



(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)



1-0053967

(51)^{2020.01} H04N 19/51; H04N 19/52

(13) B

(21) 1-2021-05388

(22) 21/02/2020

(86) PCT/CN2020/076178 21/02/2020

(87) WO2020/169083 27/08/2020

(30) 201931007114 22/02/2019 IN

(45) 25/11/2025 452

(43) 25/11/2021 404A

(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)

Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, China

(72) ESENLİK, Semih (TR); SETHURAMAN, Sriram (IN); A, Jeeva Raj (IN);
KOTECHA, Sagar (IN).

(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP TẠO MÃ VIDEO, BỘ MÃ HÓA VÀ BỘ GIẢI MÃ

(21) 1-2021-05388

(57) Sáng chế đề cập đến phương pháp tạo mã video được thực hiện trong thiết bị giải mã hoặc thiết bị mã hóa, phương pháp này bao gồm các bước: thu nhận các vector chuyển động ban đầu đối với khối hiện hành; thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu; tính toán chi phí so khớp thứ nhất theo các dự đoán thứ nhất; xác định liệu quy trình tinh chế dòng quang học có nên được thực hiện hay không, theo ít nhất một điều kiện thiết đặt trước, ít nhất một điều kiện thiết đặt trước này bao gồm điều kiện về việc liệu chi phí so khớp thứ nhất tính toán được có bằng hoặc lớn hơn ngưỡng hay không; và thực hiện quy trình tinh chế dòng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành, khi xác định được rằng quy trình tinh chế dòng quang học nên được thực hiện. Ngoài ra, sáng chế đề cập đến bộ mã hóa và bộ giải mã.

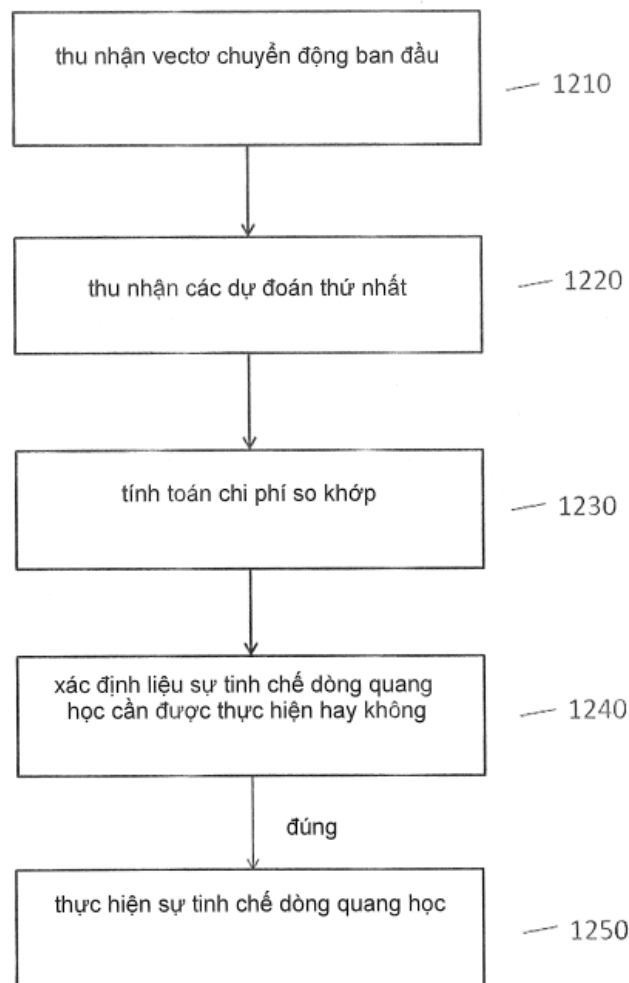


Fig.10

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến lĩnh vực xử lý hình ảnh và cụ thể hơn là đến sự tinh chế dòng quang học.

Tình trạng kỹ thuật của sáng chế

Tạo mã video (mã hóa và giải mã video) được sử dụng trong phạm vi rộng rãi của các ứng dụng video số, ví dụ TV số phát rộng, truyền video qua internet và mạng di động, các ứng dụng đàm thoại thời gian thực như tán gẫu bằng video, hội nghị bằng video, các đĩa DVD và Blu-ray, các hệ thống thu thập và biên tập nội dung video, và các máy quay của các ứng dụng bảo mật.

Lượng dữ liệu video cần để mô tả thậm chí video tương đối ngắn có thể là quan trọng, mà có thể dẫn đến những khó khăn khi dữ liệu cần được tạo luồng hoặc nếu không thì được truyền thông qua mạng truyền thông với dung lượng dải thông hạn chế. Vì vậy, dữ liệu video thường được nén trước khi được truyền thông qua các mạng viễn thông đương thời. Kích thước của video cũng có thể là vấn đề khi video được lưu trữ trên thiết bị lưu trữ do các tài nguyên bộ nhớ có thể bị giới hạn. Các thiết bị nén video thường sử dụng phần mềm và/hoặc phần cứng ở nguồn để tạo mã dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó làm giảm lượng dữ liệu cần để đại diện cho các ảnh video số. Dữ liệu đã nén sau đó được nhận tại điểm đến bởi thiết bị giải nén video để giải mã dữ liệu video. Với các tài nguyên mạng hạn chế và các nhu cầu luôn tăng lên về chất lượng video cao hơn, các kỹ thuật nén và giải nén cải tiến để nâng cao tỷ lệ nén với việc đánh đổi một ít cho đến không phải đánh đổi về chất lượng hình ảnh luôn được mong đợi.

Gần đây, việc tạo mã dự đoán liên ảnh đã được cải tiến bằng sự tinh chế dòng quang học song dự đoán. Kỹ thuật này có thể nâng cao độ chính xác của việc dự đoán liên ảnh đối với khối hiện hành của hình ảnh cần được tạo mã. Tuy nhiên, sự tinh chế dòng quang học song dự đoán là tương đối tốn kém trên phương diện gánh nặng tính toán. Vì vậy, sự thỏa hiệp giữa dự đoán liên ảnh chính xác và gánh nặng tính toán cần phải được tìm ra. Sáng chế giải quyết vấn đề này.

Bản chất kỹ thuật của sáng chế

Các phương án của sáng chế đề xuất các thiết bị và các phương pháp để mã hóa và giải mã theo các điểm yêu cầu bảo hộ độc lập.

Các mục đích trên đây và các mục đích khác sẽ đạt được nhờ đối tượng của các điểm yêu cầu bảo hộ độc lập. Các hình thức thực hiện thêm nữa sẽ rõ ràng từ các điểm yêu cầu bảo hộ phụ thuộc, phần mô tả và các hình vẽ.

Sáng chế đề xuất phương pháp tạo mã video được thực hiện trong thiết bị giải mã hoặc thiết bị mã hóa, phương pháp này bao gồm các bước:

thu nhận các vector chuyển động ban đầu đối với khối hiện hành (ví dụ, khối tạo mã hoặc khối dự đoán hoặc khối con);

thu nhận các dự đoán thứ nhất (hai giá trị dự đoán để song dự đoán liên ảnh) đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu;

tính toán chi phí so khớp thứ nhất (ví dụ, gồm hoặc gồm đo lường sự tương tự (hoặc không tương tự) nào đó; cũng xem phần mô tả chi tiết dưới đây) theo các dự đoán thứ nhất;

xác định liệu quy trình tinh chế dòng quang học có nên được thực hiện hay không, theo ít nhất một điều kiện thiết đặt trước, ít nhất một điều kiện thiết đặt trước này bao gồm điều kiện về việc liệu chi phí so khớp thứ nhất tính toán được có bằng hoặc lớn hơn ngưỡng hay không;

thực hiện quy trình tinh chế dòng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành, khi xác định được rằng quy trình tinh chế dòng quang học nên được thực hiện.

Vì vậy, theo sáng chế, hiệu năng của sự tinh chế dòng quang học, cụ thể, sự tinh chế dòng quang học hai chiều được thực hiện trên cơ sở có điều kiện. Sự tinh chế dòng quang học tương đối tốn kém chỉ được thực hiện trong các trường hợp nhất định mà cho phép sự cải thiện mong muốn thích hợp về độ chính xác của toàn bộ quy trình dự đoán liên ảnh. Nếu xác định được rằng sự tinh chế dòng quang học có thể không dẫn đến sự cải thiện độ chính xác của dự đoán liên ảnh mà là đáng giá gánh nặng tính toán tương đối cao cần để thực hiện sự tinh chế dòng quang học, sự tinh chế dòng quang học có thể

được chặn lại. Vì vậy, thời gian giải mã có thể được giảm đi đáng kể. Các vector chuyển động ban đầu có thể được báo hiệu trong dòng bit. Theo cách khác, các dự đoán vector chuyển động và các thành phần chênh lệch vector chuyển động có thể được cung cấp cho các vector chuyển động ban đầu.

Ví dụ, ít nhất một điều kiện thiết đặt trước bao gồm điều kiện mà khối hiện hành được phép được dự báo bởi sự tinh chế vector chuyển động phía bộ giải mã. Điều kiện cụ thể này có trong ít nhất một điều kiện thiết đặt trước có thể được kiểm tra ở địa điểm thứ nhất để tránh các nỗ lực tính toán không cần thiết.

Theo một phương án cụ thể, xác định được rằng quy trình tinh chế dòng quang học sẽ được thực hiện, khi xác định được rằng tất cả trong số ít nhất một điều kiện được thiết đặt trước được đáp ứng. Về nguyên tắc, ít nhất một điều kiện thiết đặt trước có thể bao gồm một hoặc nhiều điều kiện bổ sung. Ví dụ, ít nhất một điều kiện được thiết đặt trước có thể bao gồm điều kiện mà cờ cụ thể được thiết đặt (ở 1 chẳng hạn) để có quy trình tinh chế quang học được thực hiện. Nếu các điều kiện không phải tất cả đều được đáp ứng, thì không có sự tinh chế dòng quang học nào có thể được thực hiện theo một phương án cụ thể để làm giảm các nhu cầu tính toán.

Các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành có thể được thu nhận dựa trên bộ lọc nội suy thứ nhất để thu được độ chính xác điểm ảnh con nào đó. Cụ thể, bộ lọc nội suy thứ nhất có thể là bộ lọc nội suy song tuyến tính tương đối đơn giản mà cho phép xử lý lọc nhanh.

Phương pháp tạo mã video được thực hiện trong thiết bị giải mã hoặc thiết bị mã hóa có thể bao gồm sự tinh chế vector chuyển động nào đó khác với sự tinh chế dòng quang học. Vì vậy, phương pháp có thể bao gồm bước thu nhận các vector chuyển động đã tinh chế dựa trên các vector chuyển động ban đầu và chi phí so khớp thứ nhất; thu nhận các dự đoán thứ hai đối với giá trị mẫu trong khối hiện hành theo các vector chuyển động đã tinh chế, khi xác định được rằng quy trình tinh chế dòng quang học sẽ được thực hiện, thực hiện sự tinh chế dòng quang học dựa trên các dự đoán thứ hai (đại diện cho các dự đoán đã được tinh chế). Tổng độ chính xác của quy trình dự đoán liên ảnh có thể được nâng cao nhờ sử dụng các vector chuyển động đã tinh chế.

Cần lưu ý rằng các dự đoán thứ nhất cũng như chi phí so khớp thứ nhất đã được

tính toán cho sự tinh chế vector chuyển động. Do đó, không có các phép tính thêm nữa cần cho quyết định về sự chấm dứt/ngăn chặn sớm quy trình tinh chế dòng quang học, nhưng các kết quả của các phép tính trước đó liên quan đến quy trình tinh chế vector chuyển động có thể được sử dụng lại.

Trong mỗi phương án trong số các phương án được mô tả trên đây của phương pháp này, bước thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu có thể bao gồm thu nhận số lượng cặp ứng viên dựa trên các vector chuyển động ban đầu và thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên ít nhất một cặp trong số các cặp ứng viên và việc tính toán chi phí so khớp thứ nhất theo các dự đoán thứ nhất có thể bao gồm xác định chi phí so khớp đối với mỗi cặp trong số các cặp ứng viên dựa trên các dự đoán thứ nhất và xác định chi phí so khớp nhỏ nhất của các chi phí so khớp xác định được làm chi phí so khớp thứ nhất.

Đối với sự tinh chế vector chuyển động, số lượng cặp ứng viên đối với các vector chuyển động đã tinh chế có thể được thu nhận, các cặp này bao gồm cặp vector chuyển động ban đầu. Ví dụ, các cặp ứng viên đối với các vector chuyển động đã tinh chế bao gồm cặp vector chuyển động ban đầu (MV_0, MV_1) và các cặp ($MV_0 + (0,1), MV_1 + (0,-1)$), ($MV_0 + (1,0), MV_1 + (-1,0)$), ($MV_0 + (0,-1), MV_1 + (0,1)$), ($MV_0 + (-1,0), MV_1 + (1,0)$), trong đó $(1,-1)$ biểu thị vector mà có sự dịch chuyển là 1 theo chiều ngang (hoặc x) và sự dịch chuyển là -1 theo chiều thẳng đứng (hoặc y). Đối với mỗi cặp trong số các cặp này, chi phí so khớp tương ứng với cặp đó có thể được xác định và chi phí so khớp thứ nhất nêu trên có thể được xác định là chi phí nhỏ nhất trong số các chi phí so khớp được xác định cho các cặp ứng viên đối với các vector chuyển động đã tinh chế. Theo các ví dụ cụ thể, nó có thể là chi phí so khớp tương ứng với cặp vector chuyển động ban đầu (MV_0, MV_1) hoặc ($MV_0' = MV_0 + (0,1), MV_1' = MV_1 + (0,-1)$) với các vector chuyển động đã tinh chế MV_0' và MV_1' .

Việc sử dụng loại chi phí so khớp thứ nhất đó có thể là thuận lợi trên phương diện toàn bộ việc tạo mã.

Các dự đoán thứ hai nêu trên đối với giá trị mẫu trong khối hiện hành có thể được thu nhận theo bộ lọc nội suy thứ hai. Bộ lọc nội suy thứ hai này có thể là bộ lọc nội suy

6 điểm nối hoặc 8 điểm nối mà tương đối tốn kém nhưng có thuận lợi trên phương diện độ chính xác điểm ảnh con.

Các vector chuyển động đã tinh chế nêu trên có thể được thu nhận theo chi phí so khớp thứ hai để điều khiển sự thích hợp của các vector chuyển động đã tinh chế đối với dự đoán liên ảnh. Khi giá trị của chi phí so khớp thứ hai lớn hơn hoặc bằng một giá trị ngưỡng khác, thì có thể xác định được rằng quy trình tinh chế dòng quang học sẽ được thực hiện. Nếu không, có thể xác định được rằng không đáng để thực hiện việc xử lý tinh chế dòng quang học bất kỳ.

Theo một phương án khác, chỉ khi xác định được rằng quy trình tinh chế dòng quang học không nên được thực hiện, thì dự đoán liên ảnh cuối cùng được thu nhận bởi tổng có trọng số của các dự đoán thứ hai. Tổng có trọng số của các dự đoán thứ hai cung cấp độ chính xác nào đó mà có thể được coi là đủ trong các trường hợp mà nó không được coi là thích hợp để thực hiện quy trình tinh chế dòng quang học tương đối tốn kém.

Nói chung, giá trị ngưỡng hoặc giá trị ngưỡng khác có thể là giá trị mà được tính dựa trên độ sâu bit của các dự đoán thứ nhất. Hơn nữa, giá trị ngưỡng có thể được thu nhận theo số lượng mẫu được dự đoán mà được sử dụng để tính chi phí so khớp thứ nhất theo các dự đoán thứ nhất. Hơn nữa, giá trị ngưỡng có thể được thu nhận theo kích thước (độ rộng và độ cao xét về số lượng điểm ảnh) của khối hiện hành. Ví dụ, ngưỡng có thể là $thr = nCbW \times nCbH \times K$, trong đó K là giá trị lớn hơn không (zero), $nCbW$ và $nCbH$ là độ rộng và độ cao của khối hiện hành. Ví dụ, $K = 2$.

Hơn nữa, chi phí so khớp thứ hai nêu trên có thể là chi phí được dẫn xuất thu nhận được nhờ sử dụng các chi phí so khớp được đánh giá trong suốt quá trình tinh chế vector chuyển động và mô hình định trước đối với hình dạng của chi phí so khớp gần vị trí chi phí so khớp tối thiểu. Mô hình định trước trong ngữ cảnh này có thể là mô hình tổ hợp tuyến tính.

Việc sử dụng mô hình định trước đối với hình dạng của chi phí so khớp gần vị trí chi phí so khớp tối thiểu có thể nâng cao độ chính xác của quy trình dự đoán liên ảnh.

Phương pháp theo tất cả các phương án được mô tả trên đây còn có thể bao gồm bước tạo khối dự đoán liên ảnh bao gồm dự đoán liên ảnh cuối cùng đối với giá trị mẫu

trong khối hiện hành.

Hơn nữa, sáng chế đề xuất bộ mã hóa hoặc bộ giải mã bao gồm mạch xử lý nào đó để thực hiện phương pháp theo phương án bất kỳ trong số các phương án được mô tả trên đây. Hơn nữa, sáng chế đề xuất sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo phương án bất kỳ trong số các phương án được mô tả trên đây.

Tất cả các biến thể được mô tả trên đây của phương pháp tạo mã video có thể được thực thi trong bộ giải mã hoặc bộ mã hóa. Vì vậy, sáng chế đề xuất bộ giải mã hoặc bộ mã hóa, bao gồm:

một hoặc nhiều bộ xử lý và phương tiện lưu trữ lâu dài đọc được bằng máy tính được ghép nối với các bộ xử lý và lưu trữ chương trình để thực hiện bởi các bộ xử lý, trong đó chương trình, khi được thực hiện bởi các bộ xử lý, tạo cấu hình bộ giải mã để thực hiện phương pháp theo phương án bất kỳ trong số các phương án được mô tả trên đây.

Tất cả các biến thể được mô tả trên đây của phương pháp tạo mã video có thể được thực thi trong thiết bị để sử dụng trong bộ mã hóa ảnh và/hoặc bộ giải mã ảnh để giải quyết nhu cầu nêu trên. Vì vậy, sáng chế đề xuất thiết bị để sử dụng trong bộ mã hóa ảnh và/hoặc bộ giải mã ảnh, thiết bị này bao gồm bộ phận vector chuyển động ban đầu được tạo cấu hình để thu nhận các vector chuyển động ban đầu đối với khối hiện hành (ví dụ, khối tạo mã hoặc khối dự đoán hoặc khối con); bộ phận dự đoán thứ nhất được tạo cấu hình để thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu; bộ phận tính toán chi phí so khớp thứ nhất được tạo cấu hình để tính toán chi phí so khớp thứ nhất (ví dụ, đo lường sự tương tự hoặc không tương tự) theo các dự đoán thứ nhất; bộ phận xác định quy trình tinh chế dòng quang học được tạo cấu hình để xác định liệu quy trình tinh chế dòng quang học có nên được thực hiện hay không, theo ít nhất một điều kiện thiết đặt trước, ít nhất một điều kiện thiết đặt trước này bao gồm điều kiện về việc liệu chi phí so khớp thứ nhất tính toán được có bằng hoặc lớn hơn ngưỡng hay không; và bộ phận thực hiện quy trình tinh chế dòng quang học được tạo cấu hình để thực hiện quy trình tinh chế dòng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành,

khi xác định được rằng quy trình tinh chế dòng quang học nên được thực hiện.

Thiết bị này, như được mô tả trên đây và với các biến thể của nó được mô tả dưới đây mang lại các ưu điểm giống như các phương pháp được mô tả trên đây.

Ít nhất một điều kiện thiết đặt trước có thể bao gồm điều kiện mà khối hiện hành được phép được dự báo bởi sự tinh chế vector chuyển động phía bộ giải mã.

Bộ phận xác định quy trình tinh chế dòng quang học có thể được tạo cấu hình để xác định rằng quy trình tinh chế dòng quang học sẽ được thực hiện, khi xác định được rằng tất cả trong số ít nhất một điều kiện được thiết đặt trước được đáp ứng.

Thiết bị này có thể bao gồm bộ lọc nội suy thứ nhất (ví dụ, bộ lọc nội suy song tuyến tính) và bộ phận dự đoán thứ nhất có thể được tạo cấu hình để thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành nhờ bộ lọc nội suy thứ nhất.

Thiết bị này còn có thể bao gồm bộ phận vector chuyển động đã tinh chế được tạo cấu hình để thu nhận các vector chuyển động đã tinh chế dựa trên các vector chuyển động ban đầu và chi phí so khớp thứ nhất;

bộ phận dự đoán thứ hai được tạo cấu hình để thu nhận các dự đoán thứ hai đối với giá trị mẫu trong khối hiện hành theo các vector chuyển động đã tinh chế; và

bộ phận thực hiện quy trình tinh chế dòng quang học có thể được tạo cấu hình để thực hiện sự tinh chế dòng quang học dựa trên các dự đoán thứ hai, khi xác định được bởi bộ phận xác định quy trình tinh chế dòng quang học rằng quy trình tinh chế dòng quang học sẽ được thực hiện.

Theo các phương án được mô tả trên đây của thiết bị, bộ phận dự đoán thứ nhất có thể được tạo cấu hình để thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu bằng cách thu nhận số lượng cặp ứng viên dựa trên các vector chuyển động ban đầu và thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên ít nhất một trong số các cặp ứng viên. Hơn nữa, bộ phận tính toán chi phí so khớp thứ nhất có thể được tạo cấu hình để tính toán chi phí so khớp thứ nhất theo các dự đoán thứ nhất bằng cách xác định chi phí so khớp đối với mỗi cặp trong số các cặp ứng viên dựa trên các dự đoán thứ nhất và xác định chi phí so khớp nhỏ nhất trong số các chi phí so khớp xác định được làm chi phí so khớp

thứ nhất.

Theo một phương án, thiết bị này còn có thể bao gồm bộ lọc nội suy thứ hai (ví dụ, bộ lọc nội suy 6 điểm nối hoặc 8 điểm nối tương đối tổn kém của độ chính xác điểm ảnh con tương đối cao) và bộ phận dự đoán thứ hai có thể được tạo cấu hình để thu nhận các dự đoán thứ hai đối với giá trị mẫu trong khối hiện hành nhờ bộ lọc nội suy thứ hai.

Theo một phương án khác, thiết bị này còn bao gồm bộ phận tính toán chi phí so khớp thứ hai được tạo cấu hình để tính toán chi phí so khớp thứ hai và trong đó bộ phận vector chuyển động đã tinh chế được tạo cấu hình để thu nhận các vector chuyển động đã tinh chế theo chi phí so khớp thứ hai. Trong trường hợp này, bộ phận xác định quy trình tinh chế dòng quang học có thể được tạo cấu hình để xác định rằng quy trình tinh chế dòng quang học sẽ được thực hiện khi giá trị của chi phí so khớp thứ hai lớn hơn hoặc bằng một giá trị ngưỡng khác.

Thiết bị này còn có thể bao gồm bộ phận dự đoán tổng có trọng số được tạo cấu hình để thu nhận dự đoán liên ảnh cuối cùng bởi tổng có trọng số của các dự đoán thứ hai chỉ khi xác định được bởi bộ phận xác định quy trình tinh chế dòng quang học rằng quy trình tinh chế dòng quang học không nên được thực hiện.

Hơn nữa, thiết bị này có thể bao gồm bộ phận tính toán ngưỡng được tạo cấu hình để tính toán giá trị ngưỡng hoặc ngưỡng khác dựa trên độ sâu bit của các dự đoán thứ nhất. Ngoài ra, thiết bị này còn có thể bao gồm bộ phận tính toán ngưỡng được tạo cấu hình để tính toán ngưỡng theo số lượng mẫu được dự đoán mà được sử dụng để tính chi phí so khớp thứ nhất theo các dự đoán thứ nhất bởi bộ phận tính toán chi phí so khớp thứ nhất. Ngoài ra, thiết bị này còn có thể bao gồm bộ phận tính toán ngưỡng được tạo cấu hình để tính toán ngưỡng theo kích thước của khối hiện hành. Ví dụ, ngưỡng có thể là $thr = nCbW \times nCbH \times K$, trong đó K là giá trị lớn hơn không (zero), $nCbW$ và $nCbH$ là độ rộng và độ cao của khối hiện hành. Ví dụ, $K = 2$.

Theo các phương án cụ thể, bộ phận tính toán chi phí so khớp thứ hai được tạo cấu hình để tính toán chi phí so khớp thứ hai như là chi phí được dẫn xuất thu nhận được nhờ sử dụng các chi phí so khớp được đánh giá trong suốt quá trình tinh chế vector chuyển động được thực hiện bởi bộ phận vector chuyển động đã tinh chế và mô hình định trước (ví dụ, mô hình tổ hợp tuyến tính) đối với hình dạng của chi phí so khớp gần

vị trí chi phí so khớp tối thiểu.

Thiết bị theo phương án bất kỳ trong số các phương án được mô tả trên đây còn có thể bao gồm bộ phận tạo khối dự đoán liên ảnh được tạo cấu hình để tạo khối dự đoán liên ảnh bao gồm dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành.

Các nội dung chi tiết của một hoặc nhiều phương án được nêu trong các hình vẽ kèm theo và phần mô tả dưới đây. Các dấu hiệu, các mục đích, và các ưu điểm khác sẽ rõ ràng từ phần mô tả, hình vẽ, và yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Các phương án sau đây của sáng chế được mô tả chi tiết hơn dựa vào các hình vẽ kèm theo, trong đó:

Fig.1A là sơ đồ khối thể hiện một ví dụ về hệ thống tạo mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.1B là sơ đồ khối thể hiện một ví dụ khác về hệ thống tạo mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.2 là sơ đồ khối thể hiện một ví dụ về bộ mã hóa video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.3 là sơ đồ khối thể hiện cấu trúc ví dụ của bộ giải mã video được tạo cấu hình để thực hiện các phương án của sáng chế;

Fig.4 là sơ đồ khối minh họa một ví dụ về máy mã hóa hoặc máy giải mã;

Fig.5 là sơ đồ khối minh họa một ví dụ khác về máy mã hóa hoặc máy giải mã;

Fig.6 là lưu đồ minh họa một phương án của quy trình tinh chế quang học;

Fig.7 là lưu đồ minh họa một phương án khác của quy trình tinh chế quang học;

Fig.8 là lưu đồ minh họa một phương án khác của quy trình tinh chế quang học;

Fig.9 là lưu đồ minh họa một phương án khác của quy trình tinh chế quang học.

Fig.10 là lưu đồ minh họa phương pháp tạo mã video được thực hiện trong thiết bị giải mã hoặc thiết bị mã hóa theo một phương án của sáng chế.

Fig.11 minh họa thiết bị để sử dụng trong bộ mã hóa ảnh và/hoặc bộ giải mã ảnh theo một phương án của sáng chế.

Trong phần sau đây, các ký hiệu chỉ dẫn giống hệt tham chiếu đến các dấu hiệu giống hệt hoặc ít nhất các dấu hiệu tương đương về mặt chức năng nếu không được quy định ngược lại rõ ràng.

Mô tả chi tiết sáng chế

Trong phần mô tả sau đây, việc tham khảo được thực hiện đến các hình vẽ kèm theo, mà tạo thành một phần của bản mô tả, và thể hiện, bằng cách minh họa, các khía cạnh cụ thể của các phương án của sáng chế hoặc các khía cạnh cụ thể trong đó các phương án của sáng chế có thể được sử dụng. Cần hiểu rằng các phương án của sáng chế có thể được sử dụng theo các khía cạnh khác và bao gồm các sự thay đổi về cấu trúc hoặc logic không được thể hiện trên các hình vẽ. Do đó, phần mô tả chi tiết dưới đây không được lấy làm giới hạn, và phạm vi của sáng chế được xác định bởi yêu cầu bảo hộ kèm theo.

Ví dụ, cần hiểu rằng sáng chế có liên quan đến phương pháp được mô tả cũng có thể vẫn là đúng đối với thiết bị hoặc hệ thống tương ứng được tạo cấu hình để thực hiện phương pháp này và ngược lại. Ví dụ, nếu một hoặc nhiều bước phương pháp cụ thể được mô tả, thiết bị tương ứng có thể bao gồm một hoặc nhiều bộ phận, ví dụ các bộ phận chức năng, để thực hiện một hoặc nhiều bước của phương pháp được mô tả (ví dụ một bộ phận thực hiện một hoặc nhiều bước, hoặc nhiều bộ phận, mỗi bộ phận thực hiện một hoặc nhiều hơn một bước trong số các bước đó), ngay cả khi một hoặc nhiều bộ phận này không được mô tả hoặc được minh họa rõ ràng trên các hình vẽ. Mặt khác, ví dụ, nếu thiết bị cụ thể được mô tả dựa trên một hoặc nhiều bộ phận, ví dụ các bộ phận chức năng, phương pháp tương ứng có thể bao gồm một bước để thực hiện chức năng của một hoặc nhiều bộ phận (ví dụ một bước thực hiện chức năng của một hoặc nhiều bộ phận, hoặc nhiều bước, mỗi bước thực hiện chức năng của một hoặc nhiều hơn một bộ phận trong số các bộ phận này), ngay cả khi một hoặc nhiều bước này không được mô tả hoặc được minh họa rõ ràng trên các hình vẽ. Hơn nữa, cần hiểu rằng các dấu hiệu của các phương án và/hoặc các khía cạnh làm ví dụ khác nhau được mô tả ở đây có thể được kết hợp với nhau, trừ khi được lưu ý cụ thể là ngược lại.

Tạo mã video thường liên quan đến việc xử lý chuỗi các hình ảnh, mà tạo thành video hoặc chuỗi video. Thay vì thuật ngữ “hình ảnh”, thuật ngữ “khung” hoặc “ảnh” có thể được sử dụng làm các từ đồng nghĩa trong lĩnh vực tạo mã video. Tạo mã video (hoặc tạo mã nói chung) bao gồm hai phần mã hóa video và giải mã video. Mã hóa video được thực hiện ở phía nguồn, thường bao gồm việc xử lý (ví dụ bằng cách nén) các hình ảnh video gốc để làm giảm lượng dữ liệu cần để đại diện cho các hình ảnh video (để lưu trữ và/hoặc truyền hiệu quả hơn). Giải mã video được thực hiện ở phía đích và thường bao gồm việc xử lý ngược so với bộ mã hóa để tái tạo các hình ảnh video. Các phương án đề cập đến “tạo mã” các hình ảnh video (hoặc các hình ảnh nói chung) sẽ được hiểu là đề cập đến “mã hóa” hoặc “giải mã” các hình ảnh video hoặc các chuỗi video tương ứng. Sự kết hợp của phần mã hóa và phần giải mã cũng được gọi là CODEC (tạo mã và giải mã).

Trong trường hợp tạo mã video không tổn hao, các hình ảnh video gốc có thể được tái tạo, tức là các hình ảnh video được tái tạo có chất lượng giống như các hình ảnh video gốc (giả sử không có tổn hao truyền dẫn hoặc tổn hao dữ liệu khác trong suốt quá trình lưu trữ hoặc truyền). Trong trường hợp tạo mã video tổn hao, việc nén thêm, ví dụ bằng cách lượng tử hóa, được thực hiện, để làm giảm lượng dữ liệu đại diện cho các hình ảnh video, mà không thể được tái tạo hoàn toàn ở bộ giải mã, tức là chất lượng của các hình ảnh video được tái tạo thấp hơn hoặc kém hơn so với chất lượng của các hình ảnh video gốc.

Một số tiêu chuẩn tạo mã video thuộc về nhóm của “các codec video lai tổn hao” (“lossy hybrid video codecs”) (tức là kết hợp dự đoán không gian và thời gian trong miền mẫu và tạo mã biến đổi 2D để áp dụng lượng tử hóa trong miền biến đổi). Mỗi hình ảnh của chuỗi video thường được phân chia thành tập hợp của các khối không chồng chéo và việc tạo mã thường được thực hiện ở mức khối. Nói cách khác, ở bộ mã hóa, video thường được xử lý, tức là được mã hóa, ở mức khối (khối video), ví dụ bằng cách sử dụng dự đoán không gian (trong hình ảnh) và/hoặc dự đoán thời gian (liên hình ảnh) để tạo khối dự đoán, trừ khối dự đoán từ khối hiện hành (khối hiện được xử lý/cần được xử lý) để thu nhận khối dư, biến đổi khối dư và lượng tử hóa khối dư trong miền biến đổi để làm giảm lượng dữ liệu cần được truyền (nén), trong khi đó ở bộ giải mã, việc xử lý ngược so với bộ mã hóa được áp dụng vào khối được mã hóa hoặc được nén

để tái tạo khối hiện hành để biểu diễn. Hơn nữa, bộ mã hóa gấp đôi vòng xử lý của bộ giải mã sao cho cả hai sẽ tạo các sự dự đoán giống hệt nhau (ví dụ các dự đoán trong ảnh và liên ảnh) và/hoặc các sự tái tạo để xử lý, tức là tạo mã, các khối tiếp theo.

Trong các phương án sau đây của hệ thống tạo mã video 10, bộ mã hóa video 20 và bộ giải mã video 30 được mô tả dựa trên các hình vẽ từ Fig.1 đến Fig.3.

Fig.1A là sơ đồ khối giản lược minh họa một hệ thống tạo mã ví dụ 10, ví dụ hệ thống tạo mã video 10 (hoặc gọi tắt là hệ thống tạo mã 10) mà có thể sử dụng các kỹ thuật của sáng chế. Bộ mã hóa video 20 (hoặc gọi tắt là bộ mã hóa 20) và bộ giải mã video 30 (hoặc gọi tắt là bộ giải mã 30) của hệ thống tạo mã video 10 thể hiện các ví dụ về các thiết bị mà có thể được tạo cấu hình để thực hiện các kỹ thuật theo các ví dụ khác nhau được mô tả trong sáng chế.

Như được thể hiện trên Fig.1A, hệ thống tạo mã 10 bao gồm thiết bị nguồn 12 được tạo cấu hình để cung cấp dữ liệu hình ảnh được mã hóa 21, ví dụ, cho thiết bị đích 14 để giải mã dữ liệu hình ảnh được mã hóa 13.

Thiết bị nguồn 12 bao gồm bộ mã hóa 20, và ngoài ra có thể, tức là tùy chọn là, bao gồm nguồn hình ảnh 16, bộ tiền xử lý (hoặc bộ phận tiền xử lý) 18, ví dụ bộ tiền xử lý hình ảnh 18, và giao diện truyền thông hoặc bộ phận truyền thông 22.

Nguồn hình ảnh 16 có thể bao gồm hoặc là loại thiết bị chụp ảnh bất kỳ, ví dụ camera để chụp hình ảnh thế giới thực, và/hoặc loại thiết bị tạo ảnh bất kỳ, ví dụ bộ xử lý đồ họa máy tính để tạo hình ảnh động máy tính, hoặc loại thiết bị khác bất kỳ để thu nhận và/hoặc cung cấp hình ảnh thế giới thực, hình ảnh được tạo bằng máy tính (ví dụ nội dung màn hình, hình ảnh thực tế ảo (virtual reality, VR)) và/hoặc sự kết hợp bất kỳ của chúng (ví dụ hình ảnh thực tế tăng cường (augmented reality, AR)). Nguồn hình ảnh có thể loại bộ nhớ hoặc bộ lưu trữ bất kỳ lưu trữ hình ảnh bất kỳ trong số các hình ảnh nêu trên.

Ngoài bộ tiền xử lý 18 và việc xử lý được thực hiện bởi bộ phận tiền xử lý 18, hình ảnh hoặc dữ liệu hình ảnh 17 cũng có thể được gọi là hình ảnh thô hoặc dữ liệu hình ảnh thô 17.

Bộ tiền xử lý 18 được tạo cấu hình để nhận dữ liệu hình ảnh (thô) 17 và để thực hiện việc tiền xử lý trên dữ liệu hình ảnh 17 để thu nhận hình ảnh được tiền xử lý 19 hoặc dữ liệu hình ảnh được tiền xử lý 19. Việc tiền xử lý được thực hiện bởi bộ tiền xử lý 18 có thể, ví dụ, bao gồm cắt tỉa, chuyển đổi định dạng màu (ví dụ từ RGB sang YCbCr), hiệu chỉnh màu, hoặc khử nhiễu. Có thể hiểu rằng bộ phận tiền xử lý 18 có thể là thành phần tùy chọn.

Bộ mã hóa video 20 được tạo cấu hình để nhận dữ liệu hình ảnh được tiền xử lý 19 và cung cấp dữ liệu hình ảnh được mã hóa 21 (các nội dung chi tiết thêm nữa sẽ được mô tả dưới đây, ví dụ, dựa trên Fig.2).

Giao diện truyền thông 22 của thiết bị nguồn 12 có thể được tạo cấu hình để nhận dữ liệu hình ảnh được mã hóa 21 và để truyền dữ liệu hình ảnh được mã hóa 21 (hoặc phiên bản được xử lý thêm nữa bất kỳ của nó) qua kênh truyền thông 13 đến thiết bị khác, ví dụ thiết bị đích 14 hoặc thiết bị khác bất kỳ, để lưu trữ hoặc tái tạo trực tiếp.

Thiết bị đích 14 bao gồm bộ giải mã 30 (ví dụ bộ giải mã video 30), và ngoài ra có thể, tức là tùy chọn là, bao gồm giao diện truyền thông hoặc bộ phận truyền thông 28, bộ hậu xử lý 32 (hoặc bộ phận hậu xử lý 32) và thiết bị hiển thị 34.

Giao diện truyền thông 28 của thiết bị đích 14 được tạo cấu hình để nhận dữ liệu hình ảnh được mã hóa 21 (hoặc phiên bản được xử lý thêm nữa bất kỳ của nó), ví dụ trực tiếp từ thiết bị nguồn 12 hoặc từ nguồn khác bất kỳ, ví dụ thiết bị lưu trữ, ví dụ thiết bị lưu trữ dữ liệu hình ảnh được mã hóa, và cung cấp dữ liệu hình ảnh được mã hóa 21 cho bộ giải mã 30.

Giao diện truyền thông 22 và giao diện truyền thông 28 có thể được tạo cấu hình để truyền hoặc nhận dữ liệu hình ảnh được mã hóa 21 hoặc dữ liệu được mã hóa 13 qua liên kết truyền thông trực tiếp giữa thiết bị nguồn 12 và thiết bị đích 14, ví dụ kết nối hữu tuyến hoặc vô tuyến trực tiếp, hoặc qua loại mạng bất kỳ, ví dụ mạng hữu tuyến hoặc vô tuyến hoặc sự kết hợp bất kỳ của chúng, hoặc loại mạng riêng và chung bất kỳ, hoặc loại kết hợp bất kỳ của chúng.

Giao diện truyền thông 22 có thể, ví dụ, được tạo cấu hình để đóng gói dữ liệu hình ảnh được mã hóa 21 thành định dạng thích hợp, ví dụ các gói, và/hoặc xử lý dữ liệu hình

ảnh được mã hóa nhờ sử dụng loại bất kỳ trong số mã hóa truyền hoặc xử lý để truyền qua liên kết truyền thông hoặc mạng truyền thông.

Giao diện truyền thông 28, tạo thành phần đối chiếu của giao diện truyền thông 22, có thể, ví dụ, được tạo cấu hình để nhận dữ liệu được truyền và xử lý dữ liệu truyền nhờ sử dụng loại bất kỳ trong số giải mã truyền tương ứng hoặc xử lý và/hoặc giải gói để thu nhận dữ liệu hình ảnh được mã hóa 21.

Cả giao diện truyền thông 22 lẫn giao diện truyền thông 28 có thể được tạo cấu hình như là các giao diện truyền thông đơn hướng như được chỉ báo bởi mũi tên đối với kênh truyền thông 13 trên Fig.1A trở từ thiết bị nguồn 12 đến thiết bị đích 14, hoặc các giao diện truyền thông hai chiều, và có thể được tạo cấu hình, ví dụ để gửi và nhận các tin nhắn, ví dụ để thiết lập kết nối, để báo nhận và trao đổi thông tin khác bất kỳ liên quan đến liên kết truyền thông và/hoặc hoạt động truyền dữ liệu, ví dụ truyền dữ liệu hình ảnh được mã hóa.

Bộ giải mã 30 được tạo cấu hình để nhận dữ liệu hình ảnh được mã hóa 21 và cung cấp dữ liệu hình ảnh được giải mã 31 hoặc hình ảnh được giải mã 31 (các nội dung chi tiết thêm nữa sẽ được mô tả dưới đây, ví dụ, dựa trên Fig.3 hoặc Fig.5).

Bộ hậu xử lý 32 của thiết bị đích 14 được tạo cấu hình để hậu xử lý dữ liệu hình ảnh được giải mã 31 (cũng được gọi là dữ liệu hình ảnh được tái tạo), ví dụ hình ảnh được giải mã 31, để thu nhận dữ liệu hình ảnh được hậu xử lý 33, ví dụ hình ảnh được hậu xử lý 33. Việc hậu xử lý được thực hiện bởi bộ phận hậu xử lý 32 có thể bao gồm, ví dụ, chuyển đổi định dạng màu (ví dụ từ YCbCr sang RGB), hiệu chỉnh màu, cắt tỉa, hoặc lấy mẫu lại, hoặc việc xử lý khác bất kỳ, ví dụ để chuẩn bị dữ liệu hình ảnh được giải mã 31 để hiển thị, ví dụ bởi thiết bị hiển thị 34.

Thiết bị hiển thị 34 của thiết bị đích 14 được tạo cấu hình để nhận dữ liệu hình ảnh được hậu xử lý 33 để hiển thị hình ảnh, ví dụ cho người dùng hoặc người xem. Thiết bị hiển thị 34 có thể là hoặc bao gồm loại bộ hiển thị bất kỳ để đại diện cho hình ảnh được tái tạo, ví dụ bộ hiển thị hoặc màn hình ngoài hoặc tích hợp. Bộ hiển thị có thể, ví dụ, bao gồm các bộ hiển thị tinh thể lỏng (LCD), các bộ hiển thị điốt phát quang hữu cơ (organic light emitting diode, OLED), các bộ hiển thị plasma, các máy chiếu, các bộ

hiển thị vi LED, tinh thể lỏng trên silic (liquid crystal on silicon, LCoS), bộ xử lý ánh sáng số (DLP) hoặc loại bộ hiển thị khác bất kỳ.

Mặc dù Fig.1A thể hiện thiết bị nguồn 12 và thiết bị đích 14 như là các thiết bị riêng rẽ, nhưng các phương án về các thiết bị cũng có thể bao gồm cả hai thiết bị hoặc cả hai chức năng, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng. Trong các phương án này, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng có thể được thực thi nhờ sử dụng cùng một phần cứng và/hoặc phần mềm hoặc bởi phần cứng và/hoặc phần mềm riêng rẽ hoặc sự kết hợp bất kỳ của chúng.

Người có hiểu biết trung bình trong lĩnh vực tương ứng sẽ hiểu rõ dựa trên phần mô tả rằng sự có mặt và chia tách (chính xác) của các chức năng của các bộ phận khác nhau hoặc các chức năng trong thiết bị nguồn 12 và/hoặc thiết bị đích 14 như được thể hiện trên Fig.1A có thể thay đổi phụ thuộc vào thiết bị và ứng dụng thực tế.

Bộ mã hóa 20 (ví dụ, bộ mã hóa video 20) hoặc bộ giải mã 30 (ví dụ, bộ giải mã video 30) hoặc cả bộ mã hóa 20 lẫn bộ giải mã 30 có thể được thực thi qua mạch xử lý như được thể hiện trên Fig.1B, như một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (DSP), mạch tích hợp chuyên dụng (application-specific integrated circuit, ASIC), mảng cổng lập trình được công định trường (field-programmable gate array, FPGA), bộ logic rời rạc, phần cứng, tạo mã video chuyên dụng hoặc các sự kết hợp bất kỳ của chúng. Bộ mã hóa 20 có thể được thực thi qua mạch xử lý 46 để bao gồm các môđun khác nhau như được thảo luận liên quan đến bộ mã hóa 20 trên Fig.2 và/hoặc hệ thống hoặc hệ thống con bộ mã hóa khác bất kỳ được mô tả ở đây. Bộ giải mã 30 có thể được thực thi qua mạch xử lý 46 để bao gồm các môđun khác nhau như được thảo luận liên quan đến bộ giải mã 30 trên Fig.3 và/hoặc hệ thống hoặc hệ thống con bộ giải mã khác bất kỳ được mô tả ở đây. Mạch xử lý có thể được tạo cấu hình để thực hiện các hoạt động khác nhau như được thảo luận sau đây. Như được thể hiện trên Fig.5, nếu các kỹ thuật được thực hiện một phần trong phần mềm, thì thiết bị có thể lưu trữ các lệnh đối với phần mềm trong phương tiện lưu trữ lâu dài được bằng máy tính thích hợp và có thể thực hiện các lệnh trong phần cứng nhờ sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế. Bộ mã hóa video 20 hoặc bộ giải mã video 30 có thể được tích hợp

như là một phần của bộ mã hóa/bộ giải mã kết hợp (CODEC - CODEC) trong một thiết bị đơn lẻ, ví dụ, như được thể hiện trên Fig.1B.

Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm thiết bị bất kỳ trong phạm vi thiết bị rộng, bao gồm thiết bị bất kỳ trong số các thiết bị cầm tay hoặc cố định, ví dụ các máy tính xách tay loại notebook hoặc máy tính xách tay loại laptop, các điện thoại di động, các điện thoại thông minh, các máy tính bảng hoặc các máy tính dạng bảng, các camera, các máy tính để bàn, các thiết bị giải mã tín hiệu và truyền dữ liệu (set-top box), các máy thu hình, các thiết bị hiển thị, các máy phát phương tiện số, các bàn giao tiếp trò chơi video, các thiết bị tạo luồng video (như các máy chủ dịch vụ nội dung hoặc các máy chủ truyền nội dung), thiết bị nhận phát rộng, thiết bị truyền phát rộng, hoặc tương tự và có thể không sử dụng hệ điều hành hoặc sử dụng loại hệ điều hành bất kỳ. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị để truyền thông vô tuyến. Vì vậy, thiết bị nguồn 12 và thiết bị đích 14 có thể là các thiết bị truyền thông vô tuyến.

Trong một số trường hợp, hệ thống tạo mã video 10 được minh họa trên Fig.1A chỉ là ví dụ và các kỹ thuật của sáng chế có thể áp dụng vào các sự thiết lập tạo mã video (ví dụ, mã hóa video hoặc giải mã video) mà không nhất thiết bao gồm việc truyền thông dữ liệu bất kỳ giữa các thiết bị mã hóa và giải mã. Theo các ví dụ khác, dữ liệu được lấy từ bộ nhớ cục bộ, được tạo luồng qua mạng, hoặc tương tự. Thiết bị mã hóa video có thể mã hóa và lưu trữ dữ liệu vào bộ nhớ, và/hoặc thiết bị giải mã video có thể lấy và giải mã dữ liệu từ bộ nhớ. Trong một số ví dụ, việc mã hóa và giải mã được thực hiện bởi các thiết bị mà không truyền thông với một thiết bị khác, mà chỉ đơn giản là mã hóa dữ liệu vào bộ nhớ và/hoặc lấy và giải mã dữ liệu từ bộ nhớ.

Để thuận tiện cho việc mô tả, các phương án của sáng chế được mô tả ở đây, ví dụ, bằng cách tham chiếu đến tạo mã video hiệu suất cao (High-Efficiency Video Coding, HEVC) hoặc đến phần mềm tham chiếu của tạo mã video đa năng (Versatile Video coding, VVC), tiêu chuẩn tạo mã video thế hệ tiếp theo được phát triển bởi nhóm hợp tác chung về tạo mã video (Joint Collaboration Team on Video Coding, JCT-VC) của nhóm chuyên gia tạo mã video (Video Coding Experts Group, VCEG) ITU-T và nhóm chuyên gia hình ảnh động (Motion Picture Experts Group, MPEG) ISO/IEC. Người có

hiệu biết trung bình trong lĩnh vực tương ứng sẽ hiểu rằng các phương án của sáng chế không bị giới hạn ở HEVC hoặc VVC.

Bộ mã hóa và phương pháp mã hóa

Fig.2 thể hiện sơ đồ khối giản lược của bộ mã hóa video ví dụ 20 mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Trong ví dụ trên Fig.2, bộ mã hóa video 20 bao gồm đầu vào 201 (hoặc giao diện đầu vào 201), bộ phận tính toán dư 204, bộ phận xử lý biến đổi 206, bộ phận lượng tử hóa 208, bộ phận lượng tử hóa ngược 210, và bộ phận xử lý biến đổi ngược 212, bộ phận tái tạo 214, bộ phận lọc vòng 220, bộ đệm hình ảnh được giải mã (decoded picture buffer, DPB) 230, bộ phận chọn chế độ 260, bộ phận mã hóa entropi 270 và đầu ra 272 (hoặc giao diện đầu ra 272). Bộ phận chọn chế độ 260 có thể bao gồm bộ phận dự đoán liên ảnh 244, bộ phận dự đoán trong ảnh 254 và bộ phận phân chia 262. Bộ phận dự đoán liên ảnh 244 có thể bao gồm bộ phận ước lượng chuyển động và bộ phận bù chuyển động (không được thể hiện trên hình vẽ). Bộ mã hóa video 20 như được thể hiện trên Fig.2 cũng có thể được gọi là bộ mã hóa video lai hoặc bộ mã hóa video theo codec video lai.

Bộ phận tính toán dư 204, bộ phận xử lý biến đổi 206, bộ phận lượng tử hóa 208, bộ phận chọn chế độ 260 có thể được coi là tạo thành đường tín hiệu xuôi của bộ mã hóa 20, trong khi đó bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận tái tạo 214, bộ đệm 216, bộ lọc vòng 220, bộ đệm hình ảnh được giải mã (decoded picture buffer, DPB) 230, bộ phận dự đoán liên ảnh 244 và bộ phận dự đoán trong ảnh 254 có thể được coi là tạo thành đường tín hiệu ngược của bộ mã hóa video 20, trong đó đường tín hiệu ngược của bộ mã hóa video 20 tương ứng với đường tín hiệu của bộ giải mã (xem bộ giải mã video 30 trên Fig.3). Bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận tái tạo 214, bộ lọc vòng 220, bộ đệm hình ảnh được giải mã (DPB) 230, bộ phận dự đoán liên ảnh 244 và bộ phận dự đoán trong ảnh 254 cũng được coi là tạo thành “bộ giải mã lắp sẵn” của bộ mã hóa video 20.

Các hình ảnh & phân chia hình ảnh (các hình ảnh & các khối)

Bộ mã hóa 20 có thể được tạo cấu hình để nhận, ví dụ qua đầu vào 201, hình ảnh 17 (hoặc dữ liệu hình ảnh 17), ví dụ hình ảnh của chuỗi các hình ảnh tạo thành video hoặc chuỗi video. Hình ảnh hoặc dữ liệu hình ảnh nhận được cũng có thể là hình ảnh

được tiền xử lý 19 (hoặc dữ liệu hình ảnh được tiền xử lý 19). Để đơn giản, phần mô tả sau đây liên quan đến hình ảnh 17. Hình ảnh 17 cũng có thể được gọi là hình ảnh hiện hành hoặc hình ảnh cần được tạo mã (cụ thể trong tạo mã video để phân biệt hình ảnh hiện hành với các hình ảnh khác, ví dụ các hình ảnh được mã hóa và/hoặc được giải mã trước đó của cùng chuỗi video, tức là chuỗi video mà cũng bao gồm hình ảnh hiện hành).

Hình ảnh (số) được hoặc có thể được coi là mảng hoặc ma trận hai chiều của các mẫu với các giá trị cường độ. Mẫu trong mảng cũng có thể được gọi là điểm ảnh (pixel) (dạng rút gọn của phần tử hình ảnh) hoặc pel (điểm ảnh). Số lượng mẫu theo chiều (hoặc trục) ngang và thẳng đứng của mảng hoặc hình ảnh xác định kích thước và/hoặc độ phân giải của hình ảnh. Để biểu diễn màu, thông thường ba thành phần màu được sử dụng, tức là hình ảnh có thể được biểu diễn hoặc bao gồm ba mảng mẫu. Theo định dạng hoặc không gian màu RGB, hình ảnh bao gồm mảng mẫu đỏ, lục, lam tương ứng. Tuy nhiên, trong tạo mã video, mỗi điểm ảnh thường được biểu diễn theo định dạng hoặc không gian màu độ chói và sắc độ, ví dụ YCbCr, mà bao gồm thành phần độ chói được chỉ báo bởi Y (đôi lúc L được sử dụng để thay thế) và hai thành phần sắc độ được chỉ báo bởi Cb và Cr. Thành phần độ chói (luminance) (hoặc gọi tắt là luma) Y đại diện cho độ sáng hoặc cường độ độ xám (ví dụ như trong hình ảnh thang màu xám), trong khi hai thành phần sắc độ (chrominance) (hoặc gọi tắt là chroma) Cb và Cr đại diện cho sắc độ hoặc các thành phần thông tin màu. Theo đó, hình ảnh theo định dạng YCbCr bao gồm mảng mẫu độ chói của các giá trị mẫu độ chói (Y), và hai mảng mẫu sắc độ của các giá trị sắc độ (Cb và Cr). Các hình ảnh ở định dạng RGB có thể được chuyển đổi hoặc được biến đổi thành định dạng YCbCr và ngược lại, quy trình này cũng được biết đến là biến đổi hoặc chuyển đổi màu. Nếu hình ảnh là đơn sắc, thì hình ảnh này có thể bao gồm chỉ mảng mẫu độ chói. Theo đó, hình ảnh có thể là, ví dụ, mảng của các mẫu độ chói theo định dạng đơn sắc hoặc mảng của các mẫu độ chói và hai mảng tương ứng của các mẫu sắc độ theo định dạng màu 4:2:0, 4:2:2, và 4:4:4.

Các phương án của bộ mã hóa video 20 có thể bao gồm bộ phận phân chia hình ảnh (không được thể hiện trên Fig.2) được tạo cấu hình để phân chia hình ảnh 17 thành các khối hình ảnh 203 (thường là không chồng chéo). Các khối này cũng có thể được gọi là các khối gốc, các khối macrô (H.264/AVC) hoặc các khối cây tạo mã (CTB) hoặc các đơn vị cây tạo mã (CTU) (H.265/HEVC và VVC). Bộ phận phân chia hình ảnh có

thể được tạo cấu hình để sử dụng cùng kích thước khối đối với tất cả các hình ảnh của chuỗi video và lưới tương ứng xác định kích thước khối, hoặc để thay đổi kích thước khối giữa các hình ảnh hoặc các tập con hoặc các nhóm của các hình ảnh, và phân chia mỗi hình ảnh thành các khối tương ứng.

Theo các phương án thêm nữa, bộ mã hóa video có thể được tạo cấu hình để nhận trực tiếp khối 203 của hình ảnh 17, ví dụ một, một số hoặc tất cả các khối tạo thành hình ảnh 17. Khối hình ảnh 203 cũng có thể được gọi là khối hình ảnh hiện hành hoặc khối hình ảnh cần được tạo mã.

Giống như hình ảnh 17, khối hình ảnh 203 mặt khác là hoặc có thể được coi là mảng hoặc ma trận hai chiều của các mẫu với các giá trị cường độ (các giá trị mẫu), mặc dù kích thước nhỏ hơn hình ảnh 17. Nói cách khác, khối 203 có thể bao gồm, ví dụ, một mảng mẫu (ví dụ, mảng độ chói trong trường hợp hình ảnh đơn sắc 17, hoặc mảng độ chói hoặc sắc độ trong trường hợp hình ảnh màu) hoặc ba mảng mẫu (ví dụ một mảng độ chói và hai mảng sắc độ trong trường hợp hình ảnh màu 17) hoặc số lượng và/hoặc loại mảng khác bất kỳ phụ thuộc vào định dạng màu được áp dụng. Số lượng mẫu theo chiều (hoặc trục) ngang và thẳng đứng của khối 203 xác định kích thước của khối 203. Theo đó, khối có thể là, ví dụ, mảng $M \times N$ (M -cột nhân với N -hàng) của các mẫu, hoặc mảng $M \times N$ của các hệ số biến đổi.

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể được tạo cấu hình để mã hóa hình ảnh 17 theo từng khối, ví dụ việc mã hóa và dự đoán được thực hiện trên từng khối 203.

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân chia và/hoặc mã hóa hình ảnh bằng cách sử dụng các lát (slice) (cũng được gọi là các lát video), trong đó hình ảnh có thể được phân chia thành hoặc được mã hóa nhờ sử dụng một hoặc nhiều lát (thường là không chồng chéo), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU).

Các phương án của bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể còn được tạo cấu hình để phân chia và/hoặc mã hóa hình ảnh bằng cách sử dụng các nhóm ảnh xếp lớp (tile) (cũng được gọi là các nhóm ảnh xếp lớp video) và/hoặc các ảnh xếp lớp (cũng được gọi là các ảnh xếp lớp video), trong đó hình ảnh có thể được phân chia

thành hoặc được mã hóa nhờ sử dụng một hoặc nhiều nhóm ảnh xếp lớp (thường là không chồng chéo), và mỗi nhóm ảnh xếp lớp có thể bao gồm, ví dụ một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều ảnh xếp lớp, trong đó mỗi ảnh xếp lớp, ví dụ có thể là hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ các khối hoàn chỉnh hoặc các khối thành phần.

Tính toán dư

Bộ phận tính toán dư 204 có thể được tạo cấu hình để tính toán khối dư 205 (cũng được gọi là phần dư 205) dựa trên khối hình ảnh 203 và khối dự đoán 265 (các nội dung chi tiết thêm nữa về khối dự đoán 265 được đưa ra sau đây), ví dụ bằng cách trừ các giá trị mẫu của khối dự đoán 265 từ các giá trị mẫu của khối hình ảnh 203, theo từng mẫu (theo từng điểm ảnh) để thu nhận khối dư 205 trong miền mẫu.

Biến đổi

Bộ phận xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng biến đổi, ví dụ biến đổi cosin rời rạc (discrete cosine transform, DCT) hoặc biến đổi sin rời rạc (discrete sine transform, DST), trên các giá trị mẫu của khối dư 205 để thu nhận các hệ số biến đổi 207 trong miền biến đổi. Các hệ số biến đổi 207 cũng có thể được gọi là các hệ số dư biến đổi và đại diện cho khối dư 205 trong miền biến đổi.

Bộ phận xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng các phép xấp xỉ nguyên của DCT/DST, như các phép biến đổi được quy định đối với H.265/HEVC. So với biến đổi DCT trực giao, các phép xấp xỉ nguyên này thường được định tỷ lệ bởi hệ số nhất định. Để giữ chuẩn của khối dư mà được xử lý bằng các loại biến đổi thuận và ngược, các bội suất bổ sung được áp dụng như là một phần của quy trình biến đổi. Các bội suất thường được chọn dựa trên các sự ràng buộc nhất định như các bội suất là lũy thừa của hai đối với các thao tác dịch, độ sâu bit của các hệ số biến đổi, sự cân bằng giữa độ chính xác và các chi phí thực hiện, v.v.. Các bội suất cụ thể, ví dụ, được quy định đối với biến đổi ngược, ví dụ bởi bộ phận xử lý biến đổi ngược 212 (và việc biến đổi ngược tương ứng, ví dụ bởi bộ phận xử lý biến đổi ngược 312 ở bộ giải mã video 30) và các bội suất tương ứng đối với biến đổi thuận, ví dụ bởi bộ phận xử lý biến đổi 206, ở bộ mã hóa 20 có thể được quy định theo đó.

Các phương án của bộ mã hóa video 20 (tương ứng là bộ phận xử lý biến đổi 206) có thể được tạo cấu hình để kết xuất các thông số biến đổi, ví dụ loại của biến đổi hoặc các biến đổi, ví dụ trực tiếp hoặc được mã hóa hoặc được nén qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số biến đổi để giải mã.

Lượng tử hóa

Bộ phận lượng tử hóa 208 có thể được tạo cấu hình để lượng tử hóa các hệ số biến đổi 207 để thu nhận các hệ số được lượng tử hóa 209, ví dụ bằng cách áp dụng lượng tử hóa vô hướng hoặc lượng tử hóa vectơ. Các hệ số được lượng tử hóa 209 cũng có thể được gọi là các hệ số biến đổi được lượng tử hóa 209 hoặc các hệ số dư được lượng tử hóa 209.

Quy trình lượng tử hóa có thể làm giảm độ sâu bit được liên kết với một số hoặc tất cả các hệ số biến đổi 207. Ví dụ, hệ số biến đổi n bit có thể được làm tròn xuống thành hệ số biến đổi m bit trong suốt quá trình lượng tử hóa, trong đó n lớn hơn m . Mức độ lượng tử hóa có thể được thay đổi bằng cách điều chỉnh thông số lượng tử hóa (QP). Ví dụ, đối với lượng tử hóa vô hướng, việc định tỷ lệ khác nhau có thể được áp dụng để đạt được lượng tử hóa tinh hơn hoặc thô hơn. Các kích thước bước lượng tử hóa nhỏ hơn tương ứng với lượng tử hóa tinh hơn, trong khi đó các kích thước bước lượng tử hóa lớn hơn tương ứng với lượng tử hóa thô hơn. Kích thước bước lượng tử hóa có thể áp dụng được có thể được chỉ báo bởi thông số lượng tử hóa (quantization parameter, QP). Thông số lượng tử hóa có thể, ví dụ, là chỉ số đối với tập hợp được xác định trước của các kích thước bước lượng tử hóa áp dụng được. Ví dụ, các thông số lượng tử hóa nhỏ có thể tương ứng với lượng tử hóa tinh (các kích thước bước lượng tử hóa nhỏ) và các thông số lượng tử hóa lớn có thể tương ứng với lượng tử hóa thô (các kích thước bước lượng tử hóa lớn) hoặc ngược lại. Việc lượng tử hóa có thể bao gồm việc chia bởi kích thước bước lượng tử hóa và tương ứng và/hoặc khử lượng tử hóa ngược, ví dụ bởi bộ phận lượng tử hóa ngược 210, có thể bao gồm việc nhân với kích thước bước lượng tử hóa. Các phương án theo một số tiêu chuẩn, ví dụ HEVC, có thể được tạo cấu hình để sử dụng thông số lượng tử hóa để xác định kích thước bước lượng tử hóa. Nói chung, kích thước bước lượng tử hóa có thể được tính toán dựa trên thông số lượng tử hóa nhờ

sử dụng phép xấp xỉ dấu phẩy tĩnh của phương trình bao gồm phép chia. Các bội suất bổ sung có thể được đưa vào lượng tử hóa và khử lượng tử hóa để phục hồi chuẩn của khối dư, mà có thể được thay đổi do việc định tỷ lệ được sử dụng trong phép xấp xỉ dấu phẩy tĩnh của phương trình đối với kích thước bước lượng tử hóa và thông số lượng tử hóa. Theo một cách thực hiện ví dụ, việc định tỷ lệ của biến đổi ngược và khử lượng tử hóa có thể được kết hợp. Theo cách khác, các bảng lượng tử hóa được tùy biến có thể được sử dụng và được báo hiệu từ bộ mã hóa đến bộ giải mã, ví dụ trong dòng bit. Lượng tử hóa là hoạt động có tổn hao, trong đó tổn hao tăng lên cùng với việc tăng các kích thước bước lượng tử hóa.

Các phương án của bộ mã hóa video 20 (tương ứng là bộ phận lượng tử hóa 208) có thể được tạo cấu hình để kết xuất các thông số lượng tử hóa (QP), ví dụ trực tiếp hoặc được mã hóa qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã video 30 có thể nhận và áp dụng các thông số lượng tử hóa để giải mã.

Lượng tử hóa ngược

Bộ phận lượng tử hóa ngược 210 được tạo cấu hình để áp dụng lượng tử hóa ngược của bộ phận lượng tử hóa 208 trên các hệ số được lượng tử hóa để thu nhận các hệ số được khử lượng tử hóa 211, ví dụ bằng cách áp dụng sự đảo ngược của sơ đồ lượng tử hóa được áp dụng bởi bộ phận lượng tử hóa 208 dựa trên hoặc sử dụng kích thước bước lượng tử hóa giống như bộ phận lượng tử hóa 208. Các hệ số được khử lượng tử hóa 211 cũng có thể được gọi là các hệ số dư được khử lượng tử hóa 211 và tương ứng - mặc dù thường không giống hệt các hệ số biến đổi do tổn hao bởi lượng tử hóa - với các hệ số biến đổi 207.

Biến đổi ngược

Bộ phận xử lý biến đổi ngược 212 được tạo cấu hình để áp dụng biến đổi ngược của việc biến đổi được áp dụng bởi bộ phận xử lý biến đổi 206, ví dụ biến đổi cosin rời rạc (DCT) ngược hoặc biến đổi sin rời rạc (DST) ngược hoặc các phép biến đổi ngược khác, để thu nhận khối dư được tái tạo 213 (hoặc các hệ số được khử lượng tử hóa tương ứng 213) trong miền mẫu. Khối dư được tái tạo 213 cũng có thể được gọi là khối biến đổi 213.

Tái tạo

Bộ phận tái tạo 214 (ví dụ bộ tổng hoặc bộ cộng 214) được tạo cấu hình để cộng khối biến đổi 213 (tức là khối dư được tái tạo 213) vào khối dự đoán 265 để thu nhận khối được tái tạo 215 trong miền mẫu, ví dụ bằng cách cộng – theo từng mẫu - các giá trị mẫu của khối dư được tái tạo 213 và các giá trị mẫu của khối dự đoán 265.

Lọc

Bộ phận lọc vòng 220 (hoặc gọi tắt là “bộ lọc vòng” 220), được tạo cấu hình để lọc khối được tái tạo 215 để thu nhận khối được lọc 221, hoặc nói chung, để lọc các mẫu tái tạo để thu nhận các mẫu được lọc. Bộ phận lọc vòng, ví dụ, được tạo cấu hình để làm phẳng các sự chuyển tiếp điểm ảnh, hoặc theo cách khác là nâng cao chất lượng video. Bộ phận lọc vòng 220 có thể bao gồm một hoặc nhiều bộ lọc vòng như bộ lọc giải khối, bộ lọc độ lệch thích ứng mẫu (sample-adaptive offset, SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ bộ lọc hai chiều, bộ lọc vòng thích ứng (adaptive loop filter, ALF), các bộ lọc làm nhọn, các bộ lọc làm phẳng hoặc các bộ lọc cộng tác, hoặc sự kết hợp bất kỳ của chúng. Mặc dù bộ phận lọc vòng 220 được thể hiện trên Fig.2 như là bộ lọc nội vòng (in loop filter), nhưng theo các cấu hình khác, bộ phận lọc vòng 220 có thể được thực hiện như là bộ lọc sau vòng. Khối được lọc 221 cũng có thể được gọi là khối được tái tạo được lọc 221.

Các phương án của bộ mã hóa video 20 (tương ứng là bộ phận lọc vòng 220) có thể được tạo cấu hình để kết xuất các thông số lọc vòng (như thông tin độ lệch tương thích mẫu), ví dụ trực tiếp hoặc được mã hóa qua bộ phận mã hóa entropi 270, sao cho, ví dụ, bộ giải mã 30 có thể nhận và áp dụng cùng các thông số lọc vòng hoặc các bộ lọc vòng tương ứng để giải mã.

Bộ đệm hình ảnh được giải mã

Bộ đệm hình ảnh được giải mã (DPB) 230 có thể là bộ nhớ để lưu trữ các hình ảnh tham chiếu, hoặc nói chung là dữ liệu hình ảnh tham chiếu, để mã hóa dữ liệu video bằng bộ mã hóa video 20. DPB 230 có thể được tạo ra bởi loại bất kỳ trong số rất nhiều loại thiết bị nhớ, như bộ nhớ truy cập ngẫu nhiên động (dynamic random access memory, DRAM), bao gồm DRAM đồng bộ (synchronous DRAM, SDRAM), RAM từ điện trở

(magnetoresistive RAM, MRAM), RAM điện trở (resistive RAM, RRAM), hoặc các loại thiết bị nhớ khác. Bộ đệm hình ảnh được giải mã (DPB) 230 có thể được tạo cấu hình để lưu trữ một hoặc nhiều khối được lọc 221. Bộ đệm hình ảnh được giải mã 230 có thể còn được tạo cấu hình để lưu trữ các khối được lọc trước đó khác, ví dụ các khối được tái tạo và được lọc trước đó 221, của cùng hình ảnh hiện hành hoặc của các hình ảnh khác, ví dụ các hình ảnh được tái tạo trước đó, và có thể cung cấp các hình ảnh được tái tạo, tức là được giải mã, trước đó hoàn chỉnh (và các khối và các mẫu tham chiếu tương ứng) và/hoặc hình ảnh hiện hành được tái tạo một phần (và các khối và các mẫu tham chiếu tương ứng), ví dụ để dự đoán liên ảnh. Bộ đệm hình ảnh được giải mã (DPB) 230 cũng có thể được tạo cấu hình để lưu trữ một hoặc nhiều khối được tái tạo không được lọc 215, hoặc nói chung là các mẫu tái tạo không được lọc, ví dụ nếu khối được tái tạo 215 không được lọc bởi bộ phận lọc vòng 220, hoặc phiên bản được xử lý thêm nữa khác bất kỳ của các khối hoặc các mẫu được tái tạo.

Chọn chế độ (phân chia & dự đoán)

Bộ phận chọn chế độ 260 bao gồm bộ phận phân chia 262, bộ phận dự đoán liên ảnh 244 và bộ phận dự đoán trong ảnh 254, và được tạo cấu hình để nhận hoặc thu nhận dữ liệu hình ảnh gốc, ví dụ khối gốc 203 (khối hiện hành 203 của hình ảnh hiện hành 17), và dữ liệu hình ảnh được tái tạo, ví dụ các mẫu hoặc các khối tái tạo được lọc và/hoặc không được lọc của cùng hình ảnh (hiện hành) và/hoặc từ một hoặc nhiều hình ảnh được giải mã trước đó, ví dụ từ bộ đệm hình ảnh được giải mã 230 hoặc các bộ đệm khác (ví dụ bộ đệm hàng, không được thể hiện trên hình vẽ). Dữ liệu hình ảnh được tái tạo được sử dụng làm dữ liệu hình ảnh tham chiếu để dự đoán, ví dụ dự đoán liên ảnh hoặc dự đoán trong ảnh, để thu nhận khối dự đoán 265 hoặc bộ dự đoán 265.

Bộ phận chọn chế độ 260 có thể được tạo cấu hình để xác định hoặc chọn sự phân chia đối với chế độ dự đoán khối hiện hành (bao gồm không phân chia) và chế độ dự đoán (ví dụ chế độ dự đoán trong ảnh hoặc liên ảnh) và tạo khối dự đoán tương ứng 265, mà được sử dụng để tính toán khối dư 205 và để tái tạo khối được tái tạo 215.

Các phương án của bộ phận chọn chế độ 260 có thể được tạo cấu hình để chọn sự phân chia và chế độ dự đoán (ví dụ từ sự phân chia và chế độ dự đoán được hỗ trợ bởi hoặc khả dụng đối với bộ phận chọn chế độ 260), mà cung cấp sự phù hợp nhất hoặc

nói cách khác lượng dư tối thiểu (lượng dư tối thiểu có nghĩa là việc nén tốt hơn để truyền hoặc lưu trữ), hoặc lượng thông tin báo hiệu thừa tối thiểu (lượng thông tin báo hiệu thừa tối thiểu có nghĩa là việc nén tốt hơn để truyền hoặc lưu trữ), hoặc cân nhắc hoặc cân đối cả hai. Bộ phận chọn chế độ 260 có thể được tạo cấu hình để xác định sự phân chia và chế độ dự đoán dựa trên tối ưu hóa tỷ lệ méo (rate distortion optimization, RDO), tức là chọn chế độ dự đoán mà cung cấp tỷ lệ méo tối thiểu. Các thuật ngữ như “tốt nhất”, “tối thiểu”, “tối ưu” v.v. trong ngữ cảnh này không nhất thiết phải đề cập đến “tốt nhất”, “tối thiểu”, “tối ưu”, v.v. toàn phần mà cũng có thể đề cập đến sự hoàn thành tiêu chuẩn chấm dứt hoặc chọn giống như giá trị vượt quá hoặc nhỏ hơn ngưỡng hoặc các ràng buộc khác có tiềm năng dẫn đến “việc chọn tối ưu phụ” nhưng làm giảm độ phức tạp và thời gian xử lý.

Nói cách khác, bộ phận phân chia 262 có thể được tạo cấu hình để phân chia khối 203 thành các phân chia khối nhỏ hơn hoặc các khối con (mà lại tạo thành các khối), ví dụ sử dụng lặp lại phân chia cây tứ phân (QT), phân chia nhị phân (BT) hoặc phân chia cây bội ba (TT) hoặc sự kết hợp bất kỳ của chúng, và để thực hiện, ví dụ, dự đoán đối với mỗi trong số các phân chia khối hoặc các khối con, trong đó việc chọn chế độ bao gồm việc chọn các cấu trúc cây của khối được phân chia 203 và các chế độ dự đoán được áp dụng vào mỗi trong số các phân chia khối hoặc các khối con.

Trong phần sau đây, sự phân chia (ví dụ bởi bộ phận phân chia 260) và xử lý dự đoán (bởi bộ phận dự đoán liên ảnh 244 và bộ phận dự đoán trong ảnh 254) được thực hiện bởi bộ mã hóa video 20 làm ví dụ sẽ được giải thích chi tiết hơn.

Phân chia

Bộ phận phân chia 262 có thể phân chia (hoặc chia tách) khối hiện hành 203 thành các phân chia nhỏ hơn, ví dụ các khối nhỏ hơn của kích thước hình vuông hoặc hình chữ nhật. Các khối nhỏ hơn này (mà cũng có thể được gọi là các khối con) có thể còn được phân chia thành thậm chí các phân chia nhỏ hơn. Điều này cũng liên quan đến phân chia cây hoặc phân chia cây phân cấp, trong đó khối gốc, ví dụ ở mức cây gốc 0 (mức phân cấp 0, độ sâu 0), có thể được phân chia theo đệ quy, ví dụ được phân chia thành hai hoặc nhiều hơn hai khối của mức cây thấp hơn, ví dụ các nút ở mức cây 1 (mức phân cấp 1, độ sâu 1), trong đó các khối này có thể được phân chia lại thành hai

hoặc nhiều hơn hai khối của mức thấp hơn tiếp theo, ví dụ mức cây 2 (mức phân cấp 2, độ sâu 2), v.v. cho đến khi việc phân chia được chấm dứt, ví dụ do tiêu chuẩn chấm dứt được đáp ứng, ví dụ độ sâu cây tối đa hoặc kích thước khối tối thiểu được đạt đến. Các khối mà không được phân chia thêm nữa cũng được gọi là các khối lá hoặc các nút lá của cây. Cây sử dụng việc phân chia thành hai phần chia được gọi là cây nhị phân (binary-tree, BT), cây sử dụng việc phân chia thành ba phần chia được gọi là cây tam phân (TT), và cây sử dụng việc phân chia thành bốn phần chia được gọi là cây tứ phân (QT).

Như được nêu trên đây, thuật ngữ “khối” như được sử dụng ở đây có thể là một phần, cụ thể là phần hình vuông hoặc hình chữ nhật, của hình ảnh. Dựa vào, ví dụ, HEVC và VVC, khối có thể là hoặc tương ứng với đơn vị cây tạo mã (CTU), đơn vị tạo mã (CU), đơn vị dự đoán (PU), và đơn vị biến đổi (TU) và/hoặc các khối tương ứng, ví dụ khối cây tạo mã (CTB), khối tạo mã (CB), khối biến đổi (TB) hoặc khối dự đoán (PB).

Ví dụ, đơn vị cây tạo mã (CTU) có thể là hoặc bao gồm CTB của các mẫu độ chói, hai CTB tương ứng của các mẫu sắc độ của hình ảnh mà có ba mảng mẫu, hoặc CTB của các mẫu của hình ảnh đơn sắc hoặc hình ảnh mà được tạo mã nhờ sử dụng ba mặt phẳng màu riêng rẽ và các cấu trúc cú pháp được sử dụng để tạo mã các mẫu. Tương ứng, khối cây tạo mã (CTB) có thể là khối $N \times N$ của các mẫu đối với một số giá trị của N sao cho việc chia một thành phần thành các CTB là việc phân chia. Đơn vị tạo mã (CU) có thể là hoặc bao gồm khối tạo mã của các mẫu độ chói, hai khối tạo mã tương ứng của các mẫu sắc độ của hình ảnh có ba mảng mẫu, hoặc khối tạo mã của các mẫu của hình ảnh đơn sắc hoặc hình ảnh mà được tạo mã sử dụng ba mặt phẳng màu riêng rẽ và các cấu trúc cú pháp được sử dụng để tạo mã các mẫu. Tương ứng, khối tạo mã (CB) có thể là khối $M \times N$ của các mẫu đối với một số giá trị của M và N sao cho việc chia CTB thành các khối tạo mã là việc phân chia.

Trong các phương án, ví dụ, theo HEVC, đơn vị cây tạo mã (CTU) có thể được tách thành các CU bằng cách sử dụng cấu trúc cây tứ phân được biểu thị là cây mã hóa. Quyết định liệu có tạo mã khu vực ảnh sử dụng dự đoán liên hình ảnh (thời gian) hoặc trong hình ảnh (không gian) hay không được thực hiện ở mức CU. Mỗi CU có thể còn

được chia thành một, hai hoặc bốn PU theo loại chia tách PU. Bên trong một PU, cùng quy trình dự đoán được áp dụng và thông tin liên quan được truyền đến bộ giải mã trên cơ sở PU. Sau khi thu nhận khối dư bằng cách áp dụng quy trình dự đoán dựa trên loại chia tách PU, CU có thể được phân chia thành các đơn vị biến đổi (các TU) theo cấu trúc cây tứ phân khác tương tự như cây tạo mã đối với CU.

Trong các phương án, ví dụ, theo tiêu chuẩn tạo mã video mới nhất hiện nay theo hướng phát triển, mà được gọi là tạo mã video đa năng (VVC), việc phân chia cây tứ phân và cây nhị phân kết hợp (QTBT), ví dụ, được sử dụng để phân chia khối tạo mã. Trong cấu trúc khối QTBT, CU có thể có hình dạng vuông hoặc chữ nhật. Ví dụ, đơn vị cây tạo mã (coding tree unit, CTU) trước tiên được phân chia bởi cấu trúc cây tứ phân. Các nút lá cây tứ phân còn được phân chia bởi cấu trúc cây nhị phân hoặc cấu trúc cây tam phân (hoặc bội ba). Các nút lá cây phân chia được gọi là các đơn vị tạo mã (các CU), và phân đoạn đó được sử dụng để xử lý dự đoán và biến đổi mà không cần việc phân chia thêm nữa bất kỳ. Điều này có nghĩa là CU, PU và TU có cùng kích thước khối trong cấu trúc khối tạo mã QTBT. Song song với đó, đa phân chia, ví dụ, phân chia cây bội ba có thể được sử dụng cùng với cấu trúc khối QTBT.

Theo một ví dụ, bộ phận chọn chế độ 260 của bộ mã hóa video 20 có thể được tạo cấu hình để thực hiện sự kết hợp bất kỳ của các kỹ thuật phân chia được mô tả ở đây.

Như được mô tả trên đây, bộ mã hóa video 20 được tạo cấu hình để xác định hoặc chọn chế độ dự đoán tốt nhất hoặc tối ưu từ tập hợp của các chế độ dự đoán (ví dụ định trước). Tập hợp của các chế độ dự đoán có thể bao gồm, ví dụ, các chế độ dự đoán trong ảnh và/hoặc các chế độ dự đoán liên ảnh.

Dự đoán trong ảnh

Tập hợp các chế độ dự đoán trong ảnh có thể bao gồm 35 các chế độ dự đoán trong ảnh khác nhau, ví dụ các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ có hướng, ví dụ như được định nghĩa trong HEVC, hoặc có thể bao gồm 67 chế độ dự đoán trong ảnh khác nhau, ví dụ các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ có hướng, ví dụ như được định nghĩa đối với VVC.

Bộ phận dự đoán trong ảnh 254 được tạo cấu hình để sử dụng các mẫu tái tạo của các khối lân cận của cùng hình ảnh hiện hành để tạo khối dự đoán trong ảnh 265 theo chế độ dự đoán trong ảnh của tập hợp các chế độ dự đoán trong ảnh.

Bộ phận dự đoán trong ảnh 254 (hoặc nói chung là bộ phận chọn chế độ 260) còn được tạo cấu hình để kết xuất các thông số dự đoán trong ảnh (hoặc nói chung là thông tin chỉ báo chế độ dự đoán trong ảnh được chọn đối với khối) đến bộ phận mã hóa entropi 270 ở dạng các phần tử cú pháp 266 để bao gồm vào dữ liệu hình ảnh được mã hóa 21, sao cho, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số dự đoán để giải mã.

Dự đoán liên ảnh

Tập hợp các chế độ dự đoán liên ảnh (hoặc khả thi) phụ thuộc vào các hình ảnh tham chiếu khả dụng (tức là các hình ảnh được giải mã ít nhất một phần trước đó, ví dụ được lưu trữ trong DBP 230) và các thông số dự đoán liên ảnh khác, ví dụ liệu toàn bộ hình ảnh tham chiếu hoặc chỉ một phần, ví dụ khu vực cửa sổ tìm kiếm quanh khu vực của khối hiện hành, của hình ảnh tham chiếu được sử dụng để tìm kiếm khối tham chiếu phù hợp nhất, và/hoặc ví dụ liệu việc nội suy điểm ảnh có được áp dụng, ví dụ nội suy nửa/bán điểm ảnh và/hoặc một phần tư điểm ảnh, hay không.

Ngoài các chế độ dự đoán trên đây, chế độ bỏ qua và/hoặc chế độ trực tiếp có thể được áp dụng.

Bộ phận dự đoán liên ảnh 244 có thể bao gồm bộ phận ước lượng chuyển động (motion estimation, ME) và bộ phận bù chuyển động (motion compensation, MC) (đều không được thể hiện trên Fig.2). Bộ phận ước lượng chuyển động có thể được tạo cấu hình để nhận hoặc thu nhận khối hình ảnh 203 (khối hình ảnh hiện hành 203 của hình ảnh hiện hành 17) và hình ảnh được giải mã 231, hoặc ít nhất một hoặc nhiều khối được tái tạo trước đó, ví dụ các khối được tái tạo của một hoặc nhiều hình ảnh được giải mã trước đó 231 khác/khác nhau, để ước lượng chuyển động. Ví dụ, chuỗi video có thể bao gồm hình ảnh hiện hành và các hình ảnh được giải mã trước đó 231, hoặc nói cách khác, hình ảnh hiện hành và các hình ảnh được giải mã trước đó 231 có thể là một phần của hoặc tạo thành chuỗi các hình ảnh tạo thành chuỗi video.

Bộ mã hóa 20 có thể, ví dụ, được tạo cấu hình để chọn khối tham chiếu từ các khối tham chiếu của các hình ảnh giống hoặc khác nhau của các hình ảnh khác và cung cấp hình ảnh tham chiếu (hoặc chỉ số hình ảnh tham chiếu) và/hoặc độ lệch (độ lệch không gian) giữa vị trí (các tọa độ x, y) của khối tham chiếu và vị trí của khối hiện hành như là các thông số dự đoán liên ảnh cho bộ phận ước lượng chuyển động. Độ lệch này cũng được gọi là vector chuyển động (motion vector, MV).

Bộ phận bù chuyển động được tạo cấu hình để thu nhận, ví dụ nhận, thông số dự đoán liên ảnh và để thực hiện dự đoán liên ảnh dựa trên hoặc sử dụng thông số dự đoán liên ảnh để thu nhận khối dự đoán liên ảnh 265. Bù chuyển động, được thực hiện bởi bộ phận bù chuyển động, có thể bao gồm việc tìm hoặc tạo khối dự đoán dựa trên vector chuyển động/khối được xác định bằng cách ước lượng chuyển động, có thể thực hiện các phép nội suy đến độ chính xác điểm ảnh con. Lọc nội suy có thể tạo các mẫu điểm ảnh bổ sung từ các mẫu điểm ảnh đã biết, vì vậy có khả năng tăng số lượng khối dự đoán ứng viên mà có thể được sử dụng để tạo mã khối hình ảnh. Khi nhận vector chuyển động đối với PU của khối hình ảnh hiện hành, bộ phận bù chuyển động có thể định vị khối dự đoán mà vector chuyển động trỏ đến trong một trong số các danh sách hình ảnh tham chiếu.

Bộ phận bù chuyển động cũng có thể tạo các phần tử cú pháp được liên kết với các khối và các lát video để sử dụng bởi bộ giải mã video 30 khi giải mã các khối hình ảnh của lát video. Ngoài hoặc theo cách khác đối với các lát và các phần tử cú pháp tương ứng, các nhóm ảnh xếp lớp và/hoặc các ảnh xếp lớp và các phần tử cú pháp tương ứng có thể được tạo ra hoặc được sử dụng.

Mã hóa entropi

Bộ phận mã hóa entropi 270 được tạo cấu hình để áp dụng, ví dụ, thuật toán hoặc sơ đồ mã hóa entropi (ví dụ, sơ đồ tạo mã độ dài biến số (variable length coding, VLC), sơ đồ VLC thích ứng ngữ cảnh (context adaptive VLC, CAVLC), sơ đồ tạo mã số học, nhị phân hóa, tạo mã số học nhị phân thích ứng ngữ cảnh (context adaptive binary arithmetic coding, CABAC), tạo mã số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (syntax-based context-adaptive binary arithmetic coding, SBAC), tạo mã entropi phân chia khoảng xác suất (probability interval partitioning entropy, PIPE) hoặc hệ

phương pháp hoặc kỹ thuật mã hóa entropi khác) hoặc rẽ nhánh (không nén) trên các hệ số được lượng tử hóa 209, các thông số dự đoán liên ảnh, các thông số dự đoán trong ảnh, các thông số lọc vòng và/hoặc các phần tử cú pháp khác để thu nhận dữ liệu hình ảnh được mã hóa 21 mà có thể được kết xuất qua đầu ra 272, ví dụ ở dạng dòng bit được mã hóa 21, sao cho, ví dụ, bộ giải mã video 30 có thể nhận và sử dụng các thông số để giải mã. Dòng bit được mã hóa 21 có thể được truyền đến bộ giải mã video 30, hoặc được lưu trữ trong bộ nhớ để truyền hoặc truy hồi sau đó bởi bộ giải mã video 30.

Các sự thay đổi cấu trúc khác của bộ mã hóa video 20 có thể được sử dụng để mã hóa luồng video. Ví dụ, bộ mã hóa dựa trên không biến đổi 20 có thể lượng tử hóa tín hiệu dư trực tiếp mà không cần bộ phận xử lý biến đổi 206 đối với các khối hoặc các khung nhất định. Theo một cách thực hiện khác, bộ mã hóa 20 có thể có bộ phận lượng tử hóa 208 và bộ phận lượng tử hóa ngược 210 được kết hợp thành một bộ phận.

Bộ giải mã và phương pháp giải mã

Fig.3 thể hiện một ví dụ về bộ giải mã video 30 mà được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Bộ giải mã video 30 được tạo cấu hình để nhận dữ liệu hình ảnh được mã hóa 21 (ví dụ, dòng bit được mã hóa 21), ví dụ được mã hóa bởi bộ mã hóa 20, để thu nhận hình ảnh được giải mã 331. Dữ liệu hình ảnh hoặc dòng bit được mã hóa bao gồm thông tin để giải mã dữ liệu hình ảnh được mã hóa, ví dụ dữ liệu mà đại diện cho các khối hình ảnh của lát video được mã hóa (và/hoặc các nhóm ảnh xếp lớp hoặc các ảnh xếp lớp) và các phần tử cú pháp được liên kết.

Trong ví dụ trên Fig.3, bộ giải mã 30 bao gồm bộ phận giải mã entropi 304, bộ phận lượng tử hóa ngược 310, bộ phận xử lý biến đổi ngược 312, bộ phận tái tạo 314 (ví dụ, bộ cộng 314), bộ lọc vòng 320, bộ đệm hình ảnh được giải mã (decoded picture buffer, DBP) 330, bộ phận ứng dụng chế độ 360, bộ phận dự đoán liên ảnh 344 và bộ phận dự đoán trong ảnh 354. Bộ phận dự đoán liên ảnh 344 có thể là hoặc bao gồm bộ phận bù chuyển động. Bộ giải mã video 30 có thể, trong một số ví dụ, thực hiện cách giải mã thường ngược với cách mã hóa được mô tả liên quan đến bộ mã hóa video 100 từ Fig.2.

Như được giải thích đối với bộ mã hóa 20, bộ phận lượng tử hóa ngược 210, bộ phận xử lý biến đổi ngược 212, bộ phận tái tạo 214 bộ lọc vòng 220, bộ đệm hình ảnh

được giải mã (DPB) 230, bộ phận dự đoán liên ảnh 344 và bộ phận dự đoán trong ảnh 354 cũng được coi là tạo thành “bộ giải mã lắp sẵn” của bộ mã hóa video 20. Theo đó, bộ phận lượng tử hóa ngược 310 có thể giống hệt về chức năng với bộ phận lượng tử hóa ngược 110, bộ phận xử lý biến đổi ngược 312 có thể giống hệt về chức năng với bộ phận xử lý biến đổi ngược 212, bộ phận tái tạo 314 có thể giống hệt về chức năng với bộ phận tái tạo 214, bộ lọc vòng 320 có thể giống hệt về chức năng với bộ lọc vòng 220, và bộ đệm hình ảnh được giải mã 330 có thể giống hệt về chức năng với bộ đệm hình ảnh được giải mã 230. Do đó, các phần giải thích được cung cấp đối với các bộ phận và các chức năng tương ứng của bộ mã hóa video 20 áp dụng tương ứng vào các bộ phận và các chức năng tương ứng của bộ giải mã video 30.

Giải mã entropi

Bộ phận giải mã entropi 304 được tạo cấu hình để phân tách dòng bit 21 (hoặc nói chung là dữ liệu hình ảnh được mã hóa 21) và thực hiện, ví dụ, việc giải mã entropi đối với dữ liệu hình ảnh được mã hóa 21 để thu nhận, ví dụ, các hệ số được lượng tử hóa 309 và/hoặc các thông số mã hóa được giải mã (không được thể hiện trên Fig.3), ví dụ bất kỳ hoặc tất cả các thông số dự đoán liên ảnh (ví dụ, chỉ số hình ảnh tham chiếu và vector chuyển động), thông số dự đoán trong ảnh (ví dụ, chế độ dự đoán trong ảnh hoặc chỉ số), các thông số biến đổi, các thông số lượng tử hóa, các thông số lọc vòng, và/hoặc các phần tử cú pháp khác. Bộ phận giải mã entropi 304 có thể được tạo cấu hình để áp dụng các thuật toán hoặc các sơ đồ giải mã tương ứng với các sơ đồ mã hóa như được mô tả đối với bộ phận mã hóa entropi 270 của bộ mã hóa 20. Bộ phận giải mã entropi 304 có thể còn được tạo cấu hình để cung cấp các thông số dự đoán liên ảnh, thông số dự đoán trong ảnh và/hoặc các phần tử cú pháp khác cho bộ phận ứng dụng chế độ 360 và các thông số khác cho các bộ phận khác của bộ giải mã 30. Bộ giải mã video 30 có thể nhận các phần tử cú pháp ở mức lát video và/hoặc mức khối video. Ngoài hoặc theo cách khác đối với các lát và các phần tử cú pháp tương ứng, các nhóm ảnh xếp lớp và/hoặc các ảnh xếp lớp và các phần tử cú pháp tương ứng có thể được nhận và/hoặc được sử dụng.

Lượng tử hóa ngược

Bộ phận lượng tử hóa ngược 310 có thể được tạo cấu hình để nhận các thông số lượng tử hóa (QP) (hoặc nói chung là thông tin liên quan đến lượng tử hóa ngược) và các hệ số được lượng tử hóa từ dữ liệu hình ảnh được mã hóa 21 (ví dụ bằng cách phân tách và/hoặc giải mã, ví dụ bằng bộ phận giải mã entropi 304) và để áp dụng, dựa trên các thông số lượng tử hóa, lượng tử hóa ngược trên các hệ số được lượng tử hóa được giải mã 309 để thu nhận các hệ số được khử lượng tử hóa 311, mà cũng có thể được gọi là các hệ số biến đổi 311. Quy trình lượng tử hóa ngược có thể bao gồm việc sử dụng thông số lượng tử hóa được xác định bởi bộ mã hóa video 20 đối với mỗi khối video trong lát video (hoặc ảnh xếp lớp hoặc nhóm ảnh xếp lớp) để xác định mức độ lượng tử hóa và, tương tự, mức độ lượng tử hóa ngược mà sẽ được áp dụng.

Biến đổi ngược

Bộ phận xử lý biến đổi ngược 312 có thể được tạo cấu hình để nhận các hệ số được khử lượng tử hóa 311, cũng được gọi là các hệ số biến đổi 311, và để áp dụng việc biến đổi vào các hệ số được khử lượng tử hóa 311 để thu nhận các khối dư được tái tạo 213 trong miền mẫu. Các khối dư được tái tạo 213 cũng có thể được gọi là các khối biến đổi 313. Biến đổi có thể là biến đổi ngược, ví dụ, DCT ngược, DST ngược, biến đổi nguyên ngược, hoặc quy trình biến đổi ngược tương tự về mặt khái niệm. Bộ phận xử lý biến đổi ngược 312 có thể còn được tạo cấu hình để nhận các thông số biến đổi hoặc thông tin tương ứng từ dữ liệu hình ảnh được mã hóa 21 (ví dụ bằng cách phân tách và/hoặc giải mã, ví dụ bằng bộ phận giải mã entropi 304) để xác định việc biến đổi cần được áp dụng vào các hệ số được khử lượng tử hóa 311.

Tái tạo

Bộ phận tái tạo 314 (ví dụ bộ tổng hoặc bộ cộng 314) có thể được tạo cấu hình để cộng khối dư được tái tạo 313, vào khối dự đoán 365 để thu nhận khối được tái tạo 315 trong miền mẫu, ví dụ bằng cách cộng các giá trị mẫu của khối dư được tái tạo 313 và các giá trị mẫu của khối dự đoán 365.

Lọc

Bộ phận lọc vòng 320 (trong vòng mã hóa hoặc sau vòng mã hóa) được tạo cấu hình để lọc khối được tái tạo 315 để thu nhận khối được lọc 321, ví dụ để làm trơn nhẵn

các sự chuyển tiếp điểm ảnh, hoặc nếu không thì nâng cao chất lượng video. Bộ phận lọc vòng 320 có thể bao gồm một hoặc nhiều bộ lọc vòng như bộ lọc giải khối, bộ lọc độ lệch thích ứng mẫu (sample-adaptive offset, SAO) hoặc một hoặc nhiều bộ lọc khác, ví dụ bộ lọc hai chiều, bộ lọc vòng thích ứng (adaptive loop filter, ALF), các bộ lọc làm nhọn, các bộ lọc làm phẳng hoặc các bộ lọc cộng tác, hoặc sự kết hợp bất kỳ của chúng. Mặc dù bộ phận lọc vòng 320 được thể hiện trên Fig.3 như là bộ lọc nội vòng (in loop filter), nhưng theo các cấu hình khác, bộ phận lọc vòng 320 có thể được thực hiện như là bộ lọc sau vòng.

Bộ đệm hình ảnh được giải mã

Các khối video được giải mã 321 của hình ảnh sau đó được lưu trữ trong bộ đệm hình ảnh được giải mã 330, mà lưu trữ các hình ảnh được giải mã 331 như là các hình ảnh tham chiếu để bù chuyển động sau đó đối với các hình ảnh khác và/hoặc để kết xuất việc hiển thị tương ứng.

Bộ giải mã 30 được tạo cấu hình để kết xuất hình ảnh được giải mã 311, ví dụ qua đầu ra 312, để trình diễn cho người dùng hoặc cho người dùng xem.

Dự đoán

Bộ phận dự đoán liên ảnh 344 có thể giống hệt bộ phận dự đoán liên ảnh 244 (cụ thể là giống hệt bộ phận bù chuyển động) và bộ phận dự đoán trong ảnh 354 có thể giống hệt bộ phận dự đoán liên ảnh 254 về chức năng, và thực hiện các quyết định chia tách hoặc phân chia và việc dự đoán dựa trên các thông số phân chia và/hoặc dự đoán hoặc thông tin tương ứng được nhận từ dữ liệu hình ảnh được mã hóa 21 (ví dụ bằng cách phân tách và/hoặc giải mã, ví dụ bằng bộ phận giải mã entropi 304). Bộ phận ứng dụng chế độ 360 có thể được tạo cấu hình để thực hiện việc dự đoán (dự đoán trong ảnh hoặc liên ảnh) trên từng khối dựa trên các hình ảnh được tái tạo, các khối hoặc các mẫu tương ứng (được lọc hoặc không được lọc) để thu nhận khối dự đoán 365.

Khi lát video được mã hóa như là lát được mã hóa trong ảnh (I), bộ phận dự đoán trong ảnh 354 của bộ phận ứng dụng chế độ 360 được tạo cấu hình để tạo khối dự đoán 365 đối với khối hình ảnh của lát video hiện hành dựa trên chế độ dự đoán trong ảnh được báo hiệu và dữ liệu từ các khối được giải mã trước đó của hình ảnh hiện hành. Khi

hình ảnh video được tạo mã như là lát được tạo mã liên ảnh (tức là, B hoặc P), bộ phận dự đoán liên ảnh 344 (ví dụ, bộ phận bù chuyển động) của bộ phận ứng dụng chế độ 360 được tạo cấu hình để tạo ra các khối dự đoán 365 đối với khối video của lát video hiện hành dựa trên các vector chuyển động và các phần tử cú pháp khác được nhận từ bộ phận giải mã entropi 304. Đối với dự đoán liên ảnh, các khối dự đoán có thể được tạo ra từ một trong số các hình ảnh tham chiếu trong một trong số các danh sách hình ảnh tham chiếu. Bộ giải mã video 30 có thể xây dựng các danh sách khung tham chiếu, List (danh sách) 0 và List (danh sách) 1, sử dụng các kỹ thuật xây dựng mặc định dựa trên các hình ảnh tham chiếu được lưu trữ trong DPB 330. Điều giống hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án sử dụng các nhóm ảnh xếp lớp (ví dụ, các nhóm ảnh xếp lớp video) và/hoặc các ảnh xếp lớp (ví dụ, các ảnh xếp lớp video) ngoài hoặc theo cách khác đối với các lát (ví dụ, các lát video), ví dụ video có thể được tạo mã nhờ sử dụng các nhóm ảnh xếp lớp và/hoặc các ảnh xếp lớp I, P hoặc B.

Bộ phận ứng dụng chế độ 360 được tạo cấu hình để xác định thông tin dự đoán đối với khối video của lát video hiện hành bằng cách phân tách các vector chuyển động hoặc thông tin liên quan và các phần tử cú pháp khác, và sử dụng thông tin dự đoán để tạo ra các khối dự đoán đối với khối video hiện hành đang được giải mã. Ví dụ, bộ phận ứng dụng chế độ 360 sử dụng một số trong số các phần tử cú pháp được nhận để xác định chế độ dự đoán (ví dụ, dự đoán trong ảnh hoặc liên ảnh) được sử dụng để tạo mã các khối video của lát video, loại lát dự đoán liên ảnh (ví dụ, lát B, lát P, hoặc lát GPB), thông tin xây dựng đối với một hoặc nhiều danh sách trong số các danh sách hình ảnh tham chiếu đối với lát, các vector chuyển động đối với mỗi khối video được mã hóa liên ảnh của lát, trạng thái dự đoán liên ảnh đối với mỗi khối video được mã hóa liên ảnh của lát, và thông tin khác để giải mã các khối video trong lát video hiện hành. Điều giống hoặc tương tự có thể được áp dụng cho hoặc bởi các phương án sử dụng các nhóm ảnh xếp lớp (ví dụ, các nhóm ảnh xếp lớp video) và/hoặc các ảnh xếp lớp (ví dụ, các ảnh xếp lớp video) ngoài hoặc theo cách khác đối với các lát (ví dụ, các lát video), ví dụ video có thể được tạo mã nhờ sử dụng các nhóm ảnh xếp lớp và/hoặc các ảnh xếp lớp I, P hoặc B.

Các phương án của bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân chia và/hoặc giải mã hình ảnh bằng cách sử dụng các lát (cũng

được gọi là các lát video), trong đó hình ảnh có thể được phân chia thành hoặc được giải mã nhờ sử dụng một hoặc nhiều lát (thường là không chồng chéo), và mỗi lát có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU).

Các phương án của bộ giải mã video 30 như được thể hiện trên Fig.3 có thể được tạo cấu hình để phân chia và/hoặc giải mã hình ảnh bằng cách sử dụng các nhóm ảnh xếp lớp (cũng được gọi là các nhóm ảnh xếp lớp video) và/hoặc các ảnh xếp lớp (cũng được gọi là các ảnh xếp lớp video), trong đó hình ảnh có thể được phân chia thành hoặc được giải mã nhờ sử dụng một hoặc nhiều nhóm ảnh xếp lớp (thường là không chồng chéo), và mỗi nhóm ảnh xếp lớp có thể bao gồm, ví dụ một hoặc nhiều khối (ví dụ, các CTU) hoặc một hoặc nhiều ảnh xếp lớp, trong đó mỗi ảnh xếp lớp, ví dụ có thể là hình chữ nhật và có thể bao gồm một hoặc nhiều khối (ví dụ, các CTU), ví dụ các khối hoàn chỉnh hoặc các khối thành phần.

Các biến thể khác của bộ giải mã video 30 có thể được sử dụng để giải mã dữ liệu hình ảnh được mã hóa 21. Ví dụ, bộ giải mã 30 có thể tạo ra luồng video đầu ra mà không cần bộ phận lọc vòng 320. Ví dụ, bộ giải mã dựa trên không biến đổi 30 có thể lượng tử hóa ngược tín hiệu dư một cách trực tiếp mà không cần bộ phận xử lý biến đổi ngược 312 đối với các khối hoặc các khung nhất định. Theo một cách thực hiện khác, bộ giải mã video 30 có thể có bộ phận lượng tử hóa ngược 310 và bộ phận xử lý biến đổi ngược 312 được kết hợp vào một bộ phận đơn lẻ.

Cần hiểu rằng, trong bộ mã hóa 20 và bộ giải mã 30, kết quả xử lý của bước hiện hành có thể được xử lý thêm và sau đó được kết xuất sang bước tiếp theo. Ví dụ, sau khi lọc nội suy, dẫn xuất vectơ chuyển động hoặc lọc vòng, thao tác thêm nữa, như cắt hoặc dịch chuyển, có thể được thực hiện trên kết quả xử lý của lọc nội suy, dẫn xuất vectơ chuyển động hoặc lọc vòng.

Cần lưu ý rằng các thao tác thêm nữa có thể được áp dụng vào các vectơ chuyển động được dẫn xuất của khối hiện hành (bao gồm nhưng không bị giới hạn ở việc điều khiển các vectơ chuyển động điểm của chế độ afin, các vectơ chuyển động khối con trong afin, các chế độ phẳng, ATMVP, các vectơ chuyển động thời gian, v.v.). Ví dụ, giá trị của vectơ chuyển động bị ràng buộc ở khoảng được xác định trước theo bit đại diện của nó. Nếu bit đại diện của vectơ chuyển động là bitDepth, thì khoảng này là -

$2^{(\text{bitDepth}-1)} \sim 2^{(\text{bitDepth}-1)-1}$, trong đó “^” có nghĩa là mũ. Ví dụ, nếu bitDepth được thiết lập bằng 16, thì khoảng này là -32768 ~ 32767; nếu bitDepth được thiết lập bằng 18, thì khoảng này là -131072~131071. Ví dụ, giá trị của vector chuyển động được dẫn xuất (ví dụ các MV của bốn khối con 4x4 trong một khối 8x8) bị ràng buộc sao cho chênh lệch lớn nhất giữa các phần nguyên của bốn MV khối con 4x4 không lớn hơn N điểm ảnh, như không lớn hơn 1 điểm ảnh. Ở đây đề xuất hai phương pháp để ràng buộc vector chuyển động theo bitDepth.

Phương pháp 1: loại bỏ MSB tràn (bit có trọng số lớn nhất) bằng các phép toán sau đây

$$ux = (mvx + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (1)$$

$$mvx = (ux \geq 2^{\text{bitDepth}-1}) ? (ux - 2^{\text{bitDepth}}) : ux \quad (2)$$

$$uy = (mvy + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (3)$$

$$mvy = (uy \geq 2^{\text{bitDepth}-1}) ? (uy - 2^{\text{bitDepth}}) : uy \quad (4)$$

trong đó mvx là thành phần ngang của vector chuyển động của khối ảnh hoặc khối con, mvy là thành phần thẳng đứng của vector chuyển động của khối ảnh hoặc khối con, và ux và uy chỉ báo giá trị trung gian;

Ví dụ, nếu giá trị của mvx là -32769, thì sau khi áp dụng công thức (1) và (2), giá trị thu được là 32767. Trong hệ thống máy tính, các số thập phân được lưu trữ như là bù hai. Bù hai của -32769 là 1,0111,1111,1111,1111 (17 bit), sau đó MSB được loại bỏ, vì vậy bù hai thu được là 0111,1111,1111,1111 (số thập phân là 32767), mà giống như đầu ra bằng cách áp dụng công thức (1) và (2).

$$ux = (mvp_x + mvd_x + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (5)$$

$$mvx = (ux \geq 2^{\text{bitDepth}-1}) ? (ux - 2^{\text{bitDepth}}) : ux \quad (6)$$

$$uy = (mvp_y + mvd_y + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (7)$$

$$mvy = (uy \geq 2^{\text{bitDepth}-1}) ? (uy - 2^{\text{bitDepth}}) : uy \quad (8)$$

Các phép toán có thể được áp dụng trong suốt quá trình tính tổng của.mvp và mvd, như được thể hiện trong công thức từ (5) đến (8).

Phương pháp 2: loại bỏ MSB tràn bằng cách cắt giá trị

$$vx = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1} - 1, vx)$$

$$vy = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1} - 1, vy)$$

trong đó vx là thành phần ngang của vector chuyển động của khối ảnh hoặc khối con, vy là thành phần thẳng đứng của vector chuyển động của khối ảnh hoặc khối con; x, y và z lần lượt tương ứng với ba giá trị đầu vào của quy trình cắt MV, và định nghĩa của hàm Clip3 là như sau:

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

Fig.4 là hình vẽ giản lược của thiết bị tạo mã video 400 theo một phương án của sáng chế. Thiết bị tạo mã video 400 thích hợp để thực hiện các phương án được bộc lộ như được mô tả ở đây. Theo một phương án, thiết bị tạo mã video 400 có thể là bộ giải mã như bộ giải mã video 30 trên Fig.1A hoặc bộ mã hóa như bộ mã hóa video 20 trên Fig.1A.

Thiết bị tạo mã video 400 bao gồm các cổng vào 410 (hoặc các cổng đầu vào 410) và các bộ phận nhận (Rx) 420 để nhận dữ liệu; bộ xử lý, bộ phận logic, hoặc bộ xử lý trung tâm (CPU) 430 để xử lý dữ liệu; các bộ phận truyền (Tx) 440 và các cổng ra 450 (hoặc các cổng đầu ra 450) để truyền dữ liệu; và bộ nhớ 460 để lưu trữ dữ liệu. Thiết bị tạo mã video 400 cũng có thể bao gồm các thành phần quang sang điện (optical-to-electrical, OE) và các thành phần điện sang quang (electrical-to-optical, EO) được ghép nối với các cổng vào 410, các bộ phận nhận 420, các bộ phận truyền 440, và các cổng ra 450 để lấy ra hoặc lấy vào các tín hiệu điện hoặc quang.

Bộ xử lý 430 được thực thi bởi phần cứng và phần mềm. Bộ xử lý 430 có thể được thực thi như là một hoặc nhiều chip CPU, nhân (ví dụ, bộ xử lý nhiều nhân), các FPGA, các ASIC, và các DSP. Bộ xử lý 430 truyền thông với các cổng vào 410, các bộ phận nhận 420, các bộ phận truyền 440, các cổng ra 450, và bộ nhớ 460. Bộ xử lý 430 bao gồm môđun tạo mã 470. Môđun tạo mã 470 thực thi các phương án được bộc lộ được mô tả trên đây. Ví dụ, môđun tạo mã 470 thực hiện, xử lý, chuẩn bị, hoặc cung cấp các thao tác tạo mã khác nhau. Việc bao gồm môđun tạo mã 470 do đó sẽ mang lại sự cải tiến quan trọng cho chức năng của thiết bị tạo mã video 400 và đem lại sự biến đổi thiết

bị tạo mã video 400 sang trạng thái khác. Theo cách khác, môđun tạo mã 470 được thực thi dưới dạng các lệnh được lưu trữ trong bộ nhớ 460 và được thực hiện bởi bộ xử lý 430.

Bộ nhớ 460 có thể bao gồm một hoặc nhiều đĩa, ổ băng, và ổ trạng thái rắn và có thể được sử dụng làm thiết bị lưu trữ dữ liệu tràn, để lưu trữ các chương trình khi các chương trình này được chọn để thực hiện, và để lưu trữ các lệnh và dữ liệu được đọc trong suốt quá trình thực hiện chương trình. Bộ nhớ 460 có thể là, ví dụ, khả biến và/hoặc bất khả biến và có thể là bộ nhớ chỉ đọc (ROM), bộ nhớ truy cập ngẫu nhiên (RAM), bộ nhớ khả lập địa chỉ nội dung bậc ba (ternary content-addressable memory, TCAM), và/hoặc bộ nhớ truy cập ngẫu nhiên tĩnh (SRAM).

Fig.5 là sơ đồ khối được đơn giản hóa của máy 500 mà có thể được sử dụng làm một trong số hoặc cả thiết bị nguồn 12 và thiết bị đích 14 từ Fig.1 theo một phương án làm ví dụ.

Bộ xử lý 502 trong máy 500 có thể là bộ xử lý trung tâm. Theo cách khác, bộ xử lý 502 có thể là loại thiết bị khác bất kỳ, hoặc nhiều thiết bị, có khả năng thao tác hoặc xử lý thông tin hiện có hoặc được phát triển trong tương lai. Mặc dù các cách thực hiện được bộc lộ có thể được thực hành với một bộ xử lý đơn lẻ như được thể hiện, ví dụ, bộ xử lý 502, các ưu điểm về tốc độ và hiệu năng có thể đạt được nhờ sử dụng nhiều hơn một bộ xử lý.

Bộ nhớ 504 trong máy 500 có thể là thiết bị nhớ chỉ đọc (ROM) hoặc thiết bị nhớ truy cập ngẫu nhiên (RAM) theo một cách thực hiện. Loại thiết bị lưu trữ thích hợp khác bất kỳ có thể được sử dụng làm bộ nhớ 504. Bộ nhớ 504 có thể bao gồm mã và dữ liệu 506 mà được truy cập bởi bộ xử lý 502 nhờ sử dụng bus 512. Bộ nhớ 504 có thể còn bao gồm hệ điều hành 508 và các chương trình ứng dụng 510, các chương trình ứng dụng 510 bao gồm ít nhất một chương trình mà cho phép bộ xử lý 502 thực hiện các phương pháp được mô tả ở đây. Ví dụ, các chương trình ứng dụng 510 có thể bao gồm các ứng dụng từ 1 đến N, mà còn bao gồm ứng dụng tạo mã video mà thực hiện các phương pháp được mô tả ở đây.

Máy 500 cũng có thể bao gồm một hoặc nhiều thiết bị đầu ra, như bộ hiển thị 518. Bộ hiển thị 518 có thể là, theo một ví dụ, bộ hiển thị nhạy chạm mà kết hợp bộ hiển thị

với phần tử nhảy chạm mà có thể hoạt động để nhận biết các đầu vào chạm. Bộ hiển thị 518 có thể được ghép nối với bộ xử lý 502 qua bus 512.

Mặc dù được thể hiện ở đây dưới dạng một bus đơn lẻ, nhưng bus 512 của máy 500 có thể bao gồm nhiều bus. Hơn nữa, bộ phận lưu trữ thứ cấp 514 có thể được ghép nối trực tiếp với các thành phần khác của máy 500 hoặc có thể được truy cập qua mạng và có thể bao gồm một bộ phận tích hợp đơn lẻ như thẻ nhớ hoặc nhiều bộ phận như nhiều thẻ nhớ. Vì vậy, máy 500 có thể được thực thi theo rất nhiều loại cấu hình.

Sự tinh chế vector chuyển động (Motion vector refinement, MVR)

Các vector chuyển động thường được xác định ít nhất một phần tại phía bộ mã hóa và được báo hiệu cho bộ giải mã trong dòng bit đã tạo mã. Tuy nhiên, các vector chuyển động cũng có thể được tinh chế tại bộ giải mã (và cũng tại bộ mã hóa) bắt đầu từ các vector chuyển động ban đầu được chỉ báo trong dòng bit. Trong trường hợp này, ví dụ, sự tương tự giữa các phần đệm (patch) của các điểm ảnh đã được giải mã được trở bởi các vector chuyển động ban đầu có thể được sử dụng để nâng cao độ chính xác của các vector chuyển động ban đầu. Sự tinh chế chuyển động mang lại ưu điểm về giảm lượng thông tin báo hiệu thừa: độ chính xác chuyển động ban đầu được nâng cao theo cách giống nhau tại cả bộ mã hóa và bộ giải mã và vì vậy, không cần báo hiệu bổ sung cho sự tinh chế.

Cần lưu ý rằng các vector chuyển động ban đầu trước khi tinh chế có thể không phải là các vector chuyển động tốt nhất mà dẫn đến dự đoán tốt nhất. Do các vector chuyển động ban đầu được báo hiệu trong dòng bit, nên nó có thể không có khả năng đại diện cho vector chuyển động ban đầu với độ chính xác rất cao (mà sẽ làm tăng tốc độ bit), do đó quy trình tinh chế vector chuyển động được sử dụng để nâng cao vector chuyển động ban đầu. Các vector chuyển động ban đầu có thể, ví dụ, là các vector chuyển động mà được sử dụng khi dự đoán khối lân cận của khối hiện hành. Trong trường hợp này, như vậy là đủ để báo hiệu chỉ báo trong dòng bit, chỉ báo các vector chuyển động mà khối lân cận của chúng được sử dụng bởi khối hiện hành. Cơ chế dự đoán như vậy rất hữu ích để làm giảm số lượng bit để đại diện cho các vector chuyển động ban đầu. Tuy nhiên, độ chính xác của các vector chuyển động ban đầu có thể thấp, do, nói chung, các vector chuyển động của hai khối lân cận không được mong đợi là giống hệt nhau.

Để nâng cao thêm nữa độ chính xác của các vector chuyển động mà không làm tăng lượng thông tin báo hiệu thừa thêm nữa, có thể có lợi khi tinh chế thêm nữa các vector chuyển động được dẫn xuất tại phía bộ mã hóa và được cung cấp (được báo hiệu) trong dòng bit. Sự tinh chế vector chuyển động có thể được thực hiện tại bộ giải mã mà không cần sự hỗ trợ từ bộ mã hóa. Bộ mã hóa trong vòng lặp bộ giải mã của nó có thể sử dụng cùng sự tinh chế để thu nhận các vector chuyển động đã tinh chế tương ứng như sẽ là khả dụng tại bộ giải mã. Sự tinh chế đối với khối hiện hành mà đang được tái tạo trong hình ảnh hiện hành được thực hiện bằng cách xác định khuôn mẫu của các mẫu tái tạo, xác định không gian tìm kiếm quanh chuyển động ban đầu thông tin đối với khối hiện hành và tìm phần hình ảnh tham chiếu trong không gian tìm kiếm so khớp tốt nhất với khuôn mẫu. Phần so khớp tốt nhất xác định các vector chuyển động đã tinh chế đối với khối hiện hành mà sau đó được sử dụng để thu nhận các mẫu được dự đoán liên ảnh đối với khối hiện hành, tức là khối hiện hành đang được tái tạo.

Sự tinh chế vector chuyển động là một phần của bộ phận dự đoán liên ảnh (244) trên Fig.2 và 344 trên Fig.3.

Sự tinh chế vector chuyển động có thể được thực hiện theo các bước sau đây:

Thông thường, các vector chuyển động ban đầu có thể được xác định dựa trên chỉ báo trong dòng bit. Ví dụ, chỉ số có thể được báo hiệu trong dòng bit mà chỉ báo vị trí trong danh sách các vector chuyển động ứng viên. Theo một ví dụ khác, chỉ số bộ dự đoán vector chuyển động và giá trị hiệu vector chuyển động có thể được báo hiệu trong dòng bit. Các vector chuyển động mà được xác định dựa trên chỉ báo trong dòng bit được xác định là các vector chuyển động ban đầu. Trong trường hợp song dự đoán, trong đó dự đoán liên ảnh đối với khối hiện hành được thu nhận dưới dạng tổ hợp có trọng số của khối đã dự đoán của các mẫu mà được xác định theo hai vector chuyển động, để cho vector chuyển động ban đầu trong hình ảnh tham chiếu thứ nhất trong danh sách L0 được biểu thị là MV0; và vector chuyển động ban đầu trong hình ảnh tham chiếu thứ hai trong danh sách L1 được biểu thị là MV1.

Việc sử dụng các vector chuyển động ban đầu, các cặp vector chuyển động (MV) ứng viên tinh chế được xác định. Ít nhất, hai cặp ứng viên tinh chế cần được xác định. Thông thường, các cặp vector chuyển động ứng viên tinh chế được xác định dựa trên cặp

vector chuyển động ban đầu (MV0, MV1). Hơn nữa, các cặp MV ứng viên được xác định bằng cách cộng các hiệu vector chuyển động nhỏ vào MV0 và MV1. Ví dụ, các cặp MV ứng viên có thể bao gồm các trường hợp sau đây:

- (MV0, MV1)
- (MV0 + (0,1), MV1 + (0,-1))
- (MV0 + (1,0), MV1 + (-1,0))
- (MV0 + (0,-1), MV1 + (0,1))
- (MV0 + (-1,0), MV1 + (1,0))
- ...

Trong đó (1,-1) biểu thị vector mà có độ dịch chuyển là 1 theo chiều ngang (hoặc x) và độ dịch chuyển là -1 theo chiều thẳng đứng (hoặc y).

Cần lưu ý rằng danh sách trên đây của các cặp ứng viên chỉ là các ví dụ để giải thích và sáng chế không bị giới hạn ở danh sách cụ thể của các ứng viên.

Các cặp vector chuyển động (MV) ứng viên tinh chế tạo thành không gian tìm kiếm của quy trình tinh chế vector chuyển động.

Trong việc song dự đoán khối hiện hành, hai khối dự đoán thu nhận được nhờ sử dụng vector chuyển động thứ nhất tương ứng của danh sách L0 và vector chuyển động thứ hai của danh sách L1, được kết hợp vào một tín hiệu dự đoán, mà có thể cung cấp sự thích ứng tốt hơn với tín hiệu gốc so với dự đoán đơn, dẫn đến ít thông tin còn lại hơn và có thể là nén hiệu quả hơn.

Khi tinh chế vector chuyển động, hai khối dự đoán thu nhận được nhờ sử dụng vector chuyển động thứ nhất tương ứng và vector chuyển động thứ hai của cặp MV ứng viên được so sánh dựa trên chuẩn đo tương tự đối với mỗi cặp trong số các cặp MV ứng viên tinh chế. Cặp MV ứng viên dẫn đến sự tương tự cao nhất thường được chọn làm các vector chuyển động đã tinh chế. Được biểu thị là MV0' và MV1' lần lượt là vector chuyển động đã tinh chế trong hình ảnh tham chiếu thứ nhất trong danh sách L0 và vector chuyển động đã tinh chế trong hình ảnh tham chiếu thứ hai trong danh sách L1. Nói cách khác, các dự đoán được thu nhận tương ứng với vector chuyển động của danh sách L0 và vector chuyển động của danh sách L1 của cặp vector chuyển động ứng viên, mà sau

đó được so sánh dựa trên chuẩn đo tương tự. Cặp vectơ chuyển động ứng viên mà có sự tương tự liên kết cao nhất được chọn là cặp MV đã tinh chế.

Thông thường, đầu ra của quy trình tinh chế là các MC đã tinh chế. Các MC đã tinh chế có thể giống như các MV ban đầu hoặc có thể khác với các MV ban đầu, phụ thuộc vào cặp MV ứng viên nào đạt được sự tương tự cao nhất, cặp MV ứng viên được tạo ra bởi các MV ban đầu cũng nằm trong số các ứng viên cặp MV. Nói cách khác, nếu cặp MV ứng viên cao nhất mà đạt được sự tương tự cao nhất được tạo ra bởi các MV ban đầu, thì các MC đã tinh chế và các MV ban đầu là bằng nhau.

Thay vì chọn vị trí mà tối đa hóa chuẩn đo tương tự, một phương pháp khác là chọn vị trí mà tối thiểu hóa chuẩn đo không tương tự. Đo lường so sánh không tương tự có thể là SAD (Sum of absolute differences - tổng các sai phân tuyệt đối), MRSAD (mean removed sum of absolute differences - tổng các sai phân tuyệt đối bỏ trung bình), SSE (Sum of Squared Error - tổng của sai số bình phương) v.v. SAD giữa hai khối dự đoán có thể được thu nhận nhờ sử dụng cặp MV ứng viên (CMV0, CMV1), SAD có thể được tính toán như sau:

$$\begin{aligned} \text{SAD}(\text{CMV0}, \text{CMV1}) \\ &= \sum_{x=0}^{\text{nCbW}-1} \sum_{y=0}^{\text{nCbH}-1} \text{abs}(\text{predSamplesL0}[x][y] \\ &\quad - \text{predSamplesL1}[x][y]) \end{aligned}$$

trong đó nCbH và nCbW là độ cao và độ rộng của các khối dự đoán, chức năng $\text{abs}(a)$ quy định giá trị tuyệt đối của đối số a , predSamplesL0 và predSamplesL1 là các mẫu khối dự đoán được thu nhận theo cặp MV ứng viên mà được biểu thị bởi (CMV0, CMV1).

Theo cách khác, các đo lường so sánh không tương tự có thể được thu nhận bằng cách đánh giá chỉ tập con của các mẫu trong khối dự đoán, để làm giảm số lượng phép tính. Ví dụ ở dưới đây, trong đó các hàng của các mẫu theo cách khác có trong phép tính SAD (mọi hàng thứ hai được đánh giá).

SAD(CMV0, CMV1)

$$= \sum_{x=0}^{nCbW-1} \sum_{y=0}^{nCbH/2-1} \text{abs}(\text{predSamplesL0}[x][2 * y] - \text{predSamplesL1}[x][2 * y])$$

Một ví dụ về sự tinh chế vector chuyển động được giải thích trong tài liệu JVET-M1001-v3, “Versatile Video Coding (Draft 4)” (tạo mã video đa năng (bản phác thảo 4)) của JVET (of ITU-T SG 16 WP 3 và ISO/IEC JTC 1/SC 29/WG 11) mà được công bố tại "<http://phenix.it-sudparis.eu/jvet/>". Phần “8.4.3 Decoder side motion vector refinement process” (Quy trình tinh chế vector chuyển động phía bộ giải mã) trong tài liệu này lấy ví dụ sự tinh chế vector chuyển động.

Để làm giảm các yêu cầu bộ nhớ nội bộ để tinh chế, theo một số phương án, quy trình tinh chế vector chuyển động có thể được thực hiện một cách độc lập trên các khối của các mẫu độ chói được thu nhận bằng cách phân chia khối được tạo mã của các mẫu mà vượt quá độ rộng định trước hoặc độ cao định trước nhất định trong các mẫu độ chói thành các khối con của các mẫu mà nhỏ hơn hoặc bằng độ rộng định trước và độ cao định trước trong độ chói. Cặp MV đã tinh chế đối với mỗi khối con trong khối được tạo mã được phân chia có thể là khác nhau. Sau đó, dự đoán liên ảnh đối với cả độ chói và sắc độ được thực hiện cho mỗi khối con nhờ sử dụng cặp MV đã tinh chế của khối con đó.

Mỗi MV của cặp MV ban đầu có thể có độ chính xác điểm ảnh thập phân. Nói cách khác, MV chỉ báo sự dịch chuyển giữa khối hiện hành của các mẫu và vùng tham chiếu được lấy mẫu lại và sự dịch chuyển này có thể trở đến vị trí thập phân theo các chiều ngang và thẳng đứng từ lưới số nguyên của các mẫu tham chiếu được tái tạo. Thông thường, việc nội suy 2 chiều của các giá trị lưới mẫu số nguyên tham chiếu được tái tạo được thực hiện để thu nhận các giá trị mẫu tại vị trí độ lệch mẫu thập phân. Quy trình thu nhận các mẫu được dự đoán từ các hình ảnh tham chiếu được tái tạo sử dụng cặp MV ứng viên có thể qua một trong số các phương pháp sau đây:

- Làm tròn phần thập phân của cặp MV ban đầu đến vị trí số nguyên gần nhất và thu được các giá trị lưới số nguyên của các hình ảnh tham chiếu được tái tạo.

- Thực hiện nội suy song tuyến tính tách rời được 2 điểm nối (ví dụ, song tuyến tính) để thu nhận các giá trị mẫu đã dự đoán tại độ chính xác điểm ảnh thập phân được chỉ báo bởi cặp MV ban đầu.

- Thực hiện nội suy tách rời được điểm nối cao hơn (ví dụ 8 điểm nối hoặc 6 điểm nối) để thu nhận các giá trị mẫu đã dự đoán tại độ chính xác điểm ảnh thập phân được chỉ báo bởi cặp MV ban đầu.

Mặc dù các cặp MV ứng viên có thể có độ lệch điểm ảnh con tùy ý liên quan đến cặp MV ban đầu, nhưng theo một số phương án, để đơn giản cho việc tìm kiếm, các cặp MV ứng viên được chọn với khoảng cách điểm ảnh nguyên đối với cặp MV ban đầu. Trong các trường hợp này, các mẫu đã dự đoán qua tất cả cặp MV ứng viên có thể thu được bằng cách thực hiện dự đoán đối với khối của các mẫu quanh cặp MV ban đầu để bao trùm tất cả các vị trí tinh chế quanh cặp MV ban đầu.

Theo một số phương án, một khi giá trị chi phí không tương tự tại tất cả các cặp MV ứng viên tại khoảng cách nguyên từ cặp MV ban đầu đã được đánh giá, các cặp MV ứng viên bổ sung tại các độ lệch khoảng cách điểm ảnh con từ vị trí giá trị chi phí tốt nhất được bổ sung. Các mẫu đã dự đoán được thu nhận cho mỗi vị trí trong số các vị trí này sử dụng một trong số các phương pháp được mô tả trên đây và các chi phí không tương tự được đánh giá và được so sánh để thu nhận vị trí không tương tự thấp nhất. Trong các phương án khác nhất định, để tránh quy trình dự đoán tốn kém về mặt tính toán đối với mỗi vị trí khoảng cách điểm ảnh con quanh vị trí khoảng cách nguyên chi phí tốt nhất, các giá trị chi phí khoảng cách nguyên được đánh giá được nhớ và bề mặt sai số thông số được làm khít ở lân cận của vị trí khoảng cách nguyên tốt nhất. Sau đó, tối thiểu của bề mặt sai số này được tính theo giải tích và được sử dụng làm vị trí với mức không tương tự tối thiểu. Trong các trường hợp này, giá trị chi phí không tương tự được coi là thu được từ các giá trị chi phí khoảng cách nguyên tính được.

Việc ứng dụng sự tinh chế vector chuyển động cho khối được tạo mã đã biết của các mẫu có thể tùy thuộc vào các tính chất tạo mã nhất định của khối được tạo mã của các mẫu. Một số ví dụ về các tính chất tạo mã này có thể là:

- Khoảng cách theo số lượng hình ảnh (khi được lấy mẫu tại tốc độ khung đều) từ hình ảnh hiện hành đến hai hình ảnh tham chiếu được sử dụng cho song dự đoán khối

được tạo mã của các mẫu là bằng nhau và nằm ở các phía ngược nhau của hình ảnh hiện hành.

- Sự không tương tự ban đầu giữa hai các khối đã dự đoán thu nhận được nhờ sử dụng cặp MV ban đầu nhỏ hơn ngưỡng cho từng mẫu định trước.

Sự tinh chế dòng quang học song dự đoán (Bi-predictive Optical flow refinement, BPOF)

Sự tinh chế dòng quang học song dự đoán là quy trình để nâng cao độ chính xác của song dự đoán khối, mà không cần báo hiệu bổ sung rõ ràng trong dòng bit không phải là được báo hiệu cho song dự đoán. Đây là một phần của bộ phận dự đoán liên ảnh (244) trên Fig.2 và 344 trên Fig.3.

Trong song dự đoán, 2 dự đoán liên ảnh được thu nhận theo hai vectơ chuyển động, sau đó các dự đoán được kết hợp nhờ áp dụng bình quân gia quyền. Việc dự đoán kết hợp có thể dẫn đến năng lượng dư giảm đi do nhiễu âm lượng tử hóa trong hai phần đệm tham chiếu bị triệt tiêu, nhờ đó mang lại hiệu suất tạo mã tốt hơn dự đoán đơn. Tổ hợp có trọng số trong song dự đoán có thể được thực hiện bởi phương trình:

$$\text{Song dự đoán} = \text{Prediction1} * W1 + \text{Prediction2} * W2 + K,$$

trong đó W1 và W2 là các hệ số trọng số mà có thể được báo hiệu trong dòng bit hoặc có thể được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. K là hệ số cộng thêm mà cũng có thể được báo hiệu trong dòng bit hoặc được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. Ví dụ, song dự đoán có thể thu được nhờ sử dụng

$$\text{Song dự đoán} = (\text{Prediction1} + \text{Prediction2})/2,$$

trong đó W1 và W2 được thiết đặt ở $\frac{1}{2}$ và K được thiết đặt ở 0.

Mục tiêu của sự tinh chế dòng quang học là để nâng cao độ chính xác của song dự đoán. Dòng quang học là mẫu hình của chuyển động rõ ràng của các đối tượng ảnh giữa hai khung liên tiếp, dòng quang học được gây ra bởi chuyển động của đối tượng hoặc camera. Quy trình tinh chế dòng quang học nâng cao độ chính xác của song dự đoán nhờ áp dụng phương trình dòng quang học (giải phương trình dòng quang học).

Theo một ví dụ, điểm ảnh $I(x,y,t)$ được đặt trong khung thứ nhất (x và y tương ứng với các tọa độ không gian, t tương ứng với chiều thời gian). Đối tượng được biểu diễn

bởi điểm ảnh di chuyển theo khoảng cách (dx, dy) ở khung tiếp theo được lấy sau thời gian dt . Do các điểm ảnh đó là giống nhau và cường độ không thay đổi, nên phương trình dòng quang học được đưa ra theo:

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

$I(x, y, t)$ chỉ ra cường độ (giá trị mẫu) của điểm ảnh tại các tọa độ là (x, y, t) .

Theo một ví dụ khác, các sự dịch chuyển nhỏ và số hạng thứ tự cao hơn trong khai triển chuỗi Taylor (Taylor series expansion) được bỏ qua, các phương trình dòng quang học cũng có thể được viết là:

$$\frac{\partial I}{\partial t} + v_x \frac{\partial I}{\partial x} + v_y \frac{\partial I}{\partial y} = 0$$

Trong đó $\frac{\partial I}{\partial x}$ và $\frac{\partial I}{\partial y}$ là các gradient mẫu không gian ngang và thẳng đứng tại vị trí (x, y) và $\frac{\partial I}{\partial t}$ là đạo hàm riêng theo thời gian tại (x, y) .

Sự tinh chế dòng quang học sử dụng nguyên tắc trên đây để nâng cao chất lượng của song dự đoán.

Cách thực hiện sự tinh chế dòng quang học thường bao gồm các bước:

1. Tính toán các gradient mẫu;
2. Tính toán sự chênh lệch giữa dự đoán thứ nhất và dự đoán thứ hai;
3. Tính toán sự dịch chuyển của các điểm ảnh hoặc nhóm các điểm ảnh mà tối thiểu hóa sai số Δ giữa hai phần đệm tham chiếu thu nhận được nhờ sử dụng phương trình dòng quang học

$$\Delta = (I^{(0)} - I^{(1)}) + v_x(\tau_0 \frac{\partial I^{(0)}}{\partial x} + \tau_1 \frac{\partial I^{(1)}}{\partial x}) + v_y(\tau_0 \frac{\partial I^{(0)}}{\partial y} + \tau_1 \frac{\partial I^{(1)}}{\partial y})$$

trong đó $I^{(0)}$ tương ứng với giá trị mẫu trong dự đoán thứ nhất, $I^{(1)}$ là giá trị mẫu trong dự đoán thứ hai, , và $\partial I^{(0)} / \partial x$ và $\partial I^{(0)} / \partial y$ là các gradient theo các chiều $-x$ và $-y$. τ_1 và τ_0 biểu thị các khoảng cách đến các hình ảnh tham chiếu, trong đó dự đoán thứ nhất và dự đoán thứ hai được thu nhận. Vectơ chuyển động (v_x, v_y) thu được bởi quy trình tối thiểu hóa. Một số cách tiếp cận tối thiểu hóa tổng của các sai số bình phương trong khi một số cách tiếp cận tối thiểu hóa tổng của sai số tuyệt đối.

4. Việc sử dụng cách thực hiện phương trình dòng quang học, như dưới đây:

$$pred_{BIO} = 1/2 \cdot \left(I^{(0)} + I^{(1)} + v_x/2 \cdot \left(\tau_1 \partial I^{(1)} / \partial x - \tau_0 \partial I^{(0)} / \partial x \right) + v_y/2 \cdot \left(\tau_1 \partial I^{(1)} / \partial y - \tau_0 \partial I^{(0)} / \partial y \right) \right)$$

Trong đó $pred_{BIO}$ chỉ ra dự đoán được thay đổi mà là đầu ra của quy trình tính chế dòng quang học.

Các gradien mẫu có thể thu được bởi công thức sau đây

- $\partial I(x, y, t) / \partial x = I(x + 1, y, t) - I(x - 1, y, t)$
- $\partial I(x, y, t) / \partial y = I(x, y + 1, t) - I(x, y - 1, t)$

Theo một số phương án, để đơn giản hóa độ phức tạp của việc ước lượng sự dịch chuyển đối với mỗi điểm ảnh, sự dịch chuyển được ước lượng đối với nhóm các điểm ảnh. Trong một số ví dụ, để tính toán song dự đoán được cải tiến đối với khối có 4x4 mẫu độ chói, các sự dịch chuyển được ước lượng nhờ sử dụng các giá trị mẫu của khối có 8x8 mẫu độ chói với 4x4 khối của các mẫu ở tâm của nó.

Đầu vào của quy trình tính chế dòng quang học là các mẫu dự đoán từ hai hình ảnh tham chiếu và đầu ra của sự tính chế dòng quang học là dự đoán kết hợp ($pred_{BIO}$) mà được tính toán theo phương trình dòng quang học.

Một ví dụ về sự tính chế dòng quang học được giải thích trong phần 8.4.7.4 “Bidirectional optical flow prediction process” (Quy trình dự đoán dòng quang học hai chiều) của tài liệu JVET-M1001, Versatile Video Coding (tạo mã video đa năng) (Draft 4 - bản phác thảo 4).

Các thuật ngữ sự tính chế dòng quang học, sự tính chế dòng quang học song dự đoán và sự tính chế dòng quang học hai chiều được sử dụng hoán đổi trong bản mô tả, do các thuật ngữ này tương đương về bản chất.

Theo một ví dụ, sự tính chế vector chuyển động và sự tính chế dòng quang học được áp dụng liên tiếp như sau:

Bước 0, thu nhận các vector chuyển động ban đầu như ở 1010 trên Fig.8.

Bước 1, sự tính chế vector chuyển động được áp dụng ở 1020, và các vector chuyển động đã tính chế 1030 được thu nhận.

Bước 2, các dự đoán được thu nhận theo các vector chuyển động tinh chế 1040. Các dự đoán thu được là $I^{(0)}$ và $I^{(1)}$, mà là đầu vào của quy trình tinh chế dòng quang học.

Bước 3, quy trình tinh chế dòng quang học được áp dụng vào các dự đoán, để thu nhận dự đoán đã thay đổi. Dự đoán đã thay đổi được thu nhận theo phương trình dòng quang học và được biểu thị là pred_{BIO} .

Tuy nhiên, quy trình tinh chế dòng quang học mạnh về tính toán. Thời gian giải mã được tăng lên do áp dụng sự tinh chế dòng quang học.

Theo một phương án của sáng chế, phương pháp quyết định liệu có áp dụng sự tinh chế dòng quang học hay không được bộc lộ, quyết định này có thể được thực hiện theo các phép tính được thực hiện trong suốt quy trình tinh chế vector chuyển động.

Cụ thể hơn, kết quả của các phép tính được thực hiện trong suốt quy trình tinh chế vector chuyển động được sử dụng để xác định liệu có áp dụng sự tinh chế dòng quang học hay không.

Mục tiêu của sáng chế là bỏ qua việc áp dụng sự tinh chế dòng quang học theo điều kiện quy định, sao cho thời gian giải mã trung bình được giảm bớt (bằng cách bỏ qua các phép tính cần thiết).

Theo phương án làm ví dụ thứ nhất, các bước sau đây được áp dụng để thu được dự đoán đối với khôi tạo mã hiện hành:

Bước 0: thu nhận các vector chuyển động ban đầu dựa trên thông tin chỉ báo trong dòng bit.

Bước 1: thu nhận các dự đoán thứ nhất dựa trên các vector chuyển động ban đầu và bộ lọc nội suy M điểm nối.

Bước 2: thu nhận chỉ phí so khớp theo các dự đoán thứ nhất.

Bước 3: thu nhận vector chuyển động đã tinh chế theo các vector chuyển động ban đầu và chỉ phí so khớp.

Bước 4: thu nhận dự đoán thứ hai theo vector chuyển động đã tinh chế và bộ lọc nội suy K điểm nối.

Bước 5: xác định liệu có thực hiện quy trình tinh chế dòng quang học theo chi phí so khớp hay không. Theo một ví dụ, chi phí so khớp được so sánh với ngưỡng, và quy trình tinh chế dòng quang học được thực hiện khi giá trị của chi phí so khớp lớn hơn hoặc bằng ngưỡng. Bước 5 cũng có thể được thực hiện trước bước 3 hoặc bước 4.

Bước 6: Khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện, thì sự tinh chế dòng quang học được áp dụng với dự đoán thứ hai như là đầu vào và dự đoán thứ hai đã thay đổi như là đầu ra. Nếu xác định được là không phải như vậy, thì sự tinh chế dòng quang học không được áp dụng trên dự đoán thứ hai. Nói cách khác, khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện, thì dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và theo quy trình tinh chế dòng quang học. Nếu không, dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và không áp dụng quy trình tinh chế dòng quang học.

Phần giải thích chi tiết về các bước như sau:

Ở bước 0, hai vector chuyển động ban đầu được thu nhận như là đầu vào. Các vector chuyển động ban đầu có thể được xác định dựa trên thông tin chỉ báo trong dòng bit. Ví dụ, chỉ số có thể được báo hiệu trong dòng bit, chỉ số này chỉ báo vị trí trong danh sách các vector chuyển động ứng viên. Theo một ví dụ khác, chỉ số bộ dự đoán vector chuyển động và giá trị hiệu vector chuyển động có thể được báo hiệu trong dòng bit. Vector chuyển động mà được xác định dựa trên thông tin chỉ báo trong dòng bit được xác định là các vector chuyển động ban đầu.

Theo một ví dụ khác, các chỉ báo hình ảnh tham chiếu có thể thu được từ dòng bit, các vector chuyển động ban đầu được thu nhận dựa trên các chỉ báo hình ảnh tham chiếu. Các chỉ báo hình ảnh tham chiếu được sử dụng để xác định các hình ảnh tham chiếu mà được trỏ bởi các vector chuyển động ban đầu.

Bước 1, bước 2 và bước 3 tương ứng với quy trình tinh chế vector chuyển động như được giải thích trong các ví dụ trên đây. Các vector chuyển động ban đầu được tinh chế theo sự tinh chế vector chuyển động. Theo một ví dụ, chi phí so khớp là đo lường sự tương tự mà được sử dụng trong sự tinh chế vector chuyển động.

Theo bước 1, các dự đoán thứ nhất được thu nhận tương ứng với các vectơ chuyển động ban đầu. Theo một ví dụ, có ít nhất 2 cặp vectơ chuyển động ứng viên trong quy trình tinh chế vectơ chuyển động, một trong số đó thường là cặp được tạo ra bởi các vectơ chuyển động ban đầu (MV0, MV1). Nói cách khác, tập hợp các vectơ chuyển động ứng viên thường bao gồm nhiều hơn một cặp, trong đó một trong số các cặp thường là (MV0, MV1). Cặp còn lại của các vectơ chuyển động ứng viên được xác định dựa trên (MV0, MV1), bằng cách cộng các nhiễu loạn nhỏ vào các vectơ chuyển động (như được giải thích trong các ví dụ trên đây).

Ở bước 1, các dự đoán thứ nhất tương ứng với mỗi cặp vectơ chuyển động ứng viên được thu nhận dựa trên bộ lọc nội suy M điểm nối. Ví dụ, một dự đoán tương ứng với MV0 có thể được thu nhận bằng cách đặt khối hình chữ nhật trong hình ảnh tham chiếu (hình ảnh mà đã được mã hóa trong bộ mã hóa hoặc được giải mã trong bộ giải mã), trong đó khối nào được trả bởi MV0. Sau đó, bộ lọc nội suy được áp dụng thuận lợi vào các mẫu trong khối được trả bởi MV0. Để cung cấp ước lượng chuyển động chính xác hơn, độ phân giải của hình ảnh tham chiếu có thể được nâng cao bằng cách nội suy các mẫu giữa các điểm ảnh. Điểm ảnh nội suy thập phân có thể được thực hiện bằng cách bình quân gia quyền các điểm ảnh gần nhất. Ở đây, bộ lọc M điểm nối thường có thể là bộ lọc 2, 4, 6, hoặc 8 điểm nối (không bị giới hạn ở các tùy chọn này), có nghĩa là bộ lọc có M hệ số nhân. Dự đoán tương ứng với MV1 có thể được thu nhận tương tự, bằng cách đặt khối hình chữ nhật trong cùng hình ảnh tham chiếu giống hoặc hình ảnh tham chiếu khác. Kích thước của hình chữ nhật tỷ lệ với kích thước của khối tạo mã hiện hành.

Ở bước 2, chi phí so khớp được liên kết với mỗi cặp vectơ chuyển động ứng viên được xác định theo các dự đoán thứ nhất.

Theo bước 2, ít nhất một chi phí so khớp (ví dụ, đo lường sự tương tự) được thu nhận tương ứng với một trong số các cặp vectơ chuyển động (MV) ứng viên tinh chế. Sự tương tự giữa hai khối dự đoán càng cao, thì chi phí so khớp càng nhỏ.

Chi phí so khớp này được sử dụng trong sự tinh chế của các vectơ chuyển động ban đầu ở bước 3. Vectơ chuyển động đã tinh chế được chọn theo chi phí so khớp này.

Ở bước 4, dự đoán thứ hai được thu nhận theo vectơ chuyển động đã tinh chế và

bộ lọc nội suy K điểm nối. Trong trường hợp 2 vectơ chuyển động đã tinh chế ($MV0'$ và $MV1'$), mà là trường hợp song dự đoán, hai dự đoán thứ hai được thu nhận.

Dự đoán thứ hai được thu nhận bằng cách áp dụng bộ lọc nội suy thứ hai (bộ lọc K điểm nối), mà có thể hoặc có thể không giống hệt bộ lọc nội suy thứ nhất (bộ lọc M điểm nối). Dự đoán thứ hai được thu nhận tương tự như dự đoán thứ nhất, với việc áp dụng bộ lọc nội suy thứ hai và theo khối được trả bởi $MV0'$ và $MV1'$ trong hình ảnh tham chiếu.

Ở bước 5, chi phí so khớp này được sử dụng để xác định liệu có thực hiện quy trình tinh chế dòng quang học hay không, theo nội dung sau đây:

Khi giá trị của chi phí so khớp nhỏ hơn ngưỡng định trước, thì sự tinh chế dòng quang học không được áp dụng. Khi giá trị của chi phí so khớp lớn hơn hoặc bằng ngưỡng này, thì quy trình tinh chế dòng quang học được thực hiện. Nếu quy trình tinh chế dòng quang học được thực hiện, thì các mẫu của dự đoán cuối cùng được thay đổi.

Ở bước 6, theo đầu ra của bước 5, nếu chi phí so khớp lớn hơn hoặc bằng ngưỡng này, thì quy trình tinh chế dòng quang học được áp dụng vào các dự đoán thứ hai, các dự đoán thứ hai được thu nhận theo $MV0'$ và $MV1'$ (các vectơ chuyển động đã tinh chế). Dự đoán cuối cùng đối với khối tạo mã hiện hành được thu nhận bằng cách thực hiện quy trình tinh chế dòng quang học trên các dự đoán thứ hai, các dự đoán thứ hai được trả bởi $MV0'$ và $MV1'$. Nếu chi phí so khớp nhỏ hơn ngưỡng này, thì dự đoán cuối cùng được thu nhận theo các dự đoán thứ hai được trả bởi $MV0'$ và $MV1'$, mà không áp dụng sự tinh chế dòng quang học, điều này có nghĩa là, bước 6 không được thực hiện.

Theo một cách thực hiện, chi phí so khớp ở bước 2 là chi phí so khớp tương ứng với cặp vectơ chuyển động ban đầu (mà là một trong số các cặp vectơ chuyển động (MV) ứng viên tinh chế). Chi phí so khớp có thể tương ứng với cặp $MV0$, $MV1$.

Theo một cách thực hiện khác, chi phí so khớp ở bước 2 là chi phí so khớp bằng chi phí so khớp nhỏ nhất trong số các cặp vectơ chuyển động (MV) ứng viên tinh chế. Nói cách khác, chi phí so khớp được thu nhận tương ứng với mỗi cặp vectơ chuyển động ứng viên tinh chế, và chi phí so khớp này bằng chi phí so khớp nhỏ nhất trong số đó.

Theo một ví dụ, chi phí so khớp này là chi phí so khớp tương ứng với cặp vector chuyển động đã tinh chế $MV0'$ và $MV1'$, (do cặp vector chuyển động đã tinh chế ($MV0'$, $MV1'$) được chọn vì có chi phí so khớp nhỏ nhất).

Ví dụ, các cặp MV có thể được xây dựng bằng cách sau đây.

Các cặp MV ứng viên được xác định bằng cách cộng các hiệu vector chuyển động nhỏ vào $MV0$ và $MV1$. Ví dụ, các cặp MV ứng viên có thể bao gồm các trường hợp sau đây:

$$(MV0, MV1)$$

$$(MV0 + (0,1), MV1 + (0,-1))$$

$$(MV0 + (1,0), MV1 + (-1,0))$$

$MV0$ và $MV1$ là các vector chuyển động ban đầu, $MV0'$ và $MV1'$ là các vector chuyển động đã tinh chế trong toàn bộ đơn này.

Theo một cách thực hiện khác, khi quy trình tinh chế dòng quang học không được thực hiện, thì dự đoán cuối cùng được thu nhận theo công thức sau đây:

$$\text{Song dự đoán} = \text{Prediction1} * W1 + \text{Prediction2} * W2 + K,$$

trong đó $W1$ và $W2$ là các hệ số trọng số, $W1$ và $W2$ có thể được báo hiệu trong dòng bit, hoặc $W1$ và $W2$ có thể được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. K là hệ số cộng thêm mà cũng có thể được báo hiệu trong dòng bit hoặc được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. Theo một ví dụ, song dự đoán có thể thu được nhờ sử dụng

$$\text{Song dự đoán} = (\text{Prediction1} + \text{Prediction2})/2,$$

trong đó $W1$ và $W2$ được thiết đặt ở $\frac{1}{2}$ và K được thiết đặt ở 0. Prediction1 và prediction2 là các dự đoán thứ hai mà được thu nhận nhờ lọc nội suy K điểm nối, Prediction1 tương ứng với MV được tinh chế thứ nhất ($MV0'$) và Prediction2 tương ứng với MV được tinh chế thứ hai ($MV1'$).

Phương trình trên đây đạt được tổ hợp có trọng số của hai dự đoán, và kết quả là dự đoán cuối cùng đối với khối.

Ngưỡng này có thể là giá trị định trước, giá trị của ngưỡng có thể phụ thuộc vào

kích thước của khối dự đoán. Ví dụ, ngưỡng có thể là $thr = nCbW \times nCbH \times K$, trong đó K là giá trị lớn hơn không (zero), $nCbW$ và $nCbH$ là độ rộng và độ cao của khối dự đoán.

Phương án thứ nhất còn được lấy làm ví dụ bởi lưu đồ trên Fig.6.

Theo một cách thực hiện, bộ lọc M điểm nối là bộ lọc 2 điểm nối (ví dụ, bộ lọc song tuyến tính) với một trong số các điểm nối bằng không (zero). Theo cách thực hiện này, bộ lọc M điểm nối sử dụng 2 hệ số nhân, giá trị của một hệ số luôn bằng không (zero). Hệ số nào có giá trị bằng không được xác định dựa trên điểm mẫu thập phân, mà điểm mẫu thập phân được trả bởi vector chuyển động. Trong trường hợp này, giá trị của hệ số nhân thứ nhất, hoặc giá trị của hệ số nhân thứ hai có thể bằng không, phụ thuộc vào phần thập phân của vector chuyển động.

Bộ lọc như vậy (với 2 điểm nối, một trong số đó bằng không) có thể được lấy làm ví dụ theo bảng sau đây:

Vị trí mẫu thập phân p	các hệ số lọc nội suy	
	$fb_L[p][0]$	$fb_L[p][1]$
1	16	0
2	16	0
3	16	0
4	16	0
5	16	0
6	16	0
7	16	0
8	0	16
9	0	16
10	0	16
11	0	16
12	0	16
13	0	16
14	0	16
15	0	16

Vị trí mẫu phân số (p) có thể thu được theo các thành phần của vector chuyển động ban đầu hoặc vector chuyển động đã tinh chế. Ví dụ, nếu thành phần $-x$ của vector chuyển động được đưa ra bởi $MV0x$, thì vị trí mẫu phân số có thể thu được dưới dạng $p = MV0x \% 16$, trong đó “ $\%$ ” là phép toán môđun. Nói chung $p = MV0x \% K$, trong đó K là

số lượng vị trí mẫu thập phân giữa hai vị trí mẫu. Bộ lọc nội suy được lấy làm ví dụ trên đây cũng có thể được gọi là bộ lọc 1 điểm nổi, do chỉ một trong số các điểm nổi của bộ lọc taps là khác không (zero) vào một thời điểm.

Theo một cách thực hiện, giá trị của K bằng 8. Trong các ví dụ khác, giá trị của M nhỏ hơn 8.

Theo một cách thực hiện, giá trị của M và giá trị của K đều bằng 8.

Vector chuyển động ban đầu được thu nhận ở 710, mà là đầu vào của bộ phận tinh chế vector chuyển động. Không gian tìm kiếm được xây dựng quanh vector chuyển động ban đầu bởi bộ phận tinh chế vector chuyển động (740). Theo một ví dụ, không gian tìm kiếm bao gồm các cặp vector chuyển động ứng viên, vector chuyển động thứ nhất của cặp tương ứng với hình ảnh tham chiếu thứ nhất và vector chuyển động thứ hai của cặp tương ứng với hình ảnh tham chiếu thứ hai. Các dự đoán thứ nhất tương ứng với mỗi cặp vector chuyển động ứng viên được thu nhận ở bước 710, nhờ áp dụng bộ lọc nội suy M điểm nổi. Như là một phần của sự tinh chế vector chuyển động, chi phí so khớp được tính toán tương ứng với một trong số các cặp vector chuyển động trong không gian tìm kiếm (720). Chi phí so khớp này được sử dụng như là một phần của hai quy trình, quy trình thứ nhất là sự tinh chế vector chuyển động (740), trong đó chi phí so khớp được sử dụng để quyết định cặp vector chuyển động nào được chọn làm cặp vector chuyển động đã tinh chế (750). Quy trình thứ hai là sự quyết định về việc liệu sự tinh chế dòng quang học (770) có được áp dụng hay không. Sau khi vector chuyển động đã tinh chế được thu nhận, dự đoán thứ hai đối với khối hiện hành được thu nhận bởi (760). Nếu chi phí so khớp lớn hơn hoặc bằng ngưỡng, thì sự tinh chế dòng quang học được áp dụng và dự đoán ở 760 được thay đổi bởi 770 để thu nhận dự đoán đã thay đổi (780). Dự đoán đã thay đổi thường khác về các giá trị mẫu với dự đoán thứ hai ở bước 760.

Theo một ví dụ, quy trình tinh chế vector chuyển động được thực hiện nhiều hơn một lần để tinh chế vector chuyển động thêm nữa. Trong ví dụ này, các vector chuyển động ban đầu trước tiên được tinh chế bởi quy trình tinh chế vector chuyển động, để thu nhận vector chuyển động đã tinh chế thứ nhất. Sau đó, sự tinh chế vector chuyển động được thực hiện một lần nữa, trong trường hợp này, vector chuyển động đã tinh chế thứ nhất được coi là các vector chuyển động ban đầu đối với sự tinh chế vector chuyển động

thứ hai.

Theo phương án làm ví dụ thứ hai, các bước sau đây được áp dụng để thu nhận dự đoán đối với khối tạo mã hiện hành:

Bước 0: thu nhận các vector chuyển động ban đầu dựa trên thông tin chỉ báo trong dòng bit.

Bước 1: thu nhận các dự đoán thứ nhất dựa trên các vector chuyển động ban đầu và bộ lọc nội suy M điểm nối.

Bước 2: thu nhận N chi phí so khớp theo các dự đoán thứ nhất.

Bước 3: thu nhận vector chuyển động đã tinh chế theo các vector chuyển động ban đầu và N chi phí so khớp, dựa trên hàm thứ nhất.

Bước 4: thu nhận dự đoán thứ hai theo vector chuyển động đã tinh chế và bộ lọc nội suy K điểm nối.

Bước 5: xác định liệu có thực hiện quy trình tinh chế dòng quang học theo N chi phí so khớp. Chi phí được dẫn xuất được thu nhận theo N chi phí so khớp và hàm thứ hai. Theo một ví dụ, chi phí được dẫn xuất được so sánh với ngưỡng, và quy trình tinh chế dòng quang học được thực hiện khi giá trị của chi phí được dẫn xuất lớn hơn hoặc bằng ngưỡng. Bước 5 cũng có thể được thực hiện trước bước 3 hoặc bước 4.

Bước 6: thay đổi ít nhất một mẫu của dự đoán của khối tạo mã hiện hành nhờ áp dụng sự tinh chế dòng quang học, khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện.

Khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện, thì sự tinh chế dòng quang học được áp dụng với dự đoán thứ hai như là đầu vào và dự đoán thứ hai đã thay đổi như là đầu ra. Nếu xác định được là không phải như vậy, thì sự tinh chế dòng quang học không được áp dụng trên dự đoán thứ hai. Nói cách khác, khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện, thì dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và theo quy trình tinh chế dòng quang học. Nếu không, dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và không áp dụng quy trình tinh chế dòng quang học.

Phân giải thích chi tiết về các bước như sau:

Ở bước 0, hai vector chuyển động ban đầu được thu nhận như là đầu vào. Vector chuyển động ban đầu có thể được xác định dựa trên thông tin chỉ báo trong dòng bit. Ví dụ, chỉ số có thể được báo hiệu trong dòng bit, chỉ số này chỉ báo vị trí trong danh sách các vector chuyển động ứng viên. Theo một ví dụ khác, chỉ số bộ dự đoán vector chuyển động và giá trị hiệu vector chuyển động có thể được báo hiệu trong dòng bit. Vector chuyển động mà được xác định dựa trên thông tin chỉ báo trong dòng bit được xác định là các vector chuyển động ban đầu.

Theo một ví dụ khác, các chỉ báo hình ảnh tham chiếu có thể thu được từ dòng bit, các vector chuyển động ban đầu được thu nhận dựa trên các chỉ báo hình ảnh tham chiếu. Các chỉ báo hình ảnh tham chiếu được sử dụng để xác định các hình ảnh tham chiếu mà được trỏ bởi các vector chuyển động ban đầu.

Bước 1, bước 2 và bước 3 tương ứng với quy trình tinh chế vector chuyển động như được giải thích trong các ví dụ trên đây. Các vector chuyển động ban đầu được tinh chế theo sự tinh chế vector chuyển động. Theo một ví dụ, chỉ phí so khớp là đo lường sự tương tự mà được sử dụng trong sự tinh chế vector chuyển động.

Theo bước 1, các dự đoán thứ nhất được thu nhận tương ứng với các vector chuyển động ban đầu. Theo một ví dụ, có ít nhất 2 cặp vector chuyển động ứng viên trong quy trình tinh chế vector chuyển động, một trong số đó thường là cặp được tạo ra bởi các vector chuyển động ban đầu (MV_0 , MV_1). Và cặp còn lại của các vector chuyển động ứng viên được xác định dựa trên (MV_0 , MV_1), bằng cách cộng các nhiễu loạn nhỏ vào các vector chuyển động (như được giải thích trong ví dụ trên đây).

Ở bước 1, các dự đoán thứ nhất tương ứng với mỗi cặp vector chuyển động ứng viên được thu nhận dựa trên bộ lọc nội suy M điểm nối.

Ở bước 2, N chỉ phí so khớp được liên kết với N cặp vector chuyển động ứng viên được xác định theo các dự đoán thứ nhất.

Theo bước 2, N chỉ phí so khớp (đo lường sự tương tự) được thu nhận tương ứng với N cặp trong số các cặp vector chuyển động (MV) ứng viên tinh chế. Sự tương tự giữa hai khối dự đoán càng cao, thì chỉ phí so khớp càng nhỏ.

N chi phí so khớp này được sử dụng khi tính chế các vectơ chuyển động ban đầu ở bước 3.

Vectơ chuyển động đã tính chế được xác định theo hàm thứ nhất và N chi phí so khớp.

Theo một ví dụ, vectơ chuyển động đã tính chế có thể thu được theo hàm sau đây:

- nếu (sad[3] + sad[5]) bằng (sad[4] << 1), thì dmVOffset[0] được thiết đặt bằng 0,

- Nếu không, phần sau đây áp dụng:

$$\text{dmVOffset}[0] = ((\text{sad}[3] - \text{sad}[5]) \ll 3) / (\text{sad}[3] + \text{sad}[5] - (\text{sad}[4] \ll 1))$$

- nếu (sad[1] + sad[7]) bằng (sad[4] << 1), dmVOffset[1] được thiết đặt bằng 0,

- Nếu không, phần sau đây áp dụng:

$$\text{dmVOffset}[1] = ((\text{sad}[1] - \text{sad}[7]) \ll 3) / (\text{sad}[1] + \text{sad}[7] - (\text{sad}[4] \ll 1))$$

trong đó dmVOffset[0] và dmVOffset[1] chỉ ra hiệu giữa vectơ chuyển động ban đầu và vectơ chuyển động đã tính chế. Theo một ví dụ, dmVOffset[0] và dmVOffset[1] chỉ ra thành phần -x và -y của hiệu giữa được các vectơ chuyển động tính chế và vectơ chuyển động ban đầu. sad[0] đến sad[7] là N chi phí so khớp, tương ứng với N cặp vectơ chuyển động ứng viên. Vectơ chuyển động đã tính chế được thu nhận bằng cách cộng dmVOffset với các vectơ chuyển động ban đầu.

Có thể có các hàm khác mà có thể được sử dụng để xác định vectơ chuyển động đã tính chế theo N chi phí so khớp. Hàm thứ nhất theo sáng chế không bị giới hạn ở phương trình trên đây.

Ở bước 4, dự đoán thứ hai được thu nhận theo vectơ chuyển động đã tính chế và bộ lọc nội suy K điểm nối. Trong trường hợp 2 vectơ chuyển động đã tính chế (MV0' và MV1'), mà là trường hợp song dự đoán, hai dự đoán thứ hai được thu nhận.

Dự đoán thứ hai được thu nhận bằng cách áp dụng bộ lọc nội suy thứ hai (bộ lọc K điểm nối), mà có thể hoặc có thể không giống hệt bộ lọc nội suy thứ nhất (bộ lọc M điểm nối). Dự đoán thứ hai được thu nhận tương tự như dự đoán thứ nhất, với việc áp

dụng bộ lọc nội suy thứ hai và theo khối được trả bởi $MV0'$ và $MV1'$ trong hình ảnh tham chiếu.

Ở bước 5, chi phí được dẫn xuất được thu nhận theo hàm thứ hai và N chi phí so khớp này. Chi phí được dẫn xuất được sử dụng để xác định liệu có thực hiện quy trình tinh chế dòng quang học hay không. Khi giá trị của chi phí được dẫn xuất này nhỏ hơn ngưỡng định trước, thì quy trình tinh chế dòng quang học không được thực hiện. Khi giá trị của chi phí được dẫn xuất lớn hơn hoặc bằng ngưỡng, thì quy trình tinh chế dòng quang học được thực hiện. Nếu quy trình tinh chế dòng quang học được thực hiện, thì các mẫu của dự đoán cuối cùng được thay đổi.

Ở bước 6, theo đầu ra của bước 5, nếu chi phí được dẫn xuất lớn hơn ngưỡng này, thì quy trình tinh chế dòng quang học được áp dụng vào các dự đoán thứ hai, các dự đoán thứ hai được thu nhận theo $MV0'$ và $MV1'$ (các vectơ chuyển động đã tinh chế). Dự đoán cuối cùng đối với khối tạo mã hiện hành được thu nhận bằng cách thực hiện quy trình tinh chế dòng quang học trên các dự đoán thứ hai, các dự đoán thứ hai được trả bởi $MV0'$ và $MV1'$. Nếu chi phí so khớp nhỏ hơn ngưỡng này, thì dự đoán cuối cùng được thu nhận theo các dự đoán thứ hai được trả bởi $MV0'$ và $MV1'$, mà không áp dụng sự tinh chế dòng quang học, điều này có nghĩa là, bước 6 không được thực hiện.

Theo một cách thực hiện khác, khi quy trình tinh chế dòng quang học không được thực hiện, thì dự đoán cuối cùng được thu nhận theo công thức sau đây:

$$\text{Song dự đoán} = \text{Prediction1} * W1 + \text{Prediction2} * W2 + K,$$

trong đó $W1$ và $W2$ là các hệ số trọng số, $W1$ và $W2$ có thể được báo hiệu trong dòng bit hoặc có thể được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. K là hệ số cộng thêm mà cũng có thể được báo hiệu trong dòng bit hoặc được xác định trước ở phía bộ mã hóa hoặc ở phía bộ giải mã. Theo một ví dụ, song dự đoán có thể thu được nhờ sử dụng

$$\text{Song dự đoán} = (\text{Prediction1} + \text{Prediction2})/2,$$

trong đó $W1$ và $W2$ được thiết đặt ở $\frac{1}{2}$ và K được thiết đặt ở 0. Prediction1 và prediction2 là các dự đoán thứ hai mà được thu nhận nhờ lọc nội suy K điểm nổi, Prediction1 tương ứng với MV được tinh chế thứ nhất ($MV0'$) và Prediction2 tương ứng với MV được tinh

chế thứ hai (MV1').

Phương trình trên đây đạt được tổ hợp có trọng số của hai dự đoán, và kết quả là dự đoán cuối cùng đối với khối.

Ngưỡng này có thể là giá trị định trước, giá trị của ngưỡng phụ thuộc vào kích thước của khối dự đoán. Ví dụ, ngưỡng có thể là $thr = nCbW \times nCbH \times K$, trong đó K là giá trị lớn hơn không (zero), $nCbW$ và $nCbH$ là độ rộng và độ cao của khối dự đoán.

Phương án thứ hai còn được lấy làm ví dụ bởi lưu đồ trên Fig.7.

Theo một cách thực hiện, bộ lọc M điểm nối là bộ lọc 2 điểm nối (ví dụ, bộ lọc song tuyến tính) với một trong số các điểm nối bằng không (zero). Theo cách thực hiện này, bộ lọc M điểm nối sử dụng 2 hệ số nhân, giá trị của một hệ số luôn bằng không (zero). Hệ số mà bằng không (zero) được xác định dựa trên điểm mẫu thập phân mà được trỏ bởi vector chuyển động. Trong trường hợp này, giá trị của hệ số nhân thứ nhất, hoặc giá trị của hệ số nhân thứ hai có thể bằng không, phụ thuộc vào phần thập phân của vector chuyển động.

Bộ lọc như vậy (với 2 điểm nối, một trong số đó bằng không) có thể được lấy làm ví dụ theo bảng sau đây:

Vị trí mẫu thập phân p	các hệ số lọc nội suy	
	$fb_L[p][0]$	$fb_L[p][1]$
1	16	0
2	16	0
3	16	0
4	16	0
5	16	0
6	16	0
7	16	0
8	0	16
9	0	16
10	0	16
11	0	16
12	0	16
13	0	16
14	0	16
15	0	16

Vị trí mẫu phân số (p) có thể thu được theo các thành phần của vector chuyển động ban đầu hoặc vector chuyển động đã tinh chế. Ví dụ, nếu thành phần $-x$ của vector chuyển động được đưa ra bởi $MV0x$, thì vị trí mẫu phân số có thể thu được dưới dạng $p = MV0x \% 16$, trong đó “%” là phép toán môđun. Nói chung $p = MV0x \% K$, trong đó K là số lượng vị trí mẫu thập phân giữa hai vị trí mẫu. Bộ lọc nội suy được lấy làm ví dụ trên đây cũng có thể được gọi là bộ lọc 1 điểm nối, do chỉ một trong số các điểm nối của bộ lọc taps là khác không (zero) vào một thời điểm.

Một ví dụ khác về bộ lọc nội suy song tuyến tính có thể như sau, trong trường hợp đó cả hai hệ số lọc đều khác không (zero):

Vị trí mẫu thập phân p	các hệ số lọc nội suy	
	$fb_L[p][0]$	$fb_L[p][1]$
1	15	1
2	14	2
3	13	3
4	12	4
5	11	5
6	10	6
7	9	7
8	8	8
9	7	9
10	6	10
11	5	11
12	4	12
13	3	13
14	2	14
15	1	15

Theo một cách thực hiện, giá trị của K bằng 8. Theo các ví dụ khác, giá trị của M nhỏ hơn 8.

Theo một cách thực hiện, giá trị của M và giá trị của K đều bằng 8.

Theo một cách thực hiện, hàm thứ hai có thể là hàm để kết hợp tuyến tính N chi phí so khớp theo $dmvOffset$, trong đó $dmvOffset$ đã thu được ở bước 3. Sự kết hợp tuyến tính của x và y sẽ là biểu thức bất kỳ ở dạng $ax + by$, trong đó a và b là các hằng số. Theo một ví dụ, các hằng số a và b có thể được xác định dựa trên $dmvOffset$. Các ví dụ

đối với hàm thứ hai được đưa ra dưới đây.

Theo một cách thực hiện, hàm thứ hai có thể là:

- $Sad[1]*A + Sad[2]*B + Sad[3]*C + Sad[4]*D$, trong đó A, B, C và D lớn hơn hoặc bằng không (zero). Theo một ví dụ, A,B,C và D có thể là các số nằm giữa 0 và 1, và được cộng lại bằng 1 (tức là $A+B+C+D=1$). Theo một ví dụ khác, A,B, C và D có thể là các số lớn hơn hoặc bằng 0 và được cộng lại bằng số cố định P được xác định trước, P có thể bằng 1,2, 4, 8, 16 v.v.
- A,B,C và D có thể là các số cố định được xác định trước.
- A, B, C và D có thể được dẫn xuất theo $dmvOffset[0]$ và $dmvOffset[1]$. Theo một ví dụ, $A = dmvOffset[0]$, $B = P1 - dmvOffset[0]$, $C = dmvOffset[1]$, $D = P2 - dmvOffset[1]$. Trong đó P1 và P2 có thể bằng 1, 4, 8, 16, v.v.
- Phương trình trên đây được đưa ra làm ví dụ. Phương trình đại diện cho tổ hợp tuyến tính của 4 chi phí so khớp để thu nhận chi phí được dẫn xuất. Trong phương trình này, $dmvOffset$ được sử dụng, mà có thể thu được trong bước 3. $dmvOffset$ đại diện cho hiệu giữa vector chuyển động đã tinh chế và vector chuyển động ban đầu. Theo một cách thực hiện cụ thể, $dmvOffset$ được xác định là hiệu giữa MV0 và MV0'. Cụ thể hơn, $dmvOffset[0]$ có thể là hiệu giữa thành phần -x của MV0 và MV0', trong khi đó $dmvOffset[1]$ có thể là hiệu giữa thành phần -y của MV0 và MV0'.

Theo một cách thực hiện khác, hàm thứ hai có thể là:

- $Sad[1]*A + Sad[2]*B + Sad[3]*C$, trong đó A, B và C lớn hơn hoặc bằng không. Theo một ví dụ, A, B, và C có thể là các số nằm giữa 0 và 1, và được cộng lại bằng 1 (tức là $A+B+C=1$). Theo một ví dụ khác, A, B và C có thể là các số lớn hơn hoặc bằng 0 và được cộng lại bằng số cố định được P xác định trước, P có thể bằng 1, 2, 4, 8, 16 v.v.
- A, B và C có thể là các số cố định được xác định trước.
- A, B và C có thể được dẫn xuất theo $dmvOffset[0]$ và $dmvOffset[1]$. Theo một ví dụ, $A = P - dmvOffset[0] - dmvOffset[1]$, $B = dmvOffset[0]$, $C = dmvOffset[1]$. Trong đó P có thể bằng 1, 4, 8, 16, v.v.

- Phương trình trên đây được đưa ra làm ví dụ. Phương trình đại diện cho tổ hợp tuyến tính của 3 chi phí so khớp để thu nhận chi phí được dẫn xuất. Trong phương trình này, $dmvOffset$ được sử dụng, mà có thể thu được trong bước 3. $dmvOffset$ đại diện cho hiệu giữa vector chuyển động đã tinh chế và vector chuyển động ban đầu. Theo một ví dụ, $dmvOffset$ được xác định là hiệu giữa $MV0$ và $MV0'$. Cụ thể hơn, $dmvOffset[0]$ có thể là hiệu giữa thành phần $-x$ của $MV0$ và $MV0'$, trong khi đó $dmvOffset[1]$ có thể là hiệu giữa thành phần $-y$ của $MV0$ và $MV0'$.

Theo một cách thực hiện khác, hàm thứ hai để thu nhận chi phí được dẫn xuất có thể là:

- Nhờ sử dụng 5 giá trị chi phí không tương tự (ví dụ, các giá trị SAD) đã đánh giá tại cặp MV đã tinh chế và các cặp MV ứng viên mà ở khoảng cách nguyên từ cặp MV đã tinh chế, hàm bề mặt sai số tham số.

$$E(x,y) = A*(x - x_0)^2 + B*(y - y_0)^2 + C$$

được đáp ứng trong đó (x_0, y_0) tương ứng với vị trí mà tại đó sự không tương tự giữa hai phần đệm tham chiếu được tối thiểu hóa, C là giá trị của chi phí tại (x_0, y_0) , và A, B là các hệ số mô hình. 5 yếu tố chưa biết này có thể được giải theo cách chính xác nếu 5 giá trị chi phí là khả dụng. Nói cách khác, phương trình đối với $E(x,y)$ có giả định rằng hình dạng của các chi phí so khớp dưới dạng hàm của các vị trí không gian gần vị trí chi phí so khớp tối thiểu là hình parabol.

Theo một phương án, các cặp MV ứng viên về phía bên trái, trên cùng, bên phải, và dưới cùng của cặp MV đã tinh chế ở khoảng cách một điểm ảnh nguyên được sử dụng. Trong trường hợp này, giả sử các giá trị được đánh giá của $E(x,y)$ tại các vị trí (x,y) là $(0,0)$, $(-1,0)$, $(0,-1)$, $(1,0)$, và $(0,1)$ và phương trình tham số đối với $E(x,y)$, 5 yếu tố chưa biết A, B, C, x_0, y_0 có thể được giải như sau:

$$A = (E(-1,0) + E(1,0) - 2*E(0,0))/2$$

$$B = (E(0,-1) + E(0,1) - 2*E(0,0))/2$$

$$x_0 = (E(-1,0) - E(1,0))/(2*(E(-1,0) + E(1,0) - 2*E(0,0)))$$

$$y_0 = (E(0,-1) - E(0,1))/(2*(E(0,-1) + E(0,1) - 2*E(0,0)))$$

$$C = \text{Dẫn xuất}_{\text{chi phí}} = E(0,0) - \frac{(E(-1,0) - E(1,0))^2}{8 * (E(-1,0) + E(1,0) - 2 * E(0,0))} - \frac{(E(0,-1) - E(0,1))^2}{8 * (E(0,-1) + E(0,1) - 2 * E(0,0))}$$

Mặt khác, nếu các giá trị chi phí ở nhiều hơn 5 vị trí là khả dụng, 5 yếu tố chưa biết có thể được giải nhờ sử dụng bình phương tối thiểu hoặc cách tiếp cận tương tự. Giá trị của C thu được sau đó trở thành chi phí được dẫn xuất.

Theo một cách thực hiện, hàm thứ hai có thể như sau:

$$\begin{aligned} \text{Dẫn xuất}_{\text{chi phí}} &= SAD[0] - \frac{(SAD[1] - SAD[2])^2}{K * (SAD[1] + SAD[2] - 2 * SAD[0])} \\ &\quad - \frac{(SAD[3] - SAD[4])^2}{K * (SAD[3] + SAD[4] - 2 * SAD[0])} \end{aligned}$$

Trong đó K lớn hơn 0 theo vô hướng và sad[0] đến sad[4] là N chi phí so khớp.

Theo một ví dụ, quy trình tinh chế vector chuyển động được thực hiện nhiều hơn một lần để tinh chế vector chuyển động thêm nữa. Trong ví dụ này, các vector chuyển động ban đầu trước tiên được tinh chế bởi quy trình tinh chế vector chuyển động, để thu nhận vector chuyển động đã tinh chế thứ nhất. Sau đó, sự tinh chế vector chuyển động được thực hiện một lần nữa, trong trường hợp này, vector chuyển động đã tinh chế thứ nhất được coi là các vector chuyển động ban đầu đối với sự tinh chế vector chuyển động thứ hai.

Vector chuyển động ban đầu được thu nhận ở 925, mà là đầu vào của bộ phận tinh chế vector chuyển động. Không gian tìm kiếm được xây dựng quanh vector chuyển động ban đầu bởi bộ phận tinh chế vector chuyển động (930). Theo một ví dụ, không gian tìm kiếm bao gồm các cặp vector chuyển động ứng viên, vector chuyển động thứ nhất của cặp tương ứng với hình ảnh tham chiếu thứ nhất và vector chuyển động thứ hai của cặp tương ứng với hình ảnh tham chiếu thứ hai. Các dự đoán thứ nhất tương ứng với mỗi cặp vector chuyển động ứng viên được thu nhận ở bước 910, nhờ áp dụng bộ lọc nội suy M điểm nối. Như là một phần của sự tinh chế vector chuyển động, chi phí so khớp được tính toán tương ứng với N cặp vector chuyển động trong không gian tìm kiếm (915). N chi phí so khớp này được sử dụng như là một phần của hai quy trình, quy trình thứ nhất là sự tinh

chế vector chuyển động (930), trong đó các chi phí so khớp được sử dụng để tính toán cặp vector chuyển động đã tinh chế (935) theo hàm mà lấy N chi phí so khớp làm đầu vào. Quy trình thứ hai là sự quyết định về việc liệu sự tinh chế dòng quang học (950) có được áp dụng hay không, trong đó quyết định này được lấy bởi 945. Sau khi vector chuyển động đã tinh chế được thu nhận, dự đoán thứ hai đối với khối hiện hành được thu nhận bởi (940). Nếu chi phí so khớp lớn hơn ngưỡng, thì sự tinh chế dòng quang học được áp dụng và việc dự đoán ở 940 được thay đổi bởi 950 để thu nhận dự đoán đã thay đổi (955-960). Dự đoán đã thay đổi thường khác về các giá trị mẫu với dự đoán thứ hai ở bước 940. Nếu chi phí so khớp nhỏ hơn ngưỡng, thì sự tinh chế dòng quang học không được áp dụng và dự đoán thứ hai được thiết đặt là đầu ra (dự đoán cuối cùng của khối hiện hành).

Theo phương án làm ví dụ thứ ba của sáng chế, các bước sau đây được áp dụng để thu nhận dự đoán đối với khối tạo mã hiện hành:

Bước 0: thu nhận cặp vector chuyển động ban đầu dựa trên thông tin chỉ báo trong dòng bit.

Bước 1: thu nhận tập hợp thứ nhất của các mẫu đã dự đoán dựa trên cặp MV ban đầu và bộ lọc nội suy M điểm nối.

Bước 2: thu nhận chi phí so khớp thứ nhất tương ứng với cặp MV ban đầu sử dụng tập hợp thứ nhất của các mẫu đã dự đoán

Bước 3: xác định liệu khối tạo mã hiện hành có thích hợp để thực hiện sự tinh chế vector chuyển động hay không.

Bước 4: Nếu khối tạo mã hiện hành được xác định là thích hợp để thực hiện MVR ở bước 3,

Bước 4a: thu nhận cặp MV đã tinh chế và chi phí so khớp tương ứng với cặp MV đã tinh chế theo cặp MV ban đầu và chi phí so khớp sử dụng quy trình tinh chế vector chuyển động.

Bước 4b: thu nhận tập hợp thứ hai của các mẫu đã dự đoán theo cặp MV đã tinh chế và bộ lọc nội suy K điểm nối.

Bước 4c: xác định liệu có thực hiện quy trình tinh chế dòng quang học theo chi phí

so khớp thứ hai hay không. Theo một ví dụ, chỉ phí so khớp được so sánh với ngưỡng, và quy trình tinh chế dòng quang học được thực hiện khi giá trị của chỉ phí so khớp lớn hơn hoặc bằng ngưỡng.

Bước 5: Nếu không (nếu khối tạo mã hiện hành được xác định là không thích hợp để thực hiện MVR ở bước 3),

Bước 5a: thu nhận tập hợp thứ hai của các mẫu đã dự đoán theo cặp MV ban đầu và bộ lọc nội suy K điểm nổi

Bước 5b: xác định liệu có thực hiện quy trình tinh chế dòng quang học theo chỉ phí so khớp thứ nhất hay không. Theo một ví dụ, chỉ phí so khớp được so sánh với ngưỡng, và quy trình tinh chế dòng quang học được thực hiện khi giá trị của chỉ phí so khớp lớn hơn hoặc bằng ngưỡng.

Bước 6: Khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện (ở bước 4c hoặc bước 5b), thì sự tinh chế dòng quang học được áp dụng với dự đoán thứ hai như là đầu vào và dự đoán thứ hai đã thay đổi như là đầu ra. Nếu xác định được là không phải như vậy, thì sự tinh chế dòng quang học không được áp dụng trên dự đoán thứ hai. Nói cách khác, khi xác định được rằng quy trình tinh chế dòng quang học cần được thực hiện, thì dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và theo quy trình tinh chế dòng quang học. Nếu không, dự đoán cuối cùng của khối tạo mã hiện hành được thu nhận theo dự đoán thứ hai và không áp dụng quy trình tinh chế dòng quang học.

Phương án này còn được minh họa trong lưu đồ trên Fig.9. Khối 1110 nhận cặp MV ban đầu đối với khối tạo mã hiện hành để dự đoán trong các tham chiếu L0 và L1. Khối 1110 tương ứng với bước 1, trong đó tập hợp thứ nhất của các mẫu đã dự đoán được thu nhận nhờ sử dụng cặp MV ban đầu và các mẫu tham chiếu được tái tạo của các hình ảnh L0 và L1. Khối 1120 tương ứng với bước 2, trong đó chỉ phí so khớp thứ nhất (hoặc chuẩn đo không tương tự như SAD) được đánh giá giữa tập hợp thứ nhất của khối đã dự đoán của các mẫu tương ứng với cặp MV ban đầu (như được mô tả trong phần MVR khái quát). Khối 1130 tương ứng với bước 3, trong đó các điều kiện để hợp lệ của khối tạo mã hiện hành để thực hiện MVR được kiểm tra. Khối 1140 tương ứng với bước 4a, trong đó nếu khối tạo mã hiện hành được thấy là thích hợp để thực hiện

MVR, thì cặp MV đã tinh chế được thu nhận bằng cách thực hiện MVR (như được mô tả trong phần MVR khái quát) và chi phí so khớp thứ hai (hoặc chuẩn đo không tương tự) tương ứng với cặp MV đã tinh chế được thu nhận. Khối 1150 tương ứng với bước 4b, trong đó tập hợp thứ hai của các mẫu đã dự đoán được thu nhận nhờ sử dụng bộ lọc nội suy K điểm nổi (theo các chiều ngang và thẳng đứng) nhờ sử dụng cặp MV đã tinh chế. Khối 1160 tương ứng với bước 4c, trong đó điều được kiểm tra là liệu chi phí so khớp thứ hai có nhỏ hơn ngưỡng định trước dưới đây mà dòng quang học song dự đoán dựa trên tinh chế và song dự đoán được bỏ qua. Khối 1180 tương ứng với bước 5a, trong đó khối tạo mã hiện hành bỏ qua MVR và thu nhận tập hợp thứ hai của các mẫu đã dự đoán nhờ sử dụng bộ lọc nội suy K điểm nổi sử dụng cặp MV ban đầu. Khối 1185 tương ứng với bước 5b, trong đó điều được kiểm tra là liệu chi phí so khớp thứ nhất có nhỏ hơn ngưỡng định trước dưới đây hay không mà dưới đó BPOF được bỏ qua. Các khối 1170 và 1195 tương ứng với một phần của bước 6, trong đó nếu việc kiểm tra ở bước 4c hoặc bước 5b chỉ báo tương ứng rằng chi phí so khớp thứ hai hoặc thứ nhất nhỏ hơn ngưỡng định trước mà dưới đó BPOF được bỏ qua, thì bình quân gia quyền song dự đoán không có BPOF được thực hiện nhờ sử dụng tập hợp thứ hai của các mẫu đã dự đoán. Khối 1175 tương ứng với một phần của bước 6, trong đó nếu việc kiểm tra ở bước 4c hoặc bước 5b chỉ báo rằng chi phí so khớp thứ hai hoặc thứ nhất không nhỏ hơn ngưỡng định trước mà dưới đó BPOF được bỏ qua, dòng quang học được ước lượng được thu nhận và song dự đoán cuối cùng được thu nhận nhờ sử dụng tập hợp thứ hai của các mẫu đã dự đoán, các gradient của tập hợp thứ hai của các mẫu đã dự đoán, và dòng quang học đã ước lượng.

Cần lưu ý rằng bằng cách sử dụng chi phí so khớp thứ nhất hoặc thứ hai mà được tính trên khối con của các mẫu trong đơn vị mã hóa hiện hành như được yêu cầu bởi quy trình tinh chế vectơ chuyển động để xác định việc chấm dứt sớm quy trình tinh chế dựa trên dòng quang học song dự đoán, quyết định bỏ qua hoặc thực hiện BPOF có thể thay đổi từ một khối con MVR sang một khối con MVR khác trong một đơn vị tạo mã. BPOF sẽ được áp dụng hoặc được bỏ qua đối với tất cả các đơn vị áp dụng BPOF (ví dụ, mức điểm ảnh, hoặc khối 4x4 của mức của các mẫu) trong khối con dựa trên việc xác định được thực hiện ở bước-4c hoặc bước-5.

Theo các phương án nhất định, có thể thực hiện việc chấm dứt sớm bổ sung đối

với mỗi đơn vị áp dụng BPOF trong khối con bằng cách thu nhận các chi phí so khớp một phần tương ứng với mỗi đơn vị áp dụng BPOF trong khối con MVR.

Giá trị ngưỡng định trước thường được chọn dưới dạng giá trị ngưỡng cho một mẫu mà phụ thuộc vào độ sâu bit của dự đoán thứ nhất hoặc tập hợp thứ nhất của các mẫu đã dự đoán. Ví dụ, nếu giá trị mẫu dự đoán thứ nhất thu nhận được nhờ sử dụng nội suy song tuyến tính (2 điểm nối) bị ràng buộc ở độ sâu bit b , ngưỡng cho một mẫu được tính là $k \cdot 2^{(b-10)}$, và số lượng mẫu mà chi phí so khớp được tính là N , thì giá trị ngưỡng định trước mà tương phản với nó chi phí so khớp đối với khối con hiện hành được so sánh sẽ là $k \cdot N \cdot 2^{(10-b)}$. Các giá trị mẫu đối với k là 2 (đối với độ sâu bit là 10), N là $8 \times 16 = 128$, và b là 8. Do chi phí so khớp ở cặp MV ứng viên đã biết có thể được tính toán với tập hợp các mẫu đã dự đoán thứ nhất giảm một phần mười, giá trị của N sẽ được sử dụng theo đó. Ví dụ, nếu các hàng số lẻ của khối 8×16 của các mẫu đã dự đoán được sử dụng, thì N sẽ được tính là $8 \times 8 = 64$.

Theo các phương án của sáng chế, phương pháp chấm dứt sớm được đề xuất để bỏ qua có điều kiện việc áp dụng quy trình tinh chế dòng quang học, việc áp dụng quy trình tinh chế dòng quang học được coi là mạnh về tính toán. Kết quả là, thời gian giải mã trung bình được giảm bớt.

Ngoài ra, điều kiện để bỏ qua có điều kiện việc áp dụng dòng quang học được xác định dựa trên các thông số mà được tính toán bởi một quy trình khác (tính toán các chi phí so khớp trong quá trình tinh chế vector chuyển động). Do các giá trị đã được tính toán được sử dụng, nên không có các phép tính bổ sung cần được thực hiện.

Cụ thể, sáng chế đề xuất phương pháp tạo mã video được thực hiện trong thiết bị giải mã hoặc thiết bị mã hóa như được minh họa trên Fig.10. Phương pháp này bao gồm các bước sau đây mà có thể được thực hiện theo thứ tự đã biết. Vector chuyển động ban đầu được thu nhận đối với khối hiện hành 1210. Khối hiện hành có thể là khối tạo mã hiện hành. Các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành được thu nhận dựa trên các vector chuyển động ban đầu 1220. Chi phí so khớp được tính toán theo các dự đoán thứ nhất 1230.

Sau khi chi phí so khớp thứ nhất được thu nhận, điều được xác định là liệu quy trình tinh chế dòng quang học có nên được thực hiện hay không 1240, theo ít nhất một

điều kiện thiết đặt trước, ít nhất một điều kiện thiết đặt trước bao gồm điều kiện về việc liệu chi phí so khớp được tính toán (ví dụ, trên phương diện đo lường sự tương tự; xem phần mô tả trên đây) có bằng hoặc lớn hơn ngưỡng định trước hay không. Quy trình tinh chế dòng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành được thực hiện 1250, khi xác định được rằng quy trình tinh chế dòng quang học nên được thực hiện. Khi xác định được rằng quy trình tinh chế dòng quang học không nên được thực hiện, các chi phí tính toán có thể được tiết kiệm bằng cách bỏ qua quy trình tinh chế dòng quang học.

Phương pháp này có thể được thực thi trong các máy được mô tả trên đây dựa vào các hình vẽ từ Fig.1a đến Fig.5.

Cụ thể, phương pháp có thể được thực thi trong ngữ cảnh của quy trình tinh chế vector chuyển động phía bộ giải mã. Các đầu vào của quy trình là:

vị trí độ chói (x_{Sb} , y_{Sb}) quy định mẫu trên cùng bên trái của khối con tạo mã hiện hành so với mẫu độ chói trên cùng bên trái của hình ảnh hiện hành,

biến số $sbWidth$ quy định độ rộng của khối con tạo mã hiện hành trong các mẫu độ chói,

biến số $sbHeight$ quy định độ cao của khối con tạo mã hiện hành trong các mẫu độ chói,

các vector chuyển động độ chói trong độ chính xác thập phân-mẫu $1/16$ $mvL0$ và $mvL1$,

các mảng mẫu hình ảnh tham chiếu độ chói được chọn $refPicL0_L$ và $refPicL1_L$.

Các đầu ra của quy trình này là: các vector chuyển động độ chói delta $dMvL0$ và $dMvL$ và biến số $dmvrSad$ quy định tổng các sai phân tuyệt đối tối thiểu của các dự đoán thứ nhất (ví dụ, tính toán SAD được mô tả trên đây).

Vector chuyển động độ chói delta $dMvL0$ có thể được dẫn xuất bởi $dMvL0[0] += 16 * intOffX$ và $dMvL0[1] += 16 * intOffY$, trong đó $intOffX$ và $intOffY$ lần lượt là các độ lệch mẫu số nguyên theo chiều x và y. Hơn nữa, vector chuyển động độ chói delta $dMvL$ có thể được tính toán như là $dMvL1[0] = -dMvL0[0]$ và $dMvL1[1] = -dMvL0[1]$.

Các giá trị mẫu độ chói dự đoán thứ nhất được dẫn xuất bởi nội suy song tuyến tính mẫu thập phân. Trong quy trình giải mã các khối được dự đoán liên ảnh, quy trình dự đoán mẫu dòng quang học hai chiều có thể hoặc có thể không được áp dụng. Nếu nó không được áp dụng thì quy trình dự đoán mẫu có trọng số được áp dụng vào các dự đoán thứ hai được tinh chế thu được dựa trên các vector chuyển động đã tinh chế. Nếu quy trình dự đoán mẫu dòng quang học hai chiều được áp dụng, thì nó nhận các dự đoán thứ hai thu được dựa trên các vector chuyển động đã tinh chế như là đầu vào và kết xuất các dự đoán cuối cùng.

Còn có thể được sử dụng để báo hiệu liệu quy trình dự đoán mẫu dòng quang học hai chiều có thể được áp dụng hay không. Ví dụ, có thể xem xét điều kiện cần đối với quy trình dự đoán mẫu dòng quang học hai chiều để được thực hiện mà còn là đúng (TRUE). Tuy nhiên, điều kiện cần này có thể không phải là điều kiện đủ để quy trình dự đoán mẫu dòng quang học hai chiều được thực hiện. Điều kiện đủ có thể là cả còn là đúng (TRUE) và chi phí so khớp được mô tả trên đây bằng hoặc lớn hơn ngưỡng định trước. Ví dụ, chi phí so khớp có thể được xác định dựa trên biến số $dmvrSad$ quy định tổng các sai phân tuyệt đối tối thiểu của các dự đoán thứ nhất.

Mặt khác, có thể xem xét điều kiện đủ để không thực hiện quy trình dự đoán mẫu dòng quang học nhưng thực hiện quy trình dự đoán mẫu có trọng số nếu còn là sai (FALSE).

Hơn nữa, sáng chế đề xuất thiết bị 1300 để sử dụng trong bộ mã hóa ảnh và/hoặc bộ giải mã ảnh như được minh họa trên Fig.11. Thiết bị 1300, theo phương án làm ví dụ này, bao gồm bộ phận vector chuyển động ban đầu 1310 mà được tạo cấu hình để thu nhận các vector chuyển động ban đầu đối với khối hiện hành. Hơn nữa, thiết bị 1300 bao gồm bộ phận dự đoán 1320 mà được tạo cấu hình để thu nhận các dự đoán thứ nhất đối với giá trị mẫu trong khối hiện hành dựa trên các vector chuyển động ban đầu. Hơn nữa, thiết bị 1300 bao gồm bộ phận tính toán chi phí so khớp 1330 mà được tạo cấu hình để tính toán chi phí so khớp theo các dự đoán thứ nhất.

Thiết bị 1300 bao gồm bộ phận xác định quy trình tinh chế dòng quang học 1340 mà được tạo cấu hình để xác định liệu quy trình tinh chế dòng quang học có nên được thực hiện hay không, theo ít nhất một điều kiện thiết đặt trước, ít nhất một điều kiện

thiết đặt trước bao gồm điều kiện về việc liệu chi phí so khớp được tính toán có bằng hoặc lớn hơn ngưỡng hay không. Hơn nữa, thiết bị 1300 bao gồm bộ phận thực hiện quy trình tinh chế dòng quang học 1350 mà được tạo cấu hình để thực hiện quy trình tinh chế dòng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành, khi xác định được rằng quy trình tinh chế dòng quang học sẽ được thực hiện.

Các toán tử toán học

Các toán tử toán học được sử dụng trong sáng chế tương tự như các toán tử toán học được sử dụng trong ngôn ngữ lập trình C. Tuy nhiên, các kết quả của phép chia số nguyên và các phép toán dịch số học được xác định chính xác hơn, và các phép toán bổ sung được xác định, như mũ hóa và chia giá trị thực. Các quy ước đánh số và đếm thường bắt đầu từ 0, ví dụ, "thứ nhất" tương đương với thứ 0, "thứ hai" tương đương với thứ 1, v.v..

Các toán tử số học

Các toán tử số học sau đây được định nghĩa như sau:

- + Cộng
- Trừ (dưới dạng toán tử hai đối số) hoặc phủ định (dưới dạng toán tử tiếp đầu ngữ một ngôi)
- * Phép nhân, bao gồm phép nhân ma trận
- x^y Mũ hóa. Quy định x lũy thừa y . Trong các ngữ cảnh khác, cách ký hiệu này được sử dụng cho chỉ số trên mà không được dự định để hiểu là mũ hóa.
- / Phép chia nguyên với việc cắt bỏ phần kết quả về phía không (zero). Ví dụ, $7 / 4$ và $-7 / -4$ được cắt bỏ thành 1 và $-7 / 4$ và $7 / -4$ được cắt bỏ thành -1.
- ÷ Được sử dụng để thể hiện phép chia trong các phương trình toán học trong đó không có việc cắt bỏ hoặc làm tròn được dự định.

$\frac{x}{y}$ Được sử dụng để thể hiện phép chia trong các phương trình toán học trong đó không có việc cắt bỏ hoặc làm tròn được dự định.

$\sum_{i=x}^y f(i)$ Tổng của $f(i)$ với i lấy tất cả các giá trị nguyên từ x cho đến và bao gồm y .

$x \% y$ Phép chia lấy dư. Phần dư của x được chia bởi y , được xác định chỉ cho các số nguyên x và y với $x \geq 0$ và $y > 0$.

Các toán tử logic

Các toán tử logic sau đây được định nghĩa như sau:

$x \&\& y$ Logic luận lý (Boolean) "and" ("và") của x và y

$x || y$ Logic luận lý (Boolean) "or" ("hoặc") của x và y

! Logic luận lý "not" (phủ định)

$x ? y : z$ Nếu x là TRUE (đúng) hoặc không bằng 0, đánh giá giá trị của y ; nếu không, đánh giá giá trị của z .

Các toán tử quan hệ

Các toán tử quan hệ sau đây được định nghĩa như sau:

> Lớn hơn

>= Lớn hơn hoặc bằng

< Nhỏ hơn

<= Nhỏ hơn hoặc bằng

= = Bằng

!= Không bằng

Khi toán tử quan hệ được áp dụng vào phần tử cú pháp hoặc biến số mà đã được ấn định giá trị "na" (not applicable - không áp dụng được), giá trị "na" được xử lý như là giá trị phân biệt đối với phần tử cú pháp hoặc biến số. Giá trị "na" được coi là không bằng giá

trị khác bất kỳ.

Các toán tử theo từng bit

Các toán tử theo từng bit sau đây được định nghĩa như sau:

- & "and" ("và") theo từng bit. Khi thao tác trên các đối số nguyên, thao tác trên biểu diễn bù hai của giá trị nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit có trọng số cao hơn bằng 0.
- | "or" ("hoặc") theo từng bit. Khi thao tác trên các đối số nguyên, thao tác trên biểu diễn bù hai của giá trị nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit có trọng số cao hơn bằng 0.
- ^ "exclusive or" ("hoặc tuyệt đối") theo từng bit. Khi thao tác trên các đối số nguyên, thao tác trên biểu diễn bù hai của giá trị nguyên. Khi thao tác trên đối số nhị phân mà chứa ít bit hơn so với đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung các bit có trọng số cao hơn bằng 0.
- $x \gg y$ Dịch chuyển phải số học của biểu diễn nguyên bù hai của x theo y chữ số nhị phân. Hàm này được định nghĩa chỉ đối với các giá trị nguyên không âm của y . Các bit được dịch chuyển vào các bit có trọng số lớn nhất (các MSB) do phép dịch chuyển phải có giá trị bằng MSB của x trước phép toán dịch chuyển.
- $x \ll y$ Dịch chuyển trái số học của biểu diễn nguyên bù hai của x theo y chữ số nhị phân. Hàm này được định nghĩa chỉ đối với các giá trị nguyên không âm của y . Các bit được dịch chuyển vào các bit có trọng số nhỏ nhất (các LSB) do phép dịch chuyển trái có giá trị bằng 0.

Các toán tử gán

Các toán tử số học sau đây được định nghĩa như sau:

= Toán tử gán

- ++ Tăng, tức là, $x++$ tương đương với $x = x + 1$; khi được sử dụng trong chỉ số mảng, đánh giá đến giá trị của biến số trước phép toán tăng.
- Giảm, tức là, $x--$ tương đương với $x = x - 1$; khi được sử dụng trong chỉ số mảng, đánh giá đến giá trị của biến số trước phép toán giảm.
- += Tăng theo lượng được quy định, tức là, $x += 3$ tương đương với $x = x + 3$, và $x += (-3)$ tương đương với $x = x + (-3)$.
- Giảm theo lượng được quy định, tức là, $x -= 3$ tương đương với $x = x - 3$, và $x -= (-3)$ tương đương với $x = x - (-3)$.

Ký hiệu khoảng

Ký hiệu sau đây được sử dụng để quy định khoảng của các giá trị:

$x = y..z$ x lấy trên các giá trị nguyên bắt đầu từ y đến z , bao gồm cả y và z , với x , y , và z là các số nguyên và z lớn hơn y .

Các hàm toán học

Các hàm toán học sau đây được định nghĩa:

$$\text{Abs}(x) = \begin{cases} x & ; \quad x \geq 0 \\ -x & ; \quad x < 0 \end{cases}$$

$\text{Asin}(x)$ hàm sin lượng giác ngược, thao tác trên đối số x mà nằm trong khoảng từ $-1,0$ đến $1,0$, bao gồm cả hai giá trị này, với giá trị đầu ra nằm trong khoảng từ $-\pi/2$ đến $\pi/2$, bao gồm cả hai giá trị này, theo đơn vị radian

$\text{Atan}(x)$ hàm tang lượng giác ngược, thao tác trên đối số x , với giá trị đầu ra nằm trong khoảng từ $-\pi/2$ đến $\pi/2$, bao gồm cả hai giá trị này, theo đơn vị radian

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; \quad x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; \quad x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; \quad x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; \quad x == 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \quad \text{nếu không} \end{cases}$$

$\text{Ceil}(x)$ số nguyên nhỏ nhất lớn hơn hoặc bằng x .

$$\text{Clip1}_Y(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_Y) - 1, x)$$

$$\text{Clip1}_C(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_C) - 1, x)$$

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

$\text{Cos}(x)$ hàm cosin lượng giác thao tác trên đối số x theo đơn vị radian.

$\text{Floor}(x)$ số nguyên lớn nhất nhỏ hơn hoặc bằng x .

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; \quad b - a \geq d/2 \\ c - d & ; \quad a - b > d/2 \\ c & ; \quad \text{nếu không} \end{cases}$$

$\text{Ln}(x)$ lôgarit tự nhiên của x (lôgarit cơ số e , trong đó e là hằng số cơ số của lôgarit tự nhiên 2.718 281 828...).

$\text{Log2}(x)$ lôgarit cơ số 2 của x .

$\text{Log10}(x)$ lôgarit cơ số 10 của x .

$$\text{Min}(x, y) = \begin{cases} x & ; \quad x \leq y \\ y & ; \quad x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0,5)$$

$$\text{Sign}(x) = \begin{cases} 1 & ; \quad x > 0 \\ 0 & ; \quad x == 0 \\ -1 & ; \quad x < 0 \end{cases}$$

$\text{Sin}(x)$ hàm sin lượng giác thao tác trên đối số x theo đơn vị radian

$$\text{Sqrt}(x) = \sqrt{x}$$

$$\text{Swap}(x, y) = (y, x)$$

$\text{Tan}(x)$ hàm tang lượng giác thao tác trên đối số x theo đơn vị radian

Thứ tự ưu tiên phép toán

Khi thứ tự ưu tiên trong biểu thức không được chỉ ra rõ ràng bằng cách sử dụng ngoặc, các quy tắc sau đây áp dụng:

- Các phép toán có mức ưu tiên cao hơn được đánh giá là trước phép toán bất kỳ có mức ưu tiên thấp hơn.
- Các phép toán có cùng mức ưu tiên được đánh giá là tuần tự từ trái sang phải.

Bảng dưới đây quy định mức ưu tiên của các phép toán từ cao nhất đến thấp nhất; vị trí cao hơn trong bảng chỉ ra mức ưu tiên cao hơn.

Đối với các toán tử mà cũng được sử dụng trong ngôn ngữ lập trình C, thứ tự ưu tiên được sử dụng trong bản mô tả này giống như được sử dụng trong ngôn ngữ lập trình C.

Bảng: Mức ưu tiên phép toán từ cao nhất (ở đỉnh bảng) đến thấp nhất (ở đáy bảng)

các phép toán (với các toán hạng x, y, và z)
"x++", "x--"
"!x", "-x" (như là toán tử tiếp đầu ngữ một ngôi)
x^y
"x * y", "x / y", "x ÷ y", " $\frac{x}{y}$ ", "x % y"
"x + y", "x - y" (như là toán tử hai đối số), " $\sum_{i=x}^y f(i)$ "
"x << y", "x >> y"
"x < y", "x <= y", "x > y", "x >= y"
"x == y", "x != y"
"x & y"
"x y"
"x && y"
"x y"
"x ? y : z"
"x..y"
"x = y", "x += y", "x -= y"

Phần mô tả bằng văn bản của các phép toán logic

Trong phần văn bản này, câu lệnh của các phép toán logic như sẽ được mô tả toán học theo hình thức sau đây:

nếu (điều kiện 0)

câu lệnh 0

còn nếu (điều kiện 1)

câu lệnh 1

...

còn không /* chú thích mang tính thông tin về điều kiện còn lại */

câu lệnh n

có thể được mô tả theo cách sau đây:

... như sau / ... phần sau đây áp dụng:

- Nếu điều kiện 0, câu lệnh 0
- Nếu không, nếu điều kiện 1, câu lệnh 1
- ...
- Nếu không (chú thích mang tính thông tin về điều kiện còn lại), câu lệnh n

Mỗi "Nếu ... Nếu không, nếu ... Nếu không, ..." câu lệnh trong văn bản này được giới thiệu với "... như sau" hoặc "... phần sau đây áp dụng" được theo ngay sau bởi "Nếu ...". Điều kiện cuối cùng của "Nếu ... Nếu không, nếu ... Nếu không, ..." luôn là "Nếu không, ...". Được đan xen "Nếu ... Nếu không, nếu ... Nếu không, ..." các câu lệnh có thể được nhận dạng bằng cách so khớp "... như sau" hoặc "... phần sau đây áp dụng" với kết thúc "Nếu không, ...".

Trong phần văn bản này, câu lệnh của các phép toán logic như sẽ được mô tả toán học theo hình thức sau đây:

nếu (điều kiện 0a && điều kiện 0b)
 câu lệnh 0
 còn không nếu (điều kiện 1a || điều kiện 1b)
 câu lệnh 1
 ...
 còn không
 câu lệnh n

có thể được mô tả theo cách sau đây:

... như sau / ... phần sau đây áp dụng:

- Nếu tất cả các điều kiện sau đây là đúng, câu lệnh 0:
 - điều kiện 0a
 - điều kiện 0b
- Nếu không, nếu một hoặc nhiều điều kiện sau đây là đúng, câu lệnh 1:
 - điều kiện 1a

- điều kiện 1b
- ...
- Nếu không, câu lệnh n

Trong phần văn bản này, câu lệnh của các phép toán logic như sẽ được mô tả toán học theo hình thức sau đây:

nếu (điều kiện 0)

câu lệnh 0

nếu (điều kiện 1)

câu lệnh 1

có thể được mô tả theo cách sau đây:

Khi điều kiện 0, câu lệnh 0

Khi điều kiện 1, câu lệnh 1.

Mặc dù các phương án của sáng chế đã được mô tả sơ bộ dựa trên việc tạo mã video, nhưng cần lưu ý rằng các phương án của hệ thống tạo mã 10, bộ mã hóa 20 và bộ giải mã 30 (và tương ứng là hệ thống 10) và các phương án khác được mô tả ở đây cũng có thể được tạo cấu hình để xử lý hoặc tạo mã hình ảnh tĩnh, tức là xử lý hoặc tạo mã hình ảnh riêng độc lập với hình ảnh trước hoặc liên tiếp bất kỳ như trong việc tạo mã video. Nói chung, chỉ các bộ phận dự đoán liên ảnh 244 (bộ mã hóa) và 344 (bộ giải mã) có thể không khả dụng trong trường hợp việc tạo mã xử lý hình ảnh được giới hạn ở một hình ảnh 17. Tất cả các chức năng khác (cũng được gọi là các công cụ hoặc các kỹ thuật) của bộ mã hóa video 20 và bộ giải mã video 30 có thể được sử dụng tương đương để xử lý hình ảnh tĩnh, ví dụ tính toán dư 204/304, biến đổi 206, lượng tử hóa 208, lượng tử hóa ngược 210/310, (biến đổi ngược 212/312, phân chia 262/362, dự đoán trong ảnh 254/354, và/hoặc lọc vòng 220, 320, và mã hóa entropi 270 và giải mã entropi 304.

Các phương án, ví dụ của bộ mã hóa 20 và bộ giải mã 30, và các chức năng được mô tả ở đây, ví dụ dựa vào bộ mã hóa 20 và bộ giải mã 30, có thể được thực hiện trong phần cứng, phần mềm, phần sụn, hoặc sự kết hợp bất kỳ của chúng. Nếu được thực hiện trong phần mềm, các chức năng này có thể được lưu trữ trên phương tiện đọc được bằng máy tính hoặc được truyền qua phương tiện truyền thông dưới dạng một hoặc nhiều lệnh

hoặc mã và được thực hiện bởi bộ phận xử lý dựa trên phần cứng. Phương tiện đọc được bằng máy tính có thể bao gồm phương tiện lưu trữ đọc được bằng máy tính, mà tương ứng với phương tiện hữu hình như phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông bao gồm phương tiện bất kỳ hỗ trợ truyền chương trình máy tính từ nơi này đến nơi khác, ví dụ, theo giao thức truyền thông. Theo cách này, phương tiện đọc được bằng máy tính nói chung có thể tương ứng với (1) phương tiện lưu trữ đọc được bằng máy tính hữu hình mà là lâu dài hoặc (2) phương tiện truyền thông như sóng tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ liệu có thể là phương tiện khả dụng bất kỳ mà có thể được truy cập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để lấy các lệnh, mã và/hoặc các cấu trúc dữ liệu để thực thi các kỹ thuật được mô tả trong sáng chế. Sản phẩm chương trình máy tính có thể bao gồm phương tiện đọc được bằng máy tính.

Ví dụ, và không phải giới hạn, phương tiện lưu trữ đọc được bằng máy tính như vậy có thể bao gồm RAM, ROM, EEPROM, CD-ROM hoặc bộ lưu trữ đĩa quang khác, bộ lưu trữ đĩa từ, hoặc các thiết bị lưu trữ từ tính khác, bộ nhớ tác động nhanh (flash memory), hoặc phương tiện khác bất kỳ mà có thể được sử dụng để lưu trữ mã chương trình mong muốn ở dạng các lệnh hoặc các cấu trúc dữ liệu và có thể được truy cập bởi máy tính. Ngoài ra, kết nối bất kỳ được gọi là phương tiện đọc được bằng máy tính. Ví dụ, nếu các lệnh được truyền từ vị trí web, máy chủ, hoặc nguồn ở xa khác sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn đôi, đường thuê bao số (DSL), hoặc các kỹ thuật vô tuyến như hồng ngoại, radiô, và vi sóng, thì cáp đồng trục, cáp sợi quang, cáp xoắn đôi, DSL, hoặc các kỹ thuật vô tuyến như hồng ngoại, radiô, và vi sóng có trong định nghĩa về phương tiện. Tuy nhiên, nên hiểu rằng phương tiện lưu trữ đọc được bằng máy tính và phương tiện lưu trữ dữ liệu không bao gồm các kết nối, các sóng mang, các tín hiệu, hoặc các phương tiện tạm thời khác, mà thay vào đó được hướng đến phương tiện lưu trữ lâu dài, hữu hình. Đĩa (disk) và đĩa (disc), như được sử dụng ở đây, bao gồm đĩa (disc) compact (CD), đĩa laser, đĩa quang, đĩa đa năng số (DVD), đĩa mềm và đĩa Blu-ray, trong đó các đĩa (disk) thường tái tạo dữ liệu về mặt từ tính, trong khi các đĩa (disc) tái tạo dữ liệu về mặt quang học bằng laser. Các sự kết hợp của các nội dung trên đây cũng sẽ có trong phạm vi của phương tiện đọc được bằng máy tính.

Các lệnh có thể được thực hiện bởi một hoặc nhiều bộ xử lý, như một hoặc nhiều bộ xử lý tín hiệu số (digital signal processor, DSP), các bộ vi xử lý đa năng, các mạch

tích hợp chuyên dụng (application specific integrated circuit, ASIC), các mảng logic lập trình được dạng trường (FPGA), hoặc các loại tương đương được tích hợp hoặc mạch logic rời rạc. Theo đó, thuật ngữ “bộ xử lý”, như được sử dụng ở đây có thể đề cập đến cấu trúc bất kỳ trong số cấu trúc trên đây hoặc cấu trúc khác bất kỳ thích hợp để thực hiện các kỹ thuật được mô tả ở đây. Ngoài ra, theo một số khía cạnh, chức năng được mô tả ở đây có thể được cung cấp trong phần cứng chuyên dụng và/hoặc các môđun phần mềm được tạo cấu hình để mã hóa và giải mã, hoặc được kết hợp trong codec kết hợp. Ngoài ra, các kỹ thuật có thể được thực hiện đầy đủ trong một hoặc nhiều mạch hoặc phần tử logic.

Các kỹ thuật của sáng chế có thể được thực hiện trong rất nhiều loại thiết bị hoặc máy, bao gồm máy cầm tay vô tuyến, mạch tích hợp (IC) hoặc tập hợp các IC (ví dụ, bộ chip). Các thành phần, các môđun, hoặc các bộ phận khác nhau được mô tả trong sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật được bộc lộ, mà không nhất thiết phải đòi hỏi thực hiện bởi các bộ phận phần cứng khác nhau. Thêm nữa, như được mô tả trên đây, các bộ phận khác nhau có thể được kết hợp trong bộ phận phần cứng codec hoặc được cung cấp bởi tập hợp các bộ phận phần cứng liên động, bao gồm một hoặc nhiều bộ xử lý như được mô tả trên đây, kết hợp với phần mềm và/hoặc phần sụn thích hợp.

YÊU CẦU BẢO HỘ

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

thu nhận cặp vector chuyển động ban đầu đối với khối hiện hành;

thu nhận (1110) tập hợp thứ nhất của các mẫu đã dự đoán đối với các mẫu trong khối hiện hành dựa trên cặp vector chuyển động ban đầu;

tính toán (1120) chi phí so khớp thứ nhất theo tập hợp thứ nhất của các mẫu đã dự đoán, trong đó chi phí so khớp thứ nhất là tổng các sai phân tuyệt đối (sum of absolute differences, SAD) giữa hai khối dự đoán thu nhận được nhờ sử dụng vector chuyển động ban đầu thứ nhất và vector chuyển động ban đầu thứ hai tương ứng của cặp vector chuyển động ban đầu;

xác định (1130) liệu khối hiện hành có thích hợp để thực hiện sự tinh chế vector chuyển động (motion vector refinement, MVR) hay không;

nếu khối hiện hành được xác định là thích hợp để thực hiện MVR:

thu nhận (1140) cặp vector chuyển động đã tinh chế và chi phí so khớp thứ hai tương ứng với cặp vector chuyển động đã tinh chế bằng cách thực hiện MVR nhờ sử dụng cặp vector chuyển động ban đầu;

thu nhận (1150) tập hợp thứ hai của các mẫu đã dự đoán đối với các mẫu trong khối hiện hành nhờ sử dụng bộ lọc nội suy K điểm nối (K-tap interpolation filter) sử dụng cặp vector chuyển động đã tinh chế;

xác định (1160) liệu quy trình tinh chế luồng quang học có nên được thực hiện hay không, theo ít nhất một điều kiện thiết đặt trước, ít nhất một điều kiện thiết đặt trước bao gồm điều kiện về việc liệu chi phí so khớp thứ hai có nhỏ hơn ngưỡng hay không;

thực hiện (1175) quy trình tinh chế luồng quang học để thu nhận dự đoán liên ảnh cuối cùng đối với giá trị mẫu trong khối hiện hành, khi xác định được rằng chi phí so khớp thứ hai không nhỏ hơn ngưỡng;

trong đó bước thực hiện quy trình tinh chế luồng quang học bao gồm: thực hiện sự tinh chế luồng quang học dựa trên tập hợp thứ hai của các mẫu đã dự đoán;

trong đó chỉ khi xác định được rằng quy trình tinh chế luồng quang học không nên

được thực hiện, thì dự đoán liên ảnh cuối cùng được thu nhận bởi tổng có trọng số của tập hợp thứ hai của các mẫu đã dự đoán (1170).

2. Phương pháp theo một điểm trong số các điểm trên đây, trong đó tập hợp thứ nhất của các mẫu đã dự đoán đối với các mẫu trong khối hiện hành được thu nhận dựa trên bộ lọc nội suy thứ nhất.

3. Phương pháp theo điểm 2, trong đó bộ lọc nội suy thứ nhất là bộ lọc nội suy song tuyến tính.

4. Phương pháp theo điểm 1, trong đó bộ lọc nội suy K điểm nối là bộ lọc nội suy 6 điểm nối hoặc 8 điểm nối.

5. Phương pháp theo một điểm trong số các điểm trên đây, trong đó giá trị ngưỡng được thu nhận theo kích thước của khối hiện hành.

6. Phương pháp theo một điểm trong số các điểm trên đây, trong đó khối hiện hành là khối tạo mã hoặc khối con.

7. Phương pháp theo một điểm trong số các điểm trên đây, phương pháp này còn bao gồm bước tạo khối dự đoán liên ảnh bao gồm dự đoán liên ảnh cuối cùng đối với các mẫu trong khối hiện hành.

8. Bộ mã hóa (20) bao gồm mạch xử lý để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

9. Bộ giải mã (30) bao gồm mạch xử lý để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

10. Phương tiện lưu trữ đọc được bằng máy tính lưu trữ sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

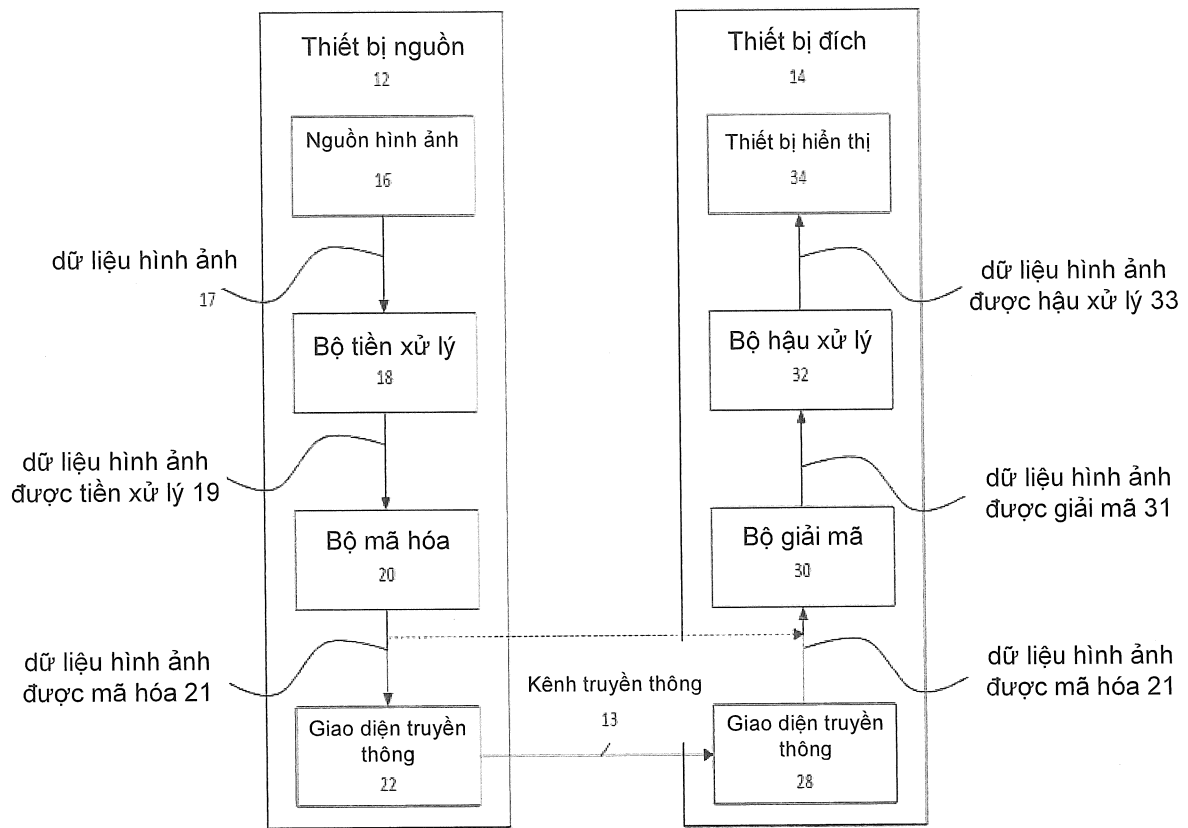


Fig.1A

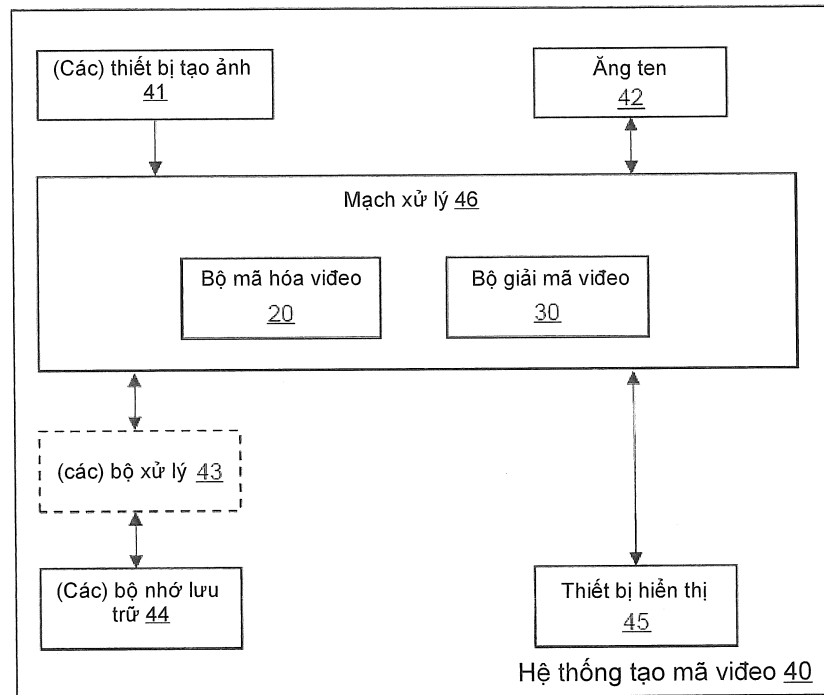


Fig.1B

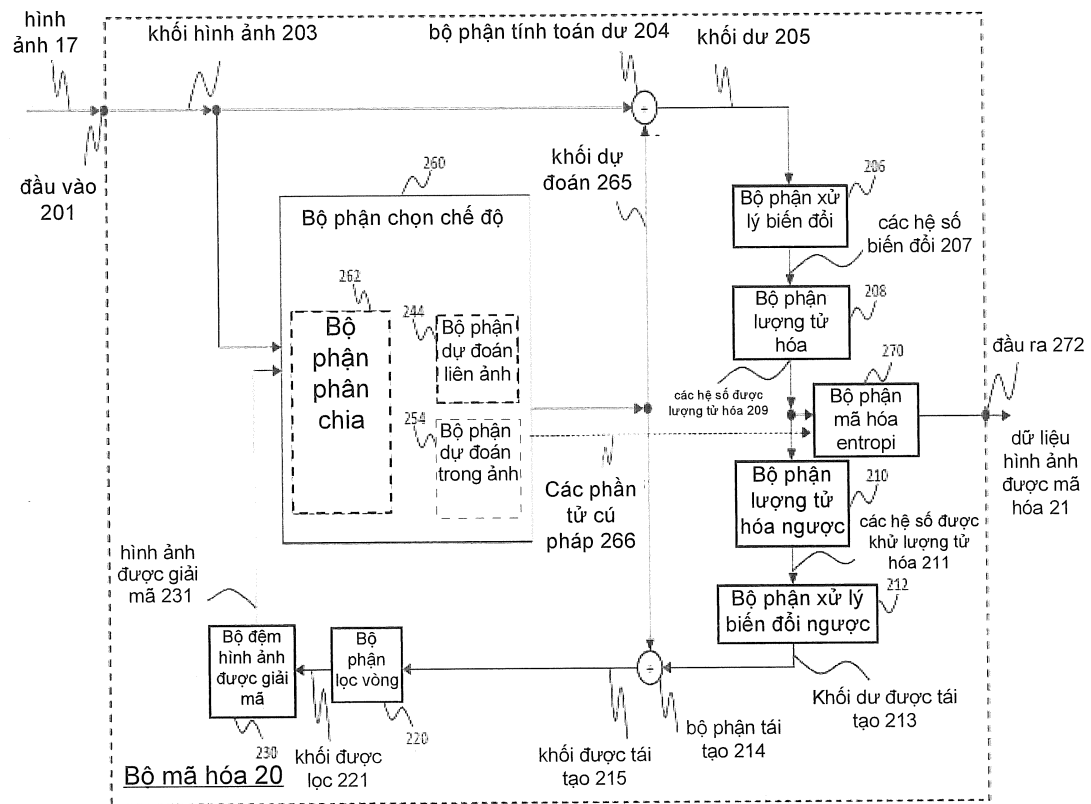


Fig.2

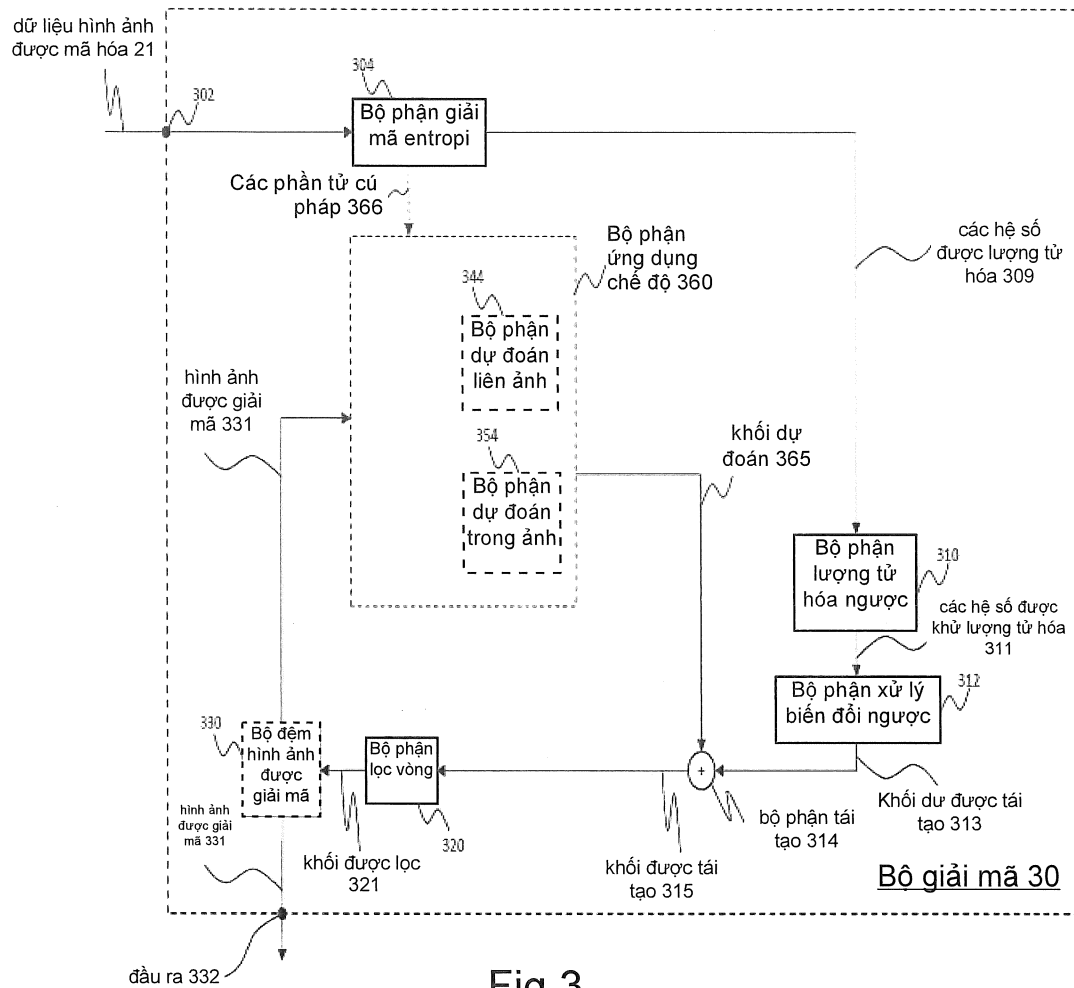
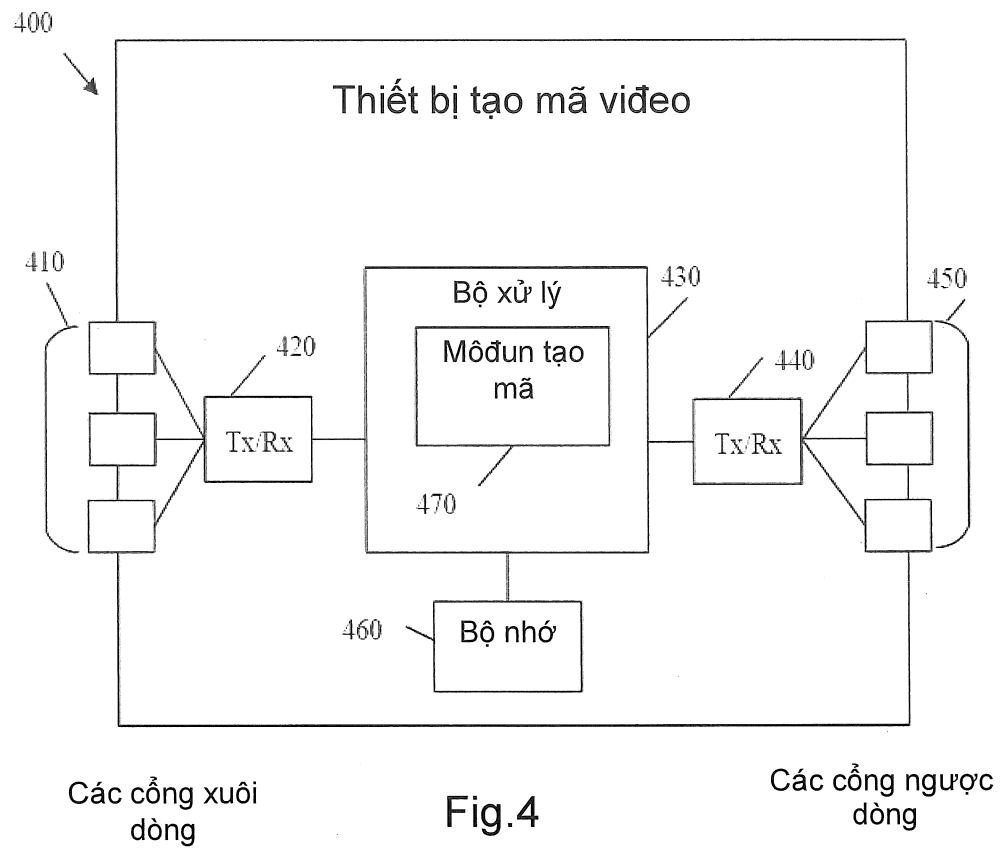


Fig.3



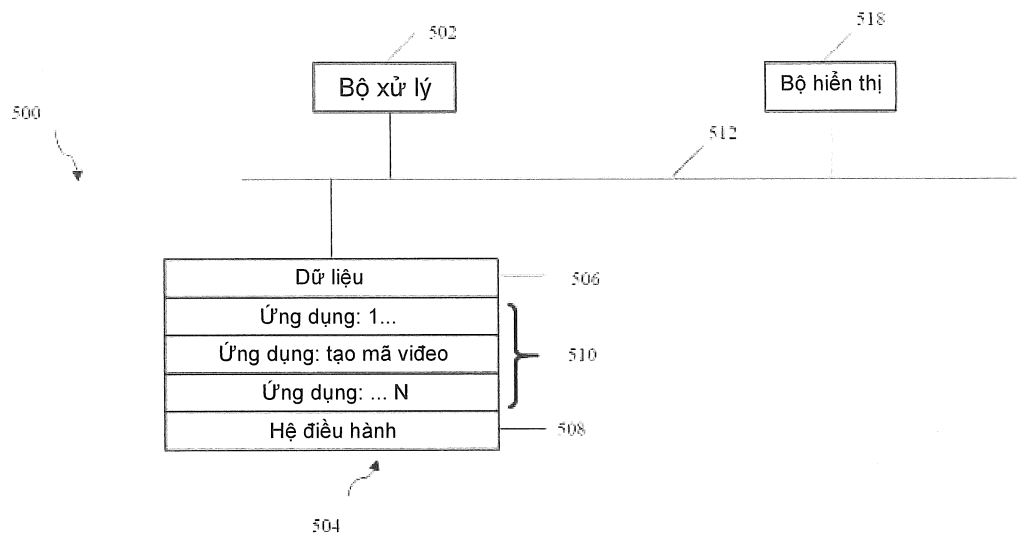


Fig.5

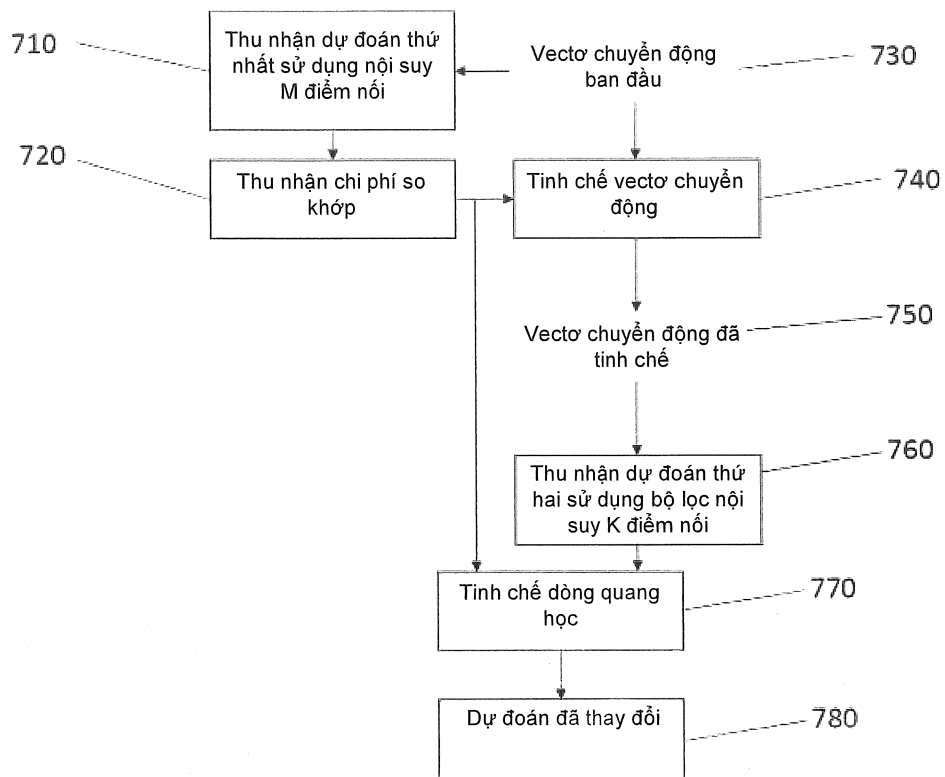


Fig.6

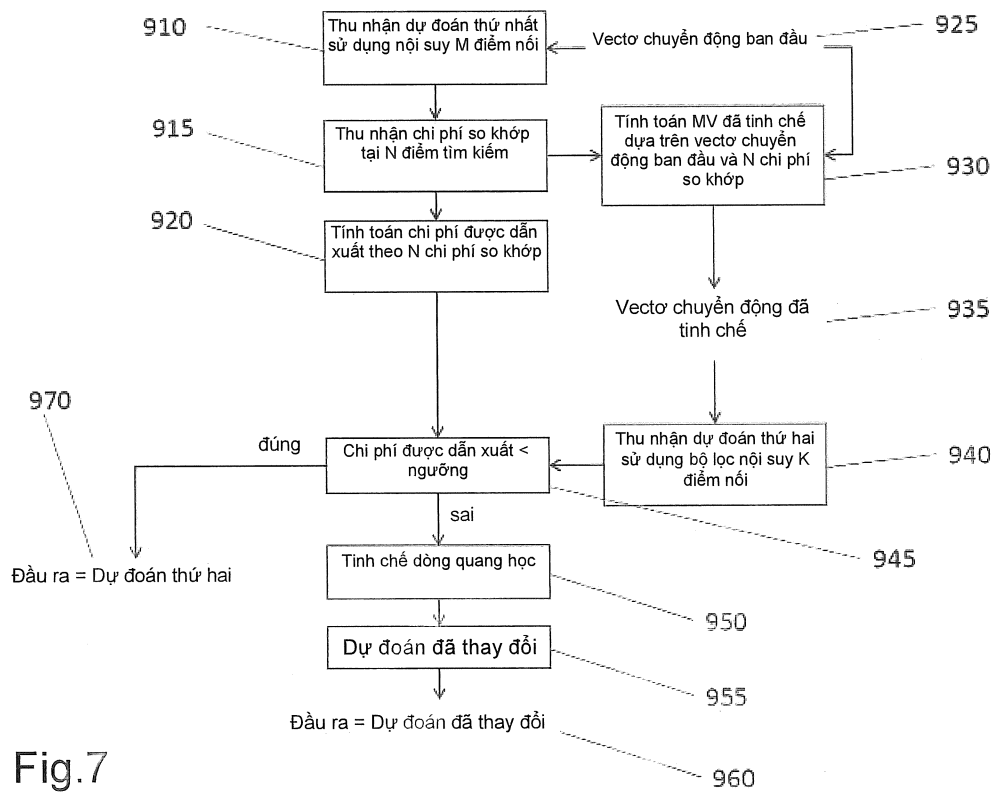


Fig.7

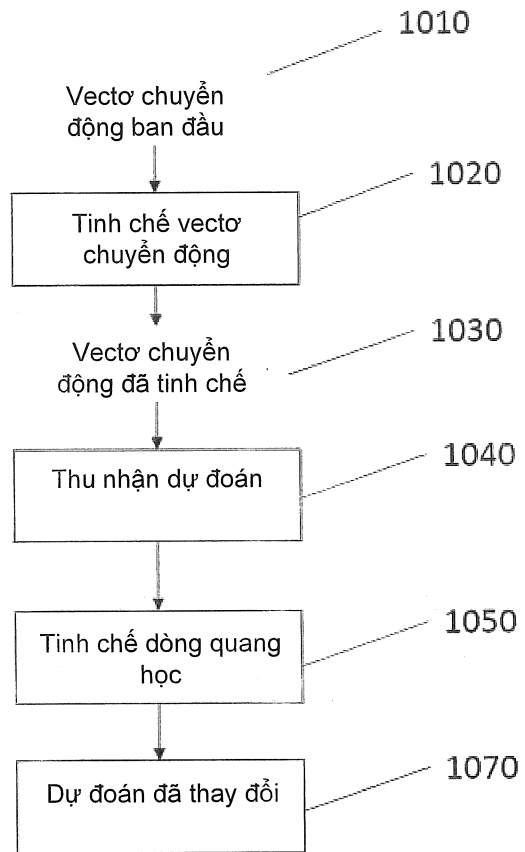
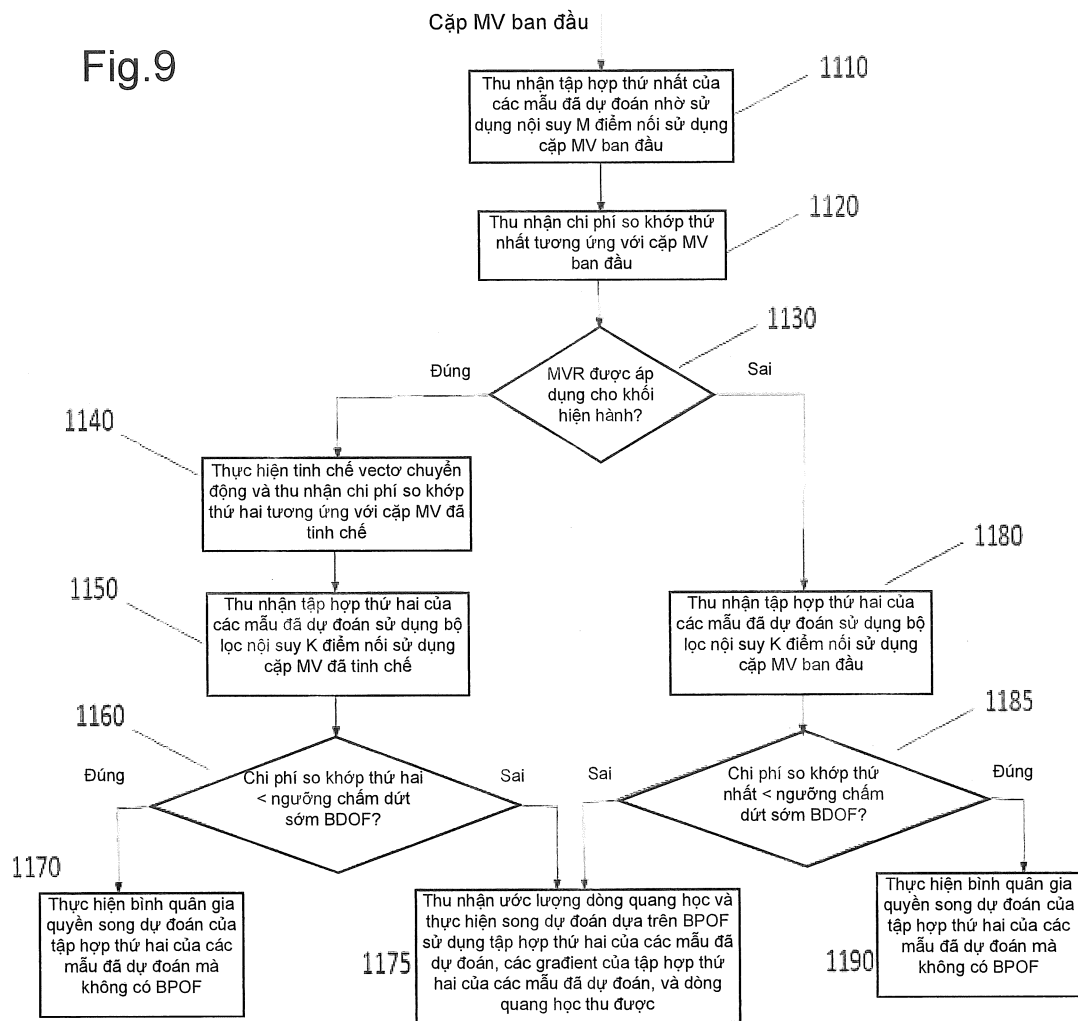


Fig.8

Fig.9



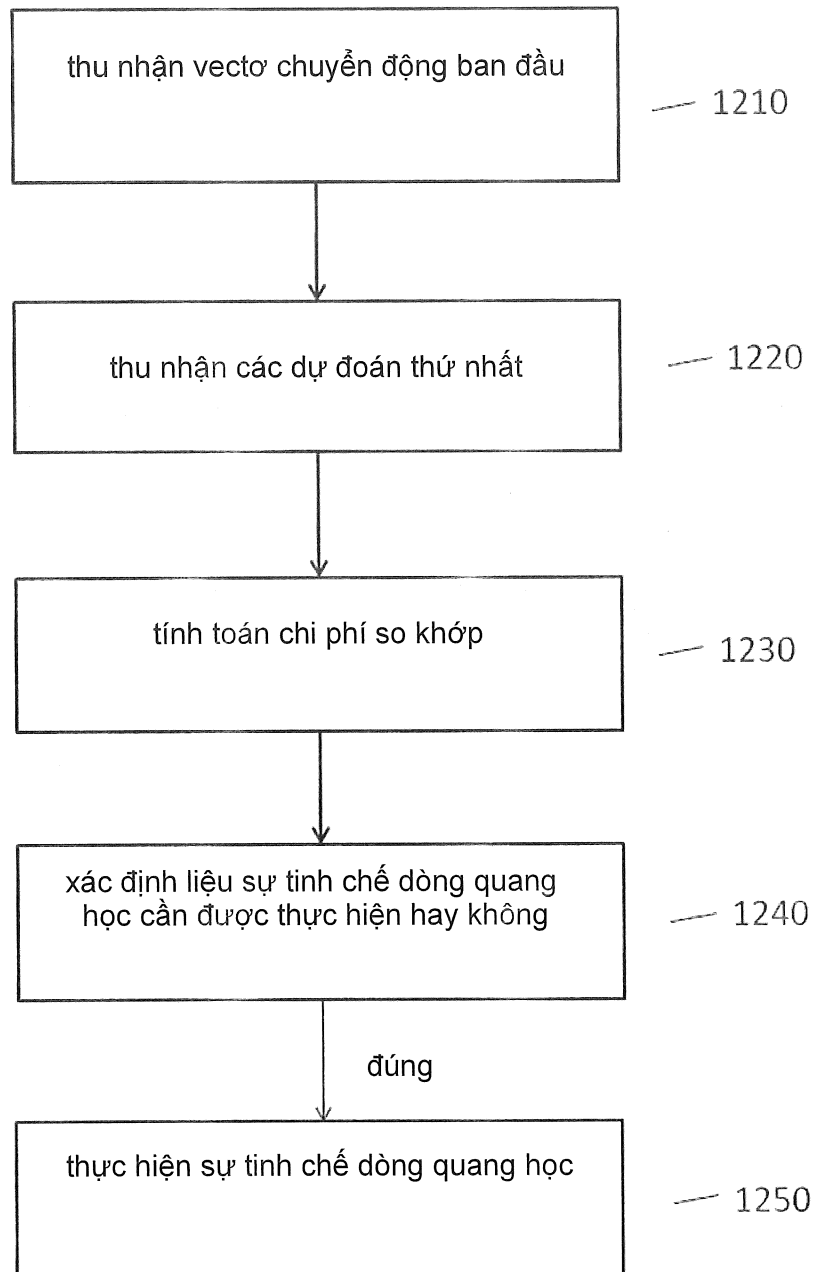


Fig.10

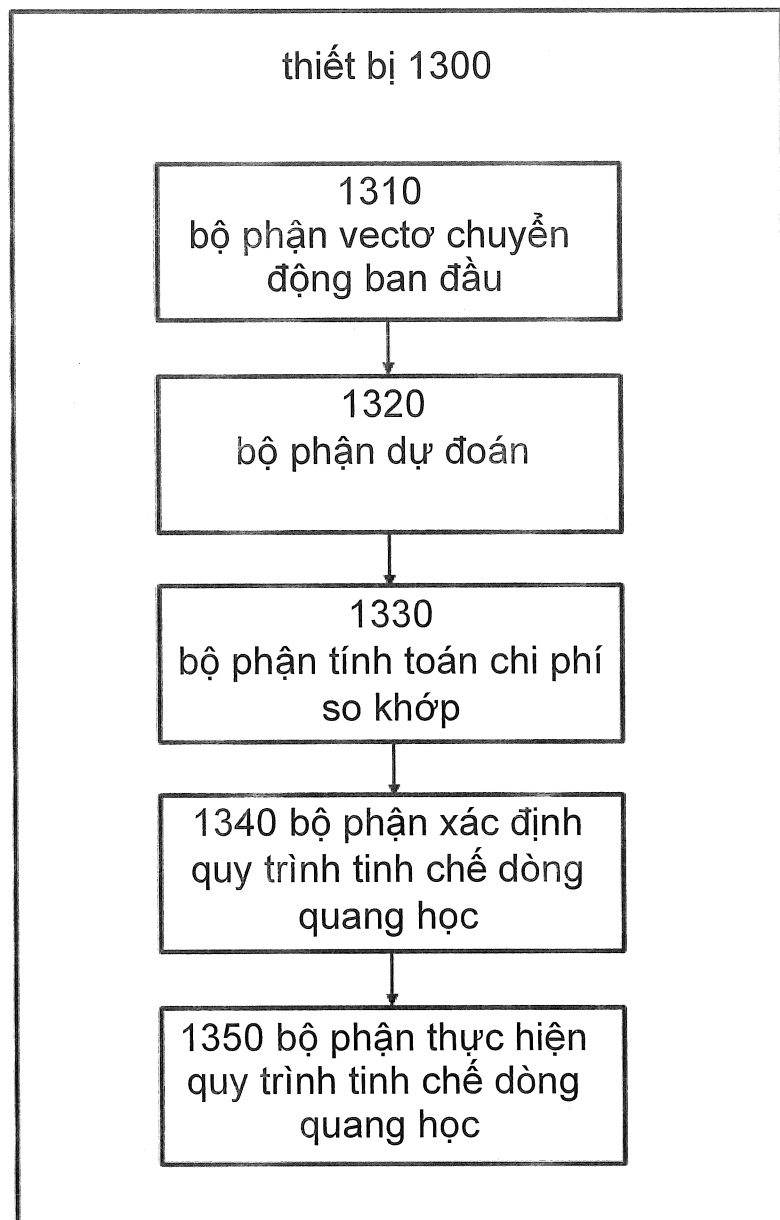


Fig.11