



(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)



1-0045014

(51)^{2020.01} H04N 19/56

(13) B

(21) 1-2021-00494

(22) 28/06/2019

(86) PCT/CN2019/093465 28/06/2019

(87) WO 2020/001574 02/01/2020

(30) 62/691,594 28/06/2018 US; 62/691,582 28/06/2018 US

(45) 25/04/2025 445

(43) 26/04/2021 397A

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

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

(72) ESENLİK, Semih (TR); KRASNOV, Ivan (RU); ZHAO, Zhijie (CN); CHEN, Jianle (CN).

(74) Công ty TNHH Tầm nhìn và Liên danh (VISION & ASSOCIATES CO.LTD.)

(54) THIẾT BỊ VÀ PHƯƠNG PHÁP MÃ HÓA XÁC ĐỊNH BỘ DỰ ĐOÁN ĐỂ MÃ HÓA KHỐI HÌNH ẢNH VIDEO VÀ THIẾT BỊ VÀ PHƯƠNG PHÁP GIẢI MÃ XÁC ĐỊNH BỘ DỰ ĐOÁN ĐỂ GIẢI MÃ KHỐI HÌNH ẢNH VIDEO

(21) 1-2021-00494

(57) Sáng chế đề cập đến việc xác định vector chuyển động bằng cách sử dụng so khớp mẫu hoặc hai chiều và tạo ra bộ dự đoán dựa vào vector chuyển động này. So khớp mẫu hoặc hai chiều và/hoặc tạo ra bộ dự đoán sử dụng quá trình lọc nội suy. Hoạt động lọc nội suy truy nhập các vị trí mẫu nguyên trong cửa sổ, và còn sử dụng các giá trị mẫu được đệm cho các vị trí mẫu nguyên bên ngoài cửa sổ, dựa vào ít nhất một mẫu trong cửa sổ này, và sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị mẫu được đệm để thực hiện so khớp mẫu hoặc hai chiều và/hoặc tạo ra bộ dự đoán.

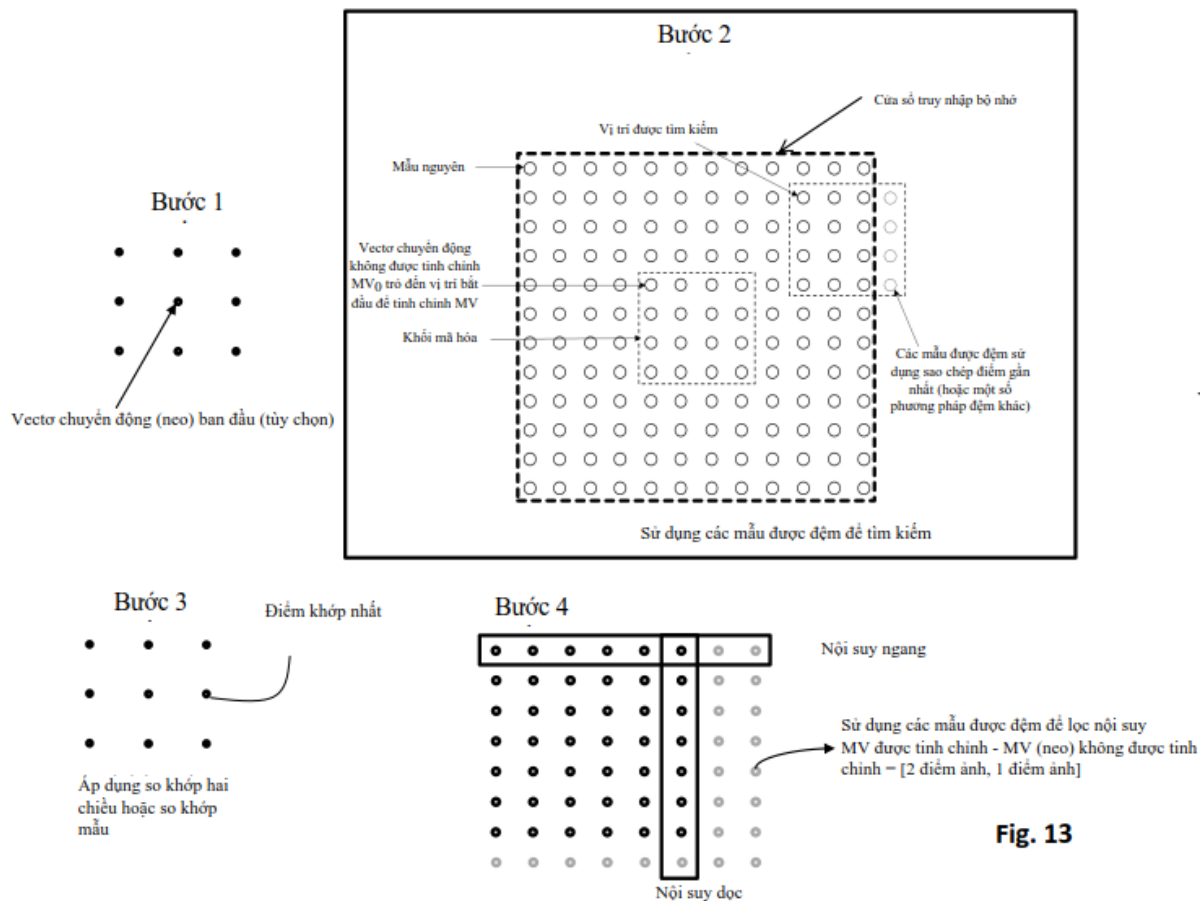


Fig. 13

FIG. 8

Tình trạng kỹ thuật được đề cập

Sáng chế đề cập đến việc xác định và tinh chỉnh vector chuyển động cũng như xác định bộ dự đoán để bù chuyển động có thể được sử dụng trong quá trình mã hóa và giải mã các video.

Tình trạng kỹ thuật của sáng chế

Các bộ mã hóa/giải mã lai hiện tại sử dụng mã hóa dự đoán. Một ảnh của chuỗi video được chia nhỏ thành các khối của các điểm ảnh và các khối này sau đó được mã hóa. Thay vì mã hóa khối theo từng điểm ảnh, toàn bộ khối được dự đoán bằng cách sử dụng các điểm ảnh đã được mã hóa gần khối này theo không gian hoặc thời gian. Bộ mã hóa chỉ xử lý thêm sự khác biệt giữa một khối và khối dự đoán của nó. Bước xử lý khác thường bao gồm biến đổi các điểm ảnh khối thành các hệ số trong miền biến đổi. Các hệ số này có thể sau đó còn được nén bằng cách lượng tử hóa và còn được nén chặt bằng cách mã hóa entropy để tạo thành dòng bit. Dòng bit này còn bao gồm thông tin báo hiệu bất kỳ cho phép bộ giải mã giải mã video được mã hóa. Ví dụ, việc báo hiệu có thể bao gồm các cài đặt liên quan đến các cài đặt bộ mã hóa chẳng hạn như kích thước của ảnh đầu ra, tốc độ khung, chỉ báo bước lượng tử hóa, dự đoán được áp dụng cho các khối của các ảnh, hoặc tương tự.

Dự đoán thời gian khai thác mối tương quan thời gian giữa các ảnh, cũng được gọi là các khung, của video. Dự đoán thời gian cũng được gọi là dự đoán liên ảnh, vì nó là dự đoán sử dụng sự phụ thuộc giữa các khung video khác nhau (liên). Theo đó, khối đang được mã hóa, cũng được gọi là khối hiện thời, được dự đoán từ một hoặc nhiều ảnh được mã hóa trước đó được gọi là (các) ảnh tham chiếu. Ảnh tham chiếu không nhất thiết phải là ảnh trước ảnh hiện thời trong đó khối hiện thời được đặt theo thứ tự hiển thị của chuỗi video. Bộ mã hóa có thể mã hóa các ảnh theo thứ tự mã hóa khác thứ tự hiển thị. Khi dự đoán khối hiện thời, khối được đặt cùng trong ảnh tham chiếu có thể được xác định. Khối được đặt cùng là khối được đặt trong ảnh tham chiếu ở cùng vị trí

với khối hiện thời trong ảnh hiện thời. Dự đoán này chính xác đối với các vùng ảnh không chuyển động, tức là các vùng ảnh không có chuyển động từ ảnh này sang ảnh khác.

Để thu được bộ dự đoán có tính đến sự di chuyển, tức là bộ dự đoán được bù chuyển động, ước tính chuyển động thường được sử dụng khi xác định dự đoán khối hiện thời. Theo đó, khối hiện thời được dự đoán bởi khối trong ảnh tham chiếu, được đặt ở khoảng cách được cho bởi vector chuyển động từ vị trí của khối được đặt cùng. Để cho phép bộ giải mã xác định cùng dự đoán khối hiện thời, vector chuyển động có thể được báo hiệu trong dòng bit. Để giảm thêm tài nguyên tốn thêm cho việc báo hiệu gây ra bởi việc báo hiệu vector chuyển động cho mỗi trong số các khối, chính vector chuyển động có thể được ước tính. Việc ước tính vector chuyển động có thể được thực hiện dựa vào các vector chuyển động của các khối lân cận trong miền không gian và/hoặc thời gian.

Dự đoán khối hiện thời có thể được tính bằng cách sử dụng một ảnh tham chiếu hoặc bằng các dự đoán có trọng số được thu từ hai hoặc nhiều hơn hai ảnh tham chiếu. Ảnh tham chiếu có thể là ảnh liền kề, tức là ảnh ngay phía trước và/hoặc ảnh ngay phía sau ảnh hiện thời theo thứ tự hiển thị do các ảnh liền kề rất có thể tương tự với ảnh hiện thời. Tuy nhiên, nói chung, ảnh tham chiếu cũng có thể là ảnh khác ảnh trước hoặc sau ảnh hiện thời theo thứ tự hiển thị và trước ảnh hiện thời trong dòng bit (thứ tự giải mã). Điều này có thể mang lại nhiều ưu điểm ví dụ trong trường hợp tắc và/hoặc di chuyển không tuyến tính trong nội dung video. Do đó, sự nhận dạng ảnh tham chiếu cũng có thể được báo hiệu trong dòng bit.

Chế độ dự đoán liên ảnh đặc biệt cũng được gọi là dự đoán hai chiều trong đó hai ảnh tham chiếu được sử dụng để tạo ra dự đoán khối hiện thời. Cụ thể, hai dự đoán được xác định trong hai ảnh tham chiếu tương ứng được kết hợp thành tín hiệu dự đoán của khối hiện thời. Dự đoán hai chiều có thể làm cho dự đoán khối hiện thời chính xác hơn dự đoán một chiều, tức là dự đoán chỉ sử dụng một ảnh tham chiếu. Dự đoán chính xác hơn dẫn đến sự khác biệt nhỏ hơn giữa các điểm ảnh của khối hiện thời và dự đoán (cũng được gọi là “các phần dư”), có thể được mã hóa hiệu quả hơn, tức là được nén thành dòng bit ngắn hơn. Nói chung, nhiều hơn hai ảnh tham chiếu có thể được sử dụng để tìm nhiều hơn hai khối tham chiếu tương ứng để dự đoán khối hiện thời, tức là dự đoán

liên ảnh nhiều tham chiếu có thể được áp dụng. Do đó, thuật ngữ dự đoán nhiều tham chiếu bao gồm dự đoán hai chiều cũng như các dự đoán sử dụng nhiều hơn hai ảnh tham chiếu.

Để tạo ra ước tính chuyển động chính xác hơn, độ phân giải của ảnh tham chiếu có thể được tăng cường bằng cách nội suy các mẫu giữa các điểm ảnh. Nội suy điểm ảnh phân số có thể được thực hiện bằng cách tính trung bình có trọng số của các điểm ảnh gần nhất. Trong trường hợp độ phân giải nửa điểm ảnh, ví dụ nội suy song tuyến tính thường được sử dụng. Các điểm ảnh phân số khác được tính là trung bình của các điểm ảnh gần nhất có trọng số bằng nghịch đảo khoảng cách giữa các điểm ảnh gần nhất tương ứng đến điểm ảnh đang được dự đoán.

Sự ước tính vector chuyển động là tác vụ phức tạp về mặt tính toán trong đó sự tương tự được tính giữa khối hiện thời và các khối dự đoán (các bộ dự đoán) tương ứng được trở đến bởi các vector chuyển động ứng viên trong ảnh tham chiếu. Thông thường, vùng tìm kiếm bao gồm $M \times M$ mẫu của ảnh và mỗi vị trí mẫu trong số $M \times M$ vị trí ứng viên được kiểm tra. Việc kiểm tra bao gồm tính toán phạm vi tương tự giữa $N \times N$ khối tham chiếu C và khối R , được đặt ở vị trí ứng viên được kiểm tra của vùng tìm kiếm. Để đơn giản, tổng các sự khác biệt tuyệt đối (Sum of Absolute Difference, SAD) là giới hạn thường được sử dụng cho mục đích này và được cho bởi:

$$SAD(x, y) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |R_{i,j}(x, y) - C_{i,j}|$$

Trong công thức nêu trên, x và y xác định vị trí ứng viên trong vùng tìm kiếm, trong khi các chỉ số i và j biểu thị các mẫu trong khối tham chiếu C và khối ứng viên R . Vị trí ứng viên thường được gọi là sự dịch chuyển hoặc độ lệch khối, phản ánh phần biểu diễn khối khớp với sự dịch chuyển của khối tham chiếu trong vùng tìm kiếm và tính sự tương tự giữa khối tham chiếu C và phần chồng lên vùng tìm kiếm. Để giảm độ phức tạp, số lượng vector chuyển động ứng viên thường được giảm đi bằng cách giới hạn các vector chuyển động ứng viên ở không gian tìm kiếm nhất định. Không gian tìm kiếm có thể, ví dụ, được xác định bởi số lượng và/hoặc vị trí của các điểm ảnh xung quanh vị trí trong ảnh tham chiếu tương ứng với vị trí của khối hiện thời trong hình ảnh hiện thời. Sau bước tính SAD cho tất cả $M \times M$ vị trí ứng viên x và y , khối khớp nhất R

là khối ở vị trí dẫn đến SAD thấp nhất, tương ứng với sự tương tự lớn nhất với khối tham chiếu C. Mặt khác, các vector chuyển động ứng viên có thể được xác định bởi danh sách các vector chuyển động ứng viên được tạo thành bởi các vector chuyển động của các khối lân cận.

Các vector chuyển động thường được xác định ít nhất một phần ở phía bộ mã hóa và được báo hiệu cho bộ giải mã trong dòng bit được mã hóa. Tuy nhiên, các vector chuyển động cũng có thể được thu ở bộ giải mã. Trong trường hợp này, khối hiện thời không có sẵn ở bộ giải mã và không thể được sử dụng để tính sự tương tự với các khối mà các vector chuyển động ứng viên trở đến trong ảnh tham chiếu. Do đó, thay vì khối hiện thời, một mẫu được sử dụng được tạo ra từ các điểm ảnh của các khối đã được giải mã (“so khớp mẫu”). Ví dụ, các điểm ảnh đã được giải mã liền kề khối hiện thời có thể được sử dụng. Sự ước tính chuyển động này mang lại ưu điểm là làm giảm báo hiệu: vector chuyển động được thu theo cùng cách ở cả bộ mã hóa và bộ giải mã và do đó, không cần báo hiệu. Mặt khác, độ chính xác của việc ước tính chuyển động như vậy có thể thấp hơn.

Trong trường hợp dự đoán hai chiều, thay vì sử dụng mẫu, cũng có thể sử dụng so sánh giữa các khối mã hóa ứng viên trong các ảnh tham chiếu khác nhau mà các vector chuyển động ứng viên trở đến (“so khớp hai chiều”), như sẽ được mô tả chi tiết dưới đây.

Để tạo ra sự cân bằng giữa độ chính xác và tài nguyên tốn thêm cho việc báo hiệu, ước tính vector chuyển động có thể được chia thành hai bước: thu vector chuyển động và tinh chỉnh vector chuyển động. Ví dụ, việc thu vector chuyển động có thể bao gồm chọn vector chuyển động từ danh sách ứng viên. Vector chuyển động được chọn này có thể còn được tinh chỉnh ví dụ bằng cách tìm kiếm trong không gian tìm kiếm. Việc tìm kiếm trong không gian tìm kiếm là dựa vào việc tính hàm chi phí cho mỗi vector chuyển động ứng viên, tức là cho mỗi vị trí ứng viên của khối mà vector chuyển động ứng viên trở đến.

Tài liệu JVET-D0029: Tinh chỉnh vector chuyển động phía bộ giải mã dựa vào so khớp mẫu hai chiều, X. Chen, J. An, J. Zheng (có thể tìm thấy tài liệu này ở: trang <http://phenix.it-sudparis.eu/jvet/>) thể hiện việc tinh chỉnh vector chuyển động trong đó

vector chuyển động thứ nhất ở độ phân giải điểm ảnh nguyên được tìm thấy và còn được tinh chỉnh bằng cách tìm kiếm ở độ phân giải nửa điểm ảnh trong không gian tìm kiếm xung quanh vector chuyển động thứ nhất.

Để thực hiện tinh chỉnh vector chuyển động, cần lưu ít nhất các mẫu này trong bộ nhớ, cần cho khối hiện thời để thực hiện việc tinh chỉnh, tức là các mẫu tương ứng với không gian tìm kiếm và các mẫu này có thể được truy nhập khi so khớp mẫu hoặc so khớp hai chiều trong không gian tìm kiếm được thực hiện.

Truy nhập bộ nhớ ngoài là tham số thiết kế quan trọng trong các kiến trúc phần cứng và/hoặc phương án thực hiện phần mềm hiện nay. Điều này là do thực tế việc truy nhập bộ nhớ ngoài làm chậm quá trình xử lý so với việc sử dụng bộ nhớ trong. Mặt khác, bộ nhớ trong trên chip bị giới hạn, ví dụ do phương án thực hiện kích thước chip.

Bản chất kỹ thuật của sáng chế

Ví dụ, sáng chế dựa vào nhận xét rằng việc tinh chỉnh vector chuyển động khi được thực hiện kết hợp với nội suy phân số có thể còn yêu cầu tăng kích thước bộ nhớ trên chip hoặc thậm chí là truy nhập bộ nhớ ngoài. Cả hai lựa chọn có thể không được mong muốn. Điều tương tự cũng được áp dụng khi nội suy được áp dụng để tăng cường độ phân giải của bộ dự đoán bằng cách bao gồm các vị trí mẫu phân số.

Theo quan điểm về vấn đề được đề cập trên đây, sáng chế đề xuất dự đoán vector chuyển động cho phép tính đến số lượng truy nhập vào bộ nhớ ngoài và số lượng mẫu cần để truy nhập được để tinh chỉnh vector chuyển động của vector chuyển động cho khối mã hóa và tính bộ dự đoán cho khối.

Đạt được điều này bằng cách đệm (thay thế) các giá trị được đặt ngoài cửa sổ truy nhập bộ nhớ được xác định trước cho mục đích tinh chỉnh vector chuyển động và/hoặc nội suy phân số bằng các giá trị thay thế dựa vào một hoặc nhiều mẫu từ trong cửa sổ truy nhập bộ nhớ này.

Đạt được điều này bằng các dấu hiệu của các điểm độc lập.

Theo một khía cạnh của sáng chế, thiết bị mã hóa xác định bộ dự đoán để mã hóa khối hình ảnh video được đề xuất. Thiết bị mã hóa này bao gồm hệ mạch xử lý được tạo

cấu hình để thu được vector chuyển động ban đầu cho khối, thu được vector chuyển động được tinh chỉnh cho khối này dựa vào vector chuyển động ban đầu, và xác định bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh. Mạch xử lý được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy và để sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

Theo một khía cạnh khác của sáng chế, thiết bị giải mã xác định bộ dự đoán để giải mã khối hình ảnh video được đề xuất. Thiết bị giải mã này bao gồm hệ mạch xử lý được tạo cấu hình để thu được vector chuyển động ban đầu cho khối, thu được vector chuyển động được tinh chỉnh cho khối dự đoán dựa vào vector chuyển động ban đầu, và xác định bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh. Mạch xử lý được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy và để sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

Theo một khía cạnh khác của sáng chế, phương pháp mã hóa xác định bộ dự đoán để mã hóa khối hình ảnh video được đề xuất. Phương pháp mã hóa này bao gồm các bước thu được vector chuyển động ban đầu cho khối, thu được vector chuyển động được tinh chỉnh cho khối này dựa vào vector chuyển động ban đầu, và xác định bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh, trong đó bước xác định xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, và sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm dựa vào ít nhất một giá trị mẫu của mẫu nguyên được đặt bên trong cửa sổ.

Theo một khía cạnh khác nữa của sáng chế, phương pháp giải mã xác định bộ dự đoán để giải mã khối hình ảnh video được đề xuất. Phương pháp giải mã này bao gồm các bước thu được vector chuyển động ban đầu cho khối, thu được vector chuyển động

được tinh chỉnh cho khối dự đoán dựa vào vector chuyển động ban đầu, và xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm này dựa vào ít nhất một giá trị mẫu của mẫu nguyên được đặt bên trong cửa sổ.

Các phương pháp và các thiết bị này mang lại ưu điểm về giới hạn số lượng mẫu cần có sẵn cho mục đích xác định bộ dự đoán bằng cách lọc nội suy trong khi còn tránh được các truy nhập bổ sung vào bộ nhớ lưu trữ /(ngoài) lưu trữ toàn bộ các ảnh tham chiếu.

Theo các phương án của các khía cạnh nêu trên, cửa sổ là cửa sổ để lọc nội suy.

Theo các phương án, việc nội suy có thể bao gồm truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ và/hoặc đệm ít nhất một giá trị mẫu cho vị trí mẫu nguyên được đặt ngoài cửa sổ này bằng giá trị đệm dựa vào ít nhất một mẫu trong cửa sổ và/hoặc sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị mẫu được đệm để thực hiện lọc nội suy.

Ví dụ, hệ mạch xử lý có thể được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu dựa vào không gian tìm kiếm của các vector chuyển động ứng viên và để xác định vector chuyển động được tinh chỉnh dựa vào vector chuyển động ban đầu và việc tinh chỉnh.

Ngoài ra, ví dụ, hệ mạch xử lý có thể được tạo cấu hình để xác định không gian tìm kiếm dựa vào vector chuyển động ban đầu. Cụ thể hơn là, không gian tìm kiếm có thể được đặt ở vị trí được xác định bởi vector chuyển động ban đầu. Ví dụ, không gian tìm kiếm có thể được tập trung xung quanh vector chuyển động ban đầu. Cụ thể hơn là, theo một ví dụ, không gian tìm kiếm có thể bao gồm chín ứng viên vector chuyển động mẫu nguyên, tức là một vector chuyển động ban đầu và tám ứng viên vector chuyển động được đặt xung quanh.

Theo các phương án, hệ mạch xử lý còn được tạo cấu hình để tái tạo khối dựa vào bộ dự đoán. Việc tái tạo có thể được thực hiện ở thiết bị giải mã cũng như ở vòng lặp

giải mã của thiết bị mã hóa. Các khía cạnh của sáng chế đề cập đến phương pháp có thể bao gồm bước tái tạo khối tương ứng dựa vào bộ dự đoán.

Ví dụ, hệ mạch xử lý có thể được tạo cấu hình để thu được sự khác biệt giữa khối và bộ dự đoán, cũng được gọi là “phần dư”. Cụ thể hơn là, hệ mạch xử lý có thể được tạo cấu hình để tái tạo khối dựa vào bộ dự đoán và phần dư, cụ thể hơn nữa là tổng của bộ dự đoán và phần dư.

Ví dụ, giá trị mẫu được đệm (giá trị thay thế) được xác định bằng cách phản chiếu đối với trục của bộ lọc nội suy giá trị của vị trí mẫu nguyên (được truy nhập) được sử dụng tương ứng. Quá trình lọc nội suy có thể là lọc một chiều hoặc hai chiều tách được.

Theo một phương án thực hiện làm ví dụ, cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để thu được bộ dự đoán ở vị trí được chỉ định bởi vector chuyển động ban đầu.

Cửa sổ được xác định, ví dụ, bởi kích thước dọc và/hoặc ngang của nó đối với khối có kích thước của khối được đặt ở vị trí vector chuyển động ban đầu, hoặc vị trí vector chuyển động ban đầu, hoặc chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán. Việc xác định khối đang “được đặt ở” vị trí cụ thể, theo các phương án, ví dụ, là đề cập đến vị trí mẫu ở trên bên trái của khối. Theo cách khác, vị trí mẫu khác có thể được gọi là, hoặc vị trí khối có thể được xác định bằng cách được tập trung xung quanh vị trí cụ thể này. Mạch xử lý có thể được tạo cấu hình để định nghĩa hoặc xác định vị trí và/hoặc kích thước của cửa sổ theo vị trí vector chuyển động ban đầu, kích thước của khối và /hoặc chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.

Ví dụ khác, giá trị mẫu được đệm cho vị trí nguyên bên ngoài cửa sổ được xác định bằng giá trị mẫu của ít nhất một vị trí mẫu nguyên được sử dụng được đặt bên trong cửa sổ, gần nhất với vị trí mẫu nguyên được đặt bên ngoài cửa sổ. Trong một ví dụ cụ thể, trong trường hợp bộ lọc nội suy một chiều, các giá trị mẫu được đệm cho các vị trí mẫu nguyên được đặt bên ngoài cửa sổ được sử dụng để xác định bộ dự đoán sử dụng nội suy bằng bộ lọc nội suy và các giá trị mẫu được đệm cho tất cả các vị trí mẫu nguyên bên ngoài cửa sổ được xác định bằng giá trị mẫu của ít nhất một vị trí mẫu nguyên được sử dụng được đặt bên trong cửa sổ, gần nhất với các vị trí mẫu nguyên được đặt bên

ngoài cửa sổ.

Theo các phương án, số lượng giá trị mẫu cần được đệm phụ thuộc vào sự khác biệt giữa vector chuyển động ban đầu và vector chuyển động được tinh chỉnh.

Theo các phương án, hệ mạch xử lý được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp một mẫu bằng so khớp mẫu hoặc hai chiều. So khớp mẫu hoặc hai chiều truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ khác. Cửa sổ khác này bao gồm không gian tìm kiếm và các vị trí mẫu nguyên khác truy nhập được để so khớp mẫu hoặc hai chiều, so khớp mẫu hoặc hai chiều còn đệm ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ khác bằng giá trị (thay thế) mẫu được đệm dựa vào ít nhất một mẫu trong cửa sổ khác và sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị mẫu được đệm để thực hiện so khớp mẫu hoặc hai chiều.

Theo một phương án, không gian tìm kiếm này bao gồm ít nhất một vị trí mẫu phân số có giá trị được thu bằng cách lọc nội suy bằng bộ lọc có kích thước tap được xác định trước; và lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được truy nhập này và ít nhất một giá trị (mẫu được đệm) thay thế để thu được ít nhất một giá trị vị trí mẫu phân số này.

Cụ thể hơn là, giá trị mẫu được đệm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng cách phản chiếu đối với trục của bộ lọc giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

Theo cách khác, giá trị mẫu được đệm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

Theo các phương án, cửa sổ khác được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số không gian tìm kiếm, khối có kích thước của khối được đặt ở vị trí vector chuyển động ban đầu, và vị trí vector chuyển động ban đầu.

Tốt nhất là, cửa sổ khác giống cửa sổ này.

Theo các phương án, việc tinh chỉnh được xác định bằng cách so khớp hai chiều

và hệ mạch xử lý được tạo cấu hình để thu được ít nhất hai vectơ chuyển động ban đầu trở đến các vị trí trong các ảnh tham chiếu khác nhau và tạo thành cặp vectơ chuyển động ban đầu.

Theo các phương án khác, việc tinh chỉnh được xác định bằng cách so khớp mẫu và hệ mạch xử lý còn được tạo cấu hình để thu được mẫu cho khối và xác định tinh chỉnh vectơ chuyển động ban đầu bằng cách so khớp mẫu với mẫu trong không gian tìm kiếm.

Theo một phương án, không gian tìm kiếm này bao gồm ít nhất một vị trí mẫu phân số có giá trị được thu bằng cách lọc nội suy bằng bộ lọc có kích thước tap được xác định trước; và lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được sử dụng và ít nhất một giá trị mẫu được đệm để thu được ít nhất một giá trị vị trí mẫu phân số.

Theo một khía cạnh của sáng chế vật ghi lưu trữ lâu dài đọc được bằng máy tính được đề xuất lưu trữ các lệnh mà khi được thực thi bởi bộ xử lý/hệ mạch xử lý thực hiện các bước theo một bất kỳ trong số các khía cạnh hoặc các phương án nêu trên hoặc các sự kết hợp của chúng.

Theo một khía cạnh cụ thể khác của sáng chế, thiết bị mã hóa được đề xuất để mã hóa các hình ảnh video được chia thành các khối dự đoán thành dòng bit hoặc tương ứng để mã hóa khối dự đoán của hình ảnh video thành dòng bit. Thiết bị mã hóa này bao gồm hệ mạch xử lý để xác định vectơ chuyển động cho khối dự đoán. Mạch xử lý này được tạo cấu hình để thu được ít nhất một vectơ chuyển động ban đầu và xác định tinh chỉnh vectơ chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vectơ chuyển động ứng viên, để thu được vectơ chuyển động được tinh chỉnh. Không gian tìm kiếm được đặt ở vị trí được cho bởi vectơ chuyển động ban đầu. Thiết bị mã hóa này còn bao gồm hệ mạch mã hóa (có thể nằm trong hệ mạch xử lý) để mã hóa sự khác biệt giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào vectơ chuyển động được xác định và để tạo ra dòng bit bao gồm sự khác biệt được mã hóa và vectơ chuyển động ban đầu. Thiết bị mã hóa này còn bao gồm bộ phận dự đoán chuyển động (có thể nằm trong hệ mạch xử lý) để xác định bộ dự đoán theo vectơ chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy. Lọc nội suy truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước, thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên

được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ và sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Theo một khía cạnh khác nữa của sáng chế, thiết bị giải mã được đề xuất để giải mã từ dòng bit các hình ảnh video được chia thành các khối dự đoán. Thiết bị giải mã này bao gồm bộ phận phân tích cú pháp (có thể nằm trong hệ mạch xử lý của thiết bị giải mã) để phân tích cú pháp từ dòng bit vector chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh. Thiết bị giải mã này bao gồm hệ mạch xử lý (ví dụ hệ mạch xử lý của thiết bị giải mã) để xác định vector chuyển động cho khối dự đoán. Hệ mạch xử lý được tạo cấu hình để thu được ít nhất một vector chuyển động ban đầu và xác định tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh. Không gian tìm kiếm được đặt ở vị trí được cho bởi vector chuyển động ban đầu. Hơn nữa, thiết bị giải mã này bao gồm hệ mạch giải mã (có thể nằm trong hệ mạch xử lý của thiết bị giải mã) để tái tạo khối dự đoán ở dạng tổng khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh được xác định bởi hệ mạch xử lý để xác định vector chuyển động. Hệ mạch giải mã này còn được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy. Lọc nội suy truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước, thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ và sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Theo một khía cạnh khác nữa của sáng chế, phương pháp mã hóa được đề xuất để mã hóa các hình ảnh video được chia thành các khối dự đoán thành dòng bit. Phương pháp mã hóa này bao gồm bước xác định vector chuyển động cho khối dự đoán. Bước xác định vector chuyển động bao gồm các bước thu được ít nhất một vector chuyển động ban đầu và xác định tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh. Không gian tìm kiếm được đặt ở vị trí được cho bởi vector chuyển động ban đầu.

Phương pháp mã hóa này còn bao gồm bước mã hóa sự khác biệt giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào vector chuyển động được xác định và để tạo ra dòng bit bao gồm sự khác biệt được mã hóa và vector chuyển động ban đầu. Phương pháp này còn bao gồm bước xác định bộ dự đoán theo vector chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy. Lọc nội suy truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước, thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ và sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Theo một khía cạnh bổ sung nữa của sáng chế, phương pháp giải mã để giải mã từ dòng bit các hình ảnh video được chia thành các khối dự đoán được đề xuất. Bước giải mã bao gồm các bước phân tích cú pháp từ dòng bit vector chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh. Phương pháp này còn bao gồm bước xác định vector chuyển động cho khối dự đoán. Bước xác định vector chuyển động bao gồm các bước thu được ít nhất một vector chuyển động ban đầu và xác định tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh. Không gian tìm kiếm được đặt ở vị trí được cho bởi vector chuyển động ban đầu. Phương pháp giải mã này còn bao gồm bước tái tạo khối dự đoán ở dạng tổng khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh được xác định bởi bước xác định vector chuyển động. Hơn nữa, phương pháp này còn bao gồm bước xác định bộ dự đoán theo vector chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy. Lọc nội suy truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước, thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ và sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Các ưu điểm và các phương án khác của sáng chế được nêu trong các điểm phụ thuộc.

Mô tả vắn tắt các hình vẽ

Trong phần tiếp theo các phương án làm ví dụ được mô tả chi tiết với tham chiếu đến các ảnh và các hình vẽ kèm theo, trong đó:

FIG.1 là sơ đồ khối thể hiện cấu trúc làm ví dụ của bộ mã hóa trong đó việc thu và tinh chỉnh vector chuyển động có thể được sử dụng;

FIG.2 là sơ đồ khối thể hiện cấu trúc làm ví dụ của bộ giải mã trong đó việc thu và tinh chỉnh vector chuyển động có thể được sử dụng;

FIG.3 là hình vẽ giản lược minh họa so khớp mẫu làm ví dụ thích hợp để dự đoán hai chiều;

FIG.4 là hình vẽ giản lược minh họa so khớp mẫu làm ví dụ thích hợp để dự đoán một chiều và hai chiều;

FIG.5 là sơ đồ khối minh họa các giai đoạn của hoạt động thu vector chuyển động mà không tạo ra các vector chuyển động ban đầu cần được tinh chỉnh trong dòng bit;

FIG.6 là hình vẽ giản lược minh họa khái niệm so khớp hai chiều;

FIG.7 là sơ đồ khối minh họa phần cứng làm ví dụ để thực hiện một phương án của sáng chế;

FIG.8 là hình vẽ giản lược minh họa cho khối mã hóa của cửa sổ làm ví dụ có các mẫu cần có sẵn để được truy nhập;

FIG.9 là hình vẽ giản lược minh họa không gian tìm kiếm lặp lại;

FIG.10 là hình vẽ giản lược minh họa sự mở rộng cửa sổ truy nhập bộ nhớ theo chiều dọc do lọc nội suy;

FIG.11 là hình vẽ minh họa giản đồ thực hiện tinh chỉnh vector chuyển động thông thường dựa vào không gian tìm kiếm và thu được dự đoán bằng nội suy;

FIG.12 là lưu đồ minh họa việc thu vector chuyển động và thu được dự đoán theo ví dụ thông thường trên FIG.11;

FIG.13 là hình vẽ minh họa giản đồ thực hiện tinh chỉnh vector chuyển động dựa

vào không gian tìm kiếm và thu được dự đoán bằng nội suy theo các phương án làm ví dụ của sáng chế;

FIG.14 là hình vẽ giản lược minh họa việc đếm hai mẫu để cho phép tính toán vị trí phân số;

FIG.15 là hình vẽ giản lược minh họa việc đếm bốn mẫu để cho phép so khớp mẫu ở các biên không gian tìm kiếm;

FIG.16 là hình vẽ giản lược minh họa các vị trí phân số làm ví dụ mà nội suy không yêu cầu đếm;

FIG.17 là sơ đồ luồng minh họa việc tinh chỉnh vector chuyển động và thu được dự đoán theo các phương án làm ví dụ trên FIG.13;

FIG.18 là hình vẽ giản lược minh họa sự xác định cửa sổ truy nhập bộ nhớ đối với bộ phận dự đoán;

FIG.19 mô tả phương pháp theo một phương án;

FIG.20 là hình vẽ minh họa giản đồ thực hiện tinh chỉnh vector chuyển động dựa vào không gian tìm kiếm và thu được dự đoán bằng nội suy theo các phương án làm ví dụ khác của sáng chế; và

FIG.21 là sơ đồ luồng minh họa việc tinh chỉnh vector chuyển động và thu được dự đoán theo các phương án làm ví dụ trên FIG.20.

Mô tả chi tiết sáng chế

Sáng chế đề cập đến việc điều chỉnh số lượng mẫu cần truy nhập được để thực hiện tinh chỉnh vector chuyển động và nội suy để thu được các vị trí phân số trong ảnh tham chiếu cũng như trong bộ dự đoán.

Như được đề cập trên đây, truy nhập bộ nhớ ngoài là một trong các cân nhắc thiết kế quan trọng nhất trong các kiến trúc phần cứng và phần mềm ngày nay. Ước tính vector chuyển động, đặc biệt khi bao gồm so khớp mẫu hoặc so khớp hai chiều, ví dụ trong trường hợp tinh chỉnh vector chuyển động, cũng có thể được sử dụng bằng lọc nội suy

để thu được các vị trí phân số của không gian tìm kiếm. Việc sử dụng lọc nội suy có thể yêu cầu tăng số lượng mẫu cần được truy nhập từ bộ nhớ. Tuy nhiên, điều này có thể dẫn đến tăng bộ nhớ trên chip đắt tiền hoặc tăng số lượng truy nhập bộ nhớ ngoài, mặt khác làm chậm quá trình khai triển. Đặc biệt là ở phía bộ giải mã, các vấn đề này có thể dẫn đến các ứng dụng đắt tiền hoặc chậm hơn mà không được mong muốn.

Để ngăn các trường hợp này, sáng chế đề xuất hạn chế về quyền truy nhập bộ nhớ ngoài vào các cửa sổ định trước.

Các cửa sổ này có thể có kích thước được xác định trước. Ví dụ, trong quá trình tinh chỉnh vector chuyển động, cửa sổ có thể bao gồm ít nhất tất cả các vị trí không gian tìm kiếm. Các mẫu được truy nhập là các mẫu mà được sử dụng thực tế cho khối cụ thể và không gian tìm kiếm của nó để thực hiện so khớp mẫu hoặc hai chiều, ví dụ, tất cả các mẫu được sử dụng để tính hàm chi phí bằng mẫu hoặc hàm chi phí hai chiều. Các mẫu truy nhập được bao gồm các mẫu bổ sung không cần được truy nhập trong quá trình so khớp mẫu hoặc hai chiều cụ thể cho khối cụ thể ví dụ vì không gian tìm kiếm đã được xác định theo cách lặp lại để bao gồm các mẫu khác nhau và so khớp mẫu hoặc hai chiều đối với các mẫu này không yêu cầu tất cả các điểm ảnh truy nhập được. Các mẫu truy nhập được cũng bao gồm các mẫu có thể là kết quả từ không gian tìm kiếm được xác định lặp đi lặp lại (tức là giả sử nội dung khối bất kỳ) và cũng có thể là các mẫu cần được truy nhập để thực hiện so khớp mẫu trong các không gian tìm kiếm khả dụng này.

Trong trường hợp tạo ra dự đoán cuối cùng bằng lọc nội suy lượng mở rộng cửa sổ truy nhập bộ nhớ được yêu cầu theo cách tiếp cận thông thường phụ thuộc vào sự khác biệt lớn nhất giữa vector chuyển động được tinh chỉnh được xác định và vector chuyển động ban đầu.

Cần lưu ý rằng cửa sổ có thể được xác định nhỏ hơn tất cả các mẫu truy nhập được. Sự xác định cửa sổ này cho phép duy trì số lượng mẫu thấp hơn được lưu trữ trong bộ nhớ trên chip và được tìm nạp từ bộ nhớ ngoài. Do đó, sáng chế giới hạn số lượng mẫu nguyên được lưu trữ/được lưu đệm/được đệm cho mục đích hoạt động liên quan đến so khớp mẫu hoặc hai chiều cho khối cụ thể. Điều này không có nghĩa là các mẫu khác thường không tồn tại/có sẵn trong bộ nhớ hoặc bộ lưu trữ khác. Như được thảo luận trên đây, thông thường toàn bộ các ảnh tham chiếu có thể được lưu ở bộ nhớ ngoài.

Tuy nhiên, đối với mục đích so khớp mẫu hoặc hai chiều, chỉ một phần trong số chúng, cụ thể các mẫu trong cửa sổ có thể được tải và được sử dụng để so khớp mẫu.

Việc tạo ra cửa sổ giới hạn như vậy cho phép các phương án thực hiện khai thác cửa sổ giới hạn này và thực tế chỉ tải các mẫu cửa sổ. Để đảm bảo hoạt động tương tự của bộ mã hóa và bộ giải mã, việc xác định cửa sổ có thể được xác định trước theo chuẩn hoặc được báo hiệu trong dòng bit. Đặc biệt là nếu các vector chuyển động được tinh chỉnh được sử dụng để tạo thành các ảnh tham chiếu, thì cả bộ mã hóa và bộ giải mã cần sử dụng cùng cách tiếp cận.

FIG.1 thể hiện bộ mã hóa 100 bao gồm đầu vào để nhận các mẫu ảnh đầu vào của các khung hoặc các ảnh của dòng video và đầu ra để tạo ra dòng bit video được mã hóa. Thuật ngữ “khung” trong bản mô tả này được sử dụng là từ đồng nghĩa với ảnh. Tuy nhiên, cần lưu ý rằng sáng chế cũng áp dụng được với các lĩnh vực trong trường hợp kỹ thuật xen kẽ được áp dụng. Nói chung, một ảnh bao gồm m nhân n điểm ảnh. Điều này tương ứng với các mẫu hình ảnh và có thể bao gồm một hoặc nhiều thành phần màu. Để đơn giản, phần mô tả tiếp theo đề cập đến các điểm ảnh nghĩa là các mẫu độ sáng. Tuy nhiên, cần lưu ý rằng việc tìm kiếm vector chuyển động theo sáng chế có thể được áp dụng cho thành phần màu bất kỳ bao gồm thành phần sắc độ hoặc các thành phần không gian tìm kiếm chẳng hạn như RGB hoặc tương tự. Mặt khác, có thể có lợi nếu chỉ thực hiện ước tính vector chuyển động đối với một thành phần và áp dụng vector chuyển động được xác định cho nhiều (hoặc tất cả) thành phần.

Các khối đầu vào cần được mã hóa không nhất thiết có cùng kích thước. Một ảnh có thể bao gồm các khối có các kích thước khác nhau và bộ quét khối của các ảnh khác nhau cũng có thể khác nhau.

Trong thực tế giải thích, bộ mã hóa 100 được tạo cấu hình để áp dụng dự đoán, biến đổi, lượng tử hóa, và mã hóa entropy cho dòng video. Biến đổi, lượng tử hóa, và mã hóa entropy được thực hiện tương ứng bởi bộ phận biến đổi 106, bộ phận lượng tử hóa 108 và bộ phận mã hóa entropy 170 để tạo ra ở dạng đầu ra dòng bit video được mã hóa.

Dòng video có thể bao gồm các khung, trong đó mỗi khung được chia thành các khối có kích thước nhất định được mã hóa nội ảnh hoặc liên ảnh. Các khối ví dụ của

khung thứ nhất của dòng video được mã hóa nội ảnh bằng bộ phận dự đoán nội ảnh 154. Khung nội ảnh được mã hóa bằng cách chỉ sử dụng thông tin trong cùng khung, để có thể được giải mã một cách độc lập và có thể tạo ra điểm nhập trong dòng bit để truy nhập ngẫu nhiên. Các khối của các khung khác của dòng video có thể được mã hóa liên ảnh bằng bộ phận dự đoán liên ảnh 144: thông tin từ các khung được mã hóa trước đó (các khung tham chiếu) được sử dụng để giảm phần dư theo thời gian, để mỗi khối của khung được mã hóa liên ảnh được dự đoán từ khối trong khung tham chiếu. Bộ phận chọn chế độ 160 được tạo cấu hình để chọn xem liệu khối của một khung cần được xử lý bởi bộ phận dự đoán nội ảnh 154 hay bộ phận dự đoán liên ảnh 144. Bộ phận chọn chế độ 160 cũng điều khiển các tham số của dự đoán nội ảnh hoặc liên ảnh. Để cho phép tinh chỉnh thông tin hình ảnh, các khối được mã hóa nội ảnh có thể được tạo ra trong các khung được mã hóa liên ảnh. Hơn nữa, các khung nội ảnh mà chỉ chứa các khối được mã hóa nội ảnh có thể được chèn đều đặn vào chuỗi video để tạo ra các điểm nhập để giải mã, tức là các điểm trong đó bộ giải mã có thể bắt đầu giải mã mà không có thông tin từ các khung được mã hóa trước đó.

Bộ phận ước tính nội ảnh 152 và bộ phận dự đoán nội ảnh 154 là các bộ phận thực hiện dự đoán nội ảnh. Cụ thể là, bộ phận ước tính nội ảnh 152 có thể thu chế độ dự đoán cũng dựa vào hiểu biết về hình ảnh gốc trong khi bộ phận dự đoán nội ảnh 154 tạo ra bộ dự đoán tương ứng, tức là các mẫu được dự đoán sử dụng chế độ dự đoán được chọn, để mã hóa sự khác biệt. Để thực hiện dự đoán không gian hoặc thời gian, các khối được mã hóa có thể còn được xử lý bởi bộ phận lượng tử hóa ngược 110, và bộ phận biến đổi ngược 112. Sau khi tái tạo khối bộ phận lọc vòng 120 được áp dụng để cải thiện hơn chất lượng hình ảnh được giải mã. Các khối được lọc sau đó tạo thành các khung tham chiếu mà sau đó được lưu trữ trong bộ đệm ảnh được giải mã 130. Vòng lặp giải mã (bộ giải mã) như vậy ở phía bộ mã hóa mang lại ưu điểm là tạo ra các khung tham chiếu giống các ảnh tham chiếu được tái tạo ở phía bộ giải mã. Theo đó, phía bộ mã hóa và bộ giải mã hoạt động theo cách tương ứng. Thuật ngữ “tái tạo” trong bản mô tả này đề cập đến việc thu được khối được tái tạo bằng cách thêm vào khối dư được giải mã khối dự đoán.

Bộ phận ước tính liên ảnh 142 nhận ở dạng đầu vào khối của khung hoặc ảnh hiện thời cần được mã hóa liên ảnh và một hoặc một số khung tham chiếu từ bộ đệm ảnh

được giải mã 130. Ước tính chuyển động được thực hiện bởi bộ phận ước tính liên ảnh 142 trong khi bù chuyển động được áp dụng bởi bộ phận dự đoán liên ảnh 144. Ước tính chuyển động được sử dụng để thu được vector chuyển động và khung tham chiếu dựa vào hàm chi phí nhất định, ví dụ cũng sử dụng hình ảnh gốc cần được mã hóa. Ví dụ, bộ phận ước tính chuyển động 142 có thể tạo ra ước tính vector chuyển động ban đầu. Vector chuyển động ban đầu có thể sau đó được báo hiệu trong dòng bit ở dạng vector một cách trực tiếp hoặc ở dạng chỉ số tham chiếu đến ứng viên vector chuyển động trong danh sách ứng viên được tạo ra dựa vào quy tắc định trước theo cùng cách ở bộ mã hóa và bộ giải mã. Bù chuyển động sau đó thu bộ dự đoán của khối hiện thời ở dạng dịch chuyển khối được đặt cùng với khối hiện thời trong khung tham chiếu sang khối tham chiếu trong khung tham chiếu, tức là bởi vector chuyển động. Bộ phận dự đoán liên ảnh 144 xuất ra khối dự đoán cho khối hiện thời, trong đó khối dự đoán này giảm thiểu hàm chi phí. Ví dụ, hàm chi phí có thể là sự khác biệt giữa khối hiện thời cần được mã hóa và khối dự đoán của nó, tức là hàm chi phí giảm thiểu khối dư. Ví dụ, sự giảm thiểu khối dư dựa vào việc tính tổng các sự khác biệt tuyệt đối (Sum of Absolute Difference, SAD) giữa tất cả các điểm ảnh (các mẫu) của khối hiện thời và khối ứng viên trong ảnh tham chiếu ứng viên. Tuy nhiên, nói chung, số liệu tương tự khác bất kỳ có thể được sử dụng, chẳng hạn như sai số bình phương trung bình (Mean Square Error, MSE) hoặc số liệu tương tự cấu trúc (Structural Similarity Metric, SSIM).

Tuy nhiên, hàm chi phí cũng có thể là số lượng bit cần để mã hóa liên khối và/hoặc biến dạng do việc mã hóa này. Do đó, quy trình tối ưu tỷ lệ biến dạng có thể được sử dụng để quyết định về việc chọn vector chuyển động và/hoặc nói chung về các tham số mã hóa chẳng hạn như xem liệu sử dụng dự đoán liên ảnh hay nội ảnh cho một khối và với các cài đặt nào.

Bộ phận ước tính nội ảnh 152 và bộ phận dự đoán nội ảnh 154 nhận ở dạng đầu vào khối của khung hoặc ảnh hiện thời cần được mã hóa nội ảnh và một hoặc một số mẫu tham chiếu từ vùng được tái tạo của khung hiện thời. Dự đoán nội ảnh sau đó mô tả các điểm ảnh của khối hiện thời của khung hiện thời về mặt chức năng của các mẫu tham chiếu của khung hiện thời. Bộ phận dự đoán nội ảnh 154 xuất ra khối dự đoán cho khối hiện thời, trong đó khối dự đoán này giảm thiểu khác biệt giữa khối hiện thời cần được mã hóa và khối dự đoán của nó theo cách có lợi, tức là, nó giảm thiểu khối dư. Ví

dự, sự giảm thiểu khối dư có thể dựa vào quy trình tối ưu tỷ lệ biến dạng. Cụ thể là, khối dư đoán được thu ở dạng nội suy định hướng của các mẫu tham chiếu. Định hướng có thể được xác định bằng cách tối ưu tỷ lệ biến dạng và/hoặc bằng cách tính phạm vi tương tự như được đề cập trên đây liên quan đến dự đoán liên ảnh.

Bộ phận ước tính liên ảnh 142 nhận ở dạng đầu vào khối hoặc mẫu hình ảnh được tạo thành phổ quát hơn của khung hoặc ảnh hiện thời cần được mã hóa liên ảnh và hai hoặc nhiều hơn hai ảnh đã được giải mã 231. Dự đoán liên ảnh sau đó mô tả các mẫu hình ảnh hiện thời của khung hiện thời về mặt các vector chuyển động đến các mẫu hình ảnh tham chiếu của các ảnh tham chiếu. Bộ phận dự đoán liên ảnh 142 xuất ra một hoặc nhiều vector chuyển động cho các mẫu hình ảnh hiện thời, trong đó các mẫu hình ảnh tham chiếu này được trở đến bởi các vector chuyển động để giảm thiểu sự khác biệt giữa các mẫu hình ảnh hiện thời cần được mã hóa và các mẫu hình ảnh tham chiếu của nó theo cách có lợi, tức là, nó giảm thiểu mẫu hình ảnh dư. Bộ phận dự đoán cho khối hiện thời sau đó được tạo ra bởi bộ phận dự đoán liên ảnh 144 để mã hóa sự khác biệt.

Sự khác biệt giữa khối hiện thời và dự đoán của nó, tức là khối dư 105, sau đó được biến đổi bởi bộ phận biến đổi 106. Các hệ số biến đổi 107 được lượng tử hóa bởi bộ phận lượng tử hóa 108 và được mã hóa entropy bởi bộ phận mã hóa entropy 170. Do đó, dữ liệu ảnh được mã hóa được tạo ra 171, tức là dòng bit video được mã hóa, bao gồm các khối được mã hóa nội ảnh và các khối được mã hóa liên ảnh và báo hiệu tương ứng (chẳng hạn như chỉ báo chế độ, chỉ báo vector chuyển động, và/hoặc chiều dự đoán nội ảnh). Bộ phận biến đổi 106 có thể áp dụng biến đổi tuyến tính chẳng hạn như biến đổi Fourier hoặc biến đổi cosin rời rạc (DFT/FFT hoặc DCT). Biến đổi sang miền tần số không gian này mang lại ưu điểm là các hệ số thu được 107 thường có các giá trị cao hơn trong các tần số thấp hơn. Do đó, sau khi quét hệ số hiệu dụng (chẳng hạn như zig-zag), và lượng tử hóa, chuỗi giá trị thu được thường có một số giá trị lớn hơn ở đầu và kết thúc bằng chuỗi các số không. Điều này cho phép mã hóa hiệu quả hơn. Bộ phận lượng tử hóa 108 thực hiện nén tổn hao thực tế bằng cách giảm độ phân giải của các giá trị hệ số. Bộ phận mã hóa entropy 170 sau đó gán các giá trị hệ số cho các từ mã nhị phân để tạo ra dòng bit. Bộ phận mã hóa entropy 170 cũng mã hóa thông tin báo hiệu (không được thể hiện trên FIG.1).

FIG.2 thể hiện bộ giải mã video 200. Bộ giải mã video 200 cụ thể bao gồm bộ

đệm ảnh được giải mã 230, bộ phận dự đoán liên ảnh 244 và bộ phận dự đoán nội ảnh 254, là bộ phận dự đoán khối. Bộ đệm ảnh được giải mã 230 được tạo cấu hình để lưu trữ ít nhất một (đối với dự đoán một chiều) hoặc ít nhất hai (đối với dự đoán hai chiều) khung tham chiếu được tái tạo từ dòng bit video được mã hóa, các khung tham chiếu này khác khung hiện thời (khung hiện đang được giải mã) của dòng bit video được mã hóa. Bộ phận dự đoán nội ảnh 254 được tạo cấu hình để tạo ra khối dự đoán, là ước tính khối cần được giải mã. Bộ phận dự đoán nội ảnh 254 được tạo cấu hình để tạo ra dự đoán này dựa vào các mẫu tham chiếu mà được thu từ bộ đệm ảnh được giải mã 230.

Bộ giải mã 200 được tạo cấu hình để giải mã dòng bit video được mã hóa được tạo ra bởi bộ mã hóa video 100, và tốt nhất là cả bộ giải mã 200 và bộ mã hóa 100 tạo ra các dự đoán giống nhau cho khối tương ứng cần được mã hóa/được giải mã. Các bộ phận của bộ đệm ảnh được giải mã 230, khối được tái tạo 215, bộ đệm 216 và bộ phận dự đoán nội ảnh 254 tương tự với các bộ phận của bộ đệm ảnh được giải mã 130, khối được tái tạo 115, bộ đệm 116 và bộ phận dự đoán nội ảnh 154 trên FIG.1.

Bộ giải mã video 200 bao gồm các bộ phận khác cũng có mặt trong bộ mã hóa video 100 ví dụ bộ phận lượng tử hóa ngược 210, bộ phận biến đổi ngược 212, và bộ phận lọc vòng 220, lần lượt tương ứng với bộ phận lượng tử hóa ngược 110, bộ phận biến đổi ngược 112, và bộ phận lọc vòng 120 của bộ mã hóa video 100.

Bộ phận giải mã entropy 204 được tạo cấu hình để giải mã dòng bit video được mã hóa được nhận và để lần lượt thu được các hệ số biến đổi phần dư được lượng tử hóa 209 và thông tin báo hiệu. Các hệ số biến đổi phần dư được lượng tử hóa 209 được cấp cho bộ phận lượng tử hóa ngược 210 và bộ phận biến đổi ngược 212 để tạo ra khối dư. Khối dư này được thêm vào khối dự đoán 265 và khối được thêm này được cấp cho bộ phận lọc vòng 220 để thu được video được giải mã. Các khung của video được giải mã có thể được lưu trữ trong bộ đệm ảnh được giải mã 230 và đóng vai trò là ảnh được giải mã 231 để dự đoán liên ảnh.

Nói chung, các bộ phận dự đoán nội ảnh 154 và 254 trên FIG.1 và FIG.2 có thể sử dụng các mẫu tham chiếu từ vùng đã được mã hóa để tạo ra các tín hiệu dự đoán cho các khối cần được mã hóa hoặc cần được giải mã.

Bộ phận giải mã entropy 204 nhận làm đầu vào của nó dòng bit được mã hóa 171.

Nói chung, dòng bit được phân tích cú pháp trước tiên, tức là các tham số báo hiệu và các phần dư được trích xuất từ dòng bit. Thông thường, cú pháp và ngữ nghĩa của dòng bit được xác định bằng một chuẩn để các bộ mã hóa và các bộ giải mã có thể làm việc theo cách tương thích. Như được mô tả trong phần tình trạng kỹ thuật nêu trên, dòng bit được mã hóa không chỉ bao gồm các phần dư dự đoán. Trong trường hợp dự đoán bù chuyển động, chỉ báo vectơ chuyển động cũng được mã hóa trong dòng bit và được phân tích cú pháp từ đó ở bộ giải mã. Chỉ báo vectơ chuyển động có thể được cho bởi ảnh tham chiếu trong đó vectơ chuyển động được tạo ra và bởi các tọa độ của vectơ chuyển động. Đến lúc này, việc mã hóa các vectơ chuyển động hoàn chỉnh đã được xem xét. Tuy nhiên, cũng chỉ sự khác biệt giữa vectơ chuyển động hiện thời và vectơ chuyển động trước đó trong dòng bit có thể được mã hóa. Cách tiếp cận này cho phép khai thác phần dư giữa các vectơ chuyển động của các khối lân cận.

Để mã hóa một cách hiệu quả ảnh tham chiếu, bộ mã hóa/giải mã H.265 (ITU-T, H265, Series H: các hệ thống nghe nhìn và đa phương tiện: mã hóa video hiệu suất cao) tạo ra danh sách ảnh tham chiếu gán cho danh sách các chỉ số của các khung tham chiếu tương ứng. Khung tham chiếu sau đó được báo hiệu trong dòng bit bằng cách bao gồm trong đó chỉ số danh sách được gán tương ứng. Danh sách này có thể được xác định theo chuẩn hoặc được báo hiệu ở đầu video hoặc tập hợp các khung. Cần lưu ý rằng trong H.265 có hai danh sách ảnh tham chiếu được xác định, được gọi là L0 và L1. Ảnh tham chiếu sau đó được báo hiệu trong dòng bit bằng cách chỉ báo danh sách (L0 hoặc L1) và chỉ báo chỉ số trong danh sách được liên kết với ảnh tham chiếu được mong muốn. Việc tạo ra hai hoặc nhiều hơn hai danh sách có thể có các ưu điểm để bù tốt hơn. Ví dụ, L0 có thể được sử dụng cho cả các lát được dự đoán liên ảnh một chiều và các lát được dự đoán liên ảnh hai chiều trong khi L1 có thể chỉ được sử dụng cho các lát được dự đoán liên ảnh hai chiều. Tuy nhiên, nói chung sáng chế không bị giới hạn ở nội dung bất kỳ của các danh sách L0 và L1.

Các danh sách L0 và L1 có thể được xác định theo chuẩn và được cố định. Tuy nhiên, có thể đạt được sự linh hoạt hơn trong mã hóa/giải mã bằng cách báo hiệu chúng ở đầu chuỗi video. Theo đó, bộ mã hóa có thể tạo cấu hình các danh sách L0 và L1 có các ảnh tham chiếu cụ thể được sắp xếp theo chỉ số. Các danh sách L0 và L1 có thể có cùng kích thước được cố định. Nói chung, có thể có nhiều hơn hai danh sách. Vectơ

chuyển động có thể được báo hiệu trực tiếp bởi các tọa độ trong ảnh tham chiếu. Theo cách khác, như cũng được chỉ định trong H.265, danh sách các vector chuyển động ứng viên có thể được tạo ra và chỉ số được liên kết trong danh sách có vector chuyển động cụ thể có thể được truyền.

Các vector chuyển động của khối hiện thời thường tương quan với các vector chuyển động của các khối lân cận trong ảnh hiện thời hoặc trong các ảnh được mã hóa trước đó. Điều này là vì các khối lân cận có khả năng tương ứng với cùng một đối tượng chuyển động có chuyển động tương tự và chuyển động của đối tượng không có khả năng thay đổi đột ngột theo thời gian. Do đó, việc sử dụng các vector chuyển động trong các khối lân cận làm các bộ dự đoán làm giảm kích thước sự khác biệt vector chuyển động được báo hiệu. Các dự đoán vector chuyển động (Motion Vector Predictor, MVP) thường được thu từ các vector chuyển động đã được mã hóa/giải mã từ lân cận theo không gian hoặc từ lân cận theo thời gian hoặc các khối được đặt cùng trong ảnh tham chiếu. Trong H.264/AVC, điều này được thực hiện bằng cách thực hiện trung bình mức thành phần của ba vector chuyển động lân cận theo không gian. Bằng cách sử dụng cách tiếp cận này, không cần báo hiệu bộ dự đoán. Các MVP thời gian từ khối được đặt cùng trong ảnh tham chiếu chỉ được xem xét trong chế độ được gọi là chế độ trực tiếp thời gian của H.264/AVC. Các chế độ trực tiếp H.264/AVC cũng được sử dụng để thu dữ liệu chuyển động khác các vector chuyển động. Do đó, chúng liên quan nhiều hơn đến khái niệm hợp nhất khối trong HEVC. Trong HEVC, cách tiếp cận ngầm định thu MVP đã được thay thế bằng kỹ thuật đã biết là cạnh tranh vector chuyển động, báo hiệu rõ ràng MVP từ danh sách các MVP, được sử dụng để thu vector chuyển động. Cấu trúc khối cây tứ phân mã hóa thay đổi được trong HEVC có thể dẫn đến một khối có một số khối lân cận có các vector chuyển động là các ứng viên MVP tiềm năng. Lấy lân cận bên trái làm ví dụ, trong trường hợp tệ nhất 64×64 khối dự đoán độ sáng có thể có 16 khối dự đoán độ sáng 4×4 ở bên trái khi 64×64 khối cây mã hóa độ sáng không còn được chia và khối bên trái được chia theo độ sâu lớn nhất.

Dự đoán vector chuyển động nâng cao (Advanced Motion Vector Prediction, AMVP) được đưa vào để sửa đổi cạnh tranh vector chuyển động để giải thích cho cấu trúc khối linh hoạt như vậy. Trong quá trình phát triển HEVC, thiết kế AMVP ban đầu đã được đơn giản hóa đáng kể để mang lại sự cân bằng tốt giữa hiệu quả mã hóa và thiết

kế thân thiện với việc triển khai. Thiết kế ban đầu của AMVP được bao gồm năm MVP từ ba kiểu bộ dự đoán khác nhau: ba vector chuyển động từ các lân cận theo không gian, trung bình của ba bộ dự đoán không gian và vector chuyển động được định tỷ lệ từ khối lân cận được đặt cùng, về thời gian. Ngoài ra, danh sách các bộ dự đoán đã được sửa đổi bằng cách sắp xếp lại để đặt bộ dự đoán chuyển động có thể nhất ở vị trí thứ nhất và bằng cách loại bỏ các ứng viên phần dư để đảm bảo tài nguyên tồn thêm cho việc báo hiệu tối thiểu. Thiết kế cuối cùng của cấu trúc danh sách ứng viên AMVP bao gồm hai ứng viên MVP sau: a) lên đến hai ứng viên MVP không gian được thu từ năm khối lân cận theo không gian; b) một ứng viên MVP thời gian được thu từ hai khối theo thời gian, được đặt cùng khi cả hai ứng viên MVP không gian không có sẵn hoặc chúng giống nhau; và c) các vector chuyển động không khi các ứng viên không gian, thời gian hoặc cả hai đều không có sẵn. Nội dung chi tiết về sự xác định vector chuyển động có thể được tìm thấy trong sách của V. Sze và các đồng tác giả (Ed.), mã hóa video hiệu suất cao (High Efficiency Video Coding, HEVC): Các thuật toán và các kiến trúc, Springer, 2014, cụ thể trong chương 5, nội dung của tài liệu này được đưa vào bản mô tả này bằng cách viện dẫn.

Để cải thiện thêm việc ước tính vector chuyển động mà không tăng thêm tài nguyên tồn thêm cho việc báo hiệu, có thể có lợi khi tinh chỉnh thêm các vector chuyển động được thu ở phía bộ mã hóa và được tạo ra trong dòng bit. Việc tinh chỉnh vector chuyển động có thể được thực hiện ở bộ giải mã mà không có sự hỗ trợ từ bộ mã hóa. Bộ mã hóa trong vòng lặp bộ giải mã của nó có thể sử dụng cùng việc tinh chỉnh để thu được các vector chuyển động tương ứng. Việc tinh chỉnh vector chuyển động được thực hiện trong không gian tìm kiếm bao gồm các vị trí điểm ảnh nguyên và các vị trí điểm ảnh phân số của ảnh tham chiếu. Ví dụ, các vị trí điểm ảnh phân số có thể là các vị trí nửa điểm ảnh hoặc các vị trí một phần tư điểm ảnh hoặc các vị trí phân số khác. Các vị trí điểm ảnh phân số có thể được thu từ các vị trí (cả điểm ảnh) nguyên bằng nội suy chẳng hạn như nội suy hai chiều. Tuy nhiên, sáng chế áp dụng được như nhau cho các không gian tìm kiếm không bao gồm các vị trí điểm ảnh phân số.

Trong dự đoán hai chiều khối hiện thời, hai khối dự đoán được thu bằng cách 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, có thể tạo ra sự thích

ứng với tín hiệu gốc tốt hơn dự đoán một chiều, dẫn đến ít thông tin dư hơn và có thể nén hiệu quả hơn.

Do ở bộ giải mã, khối hiện thời không có sẵn do khối này đang được giải mã, với mục đích tinh chỉnh vector chuyển động, một mẫu được sử dụng, là ước tính khối hiện thời và được tạo ra dựa vào các phần hình ảnh đã được xử lý (tức là được mã hóa ở phía bộ mã hóa và được giải mã ở phía bộ giải mã).

Trước tiên, việc ước tính vector chuyển động thứ nhất MV0 và ước tính vector chuyển động thứ hai MV1 được nhận làm đầu vào ở bộ giải mã 200. Ở phía bộ mã hóa 100, các ước tính vector chuyển động MV0 và MV1 có thể được thu bằng cách so khớp khối và/hoặc bằng cách tìm kiếm trong danh sách các ứng viên (chẳng hạn như danh sách hợp nhất) được tạo thành bởi các vector chuyển động của các khối lân cận khối hiện thời (trong cùng ảnh hoặc trong các ảnh liên kế). MV0 và MV1 sau đó được báo hiệu cho phía bộ giải mã trong dòng bit theo cách có lợi. Tuy nhiên, cần lưu ý rằng nói chung, giai đoạn xác định thứ nhất ở bộ mã hóa có thể cũng được thực hiện bằng cách so khớp mẫu mà sẽ mang lại ưu điểm là giảm tài nguyên tốn thêm cho việc báo hiệu.

Ở phía bộ giải mã 200, các vector chuyển động MV0 và MV1 được thu dựa vào thông tin trong dòng bit theo cách có lợi. MV0 và MV1 được báo hiệu trực tiếp, hoặc được báo hiệu khác nhau, và/hoặc chỉ số trong danh sách vector chuyển động (danh sách hợp nhất) được báo hiệu. Tuy nhiên, sáng chế không bị giới hạn ở việc báo hiệu các vector chuyển động trong dòng bit. Thay vào đó, vector chuyển động có thể được xác định bằng cách so khớp mẫu đã ở giai đoạn thứ nhất, tương ứng với hoạt động bộ mã hóa. So khớp mẫu ở giai đoạn thứ nhất (thu vector chuyển động) có thể được thực hiện dựa vào không gian tìm kiếm khác không gian tìm kiếm của giai đoạn thứ hai, tinh chỉnh. Cụ thể là, việc tinh chỉnh có thể được thực hiện đối với không gian tìm kiếm có độ phân giải cao hơn (tức là khoảng cách giữa các vị trí tìm kiếm ngắn hơn).

Chỉ báo hai ảnh tham chiếu RefPic0 và RefPic1, mà biểu diễn điểm MV0 và MV1, cũng được cấp cho bộ giải mã. Các ảnh tham chiếu được lưu trữ trong bộ đệm ảnh được giải mã ở phía bộ mã hóa và bộ giải mã làm kết quả của quá trình xử lý trước đó, tức là mã hóa và giải mã tương ứng. Một trong số các ảnh tham chiếu này được chọn để tinh chỉnh vector chuyển động bằng cách tìm kiếm. Bộ phận chọn ảnh tham chiếu của bộ mã

hóa và bộ giải mã, lần lượt, được tạo cấu hình để chọn ảnh tham chiếu thứ nhất mà MV0 trở đến và ảnh tham chiếu thứ hai mà MV1 trở đến. Sau khi chọn, bộ phận chọn ảnh tham chiếu xác định xem liệu ảnh tham chiếu thứ nhất hay ảnh tham chiếu thứ hai được sử dụng để thực hiện tinh chỉnh vector chuyển động. Đối với thực hiện tinh chỉnh vector chuyển động, vùng tìm kiếm trong ảnh tham chiếu thứ nhất được xác định xung quanh vị trí ứng viên mà vector chuyển động MV0 trở đến. Các vị trí không gian tìm kiếm ứng viên trong vùng tìm kiếm được phân tích để tìm khối tương tự nhất với khối mẫu bằng cách thực hiện so khớp mẫu trong không gian tìm kiếm và xác định số liệu tương tự chẳng hạn như tổng các sự khác biệt tuyệt đối (Sum of Absolute Difference, SAD). Các vị trí của không gian tìm kiếm biểu thị các vị trí mà trên đó góc trên bên trái của mẫu được so khớp. Như đã được đề cập trên đây, góc trên bên trái chỉ là một quy ước và điểm bất kỳ của không gian tìm kiếm chẳng hạn như điểm trung tâm có thể thường được sử dụng để biểu thị vị trí so khớp.

Theo tài liệu JVET-D0029 được đề cập trên đây, tinh chỉnh vector chuyển động phía bộ giải mã (Decoder-side Motion Vector Refinement, DMVR) có đầu vào là các vector chuyển động ban đầu MV0 và MV1 trở đến hai ảnh tham chiếu tương ứng RefPict0 và RefPict1. Các vector chuyển động ban đầu này được sử dụng để xác định các không gian tìm kiếm tương ứng trong RefPict0 và RefPict1. Hơn nữa, việc sử dụng các vector chuyển động MV0 và MV1, mẫu được tạo ra dựa vào các khối tương ứng (của các mẫu) A và B được trở đến bởi MV0 và MV1 như sau:

$$\text{Mẫu} = \text{hàm} ((\text{Khối } A, \text{Khối } B)).$$

Hàm này có thể là hoạt động cắt mẫu kết hợp với tổng có trọng số mức mẫu. Mẫu này sau đó được sử dụng để thực hiện so khớp mẫu trong các không gian tìm kiếm được xác định dựa vào MV0 và MV1 trong các ảnh tham chiếu tương ứng 0 và 1. Hàm chi phí để xác định so khớp mẫu tốt nhất trong các không gian tìm kiếm tương ứng là $\text{SAD}(\text{Template}, \text{Block and } A')$, trong đó Block and A' là khối mã hóa ứng viên được trở đến bởi ứng viên MV trong không gian tìm kiếm được mở rộng ở vị trí được cho bởi MV0. FIG.3 minh họa việc xác định khối khớp nhất A' và vector chuyển động được tinh chỉnh thu được $MV0'$. Do đó, cùng mẫu được sử dụng để tìm khối khớp nhất B' và vector chuyển động tương ứng $MV1'$ mà trở đến khối B' như được thể hiện trên FIG.3. Nói cách khác, sau khi mẫu được tạo ra dựa vào khối A và B được trở đến bởi các vector

chuyển động ban đầu MV0 và MV1, các vector chuyển động được tinh chỉnh MV0' và MV1' được tìm thấy qua tìm kiếm trên RefPic0 và RefPic1 bằng mẫu này.

Các kỹ thuật thu vector chuyển động cũng thường được gọi là chuyển đổi tăng tốc độ khung (Frame Rate Up-Conversion, FRUC). Các vector chuyển động ban đầu MV0 và MV1 có thể thường được chỉ báo trong dòng bit để đảm bảo rằng bộ mã hóa và bộ giải mã có thể sử dụng cùng điểm ban đầu để tinh chỉnh vector chuyển động. Theo cách khác, các vector chuyển động ban đầu có thể được thu bằng cách tạo ra danh sách ứng viên ban đầu bao gồm một hoặc nhiều ứng viên ban đầu. Đối với mỗi trong số vector chuyển động được tinh chỉnh được xác định và cuối cùng, vector chuyển động được tinh chỉnh giảm thiểu hàm chi phí được chọn.

Cần lưu ý thêm rằng sáng chế không bị giới hạn ở so khớp mẫu như được mô tả trên đây với tham chiếu đến FIG.3. FIG.4 minh họa so khớp mẫu thay thế cũng áp dụng được cho dự đoán một chiều. Nội dung chi tiết có thể được tìm thấy trong tài liệu JVET-A1001, cụ thể là trong phần „2.4.6. Thu vector chuyển động được so khớp mẫu“ của tài liệu JVET-A1001 có tiêu đề là “Algorithm Description of Joint Exploration Test Model 1”, bởi Jianle Chen và các đồng tác giả và truy nhập được ở: <http://phenix.it-sudparis.eu/jvet/>. Mẫu trong cách tiếp cận so khớp mẫu này được xác định là các mẫu liền kề khối hiện thời trong khung hiện thời. Như được thể hiện trên FIG.1, các mẫu đã được tái tạo liền kề biên phía trên và bên trái của khối hiện thời có thể được thực hiện, được gọi là “mẫu dạng hình chữ L”.

FIG.5 minh họa kiểu thu vector chuyển động khác mà cũng có thể được sử dụng. Đầu vào cho quy trình thu vector chuyển động là cờ để chỉ báo xem liệu việc thu vector chuyển động có được áp dụng hay không. Hoàn toàn, đầu vào khác cho quy trình thu là vector chuyển động của khối đã được mã hóa/tái tạo trước đó lân cận (theo thời gian hoặc không gian). Các vector chuyển động của các khối lân cận được sử dụng là các ứng viên cho bước tìm kiếm ban đầu của quy trình thu vector chuyển động. Đầu ra của quy trình này là MV0' (có thể cũng là MV1', nếu dự đoán hai chiều được sử dụng) và lần lượt là các chỉ số ảnh tham chiếu tương ứng refPict0 và có thể là refPict1. Giai đoạn tinh chỉnh vector chuyển động sau đó bao gồm so khớp mẫu như được mô tả trên đây. Sau khi tìm một (dự đoán một chiều) hoặc nhiều (dự đoán hai chiều/dự đoán nhiều khung) vector chuyển động được tinh chỉnh, bộ dự đoán của khối hiện thời được tạo ra (để dự đoán

hai/nhiều chiều bởi dự đoán mẫu có trọng số, hoặc bằng cách tham chiếu đến các mẫu được trỏ đến bởi MV được tinh chỉnh).

Sáng chế không bị giới hạn ở hai phương pháp so khớp này (các ví dụ về so khớp mẫu) được mô tả trên đây. Ví dụ về phương pháp so khớp thứ ba được gọi là so khớp hai chiều (cũng được mô tả trong tài liệu JVET-A1001), cũng có thể được sử dụng để tinh chỉnh vector chuyển động và sáng chế áp dụng theo cách tương tự. So khớp hai chiều được giải thích với tham chiếu đến FIG.6.

Theo so khớp hai chiều, so khớp tốt nhất giữa hai khối dọc theo quỹ đạo chuyển động của khối hiện thời trong hai ảnh tham chiếu khác nhau được tìm kiếm. Như được thể hiện trên FIG.6, so khớp hai chiều được sử dụng để thu thông tin chuyển động của khối hiện thời bằng cách tìm sự khớp gần nhất giữa hai khối dọc theo quỹ đạo chuyển động của khối hiện thời trong hai ảnh tham chiếu khác nhau. Trong so khớp hai chiều hàm chi phí chẳng hạn như SAD(Block cand0', Block cand1') có thể được sử dụng trong đó Block cand0' được trỏ đến bởi MV0 và Block cand1' được trỏ đến bởi MV1.

Để giảm chi phí tính toán, để chọn các ứng viên (các cặp ứng viên) thích hợp trong số các vector chuyển động, mà các hàm chi phí được tính và được so sánh, có thể có ý nghĩa khi xem xét việc thỏa mãn điều kiện sau. Theo giả thiết về một quỹ đạo chuyển động liên tục, các vector chuyển động MV0 và MV1 trỏ đến hai khối tham chiếu sẽ tỷ lệ với các khoảng cách thời gian, tức là, TD0 và TD1, giữa ảnh hiện thời và hai ảnh tham chiếu. Là trường hợp đặc biệt, khi ảnh hiện thời là theo thời gian giữa hai ảnh tham chiếu và khoảng cách thời gian từ ảnh hiện thời đến hai ảnh tham chiếu bằng nhau, thì các sự khác biệt vector chuyển động giữa các vector chuyển động của các cặp tham chiếu đến một ảnh tham chiếu và các vector chuyển động tương ứng của cặp tham chiếu đến ảnh tham chiếu khác sẽ phản chiếu lẫn nhau.

Hệ mạch xử lý 600 được minh họa trên FIG.7. Hệ mạch xử lý này có thể bao gồm phần cứng bất kỳ và cấu hình này có thể được thực hiện bởi kiểu thiết kế chương trình hoặc phần cứng bất kỳ khi kết hợp cả hai. Ví dụ, hệ mạch xử lý có thể được tạo thành bởi một bộ xử lý chẳng hạn như bộ xử lý đa năng có phần mềm tương ứng thực hiện các bước nêu trên. Mặt khác, hệ mạch xử lý này có thể được thực hiện bởi phần cứng chuyên dụng chẳng hạn như ASIC (Application-Specific Integrated Circuit, mạch tích hợp

chuyên dụng) hoặc FPGA (Field-Programmable Gate Array, mảng cổng lập trình được dạng trường) của DSP (Digital Signal Processor, bộ xử lý tín hiệu số) hoặc tương tự.

Hệ mạch xử lý có thể bao gồm một hoặc nhiều trong số các thành phần được đề cập trên đây được kết nối để thực hiện quy trình thu vector chuyển động trên đây. Hệ mạch xử lý 600 bao gồm mạch logic tính toán để thực hiện hai chức năng: thu được vector chuyển động ban đầu (hoặc các vector chuyển động ban đầu nếu dự đoán hai/nhiều chiều được sử dụng) và mẫu nếu so khớp mẫu được sử dụng 610 và tính chỉnh vector chuyển động 620. Hai chức năng này có thể được thực hiện trên cùng một phần cứng hoặc có thể được thực hiện bởi các bộ phận phần cứng riêng biệt chẳng hạn như vector chuyển động ban đầu và bộ phận xác định mẫu 610 và bộ phận tính chỉnh vector chuyển động 620. Hệ mạch xử lý 600 có thể được nối truyền thông với bộ nhớ ngoài 650 trong đó các mẫu ảnh tham chiếu được tái tạo được lưu trữ. Hơn nữa, hệ mạch xử lý 600 có thể còn bao gồm bộ nhớ trong 640 đệm các mẫu trong cửa sổ được truyền từ bộ nhớ ngoài và được sử dụng để xác định vector chuyển động cho khối được xử lý hiện thời. Hệ mạch xử lý có thể được thể hiện trên một chip ở dạng mạch tích hợp.

Cần lưu ý rằng hệ mạch xử lý có thể thực hiện các chức năng khác của bộ mã hóa và/hoặc bộ giải mã được mô tả với tham chiếu đến FIG.1 và FIG.2. Bộ nhớ trong có thể là bộ nhớ trên chip chẳng hạn như bộ nhớ đệm hoặc bộ nhớ đường. Bộ nhớ chip được thực hiện theo cách có lợi trên chip mã hóa/giải mã để tăng tốc độ tính toán. Do kích thước của chip bị giới hạn, nên bộ nhớ trên chip thường nhỏ. Mặt khác, bộ nhớ ngoài có thể có kích thước rất lớn, tuy nhiên việc truy nhập vào bộ nhớ ngoài tốn nhiều năng lượng và việc truy nhập chậm hơn nhiều. Thông thường tất cả thông tin cần thiết được truy suất từ bộ nhớ ngoài vào bộ nhớ trên chip trước khi các tính toán được thực hiện. Trường hợp xấu nhất là truy cập bộ nhớ ngoài (hoặc bằng thông cần được cung cấp khi thiết kế bus bộ nhớ), biểu thị lượng bộ nhớ lớn nhất có thể truyền giữa bộ nhớ ngoài và chip, trong khi giải mã khung hoặc đơn vị mã hóa. Bộ nhớ (đặc biệt là bộ nhớ ngoài) có thể thường chỉ được truy nhập trong các đơn vị khối được xác định trước. Nói cách khác thường không thể truy nhập một điểm ảnh, thay vào đó đơn vị nhỏ nhất (ví dụ 8x8) cần được truy nhập. Kích thước bộ nhớ trên chip cũng là một yếu tố quan trọng trong thiết kế, vì bộ nhớ trên chip lớn hơn sẽ làm tăng chi phí.

Nói cách khác, thiết bị được đề cập trên đây có thể là mạch tích hợp còn bao gồm:

bộ nhớ trong được nhúng trong mạch tích hợp và bộ phận truy nhập bộ nhớ (giao diện) để tìm nạp các mẫu nguyên được đặt trong cửa sổ này từ bộ nhớ ngoài vào bộ nhớ trong.

Thuật ngữ “khối dự đoán” được sử dụng trên đây đề cập đến khối hiện thời cần được dự đoán. Là khối trong hình ảnh có thể được thu bằng cách chia nhỏ hơn hình ảnh thành các khối phân vùng có kích thước bằng nhau hoặc khác nhau (ví dụ bằng cách phân vùng thứ bậc đơn vị cây mã hóa (Coding Tree Unit, CTU) thành các đơn vị nhỏ hơn). Khối này có thể là hình vuông hoặc nói chung là hình chữ nhật vì đây là những hình dạng điển hình cũng được sử dụng trong các bộ mã hóa/giải mã hiện tại. Tuy nhiên, sáng chế không giới hạn ở kích thước/hình dạng bất kỳ của khối.

Thiết bị bao gồm mạch xử lý có thể là bộ mã hóa hoặc bộ giải mã hoặc thậm chí là thiết bị bao gồm bộ mã hóa hoặc bộ giải mã này, ví dụ thiết bị ghi và/hoặc thiết bị phát lại.

Các vị trí mẫu phân số là các vị trí giữa các vị trí mẫu ảnh thực được thu bằng cách tái tạo ảnh tham chiếu mà đã được mã hóa như được thể hiện trên FIG.1. Do đó, các vị trí phân số cần được thu bằng cách nội suy dựa vào các vị trí nguyên gần nhất. Nội dung chi tiết về lọc nội suy có thể được tìm thấy trong phần “5.3 Fractional Sample Interpolation” của sách mã hóa video hiệu suất cao (High Efficiency Video Coding, HEVC) bởi V. Sze và các đồng tác giả, Springer, 2014.

Lọc nội suy thường sử dụng các bộ lọc khác nhau để tạo ra các vị trí (mẫu) điểm ảnh phân số khác nhau. Ví dụ, các bộ lọc tách được 1D sau được áp dụng để tạo ra các vị trí một phần tư điểm ảnh và nửa điểm ảnh trong chuẩn nén video H.265:

Pha	Các hệ số lọc độ sáng
1/4	[-1, 4, -10, 58, 17, -5, 1]/64
1/2	[-1, 4, -11, 40, 40, -11, 4, -1]/64

Như có thể được thấy từ bảng trên đây, lọc nội suy yêu cầu một số mẫu xung quanh vị trí điểm ảnh phân số, tương ứng với các tap bộ lọc (số lượng hệ số trong bảng). Sử dụng các bộ lọc ví dụ trên đây để tạo ra 4 mẫu nguyên vị trí nửa điểm ảnh từ bên

trái/bên trên và bên phải/bên dưới được yêu cầu. Cần lưu ý rằng chiều dài của bộ lọc nội suy cho các vị trí mẫu một phần tư điểm ảnh (là 7 tap) khác các vị trí mẫu nửa điểm ảnh (là 8 tap). Các thuật ngữ mẫu, điểm ảnh và điểm ảnh được sử dụng thay thế lẫn nhau trong bản mô tả này, biểu thị một mẫu hình ảnh (trong một hoặc nhiều thành phần màu) ở thời điểm nhất định. Một mẫu về nguyên tắc là một giá trị cường độ.

Theo một số phương án của sáng chế, bộ lọc nội suy có kích thước tap được xác định trước chỉ truy nhập các mẫu nguyên trong cửa sổ được cho bởi các mẫu nguyên truy nhập được để so khớp mẫu trong không gian tìm kiếm này. Cửa sổ này cần bao gồm nhiều mẫu hơn những mẫu thực sự được sử dụng trong các tính toán của khối dự đoán nhất định. Điều này là do thực tế là hoạt động tìm kiếm tinh chỉnh thường được thực hiện bằng cách sử dụng phương pháp tìm kiếm nhanh (trái ngược với phương pháp tìm kiếm thuật toán), theo một số trong số các mẫu không được đánh giá phụ thuộc vào tiến trình dần dần của hoạt động tìm kiếm. Kết quả là, số lần lặp lại so khớp mẫu cũng như các mẫu được sử dụng trong các tính toán cho hoạt động tìm kiếm tinh chỉnh có thể thay đổi cho mỗi khối dự đoán.

FIG.8 minh họa khối mã hóa (khối dự đoán) và các mẫu tương ứng của cửa sổ. Cần lưu ý rằng các mẫu được thể hiện trên FIG.8 là các mẫu ảnh tham chiếu và khối mã hóa trong bản mô tả này thực tế là khối tương ứng về kích thước và vị trí với khối hiện thời trong khung hiện thời mà vector chuyển động cần được thu trong ảnh tham chiếu. Do đó, thực tế, khối mã hóa trên FIG.8 là khối thực tế được đặt cùng với khối mà bộ dự đoán được tìm kiếm. Tuy nhiên, với mục đích đơn giản, khối này được gọi là “khối mã hóa” trong phần tiếp theo.

Trong ví dụ này, vector chuyển động không được tinh chỉnh MV0 trở đến vị trí mẫu nguyên. Độ chi tiết tìm kiếm tinh chỉnh vector chuyển động là 1 mẫu nguyên, nghĩa là do điểm bắt đầu là mẫu nguyên, nên chỉ các điểm mẫu nguyên được tìm kiếm. Việc tìm kiếm được thực hiện, trong ví dụ này, trong không gian tìm kiếm phát triển dần dần. Điều này có nghĩa là không gian tìm kiếm được nâng cao trong mỗi lần lặp lại bằng cách thêm các vị trí tìm kiếm mới tùy thuộc vào hướng tốt nhất về hàm chi phí cho các vị trí được kiểm tra trước đó.

Cách tiếp cận này được minh họa theo cách được đơn giản hóa trên FIG.9. Trên

FIG.9, vector chuyển động ban đầu được trở đến điểm trung tâm 810. Không gian tìm kiếm dần dần được tạo ra xung quanh vị trí vector chuyển động ban đầu. Trong bước thứ nhất, bốn vị trí liền kề ngay bên trên, bên dưới, bên trái và bên phải vị trí 810 được trở đến bởi vector chuyển động ban đầu cũng như vị trí 810 được trở đến bởi vector chuyển động ban đầu được kiểm tra. Dựa vào hướng mà dẫn đến hàm chi phí thấp nhất trong số năm điểm được kiểm tra, các vị trí khác cần được kiểm tra được thêm vào không gian tìm kiếm. Trong ví dụ này, hàm chi phí thấp nhất có thể được thấy trong điểm bên phải và do đó không gian tìm kiếm đã được mở rộng thêm ba điểm theo hướng ngang bên phải trong bước thứ hai. Trong bước thứ hai hàm chi phí thấp nhất có thể được thấy trong điểm bên phải (đối với điểm giá trị thấp nhất của bước thứ nhất), dẫn đến sự mở rộng không gian tìm kiếm ba điểm theo hướng ngang bên phải. Trong bước thứ ba hàm chi phí thấp nhất được quan sát lần nữa ở điểm bên phải đối với điểm giá trị thấp nhất của bước 2 và dẫn đến sự mở rộng không gian tìm kiếm ba điểm theo hướng ngang bên phải. Theo ví dụ trên FIG.9, nhiều hơn ba bước được thực hiện theo các hướng bên trên, bên trên và bên phải theo thứ tự này. Trong ví dụ này, hình mẫu kim cương (gồm 5 điểm tìm kiếm) được sử dụng cho mỗi bước lặp và không gian tìm kiếm được mở rộng để hoàn thành các điểm tìm kiếm thiếu ở mỗi bước.

Trong mỗi bước lặp xác định không gian tìm kiếm, không gian tìm kiếm này có thể tăng thêm một hoặc nhiều vị trí mẫu nguyên. Bây giờ, trở lại FIG.8 trong ví dụ mà số lượng vòng lặp lớn nhất bằng 4. Do số lượng 4 vòng lặp lớn nhất là khả dụng, nên tất cả các mẫu được mô tả ở bên trái cần được truy suất từ bộ nhớ để thực hiện hoạt động tìm kiếm, trong trường hợp không gian tìm kiếm phát triển dần về bên trái. Tương tự, cần mở rộng 4 mẫu lên phía trên. Do đó, không gian tìm kiếm được mở rộng theo cả hai hướng (trái-phải và trên-dưới) do MV được tinh chỉnh có thể di chuyển theo cả hai hướng và việc triển khai phần cứng yêu cầu tất cả các mẫu có thể được yêu cầu phải được tìm nạp từ bộ nhớ ngoài trước ứng dụng tìm kiếm tinh chỉnh. Nếu không gian tìm kiếm phát triển theo hướng bên dưới hoặc bên phải, thì mở rộng thêm 4 mẫu là cần thiết do so khớp mẫu với mẫu tương ứng với kích thước của khối mã hóa (khối dự đoán) sẽ cần truy nhập một số trong số các mẫu này. Hơn nữa, các mẫu góc (ví dụ trên-phải) cũng cần được tìm nạp từ bộ nhớ, do các phương án thực hiện phần cứng thường không thể tìm nạp các hình dạng không đều (truy cập hình chữ nhật khả thi hơn).

Cần lưu ý rằng sự phát triển không gian tìm kiếm lặp lại được mô tả trên đây chỉ là ví dụ và các quy tắc và số lượng điểm để mở rộng không gian tìm kiếm trong mỗi bước lặp có thể khác nhau, tức là được chỉ định theo cách khác.

FIG.9 cũng thể hiện trường hợp có thể xảy ra do các quy tắc truy nhập bộ nhớ ngoài được mô tả trên đây. Số lượng mẫu được tìm nạp từ bộ nhớ ngoài cao hơn các mẫu được sử dụng thực tế trong bước tính toán. Giả sử rằng mẫu trong bản mô tả này chỉ là một mẫu lớn (cho mục đích đơn giản), các vòng tròn trắng biểu diễn các mẫu được truy suất từ bộ nhớ ngoài và các mẫu được tô đậm được sử dụng thực tế. Tuy nhiên, phần dư thừa này là cần thiết nếu số lượng truy nhập vào bộ nhớ ngoài được giữ ở mức thấp do khi khối hiện thời được bắt đầu cần được xử lý, các mẫu thực sự cần thiết vẫn chưa được biết đến.

Cần lưu ý rằng không gian tìm kiếm cũng có thể được xác định theo cách khác, ví dụ là hình dạng cố định được đặt ở vị trí được trở đến bởi vector chuyển động ban đầu. Hình dạng này có thể là hình dạng bất kỳ chẳng hạn như hình vuông, hình chữ nhật, kim cương, hoặc tương tự.

FIG.10 minh họa ví dụ trong đó không gian tìm kiếm cũng có thể bao gồm các mẫu phân số. Trên FIG.8 và FIG.9, việc tìm kiếm vector chuyển động đã được thực hiện đối với các mẫu nguyên dẫn đến các vị trí được chỉ báo bởi các dấu chấm lớn hơn nét liền nằm trong cửa sổ truy nhập. Nếu bây giờ việc tìm kiếm được thực hiện đối với mẫu có độ phân giải nửa điểm ảnh (chấm nét liền nhỏ hơn), để tạo ra mẫu phân số, được mô tả ở phía bên trái, nhiều hơn ba cột mẫu cũng cần được truy suất từ bộ nhớ, giả sử rằng bộ lọc nội suy đối xứng và có tám tap. Hơn nữa điều tương tự cần được áp dụng ở phía bên trái (mở rộng 3 cột điểm ảnh) do thực tế là hoạt động tìm kiếm là đối xứng (có thể di chuyển sang trái và phải lặp lại) để điểm ảnh phân số có thể được đặt ở bên trái của sổ.

Kết quả là do lọc nội suy số lượng mẫu cần được truy suất từ bộ nhớ được tăng thêm, được chỉ báo bởi đường nét đậm bây giờ cũng được bao gồm các vòng trong nét đứt biểu diễn các vị trí được thêm do nội suy phân số. Tương tự nếu cũng cho phép tìm kiếm một nửa trên mỗi vị trí theo hướng thẳng đứng, thì cửa sổ mẫu được truy cập từ bộ nhớ cũng cần được mở rộng theo hướng dọc (không được hiển thị trong ví dụ của

FG.10), ở trên cùng và dưới cùng.

Cửa sổ truy nhập bộ nhớ được xác định là hình chữ nhật bao quanh tất cả các mẫu cần được truy suất từ bộ nhớ để thực hiện tìm kiếm vector chuyển động cho khối dự đoán (khối mã hóa). Cửa sổ truy nhập bộ nhớ này không chỉ bao gồm các mẫu thực tế được yêu cầu, mà còn bao gồm tất cả các mẫu còn lại có khả năng đang được truy nhập trong quá trình hoạt động tìm kiếm vector chuyển động. Trong ví dụ trên FIG.9, tìm kiếm vector chuyển động được di chuyển sang bên phải. Nhưng cũng có thể đã được di chuyển sang hướng bên trái, điều này không được biết trước. Theo đó, để không truy nhập bộ nhớ ngoài nhiều lần, cửa sổ truy nhập bộ nhớ (hoặc cửa sổ truy nhập) bao gồm tất cả các mẫu truy nhập được bởi quy trình tương ứng.

Để đạt được điều này, sáng chế thực hiện đệm các mẫu được đặt ngoài cửa sổ truy nhập bộ nhớ nếu cần, ví dụ, để nội suy các mẫu phân số.

Việc cấp các mẫu thay thế (đệm các mẫu) có thể còn có lợi liên quan đến so khớp mẫu đối với các không gian tìm kiếm cũng bao gồm các mẫu phân số. Nói cách khác, không gian tìm kiếm cho khối dự đoán có thể bao gồm ít nhất một vị trí mẫu phân số có giá trị được thu bởi lọc nội suy bằng bộ lọc có kích thước tap được xác định trước. Lọc nội suy sử dụng các giá trị vị trí mẫu từ cửa sổ và ít nhất một giá trị thay thế để thu được ít nhất một giá trị vị trí mẫu phân số.

FIG.11 minh họa các bước cơ bản của giản đồ thực hiện thu/tính chỉnh vector chuyển động đã biết như được trình bày trước đây. Trong bước thứ nhất (bước 1), “không gian tìm kiếm” của ít nhất hai vector chuyển động ứng viên (trong chuyển động được minh họa, có chín vector chuyển động ứng viên) được thu. Một cách tùy ý, một trong số các vector chuyển động ứng viên tạo thành không gian tìm kiếm có thể được coi là vector chuyển động ban đầu (theo thuật ngữ được sử dụng trong bản mô tả này cũng được gọi là vector chuyển động “neo” hoặc vector chuyển động neo ứng viên). Vector chuyển động ban đầu hoặc neo có thể, ví dụ, là vector chuyển động nằm trong dòng bit và được báo hiệu rõ ràng hoặc ngầm định cho bộ giải mã.

Trong bước tiếp theo (bước 2), quy trình so khớp được áp dụng đối với không gian tìm kiếm để tìm điểm khớp tốt nhất của không gian tìm kiếm (nói cách khác, điểm của không gian tìm kiếm tối thiểu hàm chi phí, tương ứng với vector chuyển động ứng

viên thích hợp nhất). Trong ví dụ được minh họa, so khớp hai chiều được áp dụng, tuy nhiên, các giản đồ khác để xác định hàm chi phí chẳng hạn như so khớp mẫu có thể được áp dụng như nhau.

Bước cuối cùng trên hình vẽ (Bước 3), dự đoán cuối cùng được thu bởi nội suy trên cơ sở vector chuyển động ứng viên thích hợp nhất được chọn theo quy trình xử lý trong bước 2.

Khái niệm trên FIG.11 còn được mô tả trong sơ đồ luồng trên FIG.12. Trên FIG.12, hình minh họa được đưa ra cụ thể cho trường hợp các cặp vector chuyển động ứng viên, tức là đối với so khớp hai chiều (cũng áp dụng được khi so khớp mẫu, ví dụ, được sử dụng trong dự đoán hai chiều, chẳng hạn như được minh họa trên đây với tham chiếu đến FIG.3).

Trong bước thứ nhất (1310), không gian tìm kiếm của cặp vector chuyển động ứng viên được tạo ra. Bước này tương ứng với “bước 1” trên FIG.11.

Trong bước tiếp theo (1320) dự đoán được tạo ra cho mỗi cặp vector chuyển động trong không gian tìm kiếm, bằng cách áp dụng lọc nội suy. Trong bước tiếp theo (1330), đối với mỗi cặp vector chuyển động trong không gian tìm kiếm, một mẫu được tạo ra (được thu). Như được giải thích trên đây, bước này là tùy chọn, tức là nó chỉ cần trong trường hợp so khớp mẫu mà không cần đối với so khớp hai chiều. Trong bước tiếp theo (1340), đối với mỗi cặp vector chuyển động trong không gian tìm kiếm, chi phí so khớp mẫu (hoặc chi phí so khớp hai chiều) được tính bằng cách so sánh các khối dự đoán được trở đến bởi hai vector chuyển động của cặp (hoặc khối dự đoán của một vector chuyển động ứng viên và mẫu). Trong bước tiếp theo (1370), cặp vector chuyển động thu được tối thiểu hàm chi phí mẫu hoặc hai chiều (nói chung: so khớp tốt nhất, tức là được thu là thích hợp nhất bằng quy trình so khớp) được chọn và được xuất ra ở dạng cặp vector chuyển động cho khối dự đoán. Theo thuật ngữ được giải thích trên đây, điều này có thể được gọi là “cặp vector chuyển động được tinh chỉnh”. Các bước nêu trên 1320, 1330, 1340, và 1370 tương ứng với “bước 2” trên FIG.11.

Bước tiếp theo (1380) tạo ra dự đoán cuối cùng bằng nội suy sử dụng cặp vector chuyển động được chọn trong bước 1370. Bước này tương ứng với “bước 3” trên FIG.11.

FIG.13 minh họa quy trình tinh chỉnh vector chuyển động và tạo ra dự đoán cuối cùng theo các phương án làm ví dụ của sáng chế. Sự khác biệt chính giữa cách tiếp cận trên FIG.13 và cách tiếp cận thông thường được minh họa trên FIG.11 nằm ở chỗ cách tiếp cận theo FIG.13 của sáng chế sử dụng đệm mẫu cho cả tinh chỉnh vector chuyển động và tạo ra bộ dự đoán cuối cùng bằng cách sử dụng lọc nội suy.

Sử dụng các mẫu được đệm để tinh chỉnh vector chuyển động (tùy chọn) được minh họa trong bước bổ sung 2, sẽ được mô tả chi tiết dưới đây với tham chiếu đến các hình vẽ từ FIG.14 đến FIG.18. Nhờ đó, ngay cả trong trường hợp các vector chuyển động ứng viên có các tọa độ phân số (tức là các tọa độ phân số trong các không gian tìm kiếm), các khối ứng viên được mong muốn để thực hiện so khớp hai chiều hoặc mẫu có thể được thực hiện bằng cách sử dụng cửa sổ truy nhập được giới hạn được xác định trước, giống trong trường hợp không gian tìm kiếm chỉ bao gồm các điểm nguyên. Việc thu hàm chi phí thực tế bằng cách áp dụng so khớp hai chiều hoặc mẫu được thực hiện trong bước 3, giống bước 2 trên FIG.11.

Ngoài ra, việc đệm các mẫu (cũng hoặc chỉ) được sử dụng trong bước cuối cùng 4 để thực hiện việc tạo ra bộ dự đoán thực tế. Bước này thường tương ứng với bước 3 trên FIG.11 nhưng khác ở chỗ sử dụng khái niệm đệm. Cụ thể là, trong ví dụ đã cho các giá trị mẫu được đặt trong hai cột bên phải ngoài cùng và trong hàng dưới cùng không được truy nhập mà được đệm bằng các giá trị mẫu đệm dựa vào các giá trị mẫu khác được truy nhập.

Lượng mẫu cần được đệm (được thay thế bằng các giá trị thay thế do không khả dụng) phụ thuộc vào sự khác biệt giữa vector chuyển động (“được tinh chỉnh”) được chọn cuối cùng và xem vector chuyển động (“neo”) ban đầu. Trong ví dụ được minh họa, sự khác biệt là hai điểm ảnh theo hướng dọc và một điểm ảnh theo hướng ngang. Theo đó, để nội suy theo chiều ngang và chiều dọc một điểm ảnh tại điểm giao nhau của hai hình chữ nhật trên hình minh họa bước 5, các giá trị mẫu của hai cột điểm ảnh bên phải cũng như hàng điểm ảnh dưới cùng không có sẵn và phải được thay thế (đệm) bởi các giá trị thay thế phù hợp với các quy tắc định trước như đã nêu ở trên và được thảo luận chi tiết hơn dưới đây.

Thuận lợi là, nếu đệm cũng được sử dụng cho tìm kiếm trong bước 2, thì cùng cửa

sổ truy nhập được sử dụng cho các hoạt động lọc nội suy trong bước 2 và bước 4.

Thuận lợi là các cửa sổ truy nhập bộ nhớ được sử dụng trong bước 2 và bước 4 trên FIG.13 được xác định dựa vào vectơ chuyển động ứng viên cụ thể trong không gian tìm kiếm. Vectơ chuyển động cụ thể (được gọi là vectơ chuyển động neo) có thể, ví dụ, là vectơ chuyển động nằm trong dòng bit và được báo hiệu rõ ràng hoặc ngầm định cho bộ giải mã. Việc xác định chọn vectơ chuyển động neo được thực hiện theo quy tắc được xác định trước mà cả bộ mã hóa và bộ giải mã đều biết.

FIG.14 thể hiện cửa sổ truy nhập bộ nhớ với mục đích tinh chỉnh vectơ chuyển động. Cửa sổ này được mở rộng xung quanh điểm 1010 ở tâm là vị trí được trỏ đến bởi vectơ chuyển động ban đầu. Hơn nữa, vị trí phân số 1020 cũng thuộc không gian tìm kiếm. Tuy nhiên, vị trí phân số 1020 cần được nội suy bởi bộ lọc 6 tap sử dụng 3 vị trí mẫu nguyên ở bên trái vị trí phân số và ba vị trí mẫu nguyên ở bên phải vị trí phân số. Do vị trí phân số 1020 được đặt trên đường có các vị trí mẫu nguyên, nên bộ lọc một chiều có thể được áp dụng để nội suy, ví dụ một vị trí được thể hiện trong ví dụ trên đây được mô tả với tham chiếu đến FIG.10. Như có thể được thấy trên FIG.14, để nội suy mẫu phân số 1020 hai vị trí mẫu khác 1030 sẽ cần được đặt ngoài cửa sổ. Để tránh việc tải bổ sung của các mẫu này từ bộ nhớ ngoài, theo sáng chế, các giá trị của các vị trí ngoài cửa sổ 1030 được thay thế bằng các giá trị được thu trên cơ sở các mẫu trong cửa sổ.

FIG.14 minh họa vị trí mẫu phân số 1020 ở giữa hai vị trí mẫu nguyên và do đó vị trí nửa điểm ảnh (nửa điểm ảnh). Tuy nhiên, sáng chế có thể được áp dụng như nhau cho các vị trí phân số khác chẳng hạn như các vị trí một phần tư hoặc 1/8. Hơn nữa, vị trí mẫu 1020 được đặt ở đường nằm ngang tương ứng với hàng của các vị trí mẫu nguyên do đó lọc 1D ngang có thể được sử dụng để nội suy chính nó. Tuy nhiên, sáng chế không bị giới hạn ở các vị trí phân số này. Thay vào đó, các vị trí phân số có thể được đặt trên đường dọc tương ứng với cột của các vị trí mẫu nguyên do đó lọc 1D dọc có thể được sử dụng để nội suy chính nó. Hơn nữa, các vị trí phân số không cần được đặt trên cùng đường điểm ảnh như các mẫu nguyên. Trong trường hợp này, lọc hai chiều có thể được sử dụng để nội suy các mẫu này; lọc 2D như vậy có thể tách được thành lọc 1D dọc và ngang.

Các ví dụ về các vị trí phân số khác nhau được thể hiện trên FIG.16. Cụ thể là, vị trí phân số 1 là vị trí nội suy có thể áp dụng cho cả lọc ngang và dọc vì nó không được đặt trên một đường với các hàng và cột của các vị trí mẫu nguyên. Vị trí 2 là vị trí nội suy mà chỉ lọc ngang được sử dụng trong khi vị trí 6 là vị trí nội suy mà chỉ lọc dọc được sử dụng. Tất cả các vị trí phân số được thể hiện trên FIG.16 có thể được thu bằng cách chỉ sử dụng các vị trí nguyên trong cửa sổ.

Cửa sổ truy nhập bộ nhớ có thể được xác định theo các cách khác nhau. FIG.18 minh họa ví dụ trong đó cửa sổ truy nhập bộ nhớ được xác định là các phần mở rộng (Extension, EXT) ở bên trái/bên phải hoặc trên/dưới khối mã hóa (tương ứng với vị trí của khối mã hóa được cho bởi vector chuyển động ban đầu). Lượng mở rộng có thể phụ thuộc vào kích thước và hình dạng khối mã hóa hoặc dự đoán. Trên FIG.18, phần mở rộng là 4 mẫu dài theo mỗi hướng (trên, dưới, trái, phải). Tuy nhiên, cần lưu ý rằng EXT cũng có thể nhận các giá trị khác nhau cho các hướng khác nhau, phụ thuộc vào kích thước khối (có thể có kích thước khác nhau theo hướng dọc và ngang) và/hoặc dạng và kích thước không gian tìm kiếm.

Ví dụ, theo một ví dụ, cửa sổ được xác định là N cột mẫu nguyên và M hàng mẫu nguyên đối với khối dự đoán vector chuyển động ban đầu, N và M là các số nguyên khác không. Trên FIG.18, N và M được chỉ báo nhưng có cùng kích thước. Như được đề cập trên đây, N và M có thể có kích thước khác nhau. N và M là các số nguyên và ít nhất một trong số chúng khác không. Việc lấy các tham số N và M và dạng và kích thước mẫu, kích thước của cửa sổ truy nhập có thể được xác định. Cụ thể là, nếu mẫu có T1 hàng và T2 cột, thì kích thước của cửa sổ truy nhập bộ nhớ có thể được tính là $(N+T1+N)$ hàng và $(M+T2+M)$ cột. Điều này là vì tìm kiếm có thể đi N mẫu sang trái hoặc phải dẫn đến 2N mẫu theo chiều ngang và M mẫu lên hoặc xuống dẫn đến 2M mẫu theo chiều dọc. Ngoài ra, mẫu có thể được so khớp với $T1 \times T2$ mẫu trên vị trí bất kỳ trong số $2N \times 2M$ vị trí mẫu.

Mặt khác, đối với các cách tiếp cận cụ thể của các cấu trúc không gian tìm kiếm như cách tiếp cận được mô tả với tham chiếu đến FIG.8 và FIG.9, cửa sổ truy nhập bộ nhớ có thể được xác định về mặt số lượng bước lặp tinh chỉnh lớn nhất (các bước lặp cấu trúc không gian tìm kiếm) và kích thước bước lặp (về mặt khoảng cách mẫu lớn nhất đạt được trong mỗi bước lặp), sau đó có thể được chuyển đổi thành lượng dịch

chuyển lớn nhất sang trái, phải, lên và xuống. Do đó cửa sổ truy nhập bộ nhớ được xác định là dịch chuyển lớn nhất theo mỗi hướng. Ví dụ, 4 bước lặp trong đó mỗi bước lặp có thể nâng cao không gian tìm kiếm trong một vị trí mẫu nguyên lớn nhất cho kết quả là $EXT=4$.

Nói cách khác, theo ví dụ này, hệ mạch xử lý được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu nêu trên trong không gian tìm kiếm được mở rộng theo cách lặp lại theo hướng được cho bởi một (hoặc nhiều) trong số nhiều vị trí khớp tốt nhất của không gian tìm kiếm trong lần lặp gần đây nhất, cửa sổ được xác định bởi số lượng bước lặp lớn nhất được xác định trước. Điều tương tự cũng áp dụng các sửa đổi phù hợp cho trường hợp so khớp hai chiều.

Cần lưu ý rằng nói chung, sáng chế không bị giới hạn ở hình dạng hoặc dạng hoặc kiểu xác định không gian tìm kiếm cụ thể bất kỳ. Ví dụ khác, không gian tìm kiếm là một cửa sổ con hình chữ nhật của cửa sổ hoặc có thể bằng cửa sổ. Sự xác định cửa sổ truy nhập bộ nhớ có thể có liên quan để có thể đưa tham số báo hiệu tương ứng vào dòng bit (ví dụ tham số EXT hoặc các tham số N và M). Tuy nhiên, kích thước cửa sổ truy nhập bộ nhớ cũng có thể được xác định theo chuẩn hoặc thu được trên cơ sở các tham số mã hóa khác (chẳng hạn như kích thước mẫu, kích thước khối dự đoán, độ phân giải hình ảnh, v.v.). Trở lại FIG.14, theo một phương án, cửa sổ truy nhập bộ nhớ để tinh chỉnh được xác định xung quanh vị trí được trỏ đến bởi vector chuyển động không được tinh chỉnh 1010. Cửa sổ nhận dạng số lượng mẫu điểm ảnh lớn nhất cần được truy nhập từ bộ nhớ để thực hiện tìm kiếm tinh chỉnh vector chuyển động. Trong bản mô tả này không gian tìm kiếm tương ứng với cửa sổ truy nhập. Nói cách khác, để đơn giản, trong ví dụ này kích thước mẫu được xem xét là 1×1 , nhưng có thể và trong thực tế sẽ lớn hơn. Cửa sổ truy nhập bộ nhớ thường được xác định là phần mở rộng xung quanh khối mã hóa. Trong ví dụ này, là 4 mẫu từ trái/phải và 4 mẫu từ trên/dưới. Nếu vị trí không gian tìm kiếm hiện tại được kiểm tra trong quá trình so khớp mẫu hoặc hai chiều yêu cầu các mẫu từ bên ngoài cửa sổ truy nhập bộ nhớ, sau đó các mẫu cần để tái tạo khối mã hóa được thu bằng cách đệm. Đó là trường hợp trên FIG.14 đối với vị trí phân số 1020. Nếu so khớp mẫu cần được thực hiện đối với vị trí này hoặc sử dụng vị trí này, thì nó cần được nội suy sử dụng các vị trí được đệm bổ sung 1030 được đặt bên ngoài cửa sổ.

Các mẫu đệm có thể được tạo ra, ví dụ, sử dụng một trong số các phương pháp:

Sao chép mẫu gần nhất.

Phản chiếu dọc theo trục lọc nội suy.

Cụ thể là, sao chép mẫu gần nhất đề cập đến cách tiếp cận trong đó giá trị thay thế được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất. Ví dụ, trên FIG.14, hai giá trị vị trí còn thiếu 1030 sẽ được thay thế bằng giá trị mẫu liền kề ngay bên phải vị trí phân số 1020 cần được nội suy. Tuy nhiên, cần lưu ý rằng cách tiếp cận này chỉ là ví dụ và sáng chế không bị giới hạn ở việc sao chép lân cận đơn thuần. Theo cách khác, một vài vị trí mẫu gần nhất có thể được sử dụng để nội suy các mẫu bị thiếu 1030. Ví dụ, ba mẫu ở biên gần nhất với mẫu bị thiếu có thể được sử dụng để nội suy mẫu bị thiếu ví dụ bằng cách tính trung bình có trọng số bằng tập trọng số tỷ lệ nghịch với khoảng cách của ba mẫu từ mẫu bị thiếu.

Phản chiếu đề cập đến cách tiếp cận theo giá trị thay thế được xác định bằng cách phản chiếu giá trị của vị trí mẫu nguyên được truy nhập tương ứng đối với trục của bộ lọc nội suy. Ví dụ trên FIG.14, hai mẫu bị thiếu 1030 sẽ được thay thế bằng các giá trị mẫu ở bên trái vị trí phân số 1020. Cụ thể là, mẫu thứ hai ở bên phải vị trí phân số 1020 được thay thế bằng giá trị của vị trí thứ hai ở bên trái vị trí phân số 1020, trong khi mẫu thứ ba ở bên phải vị trí phân số 1020 được thay thế bằng giá trị của vị trí thứ ba ở bên trái vị trí phân số 1020.

Cần lưu ý rằng sự kết hợp của hai cách tiếp cận được đề cập trên đây cũng có thể được áp dụng trong đó các mẫu bị thiếu là hàm của các mẫu được phản chiếu và mẫu gần nhất.

Theo một phương án, cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để so khớp mẫu hoặc hai chiều ở các vị trí mẫu nguyên trong không gian tìm kiếm này. Đây là ví dụ trường hợp đối với mẫu trên FIG.14, trong đó các mẫu bổ sung bên ngoài cửa sổ chỉ cần để so khớp mẫu trong các vị trí mẫu không nguyên, phân số. Nói cách khác, trên FIG.14, trong ví dụ nêu trên tọa độ nửa điểm ảnh cần được tìm kiếm yêu cầu mẫu bên ngoài cửa sổ truy nhập bộ nhớ. Do đó hoạt động đệm được áp dụng để tạo ra các mẫu không có sẵn. Tuy nhiên, sáng chế không bị giới hạn ở đó.

FIG.15 thể hiện cách tiếp cận thay thế trong đó cửa sổ không bao gồm tất cả các

vị trí mẫu nguyên truy nhập được để so khớp mẫu ở các vị trí mẫu nguyên trong không gian tìm kiếm này. Trong các ví dụ trước đó kích thước khối mã hóa đã được mô tả là 1×1 để đơn giản. Trên FIG.15 kích thước này là 4×4 mẫu nguyên. Trong ví dụ này, nếu tọa độ tìm kiếm (vị trí mà trên đó so khớp mẫu hoặc hai chiều cần được tính) trở đến mẫu nguyên, thì đệm có thể vẫn được áp dụng nếu một số trong số các mẫu trong khối nằm ngoài cửa sổ truy nhập bộ nhớ. Cách tiếp cận này cho phép giới hạn kích thước của sổ nhiều hơn nữa.

Chi tiết: FIG.15 thể hiện cửa sổ 1120 bao gồm một số trong số các mẫu truy nhập được. Vector chuyển động ban đầu trở đến vị trí mà đánh dấu góc trên bên trái khối mã hóa. Không gian tìm kiếm có thể được coi ngang bằng với cửa sổ. Tuy nhiên, cần lưu ý rằng nó có thể thường nhỏ hơn cửa sổ cũng có thể có ý nghĩa do nếu mẫu lớn hơn 1×1 mẫu nguyên, thì nhất thiết các mẫu ngoài những mẫu được đưa vào không gian tìm kiếm sẽ cần thiết để thực hiện so khớp mẫu. Tuy nhiên, nếu không gian tìm kiếm xác định kích thước của cửa sổ, theo cách khác, các mẫu bị thiếu có thể được đệm. Trên FIG.15 vị trí tìm kiếm làm ví dụ ở hàng thứ hai và cột thứ 10 của cửa sổ được thể hiện. So khớp mẫu có 4×4 mẫu nguyên mẫu lớn cũng yêu cầu các điểm ảnh bên ngoài cửa sổ. Do đó, các mẫu này được đệm, tức là được thay thế bởi các giá trị được xác định hoặc được tính trên cơ sở các mẫu trong cửa sổ. Các cách tiếp cận đệm được đề cập trên đây chẳng hạn như lân cận gần nhất hoặc phản chiếu có thể được áp dụng.

Cửa sổ có thể được xác định bởi kích thước dọc và/hoặc ngang của nó đối với

không gian tìm kiếm, hoặc

khối có kích thước của khối dự đoán được đặt ở vị trí vector chuyển động ban đầu, hoặc

vị trí vector chuyển động ban đầu.

Ví dụ về xác định cửa sổ được thể hiện trên FIG.18 và được mô tả trên đây. Điều này tương ứng với xác định kích thước cửa sổ dựa vào kích thước của khối dự đoán được đặt ở vị trí vector chuyển động ban đầu. Theo cách khác, kích thước cửa sổ có thể được xác định đối với vị trí vector ban đầu bởi kích thước theo một hoặc cả hai trong số các chiều x và y. Ví dụ, cửa sổ có thể được xác định để được đặt xung quanh vị trí được

trở đến bởi vector chuyển động ban đầu và mở rộng A mẫu nguyên theo mỗi trong số hướng lên, xuống, trái, phải. Theo cách khác, nó có thể mở rộng A mẫu nguyên lần lượt sang trái và phải và B mẫu nguyên lần lượt theo hướng lên và xuống. Các cách xác định khác là khả dụng.

Cần lưu ý rằng các ví dụ trên đây đã được đề xuất để nội suy nửa điểm ảnh. Tuy nhiên, sáng chế không giới hạn ở đó. Nói chung, vị trí phân số bất kỳ chẳng hạn như $1/4$, $1/8$, hoặc tương tự có thể được sử dụng, tức là được nội suy sử dụng bộ lọc nội suy tương ứng.

Cùng cách xác định về kích thước cửa sổ cũng áp dụng được cho trường hợp đệm để lọc nội suy để tạo ra bộ dự đoán. Cụ thể là, kích thước dọc và ngang (tức là theo các hướng x và y) được xác định bởi chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.

FIG.17 là sơ đồ luồng thể hiện chi tiết hơn quá trình xử lý được minh họa trên FIG.13. Sơ đồ luồng trên FIG.17 khác sơ đồ luồng trên FIG.12 thể hiện quy trình thông thường ở chỗ các bước 1320 và 1380 lần lượt được thay thế bằng các bước 1720 và 1780. Các bước này khác các bước tương ứng trên FIG.12 ở chỗ đối với cả lọc nội suy để tạo ra các khối dự đoán ứng viên trong (các) ảnh tham chiếu để thực hiện so khớp hai chiều hoặc mẫu cũng như lọc nội suy để tạo ra dự đoán cuối cùng, đệm bằng giá trị mẫu của các mẫu thay thế được áp dụng.

Mạch xử lý được mô tả với tham chiếu đến FIG.7 có thể được sử dụng ở bộ mã hóa và/hoặc bộ giải mã như được thể hiện trên FIG.1 và FIG.2.

Cụ thể là, thiết bị mã hóa có thể được đề xuất để mã hóa các hình ảnh video được chia thành các khối dự đoán thành dòng bit, thiết bị mã hóa này bao gồm: thiết bị xác định vector chuyển động cho khối dự đoán như được mô tả trên đây bao gồm hệ mạch xử lý; và hệ mạch mã hóa để mã hóa sự khác biệt giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được xác định và để tạo ra dòng bit bao gồm sự khác biệt được mã hóa và vector chuyển động ban đầu.

Các bộ phận và các chức năng khác của bộ mã hóa được mô tả trên đây với tham chiếu đến FIG.1 cũng có thể được tạo ra hoặc được thực hiện trong hệ mạch xử lý.

Do đó, thiết bị giải mã có thể được đề xuất để giải mã từ dòng bit các hình ảnh video được chia thành các khối dự đoán, thiết bị giải mã này bao gồm: bộ phận phân tích cú pháp để phân tích cú pháp từ dòng bit vector chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh; thiết bị xác định vector chuyển động được tinh chỉnh cho khối dự đoán như được mô tả trên đây bao gồm hệ mạch xử lý; và hệ mạch giải mã để tái tạo khối dự đoán ở dạng tổng khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào bằng vector chuyển động được tinh chỉnh. Ví dụ, bộ dự đoán có thể được cho trực tiếp bởi vị trí của vector chuyển động được tinh chỉnh. Tuy nhiên, có thể có các bước xử lý khác thu được vector chuyển động của khối dự đoán hiện thời có thể còn thay đổi vector chuyển động (chẳng hạn như lọc, cắt, tinh chỉnh khác hoặc tương tự).

Các bộ phận và các chức năng khác của bộ giải mã được mô tả trên đây với tham chiếu đến FIG.2 cũng có thể được tạo ra hoặc được thực hiện trong hệ mạch xử lý.

Hơn nữa, các phương án của sáng chế đã được mô tả từ quan điểm của thiết bị có hệ mạch xử lý để thực hiện tinh chỉnh vector chuyển động. Tuy nhiên, sáng chế không bị giới hạn ở đó mà còn đề xuất các phương pháp tương ứng bao gồm các bước xử lý tương ứng với các bước để thực hiện phương pháp nêu trên được mô tả hệ mạch xử lý được tạo cấu hình.

Các tap tương ứng với các hệ số lọc. Kích thước tap tương ứng với thứ tự lọc. Trong bản mô tả này, giả sử rằng bộ lọc là bộ lọc tuyến tính. Trong một số ví dụ, bộ lọc có thể đối xứng, tức là có các hệ số đối xứng. Tuy nhiên, sáng chế không bị giới hạn ở các bộ lọc tuyến tính hoặc các bộ lọc tuyến tính hoặc các kiểu bộ lọc bất kỳ. Nói chung, các vị trí phân số có thể được thu theo cách bất kỳ dựa vào các mẫu liền kề.

Phương pháp theo một phương án xác định vector chuyển động cho khối dự đoán bằng các bước sau được minh họa trên FIG.19. Trong bước 1410 vector chuyển động ban đầu và mẫu cho khối dự đoán được thu, được theo sau bởi bước xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu nêu trên trong không gian tìm kiếm. Không gian tìm kiếm được đặt ở vị trí được cho bởi vector chuyển động ban đầu. Cửa sổ truy nhập bộ nhớ được xác định trong bước 1420. Cần lưu ý rằng điều này

không cần được thực hiện một cách rõ ràng. Cửa sổ truy nhập bộ nhớ có thể được xác định trước và đã biết. Do đó bước này là minh họa để chỉ báo rằng hiểu biết về kích thước cửa sổ truy nhập bộ nhớ là đã biết. Sau đó so khớp mẫu được thực hiện trong các bước 1430-1490.

So khớp mẫu hoặc hai chiều truy nhập các vị trí mẫu nguyên trong cửa sổ định trước, cửa sổ định trước này bao gồm không gian tìm kiếm này và các vị trí mẫu nguyên truy nhập được khác để so khớp mẫu hoặc hai chiều; và thay thế ít nhất một vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ này. Các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị thay thế được sử dụng để thực hiện so khớp mẫu. Điều này được thể hiện trên FIG.19: So khớp mẫu hoặc hai chiều được thực hiện cho tất cả các vị trí trong không gian tìm kiếm. Trong bước 1430 nó được kiểm tra xem liệu so khớp mẫu hoặc hai chiều ở vị trí hiện thời có yêu cầu mẫu hoặc các mẫu bên ngoài cửa sổ truy nhập bộ nhớ hay không. Nếu đúng (đúng trong bước 1430), sau đó bước đệm như được mô tả trên đây được thực hiện cho mẫu hoặc các mẫu bên ngoài cửa sổ trong bước 1440. Sau bước này, so khớp mẫu hoặc hai chiều (tức là tìm kiếm) được thực hiện trong bước 1450 ở vị trí hiện thời sử dụng các mẫu được đệm. Theo cách khác, nếu không có các mẫu cần thiết nào ở ngoài cửa sổ (sai trong bước 1430), thì so khớp mẫu hoặc hai chiều (tức là tìm kiếm) được thực hiện trong bước 1450 ở vị trí hiện thời bằng cách sử dụng các mẫu có sẵn. Nếu vị trí hiện thời là vị trí cuối cùng trong không gian tìm kiếm cần được kiểm tra (đúng trong bước 1460), sau đó so khớp mẫu kết thúc. Nếu không (sai trong bước 1460), sau đó vị trí tiếp theo được thu trong bước 1470 và so khớp mẫu tiếp tục như được mô tả trên đây cho vị trí mẫu trong bước 1430.

Sau khi kết thúc so khớp mẫu, vị trí khớp tốt nhất được nhận dạng (ví dụ là vị trí được lưu trữ có chi phí thấp nhất).

FIG.20 minh họa cách tiếp cận làm ví dụ thay thế để tinh chỉnh vectơ chuyển động và tạo ra bộ dự đoán áp dụng được trong chương trình khung theo các phương án của sáng chế. Sự khác biệt giữa cách tiếp cận này và cách tiếp cận trên FIG.13 nằm ở chỗ trong trường hợp trên FIG.20 lọc nội suy có đệm chỉ được áp dụng để tạo ra bộ dự đoán cuối cùng bằng cách sử dụng (cặp) vectơ chuyển động được tinh chỉnh. Đây là bước 5 trên FIG.20 tương ứng với bước 4 trên FIG.13.

Mặt khác, lọc nội suy có đệm không được áp dụng trong tinh chỉnh vector chuyển động (tìm kiếm), tức là bước 2 trên FIG.13. Thay vào đó, cách tiếp cận trên FIG.20 thực hiện phép toán làm tròn để để làm tròn tọa độ đến điểm nguyên gần nhất trong trường hợp vector chuyển động ban đầu (hoặc các vector chuyển động ứng viên ban đầu tạo thành không gian tìm kiếm ban đầu) đến điểm nguyên gần nhất (bước 2 trên FIG.20). Phép toán làm tròn này tương đương với dịch chuyển có vector định trước được chỉ báo là MV_round_diff để mô tả hướng làm tròn. Hiệu suất của phép toán làm tròn đảm bảo rằng không gian tìm kiếm được chuyển đổi để thu được không gian tìm kiếm được chuyển đổi chỉ có các điểm có tọa độ nguyên. Do đó, bước 3 sau đây thu được điểm nguyên khớp nhất bằng so khớp hai chiều (hoặc mẫu) có thể được thực hiện đối với các tọa độ nguyên để lọc nội suy không cần để tạo ra các dự đoán đối với so khớp mẫu/hai chiều.

Ngoài ra, FIG.20 bao gồm bước 4. Bước này là tùy chọn. Trong bước 4, sau khi điểm nguyên khớp nhất trong không gian tìm kiếm được sửa đổi đã được thu trong bước 3, ứng dụng làm tròn được hoàn nguyên để thu được vector chuyển động được tinh chỉnh cuối cùng trong không gian tìm kiếm (ban đầu). Nói cách khác, vector dịch chuyển ngược lại với vector của bước 2 được áp dụng cho vector chuyển động phù hợp nhất (được tinh chỉnh) tạo ra từ bước 3. Nếu bước tùy chọn 4 được bao gồm, thì dự đoán cuối cùng sau đó được thu trên cơ sở vector chuyển động được tinh chỉnh tạo ra từ bước 4. Theo cách khác, bước 3 được bỏ qua và bước 5 được áp dụng trực tiếp cho vector chuyển động tạo ra từ bước 3. Theo cách khác nữa, có thể quyết định, trên cơ sở điều kiện định trước, xem liệu bao gồm hay bỏ qua bước 4.

Sơ đồ luồng tương ứng với cách tiếp cận trên FIG.20 được thể hiện trên FIG.21. Một lần nữa, sơ đồ luồng đã được tạo ra cho trường hợp cặp vector chuyển động thay vì các vector chuyển động đơn được sử dụng để đơn giản hóa mục đích minh họa trên FIG.20.

Bước ban đầu 1310 giống trên FIG.20. Do đó bước 2115 tương ứng với phép toán làm tròn trong bước 2 trên FIG.20. Quy trình được thể hiện trong bản mô tả này là trong phạm vi tổng quát hóa, khi chuyển đổi không gian tìm kiếm ban đầu không bị giới hạn ở phép toán làm tròn, mà trong bản mô tả này, nói chung, là đối với mỗi cặp vector chuyển động trong không gian tìm kiếm ban đầu, biểu diễn cặp vector chuyển động được

tạo ra. Phép toán làm tròn trên FIG.20 là trường hợp đặc biệt của ánh xạ các vectơ chuyển động không gian tìm kiếm ban đầu này đến các đại diện.

Các bước sau, 2130 (tùy chọn đối với so khớp mẫu) và 2140 giống như trên FIG.17, có sự khác biệt duy nhất là các hoạt động tương ứng được thực hiện với các vectơ chuyển động đại diện trong không gian tìm kiếm được sửa đổi. Các bước 2150 và 2160 tương ứng với bước 4 trên FIG.20. Bước 2150 mô tả thiết lập cặp vectơ chuyển động đại diện giảm thiểu chi phí so khớp mẫu (hoặc chi phí so khớp hai chiều) là `Pair_best_rep`.

Bước 2160 (tùy chọn) tương ứng với sự hoàn nguyên phép toán làm tròn được minh họa trong bước 4 trên FIG.20. Trong bước sau (1370), (cặp) vectơ chuyển động của không gian tìm kiếm gốc được xuất ra ở dạng vectơ chuyển động cho khối dự đoán. Các bước 1370 và 1780 giống như trên FIG.17.

Hơn nữa phương pháp mã hóa được đề xuất để mã hóa các hình ảnh video được chia thành các khối dự đoán thành dòng bit, phương pháp mã hóa này bao gồm các bước xác định vectơ chuyển động cho khối dự đoán theo phương pháp bất kỳ trong số các phương pháp được mô tả trên đây; cũng như mã hóa sự khác biệt giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào vectơ chuyển động được xác định và để tạo ra dòng bit bao gồm sự khác biệt được mã hóa và vectơ chuyển động ban đầu.

Phương pháp mã hóa có thể còn bao gồm các bước được mô tả với tham chiếu đến các chức năng ở các khối trên FIG.1.

Hơn nữa, phương pháp giải mã được đề xuất để giải mã từ dòng bit các hình ảnh video được chia thành các khối dự đoán, phương pháp giải mã này bao gồm các bước: phân tích cú pháp từ dòng bit vectơ chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vectơ chuyển động được tinh chỉnh; xác định vectơ chuyển động được tinh chỉnh cho khối dự đoán theo phương pháp bất kỳ trong số các phương pháp được đề cập trên đây; và tái tạo khối dự đoán ở dạng tổng khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vectơ chuyển động được tinh chỉnh.

Phương pháp giải mã này có thể còn bao gồm các bước được mô tả với tham chiếu đến các chức năng của các khối trên FIG.2.

Tuy nhiên, cần lưu ý rằng FIG.1 và FIG.2 không giới hạn theo sáng chế. Chúng chỉ đưa ra một ví dụ không giới hạn về một phương án thực hiện của sáng chế trong bộ mã hóa và/hoặc bộ giải mã hiện có.

Xác định vector chuyển động bằng đệm nẫu như được mô tả trên đây có thể được thực hiện là một phần của mã hóa và/hoặc g, tín hiệu video (ảnh chuyển động). Tuy nhiên, xác định vector chuyển động cũng có thể được sử dụng cho các mục đích khác trong xử lý hình ảnh chẳng hạn như phát hiện sự di chuyển, phân tích sự di chuyển, hoặc tương tự mà không giới hạn được sử dụng để mã hóa/giải mã.

Sự xác định vector chuyển động có thể được thực hiện ở thiết bị. Thiết bị này có thể là sự kết hợp của phần mềm và phần cứng. Ví dụ, sự xác định vector chuyển động có thể được thực hiện bởi chip chẳng hạn như bộ xử lý đa năng, hoặc bộ xử lý tín hiệu số (Digital Signal Processor, DSP), hoặc mảng cổng lập trình được dạng trường (Field Programmable Gate Array, FPGA), hoặc tương tự. Tuy nhiên, sáng chế không bị giới hạn ở phương án thực hiện phần cứng lập trình được. Có thể được thực hiện ở mạch tích hợp chuyên dụng (Application-Specific Integrated Circuit, ASIC) hoặc bởi sự kết hợp của các thành phần phần cứng được đề cập trên đây.

Sự xác định vector chuyển động cũng có thể được thực hiện bởi các lệnh chương trình được lưu trên vật ghi đọc được bằng máy tính. Chương trình này, khi được thực thi, làm cho máy tính thực hiện các bước của các phương pháp được mô tả trên đây. Vật ghi đọc được bằng máy tính có thể là vật ghi bất kỳ mà trên đó chương trình được lưu chẳng hạn như DVD, CD, ổ đĩa (nhanh) USB, đĩa cứng, bộ lưu trữ máy chủ có sẵn qua mạng, v.v..

Bộ mã hóa và/hoặc bộ giải mã có thể được thực hiện ở các thiết bị khác nhau bao gồm bộ TV, hộp giải mã, PC, thiết bị dạng bảng, điện thoại thông minh, hoặc tương tự, tức là thiết bị ghi, mã hóa, chuyển mã, giải mã hoặc phát lại bất kỳ. Có thể là phần mềm hoặc ứng dụng thực hiện các bước phương pháp và được lưu trữ/chạy trên bộ xử lý nằm trong thiết bị điện tử như được đề cập trên đây.

Tóm lại, sáng chế đề cập đến việc xác định vectơ chuyển động bằng cách sử dụng so khớp mẫu hoặc hai chiều và tạo ra bộ dự đoán dựa vào vectơ chuyển động. So khớp mẫu hoặc hai chiều và/hoặc tạo ra bộ dự đoán sử dụng quá trình lọc nội suy. Hoạt động lọc nội suy truy nhập các vị trí mẫu nguyên trong cửa sổ, cửa sổ này bao gồm không gian tìm kiếm này và các vị trí mẫu nguyên truy nhập được khác để so khớp mẫu, đếm ít nhất một vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị mẫu được đếm dựa vào ít nhất một mẫu trong cửa sổ này, và sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị mẫu được đếm để thực hiện so khớp mẫu hoặc hai chiều và/hoặc tạo ra bộ dự đoán.

Ngoài ra, các phương án của sáng chế được đề xuất trong phần tiếp theo. Cần lưu ý rằng cách đánh số được sử dụng trong phần tiếp theo không nhất thiết phải tuân theo cách đánh số được sử dụng trong các phần trước.

Phương án 1: Thiết bị mã hóa để mã hóa khối dự đoán của hình ảnh video thành dòng bit (171), thiết bị mã hóa này bao gồm:

hệ mạch xử lý (600) được tạo cấu hình để

xác định vectơ chuyển động cho khối dự đoán (165) bằng các bước:

thu được ít nhất một vectơ chuyển động ban đầu; và

xác định tinh chỉnh vectơ chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vectơ chuyển động ứng viên, để thu được vectơ chuyển động được tinh chỉnh, trong đó không gian tìm kiếm này được đặt ở vị trí (810) được cho bởi vectơ chuyển động ban đầu;

mã hóa sự khác biệt (105) giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào vectơ chuyển động được xác định và để tạo ra dòng bit (171) bao gồm sự khác biệt được mã hóa và vectơ chuyển động ban đầu; và

xác định bộ dự đoán theo vectơ chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy, trong đó quá trình lọc nội suy này bao gồm các bước

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước,

thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ này, và

sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Phương án 2: Thiết bị giải mã để giải mã từ dòng bit (171) các hình ảnh video được chia thành các khối dự đoán, thiết bị giải mã này bao gồm hệ mạch xử lý được tạo cấu hình để:

phân tích cú pháp từ dòng bit vector chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh;

xác định vector chuyển động cho khối dự đoán bằng các bước:

thu được ít nhất một vector chuyển động ban đầu; và

xác định tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh, trong đó không gian tìm kiếm này được đặt ở vị trí (810) được cho bởi vector chuyển động ban đầu; và

tái tạo khối dự đoán ở dạng tổng khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh được xác định bởi hệ mạch xử lý (600) để xác định vector chuyển động;

trong đó hệ mạch xử lý này còn được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, trong đó quá trình nội suy này bao gồm các bước

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước,

thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ này, và

sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị thay thế

để thực hiện nội suy.

Phương án 3: Thiết bị theo phương án 1 hoặc 2, trong đó giá trị thay thế được xác định bằng cách phản chiếu đối với trục của bộ lọc nội suy giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

Phương án 4: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 3, trong đó bộ lọc nội suy là bộ lọc một chiều hoặc hai chiều tách được.

Phương án 5: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 4, trong đó cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để thu được bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động ban đầu.

Phương án 6: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 5, trong đó cửa sổ được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

khối có kích thước của khối dự đoán được đặt ở vị trí vector chuyển động ban đầu,

vị trí vector chuyển động ban đầu, và

chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.

Phương án 7: Thiết bị theo phương án bất kỳ trong số các phương án từ 1, 2 hoặc 4 đến 6, trong đó giá trị thay thế được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

Phương án 8: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 7, trong đó số lượng giá trị mẫu cần được thay thế phụ thuộc vào sự khác biệt giữa vector chuyển động ban đầu và vector chuyển động được tinh chỉnh.

Phương án 9: Thiết bị theo phương án bất kỳ trong số các phương án trước đó, trong đó hệ mạch xử lý này (600) được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với so khớp mẫu hoặc hai chiều, trong đó

so khớp mẫu hoặc hai chiều nêu trên bao gồm các bước:

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước khác (1120), cửa sổ định trước khác (1120) bao gồm không gian tìm kiếm này và các vị trí mẫu nguyên truy nhập được khác để so khớp mẫu hoặc hai chiều,

thay thế ít nhất một giá trị mẫu (1030) ở vị trí mẫu nguyên được đặt ngoài cửa sổ khác bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ khác nêu trên, và

sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị thay thế để thực hiện so khớp mẫu hoặc hai chiều.

Phương án 10: Thiết bị theo phương án 9, trong đó

không gian tìm kiếm này bao gồm ít nhất một vị trí mẫu phân số (1, 2, 6) có giá trị được thu bởi lọc nội suy bằng bộ lọc có kích thước tap được xác định trước; và

lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được truy nhập nêu trên và ít nhất một giá trị thay thế để thu được ít nhất một giá trị vị trí mẫu phân số nêu trên.

Phương án 11: Thiết bị theo phương án 10, trong đó giá trị thay thế được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng cách phản chiếu đối với trục của bộ lọc giá trị vị trí mẫu nguyên được truy nhập tương ứng.

Phương án 12: Thiết bị theo phương án 9 hoặc 10, trong đó giá trị thay thế được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

Phương án 13: Thiết bị theo phương án bất kỳ trong số các phương án từ 9 đến 12, trong đó cửa sổ định trước khác (1120) được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

không gian tìm kiếm,

khối có kích thước của khối dự đoán được đặt ở vị trí vectơ chuyển động ban đầu,

vị trí vectơ chuyển động ban đầu.

Phương án 14: Thiết bị theo phương án bất kỳ trong số các phương án từ 9 đến 13, trong đó cửa sổ định trước khác (1120) giống cửa sổ định trước nêu trên.

Phương án 15: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 14, trong đó việc tinh chỉnh nêu trên được xác định bằng cách so khớp hai chiều và hệ mạch xử lý này được tạo cấu hình để thu được ít nhất hai vector chuyển động ban đầu trở đến các vị trí trong các ảnh tham chiếu khác nhau (Ref0, Ref1) và tạo thành cặp vector chuyển động ban đầu (Pair_0, Pair_1, ..., Pair_N).

Phương án 16: Thiết bị theo phương án bất kỳ trong số các phương án từ 1 đến 14, trong đó việc tinh chỉnh nêu trên được xác định bằng cách so khớp mẫu, hệ mạch xử lý này còn được tạo cấu hình để

thu được mẫu cho khối dự đoán và

xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu trong không gian tìm kiếm này.

Phương án 17: Phương pháp mã hóa để mã hóa các hình ảnh video được chia thành các khối dự đoán thành dòng bit, phương pháp mã hóa này bao gồm các bước:

xác định vector chuyển động cho khối dự đoán, bao gồm các bước

thu được (1310, 1410) ít nhất một vector chuyển động ban đầu; và

xác định (1320, 1330, 1340, 1720; 2115, 2120, 2130, 2140, 2150, 2160) tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh, trong đó không gian tìm kiếm này được đặt ở vị trí được cho bởi vector chuyển động ban đầu;

mã hóa sự khác biệt giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí dựa vào vector chuyển động được xác định và để tạo ra dòng bit bao gồm sự khác biệt được mã hóa và vector chuyển động ban đầu; và

xác định (1780) bộ dự đoán theo vector chuyển động sử dụng lọc nội suy bằng bộ lọc nội suy, trong đó quá trình lọc nội suy này bao gồm các bước

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước,

thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ này, và

sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện lọc nội suy.

Phương án 18: Phương pháp giải mã để giải mã từ dòng bit các hình ảnh video được chia thành các khối dự đoán, bước giải mã này bao gồm các bước:

phân tích cú pháp từ dòng bit vector chuyển động ban đầu và sự khác biệt được mã hóa giữa khối dự đoán và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh;

xác định vector chuyển động cho khối dự đoán, bao gồm các bước

thu được (1310, 1410) ít nhất một vector chuyển động ban đầu; và

xác định (1320, 1330, 1340, 1720; 2115, 2120, 2130, 2140, 2150, 2160) tinh chỉnh vector chuyển động ban đầu trên cơ sở không gian tìm kiếm của các vector chuyển động ứng viên, để thu được vector chuyển động được tinh chỉnh, trong đó không gian tìm kiếm này được đặt ở vị trí được cho bởi vector chuyển động ban đầu;

tái tạo khối dự đoán ở dạng tổng sự khác biệt được phân tích cú pháp và bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động được tinh chỉnh được xác định bởi bước xác định vector chuyển động; và

xác định (1780) bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, trong đó quá trình nội suy này bao gồm các bước

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước,

thay thế ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ này, và

sử dụng các giá trị mẫu vị trí nguyên được truy nhập cũng như các giá trị thay thế để thực hiện nội suy.

Phương án 19: Phương pháp theo phương án 17 hoặc 18, trong đó giá trị thay thế được xác định bằng cách phản chiếu đối với trục của bộ lọc nội suy giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

Phương án 20: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 19, trong đó bộ lọc nội suy là bộ lọc một chiều hoặc hai chiều tách được.

Phương án 21: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 20, trong đó cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để thu được bộ dự đoán được cho bởi khối dự đoán ở vị trí được chỉ định bởi vector chuyển động ban đầu.

Phương án 22: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 21, trong đó cửa sổ được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

khối có kích thước của khối dự đoán được đặt ở vị trí vector chuyển động ban đầu,

vị trí vector chuyển động ban đầu, và

chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.

Phương án 23: Phương pháp theo phương án bất kỳ trong số các phương án từ 17, 18 hoặc 20 đến 22, trong đó giá trị thay thế được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

Phương án 24: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 23, trong đó số lượng giá trị mẫu cần được thay thế phụ thuộc vào sự khác biệt giữa vector chuyển động ban đầu và vector chuyển động được tinh chỉnh.

Phương án 25: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 24, trong đó bước (1320, 1330, 1340, 1720; 2115, 2120, 2130, 2140, 2150, 2160) xác định tinh chỉnh xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu bằng so khớp mẫu hoặc hai chiều, trong đó

so khớp mẫu hoặc hai chiều nêu trên bao gồm các bước:

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ định trước khác, cửa sổ định trước khác này bao gồm không gian tìm kiếm này và các vị trí mẫu nguyên truy nhập được khác để so khớp mẫu hoặc hai chiều,

thay thế (1440) ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa sổ khác bằng giá trị thay thế dựa vào ít nhất một mẫu trong cửa sổ khác nêu trên, và

sử dụng (1450) các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị thay thế để thực hiện so khớp mẫu hoặc hai chiều.

Phương án 26: Phương pháp theo phương án 25, trong đó

không gian tìm kiếm này bao gồm ít nhất một vị trí mẫu phân số có giá trị được thu bởi lọc nội suy bằng bộ lọc có kích thước tap được xác định trước;

và lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được truy nhập nêu trên và ít nhất một giá trị thay thế để thu được ít nhất một giá trị vị trí mẫu phân số nêu trên.

Phương án 27: Phương pháp theo phương án 26, trong đó giá trị thay thế được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng cách phản chiếu đối với trục của bộ lọc giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

Phương án 28: Phương pháp theo phương án 26 hoặc 27, trong đó giá trị thay thế được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

Phương án 29: Phương pháp theo phương án bất kỳ trong số các phương án từ 25 đến 28, trong đó cửa sổ định trước khác được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

không gian tìm kiếm,

khối có kích thước của khối dự đoán được đặt ở vị trí vector chuyển động ban đầu,

vị trí vector chuyển động ban đầu.

Phương án 30: Phương pháp theo phương án bất kỳ trong số các phương án từ 25

đến 29, trong đó cửa sổ định trước khác giống cửa sổ định trước nêu trên.

Phương án 31: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 30, trong đó việc tinh chỉnh nêu trên được xác định bằng cách so khớp hai chiều và bước thu được này thu được ít nhất hai vector chuyển động ban đầu trở đến các vị trí trong các ảnh tham chiếu khác nhau và tạo thành cặp vector chuyển động ban đầu.

Phương án 32: Phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 31, trong đó việc tinh chỉnh nêu trên được xác định bằng cách so khớp mẫu và bước xác định vector chuyển động này còn bao gồm các bước

thu được (1330; 2130) mẫu cho khối dự đoán và

xác định (1340; 2140) tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu nêu trên trong không gian tìm kiếm này.

Phương án 33: Vật ghi đọc được bằng máy tính lưu trữ các lệnh mà khi được thực thi trên bộ xử lý làm cho bộ xử lý này thực hiện phương pháp theo phương án bất kỳ trong số các phương án từ 17 đến 32.

YÊU CẦU BẢO HỘ

1. Thiết bị mã hóa xác định bộ dự đoán để mã hóa khối hình ảnh video, thiết bị mã hóa này bao gồm:

hệ mạch xử lý (600) được tạo cấu hình để

thu được vector chuyển động ban đầu cho khối này;

thu được vector chuyển động được tinh chỉnh cho khối này dựa vào vector chuyển động ban đầu; và

xác định bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh;

trong đó hệ mạch xử lý được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, và để sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm này dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

2. Thiết bị giải mã xác định bộ dự đoán để giải mã khối hình ảnh video, thiết bị giải mã này bao gồm hệ mạch xử lý được tạo cấu hình để:

thu được vector chuyển động ban đầu cho khối này;

thu được vector chuyển động được tinh chỉnh cho khối dự đoán dựa vào vector chuyển động ban đầu; và

xác định bộ dự đoán cho khối này, theo vector chuyển động được tinh chỉnh;

trong đó hệ mạch xử lý được tạo cấu hình để xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, và để sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm này dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

3. Thiết bị theo điểm 1 hoặc 2, trong đó hệ mạch xử lý (600) được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu dựa vào không gian tìm kiếm của các vector chuyển động ứng viên, và để xác định vector chuyển động được tinh chỉnh dựa vào vector chuyển động ban đầu và việc tinh chỉnh.
4. Thiết bị theo điểm 3, trong đó không gian tìm kiếm được đặt ở vị trí được xác định bởi vector chuyển động ban đầu.
5. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó hệ mạch xử lý (600) được tạo cấu hình để thu được sự khác biệt giữa khối và bộ dự đoán.
6. Thiết bị theo điểm 5, trong đó hệ mạch xử lý (600) còn được tạo cấu hình để tái tạo khối dựa vào bộ dự đoán và sự khác biệt giữa khối và bộ dự đoán.
7. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó giá trị mẫu được đệm được xác định bằng cách phản chiếu đối với trục của bộ lọc nội suy giá trị của vị trí mẫu nguyên được truy nhập tương ứng.
8. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó bộ lọc nội suy là bộ lọc một chiều hoặc bộ lọc hai chiều tách được.
9. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để thu được bộ dự đoán ở vị trí được chỉ định bởi vector chuyển động ban đầu.
10. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó cửa sổ được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:
 - khối có kích thước của khối được đặt ở vị trí vector chuyển động ban đầu,
 - vị trí vector chuyển động ban đầu, và
 - chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.
11. Thiết bị theo điểm bất kỳ trong số các điểm từ 1 đến 6 hoặc 8 đến 10, trong đó giá trị mẫu được đệm cho vị trí mẫu nguyên bên ngoài cửa sổ được xác định bằng giá trị mẫu của ít nhất một vị trí mẫu nguyên được sử dụng được đặt bên trong cửa sổ, gần nhất

với vị trí mẫu nguyên được đặt bên ngoài cửa sổ.

12. Thiết bị theo điểm bất kỳ trong số các điểm từ 1 đến 11, trong đó số lượng giá trị mẫu cần được đệm phụ thuộc vào sự khác biệt giữa vector chuyển động ban đầu và vector chuyển động được tinh chỉnh.

13. Thiết bị theo điểm bất kỳ trong số các điểm trước đó, trong đó hệ mạch xử lý (600) được tạo cấu hình để xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu bằng so khớp mẫu hoặc hai chiều, trong đó

quá trình so khớp mẫu hoặc hai chiều này bao gồm các bước:

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong cửa sổ khác (1120), cửa sổ khác (1120) bao gồm không gian tìm kiếm và các vị trí mẫu nguyên khác truy nhập được để so khớp mẫu hoặc hai chiều,

đệm ít nhất một giá trị mẫu (1030) ở vị trí mẫu nguyên được đặt ngoài cửa sổ khác bằng giá trị mẫu được đệm dựa vào ít nhất một mẫu trong cửa sổ khác này, và

sử dụng các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị mẫu được đệm để thực hiện so khớp mẫu hoặc hai chiều.

14. Thiết bị theo điểm 13, trong đó

không gian tìm kiếm này bao gồm ít nhất một vị trí mẫu phân số (1, 2, 6) có giá trị được thu bởi lọc nội suy bằng bộ lọc có kích thước tap được xác định trước; và

lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được truy nhập này và ít nhất một giá trị mẫu được đệm để thu được ít nhất một giá trị vị trí mẫu phân số.

15. Thiết bị theo điểm 14, trong đó giá trị mẫu được đệm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng cách phản chiếu đối với trục của bộ lọc giá trị vị trí mẫu nguyên được truy nhập tương ứng.

16. Thiết bị theo điểm 13 hoặc 14, trong đó giá trị mẫu được đệm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng giá trị ở các vị trí mẫu nguyên

được truy nhập gần nhất.

17. Thiết bị theo điểm bất kỳ trong số các điểm từ 13 đến 16, trong đó cửa sổ khác (1120) được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

không gian tìm kiếm,

khối có kích thước của khối được đặt ở vị trí vector chuyển động ban đầu,

vị trí vector chuyển động ban đầu.

18. Thiết bị theo điểm bất kỳ trong số các điểm từ 13 đến 17, trong đó cửa sổ khác (1120) giống cửa sổ này.

19. Thiết bị theo điểm bất kỳ trong số các điểm từ 1 đến 18, trong đó việc tinh chỉnh được xác định bằng cách so khớp hai chiều và hệ mạch xử lý này được tạo cấu hình để thu được ít nhất hai vector chuyển động ban đầu trở đến các vị trí trong các ảnh tham chiếu khác nhau (Ref0, Ref1) và tạo thành cặp vector chuyển động ban đầu (Pair_0, Pair_1, ..., Pair_N).

20. Thiết bị theo điểm bất kỳ trong số các điểm từ 1 đến 19, trong đó việc tinh chỉnh được xác định bằng cách so khớp mẫu, hệ mạch xử lý này còn được tạo cấu hình để

thu được mẫu cho khối và

xác định tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu trong không gian tìm kiếm này.

21. Phương pháp mã hóa xác định bộ dự đoán để mã hóa khối hình ảnh video, phương pháp mã hóa này bao gồm các bước:

thu được (1310, 1410) vector chuyển động ban đầu cho khối này;

thu được (1320, 1330, 1340, 1720; 2115, 2120, 2130, 2140, 2150, 2160) vector chuyển động được tinh chỉnh cho khối này dựa vào vector chuyển động ban đầu; và

xác định (1780) bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh,

trong đó bước xác định (1780) xác định bộ dự đoán theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, và sử dụng ít nhất một giá trị mẫu của vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm này dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

22. Phương pháp giải mã xác định bộ dự đoán để giải mã khối hình ảnh video, phương pháp giải mã này bao gồm các bước:

thu được vector chuyển động ban đầu cho khối này;

thu được vector chuyển động được tinh chỉnh cho khối này dự đoán dựa vào vector chuyển động ban đầu; và

xác định (1780) bộ dự đoán cho khối này theo vector chuyển động được tinh chỉnh sử dụng nội suy bằng bộ lọc nội suy, sử dụng ít nhất một giá trị mẫu vị trí mẫu nguyên được đặt bên trong cửa sổ và giá trị mẫu được đệm cho vị trí mẫu nguyên được đặt bên ngoài cửa sổ để lọc nội suy, trong đó giá trị mẫu được đệm này dựa vào ít nhất một giá trị mẫu của mẫu được đặt bên trong cửa sổ.

23. Phương pháp theo điểm 21 hoặc 22, phương pháp này còn bao gồm bước xác định tinh chỉnh vector chuyển động ban đầu dựa vào không gian tìm kiếm của các vector chuyển động ứng viên, trong đó vector chuyển động được tinh chỉnh được xác định dựa vào vector chuyển động ban đầu và việc tinh chỉnh.

24. Phương pháp theo điểm 23, trong đó không gian tìm kiếm được đặt ở vị trí được xác định bởi vector chuyển động ban đầu.

25. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 24, phương pháp này còn bao gồm bước thu được sự khác biệt giữa khối và bộ dự đoán.

26. Phương pháp theo điểm 25, phương pháp này còn bao gồm bước tái tạo khối dựa vào bộ dự đoán và sự khác biệt giữa khối và bộ dự đoán.

27. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 26, trong đó giá trị mẫu được đệm được xác định bằng cách phản chiếu đối với trục của bộ lọc nội suy giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

28. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 27, trong đó bộ lọc nội suy là bộ lọc một chiều hoặc bộ lọc hai chiều tách được.

29. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 28, trong đó cửa sổ được tạo thành bởi tất cả các vị trí mẫu nguyên truy nhập được để thu được bộ dự đoán ở vị trí được chỉ định bởi vectơ chuyển động ban đầu.

30. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 29, trong đó cửa sổ được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

khối có kích thước của khối được đặt ở vị trí vectơ chuyển động ban đầu,

vị trí vectơ chuyển động ban đầu, và

chiều dài dọc và ngang của bộ lọc nội suy được sử dụng để xác định bộ dự đoán.

31. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 26 hoặc 28 đến 30, trong đó giá trị mẫu được đệm cho vị trí mẫu nguyên bên ngoài cửa sổ được xác định bằng giá trị mẫu của ít nhất một vị trí mẫu nguyên được sử dụng được đặt bên trong cửa sổ, gần nhất với vị trí mẫu nguyên được đặt bên ngoài cửa sổ.

32. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 31, trong đó số lượng giá trị mẫu cần được đệm phụ thuộc vào sự khác biệt giữa vectơ chuyển động ban đầu và vectơ chuyển động được tinh chỉnh.

33. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 32, trong đó bước (1320, 1330, 1340, 1720; 2115, 2120, 2130, 2140, 2150, 2160) thu được sự tinh chỉnh xác định tinh chỉnh vectơ chuyển động ban đầu bằng cách so khớp mẫu bằng so khớp mẫu hoặc hai chiều, trong đó

quá trình so khớp mẫu hoặc hai chiều này bao gồm các bước:

truy nhập các giá trị mẫu ở các vị trí mẫu nguyên trong một cửa sổ khác, cửa sổ khác này bao gồm không gian tìm kiếm và các vị trí mẫu nguyên khác truy nhập được để so khớp mẫu hoặc hai chiều,

đệm (1440) ít nhất một giá trị mẫu ở vị trí mẫu nguyên được đặt ngoài cửa

sổ khác bằng giá trị mẫu được đếm dựa vào ít nhất một mẫu trong cửa sổ khác này, và

sử dụng (1450) các giá trị vị trí mẫu nguyên được truy nhập cũng như các giá trị mẫu được đếm để thực hiện so khớp mẫu hoặc hai chiều.

34. Phương pháp theo điểm 33, trong đó

không gian tìm kiếm bao gồm ít nhất một vị trí mẫu phân số có giá trị được thu bởi lọc nội suy bằng bộ lọc có kích thước tap được xác định trước;

và lọc nội suy sử dụng các giá trị vị trí mẫu nguyên được truy nhập này và ít nhất một giá trị mẫu được đếm để thu được ít nhất một giá trị vị trí mẫu phân số.

35. Phương pháp theo điểm 34, trong đó giá trị mẫu được đếm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng cách phản chiếu đối với trục của bộ lọc giá trị của vị trí mẫu nguyên được truy nhập tương ứng.

36. Phương pháp theo điểm 34 hoặc 35, trong đó giá trị mẫu được đếm được sử dụng trong quá trình so khớp mẫu hoặc hai chiều được xác định bằng giá trị ở các vị trí mẫu nguyên được truy nhập gần nhất.

37. Phương pháp theo điểm bất kỳ trong số các điểm từ 33 đến 36, trong đó cửa sổ khác được xác định bởi kích thước dọc và/hoặc ngang của nó đối với ít nhất một trong số:

không gian tìm kiếm,

khối có kích thước của khối được đặt ở vị trí vectơ chuyển động ban đầu,

vị trí vectơ chuyển động ban đầu.

38. Phương pháp theo điểm bất kỳ trong số các điểm từ 33 đến 37, trong đó cửa sổ khác giống cửa sổ này.

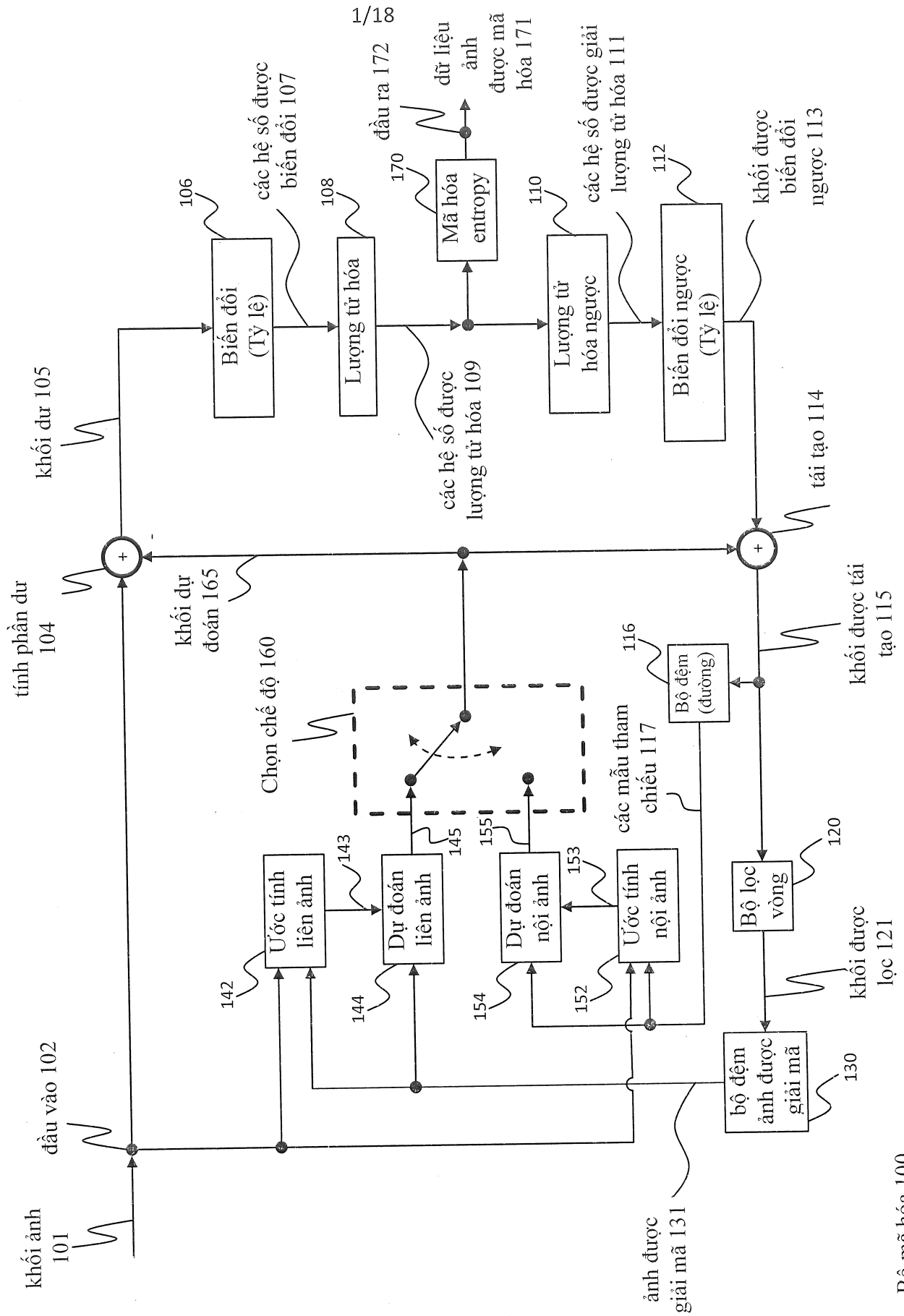
39. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 38, trong đó việc tinh chỉnh được xác định bằng cách so khớp hai chiều và bước thu được này thu được ít nhất hai vectơ chuyển động ban đầu trở đến các vị trí trong các ảnh tham chiếu khác nhau và tạo thành cặp vectơ chuyển động ban đầu.

40. Phương pháp theo điểm bất kỳ trong số các điểm từ 21 đến 39, trong đó việc tinh chỉnh được xác định bằng cách so khớp mẫu và bước xác định vector chuyển động còn bao gồm các bước:

thu được (1330; 2130) mẫu cho khối và

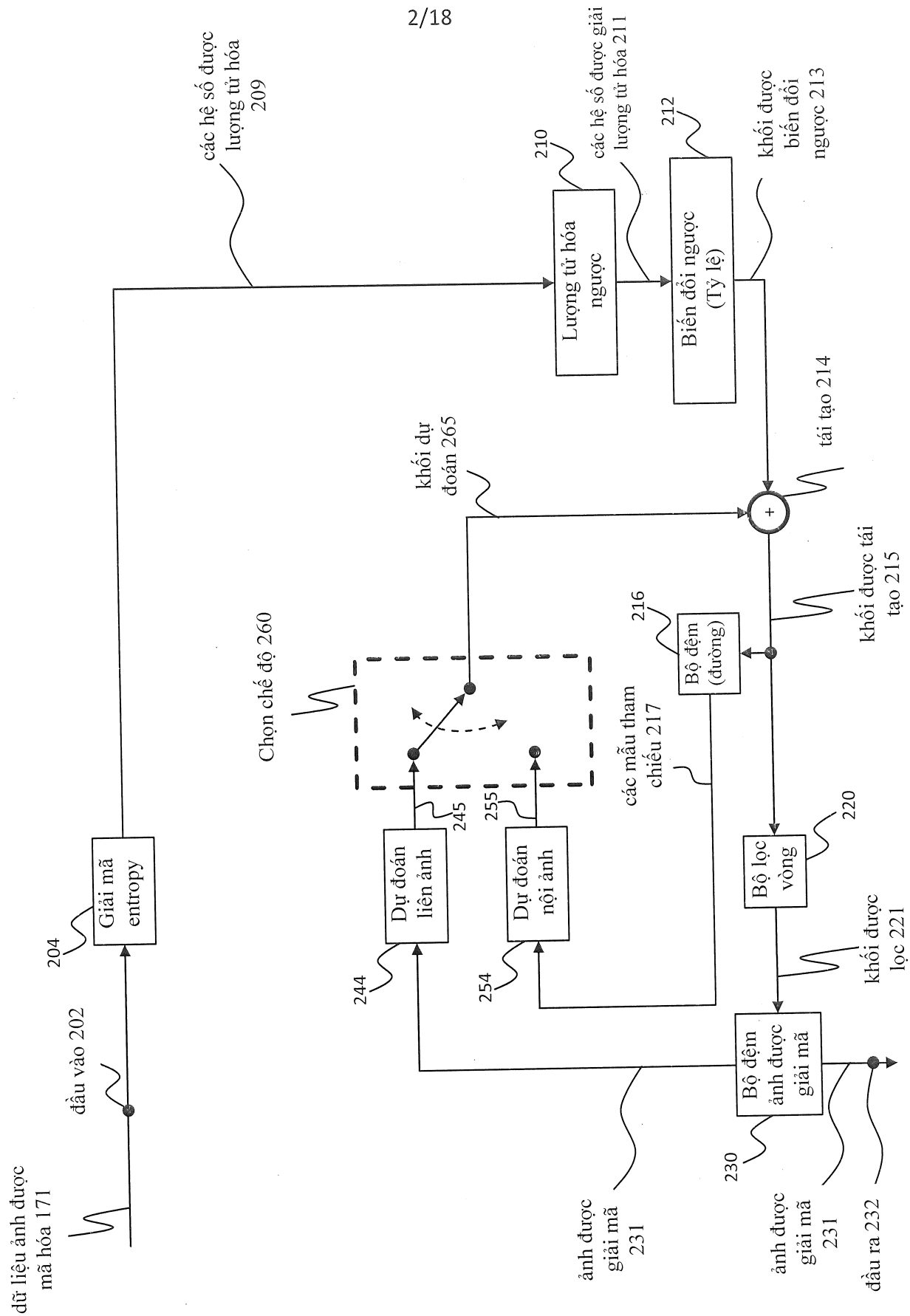
xác định (1340; 2140) tinh chỉnh vector chuyển động ban đầu bằng cách so khớp mẫu với mẫu trong không gian tìm kiếm.

41. Vật ghi đọc được bằng máy tính lưu trữ các lệnh mà khi được thực thi trên bộ xử lý làm cho bộ xử lý này thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 23 đến 40.



150

Bộ mã hóa 100



Bộ giải mã 200

Fig. 2

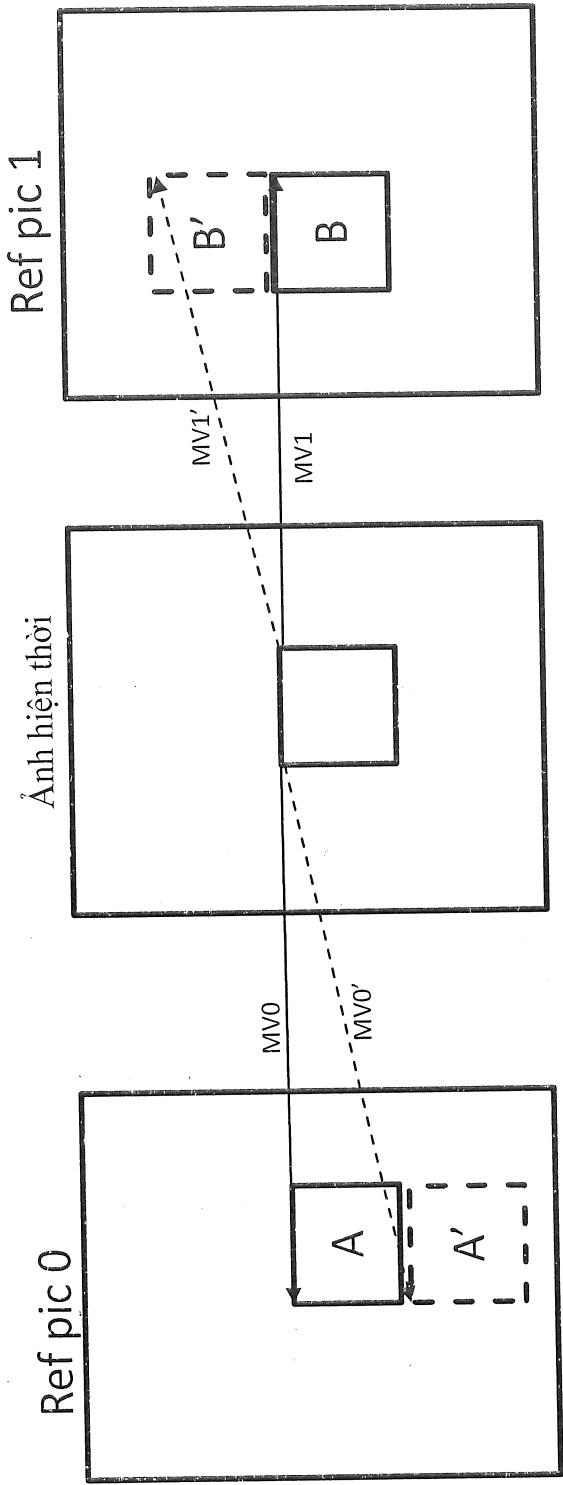


Fig. 3

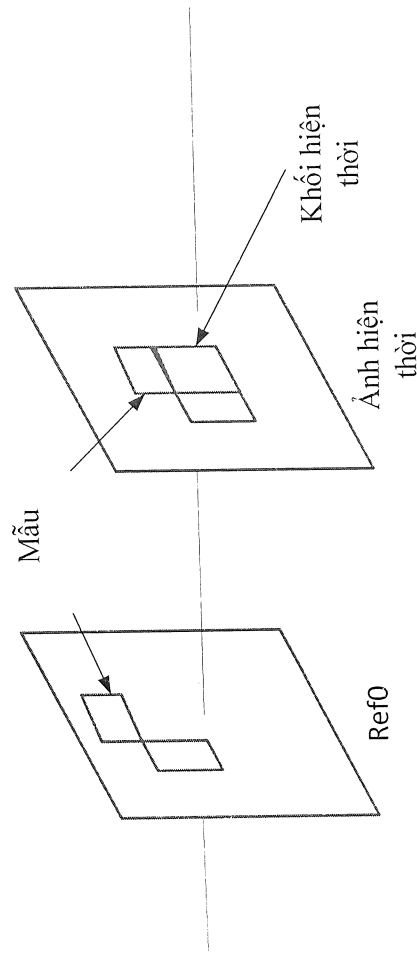


Fig. 4

4/18

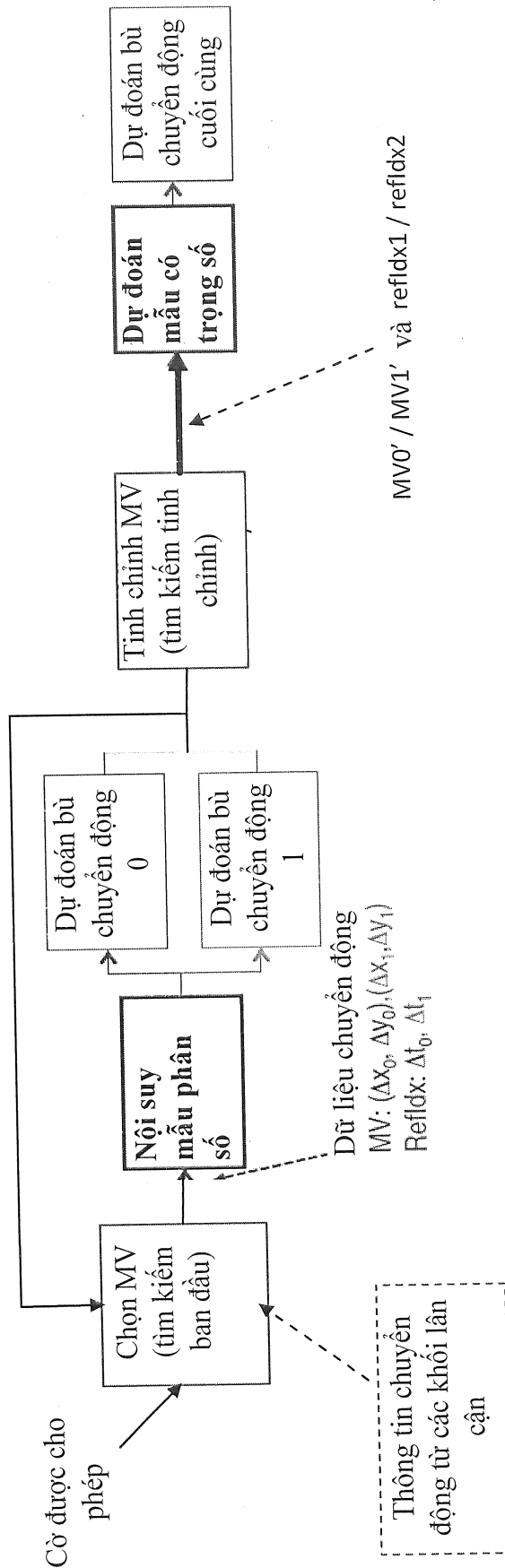


Fig. 5

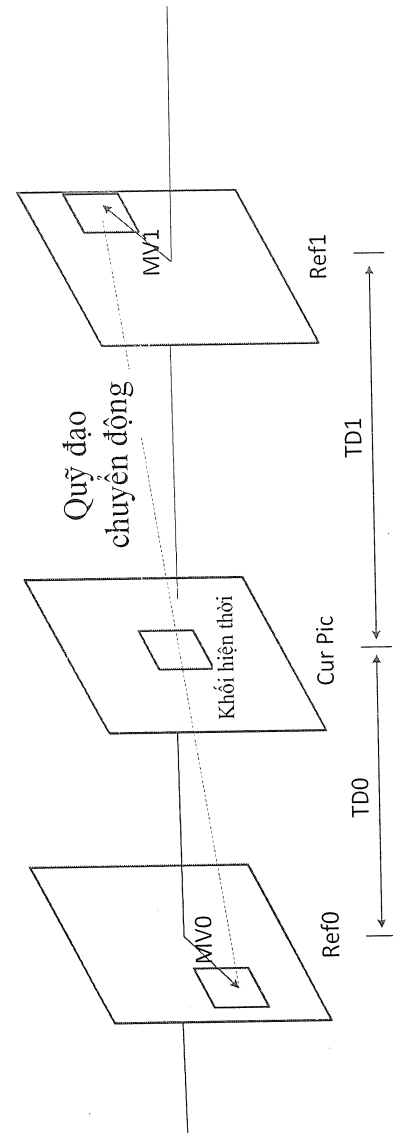


Fig. 6

5/18

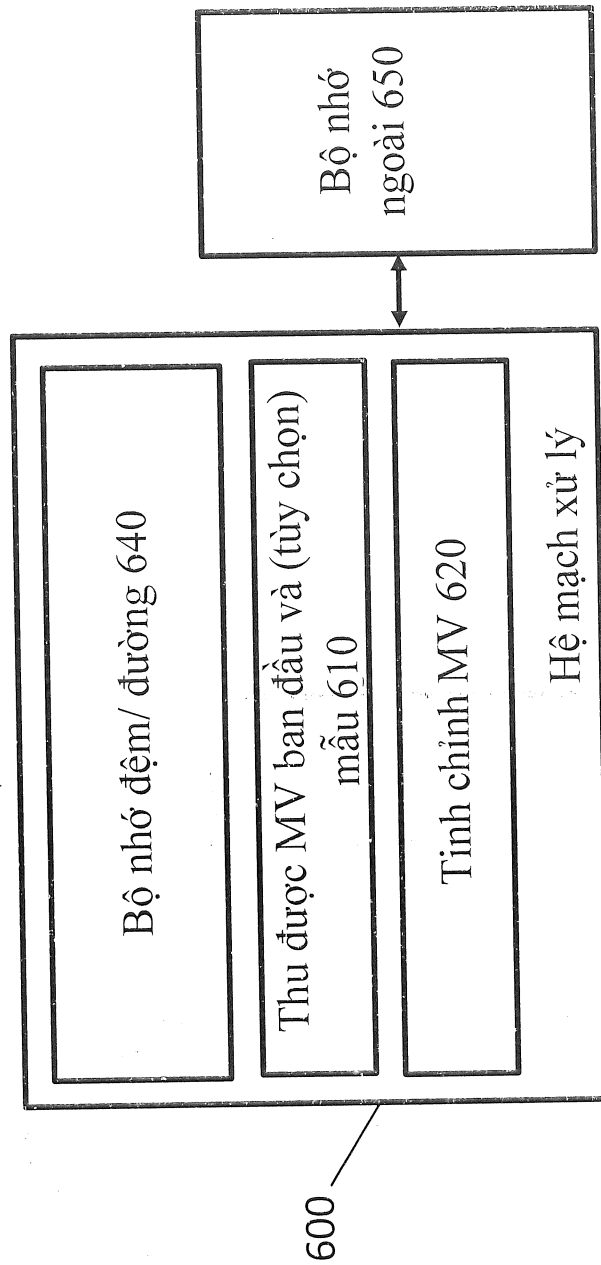


Fig. 7

6/18

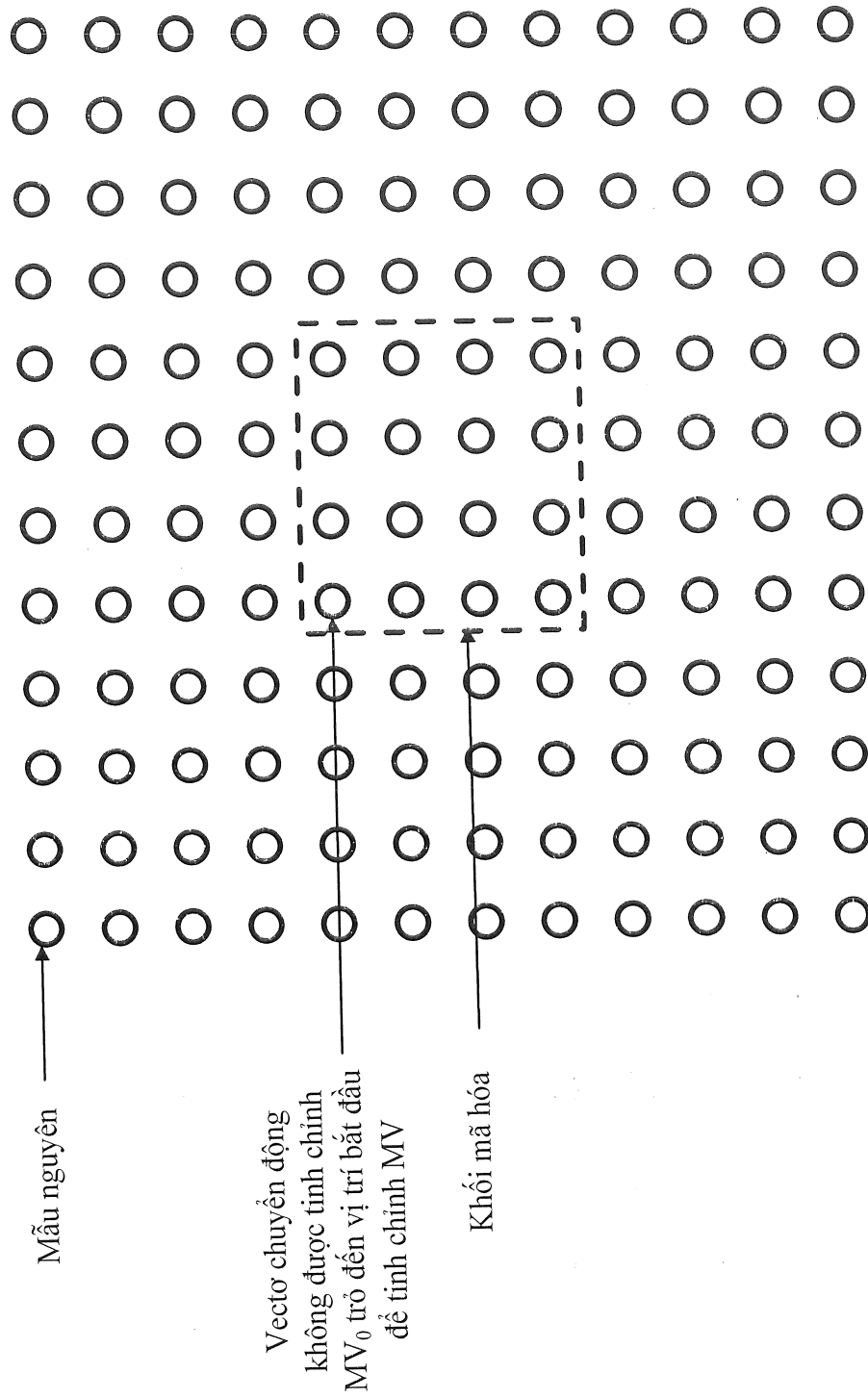


Fig. 8

7/18

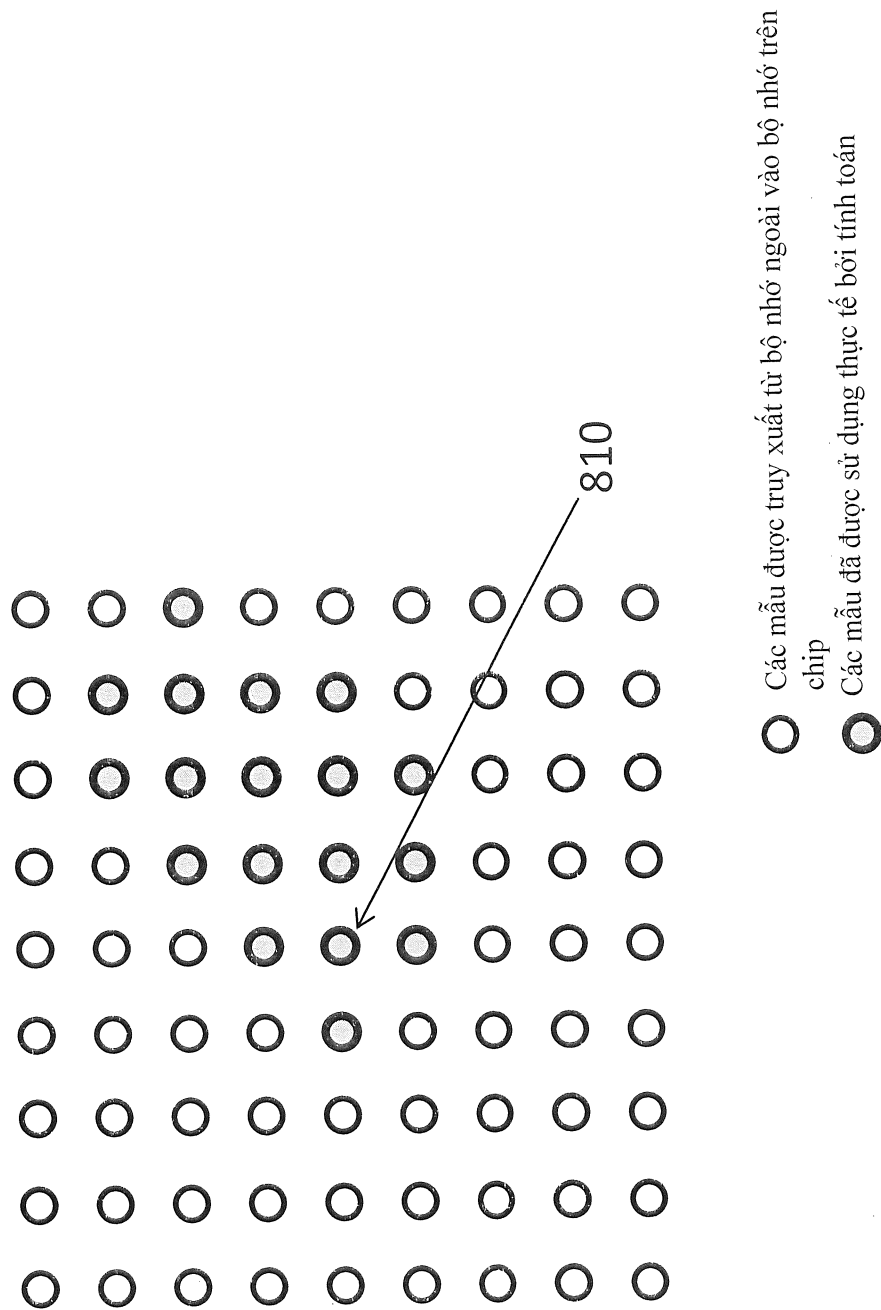


Fig. 9

8/18

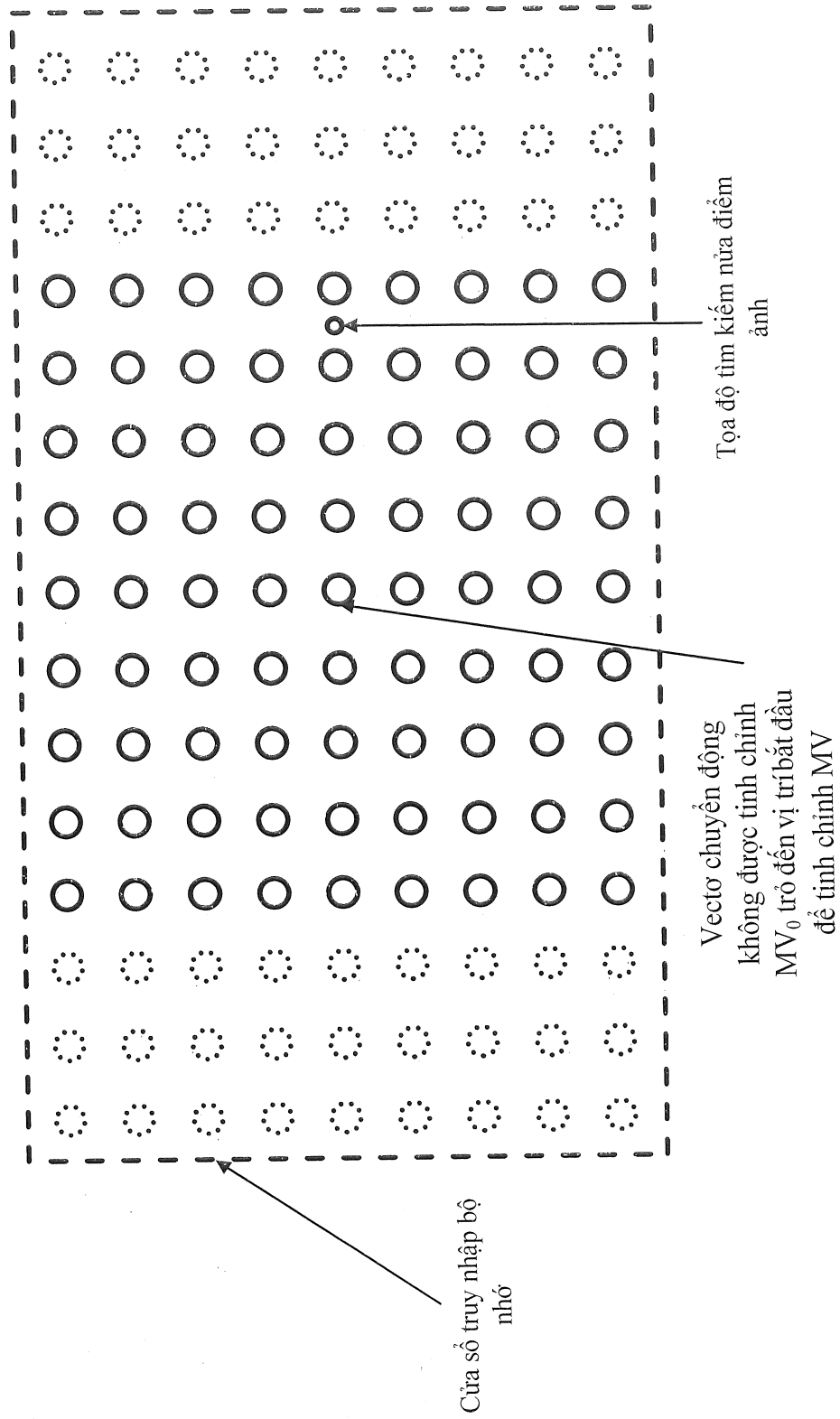


Fig. 10

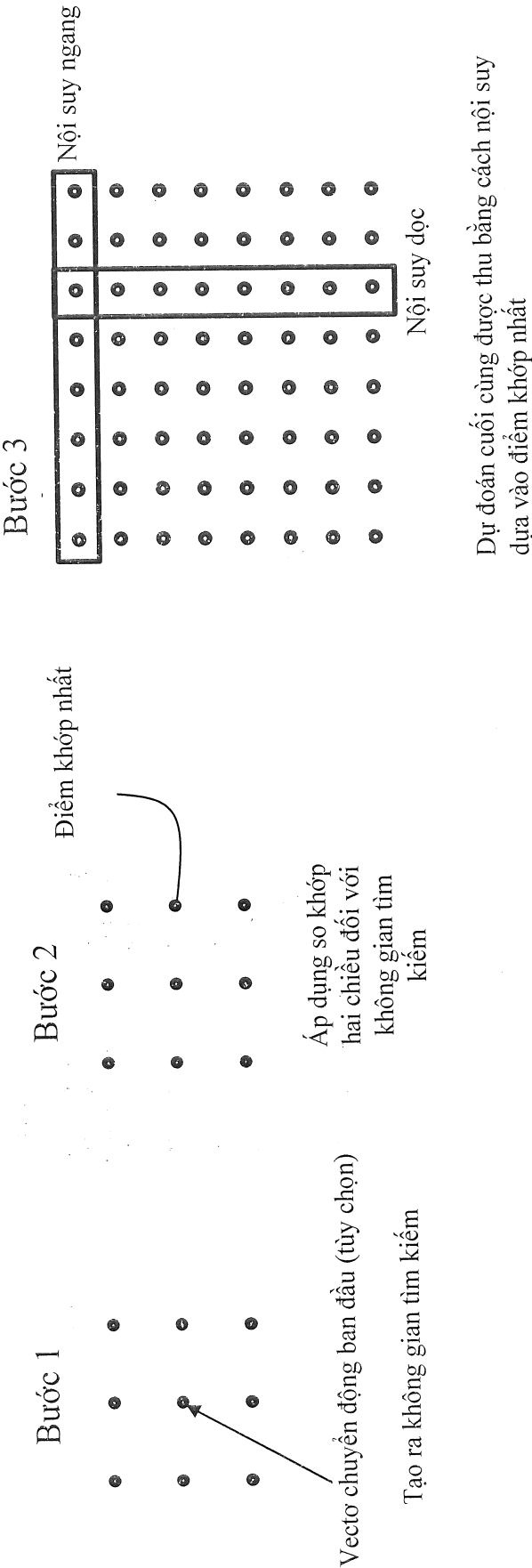


Fig. 11

10/18

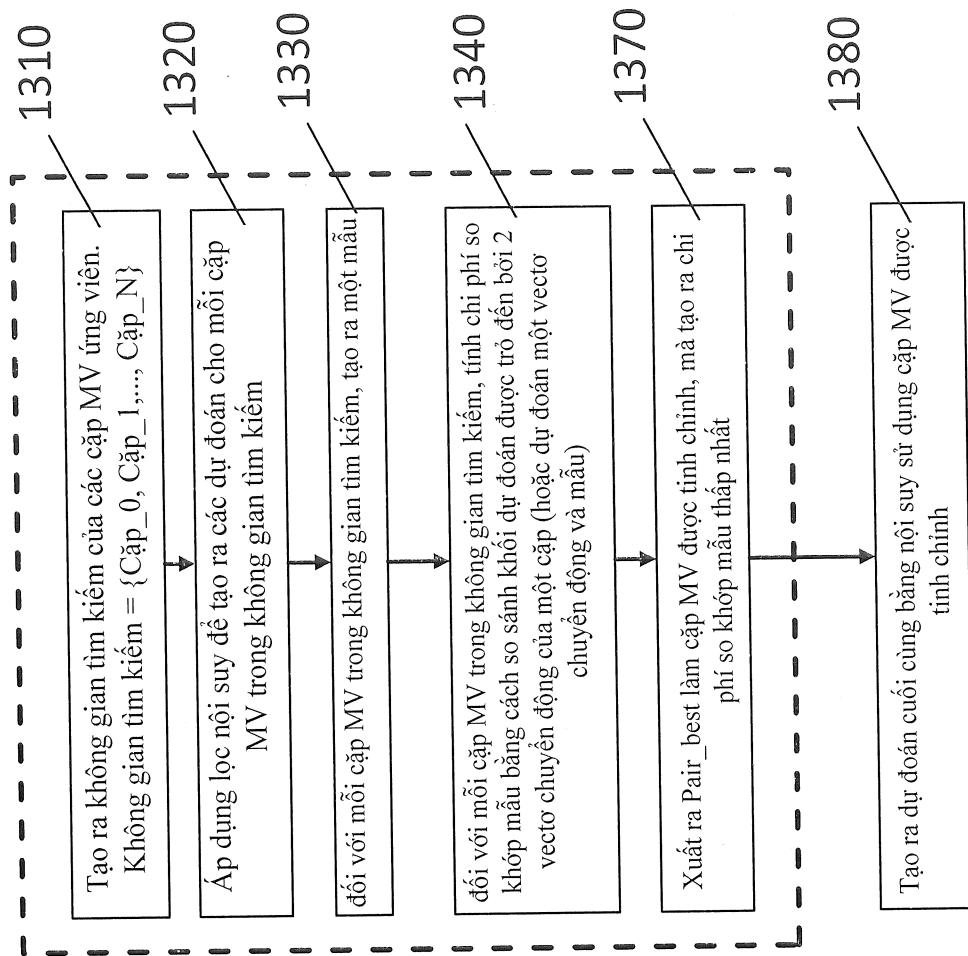


Fig. 12

11/18

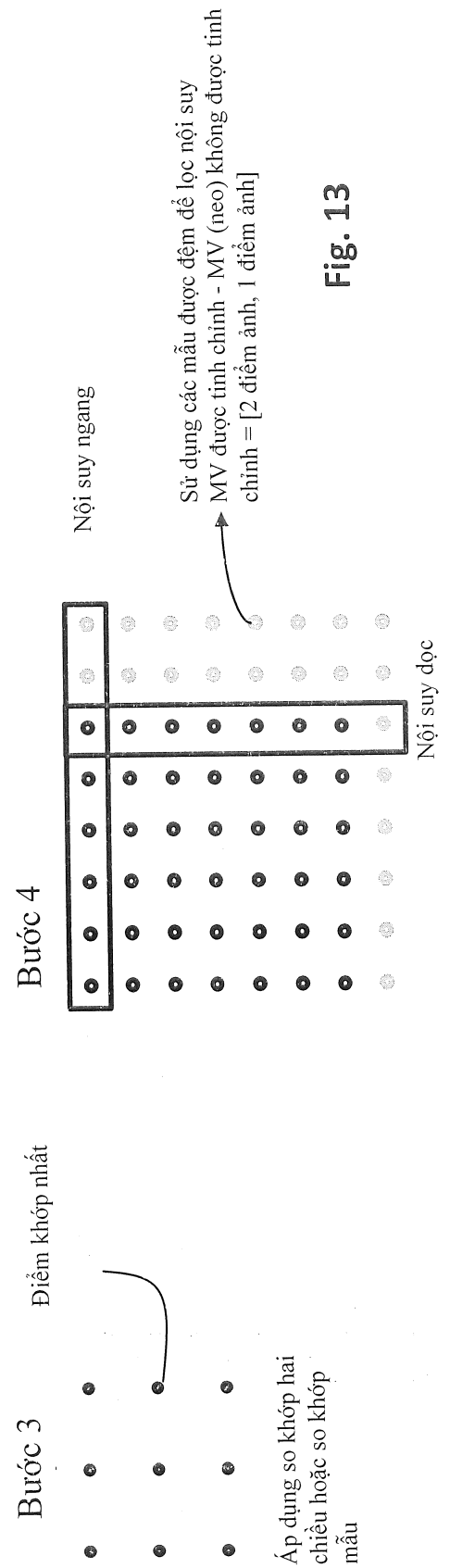
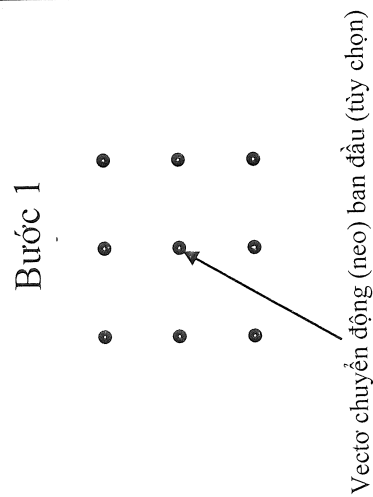
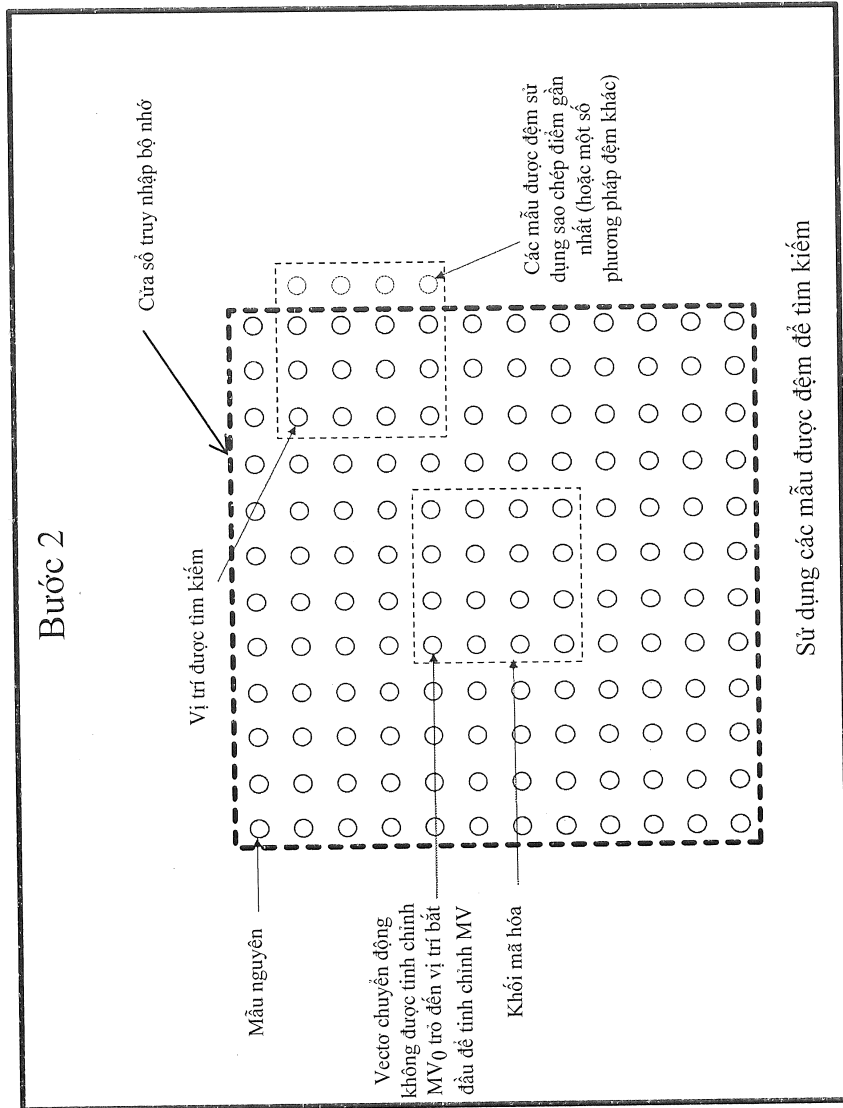


Fig. 13

12/18

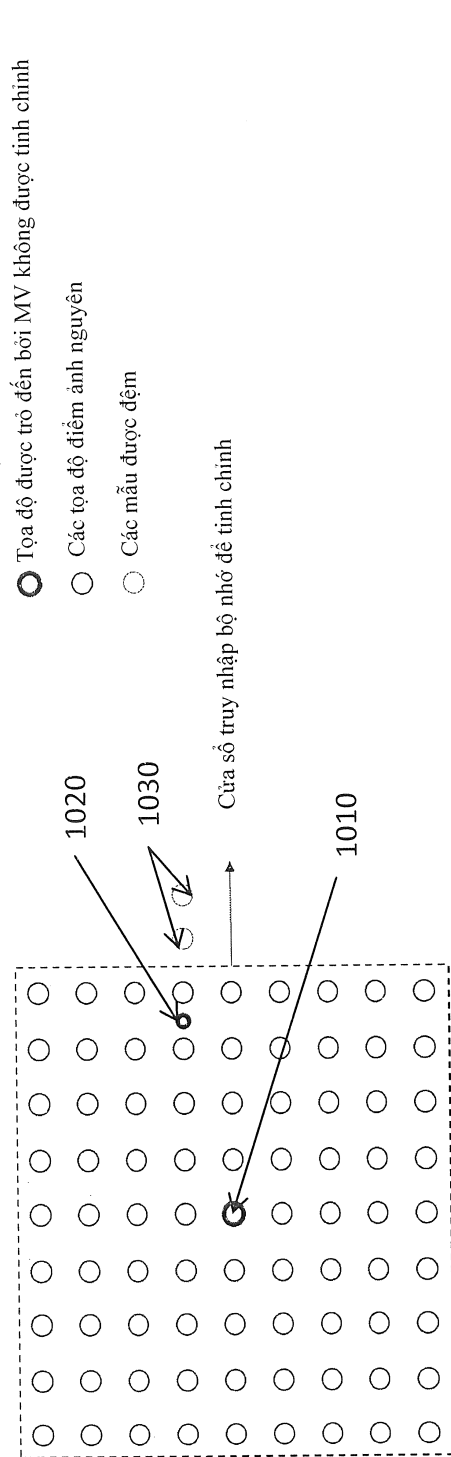


Fig. 14

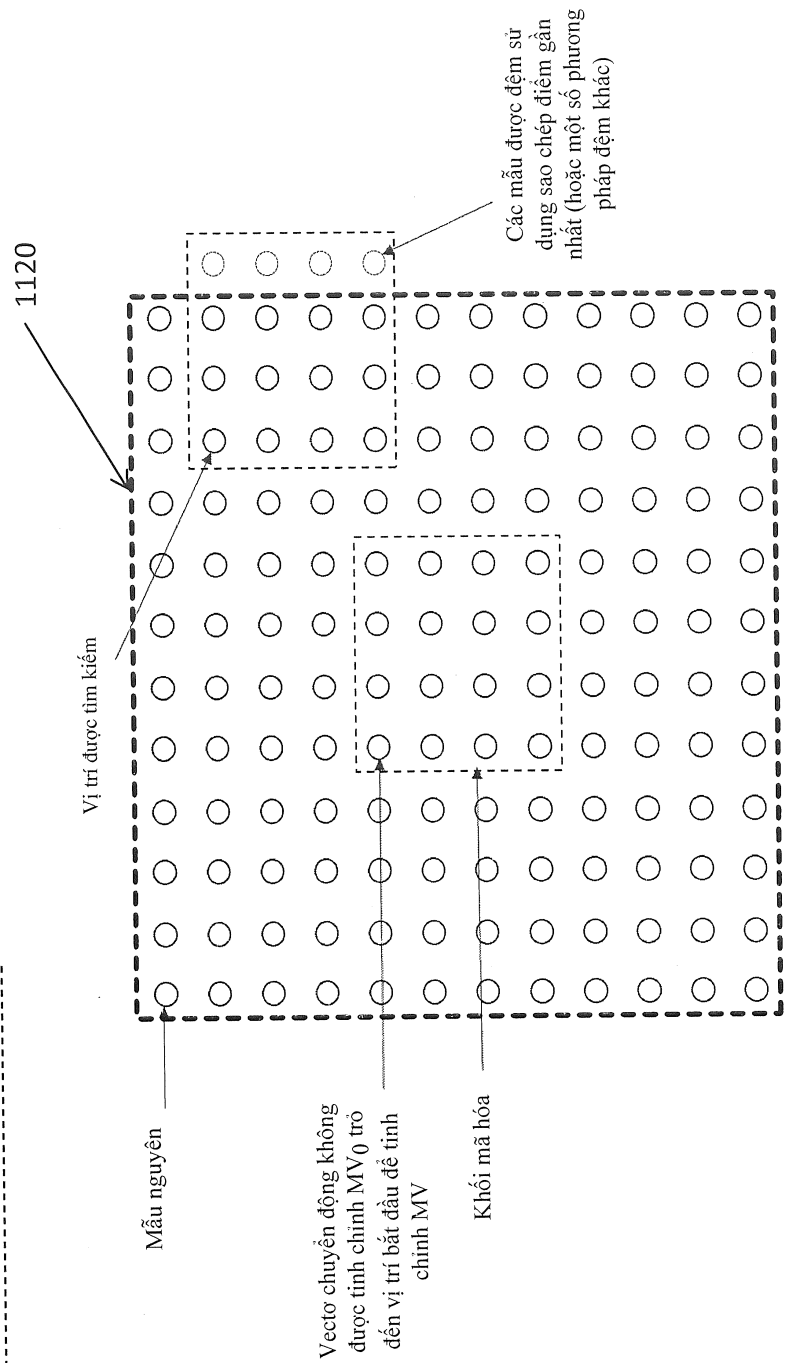


Fig. 15

13/18

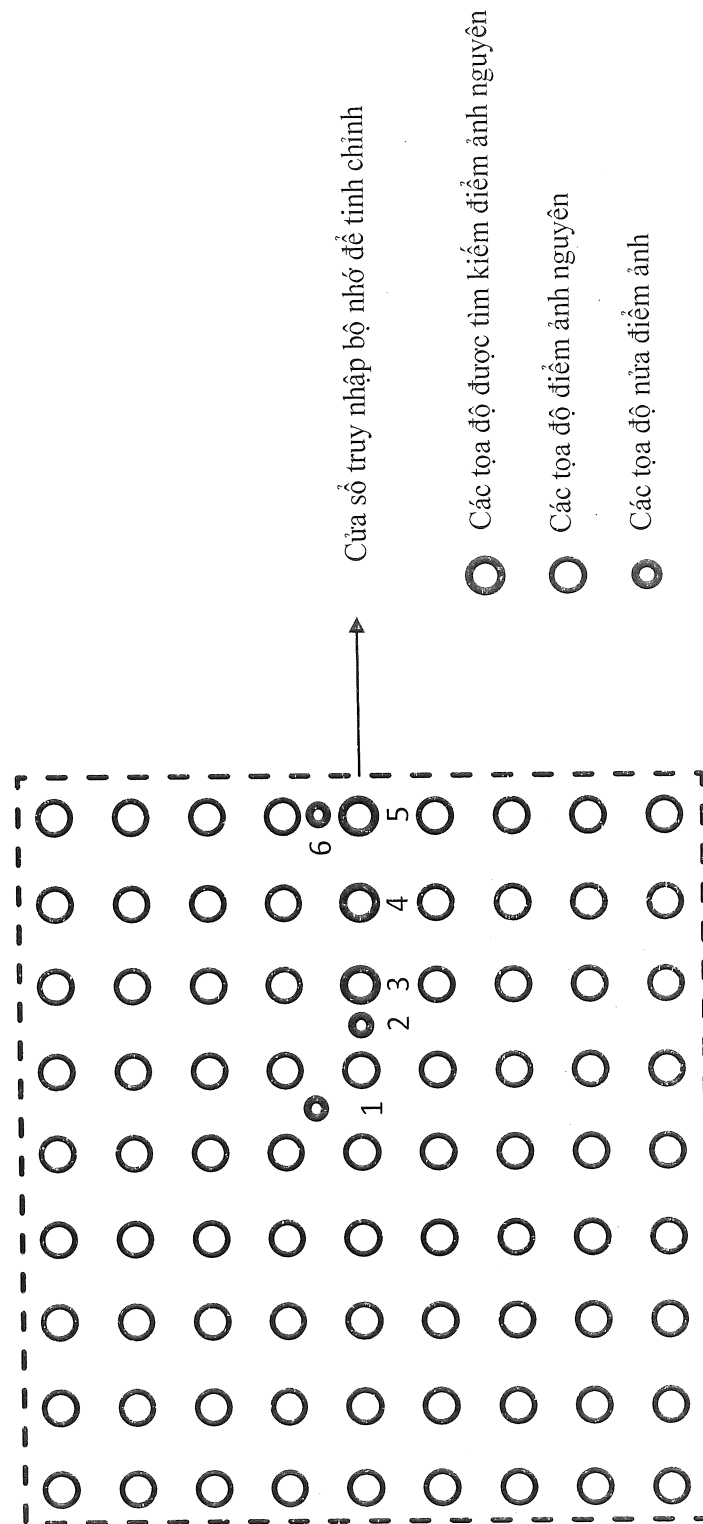


Fig. 16

14/18

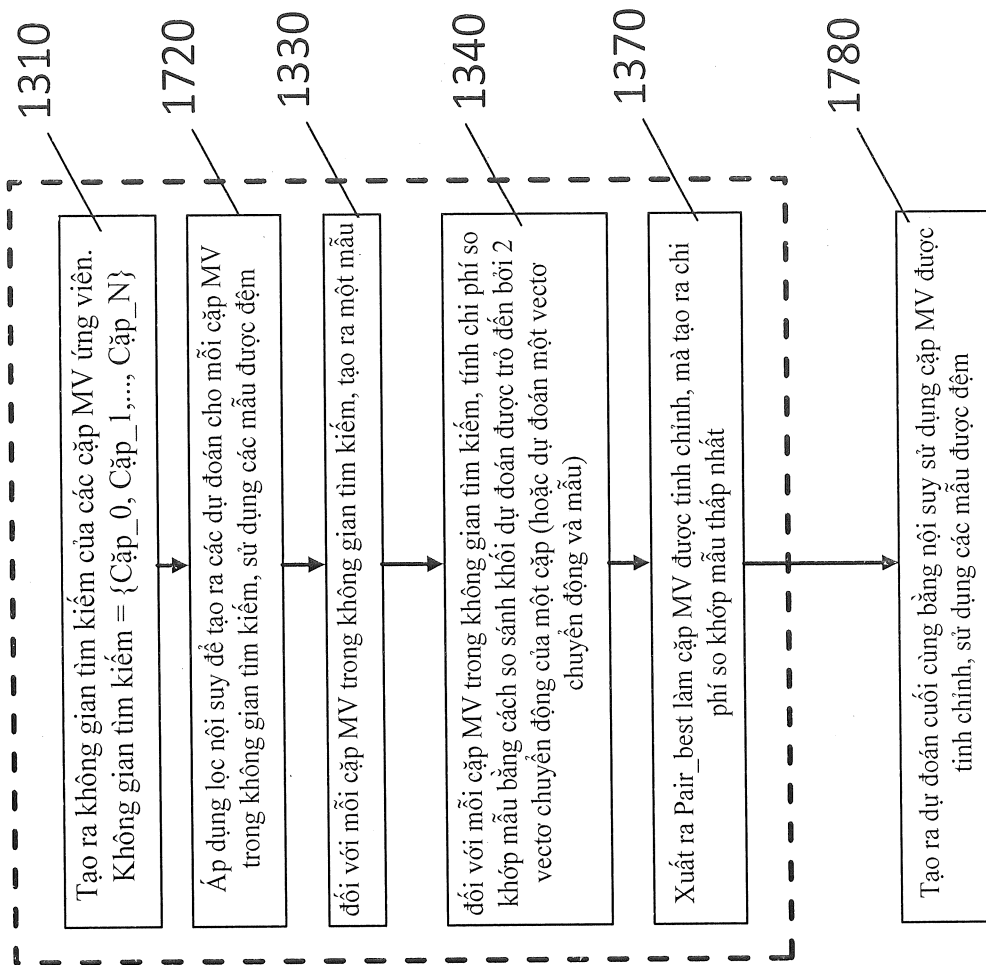


Fig. 17

15/18

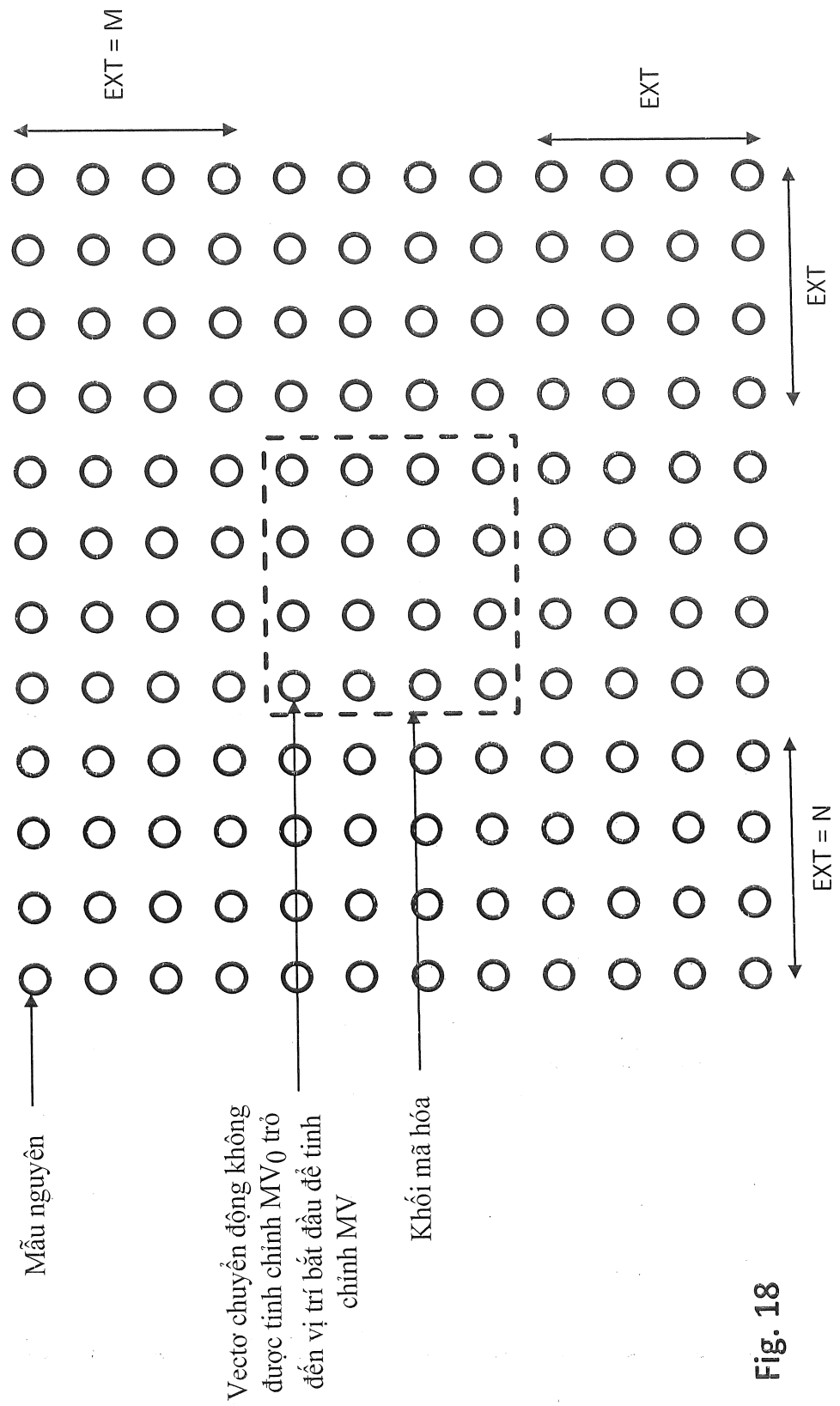


Fig. 18

16/18

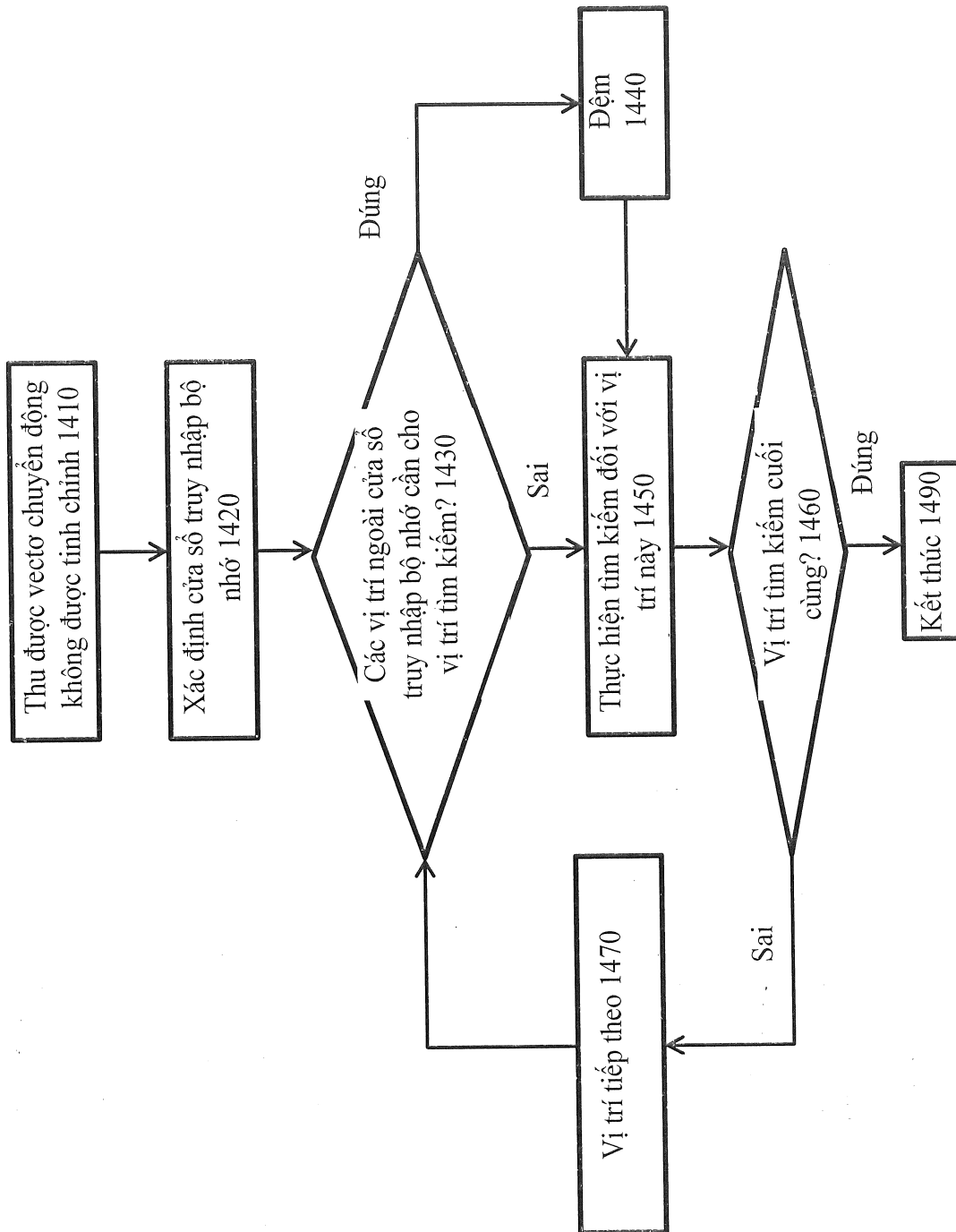


Fig. 19

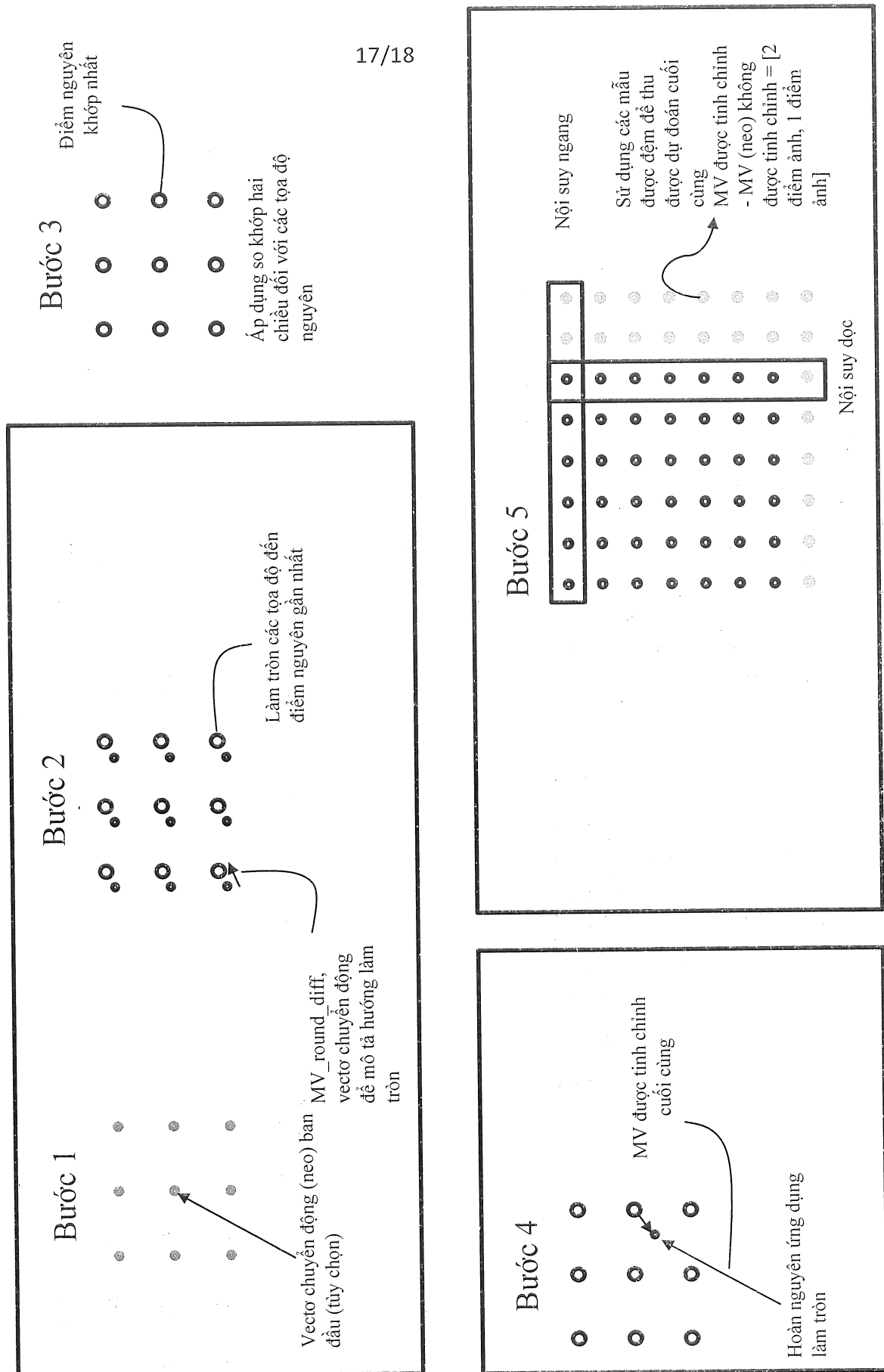


Fig. 20

18/18

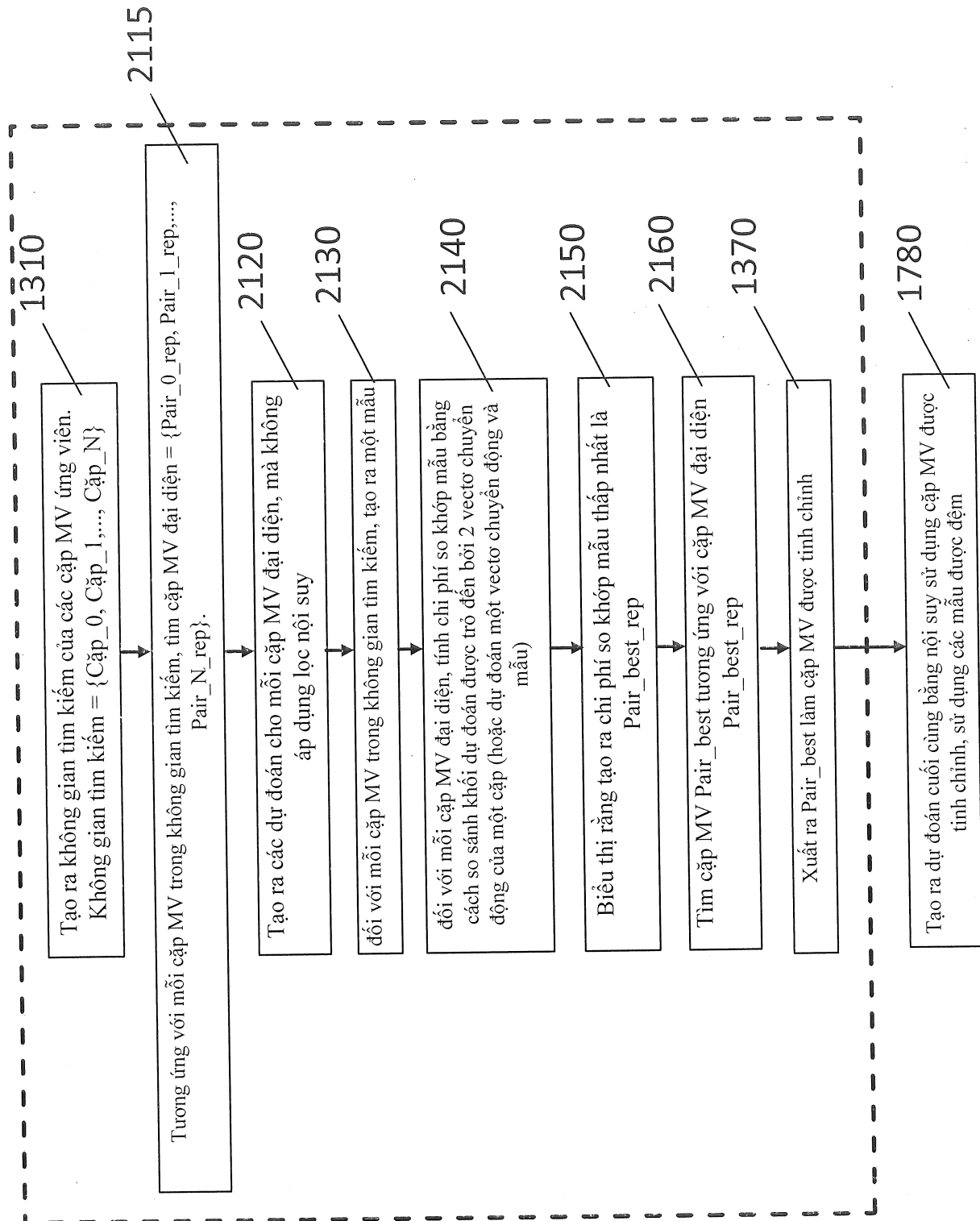


Fig. 21