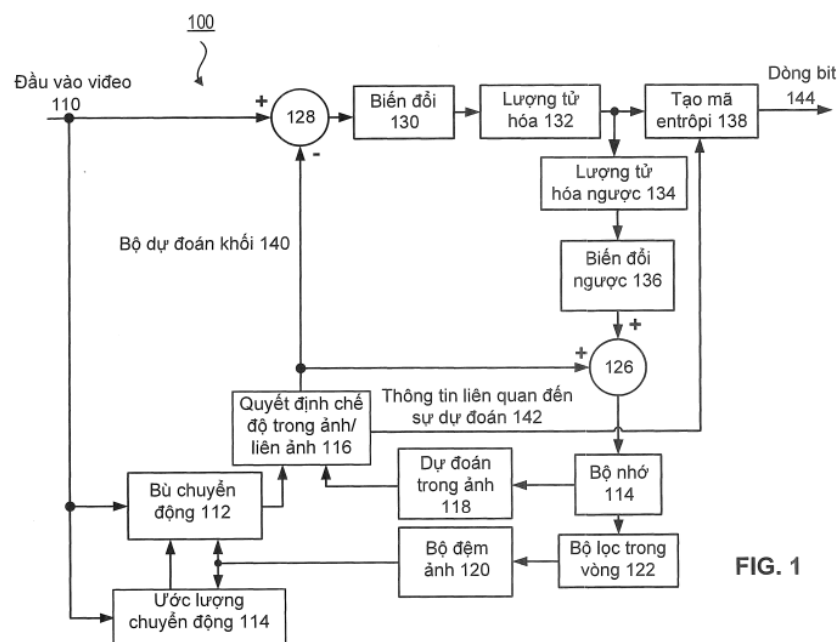


FIG. 1

Lĩnh vực kỹ thuật được đề cập

Sáng chế này liên quan đến việc tạo mã và nén video. Cụ thể hơn, sáng chế này đề cập tới các phương pháp và các thiết bị cho các phương pháp luồng quang học hai chiều (Bi-Directional Optical Flow, BDOF) cho việc tạo mã video.

Tình trạng kỹ thuật của sáng chế

Các kỹ thuật tạo mã video khác nhau có thể được sử dụng để nén dữ liệu video. Việc tạo mã video được thực hiện theo một hoặc nhiều tiêu chuẩn tạo mã video. Ví dụ, các tiêu chuẩn tạo mã video gồm việc tạo mã video vạn năng (Versatile Video Coding, VVC), tạo mã mô hình thử nghiệm thăm dò chung (Joint Exploration Test Model Coding, JEM), tạo mã video hiệu quả cao (High-Efficiency Video Coding, H.265/HEVC), tạo mã video cải tiến (Advanced Video Coding, H.264/AVC), tạo mã nhóm chuyên gia hình ảnh chuyển động (Moving Picture Experts Group Coding, MPEG), hoặc tương tự. Việc tạo mã video thường tận dụng các phương pháp dự đoán (ví dụ, sự dự đoán liên ảnh, sự dự đoán trong ảnh, hoặc dạng tương tự) mà tận dụng ưu điểm của sự dư thừa có mặt trong các hình ảnh hoặc các chuỗi video. Mục tiêu quan trọng của các kỹ thuật tạo mã video là để nén dữ liệu video thành dạng sử dụng tỉ lệ bit thấp hơn, trong khi loại bỏ hoặc giảm thiểu sự suy giảm chất lượng video.

Bản chất kỹ thuật của sáng chế

Các ví dụ của sáng chế đề xuất các phương pháp và thiết bị để điều khiển độ rộng bit cho luồng quang học hai chiều (BDOF). Theo khía cạnh thứ nhất, sáng chế đề xuất phương pháp điều khiển độ rộng bit của việc tạo mã tín hiệu video. Phương pháp có thể gồm bước thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối video. Ảnh tham chiếu thứ nhất $I^{(0)}$ có thể ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ có thể ở sau ảnh hiện tại trong thứ tự hiển thị. Phương pháp có thể còn gồm bước thu nhận các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$. Các biến i và j có thể biểu diễn tọa độ của một mẫu với ảnh hiện tại. Phương pháp có thể còn gồm bước thu nhận các mẫu dự đoán thứ hai $I^{(1)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$. Phương pháp có thể còn gồm bước điều khiển các độ rộng bit bên trong của BDOF bằng cách dẫn xuất các độ rộng bit bên trong của các thông số trung gian. Các thông số trung gian có thể gồm các thông số trung gian bao gồm các giá trị gradient ngang, các giá trị gradient dọc, và các sự chênh lệch mẫu giữa các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Phương pháp có thể còn gồm bước thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối video dựa trên BDOF đang được áp dụng cho khối video dựa trên các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Và phương pháp có thể còn gồm bước thu nhận các mẫu dự đoán đôi của khối video dựa trên các sự tinh chỉnh chuyển động.

Theo khía cạnh thứ hai, sáng chế đề xuất phương pháp điều khiển độ rộng bit của việc tạo mã tín hiệu video. Phương pháp có thể gồm bước thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối video. Ảnh tham chiếu thứ nhất $I^{(0)}$ có thể ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ có thể ở sau ảnh

hiện tại trong thứ tự hiển thị ảnh hiện tại trong thứ tự hiển thị. Phương pháp có thể gồm bước thu nhận các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$. Các biến i và j có thể biểu diễn tọa độ của một mẫu với ảnh hiện tại. Phương pháp có thể gồm bước thu nhận các mẫu dự đoán thứ hai $I^{(1)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$. Phương pháp có thể gồm bước điều khiển các độ rộng bit bên trong của BDOF bằng cách dẫn xuất các độ rộng bit bên trong của các thông số trung gian. Các thông số trung gian có thể gồm các thông số trung gian bao gồm các giá trị gradient ngang, các giá trị gradient dọc, và các sự chênh lệch mẫu giữa các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Phương pháp có thể gồm bước thu nhận sự tinh chỉnh chuyển động cho các mẫu trong khối video dựa trên BDOF và cửa sổ cục bộ đang được áp dụng cho khối video nhờ sử dụng mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Cửa sổ cục bộ bao phủ khối video và cửa sổ cục bộ có kích thước cửa sổ nhỏ hơn 6×6 điểm ảnh. Và phương pháp có thể gồm bước thu nhận các mẫu dự đoán đôi của khối video dựa trên các sự tinh chỉnh chuyển động.

Theo khía cạnh thứ ba, sáng chế đề xuất thiết bị điện toán. Thiết bị điện toán có thể gồm một hoặc nhiều bộ xử lý, bộ nhớ đọc được bởi máy tính không chuyển tiếp lưu trữ các lệnh có thể thực thi được bởi một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối video. Ảnh tham chiếu thứ nhất $I^{(0)}$ có thể ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ có thể ở sau ảnh hiện tại trong thứ tự hiển thị. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$. Các biến i và j có thể biểu diễn tọa độ của một mẫu với ảnh hiện tại. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận các mẫu dự đoán thứ hai $I^{(1)}(i, j)$ của khối video từ

khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để điều khiển các độ rộng bit bên trong của luồng quang học hai chiều (BDOF) bằng cách dẫn xuất các độ rộng bit bên trong của các thông số trung gian. Các thông số trung gian bao gồm các giá trị gradient ngang, các giá trị gradient dọc, và các sự chênh lệch mẫu giữa các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối video dựa trên BDOF đang được áp dụng cho khối video dựa trên các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Và một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận các mẫu dự đoán đôi của khối video dựa trên các sự tinh chỉnh chuyển động.

Theo khía cạnh thứ tư, sáng chế đề xuất thiết bị điện toán. Thiết bị điện toán có thể gồm một hoặc nhiều bộ xử lý, bộ nhớ đọc được bởi máy tính không chuyển tiếp lưu trữ các lệnh có thể thực thi được bởi một hoặc nhiều bộ xử lý. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối video. Ảnh tham chiếu thứ nhất $I^{(0)}$ có thể ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ có thể ở sau ảnh hiện tại trong thứ tự hiển thị. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để thu nhận các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$. Các biến i và j có thể biểu diễn tọa độ của một mẫu với ảnh hiện tại. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình thu nhận các mẫu dự đoán thứ hai $I^{(1)}(i, j)$ của khối video từ khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình để điều khiển các độ rộng bit bên trong của luồng quang học hai chiều (BDOF) bằng cách dẫn xuất các độ rộng bit bên trong của các thông số trung gian. Các thông số trung gian có thể gồm các thông số trung gian bao gồm các giá trị gradient ngang, các giá trị gradient dọc, và các sự chênh lệch mẫu giữa các mẫu dự đoán thứ nhất $I^{(0)}(i, j)$

và các mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Một hoặc nhiều bộ xử lý có thể được tạo cấu hình thu nhận sự tinh chỉnh chuyển động cho các mẫu trong khối video dựa trên BDOF và cửa sổ cục bộ đang được áp dụng cho khối video nhờ sử dụng mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Cửa sổ cục bộ có thể bao phủ khối con và cửa sổ cục bộ có thể có kích thước cửa sổ là 6×6 điểm ảnh. Và một hoặc nhiều bộ xử lý có thể được tạo cấu hình thu nhận các mẫu dự đoán đôi của khối video dựa trên các sự tinh chỉnh chuyển động.

Cần hiểu rằng cả phần mô tả tổng quát nêu trên lẫn phần mô tả chi tiết sau đây chỉ là các ví dụ và không nhằm để giới hạn sáng chế.

Mô tả vắn tắt các hình vẽ

Các hình vẽ kèm theo, vốn được tích hợp ở đây, tạo thành một phần của bản mô tả này, minh họa các khía cạnh làm ví dụ của sáng chế này và cùng với phần mô tả, có tác dụng giải thích nguyên lý của sáng chế.

Fig.1 là sơ đồ khối của bộ mã hóa, theo ví dụ của sáng chế.

Fig.2 là sơ đồ khối của bộ giải mã, theo ví dụ của sáng chế.

Fig.3A là sơ đồ minh họa các sự phân chia khối trong cấu trúc cây nhiều loại, theo ví dụ của sáng chế.

Fig.3B là sơ đồ minh họa các sự phân chia khối trong cấu trúc cây nhiều loại, theo ví dụ của sáng chế.

Fig.3C là sơ đồ minh họa các sự phân chia khối trong cấu trúc cây nhiều loại, theo ví dụ của sáng chế.

Fig.3D là sơ đồ minh họa các sự phân chia khối trong cấu trúc cây nhiều loại, theo ví dụ của sáng chế.

Fig.3E là sơ đồ minh họa các sự phân chia khối trong cấu trúc cây nhiều loại, theo ví dụ của sáng chế.

Fig.4 là sự minh họa dạng sơ đồ của mô hình luồng quang học hai chiều (BDOF), theo ví dụ của sáng chế.

Fig.5 là lưu đồ minh họa phương pháp điều khiển độ rộng bit của việc tạo mã tín hiệu video, theo ví dụ của sáng chế.

Fig.6 là lưu đồ minh họa phương pháp điều khiển độ rộng bit BDOF, theo ví dụ của sáng chế.

Fig.7 là sơ đồ minh họa môi trường điện toán được ghép nối với giao diện người dùng, theo ví dụ của sáng chế.

Mô tả chi tiết sáng chế

Bây giờ, sáng chế sẽ được mô tả chi tiết đến các phương án khác nhau, ví dụ về các phương án này được minh họa trong các hình vẽ kèm theo. Phần mô tả sau đây liên quan đến các hình vẽ kèm theo trong đó các số chỉ dẫn giống nhau trong các hình vẽ khác nhau thể hiện các thành phần giống hoặc tương tự nhau trừ khi được thể hiện khác. Các cách triển khai được nêu trong phần mô tả sau đây của các phương án không biểu diễn tất cả các cách triển khai phù hợp với sáng chế. Thay vào đó, chúng chỉ là các ví dụ về các thiết bị và các phương pháp phù hợp với các khía cạnh liên quan đến sáng chế được nêu trong các yêu cầu bảo hộ kèm theo.

Hệ thuật ngữ được sử dụng trong sáng chế chỉ nhằm mục đích mô tả các phương án cụ thể và không được dự tính để giới hạn sáng chế. Như được sử dụng trong phần mô tả của các cách thức thực hiện và các yêu cầu bảo hộ kèm theo, các dạng số ít cũng được dự tính để gồm các dạng số nhiều, trừ khi được chỉ rõ theo cách khác. Cũng cần hiểu

rằng, thuật ngữ “và/hoặc” được sử dụng ở đây để chỉ và gồm bất kỳ hoặc tất cả các tổ hợp khả thi của một hoặc nhiều mục được liệt kê liên quan.

Cần hiểu rằng, mặc dù các thuật ngữ “thứ nhất”, “thứ hai”, “thứ ba”, v.v. có thể được sử dụng ở đây để mô tả các thông tin khác nhau, nhưng không tin không bị hạn chế vào các thuật ngữ này. Các thuật ngữ này chỉ được sử dụng để phân biệt một phân loại của thông tin so với thông tin khác. Ví dụ, không tách rời khỏi phạm vi của sáng chế này, thông tin thứ nhất có thể được gọi là thông tin thứ hai, và tương tự, thông tin thứ hai cũng có thể được đặt là thông tin thứ nhất. Như được sử dụng ở đây, thuật ngữ “nếu” có thể được hiểu nghĩa là “khi” hoặc “lúc” hoặc “để đáp lại một phán quyết”, phụ thuộc vào văn cảnh.

Phiên bản thứ nhất của tiêu chuẩn HEVC được hoàn thành trong tháng 10 năm 2014, đề xuất việc tiết kiệm xấp xỉ 50% tỉ lệ bit hoặc chất lượng nhận thức tương tự khi so với tiêu chuẩn mã hóa video thế hệ trước đó H.264/MPEG AVC. Mặc dù tiêu chuẩn HEVC cung cấp các sự cải thiện tạo mã đáng kể hơn so với các tiêu chuẩn tiền nhiệm của nó, nhưng có bằng chứng là có thể đạt được hiệu quả tạo mã vượt trội hơn HEVC, với các công cụ tạo mã bổ sung. Dựa trên điều đó thì cả VCEG và MPEG đều bắt đầu công việc khám phá các kỹ thuật tạo mã mới cho việc chuẩn hóa tạo mã video trong tương lai. Một nhóm khám phá video hợp tác (Joint Video Exploration Team, JVET) được hình thành trong tháng 10 năm 2015 bởi ITU-T VCEG và ISO/IEC MPEG để bắt đầu nghiên cứu một cách cơ bản các công nghệ cải tiến có thể cho phép thực hiện việc tăng cường một cách đáng kể cho hiệu quả tạo mã. Một phần mềm tham chiếu được gọi là mô hình khám phá hợp tác (Joint Exploration Model, JEM) đã được duy trì bởi JVET bằng cách tích hợp nhiều công cụ tạo mã bổ sung trên đỉnh của mô hình kiểm tra HEVC (HEVC test Model, HM).

Vào tháng 10 năm 2017, lời kêu gọi hợp tác cho các đề xuất (Call for Proposal, CFP) về việc nén video với khả năng vượt qua HEVC đã được phát hành bởi ITU-T và ISO/IEC [9]. Vào tháng 4 năm 2018, 23 phản hồi CFP được nhận và được đánh giá tạo phiên họp JVET lần thứ 10, đã chứng minh rằng hiệu quả nén đã vượt quá HEVC khoảng 40%. Dựa trên các kết quả đánh giá này, JVET đã triển khai dự án mới để phát triển tiêu chuẩn tạo mã video thế hệ mới có tên là tạo mã video vạn năng (Versatile Video Coding, VVC). Trong cùng tháng đó, một cơ sở mã phần mềm tham chiếu, được gọi là mô hình kiểm tra VVC (VVC Test Model, VTM), được thiết lập để chứng minh việc áp dụng tham chiếu của tiêu chuẩn VVC.

Giống như HEVC, VVC được xây dựng trên khung tạo mã video lai dựa trên khối. Fig.1 đưa ra sơ đồ khối của hệ thống mã hóa video lai dựa trên khối chung. Tín hiệu video đầu vào được xử lý theo từng khối (được gọi là các đơn vị tạo mã (Coding Unit, CU). Cụ thể là, Fig.1 thể hiện bộ mã hóa điển hình 100 theo sáng chế. Bộ mã hóa 100 có đầu vào video 110, sự bù chuyển động 112, sự ước lượng chuyển động 114, sự quyết định chế độ trong ảnh/liên ảnh 116, bộ dự đoán khối 140, bộ cộng 128, sự biến đổi 130, sự lượng tử hóa 132, thông tin liên quan đến sự dự đoán 142, sự dự đoán trong ảnh 118, bộ đệm ảnh 120, sự lượng tử hóa ngược 134, sự biến đổi ngược 136, bộ cộng 126, bộ nhớ 124, bộ lọc trong vòng 122, tạo mã entropi 138, và dòng bit 144.

Trong VTM-1.0, CU có thể lên tới 128x128 điểm ảnh. Tuy nhiên, khác với HEVC vốn phân chia các khối chỉ dựa trên các cây tứ phân, thì trong VVC, một giá trị cây tạo mã (Coding Tree Unit, CTU) được chia tách thành các CU để thích ứng để biến đổi các đặc tính cục bộ dựa trên cây tứ phân/nhị phân/tam phân. Theo cách bổ sung, khái niệm của loại đơn vị nhiều sự phân chia trong HEVC được loại bỏ, tức là, việc tách ra của CU, đơn vị dự đoán (Prediction Unit, PU) và đơn vị biến đổi (Transform Unit, TU)

không tồn tại trong VVC nữa; thay vào đó, mỗi CU luôn được sử dụng như là đơn vị cơ bản cho cả sự dự đoán và biến đổi mà không có thêm các sự phân chia khác nữa. Trong cấu trúc cây nhiều loại, một CTU được phân chia đầu tiên bởi cấu trúc cây tứ phân. Sau đó, mỗi nút lá cây tứ phân có thể còn được phân chia bởi cấu trúc cây nhị phân hoặc tam phân.

Như được thể hiện trên Fig.3A, Fig.3B, Fig.3C, Fig.3D, và Fig.3E (được mô tả ở dưới đây), thì có năm loại chia tách: phân chia tứ phân, phân chia nhị phân dọc, phân chia nhị phân ngang, phân chia tam phân ngang, và phân chia tam phân dọc.

Fig.3A thể hiện sơ đồ minh họa sự phân chia tứ phân khối trong cấu trúc cây nhiều loại, theo sáng chế.

Fig.3B thể hiện sơ đồ minh họa sự phân chia nhị phân dọc khối trong cấu trúc cây nhiều loại, theo sáng chế.

Fig.3C thể hiện sơ đồ minh họa sự phân chia nhị phân ngang khối trong cấu trúc cây nhiều loại, theo sáng chế.

Fig.3D thể hiện sơ đồ minh họa sự phân chia tam phân dọc khối trong cấu trúc cây nhiều loại, theo sáng chế.

Fig.3E thể hiện sơ đồ minh họa sự phân chia tam phân ngang khối trong cấu trúc cây nhiều loại, theo sáng chế.

Trên Fig.1, sự dự đoán theo không gian và/hoặc sự dự đoán theo thời gian có thể được thực hiện. Sự dự đoán theo không gian (hoặc “sự dự đoán trong ảnh”) sử dụng các điểm ảnh từ các mẫu của các khối lân cận hiện đã được mã hóa (vốn được gọi là các mẫu tham chiếu) trong cùng một hình ảnh/phiên video để dự đoán khối video hiện tại. Sự dự đoán theo không gian làm giảm tính dư thừa về mặt không gian cố hữu trong tín

hiệu video. Sự dự đoán theo thời gian (còn được gọi là “sự dự đoán liên ảnh” hoặc “sự dự đoán được bù chuyển động”) sử dụng các điểm ảnh được tái tạo từ các ảnh video đã được tạo mã để dự đoán khối video hiện tại. Sự dự đoán theo thời gian làm giảm sự dư thừa về mặt thời gian cố hữu trong tín hiệu video. Tín hiệu dự đoán theo thời gian cho CU đã cho thường được phát tín hiệu bởi một hoặc nhiều vectơ chuyển động (Motion Vector, MV) vốn chỉ ra số lượng và hướng chuyển động giữa CU hiện tại và tham chiếu theo thời gian của nó.

Ngoài ra, nếu nhiều ảnh tham chiếu được hỗ trợ, thì một chỉ số ảnh tham chiếu sẽ được gửi theo cách bổ sung, vốn được sử dụng để nhận dạng nơi mà ảnh tham chiếu trong ảnh tham chiếu lưu trữ tín hiệu dự đoán theo thời gian, đi ra từ đó. Sau sự dự đoán theo không gian và/hoặc theo thời gian, thì khối quyết định chế độ trong bộ mã hóa chọn chế độ dự đoán tốt nhất, ví dụ dựa trên phương pháp tối ưu hóa nhiễu loạn tỉ lệ. Sau đó, khối dự đoán được trừ khỏi khối video hiện tại và phần dư dự đoán được giải tương liên nhờ sử dụng sự biến đổi và được lượng tử hóa.

Các hệ số dư được lượng tử hóa sẽ được lượng tử hóa ngược và biến đổi ngược để tạo thành phần dư được tái tạo, mà sau đó được cộng vào khối dự đoán để tạo thành tín hiệu được tái tạo của CU. Hơn nữa, việc lọc trong vòng, như bộ lọc khử khối, dịch vị tương thích mẫu (Sample Adaptive Offset, SAO), và/hoặc bộ lọc trong vòng thích ứng (Adaptive in-Loop Filter, ALF) có thể được áp dụng vào CU được tái tạo trước khi nó được nhập trong bộ phận lưu trữ ảnh tham chiếu và được sử dụng để tạo mã các khối video tương lai. Để tạo thành dòng bit video đầu ra, thì chế độ tạo mã (liên ảnh hoặc trong ảnh), thông tin chế độ dự đoán, thông tin chuyển động, và các hệ số dư được lượng tử hóa tất cả được gửi đến đơn vị tạo mã entropi để còn được nén và đóng gói để tạo thành dòng bit.

Fig.2 đưa ra sơ đồ khối chung của bộ giải mã video dựa trên khối. Cụ thể làm Fig.2 thể hiện sơ đồ khối của bộ giải mã điển hình 200, theo sáng chế. Bộ giải mã 200 có dòng bit 210, việc giải mã entropi 212, sự lượng tử hóa ngược 214, sự biến đổi ngược 216, bộ cộng 218, sự lựa chọn chế độ trong ảnh/liên ảnh 220, sự dự đoán trong ảnh 222, bộ nhớ 230, bộ lọc trong vòng 228, sự bù chuyển động 224, bộ đệm ảnh 226, thông tin liên quan đến sự dự đoán 234, và đầu ra video 232.

Trên Fig.2, dòng bit video được giải mã entropi đầu tiên tại đơn vị giải mã entropi. Chế độ tạo mã và thông tin dự đoán được gửi đến hoặc là đơn vị dự đoán theo không gian (nếu được tạo mã trong ảnh) hoặc đến đơn vị dự đoán theo thời gian (nếu được tạo mã liên ảnh) để tạo thành khối dự đoán. Các hệ số biến đổi dư được gửi đến đơn vị lượng tử hóa ngược và đơn vị biến đổi ngược để tái tạo khối dư. Sau đó, khối dự đoán và khối dư được cộng lại với nhau. Khối được tái tạo có thể còn trải qua việc lọc trong vòng trước khi nó được lưu trữ trong bộ phận lưu trữ ảnh tham chiếu. Sau đó, video được tái tạo trong bộ phận lưu trữ ảnh tham chiếu được gửi ra để điều vận thiết bị hiển thị, cũng như được sử dụng để dự đoán các khối video tương lai.

Luồng quang học hai chiều (Bi-Directional Optical Flow)

Sự dự đoán đôi truyền thống trong việc tạo mã video là tổ hợp đơn giản của hai khối dự đoán theo thời gian được thu nhận từ các ảnh tham chiếu đã được tái tạo. Tuy nhiên, do sự giới hạn của việc bù chuyển động dựa trên khối, nên có thể có chuyển động nhỏ còn lại có thể quan sát được giữa các mẫu của hai khối dự đoán, do đó làm giảm hiệu quả của sự dự đoán được bù chuyển động. Luồng quang học hai chiều (BDOF) được áp dụng trong VVC để làm giảm các tác động của chuyển động này cho mọi mẫu ở trong một khối.

Fig.4 thể hiện sự minh họa của mô hình luồng quang học hai chiều (BDOF), theo sáng chế. BDOF là sự tinh chỉnh chuyển động đảo mẫu được thực hiện trên đỉnh của các dự đoán được bù chuyển động dựa trên khối khi sự dự đoán đôi được sử dụng. Sự tinh chỉnh chuyển động (v_x, v_y) của mỗi khối con 4×4 được tính toán bằng cách tối thiểu hóa sự chênh lệch giữa các mẫu dự đoán L0 và L1 sau khi BDOF được áp dụng nằm trong một cửa sổ 6×6 Ω quanh khối con. Cụ thể là, giá trị của (v_x, v_y) được dẫn xuất là:

$$\begin{aligned} v_x &= S_1 > 0? \text{clip3} \left(-th_{BDOF}, th_{BDOF}, -((S_3 \cdot 2^3) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0 \\ v_y &= S_5 > 0? \text{clip3} \left(-th_{BDOF}, th_{BDOF}, - \left((S_6 \cdot 2^3 \right. \right. \\ &\quad \left. \left. - ((v_x S_{2,m}) \ll n_{S_2} + v_x S_{2,s}) / 2 \right) \gg \lfloor \log_2 S_5 \rfloor \right) : \end{aligned} \quad (1)$$

mà tại đó $\lfloor \cdot \rfloor$ là hàm làm tròn xuống (floor); clip3 (min, max, x) là hàm cắt xén giá trị định trước x nằm trong phạm vi [min, max]; ký hiệu $\gg$ biểu diễn hoạt động dịch chuyển sang phải của đảo bit; ký hiệu $\ll$ biểu diễn hoạt động dịch chuyển sang trái của đảo bit; th_{BDOF} là ngưỡng sự tinh chỉnh chuyển động để ngăn chặn các lỗi được lan truyền do chuyển động cục bộ không đều đặn, vốn bằng với 2^{13-BD} , trong đó BD là độ sâu bit của video đầu vào. Trong (1), $S_{2,m} = S_2 \gg n_{S_2}$, $S_{2,s} = S_2 \& (2^{n_{S_2}} - 1)$.

Các giá trị của S_1 , S_2 , S_3 , S_5 , và S_6 được tính toán là:

$$\begin{aligned} S_1 &= \sum_{(i,j) \in \Omega} \psi_x(i,j) \cdot \psi_x(i,j) \\ S_2 &= \sum_{(i,j) \in \Omega} \psi_x(i,j) \cdot \psi_y(i,j) \\ S_3 &= \sum_{(i,j) \in \Omega} \theta(i,j) \cdot \psi_x(i,j) \\ S_5 &= \sum_{(i,j) \in \Omega} \psi_y(i,j) \cdot \psi_y(i,j) \end{aligned} \quad (2)$$

$$S_6 = \sum_{(i,j) \in \Omega} \theta(i,j) \cdot \psi_y(i,j)$$

mà tại đó

$$\begin{aligned} \psi_x(i,j) &= \left(\frac{\partial I^{(1)}}{\partial x}(i,j) + \frac{\partial I^{(0)}}{\partial x}(i,j) \right) \gg 3 \\ \psi_y(i,j) &= \left(\frac{\partial I^{(1)}}{\partial y}(i,j) + \frac{\partial I^{(0)}}{\partial y}(i,j) \right) \gg 3 \\ \theta(i,j) &= (I^{(1)}(i,j) \gg 6) - (I^{(0)}(i,j) \gg 6) \end{aligned} \quad (3)$$

mà tại đó $I^{(k)}(i,j)$ là giá trị mẫu tại tọa độ (i,j) của tín hiệu dự đoán trong danh sách k , $k = 0,1$, vốn được tạo ra tại sự chính xác cao trung gian (tức là, 16 bit); $\frac{\partial I^{(k)}}{\partial x}(i,j)$ và $\frac{\partial I^{(k)}}{\partial y}(i,j)$ là các gradient ngang và dọc của mẫu mà được thu nhận bằng cách tính toán một cách trực tiếp sự chênh lệch giữa hai mẫu lân cận của nó, tức là:

$$\begin{aligned} \frac{\partial I^{(k)}}{\partial x}(i,j) &= (I^{(k)}(i+1,j) - I^{(k)}(i-1,j)) \gg 4 \\ \frac{\partial I^{(k)}}{\partial y}(i,j) &= (I^{(k)}(i,j+1) - I^{(k)}(i,j-1)) \gg 4 \end{aligned} \quad (4)$$

Dựa trên sự tinh chỉnh chuyển động được dẫn xuất trong (1), thì các mẫu dự đoán hai chiều cuối cùng của CU được tính toán bằng cách ngoại suy các mẫu dự đoán L0/L1 dọc theo quỹ đạo chuyển động dựa trên mô hình luồng quang, như được chỉ ra bởi:

$$pred_{BDOF}(x,y) = (I^{(0)}(x,y) + I^{(1)}(x,y) + b + o_{offset}) \gg shift$$

$$\begin{aligned} b &= rnd \left(\left(v_x \left(\frac{\partial I^{(1)}(x,y)}{\partial x} - \frac{\partial I^{(0)}(x,y)}{\partial x} \right) \right) / 2 \right) \\ &+ rnd \left(\left(v_y \left(\frac{\partial I^{(1)}(x,y)}{\partial y} - \frac{\partial I^{(0)}(x,y)}{\partial y} \right) \right) / 2 \right) \end{aligned} \quad (5)$$

mà tại đó $shift$ và $offset$ là giá trị dịch chuyển sang phải và giá trị dịch vị mà được áp dụng để tổ hợp các tín hiệu dự đoán L0 và L1 cho sự dự đoán đôi, vốn bằng với $15 - BD$ và $1 \ll (14 - BD) + 2 \cdot (1 \ll 13)$, một cách tương ứng. Bảng 1 minh họa các độ rộng bit cụ thể của các thông số trung gian tham gia vào trong quy trình BDOF. Các độ rộng bit, ví dụ, là số lượng của các bit cần thiết để biểu diễn giá trị. Như được thể hiện trên bảng, độ rộng bit bên trong của toàn bộ quy trình BDOF không được vượt quá 32 bit. Theo cách bổ sung, phép nhân với đầu vào kém khả thi nhất xảy ra tại tích của $v_x S_{2,m}$ trong (1) với các đầu vào 15-bit và 4-bit. Do đó, số nhân 15-bit là đủ cho BDOF.

Bảng 1 Các độ rộng bit của các thông số trung gian của BDOF trong VVC

Hoạt động	Thông số	Độ rộng bit
Sự dự đoán L0/L1	$I^{(0)}(i, j), I^{(1)}(i, j)$	16
Dẫn xuất gradient	$\frac{\partial I^{(0)}}{\partial x}(i, j), \quad \frac{\partial I^{(0)}}{\partial y}(i, j)$	13
	$\frac{\partial I^{(1)}}{\partial x}(i, j), \quad \frac{\partial I^{(1)}}{\partial y}(i, j)$	13
Tính toán thông số tương liên	$\theta(i, j) = (I^{(1)}(i, j) \gg 6) - (I^{(0)}(i, j) \gg 6)$	11
	$\psi_x(i, j) = \left(\frac{\partial I^{(1)}}{\partial x}(i, j) + \frac{\partial I^{(0)}}{\partial x}(i, j) \right) \gg 3$	11
	$\psi_y(i, j) = \left(\frac{\partial I^{(1)}}{\partial y}(i, j) + \frac{\partial I^{(0)}}{\partial y}(i, j) \right) \gg 3$	11
	$\psi_x(i, j) \cdot \psi_x(i, j)$	21
	$\psi_x(i, j) \cdot \psi_y(i, j)$	21
	$\psi_y(i, j) \cdot \psi_y(i, j)$	21

	$\theta(i, j) \cdot \psi_x(i, j)$	21
	$\psi_y(i, j) \cdot \psi_y(i, j)$	21
	$\theta(i, j) \cdot \psi_y(i, j)$	21
Tổng	$S_1 = \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \psi_x(i, j)$	27
	$S_2 = \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \psi_y(i, j)$	27
	$S_3 = \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \psi_x(i, j)$	27
	$S_5 = \sum_{(i,j) \in \Omega} \psi_y(i, j) \cdot \psi_y(i, j)$	27
	$S_6 = \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \psi_y(i, j)$	27
Dẫn xuất sự tính chỉnh chuyển động	$v_x = S_1 > 0? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, -((S_3 \cdot 2^3) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0$	4
	$S_{2,m} = S_2 \gg 12$	15
	$S_{2,s} = S_2 \& (2^{12} - 1)$	12
	$((v_x S_{2,m}) \ll 12 + v_x S_{2,s}) / 2$	30
	$S_6 \cdot 2^3 - ((v_x S_{2,m}) \ll 12 + v_x S_{2,s}) / 2$	31
	$v_y = S_5 > 0? \text{clip3} \left(-th'_{BIO}, th'_{BIO}, - \left((S_6 \cdot 2^3 - ((v_x S_{2,m}) \ll 12 + v_x S_{2,s}) / 2) \gg \lfloor \log_2 S_5 \rfloor \right) \right) : 0$	4
Tạo ra sự dự đoán	$v_x \cdot \left(\frac{\partial I^{(1)}}{\partial x}(i, j) - \frac{\partial I^{(0)}}{\partial x}(i, j) \right)$	17

đôi cuối cùng	$v_y \cdot \left(\frac{\partial I^{(1)}}{\partial y}(i, j) - \frac{\partial I^{(0)}}{\partial y}(i, j) \right)$	17
------------------	--	----

Hiệu quả của sự dự đoán dự đoán đôi

Mặc dù BDOF có thể tăng cường hiệu quả của sự dự đoán dự đoán đôi, nhưng thiết kế của nó vẫn có thể còn được cải thiện. Cụ thể là, việc điều khiển các độ rộng bit của các thông số trung gian in thiết kế BDOF sẵn có trong VVC được nhận diện trong sáng chế này.

Như được thể hiện trong Bảng 1, thông số $\theta(i, j)$ (tức là, sự chênh lệch giữa các mẫu dự đoán L0 và L1), và thông số $\psi_x(i, j)$ và $\psi_y(i, j)$ (tức là, tổng của các giá trị gradient L0 và L1 ngang/dọc) được biểu diễn trong cùng độ rộng bit của 11 bit. Mặc dù phương pháp này có thể tạo thuận tiện cho việc điều khiển tổng thể của độ rộng bit bên trong cho BDOF, nhưng phương pháp này có thể không tối ưu về độ chính xác của các sự tinh chỉnh chuyển động được dẫn xuất. Điều đó là bởi vì như được thể hiện trong (4), thì các giá trị gradient được tính toán là sự chênh lệch giữa các mẫu dự đoán lân cận; do tính chất thông cao của quá trình này, nên các gradient được dẫn xuất kém tin cậy hơn khi có nhiễu, ví dụ, nhiễu thu được tổng video gốc và nhiễu tạo mã mà được tạo ra trong quy trình tạo mã. Có nghĩa là không phải lúc nào nó cũng có lợi để biểu diễn các giá trị gradient trong độ rộng bit cao.

Như được thể hiện trong Bảng 1, việc sử dụng độ rộng bit tối đa của toàn bộ quy trình BDOF diễn ra với sự tính toán của sự tinh chỉnh chuyển động dọc v_y mà tại đó S_6 (27 bit) được dịch chuyển sang trái trước tiên là 3-bit, sau đó bị trừ đi $((v_x S_{2,m}) \ll 12 + v_x S_{2,s})/2$ (30 bit). Do đó, độ rộng bit tối đa của thiết kế hiện tại là

bằng 31 bit. Trong cách triển khai phần cứng thực tế, quy trình tạo mã với độ rộng bit bên trong tối đa lớn hơn 16 bit thường được triển khai bởi cách triển khai 32 bit. Do đó, thiết kế sẵn có không tận dụng được hoàn toàn phạm vi động hợp lệ của cách triển khai 32 bit. Điều này có thể dẫn đến tổn thất độ chính xác không cần thiết của các sự tinh chỉnh chuyển động được dẫn xuất bởi BDOF.

Cải thiện hiệu quả của sự dự đoán dự đoán đôi nhờ sử dụng BDOF

Trong sáng chế này, một phương pháp điều khiển độ rộng bit được cải thiện sẽ được đề xuất để giải quyết hai vấn đề của phương pháp điều khiển độ rộng bit, như được chỉ ra trong phần “Hiệu quả của sự dự đoán dự đoán đôi” đối với thiết kế BDOF sẵn có.

Fig.5 thể hiện phương pháp điều khiển độ rộng bit của việc tạo mã tín hiệu video theo sáng chế.

Trong bước 510, thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối con. Ảnh tham chiếu thứ nhất $I^{(0)}$ ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ ở sau ảnh hiện tại trong thứ tự hiển thị. Ví dụ, các ảnh tham chiếu có thể là các ảnh video lân cận với ảnh hiện tại đang được mã hóa.

Trong bước 512, thu nhận mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ từ khối con đến khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$. Ví dụ, mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ có thể là mẫu dự đoán sử dụng vectơ chuyển động trong danh sách L0 của ảnh tham chiếu trước đó trong thứ tự hiển thị.

Trong bước 514, thu nhận mẫu dự đoán thứ hai $I^{(1)}(i, j)$ từ khối con đến khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$. Ví dụ, mẫu dự đoán thứ hai $I^{(1)}(i, j)$ có thể là mẫu dự đoán sử dụng vectơ chuyển động trong danh sách L1 của ảnh tham chiếu tiếp theo trong thứ tự hiển thị.

Trong bước 516, điều khiển các độ rộng bit bên trong của luồng quang học hai chiều (BDOF) bằng cách thay đổi các độ rộng bit bên trong của các thông số trung gian. Các thông số trung gian gồm các giá trị gradient ngang và các giá trị gradient dọc được dẫn xuất dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Các giá trị gradient, ví dụ, là sự chênh lệch giữa hai mẫu dự đoán lân cận của nó (mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$)).

Trong bước 518, thu nhận các giá trị sự tinh chỉnh chuyển động dựa trên BDOF đang được áp dụng cho khối con sử dụng mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$.

Trong bước 520, thu nhận các mẫu dự đoán đôi của khối con dựa trên các giá trị sự tinh chỉnh chuyển động.

Trong phương án của sáng chế, để khắc phục các tác động tiêu cực của của các lỗi ước lượng gradient, thì việc dịch chuyển sang phải bổ sung n_{grad} được đưa ra trong phương pháp được đề xuất khi tính toán các giá trị gradient $\frac{\partial I^{(k)}}{\partial x}(i, j)$ và $\frac{\partial I^{(k)}}{\partial y}(i, j)$ trong (4), tức là, để làm giảm độ rộng bit bên trong của các giá trị gradient. Cụ thể là, các gradient ngang và dọc tại mỗi vị trí mẫu đều được tính toán là:

$$\begin{aligned}\frac{\partial I^{(k)}}{\partial x}(i, j) &= \left(I^{(k)}(i + 1, j) - I^{(k)}(i - 1, j) \right) \gg (4 + n_{grad}) \\ \frac{\partial I^{(k)}}{\partial y}(i, j) &= \left(I^{(k)}(i, j + 1) - I^{(k)}(i, j - 1) \right) \gg (4 + n_{grad})\end{aligned}\tag{6}$$

Thêm nữa, việc dịch chuyển bit bổ sung n_{adj} được đưa vào sự tính toán của các biến n_{adj} , $\psi_y(i, j)$ và $\theta(i, j)$ để điều khiển toàn bộ quy trình BDOF sao cho quy trình BDOF được hoạt động tại các độ rộng bit bên trong thích hợp, như được miêu tả là:

$$\begin{aligned}
\psi_x(i, j) &= \left(\frac{\partial I^{(1)}}{\partial x}(i, j) + \frac{\partial I^{(0)}}{\partial x}(i, j) \right) \gg (3 - n_{adj}) \\
\psi_y(i, j) &= \left(\frac{\partial I^{(1)}}{\partial y}(i, j) + \frac{\partial I^{(0)}}{\partial y}(i, j) \right) \gg (3 - n_{adj}) \\
\theta(i, j) &= \left(I^{(1)}(i, j) \gg (6 - n_{adj}) \right) - \left(I^{(0)}(i, j) \gg (6 - n_{adj}) \right)
\end{aligned} \tag{7}$$

Trong bảng 2, do sự sửa đổi thành số lượng của các bit được dịch chuyển sang phải mà được áp dụng trong (6) và (7), nên các phạm vi động của các thông số $\psi_x(i, j)$, $\psi_y(i, j)$ và $\theta(i, j)$ sẽ khác nhau, so với thiết kế BDOF sẵn có trong bảng 1 mà tại đó ba thông số được biểu diễn trong cùng phạm vi động (tức là, 21 bit). Sự thay đổi như vậy có thể làm tăng các độ rộng bit của các thông số bên trong S_1 , S_2 , S_3 , S_5 và S_6 , vốn có thể làm tăng một cách tiềm năng độ rộng bit tối đa của quy trình BDOF bên trong để vượt quá 32 bit. Vì thế, để đảm bảo cách triển khai 32 bit, thì hai phép tính cắt xén bổ sung được đưa vào trong việc tính toán các giá trị của S_2 và S_6 . Cụ thể là, trong phương pháp được đề xuất, các giá trị của hai thông số được tính toán là:

$$\begin{aligned}
S_2 &= Clip3 \left(-2^{B_2}, 2^{B_2} - 1, \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \psi_y(i, j) \right) \\
S_6 &= Clip3 \left(-2^{B_6}, 2^{B_6} - 1, \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \psi_y(i, j) \right)
\end{aligned} \tag{8}$$

mà tại đó B_2 và B_6 là các thông số để điều khiển các phạm vi động đầu ra của S_2 và S_6 , một cách tương ứng. Ví dụ, độ sâu bit biểu diễn số lượng của các bit được sử dụng để định rõ mỗi điểm ảnh. Cần lưu ý rằng khác với sự tính toán gradient, các phép tính cắt xén trong (8) chỉ được áp dụng lúc tính toán sự tinh chỉnh chuyển động của mỗi

khối con 4×4 ở trong một CU BDOF, tức là, đang được gọi ra dựa trên đơn vị 4×4 . Do đó, độ phức tạp tương ứng sẽ tăng do các phép tính cắt xén được đưa vào trong phương pháp được đề xuất sẽ là rất nhỏ.

Trong thực tế, các giá trị khác nhau của n_{grad} , n_{adj} , B_2 , và B_6 có thể được áp dụng để giành được các sự cân nhắc khác nhau giữa độ rộng bit trung gian và độ chính xác của các sự dẫn xuất BDOF bên trong. Theo một phương án, sáng chế đề xuất việc cài đặt n_{grad} và n_{adj} thành 2, B_2 thành 25, và B_6 thành 27. Theo một phương án, sáng chế đề xuất việc cài đặt B_2 thành 26 và B_6 thành 28. Theo một phương án, sáng chế đề xuất việc cài đặt n_{grad} và n_{adj} thành 6, 1, hoặc 4.

Bảng 2 minh họa độ rộng bit tương ứng của mỗi thông số trung gian khi phương pháp điều khiển độ rộng bit được đề xuất sẽ được áp dụng cho BDOF. Trong bảng 2, màu xám làm nổi bật các thay đổi mà được áp dụng trong phương pháp điều khiển độ rộng bit được đề xuất so với thiết kế BDOF sẵn có trong VVC (như được thể hiện trong bảng 1). Như có thể thấy trên bảng 2, với phương pháp điều khiển độ rộng bit được đề xuất thì độ rộng bit bên trong của toàn bộ quy trình BDOF không vượt quá 32 bit. Theo cách bổ sung, nhờ thiết kế được đề xuất, độ rộng bit tối đa chỉ là 32 bit, vốn có thể tận dụng một cách đầy đủ phạm vi động khả dụng của cách triển khai phần cứng 32 bit. Mặt khác, như được thể hiện trong bảng, phép nhân với đầu vào kém khả thi nhất xảy ra tại tích của đầu vào $v_x S_{2,m}$ mà tại đó $S_{2,m}$ là 14 bit và đầu vào v_x là 6 bit. Do đó, giống như thiết kế BDOF sẵn có, thì một số nhân 16 bit cũng đủ lớn khi phương pháp được đề xuất sẽ được áp dụng.

Bảng 2 Các độ rộng bit của các thông số trung gian của phương pháp được đề xuất

Hoạt động	Thông số	Độ rộng bit
-----------	----------	-------------

Sự dự đoán L0/L1	$I^{(0)}(i, j), I^{(1)}(i, j)$	16
Dẫn xuất gradient	$\frac{\partial I^{(0)}}{\partial x}(i, j), \frac{\partial I^{(0)}}{\partial y}(i, j)$	11
	$\frac{\partial I^{(1)}}{\partial x}(i, j), \frac{\partial I^{(1)}}{\partial y}(i, j)$	11
Tính toán thông số tương liên	$\theta(i, j) = (I^{(1)}(i, j) \gg 4) - (I^{(0)}(i, j) \gg 4)$	13
	$\psi_x(i, j) = \left(\frac{\partial I^{(1)}}{\partial x}(i, j) + \frac{\partial I^{(0)}}{\partial x}(i, j) \right) \gg 1$	11
	$\psi_y(i, j) = \left(\frac{\partial I^{(1)}}{\partial y}(i, j) + \frac{\partial I^{(0)}}{\partial y}(i, j) \right) \gg 1$	11
	$\psi_x(i, j) \cdot \psi_x(i, j)$	21
	$\psi_x(i, j) \cdot \psi_y(i, j)$	21
	$\theta(i, j) \cdot \psi_x(i, j)$	23
	$\psi_y(i, j) \cdot \psi_y(i, j)$	21
	$\theta(i, j) \cdot \psi_y(i, j)$	23
Tổng	$S_1 = \sum_{(i, j) \in \Omega} \psi_x(i, j) \cdot \psi_x(i, j)$	27
	$S_2 = \text{clip3} \left(-2^{25}, 2^{25} - 1, \sum_{(i, j) \in \Omega} \psi_x(i, j) \cdot \psi_y(i, j) \right)$	26
	$S_3 = \sum_{(i, j) \in \Omega} \theta(i, j) \cdot \psi_x(i, j)$	29
	$S_5 = \sum_{(i, j) \in \Omega} \psi_y(i, j) \cdot \psi_y(i, j)$	27
	$S_6 = \text{clip3} \left(-2^{27}, 2^{27} - 1, \sum_{(i, j) \in \Omega} \theta(i, j) \cdot \psi_y(i, j) \right)$	28

Dẫn xuất sự tinh chỉnh chuyển động	$v_x = S_1 > 0? clip3 \left(-th'_{BDOF}, th'_{BDOF}, -((S_3 \cdot 2^3) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0$	6
	$S_{2,m} = S_2 \gg 12$	14
	$S_{2,s} = S_2 \& (2^{12} - 1)$	12
	$\left((v_x S_{2,m}) \ll 12 + v_x S_{2,s} \right) / 2$	31
	$S_6 \cdot 2^3 - \left((v_x S_{2,m}) \ll 12 + v_x S_{2,s} \right) / 2$	32
	$v_y = S_5 > 0? clip3 \left(-th'_{BDOF}, th'_{BDOF}, - \left(\left(S_6 \cdot 2^3 - \left((v_x S_{2,m}) \ll 12 + v_x S_{2,s} \right) / 2 \right) \gg \lfloor \log_2 S_5 \rfloor \right) \right) : 0$	6
Tạo ra sự dự đoán đôi cuối cùng	$v_x \cdot \left(\frac{\partial I^{(1)}}{\partial x}(i, j) - \frac{\partial I^{(0)}}{\partial x}(i, j) \right)$	17
	$v_y \cdot \left(\frac{\partial I^{(1)}}{\partial y}(i, j) - \frac{\partial I^{(0)}}{\partial y}(i, j) \right)$	17

Trong phương pháp ở trên, các phép tính cắt xén, như được thể hiện trong phương trình (8), được bổ sung để tránh việc tràn thông số trung gian khi dẫn xuất v_x và v_y . Tuy nhiên, các sự cắt xén này chỉ cần thiết khi các thông số tương liên được tích lũy trong cửa sổ cục bộ lớn. Khi một cửa sổ nhỏ được áp dụng, thì sự tràn có thể không khả thi. Do đó, trong một phương án khác của sáng chế, phương pháp điều khiển độ sâu bit sau đây sẽ được đề xuất cho phương pháp BDOF mà không có sự cắt xén, như được mô tả như sau.

Đầu tiên, các giá trị gradient $\frac{\partial I^{(k)}}{\partial x}(i, j)$ và $\frac{\partial I^{(k)}}{\partial y}(i, j)$ trong (4) tại mỗi vị trí mẫu được tính toán là:

$$\begin{aligned}\frac{\partial I^{(k)}}{\partial x}(i, j) &= \left(I^{(k)}(i+1, j) - I^{(k)}(i-1, j) \right) \gg 6 \\ \frac{\partial I^{(k)}}{\partial y}(i, j) &= \left(I^{(k)}(i, j+1) - I^{(k)}(i, j-1) \right) \gg 6\end{aligned}\quad (9)$$

Thứ hai, sau đó các thông số tương liên $\psi_x(i, j)$, $\psi_y(i, j)$ và $\theta(i, j)$ được sử dụng cho quy trình BDOF được tính toán là:

$$\begin{aligned}\psi_x(i, j) &= \left(\frac{\partial I^{(1)}}{\partial x}(i, j) + \frac{\partial I^{(0)}}{\partial x}(i, j) \right) \gg 1 \\ \psi_y(i, j) &= \left(\frac{\partial I^{(1)}}{\partial y}(i, j) + \frac{\partial I^{(0)}}{\partial y}(i, j) \right) \gg 1 \\ \theta(i, j) &= \left(I^{(1)}(i, j) \gg 4 \right) - \left(I^{(0)}(i, j) \gg 4 \right)\end{aligned}\quad (10)$$

Thứ ba, các giá trị S_1 , S_2 , S_3 , S_5 và S_6 được tính toán là:

$$\begin{aligned}S_1 &= \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \psi_x(i, j), \\ S_3 &= \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \psi_x(i, j) \\ S_2 &= \sum_{(i,j) \in \Omega} \psi_x(i, j) \cdot \psi_y(i, j) \\ S_5 &= \sum_{(i,j) \in \Omega} \psi_y(i, j) \cdot \psi_y(i, j) \\ S_6 &= \sum_{(i,j) \in \Omega} \theta(i, j) \cdot \psi_y(i, j)\end{aligned}\quad (11)$$

Thứ tư, sự tinh chỉnh chuyển động (v_x, v_y) của mỗi 4×4 khối con được dẫn xuất là:

$$v_x = S_1 > 0 ? \text{clip3} \left(-th_{BDOF}, th_{BDOF}, -((S_3 \cdot 2^3) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0 \quad (12)$$

$$v_y = S_5 > 0? \text{clip3} \left(-th_{BDOF}, th_{BDOF}, - \left((S_6 \cdot 2^3 - ((v_x S_{2,m}) \ll n_{S_2} + v_x S_{2,s})/2) \gg [\log_2 S_5] \right) \right):$$

Đầu tiên, các mẫu dự đoán đôi cuối cùng của CU được tính toán bằng cách ngoại suy các mẫu dự đoán L0/L1 dọc theo quỹ đạo chuyển động dựa trên mô hình luồng quang học, như được chỉ ra bởi:

$$pred_{BDOF}(x, y) = (I^{(0)}(x, y) + I^{(1)}(x, y) + b + o_{offset}) \gg shift$$

$$b = rnd \left(\left(v_x \left(\frac{\partial I^{(1)}(x, y)}{\partial x} - \frac{\partial I^{(0)}(x, y)}{\partial x} \right) \right) / 2 \right) + rnd \left(\left(v_y \left(\frac{\partial I^{(1)}(x, y)}{\partial y} - \frac{\partial I^{(0)}(x, y)}{\partial y} \right) \right) / 2 \right) \quad (13)$$

Fig.6 thể hiện ví dụ phương pháp điều khiển độ rộng bit BDOF theo sáng chế.

Trong bước 610, thu nhận ảnh tham chiếu thứ nhất $I^{(0)}$ và ảnh tham chiếu thứ hai $I^{(1)}$ được kết hợp với khối con. Ảnh tham chiếu thứ nhất $I^{(0)}$ ở trước ảnh hiện tại và ảnh tham chiếu thứ hai $I^{(1)}$ ở sau ảnh hiện tại trong thứ tự hiển thị.

Trong bước 612, thu nhận mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ từ khối con đến khối tham chiếu trong ảnh tham chiếu thứ nhất $I^{(0)}$.

Trong bước 614, thu nhận mẫu dự đoán thứ hai $I^{(1)}(i, j)$ từ khối con đến khối tham chiếu trong ảnh tham chiếu thứ hai $I^{(1)}$.

Trong bước 616, điều khiển các độ rộng bit bên trong của luồng quang học hai chiều (BDOF) bằng cách thay đổi các độ rộng bit bên trong của được dẫn xuất các giá trị gradient. Các thông số trung gian gồm các giá trị gradient ngang và các giá trị gradient

đọc được dẫn xuất dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$.

Trong bước 618, thu nhận các giá trị sự tinh chỉnh chuyển động dựa trên BDOF và cửa sổ cục bộ đang được áp dụng cho khối con sử dụng mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j)$. Cửa sổ cục bộ bao phủ khối con và cửa sổ cục bộ có kích thước cửa sổ nhỏ hơn 6×6 điểm ảnh.

Trong bước 620, thu nhận các mẫu dự đoán đôi của khối con dựa trên các giá trị sự tinh chỉnh chuyển động.

Fig.7 thể hiện môi trường điện toán 710 được ghép nối với giao diện người dùng 760. Môi trường điện toán 710 có thể là một phần của máy chủ xử lý dữ liệu. Môi trường điện toán 710 gồm bộ xử lý 720, bộ nhớ 740, và I/O giao diện 750.

Bộ xử lý 720 thường điều khiển các hoạt động tổng thể của môi trường điện toán 710, như là các hoạt động được kết hợp với sự hiển thị, thu thập dữ liệu, truyền thông dữ liệu, và xử lý hình ảnh. Bộ xử lý 720 có thể gồm một hoặc nhiều bộ xử lý để thực thi các lệnh để thực hiện tất cả hoặc một số bước trong số các bước trong các phương pháp được mô tả ở trên. Ngoài ra, bộ xử lý 720 có thể gồm một hoặc nhiều môđun mà tạo thuận tiện cho sự tương tác giữa bộ xử lý 720 và các thành phần khác. Bộ xử lý có thể là đơn vị xử lý trung tâm (Central Processing Unit, CPU), bộ vi xử lý, máy chip đơn lẻ, GPU, hoặc tương tự.

Bộ nhớ 740 được tạo cấu hình để lưu trữ các loại dữ liệu khác nhau để hỗ trợ các hoạt động của môi trường điện toán 710. Các ví dụ về dữ liệu như vậy bao gồm các lệnh cho các ứng dụng hoặc các phương pháp bất kỳ mà được vận hành trên môi trường điện toán 710, các bộ dữ liệu video, dữ liệu hình ảnh, v.v. Bộ nhớ 740 có thể được triển khai bằng cách sử dụng loại bất kỳ trong số các thiết bị bộ nhớ khả biến hoặc bất khả biến,

hoặc tổ hợp của chúng, như là bộ nhớ truy cập ngẫu nhiên tĩnh (Static Random Access Memory, SRAM), bộ nhớ chỉ đọc lập trình được, xóa được bằng điện (Electrically Erasable Programmable Read-Only Memory, EEPROM), bộ nhớ chỉ đọc lập trình được, xóa được (Erasable Programmable Read-Only Memory, EPROM), bộ nhớ chỉ đọc lập trình được (Programmable Read-Only Memory, PROM), bộ nhớ chỉ đọc (Read-Only Memory, ROM), bộ nhớ từ, bộ nhớ tác động nhanh, đĩa từ hoặc quang.

Giao diện I/O 750 cung cấp giao diện giữa bộ xử lý 720 và các môđun giao diện ngoại vi, như là bàn phím, bánh xe nhấp chuột, các nút, và tương tự. Các nút có thể gồm nhưng không bị giới hạn ở, nút Home, nút khởi động quét, và nút dừng quét. Giao diện I/O 750 có thể được ghép nối với bộ mã hóa và bộ giải mã.

Trong phương án, sáng chế cũng đề xuất phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp bao gồm nhiều chương trình, như là được bao gồm trong bộ nhớ 740, có thể thực thi được bởi bộ xử lý 720 trong môi trường điện toán 710, để thực hiện các phương pháp được mô tả ở trên. Ví dụ, phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp có thể là ROM, RAM, CD-ROM, băng từ, đĩa mềm, thiết bị lưu trữ dữ liệu quang hoặc tương tự.

Phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp đang lưu trữ nhiều chương trình để thực thi bởi thiết bị điện toán đang có một hoặc nhiều bộ xử lý, trong đó nhiều chương trình, khi được thực thi bởi một hoặc nhiều bộ xử lý, sẽ làm cho thiết bị điện toán thực hiện các phương pháp được mô tả ở trên để điều khiển độ rộng bit cho BDOF.

Trong phương án, môi trường điện toán có thể được triển khai với một hoặc nhiều mạch tích hợp chuyên dụng (Application Specific Integrated Circuit, ASIC), bộ xử lý tín hiệu số (Digital Signal Processor, DSP), thiết bị xử lý tín hiệu số (Digital Signal

Processing Device, DSPD), thiết bị logic lập trình được (Programmable Logic Device, PLD), mảng cổng lập trình được dạng trường (Field Programmable Gate Array, FPGA), bộ điều khiển, bộ vi điều khiển, bộ vi xử lý, hoặc các thành phần điện tử khác, để thực hiện các phương pháp được mô tả trên đây.

YÊU CẦU BẢO HỘ

1. Phương pháp điều khiển độ rộng bit của luồng quang học hai chiều (Bi-Directional Optical Flow, BDOF) để giải mã tín hiệu video bao gồm các bước:

thu nhận ảnh tham chiếu thứ nhất và ảnh tham chiếu thứ hai được kết hợp với khối con video, trong đó ảnh tham chiếu thứ nhất ở trước ảnh hiện tại và ảnh tham chiếu thứ hai ở sau ảnh hiện tại trong thứ tự hiển thị;

thu nhận các mẫu dự đoán thứ nhất $I^{(0)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ nhất;

thu nhận các mẫu dự đoán thứ hai $I^{(1)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ hai;

thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian, trong đó các thông số trung gian bao gồm một thông số hoặc tổ hợp của các thông số sau đây: các gradient ngang, các gradient dọc, và các sự chênh lệch mẫu dựa trên các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$; và

thu nhận các mẫu dự đoán đôi cho các mẫu trong khối con video dựa trên các sự tinh chỉnh chuyển động,

trong đó bước điều khiển các độ rộng bit của các thông số trung gian bao gồm việc giảm các độ rộng bit của các gradient ngang và các gradient dọc bởi giá trị dịch chuyển bit thứ nhất, giá trị dịch chuyển bit thứ nhất là lớn hơn 4.

2. Phương pháp theo điểm 1, trong đó bước thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian bao gồm các việc, cho một mẫu trong khối con video:

thu nhận gradient ngang thứ nhất cho một mẫu nói trên dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i+1, j)$ và mẫu dự đoán thứ nhất $I^{(0)}(i-1, j)$, trong đó i và j tương ứng với các tọa độ của một mẫu;

thu nhận gradient ngang thứ hai cho một mẫu nói trên dựa trên mẫu dự đoán thứ hai $I^{(1)}(i+1, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i-1, j)$;

thu nhận gradient dọc thứ nhất cho một mẫu nói trên dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i, j+1)$ và mẫu dự đoán thứ nhất $I^{(0)}(i, j-1)$; và

thu nhận gradient dọc thứ hai cho một mẫu nói trên dựa trên mẫu dự đoán thứ hai $I^{(1)}(i, j+1)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j-1)$.

3. Phương pháp theo điểm 1, trong đó bước thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian còn bao gồm các việc, cho một mẫu trong khối con video:

thu nhận giá trị tương liên thứ nhất, trong đó giá trị tương liên thứ nhất là tổng của các gradient ngang của các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$ đối với một mẫu nói trên;

thu nhận giá trị tương liên thứ hai, trong đó giá trị tương liên thứ hai là tổng của các gradient dọc của các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$ đối với một mẫu nói trên;

thu nhận giá trị tương liên thứ nhất được sửa đổi bằng cách dịch chuyển sang phải giá trị tương liên thứ nhất nhờ sử dụng giá trị dịch chuyển bit thứ hai; và

thu nhận giá trị tương liên thứ hai được sửa đổi bằng cách dịch chuyển sang phải giá trị tương liên thứ hai nhờ sử dụng giá trị dịch chuyển bit thứ hai.

4. Phương pháp theo điểm 3, trong đó bước thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian còn bao gồm các việc, cho một mẫu trong khối con video:

thu nhận mẫu dự đoán được sửa đổi thứ nhất bằng cách dịch chuyển sang phải mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ cho một mẫu nói trên nhờ sử dụng giá trị dịch chuyển bit thứ ba;

thu nhận mẫu dự đoán được sửa đổi thứ hai bằng cách dịch chuyển sang phải mẫu dự đoán thứ hai $I^{(1)}(i, j)$ cho một mẫu nói trên nhờ sử dụng giá trị dịch chuyển bit thứ ba; và

thu nhận sự chênh lệch mẫu cho một mẫu nói trên, trong đó sự chênh lệch mẫu là sự chênh lệch giữa mẫu dự đoán được sửa đổi thứ nhất và mẫu dự đoán được sửa đổi thứ hai.

5. Phương pháp theo điểm 4, trong đó giá trị dịch chuyển bit thứ hai và giá trị dịch chuyển bit thứ ba bằng một và bốn, một cách tương ứng.

6. Phương pháp theo điểm 4, phương pháp còn bao gồm các bước:

thu nhận giá trị tổng bên trong thứ nhất dựa trên tổng của các giá trị tương liên thứ nhất và các giá trị tương liên thứ hai trong mỗi khối con 4x4 của khối video;

cắt xén giá trị tổng bên trong thứ nhất dựa trên thông số thứ nhất, trong đó thông số thứ nhất điều khiển độ sâu bit đầu ra của giá trị tổng bên trong thứ nhất;

thu nhận giá trị tổng bên trong thứ hai dựa trên tổng của các giá trị tương liên thứ hai và các giá trị tương liên thứ ba trong mỗi khối con 4x4 của khối video; và

cắt xén giá trị tổng bên trong thứ hai dựa trên thông số thứ hai, trong đó thông số thứ hai điều khiển độ sâu bit đầu ra của giá trị tổng bên trong thứ hai.

7. Phương pháp theo điểm 6, trong đó thông số thứ nhất bằng 26 và thông số thứ hai bằng 28.

8. Thiết bị điện toán bao gồm:

một hoặc nhiều bộ xử lý; và

bộ nhớ với các lệnh để giải mã tín hiệu video được lưu trữ ở trong đó, các lệnh, lúc thực thi bởi một hoặc nhiều bộ xử lý, thì làm cho thiết bị điện toán:

thu nhận ảnh tham chiếu thứ nhất và ảnh tham chiếu thứ hai được kết hợp với khối con video, trong đó ảnh tham chiếu thứ nhất ở trước ảnh hiện tại và ảnh tham chiếu thứ hai ở sau ảnh hiện tại trong thứ tự hiển thị;

thu nhận các mẫu dự đoán thứ nhất $I^{(0)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ nhất;

thu nhận các mẫu dự đoán thứ hai $I^{(1)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ hai;

thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian, trong đó các thông số trung gian bao gồm một thông số hoặc tổ hợp của các thông số sau đây: các gradient ngang, các gradient dọc, và các sự chênh lệch mẫu dựa trên các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$; và

thu nhận các mẫu dự đoán đôi cho các mẫu trong khối con video dựa trên các sự tinh chỉnh chuyển động,

trong đó việc điều khiển các độ rộng bit của các thông số trung gian bao gồm việc giảm các độ rộng bit của các gradient ngang và các gradient dọc bởi giá trị dịch chuyển bit thứ nhất, giá trị dịch chuyển bit thứ nhất là lớn hơn 4.

9. Thiết bị điện toán theo điểm 8, trong đó các lệnh, lúc thực thi bởi một hoặc nhiều bộ xử lý, thì còn làm cho thiết bị điện toán thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách, cho một mẫu trong khối con video:

thu nhận gradient ngang thứ nhất cho một mẫu nói trên dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i+1, j)$ và mẫu dự đoán thứ nhất $I^{(0)}(i-1, j)$, trong đó i và j tương ứng với các tọa độ của một mẫu;

thu nhận gradient ngang thứ hai cho một mẫu nói trên dựa trên mẫu dự đoán thứ hai $I^{(1)}(i+1, j)$ và mẫu dự đoán thứ hai $I^{(1)}(i-1, j)$;

thu nhận gradient dọc thứ nhất cho một mẫu nói trên dựa trên mẫu dự đoán thứ nhất $I^{(0)}(i, j+1)$ và mẫu dự đoán thứ nhất $I^{(0)}(i, j-1)$; và

thu nhận gradient dọc thứ hai cho một mẫu nói trên dựa trên mẫu dự đoán thứ hai $I^{(1)}(i, j+1)$ và mẫu dự đoán thứ hai $I^{(1)}(i, j-1)$.

10. Thiết bị điện toán theo điểm 8, trong đó các lệnh, lúc thực thi bởi một hoặc nhiều bộ xử lý, thì còn làm cho thiết bị điện toán thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách, cho một mẫu trong khối con video:

thu nhận giá trị tương liên thứ nhất, trong đó giá trị tương liên thứ nhất là tổng của các gradient ngang của các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$ đối với một mẫu nói trên;

thu nhận giá trị tương liên thứ hai, trong đó giá trị tương liên thứ hai là tổng của các gradient dọc của các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$ đối với một mẫu nói trên;

thu nhận giá trị tương liên thứ nhất được sửa đổi bằng cách dịch chuyển sang phải giá trị tương liên thứ nhất nhờ sử dụng giá trị dịch chuyển bit thứ hai; và

thu nhận giá trị tương liên thứ hai được sửa đổi bằng cách dịch chuyển sang phải giá trị tương liên thứ hai nhờ sử dụng giá trị dịch chuyển bit thứ hai.

11. Thiết bị điện toán theo điểm 10, trong đó các lệnh, lúc thực thi bởi một hoặc nhiều bộ xử lý, thì còn làm cho thiết bị điện toán thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách, cho một mẫu trong khối con video:

thu nhận mẫu dự đoán được sửa đổi thứ nhất bằng cách dịch chuyển sang phải mẫu dự đoán thứ nhất $I^{(0)}(i, j)$ cho một mẫu nói trên nhờ sử dụng giá trị dịch chuyển bit thứ ba;

thu nhận mẫu dự đoán được sửa đổi thứ hai bằng cách dịch chuyển sang phải mẫu dự đoán thứ hai $I^{(1)}(i, j)$ cho một mẫu nói trên nhờ sử dụng giá trị dịch chuyển bit thứ ba; và

thu nhận sự chênh lệch mẫu cho một mẫu nói trên, trong đó sự chênh lệch mẫu là sự chênh lệch giữa mẫu dự đoán được sửa đổi thứ nhất và mẫu dự đoán được sửa đổi thứ hai.

12. Thiết bị điện toán theo điểm 11, trong đó giá trị dịch chuyển bit thứ hai và giá trị dịch chuyển bit thứ ba bằng một và bốn, một cách tương ứng.

13. Thiết bị điện toán theo điểm 11, trong đó các lệnh, lúc thực thi bởi một hoặc nhiều bộ xử lý, thì còn làm cho thiết bị điện toán:

thu nhận giá trị tổng bên trong thứ nhất dựa trên tổng của các giá trị tương liên thứ nhất và các giá trị tương liên thứ hai trong mỗi khối con 4x4 của khối video;

cắt xén giá trị tổng bên trong thứ nhất dựa trên thông số thứ nhất, trong đó thông số thứ nhất điều khiển độ sâu bit đầu ra của giá trị tổng bên trong thứ nhất;

thu nhận giá trị tổng bên trong thứ hai dựa trên tổng của các giá trị tương liên thứ hai và các giá trị tương liên thứ ba trong mỗi khối con 4x4 của khối video; và

cắt xén giá trị tổng bên trong thứ hai dựa trên thông số thứ hai, trong đó thông số thứ hai điều khiển độ sâu bit đầu ra của giá trị tổng bên trong thứ hai.

14. Thiết bị điện toán theo điểm 13, trong đó thông số thứ nhất bằng 26 và thông số thứ hai bằng 28.

15. Phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp lưu trữ các lệnh để giải mã tín hiệu video, các lệnh này, lúc thực thi bởi bộ xử lý, làm cho bộ xử lý:

thu nhận ảnh tham chiếu thứ nhất và ảnh tham chiếu thứ hai được kết hợp với khối con video, trong đó ảnh tham chiếu thứ nhất ở trước ảnh hiện tại và ảnh tham chiếu thứ hai ở sau ảnh hiện tại trong thứ tự hiển thị;

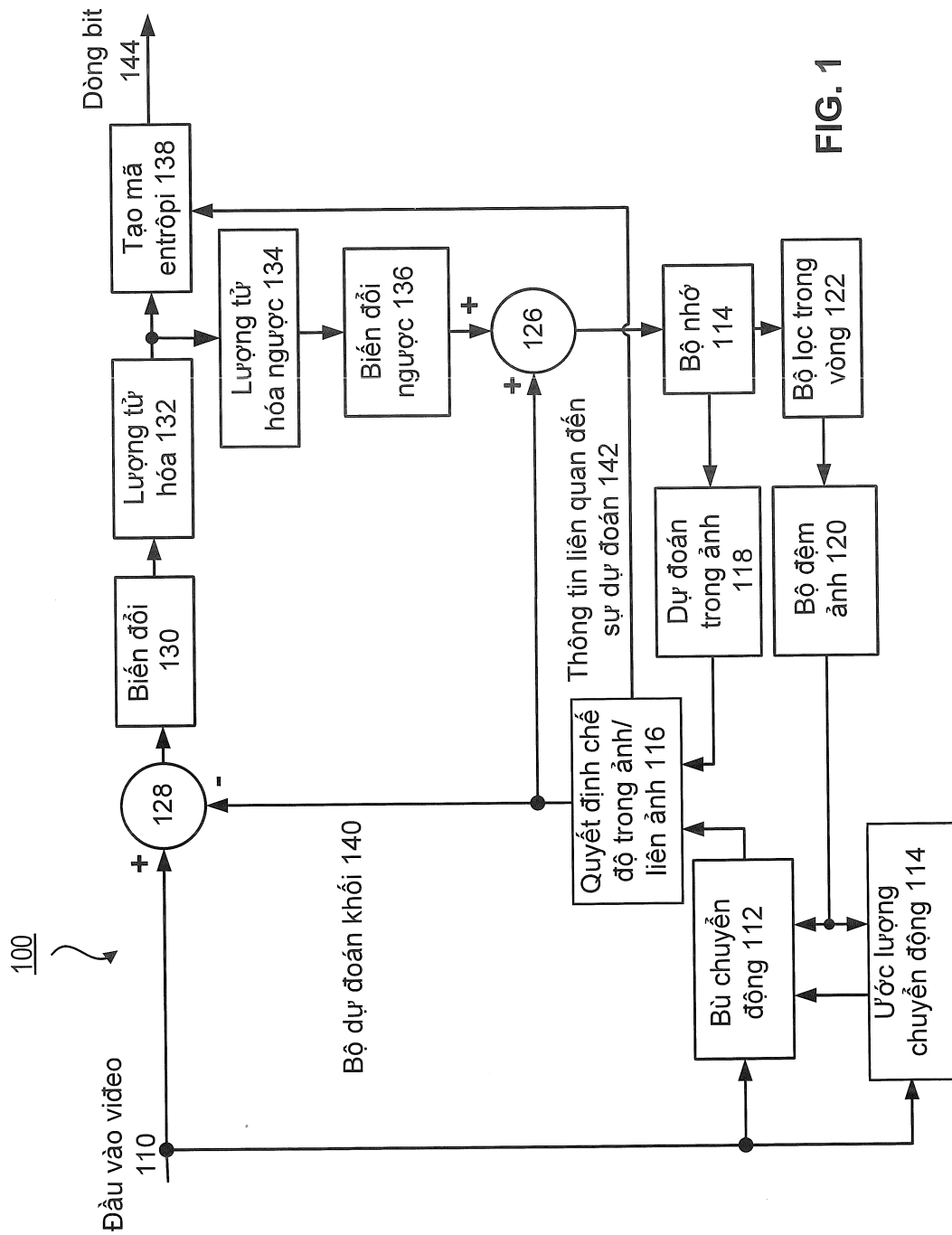
thu nhận các mẫu dự đoán thứ nhất $I^{(0)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ nhất;

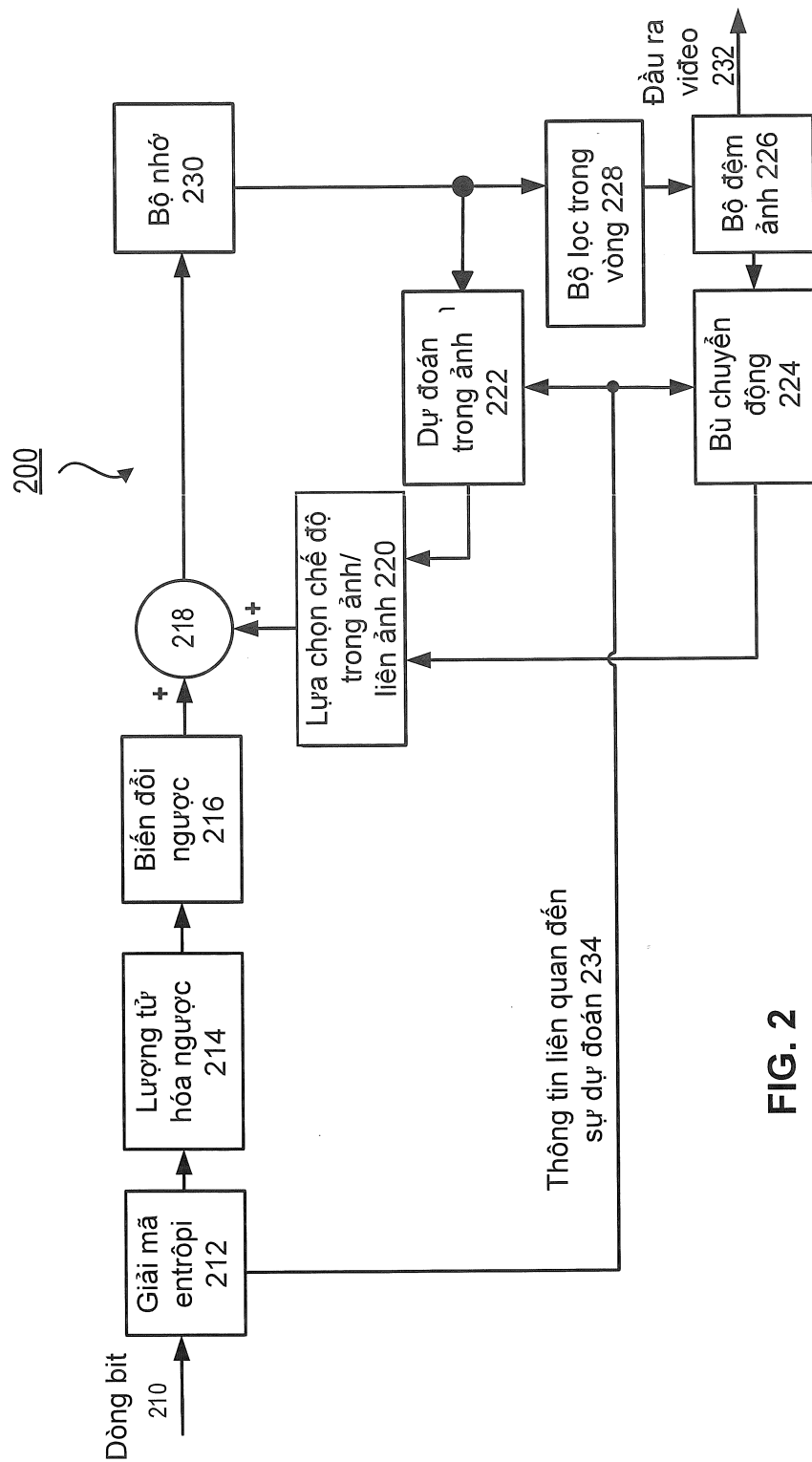
thu nhận các mẫu dự đoán thứ hai $I^{(1)}$ cho các mẫu trong khối con video từ ảnh tham chiếu thứ hai;

thu nhận các sự tinh chỉnh chuyển động cho các mẫu trong khối con video bằng cách điều khiển các độ rộng bit của các thông số trung gian, trong đó các thông số trung gian bao gồm một thông số hoặc tổ hợp của các thông số sau đây: các gradient ngang, các gradient dọc, và các sự chênh lệch mẫu dựa trên các mẫu dự đoán thứ nhất $I^{(0)}$ và các mẫu dự đoán thứ hai $I^{(1)}$; và

thu nhận các mẫu dự đoán đôi cho các mẫu trong khối con video dựa trên các sự tinh chỉnh chuyển động,

trong đó việc điều khiển các độ rộng bit của các thông số trung gian bao gồm việc giảm các độ rộng bit của các gradient ngang và các gradient dọc bởi giá trị dịch chuyển bit thứ nhất, giá trị dịch chuyển bit thứ nhất là lớn hơn 4.





3 / 6

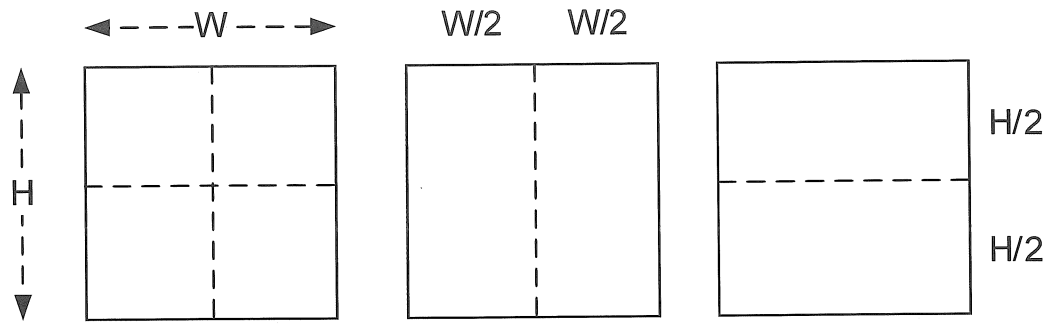


FIG. 3A

FIG. 3B

FIG. 3C

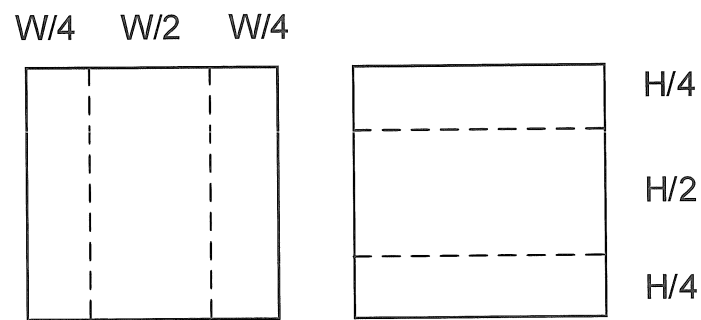


FIG. 3D

FIG. 3E

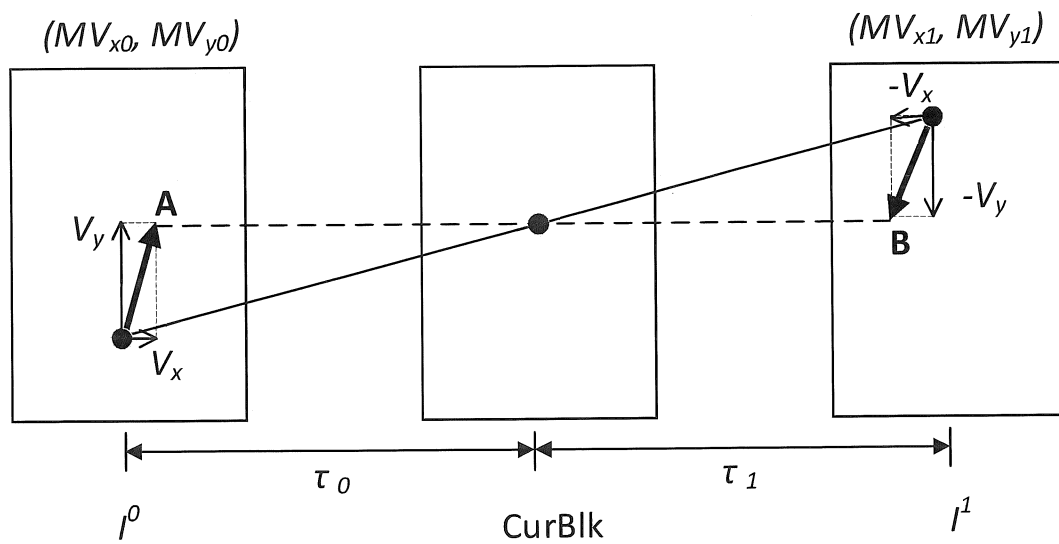


FIG. 4

4 / 6

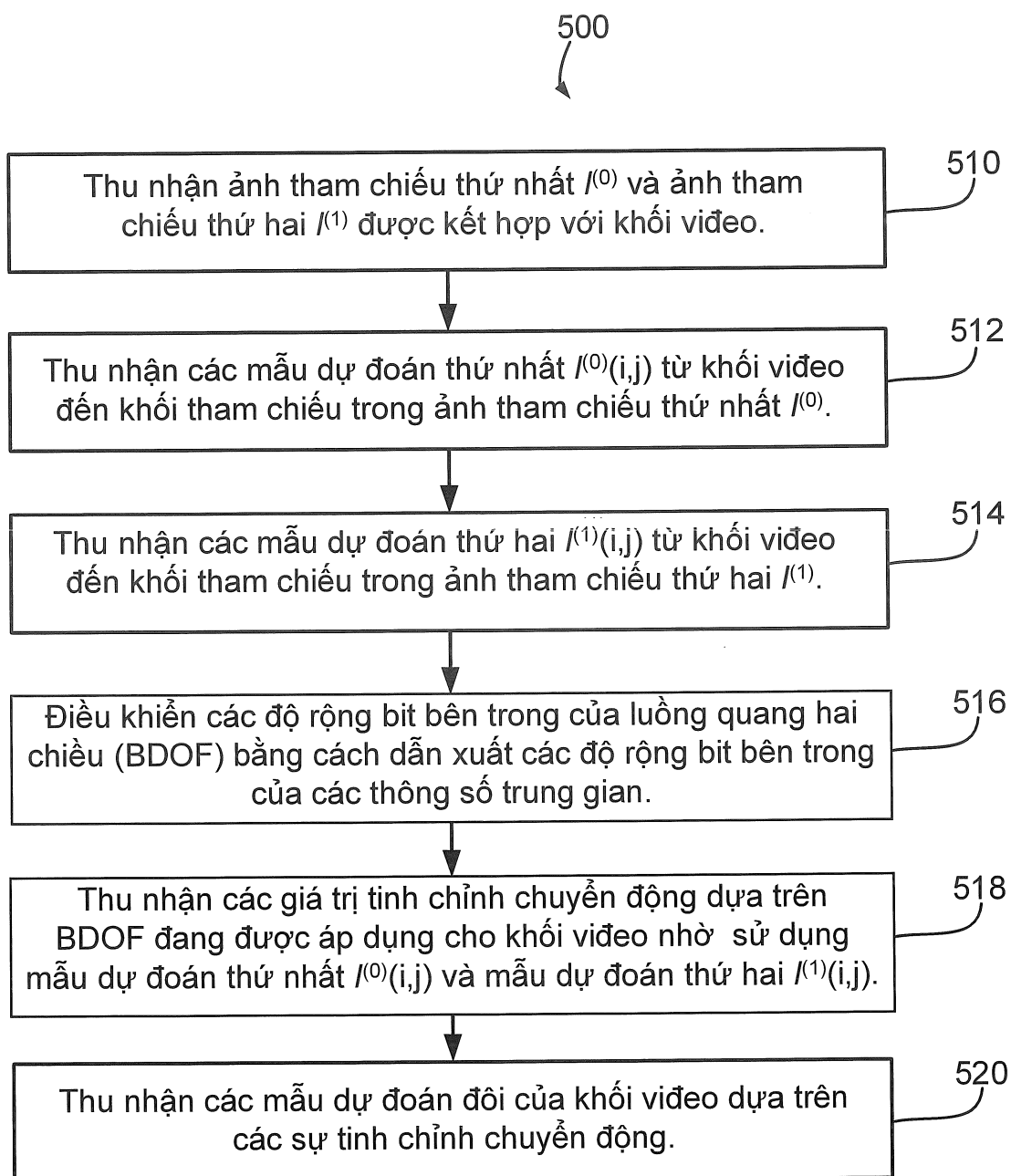


FIG. 5

5 / 6

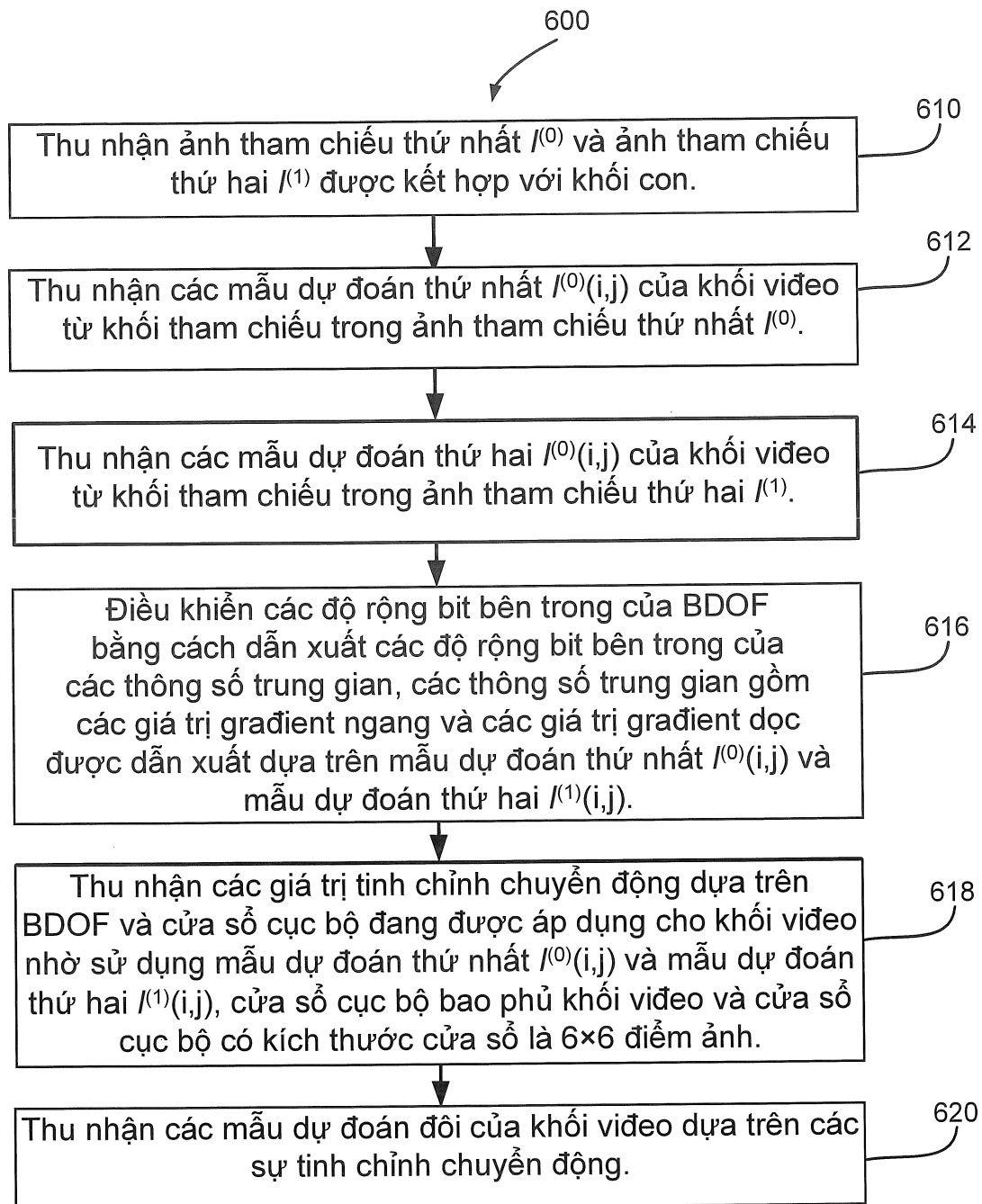


FIG. 6

6 / 6

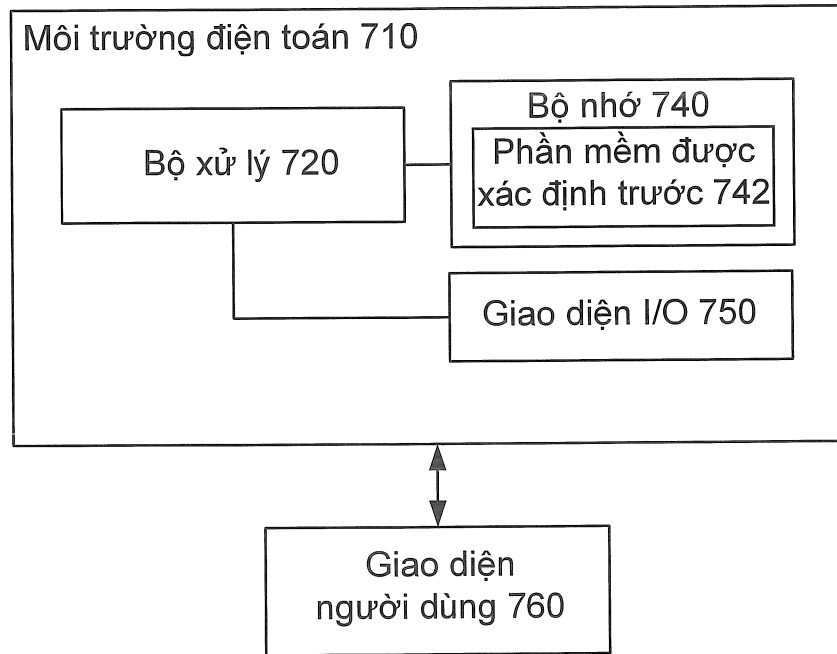


FIG. 7