



(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)



1-0049523

(51)^{2020.01} H04N 19/82

(13) B

(21) 1-2022-01354

(22) 10/09/2020

(86) PCT/JP2020/034381 10/09/2020

(87) WO 2021/049593 18/03/2021

(30) 62/898,845 11/09/2019 US

(45) 25/08/2025 449

(43) 25/07/2022 412A

(73) Panasonic Intellectual Property Corporation of America (US)

2050 W 190th Street Suite 450, Torrance, California 90504 United States of America

(72) LI, Jing Ya (CN); LIM, Chong Soon (SG); SUN, Hai Wei (SG); TEO, Han Boon (SG); KUO, Che Wei (TW); WANG, Chu Tong (CN); ABE, Kiyofumi (JP); NISHI, Takahiro (JP); TOMA, Tadamasa (JP); KATO, Yusuke (JP).

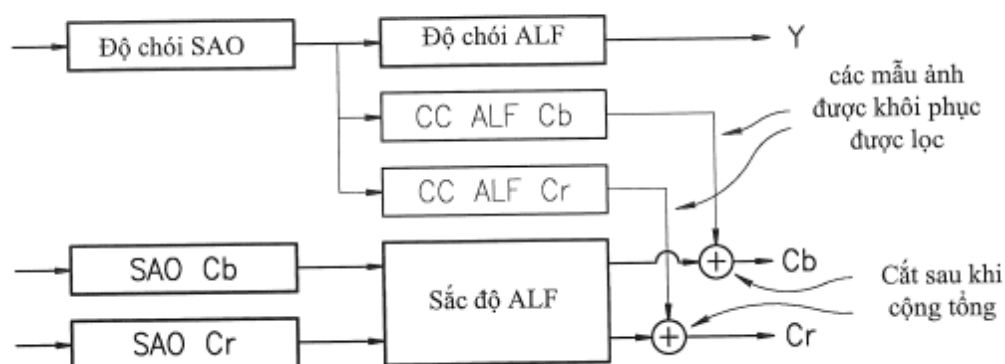
(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) BỘ MÃ HÓA, THIẾT BỊ MÃ HÓA, PHƯƠNG PHÁP MÃ HÓA VÀ PHƯƠNG PHÁP GIẢI MÃ

(21) 1-2022-01354

(57) Sáng chế đề cập đến bộ mã hóa bao gồm mạch và bộ nhớ. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba. Trong xử lý CCALF, để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

[Fig. 98]



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến việc mã hóa video, và cụ thể đề cập đến các hệ thống, các bộ phận và các phương pháp mã hóa và giải mã video trong việc mã hóa và giải mã video, như để thực hiện xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo).

Tình trạng kỹ thuật của sáng chế

Với sự cải tiến về kỹ thuật mã hóa video, từ H.261 và MPEG-1 đến H.264/AVC (Advanced Video Coding-Mã hóa video cải tiến), MPEG-LA, H.265/HEVC (High Efficiency Video Coding-Mã hóa video hiệu quả cao) và H.266/VVC (Versatile Video Codec-mã hóa giải mã video toàn diện), vẫn còn hàng số cần có các cải thiện và tối ưu đối với kỹ thuật mã hóa video để xử lý lượng dữ liệu video số tăng lên trong các ứng dụng khác nhau. Sáng chế đề cập đến các cải tiến, cải thiện và tối ưu hóa tiếp theo trong việc mã hóa video, cụ thể trong xử lý CCALF (cross component adaptive loop filtering - Lọc vòng thích nghi thành phần chéo).

Bản chất kỹ thuật của sáng chế

Theo một khía cạnh của sáng chế, bộ mã hóa được đề xuất mà bao gồm mạch và bộ nhớ được ghép nối tới mạch. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói và cắt giá trị hệ số thứ nhất. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ và cắt giá trị hệ số thứ hai. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt, và mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, mẫu ảnh được khôi phục thứ nhất được bố trí liền kề với mẫu ảnh được khôi phục thứ hai.

Theo khía cạnh khác, mạch, khi hoạt động, thiết lập giá trị hệ số thứ nhất thành 0 (zero) để phản hồi lại giá trị hệ số thứ nhất nhỏ hơn 64.

Theo khía cạnh khác, bộ mã hóa được đề xuất mà bao gồm: bộ phân chia khối, mà khi hoạt động, phân chia ảnh thứ nhất thành nhiều khối; bộ nội dự đoán, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ nhất; bộ dự đoán liên đới, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ hai khác với ảnh thứ nhất; bộ lọc vòng, mà khi hoạt động, lọc các khối được chứa trong ảnh thứ nhất; bộ biến đổi, mà khi hoạt động, biến đổi sai số dự đoán giữa tín hiệu gốc và tín hiệu dự đoán được tạo ra bởi bộ nội dự đoán hoặc bộ dự đoán liên đới, để tạo ra các hệ số biến đổi; bộ lượng tử hóa, mà khi hoạt động, lượng tử hóa các hệ số biến đổi để tạo ra các hệ số được lượng tử hóa; và bộ mã hóa entropy, mà khi

hoạt động, mã hóa biến thiên các hệ số được lượng tử hóa để tạo ra dòng bit được mã hóa bao gồm các hệ số được lượng tử hóa và thông tin điều khiển được mã hóa. Bộ lọc vòng thực hiện các bước sau đây:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

cắt giá trị hệ số thứ nhất;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

cắt giá trị hệ số thứ hai;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt; và

mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, bộ giải mã được đề xuất mà bao gồm mạch và bộ nhớ được ghép nối tới mạch. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói và cắt giá trị hệ số thứ nhất. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ và cắt giá trị hệ số thứ hai. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt, và giải mã mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, thiết bị giải mã được đề xuất mà bao gồm: bộ giải mã, mà khi hoạt động, giải mã dòng bit được mã hóa để xuất ra các hệ số được lượng tử hóa; bộ lượng tử hóa ngược, mà khi hoạt động, lượng tử hóa ngược các hệ số được lượng tử hóa để xuất ra các hệ số biến đổi; bộ biến đổi ngược, mà khi hoạt động, biến đổi ngược các hệ số biến đổi để xuất ra sai số dự đoán; bộ nội dự đoán, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ nhất; bộ dự đoán liên đới, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ hai khác với ảnh thứ nhất; bộ lọc vòng, mà khi hoạt động, lọc các khối được chứa trong ảnh thứ nhất; và bộ xuất, mà khi hoạt động, xuất ra ảnh bao gồm ảnh thứ nhất. Bộ lọc vòng thực hiện các bước sau đây:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

cắt giá trị hệ số thứ nhất;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

cắt giá trị hệ số thứ hai;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt; và

giải mã mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, phương pháp mã hóa được đề xuất, mà bao gồm:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

cắt giá trị hệ số thứ nhất;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

cắt giá trị hệ số thứ hai;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt; và

mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, phương pháp giải mã được đề xuất, mà bao gồm:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

cắt giá trị hệ số thứ nhất;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

cắt giá trị hệ số thứ hai;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất được cắt tới giá trị hệ số thứ hai được cắt; và

giải mã mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác của sáng chế, bộ mã hóa được đề xuất mà bao gồm mạch và bộ nhớ được ghép nối tới mạch. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba. Mạch xác định tham số thứ nhất mà có cùng giá trị đối với thành phần Cb và thành phần Cr của thành phần sắc độ. Mạch xác định, nhờ sử dụng tham số thứ nhất, mô hình của mã hóa entropy từ nhiều mô hình. Mạch thực hiện, nhờ sử dụng mô hình này, việc mã hóa entropy của tham số thứ hai của xử lý CCALF.

Theo khía cạnh khác, bộ giải mã được đề xuất mà bao gồm mạch và bộ nhớ được ghép nối tới mạch. Mạch, khi hoạt động, xác định tham số thứ nhất mà có cùng giá trị đối với thành phần Cb và thành phần Cr của thành phần sắc độ. Mạch xác định, nhờ sử dụng tham số thứ nhất, mô hình của mã hóa entropy từ nhiều mô hình. Mạch thực hiện, nhờ sử dụng mô hình này, việc mã hóa entropy của tham số thứ hai của xử lý CCALF. Mạch tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và giải mã mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba.

Theo khía cạnh khác, bộ mã hóa được đề xuất mà bao gồm mạch và bộ nhớ. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý

CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba. Trong xử lý CCALF, để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

Theo khía cạnh khác, bộ giải mã được đề xuất mà bao gồm mạch và bộ nhớ. Mạch, khi hoạt động, tạo ra giá trị hệ số thứ nhất bằng cách áp dụng xử lý CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới mẫu ảnh được khôi phục thứ nhất của thành phần độ chói. Mạch tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ. Mạch tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba. Trong xử lý CCALF, để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

Trong kỹ thuật mã hóa video, có mong muốn đề xuất các phương pháp mới để cải thiện hiệu quả mã hóa, nâng cao chất lượng ảnh, và làm giảm kích thước mạch. Một vài cách thức thực hiện của các phương án theo sáng chế, bao gồm các thành phần cấu thành của các phương án theo sáng chế được xem xét độc lập hoặc trong các kết hợp khác nhau, có thể hỗ trợ một hoặc nhiều trong số phần sau đây: cải thiện về hiệu quả mã hóa, nâng cao chất lượng ảnh, làm giảm sử dụng tài nguyên xử lý được kết hợp với việc mã hóa/giải mã, làm giảm kích cỡ mạch, cải thiện tốc độ xử lý của việc mã hóa/giải mã, v.v.

Ngoài ra, một vài phương án của các phương án theo sáng chế, bao gồm các thành phần cấu thành của các phương án theo sáng chế được xem xét độc lập hoặc

trong các kết hợp khác nhau, có thể hỗ trợ, trong việc mã hóa và giải mã, lựa chọn thích hợp của một hoặc nhiều thành phần, như bộ lọc, khối, kích cỡ, vector chuyển động, ảnh tham chiếu, khối tham chiếu hoặc hoạt động. Cần lưu ý rằng sáng chế bao gồm bộc lộ liên quan đến các cấu trúc và các phương pháp mà có thể mang lại các hiệu quả khác ngoài các hiệu quả nêu trên. Các ví dụ về các cấu trúc và các phương pháp bao gồm cấu trúc hoặc phương pháp để cải thiện hiệu quả mã hóa trong khi làm giảm việc tăng sử dụng tài nguyên xử lý.

Các hiệu quả và các ưu điểm khác của các phương án được bộc lộ sẽ trở nên rõ ràng từ bản mô tả và các hình vẽ. Các lợi ích và/hoặc các ưu điểm có thể đạt được riêng biệt bởi các phương án khác nhau và các dấu hiệu của bản mô tả và các hình vẽ, không phải tất cả trong số chúng cần được mô tả để thu nhận một hoặc nhiều lợi ích và/hoặc ưu điểm này.

Lưu ý rằng các phương án chung hoặc cụ thể có thể được thực hiện là hệ thống, phương pháp, mạch tích hợp, chương trình máy tính, phương tiện lưu trữ, hoặc bất kỳ kết hợp chọn lọc của chúng.

Mô tả vắn tắt các hình vẽ

FIG.1 là sơ đồ giản lược minh họa một ví dụ về cấu trúc chức năng của hệ thống truyền theo phương án của sáng chế.

FIG.2 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc phân cấp của dữ liệu trong dòng.

FIG.3 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc lát.

FIG.4 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc ô.

FIG.5 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc mã hóa trong mã hóa có thể biến đổi.

FIG.6 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc mã hóa trong mã hóa có thể biến đổi.

FIG.7 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa theo phương án của sáng chế.

FIG.8 là sơ đồ khối chức năng minh họa ví dụ bố trí của bộ mã hóa.

FIG.9 là lưu đồ thể hiện một ví dụ về xử lý mã hóa chung được thực hiện bởi bộ mã hóa.

FIG.10 là sơ đồ khái niệm để minh họa một ví dụ về phân chia khối

FIG.11 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ phân chia theo phương án của sáng chế.

FIG.12 là sơ đồ khái niệm để minh họa các ví dụ về các mẫu phân chia.

FIG.13A là sơ đồ khái niệm để minh họa một ví dụ về cây cú pháp của mẫu phân chia.

FIG.13B là sơ đồ khái niệm để minh họa ví dụ khác về cây cú pháp của mẫu phân chia.

FIG.14 là sơ đồ chỉ báo các hàm cơ sở biến đổi ví dụ đối với các loại biến đổi khác nhau.

FIG.15 là sơ đồ khái niệm để minh họa các biến đổi thay đổi theo không gian (SVT-spatially varying transforms) ví dụ.

FIG.16 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi.

FIG.17 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ biến đổi.

FIG.18 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lượng tử hóa theo phương án của sáng chế.

FIG.19 là lưu đồ minh họa một ví dụ về xử lý lượng tử hóa được thực hiện bởi bộ lượng tử hóa.

FIG.20 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ mã hóa entropy theo phương án của sáng chế.

FIG.21 là sơ đồ khái niệm để minh họa dòng xử lý mã hóa số học nhị phân thích nghi dựa trên ngữ cảnh (CABAC) trong bộ mã hóa entropy.

FIG.22 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lọc vòng theo phương án của sáng chế.

FIG.23A là sơ đồ khái niệm để minh họa một ví dụ về dạng lọc được sử dụng trong bộ lọc vòng thích nghi (ALF-adaptive loop filter).

FIG.23B là sơ đồ khái niệm để minh họa ví dụ khác về dạng lọc được sử dụng trong ALF.

FIG.23C là sơ đồ khái niệm để minh họa ví dụ khác về dạng lọc được sử dụng trong ALF.

FIG.23D là sơ đồ khái niệm để minh họa dòng ví dụ của ALF thành phần chéo (CC-ALF).

FIG.23E là sơ đồ khái niệm để minh họa ví dụ về dạng lọc được sử dụng trong CC-ALF.

FIG.23F là sơ đồ khái niệm để minh họa dòng ví dụ của CCALF sắc độ liên kết (JC-CCALF).

FIG.23G là Bảng minh họa các ứng viên chỉ số trọng số ví dụ mà có thể được sử dụng trong JC-CCALF.

FIG.24 là sơ đồ khối mà chỉ báo một ví dụ về cấu trúc cụ thể của bộ lọc vòng mà thực hiện chức năng như bộ lọc giải khối (DBF-deblocking filter).

FIG.25 là sơ đồ khái niệm để minh họa ví dụ về bộ lọc giải khối mà có đặc tính lọc đối xứng đối với biên khối.

FIG.26 là sơ đồ khái niệm để minh họa biên khối mà trên đó xử lý lọc giải khối được thực hiện.

FIG.27 là sơ đồ khái niệm để minh họa các ví dụ về các giá trị cường độ biên (Bs-Boundary strength).

FIG.28 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa.

FIG.29 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa.

FIG.30 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa.

FIG.31 là sơ đồ khái niệm để minh họa 67 chế độ nội dự đoán được sử dụng trong việc nội dự đoán theo phương án của sáng chế.

FIG.32 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ nội dự đoán.

FIG.33 là sơ đồ khái niệm để minh họa các ví dụ về các ảnh tham chiếu.

FIG.34 là sơ đồ khái niệm để minh họa các ví dụ về các danh sách ảnh tham chiếu.

FIG.35 là lưu đồ minh họa dòng xử lý cơ bản ví dụ của việc dự đoán liên đới.

FIG.36 là lưu đồ minh họa một ví dụ về xử lý thu nhận của các vector chuyển động.

FIG.37 là lưu đồ minh họa một ví dụ khác về xử lý thu nhận của các vector chuyển động.

FIG.38A là sơ đồ khái niệm để minh họa các đặc trưng ví dụ của các chế độ đối với việc thu nhận MV.

FIG.38B là sơ đồ khái niệm để minh họa các đặc trưng ví dụ của các chế độ đối với việc thu nhận MV.

FIG.39 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới trong chế độ liên đới thường.

FIG.40 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới trong chế độ hợp nhất thường.

FIG.41 là sơ đồ khái niệm để minh họa một ví dụ về xử lý thu nhận vector chuyển động trong chế độ hợp nhất.

FIG.42 là sơ đồ khái niệm để minh họa một ví dụ về xử lý thu nhận MV đối với ảnh hiện tại bởi chế độ hợp nhất HMVP.

FIG.43 là lưu đồ minh họa một ví dụ về xử lý chuyển đổi tăng tốc độ khung (FRUC-frame rate up conversion).

FIG.44 là sơ đồ khái niệm để minh họa một ví dụ về so khớp khuôn mẫu (so khớp hai chiều) giữa hai khối dọc theo đường chuyển động.

FIG.45 là sơ đồ khái niệm để minh họa một ví dụ về so khớp mẫu (so khớp khuôn mẫu) giữa mẫu trong ảnh hiện tại và khối trong ảnh tham chiếu.

FIG.46A là sơ đồ khái niệm để minh họa một ví dụ về thu nhận vector chuyển động của mỗi khối con dựa trên các vector chuyển động của các khối lân cận.

FIG.46B là sơ đồ khái niệm để minh họa một ví dụ về thu nhận vector chuyển động của mỗi khối con trong chế độ affin trong đó ba điểm điều khiển được sử dụng.

FIG.47A là sơ đồ khái niệm để minh họa việc thu nhận MV ví dụ tại các điểm điều khiển trong chế độ affin.

FIG.47B là sơ đồ khái niệm để minh họa việc thu nhận MV ví dụ tại các điểm điều khiển trong chế độ affin.

FIG.47C là sơ đồ khái niệm để minh họa việc thu nhận MV ví dụ tại các điểm điều khiển trong chế độ affin.

FIG.48A là sơ đồ khái niệm để minh họa chế độ affin trong đó hai điểm điều khiển được sử dụng.

FIG.48B là sơ đồ khái niệm để minh họa chế độ affin trong đó ba điểm điều khiển được sử dụng.

FIG.49A là sơ đồ khái niệm để minh họa một ví dụ về phương pháp thu nhận MV tại các điểm điều khiển khi số lượng điểm điều khiển đối với khối được mã hóa và số lượng điểm điều khiển đối với khối hiện tại là khác nhau.

FIG.49B là sơ đồ khái niệm để minh họa ví dụ khác về phương pháp thu nhận MV tại các điểm điều khiển khi số lượng điểm điều khiển đối với khối được mã hóa và số lượng điểm điều khiển đối với khối hiện tại là khác nhau.

FIG.50 là lưu đồ minh họa một ví dụ về xử lý trong chế độ hợp nhất affin.

FIG.51 là lưu đồ minh họa một ví dụ về xử lý trong chế độ liên đới affin.

FIG.52A là sơ đồ khái niệm để minh họa việc tạo ra hai ảnh dự đoán tam giác.

FIG.52B là sơ đồ khái niệm để minh họa các ví dụ về phần thứ nhất của phân vùng thứ nhất mà chồng lấn với phân vùng thứ hai, và các tập hợp của các mẫu thứ nhất và thứ hai mà có thể được lấy trọng số như là một phần của xử lý hiệu chỉnh.

FIG.52C là sơ đồ khái niệm để minh họa phần thứ nhất của phân vùng thứ nhất, mà là phần của phân vùng thứ nhất mà chồng lấn với phần của phân vùng liền kề.

FIG.53 là lưu đồ minh họa một ví dụ về xử lý trong chế độ tam giác.

FIG.54 là sơ đồ khái niệm để minh họa một ví dụ về chế độ dự đoán vector chuyển động theo thời gian cải tiến (ATMVP-Advanced Temporal Motion Vector Prediction) trong đó MV được thu nhận trong các đơn vị của khối con.

FIG.55 là lưu đồ minh họa quan hệ giữa chế độ hợp nhất và làm mới vector chuyển động linh động (DMVR-dynamic motion vector refreshing).

FIG.56 là sơ đồ khái niệm để minh họa một ví dụ của DMVR.

FIG.57 là sơ đồ khái niệm để minh họa ví dụ khác về DMVR để xác định MV.

FIG.58A là sơ đồ khái niệm để minh họa một ví dụ về ước lượng chuyển động trong DMVR.

FIG.58B là lưu đồ minh hoạt một ví dụ về xử lý ước lượng chuyển động trong DMVR.

FIG.59 là lưu đồ minh họa một ví dụ về xử lý tạo ra ảnh dự đoán.

FIG.60 là lưu đồ minh họa ví dụ khác về xử lý tạo ra ảnh dự đoán.

FIG.61 là lưu đồ minh hoạt một ví dụ về xử lý hiệu chỉnh của ảnh dự đoán bằng cách bù chuyển động khối chồng lấn (OBMC-overlapped block motion compensation).

FIG.62 là sơ đồ khái niệm để minh họa một ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi OBMC.

FIG.63 là sơ đồ khái niệm để minh họa mô hình giả định chuyển động tuyến tính đồng đều.

FIG.64 là lưu đồ minh họa một ví dụ về xử lý dự đoán liên đới theo BIO.

FIG.65 là sơ đồ khối chức năng minh họa một ví dụ về cấu trúc chức năng của bộ dự đoán liên đới mà có thể thực hiện việc dự đoán liên đới theo BIO.

FIG.66A là sơ đồ khái niệm để minh họa một ví dụ về xử lý của phương pháp tạo ảnh dự đoán nhờ sử dụng xử lý hiệu chỉnh độ chói được thực hiện bởi LIC.

FIG.66B là lưu đồ minh họa một ví dụ về xử lý của phương pháp tạo ảnh dự đoán nhờ sử dụng LIC.

FIG.67 là sơ đồ khối minh họa cấu trúc chức năng của bộ giải mã theo phương án của sáng chế.

FIG.68 là sơ đồ khối chức năng minh họa ví dụ bố trí của bộ giải mã.

FIG.69 là lưu đồ minh họa một ví dụ về xử lý giải mã chung được thực hiện bởi bộ giải mã.

FIG.70 là sơ đồ khái niệm để minh họa quan hệ giữa bộ xác định phân chia và các thành phần cấu thành khác.

FIG.71 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng bộ giải mã entropy.

FIG.72 là sơ đồ khái niệm để minh họa dòng xử lý CABAC ví dụ trong bộ giải mã entropy.

FIG.73 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lượng tử hóa ngược.

FIG.74 là lưu đồ minh họa một ví dụ về xử lý lượng tử hóa ngược được thực hiện bởi bộ lượng tử hóa ngược.

FIG.75 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi ngược.

FIG.76 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ biến đổi ngược.

FIG.77 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lọc vòng.

FIG.78 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã.

FIG.79 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã.

FIG.80A là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã.

FIG.80B là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã.

FIG.80C là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã.

FIG.81 là sơ đồ minh họa một ví dụ của xử lý được thực hiện bởi bộ nội dự đoán của bộ giải mã.

FIG.82 là lưu đồ minh họa một ví dụ về xử lý thu nhận MV trong bộ giải mã.

FIG.83 là lưu đồ minh họa ví dụ khác về xử lý thu nhận MV trong bộ giải mã.

FIG.84 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ liên đới thường trong bộ giải mã.

FIG.85 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ hợp nhất thường trong bộ giải mã.

FIG.86 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ FRUC trong bộ giải mã.

FIG.87 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ hợp nhất afin trong bộ giải mã.

FIG.88 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ liên đới afin trong bộ giải mã.

FIG.89 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ tam giác trong bộ giải mã.

FIG.90 là lưu đồ minh họa ví dụ về xử lý ước lượng chuyển động bởi DMVR trong bộ giải mã.

FIG.91 là lưu đồ minh họa ví dụ về xử lý ước lượng chuyển động bởi DMVR trong bộ giải mã.

FIG.92 là lưu đồ minh họa một ví dụ về xử lý tạo ảnh dự đoán trong bộ giải mã.

FIG.93 là lưu đồ minh họa một ví dụ khác về xử lý tạo ảnh dự đoán trong bộ giải mã.

FIG.94 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi OBMC trong bộ giải mã.

FIG.95 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi BIO trong bộ giải mã.

FIG.96 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi LIC trong bộ giải mã.

FIG.97 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF (cross component adaptive loop filtering-lọc vòng thích nghi thành phần chéo) theo khía cạnh thứ nhất.

FIG.98 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế.

FIG.99 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế.

FIG.100 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế.

FIG.101 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế.

FIG.102 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF theo khía cạnh thứ hai.

FIG.103A minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.103B minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.103C minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.103D minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.103E minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.103F minh họa các vị trí mẫu của các tham số cắt được phân tích từ, ví dụ, VPS, APS, SPS, PPS, thông tin tiêu đề lát, CTU, hoặc TU của dòng bit.

FIG.104 (i)-(iii) minh họa các ví dụ của các tham số cắt.

FIG.106 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng hệ số lọc theo khía cạnh thứ ba.

FIG.106A là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106B là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106C là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106D là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106E là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106F là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106G là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.106H là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107A là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107B là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107C là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107D là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107E là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107F là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107G là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.107H là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108A là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108B là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108C là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108D là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108E là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108F là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108G là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.108H là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.109A là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.109B là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.109C là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.109D là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.110A là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.110B là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.110C là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.110D là sơ đồ khái niệm của các ví dụ mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.111 là sơ đồ khái niệm của các ví dụ khác mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.112 là sơ đồ khái niệm của các ví dụ khác mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF.

FIG.113 là sơ đồ khối minh họa cấu trúc chức năng của xử lý CCALF được thực hiện bởi bộ mã hóa và bộ giải mã theo phương án của sáng chế.

FIG.114 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng bộ lọc được lựa chọn từ các bộ lọc theo khía cạnh thứ tư.

FIG.115 minh họa ví dụ về dòng xử lý để lựa chọn bộ lọc.

FIG.116-1A minh họa ví dụ của các bộ lọc.

FIG.116-1B minh họa ví dụ của các bộ lọc.

FIG.116-1C minh họa ví dụ của các bộ lọc.

FIG.116-1D minh họa ví dụ của các bộ lọc.

FIG.116-1E minh họa ví dụ của các bộ lọc.

FIG.116-1F minh họa ví dụ của các bộ lọc.

FIG.116-1G minh họa ví dụ của các bộ lọc.

FIG.116-1H minh họa ví dụ của các bộ lọc.

FIG.116-1I minh họa ví dụ của các bộ lọc.

FIG.117-2A minh họa ví dụ của các bộ lọc.

FIG.117-2B minh họa ví dụ của các bộ lọc.

FIG.117-2C minh họa ví dụ của các bộ lọc.

FIG.117-2D minh họa ví dụ của các bộ lọc.

FIG.117-2E minh họa ví dụ của các bộ lọc.

FIG.117-2F minh họa ví dụ của các bộ lọc.

FIG.117-2G minh họa ví dụ của các bộ lọc.

FIG.117-2H minh họa ví dụ của các bộ lọc.

FIG.117-2I minh họa ví dụ của các bộ lọc.

FIG.118 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ năm.

FIG.119 (i)-(iv) minh họa các ví dụ về số lượng hệ số (NumCoeff) cần được phân tích từ dòng bit.

FIG.120 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ sáu.

FIG.121 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu sắc độ hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận.

FIG.122 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu sắc độ hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận.

FIG.123 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu sắc độ hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận.

FIG.124 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận, trong đó các vị trí của các mẫu lân cận được xác định thích nghi theo loại sắc độ.

FIG.125 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận, trong đó các vị trí của các mẫu lân cận được xác định thích nghi theo loại sắc độ.

FIG.126 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói bằng cách áp dụng việc dịch bit tới giá trị đầu ra của việc tính toán trọng số.

FIG.127 là sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói bằng cách áp dụng việc dịch bit tới giá trị đầu ra của việc tính toán trọng số.

FIG.128 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ bảy.

FIG.129A minh họa các vị trí mẫu của một hoặc nhiều tham số cần được phân tích từ dòng bit, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai.

FIG.129B minh họa các vị trí mẫu của một hoặc nhiều tham số cần được phân tích từ dòng bit, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai.

FIG.129C minh họa các vị trí mẫu của một hoặc nhiều tham số cần được phân tích từ dòng bit, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai.

FIG.129D minh họa các vị trí mẫu của một hoặc nhiều tham số cần được phân tích từ dòng bit, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai.

FIG.129E minh họa các vị trí mẫu của một hoặc nhiều tham số cần được phân tích từ dòng bit, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai.

FIG.130A thể hiện các xử lý mẫu để truy hồi một hoặc nhiều tham số, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai tham số.

FIG.130B thể hiện các xử lý mẫu để truy hồi một hoặc nhiều tham số, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai tham số.

FIG.130C thể hiện các xử lý mẫu để truy hồi một hoặc nhiều tham số, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai tham số.

FIG.130D thể hiện các xử lý mẫu để truy hồi một hoặc nhiều tham số, mà có thể bao gồm tham số thứ nhất, tham số thứ hai, hoặc cả hai tham số.

FIG.131A thể hiện các giá trị mẫu của tham số thứ hai.

FIG.131B thể hiện các giá trị mẫu của tham số thứ hai.

FIG.131C thể hiện các giá trị mẫu của tham số thứ hai.

FIG.132 thể hiện ví dụ về việc phân tích tham số thứ hai nhờ sử dụng mã hóa số học.

FIG.133 là sơ đồ khái niệm về biên thể của phương án được áp dụng tới các phân vùng chữ nhật và các phân vùng không phải chữ nhật như các phân vùng tam giác.

FIG.134 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ tám.

FIG.135 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ tám.

FIG.136 thể hiện các vị trí ví dụ của các loại mẫu sắc độ 0 đến 5.

FIG.137A là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.137B là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.137C là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.137D là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.138 là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.139 là sơ đồ khái niệm minh họa đệm đối xứng mẫu.

FIG.140A là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.140B là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.140C là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.140D là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.141 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.142 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.143 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu.

FIG.144A là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.144B là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.144C là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.144D là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.145 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.146 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.147 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.148A là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.148B là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.148C là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.148D là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.149 là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.150 là sơ đồ khái niệm minh họa đệm đối xứng mẫu khác.

FIG.151A là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.151B là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.151C là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.152 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.153 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.154 là sơ đồ khái niệm minh họa đệm bất đối xứng mẫu khác.

FIG.155A minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155B minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155C minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155D minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155E minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155F minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155G minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155H minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155I minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155J minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155K minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.155L minh họa các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

FIG.156 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo ví dụ trong đó đệm đối xứng được sử dụng trên các vị trí biên ảo đối với ALF và đệm đối xứng hoặc bất đối xứng được sử dụng trên các vị trí biên ảo đối với CC-ALF.

FIG.157 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo ví dụ khác trong đó đệm đối xứng được sử dụng trên các vị trí biên ảo đối với ALF và đệm một phía được sử dụng trên các vị trí biên ảo đối với CC-ALF.

FIG.158A là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158B là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158C là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158D là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158E là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158F là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158G là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.158H là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.159A là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc và ngang.

FIG.159B là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc và ngang.

FIG.159C là sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc và ngang.

FIG.160 là lưu đồ của dòng xử lý mẫu để giải mã ảnh mà áp dụng xử lý CCALF sử dụng tham số theo khía cạnh thứ chín.

FIG.161 là sơ đồ khái niệm về ví dụ của bộ lọc cần được áp dụng trong xử lý CCALF.

FIG.162 mô tả các công thức mẫu của xử lý lọc.

FIG.163 là sơ đồ khái niệm về ví dụ của cú pháp liên quan đến CCALF.

FIG.164 là sơ đồ khái niệm về ví dụ của báo hiệu giá trị hệ số lọc sử dụng mã Golomb số mũ với bậc cố định k (được ký hiệu là EGk).

FIG.165A là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.165B là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.165C là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.165D là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.166A là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.166B là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.167A là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.167B là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.168A là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.168B là sơ đồ khái niệm về các ví dụ của EGk được áp dụng tới các hệ số lọc.

FIG.169 là sơ đồ khái niệm về ví dụ của cú pháp của các tham số được sử dụng trong xử lý ALF.

FIG.170 là sơ đồ khái niệm về ví dụ của cú pháp của các tham số được sử dụng trong xử lý CCALF.

FIG.171 là lưu đồ về ví dụ của dòng xử lý 10000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tập hợp của các hệ số.

FIG.172A là sơ đồ khái niệm của các ví dụ về dạng của các tập hợp của các hệ số lọc.

FIG.172B là sơ đồ khái niệm của các ví dụ về dạng của các tập hợp của các hệ số lọc.

FIG.172C là sơ đồ khái niệm của các ví dụ về dạng của các tập hợp của các hệ số lọc.

FIG.172D là sơ đồ khái niệm của các ví dụ về dạng của các tập hợp của các hệ số lọc.

FIG.173A là sơ đồ khái niệm của các ví dụ về các vị trí của các mẫu được khôi phục của các thành phần thứ nhất và thành phần thứ hai.

FIG.173B là sơ đồ khái niệm của các ví dụ về các vị trí của các mẫu được khôi phục của các thành phần thứ nhất và thành phần thứ hai.

FIG.173C là sơ đồ khái niệm của các ví dụ về các vị trí của các mẫu được khôi phục của các thành phần thứ nhất và thành phần thứ hai.

FIG.173D là sơ đồ khái niệm của các ví dụ về các vị trí của các mẫu được khôi phục của các thành phần thứ nhất và thành phần thứ hai.

FIG.174 là sơ đồ minh họa cấu trúc chung ví dụ của hệ thống cung cấp nội dung để thực hiện dịch vụ phân phối nội dung.

FIG.175 là sơ đồ khái niệm để minh họa ví dụ về màn ảnh hiển thị của trang mạng.

FIG.176 là sơ đồ khái niệm để minh họa ví dụ về màn ảnh hiển thị của trang mạng.

FIG.177 là sơ đồ khối minh họa một ví dụ của điện thoại thông minh.

FIG.178 là sơ đồ khối minh họa ví dụ về cấu trúc chức năng của điện thoại thông minh.

Mô tả chi tiết sáng chế

Trên các hình vẽ, các số chỉ dẫn đồng nhất ký hiệu các thành phần giống nhau, trừ khi ngữ cảnh chỉ báo khác. Các kích cỡ và các vị trí tương đối của các thành phần trong các hình vẽ không cần thiết được vẽ theo tỷ lệ.

Sau đây, (các) phương án sẽ được mô tả có viện dẫn tới các hình vẽ. Lưu ý rằng mỗi phương án được mô tả dưới đây thể hiện ví dụ chung hoặc cụ thể. Các giá trị số, các dạng, các dữ liệu, các thành phần, cách sắp xếp và kết nối của các thành phần, các bước, quan hệ và thứ tự của các bước, v.v, được chỉ báo trong (các) phương án sau đây chỉ là các ví dụ, và không nhằm mục đích làm giới hạn phạm vi của yêu cầu bảo hộ.

Các phương án của bộ mã hóa và bộ giải mã sẽ được mô tả dưới đây. Các phương án là các ví dụ của bộ mã hóa và bộ giải mã mà các xử lý và/hoặc các cấu trúc được thể hiện trong phần mô tả của các khía cạnh của sáng chế có thể được áp dụng tới. Các xử lý và/hoặc các cấu trúc có thể cũng được thực hiện trong bộ mã hóa và bộ giải mã khác với theo các phương án này. Ví dụ, liên quan đến các xử lý và/hoặc các cấu trúc như được áp dụng tới các phương án của sáng chế, bất kỳ trong số phần sau đây có thể được thực hiện:

(1) Bất kỳ các thành phần của bộ mã hóa hoặc bộ giải mã theo các phương án được thể hiện trong phần mô tả của các khía cạnh của sáng chế có thể được thay thế hoặc được kết hợp với thành phần khác được thể hiện bất kỳ trong phần mô tả của các khía cạnh của sáng chế.

(2) Trong bộ mã hóa hoặc bộ giải mã theo các phương án, các thay đổi tùy chọn có thể được thực hiện tới các chức năng hoặc các xử lý được thực hiện bởi một hoặc nhiều thành phần của bộ mã hóa hoặc bộ giải mã, như bổ sung, thay thế, loại bỏ, v.v, của các chức năng hoặc các xử lý. Ví dụ, bất kỳ chức năng hoặc xử lý có thể được thay thế hoặc được kết hợp với chức năng hoặc xử lý khác được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế.

(3) Trong các phương pháp được thực hiện bởi bộ mã hóa hoặc bộ giải mã theo các phương án, các thay đổi tùy chọn có thể được thực hiện như bổ sung, thay thế, và loại bỏ của một hoặc nhiều xử lý được nằm trong phương pháp này. Ví dụ, bất kỳ xử lý trong phương pháp này có thể được thay thế hoặc được kết hợp với xử lý khác được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế.

(4) Một hoặc nhiều thành phần được chứa trong bộ mã hóa hoặc bộ giải mã theo các phương án có thể được kết hợp với thành phần được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế, có thể được kết hợp với thành phần bao gồm một hoặc nhiều chức năng được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế, và có thể được kết hợp với thành phần mà thực hiện một hoặc nhiều xử lý được thực hiện bởi thành phần được thể hiện trong phần mô tả của các khía cạnh của sáng chế.

(5) Thành phần bao gồm một hoặc nhiều chức năng của bộ mã hóa hoặc bộ giải mã theo các phương án, hoặc thành phần mà thực hiện một hoặc nhiều xử lý của bộ mã hóa hoặc bộ giải mã theo các phương án, có thể được kết hợp hoặc được thay thế với thành phần được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế, với thành phần bao gồm một hoặc nhiều chức năng được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế, hoặc với thành phần mà thực hiện một hoặc nhiều xử lý được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế.

(6) Trong các phương pháp được thực hiện bởi bộ mã hóa hoặc bộ giải mã theo các phương án, bất kỳ xử lý được chứa trong phương pháp có thể được thay thế hoặc được kết hợp với xử lý được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế hoặc với bất kỳ xử lý tương ứng hoặc tương đương.

(7) Một hoặc nhiều xử lý được chứa trong các phương pháp được thực hiện bởi bộ mã hóa hoặc bộ giải mã theo các phương án có thể được kết hợp với xử lý được thể hiện trong bất kỳ phần mô tả của các khía cạnh của sáng chế.

(8) Cách thức thực hiện của các xử lý và/hoặc các cấu trúc được thể hiện trong phần mô tả của các khía cạnh của sáng chế không bị giới hạn ở bộ mã hóa hoặc bộ giải mã theo các phương án. Ví dụ, các xử lý và/hoặc các cấu trúc có thể được thực hiện trong thiết bị được sử dụng cho mục đích khác với bộ mã hóa ảnh động hoặc bộ giải mã ảnh động được bộc lộ trong các phương án.

(Các định nghĩa của các thuật ngữ)

Các thuật ngữ tương ứng có thể được định nghĩa như được mô tả dưới đây như là các ví dụ.

Ảnh là đơn vị dữ liệu được cấu hình với tập hợp của các điểm ảnh, là hình ảnh, hoặc bao gồm các khối nhỏ hơn điểm ảnh. Các ảnh bao gồm ảnh tĩnh ngoài video.

Ảnh là đơn vị xử lý ảnh được cấu hình với tập hợp điểm ảnh, và cũng có thể được gọi là khung hoặc trường. Ảnh có thể, ví dụ, có dạng của mảng của các mẫu độ chói trong khuôn dạng đơn sắc hoặc mảng của các mẫu độ chói và hai mảng của các mẫu sắc độ tương ứng trong khuôn dạng màu 4:2:0, 4:2:2, và 4:4:4.

Khối là đơn vị xử lý mà là tập hợp của số lượng điểm ảnh được xác định. Các khối có thể có số lượng dạng khác nhau bất kỳ. Ví dụ, khối có thể có dạng chữ nhật $M \times N$ (M -cột x N -hàng) điểm ảnh, dạng hình vuông $M \times M$ điểm ảnh, dạng tam giác, dạng hình tròn, v.v. Các ví dụ của các khối bao gồm các lát, ô, viên, CTU, siêu khối, các đơn vị phân chia cơ bản, VPDU, các đơn vị phân chia xử lý đối với phần cứng, các CU, các đơn vị khối xử lý, các đơn vị khối dự đoán (PU), các đơn vị khối biến đổi trực giao (TU), các đơn vị, và các khối con. Khối có thể có dạng của mảng $M \times N$

của các mẫu, hoặc mảng $M \times N$ của các hệ số biến đổi. Ví dụ, khối có thể là vùng hình vuông hoặc hình chữ nhật của các điểm ảnh bao gồm các ma trận một độ chói và hai sắc độ.

Điểm ảnh hoặc mẫu là điểm nhỏ nhất của ảnh. Các điểm ảnh hoặc các mẫu bao gồm điểm ảnh tại vị trí nguyên, cũng như các điểm ảnh tại các vị trí điểm ảnh con, ví dụ, được tạo ra dựa trên điểm ảnh tại vị trí nguyên.

Giá trị điểm ảnh hoặc giá trị mẫu là giá trị đặc trưng của điểm ảnh. Các giá trị điểm ảnh hoặc các giá trị mẫu có thể bao gồm một hoặc nhiều giá trị độ chói, giá trị sắc độ, mức phân cấp RGB, giá trị độ sâu, các giá trị nhị phân của 0 hoặc 1, v.v.

Sắc độ là cường độ của màu sắc, thường được biểu diễn bởi các ký tự C_b và C_r , mà chỉ rõ rằng các giá trị của mảng mẫu hoặc giá trị mẫu đơn biểu diễn các giá trị của một trong số hai tín hiệu chênh lệch màu liên quan đến các màu sơ cấp.

Độ chói là độ sáng của ảnh, thường được biểu diễn bởi ký tự hoặc ký hiệu Y hoặc L , mà chỉ rõ rằng các giá trị của mảng mẫu hoặc giá trị mẫu đơn biểu diễn các giá trị của tín hiệu đơn sắc liên quan đến các màu sơ cấp.

Cờ bao gồm một hoặc nhiều bit mà chỉ báo giá trị, ví dụ, của tham số hoặc chỉ số. Cờ có thể là cờ nhị phân mà chỉ báo giá trị nhị phân của cờ, mà cũng có thể chỉ báo giá trị không phải nhị phân của tham số.

Tín hiệu mang thông tin, mà được ký hiệu hóa bởi hoặc được mã hóa thành tín hiệu. Các tín hiệu bao gồm các tín hiệu số rời rạc và các tín hiệu tương tự liên tục.

Dòng hoặc dòng bit là chuỗi dữ liệu số của dòng dữ liệu số. Dòng hoặc dòng bit có thể là một dòng hoặc có thể được cấu hình với nhiều dòng mà có nhiều lớp phân cấp. Dòng hoặc dòng bit có thể được truyền trong truyền thông nối tiếp nhờ sử dụng một đường truyền, hoặc có thể được truyền trong truyền thông gói tin nhờ sử dụng nhiều đường truyền.

Độ chênh lệch liên quan đến các độ chênh lệch toán học khác nhau, như độ chênh lệch đơn giản $(x - y)$, giá trị chênh lệch tuyệt đối $(|x - y|)$, độ chênh lệch bình phương $(x^2 - y^2)$, căn bậc hai của độ chênh lệch $(\sqrt{x - y})$, độ chênh lệch có trọng

số ($ax - by$: a và b là các hằng số), độ chênh lệch có lượng dịch ($x - y + a$: a là độ dịch), v.v. Trong trường hợp của đại lượng vô hướng, độ chênh lệch đơn giản có thể là đủ, và việc tính toán chênh lệch được bao gồm.

Tổng liên quan đến các tổng toán học khác nhau, như tổng đơn giản ($x + y$), giá của tổng tuyệt đối ($|x + y|$), tổng bình phương ($x^2 + y^2$), căn bậc hai của tổng ($\sqrt{x + y}$), tổng có trọng số ($ax + by$: a và b là các hằng số), tổng có độ dịch ($x + y + a$: a là độ dịch), v.v. Trong trường hợp của đại lượng vô hướng, tổng đơn giản có thể là đủ, và việc tính toán tổng được bao gồm.

Khung là cấu thành của trường trên cùng và trường dưới cùng, trong đó các hàng mẫu 0, 2, 4, ... bắt nguồn từ trường trên cùng và các hàng mẫu 1, 3, 5, ... bắt nguồn từ trường dưới cùng.

Lát là số lượng nguyên của các đơn vị cây mã hóa được chứa trong một phân đoạn lát độc lập và tất cả các phân đoạn lát phụ thuộc tiếp theo (nếu có) mà đứng trước phân đoạn lát độc lập tiếp theo (nếu có) trong cùng đơn vị truy nhập.

Ô là vùng chữ nhật của khối cây mã hóa trong cột ô cụ thể và hàng ô cụ thể trong ảnh. Ô có thể là vùng chữ nhật của khung mà được dự định để có thể được mã hóa và giải mã một cách độc lập, mặc dù việc lọc vòng qua các biên ô có thể vẫn được áp dụng.

Đơn vị cây mã hóa (CTU-coding tree unit) có thể là khối cây mã hóa của các mẫu độ chói của ảnh mà có ba mảng mẫu, hoặc hai khối cây mã hóa tương ứng của các mẫu sắc độ. Ngoài ra, CTU có thể là khối mã hóa của các mẫu của một trong số ảnh đơn sắc và ảnh mà được mã hóa nhờ sử dụng ba mặt phẳng màu riêng biệt và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu. Siêu khối có thể là khối hình vuông của điểm ảnh 64×64 mà bao gồm 1 hoặc 2 các khối thông tin chế độ hoặc được phân chia đệ quy thành bốn khối 32×32 , mà chúng có thể còn được phân chia tiếp.

(Cấu trúc hệ thống)

Đầu tiên, hệ thống truyền theo phương án sẽ được mô tả. FIG.1 là sơ đồ giản lược minh họa một ví dụ về cấu trúc của hệ thống truyền 400 theo phương án của sáng chế.

Hệ thống truyền 400 là hệ thống mà truyền dòng được tạo ra bằng cách mã hóa ảnh và giải mã dòng được truyền. Như được minh họa, hệ thống truyền 400 bao gồm bộ mã hóa 100, mạng 300, và bộ giải mã 200 như được minh họa trong FIG.1.

Ảnh được đưa vào bộ mã hóa 100. Bộ mã hóa 100 tạo ra dòng bằng cách mã hóa ảnh đầu vào, và xuất ra dòng tới mạng 300. Dòng bao gồm, ví dụ, ảnh được mã hóa và thông tin điều khiển để giải mã ảnh được mã hóa. Ảnh được nén bằng cách mã hóa.

Cần lưu ý rằng ảnh trước khi được mã hóa bởi bộ mã hóa 100 cũng được gọi là ảnh gốc, tín hiệu gốc, hoặc mẫu gốc. Ảnh có thể là video hoặc ảnh tĩnh. Ảnh là khái niệm chung của chuỗi, hình ảnh, và khối, và do đó không bị giới hạn ở vùng không gian mà có kích cỡ cụ thể và vùng thời gian mà có kích cỡ cụ thể trừ khi được chỉ rõ khác. Ảnh là mảng của các điểm ảnh hoặc các giá trị điểm ảnh, và tín hiệu mà biểu diễn ảnh hoặc các giá trị điểm ảnh cũng được gọi là các mẫu. Dòng có thể được gọi là dòng bit, dòng bit được mã hóa, dòng bit được nén, hoặc tín hiệu được mã hóa. Ngoài ra, bộ mã hóa 100 có thể được gọi là bộ mã hóa ảnh hoặc bộ mã hóa video. Phương pháp mã hóa được thực hiện bởi bộ mã hóa 100 có thể được gọi là phương pháp mã hóa, phương pháp mã hóa ảnh, hoặc phương pháp mã hóa video.

Mạng 300 truyền dòng được tạo ra bởi bộ mã hóa 100 tới bộ giải mã 200. Mạng 200 có thể là mạng Internet, mạng vùng rộng (WAN-Wide Area Network), mạng vùng cục bộ (LAN-Local Area Network), hoặc kết hợp bất kỳ của các mạng. Mạng 300 không bị giới hạn ở mạng truyền thông hai chiều, và có thể là mạng truyền thông một chiều mà truyền các sóng quảng bá của quảng bá mặt đất số, quảng bá vệ tinh, hoặc loại tương tự. Ngoài ra, mạng 300 có thể được thay thế bởi vật ghi như Đĩa đa năng số (DVD-Digital Versatile Disc) và đĩa Blue-Ray (BD-Blue-Ray), v.v mà trên đó dòng được ghi.

Bộ giải mã 200 tạo ra, ví dụ, ảnh được giải mã mà là ảnh không được nén, bằng cách giải mã dòng được truyền bởi mạng 300. Ví dụ, bộ giải mã giải mã dòng

theo phương pháp giải mã tương ứng với phương pháp mã hóa được sử dụng bởi bộ mã hóa 100.

Cần lưu ý rằng bộ giải mã 200 có thể cũng được gọi là bộ giải mã ảnh hoặc bộ giải mã video, và phương pháp giải mã được thực hiện bởi bộ giải mã 200 có thể cũng được gọi là phương pháp giải mã, phương pháp giải mã ảnh, hoặc phương pháp giải mã video.

(Cấu trúc dữ liệu)

FIG.2 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc phân cấp của dữ liệu trong dòng. Nhằm thuận tiện, FIG.2 sẽ được mô tả có viện dẫn tới hệ thống truyền 400 của FIG.1. Dòng bao gồm, ví dụ, chuỗi video. Như được minh họa trong phần (a) của FIG.2, chuỗi video bao gồm một hoặc nhiều tập hợp tham số video (VPS-video parameter set), một hoặc nhiều tập hợp tham số chuỗi (SPS-sequence parameter set), một hoặc nhiều tập hợp tham số ảnh (PPS-picture parameter set), thông tin nâng cao bổ sung (SEI-supplemental enhancement information), và nhiều ảnh.

Trong video mà có nhiều lớp, VPS có thể bao gồm tham số mã hóa mà là chung giữa một vài lớp, và tham số mã hóa liên quan đến một vài lớp được chứa trong video hoặc liên quan đến lớp riêng biệt.

SPS bao gồm tham số mà được sử dụng cho chuỗi, tức là, tham số mã hóa mà bộ giải mã 200 viện dẫn tới để giải mã chuỗi. Ví dụ, tham số mã hóa có thể chỉ báo độ rộng hoặc độ cao của ảnh. Cần lưu ý rằng các SPS có thể được hiện diện.

PPS bao gồm tham số mà được sử dụng cho ảnh, tức là, tham số mã hóa mà bộ giải mã 200 viện dẫn tới để giải mã mỗi ảnh trong chuỗi. Ví dụ, tham chiếu mã hóa có thể bao gồm giá trị tham chiếu đối với độ rộng lượng tử hóa mà được sử dụng để giải mã ảnh và cờ mà chỉ báo việc áp dụng dự đoán có trọng số. Cần lưu ý rằng các PPS có thể được hiện diện. Mỗi SPS và PPS có thể được gọi đơn giản là tập hợp tham số.

Như được minh họa trong phần (b) của FIG.2, ảnh có thể bao gồm thông tin tiêu đề ảnh và một hoặc nhiều lát. Thông tin tiêu đề ảnh bao gồm tham số mã hóa mà bộ giải mã 200 viện dẫn tới để giải mã một hoặc nhiều lát.

Như được minh họa trong phần (c) của FIG.2, lát bao gồm thông tin tiêu đề lát và một hoặc nhiều viên. Thông tin tiêu đề lát bao gồm tham số mã hóa mà bộ giải mã 200 viện dẫn tới để giải mã một hoặc nhiều viên.

Như được minh họa trong phần (d) của FIG.2, viên bao gồm một hoặc nhiều đơn vị cây mã hóa (CTU-coding tree unit).

Cần lưu ý rằng ảnh có thể không bao gồm bất kỳ lát và có thể bao gồm nhóm ô thay vì lát. Trong trường hợp này, nhóm ô bao gồm ít nhất một ô. Ngoài ra, viên có thể bao gồm lát.

CTU cũng được gọi là siêu khối hoặc đơn vị phân chia cơ sở. Như được minh họa trong phần (e) của FIG.2, CTU bao gồm thông tin tiêu đề CTU và ít nhất một đơn vị mã hóa (CU-coding unit). Như được minh họa, CTU bao gồm bốn đơn vị mã hóa CU(10), CU(11), CU(12) và CU(13). Thông tin tiêu đề CTU bao gồm tham số mã hóa mà bộ giải mã 200 viện dẫn tới để giải mã ít nhất một CU.

CU có thể được phân chia thành nhiều CU nhỏ hơn. Như được thể hiện, CU(10) không được chia thành các đơn vị mã hóa nhỏ hơn; CU(11) được chia thành bốn đơn vị mã hóa nhỏ hơn CU(110), CU(111), CU(112) và CU(113); CU(12) không được chia thành các đơn vị mã hóa nhỏ hơn; và CU(13) được chia thành bảy đơn vị mã hóa nhỏ hơn CU(1310), CU(1311), CU(1312), CU(1313), CU(132), CU(133) và CU(134). Như được minh họa trong (f) của FIG.2, CU bao gồm thông tin tiêu đề CU, thông tin dự đoán, và thông tin hệ số dư. Thông tin dự đoán là thông tin để dự đoán CU, và thông tin hệ số dư là thông tin mà chỉ báo phần dư dự đoán được mô tả sau đây. Mặc dù CU về cơ bản tương tự như đơn vị dự đoán (PU-prediction unit) và đơn vị biến đổi (TU-transform unit), cần lưu ý rằng, ví dụ, biến đổi khối con (SBT-sub-block transform) được mô tả sau đây có thể bao gồm nhiều TU nhỏ hơn CU. Ngoài ra, CU có thể được xử lý đối với mỗi đơn vị mã hóa đường ống ảo (VPDU-virtual pipeline decoding unit) được chứa trong CU. VPDU là, ví dụ, đơn

vị cố định mà có thể được xử lý tại một giai đoạn khi xử lý đường ống được thực hiện trong phần cứng.

Cần lưu ý rằng dòng có thể không bao gồm tất cả lớp phân cấp được minh họa trong FIG.2. Bậc của lớp phân cấp có thể được trao đổi, hoặc bất kỳ các lớp phân cấp có thể được thay thế bởi lớp phân cấp khác. Ở đây, ảnh mà là đích cho xử lý mà sẽ được thực hiện bởi thiết bị như bộ mã hóa 100 hoặc bộ giải mã 200 được gọi là ảnh hiện tại. Ảnh hiện tại có nghĩa là ảnh hiện tại cần được mã hóa khi xử lý là xử lý mã hóa, và ảnh hiện tại có nghĩa là ảnh hiện tại cần được giải mã khi xử lý là xử lý giải mã. Tương tự, ví dụ, CU hoặc khối của các CU mà là đích cho xử lý mà sẽ được thực hiện bởi thiết bị như bộ mã hóa 100 hoặc bộ giải mã 200 được gọi là khối hiện tại. Khối hiện tại có nghĩa là khối hiện tại cần được mã hóa khi xử lý là xử lý mã hóa, và khối hiện tại có nghĩa là khối hiện tại cần được giải mã khi xử lý là xử lý giải mã.

(Cấu trúc ảnh: Lát/ô)

Ảnh có thể được cấu hình với một hoặc nhiều đơn vị lát hoặc một hoặc nhiều đơn vị ô để hỗ trợ việc mã hóa/giải mã của ảnh một cách song song.

Các lát là các đơn vị mã hóa cơ bản được chứa trong ảnh. Ảnh có thể bao gồm, ví dụ, một hoặc nhiều lát. Ngoài ra, lát bao gồm một hoặc nhiều đơn vị cây mã hóa (CTU-coding tree unit).

FIG.3 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc lát. Ví dụ, trong FIG.3 ảnh bao gồm các CTU 11×8 CTU, và được phân chia thành bốn lát (các lát 1 đến 4). Lát 1 bao gồm mười sáu CTU, lát 2 bao gồm hai mươi một CTU, lát 3 bao gồm hai mươi chín CTU, và lát 4 bao gồm hai mươi hai CTU. Ở đây, mỗi CTU trong ảnh thuộc về một trong số các lát. Dạng của mỗi lát là dạng thu được bằng cách phân chia ảnh theo chiều ngang. Biên của mỗi lát không cần trùng với đầu mút của ảnh, và có thể trùng với bất kỳ trong số các biên giữa các CTU trong ảnh. Thứ tự xử lý của các CTUs trong lát (thứ tự mã hóa hoặc thứ tự giải mã) là, ví dụ, thứ tự quét mảnh. Lát bao gồm thông tin tiêu đề lát và dữ liệu được mã hóa. Các đặc điểm của lát có thể được ghi trong thông tin tiêu đề lát. Các đặc điểm này có thể bao gồm địa chỉ CTU của CTU trên cùng trong lát, loại lát, v.v.

Ô là đơn vị của vùng chữ nhật được chứa trong ảnh. Các ô của ảnh có thể được gán với số hiệu được gọi là TileId trong thứ tự quét màn hình.

FIG.4 là sơ đồ khái niệm để minh họa một ví dụ về cấu trúc ô. Ví dụ, trong FIG.4 ảnh bao gồm các CTU 11×8 , và được phân chia thành bốn ô của các vùng chữ nhật (các ô 1 đến 4). Khi các ô được sử dụng, thứ tự xử lý của các CTU có thể khác với thứ tự xử lý trong trường hợp trong đó các ô không được sử dụng. Khi không có ô được sử dụng, các CTU trong ảnh nói chung được xử lý trong thứ tự quét màn hình. Khi nhiều ô được sử dụng, ít nhất một CTU trong mỗi ô được xử lý trong thứ tự quét màn hình. Ví dụ, như được minh họa trong FIG.4 thứ tự xử lý của các CTU được chứa trong ô 1 từ phía trái của cột thứ nhất của ô 1 hướng tới phía phải của cột thứ nhất của ô 1 và sau đó tiếp tục từ phía trái của cột thứ hai của ô 1 hướng tới phía phải của cột thứ hai của ô 1.

Cần lưu ý rằng một ô có thể bao gồm một hoặc nhiều lát, và một lát có thể bao gồm một hoặc nhiều ô.

Cần lưu ý rằng ảnh có thể được cấu hình với một hoặc nhiều tập hợp ô. Tập hợp ô có thể bao gồm một hoặc nhiều nhóm ô, hoặc một hoặc nhiều ô. Ảnh có thể được cấu hình với một trong số tập hợp ô, nhóm ô, và ô. Ví dụ, thứ tự quét của các ô đối với mỗi tập hợp ô trong thứ tự quét màn hình được giả định là thứ tự mã hóa cơ sở của các ô. Tập hợp của một hoặc nhiều ô mà liên tục trong thứ tự mã hóa cơ sở trong mỗi tập hợp ô được giả định là nhóm ô. Ảnh này có thể được cấu hình bởi bộ phân chia 102 (xem FIG.7) được mô tả sau đây.

(Mã hóa có thể biến đổi)

Các Fig.5 và Fig.6 là các sơ đồ khái niệm minh họa các ví dụ về các cấu trúc dòng có thể biến đổi, và sẽ được mô tả nhằm thuận tiện có viện dẫn tới FIG.1.

Như được minh họa trong FIG.5, bộ mã hóa 100 có thể tạo ra dòng có thể biến đổi theo thời gian/không gian bằng cách chia mỗi ảnh thành bất kỳ lớp và mã hóa ảnh trong lớp. Ví dụ, bộ mã hóa 100 mã hóa ảnh đối với mỗi lớp, nhờ đó đạt được khả năng biến đổi trong đó lớp nâng cao nằm phía trên lớp cơ sở. Việc mã hóa này của mỗi ảnh cũng được gọi là mã hóa có thể biến đổi. Theo cách này, bộ giải mã 200 có thể chuyển đổi chất lượng ảnh của ảnh mà được hiển thị bằng cách giải

mã dòng. Nói cách khác, bộ giải mã 200 có thể xác định lớp nào để giải mã dựa trên các yếu tố bên trong như khả năng xử lý của bộ giải mã 200 và các yếu tố bên ngoài như trạng thái của băng thông truyền thông. Kết quả là, bộ giải mã 200 có khả năng giải mã nội dung trong khi chuyển đổi tự do giữa độ phân giải thấp và độ phân giải cao. Ví dụ, người dùng của dòng xem video của dòng nửa chừng nhờ sử dụng điện thoại thông minh trên đường về nhà, và tiếp tục xem video tại nhà trên thiết bị như TV được kết nối tới Internet. Cần lưu ý rằng mỗi điện thoại thông minh và thiết bị được mô tả nêu trên bao gồm bộ giải mã 200 mà có các chức năng giống hoặc khác nhau. Trong trường hợp này, khi thiết bị giải mã các lớp tới lớp cao hơn trong dòng, người dùng có thể xem video tại chất lượng cao ở nhà. Theo cách này, bộ mã hóa 100 không cần tạo ra các dòng mà có các chất lượng ảnh khác nhau của cùng nội dung, và nhờ đó tải xử lý có thể được làm giảm.

Ngoài ra, lớp nâng cao có thể bao gồm siêu thông tin dựa trên thông tin thống kê về ảnh. Bộ giải mã 200 có thể tạo ra video mà chất lượng ảnh của nó được tăng cường bằng cách thực hiện việc tạo ảnh siêu phân giải trên ảnh trong lớp cơ sở dựa trên siêu dữ liệu. Việc tạo ảnh siêu phân giải có thể bao gồm, ví dụ, việc cải thiện về tỷ lệ SN trong cùng độ phân giải, gia tăng độ phân giải, v.v. Siêu dữ liệu có thể bao gồm, ví dụ, thông tin để nhận dạng hệ số lọc tuyến tính hoặc phi tuyến, như được sử dụng trong xử lý siêu phân giải, hoặc thông tin mà nhận dạng giá trị tham số trong xử lý lọc, học máy, hoặc phương pháp bình phương tối thiểu được sử dụng trong xử lý siêu phân giải, v.v.

Theo phương án của sáng chế, cấu trúc có thể được đề xuất trong đó ảnh được chia thành, ví dụ, các lát theo, ví dụ, ý nghĩa của đối tượng trong ảnh. Trong trường hợp này, bộ giải mã 200 có thể giải mã chỉ một phần vùng trong ảnh bằng cách lựa chọn ô cần được giải mã. Ngoài ra, thuộc tính của đối tượng (người, xe, bóng, v.v.) và vị trí của đối tượng trong ảnh (các tọa độ trong các ảnh đồng nhất) có thể được lưu trữ như là siêu dữ liệu. Trong trường hợp này, bộ giải mã 200 có khả năng nhận dạng vị trí của đối tượng mong muốn dựa trên siêu dữ liệu, và xác định ô bao gồm đối tượng. Ví dụ, như được minh họa trong FIG.6, siêu dữ liệu có thể được lưu trữ nhờ sử dụng cấu trúc lưu trữ dữ liệu khác với dữ liệu ảnh, như bản tin SEI

(supplemental enhancement information-thông tin nâng cao bổ sung) trong HEVC. Siêu dữ liệu này chỉ báo, ví dụ, vị trí, kích cỡ, hoặc màu của đối tượng chính.

Siêu dữ liệu có thể được lưu trữ trong các đơn vị của các ảnh, như dòng, chuỗi, hoặc đơn vị truy nhập ngẫu nhiên. Theo cách này, bộ giải mã 200 có thể thu nhận, ví dụ, thời điểm mà tại đó người cụ thể xuất hiện trong video, và bằng cách ghép thông tin thời gian với thông tin đơn vị ảnh, có thể nhận dạng ảnh trong đó đối tượng (người) được hiện diện và xác định vị trí của đối tượng trong ảnh.

(Bộ mã hóa)

Bộ mã hóa theo phương án của sáng chế sẽ được mô tả. FIG.7 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa 100 theo phương án của sáng chế. Bộ mã hóa 100 là bộ mã hóa video mà mã hóa video trong các đơn vị của khối.

Như được minh họa trong FIG.7, bộ mã hóa 100 là thiết bị mà mã hóa ảnh trong các đơn vị của khối, và bao gồm bộ phân chia 102, bộ trừ 104, bộ biến đổi 106, bộ lượng tử hóa 108, bộ mã hóa entropy 110, bộ lượng tử hóa ngược 112, bộ biến đổi ngược 114, bộ cộng 116, bộ nhớ khối 118, bộ lọc vòng 120, bộ nhớ khung 122, bộ nội dự đoán 124, bộ dự đoán liên đới 126, bộ điều khiển dự đoán 128, và bộ tạo tham số dự đoán 130. Như được minh họa, bộ nội dự đoán 124 và bộ dự đoán liên đới 126 là một phần của bộ điều khiển dự đoán.

Bộ mã hóa 100 được thực hiện như là, ví dụ, bộ xử lý chung và bộ nhớ. Trong trường hợp này, khi chương trình phần mềm được lưu trữ trong bộ nhớ được thực thi bởi bộ xử lý, bộ xử lý thực hiện chức năng như là bộ phân chia 102, bộ trừ 104, bộ biến đổi 106, bộ lượng tử hóa 108, bộ mã hóa entropy 110, bộ lượng tử hóa ngược 112, bộ biến đổi ngược 114, bộ cộng 116, bộ lọc vòng 120, bộ dự đoán trong 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128. Ngoài ra, bộ mã hóa 100 có thể được thực hiện như là một hoặc nhiều mạch điện tử dành riêng tương ứng với bộ phân chia 102, bộ trừ 104, bộ biến đổi 106, bộ lượng tử hóa 108, bộ mã hóa entropy 110, bộ lượng tử hóa ngược 112, bộ biến đổi ngược 114, bộ cộng 116, bộ lọc vòng 120, bộ dự đoán trong 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128.

(Ví dụ bố trí của bộ mã hóa)

FIG.8 là sơ đồ khối chức năng minh họa ví dụ bố trí của bộ mã hóa 100. Bộ mã hóa 100 bao gồm bộ xử lý a1 và bộ nhớ a2. Ví dụ, các bộ phận cấu thành của bộ mã hóa 100 được minh họa trong FIG.7 được bố trí trên bộ xử lý a1 và bộ nhớ a2 được minh họa trong FIG.8.

Bộ xử lý a1 là mạch mà thực hiện xử lý thông tin và được ghép nối với bộ nhớ a2. Ví dụ, bộ xử lý a1 là mạch điện tử dành riêng hoặc chung mà mã hóa ảnh. Bộ xử lý a1 có thể là bộ xử lý như CPU. Ngoài ra, bộ xử lý a1 có thể là kết hợp của nhiều mạch điện tử. Ngoài ra, ví dụ, bộ xử lý a1 có thể đóng vai trò của hai thành phần cấu thành hoặc nhiều hơn trong số các thành phần cấu thành của bộ mã hóa 100 được minh họa trong FIG.7, v.v.

Bộ nhớ a2 là bộ nhớ chung hoặc dành riêng để lưu trữ thông tin mà được sử dụng bởi bộ xử lý a1 để mã hóa ảnh. Bộ nhớ a2 có thể là mạch điện tử, và có thể được kết nối tới bộ xử lý a1. Ngoài ra, bộ nhớ a2 có thể được chứa trong bộ xử lý a1. Ngoài ra, bộ nhớ a2 có thể là kết hợp của nhiều mạch điện tử. Ngoài ra, bộ nhớ a2 có thể là đĩa từ, đĩa quang, hoặc loại tương tự, hoặc có thể được biểu diễn như là bộ lưu trữ, vật ghi, hoặc loại tương tự. Ngoài ra, bộ nhớ a2 có thể là bộ nhớ bất biến, hoặc bộ nhớ khả biến.

Ví dụ, bộ nhớ a2 có thể lưu trữ ảnh cần được mã hóa hoặc dòng bit tương ứng với ảnh được mã hóa. Ngoài ra, bộ nhớ a2 có thể lưu trữ chương trình để làm cho bộ xử lý a1 mã hóa ảnh.

Ngoài ra, ví dụ, bộ nhớ a2 có thể đóng vai trò của hai thành phần cấu thành hoặc nhiều hơn để lưu trữ thông tin trong số các thành phần cấu thành của bộ mã hóa 100 được minh họa trong FIG.7, v.v. ví dụ, bộ nhớ a2 có thể đóng vai trò của bộ nhớ khối 118 và bộ nhớ khung 122 được minh họa trong FIG.7. Cụ thể hơn, bộ nhớ a2 có thể lưu trữ khối được khôi phục, ảnh được khôi phục, v.v.

Cần lưu ý rằng, trong bộ mã hóa 100, tất cả các thành phần cấu thành được chỉ báo trong FIG.7, v.v. có thể không được thực hiện, và tất cả xử lý được mô tả ở đây có thể không được thực hiện. Một phần của các thành phần cấu thành được thể hiện trong FIG.7, v.v. có thể được chứa trong thiết bị khác, hoặc một phần của các xử lý được mô tả ở đây có thể được thực hiện bởi thiết bị khác.

Sau đây, dòng xử lý chung được thực hiện bởi bộ mã hóa 100 được mô tả, và sau đó mỗi thành phần cấu thành được chứa trong bộ mã hóa 100 sẽ được mô tả.

(Dòng xử lý mã hóa chung)

FIG.9 là lưu đồ mà thể hiện một ví dụ về xử lý mã hóa chung được thực hiện bởi bộ mã hóa 100, và nhằm thuận tiện sẽ được mô tả có viện dẫn tới FIG.7.

Đầu tiên, bộ phân chia 102 của bộ mã hóa 100 phân chia mỗi ảnh được chứa trong ảnh đầu vào thành các khối có kích cỡ cố định (ví dụ, các điểm ảnh 128×128) (bước Sa_1). Sau đó, bộ phân chia 102 lựa chọn mẫu phân chia đối với khối có kích cỡ cố định (cũng được gọi là dạng khối) (bước Sa_2). Nói cách khác, bộ phân chia 102 còn phân chia khối có kích cỡ cố định thành các khối mà tạo thành mẫu phân chia được lựa chọn. Bộ mã hóa 100 thực hiện, đối với mỗi khối, các bước Sa_3 đến Sa_9 đối với khối (mà là khối hiện tại cần được mã hóa).

Bộ điều khiển dự đoán 128 và bộ thực thi dự đoán (mà bao gồm bộ nội dự đoán 124 và bộ dự đoán liên đới 126) tạo ra ảnh dự đoán của khối hiện tại (bước Sa-3). Ảnh dự đoán có thể cũng được gọi là tín hiệu dự đoán, khối dự đoán, hoặc các mẫu dự đoán.

Tiếp theo, bộ trừ 104 tạo ra độ chênh lệch giữa khối hiện tại và ảnh dự đoán như là phần dư dự đoán (bước Sa_4). Phần dư dự đoán có thể cũng được gọi là sai số dự đoán.

Tiếp theo, bộ biến đổi 106 biến đổi ảnh dự đoán và bộ lượng tử hóa 108 lượng tử hóa kết quả thu được, để tạo ra các hệ số được lượng tử hóa (bước Sa_5). Các hệ số được lượng tử hóa có thể đôi lúc được gọi là khối hệ số.

Tiếp theo, bộ mã hóa entropy 110 mã hóa (cụ thể, mã hóa entropy) các hệ số được lượng tử hóa và tham số dự đoán liên quan đến việc tạo ra ảnh dự đoán, để tạo ra dòng (bước Sa_6). Dòng có thể đôi lúc được gọi là dòng bit được mã hóa hoặc dòng bit được nén.

Tiếp theo, bộ lượng tử hóa ngược 112 thực hiện việc lượng tử hóa ngược của các hệ số được lượng tử hóa và bộ biến đổi ngược 114 thực hiện việc biến đổi ngược của kết quả thu được, để khôi phục phần dư dự đoán (bước Sa_7).

Tiếp theo, bộ cộng 116 cộng ảnh dự đoán vào phần dư dự đoán được lưu trữ để khôi phục khối hiện tại (bước Sa_8). Theo cách này, ảnh được khôi phục được tạo ra. Ảnh được khôi phục có thể cũng được gọi là khối được khôi phục hoặc khối ảnh được giải mã.

Khi ảnh được khôi phục được tạo ra, bộ lọc vòng 120 thực hiện việc lọc của ảnh được khôi phục nếu cần (bước Sa_9).

Bộ mã hóa 100 then xác định rằng việc mã hóa của toàn bộ ảnh đã được hoàn thành hay chưa (bước Sa_10). Khi xác định rằng việc mã hóa chưa được hoàn thành (Không trong bước Sa_10), việc thực hiện xử lý từ bước Sa_2 được lặp lại đối với khối tiếp theo của ảnh.

Mặc dù bộ mã hóa 100 lựa chọn một mẫu phân chia đối với khối có kích cỡ cố định, và mã hóa mỗi khối theo mẫu phân chia trong ví dụ nêu trên, cần lưu ý rằng mỗi khối có thể được mã hóa theo một trong số các mẫu phân chia tương ứng. Trong trường hợp này, bộ mã hóa 100 có thể đánh giá giá trị đối với mỗi mẫu phân chia, và ví dụ, có thể lựa chọn dòng có thể thu được bằng cách mã hóa theo mẫu phân chia mà có giá trị nhỏ nhất như là dòng mà được xuất ra.

Như được minh họa, các xử lý trong các bước Sa_1 đến Sa_10 được thực hiện tuần tự bởi bộ mã hóa 100. Ngoài ra, hai xử lý hoặc nhiều hơn có thể được thực hiện một cách song song, các xử lý có thể được sắp xếp lại, v.v.

Xử lý mã hóa được sử dụng bởi bộ mã hóa 100 là việc mã hóa lai sử dụng mã hóa dự đoán và mã hóa biến đổi. Ngoài ra, việc mã hóa dự đoán được thực hiện bởi vòng mã hóa được cấu hình với bộ trừ 104, bộ biến đổi 106, bộ lượng tử hóa 108, bộ lượng tử hóa ngược 112, bộ biến đổi ngược 114, bộ cộng 116, bộ lọc vòng 120, bộ nhớ khối 118, bộ nhớ khung 122, bộ nội dự đoán 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128. Nói cách khác, bộ thực thi dự đoán được cấu hình với bộ nội dự đoán 124 và bộ dự đoán liên đới 126 là một phần của vòng mã hóa.

(Bộ phân chia)

Bộ phân chia 102 phân chia mỗi ảnh được chứa trong ảnh gốc thành nhiều khối, và xuất ra mỗi khối tới bộ trừ 104. Ví dụ, bộ phân chia 102 đầu tiên phân chia

ảnh thành các khối có kích cỡ cố định (ví dụ, 128×128 điểm ảnh). Các kích cỡ khối cố định khác có thể được sử dụng. Khối có kích cỡ cố định cũng được gọi là đơn vị cây mã hóa (CTU-coding tree unit). Bộ phân chia 102 sau đó phân chia mỗi khối có kích cỡ cố định thành các khối có các kích cỡ biến thiên (ví dụ, 64×64 điểm ảnh hoặc nhỏ hơn), dựa trên phân chia cây tứ phân đệ quy và/hoặc phân chia khối cây nhị phân. Nói cách khác, bộ phân chia 102 lựa chọn mẫu phân chia. Khối có kích cỡ biến thiên có thể cũng được gọi là đơn vị mã hóa (CU-coding unit), đơn vị dự đoán (PU-prediction unit), hoặc đơn vị biến đổi (TU-transform unit). Cần lưu ý rằng, trong các loại ví dụ xử lý khác nhau, không cần phân biệt giữa CU, PU, và TU; tất cả hoặc một vài khối trong ảnh có thể được xử lý trong các đơn vị của CU, PU, hoặc TU.

FIG.10 là sơ đồ khái niệm để minh họa một ví dụ về phân chia khối theo phương án của sáng chế. Trong FIG.10, các đường liền nét thể hiện các biên khối của các khối được phân chia bằng cách phân chia khối cây tứ phân, và các đường đứt nét thể hiện các biên khối của các khối được phân chia bằng cách phân chia khối cây nhị phân.

Ở đây, khối 10 là khối vuông có 128×128 điểm ảnh (khối 128×128). Khối 128×128 10 này đầu tiên được phân chia thành bốn khối vuông 64×64 điểm ảnh (phân chia khối cây tứ phân).

Khối 64×64 điểm ảnh phía trên bên trái còn được phân chia theo chiều dọc thành hai khối 32×64 điểm ảnh hình chữ nhật, và khối 32×64 điểm ảnh bên trái còn được phân chia theo chiều dọc thành hai khối 16×64 điểm ảnh hình chữ nhật (phân chia khối cây nhị phân). Kết quả là, khối 64×64 điểm ảnh phía trên bên trái được phân chia thành hai khối 16×64 điểm ảnh 11 và 12 và một khối 32×64 điểm ảnh 13.

Khối 64×64 điểm ảnh phía trên bên phải được phân chia theo chiều ngang thành hai khối 64×32 điểm ảnh hình chữ nhật 14 và 15 (phân chia khối cây nhị phân).

Khối 64×64 điểm ảnh phía dưới bên trái được phân chia lần thứ nhất thành bốn khối 32×32 điểm ảnh hình vuông (phân chia khối cây tứ phân). Khối phía trên bên trái và khối phía dưới bên phải trong số bốn khối vuông 32×32 điểm ảnh được phân chia tiếp. Khối 32×32 điểm ảnh phía trên bên trái được phân chia theo chiều

dọc thành hai khối 16×32 điểm ảnh hình chữ nhật, và khối 16×32 điểm ảnh bên phải được phân chia theo chiều ngang tiếp thành hai khối 16×16 điểm ảnh (phân chia khối cây nhị phân). Khối 32×32 điểm ảnh phía dưới bên phải được phân chia theo chiều ngang thành hai khối 32×16 điểm ảnh (phân chia khối cây nhị phân). Khối 32×32 điểm ảnh hình vuông phía trên bên phải được phân chia theo chiều ngang thành hai khối 32×16 điểm ảnh hình chữ nhật (phân chia khối cây nhị phân). Kết quả là, khối vuông 64×64 điểm ảnh phía dưới bên trái được phân chia thành khối chữ nhật 16×32 điểm ảnh 16, hai khối vuông 16×16 điểm ảnh 17 và 18, hai khối vuông 32×32 điểm ảnh 19 và 20, và hai khối chữ nhật 32×16 điểm ảnh 21 và 22.

Khối phía dưới bên phải 64×64 điểm ảnh 23 không được phân chia.

Như được mô tả nêu trên, trong FIG.10, khối 10 được phân chia thành 13 khối có kích cỡ biến thiên 11 đến 23 dựa trên phân chia khối cây nhị phân hoặc cây tứ phân đệ quy. Loại phân chia này cũng được gọi là phân chia cây nhị phân cộng cây tứ phân (QTBT-quadtrees plus binary tree).

Cần lưu ý rằng, trong FIG.10, một khối được phân chia thành bốn hoặc hai khối (phân chia khối cây nhị phân hoặc cây tứ phân), nhưng việc phân chia không bị giới hạn ở các ví dụ này. Ví dụ, một khối có thể được chia thành ba khối (phân chia khối bậc ba). Việc phân chia bao gồm phân chia khối bậc ba này cũng được gọi là phân chia cây đa loại (MBT-multi-type tree).

FIG.11 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ phân chia theo phương án của sáng chế. Như được minh họa trong FIG.11, bộ phân chia 102 có thể bao gồm bộ xác định phân chia khối 102a. Bộ xác định phân chia khối 102a có thể thực hiện các xử lý sau đây như là ví dụ.

Ví dụ, bộ xác định phân chia khối 102a có thể thu nhận hoặc truy hồi thông tin khối từ bộ nhớ khối 118 và/hoặc bộ nhớ khung 122, và xác định mẫu phân chia (ví dụ, mẫu phân chia nêu trên) dựa trên thông tin khối. Bộ phân chia 102 chia ảnh gốc theo mẫu phân chia, và xuất ra ít nhất một khối thu được bằng cách phân chia tới bộ trừ 104.

Ngoài ra, ví dụ, bộ xác định phân chia khối 102a xuất ra một hoặc nhiều tham số mà chỉ báo mẫu phân chia được xác định (ví dụ, mẫu phân chia nêu trên) tới bộ

biến đổi 106, bộ biến đổi ngược 114, bộ nội dự đoán 124, bộ dự đoán liên đới 126, và bộ mã hóa entropy 110. Bộ biến đổi 106 có thể biến đổi phần dư dự đoán dựa trên một hoặc nhiều tham số. Bộ nội dự đoán 124 và bộ dự đoán liên đới 126 có thể tạo ra ảnh dự đoán dựa trên một hoặc nhiều tham số. Ngoài ra, bộ mã hóa entropy 110 có thể mã hóa entropy một hoặc nhiều tham số.

Tham số liên quan đến mẫu phân chia có thể được ghi trong dòng như được thể hiện dưới đây như là một ví dụ.

FIG.12 là sơ đồ khái niệm để minh họa các ví dụ về các mẫu phân chia. Các ví dụ của các mẫu phân chia bao gồm: phân chia thành bốn vùng (QT) trong đó khối được phân chia thành hai vùng cả theo chiều dọc và chiều ngang; phân chia thành ba vùng (HT hoặc VT) trong đó khối được phân chia trong cùng chiều trong tỷ lệ của 1:2:1; phân chia khối thành hai vùng (HB hoặc VB) trong đó khối được phân chia trong cùng chiều trong tỷ lệ của 1:1; và không phân chia (NS).

Cần lưu ý rằng mẫu phân chia không có chiều phân chia khối trong trường hợp phân chia thành bốn vùng và không phân chia, và mẫu phân chia có thông tin chiều phân chia trong trường hợp phân chia thành hai vùng hoặc ba vùng.

FIG.13A là sơ đồ khái niệm để minh họa một ví dụ về cây cú pháp của mẫu phân chia.

FIG.13B là sơ đồ khái niệm để minh họa ví dụ khác về cây cú pháp của mẫu phân chia.

Các Fig.13A và Fig.13B là các sơ đồ khái niệm để minh họa các ví dụ về cây cú pháp của mẫu phân chia. Trong ví dụ của FIG.13A, đầu tiên, thông tin mà chỉ báo có thực hiện thực hiện việc phân chia hay không (S: Cờ phân chia) được hiện diện, và thông tin mà chỉ báo rằng có thực hiện việc phân chia thành bốn vùng hay không (QT: Cờ QT) được hiện diện tiếp theo. Thông tin mà chỉ báo việc phân chia nào trong số phân chia thành ba vùng và hai vùng cần được thực hiện (TT: Cờ TT hoặc BT: Cờ BT) được hiện diện tiếp theo, và thông tin mà chỉ báo chiều phân chia (Ver: Cờ dọc hoặc Hor: Cờ ngang) được hiện diện sau đó. Cần lưu ý rằng mỗi trong số ít nhất một khối thu được bằng cách phân chia theo mẫu phân chia này có thể được phân chia tiếp lặp lại trong xử lý tương tự. Nói cách khác, như là một ví dụ, việc

phân chia có được thực hiện hay không, việc phân chia thành bốn vùng có được thực hiện hay không, chiều nào trong số chiều ngang và chiều dọc là chiều trong đó phương pháp phân chia cần được thực hiện, việc phân chia nào trong số phân chia thành ba vùng và phân chia thành hai cùng cần được thực hiện có thể được xác định đệ quy, và các kết quả xác định có thể được mã hóa trong dòng theo thứ tự mã hóa được bộc lộ bởi cây cú pháp được minh họa trong FIG.13A.

Ngoài ra, mặc dù các mục thông tin một cách lần lượt chỉ báo S, QT, TT, và Ver được sắp xếp trong thứ tự được liệt kê trong cây cú pháp được minh họa trong FIG.13A, các mục thông tin một cách lần lượt chỉ báo S, QT, Ver, và BT có thể được sắp xếp trong thứ tự được liệt kê. Nói cách khác, trong ví dụ của FIG.13B, đầu tiên, thông tin mà chỉ báo có thực hiện thực hiện việc phân chia hay không (S: Cờ phân chia) được hiện diện, và thông tin mà chỉ báo rằng có thực hiện việc phân chia thành bốn vùng hay không (QT: Cờ QT) được hiện diện tiếp theo. Thông tin mà chỉ báo chiều phân chia (Ver: Cờ dọc hoặc Hor: Cờ ngang) được hiện diện tiếp theo, và thông tin mà chỉ báo việc phân chia nào trong số phân chia thành hai vùng và phân chia thành ba vùng cần được thực hiện (BT: Cờ BT hoặc TT: Cờ TT) được hiện diện sau đó.

Cần lưu ý rằng các mẫu phân chia được mô tả nêu trên là các ví dụ, và các mẫu phân chia khác ngoài các mẫu phân chia được mô tả có thể được sử dụng, hoặc một phần của các mẫu phân chia được mô tả có thể được sử dụng.

(Bộ trừ)

Bộ trừ 104 trừ ảnh dự đoán (mẫu dự đoán mà được đưa vào từ bộ điều khiển dự đoán 128 được thể hiện dưới đây) từ ảnh gốc trong các đơn vị của khối được đưa vào từ bộ phân chia 102 và được phân chia bởi bộ phân chia 102. Nói cách khác, bộ trừ 104 tính toán các phần dư dự đoán (cũng được gọi là các sai số) của khối hiện tại. Bộ trừ 104 sau đó xuất các phần dư dự đoán được tính toán tới bộ biến đổi 106.

Ảnh gốc có thể là ảnh mà đã được đưa vào bộ mã hóa 100 như là tín hiệu mà biểu diễn ảnh của mỗi hình ảnh được chứa trong video (ví dụ, tín hiệu độ chói và hai tín hiệu sắc độ). Tín hiệu mà biểu diễn ảnh cũng có thể được gọi là mẫu.

(Bộ biến đổi)

Bộ biến đổi 106 biến đổi các phần dư dự đoán trong miền không gian thành các hệ số biến đổi trong miền tần số, và xuất ra các hệ số biến đổi tới bộ lượng tử hóa 108. Cụ thể hơn, bộ biến đổi 106 áp dụng, ví dụ, biến đổi cosin rời rạc (DCT) hoặc biến đổi sin rời rạc (DST) được xác định tới các phần dư dự đoán trong miền không gian. DCT hoặc DST được xác định có thể được xác định trước.

Cần lưu ý rằng bộ biến đổi 106 có thể lựa chọn thích nghi loại biến đổi từ trong số nhiều loại biến đổi, và biến đổi các phần dư dự đoán thành các hệ số biến đổi bằng cách sử dụng hàm cơ sở biến đổi tương ứng với loại biến đổi được lựa chọn. Loại biến đổi này cũng được gọi là biến đổi đa lõi rõ ràng (EMT-explicit multiple core transform) hoặc đa biến đổi thích nghi (AMT-adaptive multiple transform). Hàm cơ sở biến đổi có thể cũng được gọi là cơ sở.

Các loại biến đổi bao gồm, ví dụ, DCT-II, DCT-V, DCT-VIII, DST-I, và DST-VII. Cần lưu ý rằng các loại biến đổi này có thể được biểu diễn là DCT2, DCT5, DCT8, DST1 và DST7. FIG.14 là sơ đồ chỉ báo các hàm cơ sở biến đổi ví dụ đối với các loại biến đổi ví dụ. Trong FIG.14, N chỉ báo số lượng điểm ảnh đầu vào. Ví dụ, việc lựa chọn loại biến đổi từ trong số nhiều loại biến đổi có thể phụ thuộc vào loại dự đoán (một trong số dự đoán trong và dự đoán liên đới) và có thể phụ thuộc vào chế độ nội dự đoán.

Thông tin mà chỉ báo rằng áp dụng EMT hay AMT (được gọi là, ví dụ, cờ EMT hoặc cờ AMT) và thông tin mà chỉ báo loại biến đổi được lựa chọn thường được báo hiệu tại cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin này không cần được thực hiện tại cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp ảnh, cấp lớp, cấp lát, hoặc cấp CTU).

Ngoài ra, bộ biến đổi 106 có thể biến đổi lại các hệ số biến đổi (mà là các kết quả biến đổi). Việc biến đổi lại này cũng được gọi là biến đổi thứ cấp thích nghi (AST-adaptive secondary transform) hoặc biến đổi thứ cấp không thể tách rời (NSST-non-separable secondary transform). Ví dụ, bộ biến đổi 106 thực hiện việc biến đổi lại trong các đơn vị của khối con (ví dụ, khối con 4×4 điểm ảnh) được chứa trong khối hệ số biến đổi tương ứng với phần dư nội dự đoán. Thông tin mà chỉ báo

rằng có áp dụng NSST hay không và thông tin liên quan đến ma trận biến đổi được sử dụng trong NSST thường được báo hiệu tại cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin này không cần được thực hiện tại cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp ảnh, cấp lớp, cấp lát, hoặc cấp CTU).

Bộ biến đổi 106 có thể sử dụng biến đổi có thể tách rời và biến đổi không thể tách rời. Biến đổi có thể tách rời là phương pháp trong đó việc biến đổi được thực hiện nhiều lần bằng cách thực hiện tách biệt việc biến đổi đối với mỗi số lượng chiều theo số lượng chiều của các đầu vào. Biến đổi không thể tách rời là phương pháp để thực hiện việc biến đổi chung trong đó hai chiều hoặc nhiều hơn trong các đầu vào đa chiều được xem chung là một chiều.

Trong một ví dụ về biến đổi không thể tách rời, khi đầu vào là khối 4×4 điểm ảnh, khối 4×4 điểm ảnh được xem là một mảng bao gồm 16 phần tử, và biến đổi này áp dụng ma trận biến đổi 16×16 tới mảng này.

Trong ví dụ khác của biến đổi không thể tách rời, khối đầu vào 4×4 điểm ảnh được xem là mảng đơn bao gồm 16 phần tử, và sau đó biến đổi (biến đổi siêu lập phương) trong đó phép xoay được thực hiện trên mảng nhiều lần có thể được thực hiện.

Trong biến đổi trong bộ biến đổi 106, các loại biến đổi của các hàm cơ sở biến đổi cần được biến đổi thành các vùng miền tần số theo trong CU có thể được chuyển đổi. Các ví dụ bao gồm biến đổi thay đổi theo không gian (SVT).

FIG.15 là sơ đồ khái niệm để minh họa một ví dụ của SVT.

Trong SVT, như được minh họa trong FIG.15, các CU được chia thành hai vùng bằng nhau theo chiều ngang hoặc chiều dọc, và chỉ một trong số hai vùng được biến đổi thành miền tần số. Loại cơ sở biến đổi có thể được thiết lập đối với mỗi vùng. Ví dụ, DST7 và DST8 được sử dụng. Ví dụ, trong số hai vùng thu được bằng cách phân chia CU theo chiều dọc thành hai vùng bằng nhau, DST7 và DCT8 có thể được sử dụng đối với vùng tại vị trí 0. Ngoài ra, trong số hai vùng, DST7 được sử dụng cho vùng tại vị trí 1. Tương tự, trong số hai vùng thu được bằng cách phân chia CU theo chiều ngang thành hai vùng bằng nhau, DST7 và DCT8 được sử dụng cho vùng tại vị trí 0. Ngoài ra, trong số hai vùng, DST7 được sử dụng cho vùng tại vị trí 1.

1. Mặc dù chỉ một trong số hai vùng trong CU được biến đổi và vùng còn lại không được biến đổi trong ví dụ được minh họa trong FIG.15, mỗi hai vùng có thể được biến đổi. Ngoài ra, phương pháp phân chia có thể bao gồm không chỉ việc phân chia thành hai vùng mà còn việc phân chia thành bốn vùng. Ngoài ra, phương pháp phân chia có thể là linh hoạt hơn. Ví dụ, thông tin mà chỉ báo phương pháp phân chia có thể được mã hóa và có thể được báo hiệu trong cùng cách thức như phân chia CU. Cần lưu ý rằng SVT cũng có thể được gọi là biến đổi khối con (SBT-sub-block transform).

AMT và EMT được mô tả nêu trên có thể được gọi là MTS (multiple transform selection - Lựa chọn đa biến đổi). Khi MTS được áp dụng, loại biến đổi mà là DST7, DCT8, hoặc loại tương tự có thể được lựa chọn, và thông tin mà chỉ báo loại biến đổi được lựa chọn có thể được mã hóa là thông tin chỉ số đối với mỗi CU. Có xử lý khác được gọi là IMTS (MTS ẩn) như là xử lý để lựa chọn loại biến đổi cần được sử dụng đối với biến đổi trực giao được thực hiện mà không mã hóa thông tin chỉ số. Khi IMTS được áp dụng, ví dụ, khi CU có dạng chữ nhật, biến đổi trực giao của dạng chữ nhật có thể được thực hiện nhờ sử dụng DST7 đối với cạnh ngắn và DST2 đối với cạnh dài. Ngoài ra, ví dụ, khi CU có dạng hình vuông, biến đổi trực giao của dạng chữ nhật có thể được thực hiện nhờ sử dụng DCT2 khi MTS có hiệu lực trong chuỗi và nhờ sử dụng DST7 khi MTS không có hiệu lực trong chuỗi. DCT2 và DST7 chỉ là các ví dụ. Các loại biến đổi khác có thể được sử dụng, và cũng có thể thay đổi kết hợp của các loại biến đổi được sử dụng thành kết hợp của các loại biến đổi khác nhau. IMTS có thể được sử dụng chỉ đối với các khối nội dự đoán, hoặc có thể được sử dụng đối với cả các khối nội dự đoán và khối dự đoán liên đới.

Ba xử lý MTS, SBT, và IMTS đã được mô tả nêu trên như là các xử lý lựa chọn để chuyển đổi một cách chọn lọc các loại biến đổi được sử dụng trong biến đổi trực giao. Tuy nhiên, tất cả ba xử lý lựa chọn có thể được sử dụng, hoặc chỉ một phần của các xử lý chọn lọc có thể được sử dụng một cách chọn lọc. Việc một hoặc nhiều trong số các xử lý chọn lọc có được sử dụng hay không có thể được nhận dạng, ví dụ, dựa trên thông tin cờ hoặc loại tương tự trong thông tin tiêu đề như SPS. Ví dụ, khi tất cả ba xử lý lựa chọn là khả dụng để sử dụng, một trong số ba xử lý lựa

chọn được lựa chọn đối với mỗi CU và biến đổi trực giao của CU được thực hiện. Cần lưu ý rằng các xử lý lựa chọn để chuyển đổi một cách chọn lọc các loại biến đổi có thể là các xử lý lựa chọn khác với ba xử lý lựa chọn nêu trên, hoặc mỗi ba xử lý lựa chọn có thể được thay thế bởi xử lý khác. Một cách điển hình, ít nhất một trong số bốn chức năng truyền [1] đến [4] sau đây được thực hiện. Chức năng [1] là chức năng để thực hiện biến đổi trực giao của toàn bộ CU và mã hóa thông tin mà chỉ báo loại biến đổi được sử dụng trong việc biến đổi. Chức năng [2] là chức năng để thực hiện biến đổi trực giao của toàn bộ CU và xác định loại biến đổi dựa trên quy tắc được xác định mà không mã hóa thông tin mà chỉ báo loại biến đổi. Chức năng [3] là chức năng để thực hiện biến đổi trực giao của một phần vùng CU và mã hóa thông tin mà chỉ báo loại biến đổi được sử dụng trong việc biến đổi. Chức năng [4] là chức năng để thực hiện biến đổi trực giao của một phần vùng CU và xác định loại biến đổi dựa trên quy tắc được xác định mà không mã hóa thông tin mà chỉ báo loại biến đổi được sử dụng trong việc biến đổi. Các quy tắc được xác định có thể được xác định trước.

Cần lưu ý rằng việc MTS, IMTS, và/hoặc SBT có được áp dụng hay không có thể được xác định đối với mỗi đơn vị xử lý. Ví dụ, việc MTS, IMTS, và/hoặc SBT có được áp dụng hay không có thể được xác định đối với mỗi chuỗi, ảnh, viên, lát, CTU, hoặc CU.

Cần lưu ý rằng công cụ để chuyển đổi một cách chọn lọc các loại biến đổi trong sáng chế có thể được mô tả là phương pháp để lựa chọn một cách chọn lọc cơ sở để sử dụng trong xử lý biến đổi, xử lý lựa chọn, hoặc xử lý để lựa chọn cơ sở. Ngoài ra, công cụ để chuyển đổi một cách chọn lọc các loại biến đổi có thể được mô tả là chế độ để lựa chọn một cách thích nghi các loại biến đổi.

FIG.16 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi 106, và sẽ được mô tả nhằm thuận tiện có viện dẫn tới FIG.7.

Ví dụ, bộ biến đổi 106 xác định rằng có thực hiện biến đổi trực giao hay không (bước St_1). Ở đây, khi xác định để thực hiện biến đổi trực giao (Có trong bước St_1), bộ biến đổi 106 lựa chọn loại biến đổi để sử dụng trong biến đổi trực giao từ các loại biến đổi (bước St_2). Tiếp theo, bộ biến đổi 106 thực hiện biến đổi

trực giao bằng cách áp dụng loại biến đổi được lựa chọn tới phần dư dự đoán của khối hiện tại (bước St_3). Sau đó, bộ biến đổi 106 xuất ra thông tin mà chỉ báo loại biến đổi được lựa chọn tới bộ mã hóa entropy 110, để cho phép bộ mã hóa entropy 110 mã hóa thông tin (bước St_4). Mặt khác, khi xác định không thực hiện biến đổi trực giao (Không trong bước St_1), bộ biến đổi 106 xuất ra thông tin mà chỉ báo rằng không có biến đổi trực giao được thực hiện, để cho phép bộ mã hóa entropy 110 mã hóa thông tin (bước St_5). Cần lưu ý rằng việc có thực hiện biến đổi trực giao hay không trong bước St_1 có thể được xác định dựa trên, ví dụ, kích cỡ của khối biến đổi, chế độ dự đoán được áp dụng tới CU, v.v. Ngoài ra, biến đổi trực giao có thể được thực hiện nhờ sử dụng loại biến đổi được xác định mà không mã hóa thông tin mà chỉ báo loại biến đổi để sử dụng trong biến đổi trực giao. Loại biến đổi được xác định có thể được xác định trước.

FIG.17 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi 106, và sẽ được mô tả nhằm thuận tiện có viện dẫn tới FIG.7. Cần lưu ý rằng ví dụ được minh họa trong FIG.17 là ví dụ về biến đổi trực giao trong trường hợp trong đó các loại biến đổi để sử dụng trong biến đổi trực giao được chuyển đổi một cách chọn lọc như trong trường hợp của ví dụ được minh họa trong FIG.16.

Theo một ví dụ, nhóm loại biến đổi thứ nhất có thể bao gồm DCT2, DST7, và DCT8. Theo ví dụ khác, nhóm loại biến đổi thứ hai có thể bao gồm DCT2. Các loại biến đổi được chứa trong nhóm loại biến đổi thứ nhất và các loại biến đổi được chứa trong nhóm loại biến đổi thứ hai có thể chồng lấn một phần với nhau, hoặc có thể hoàn toàn khác nhau.

Bộ biến đổi 106 xác định rằng kích cỡ biến đổi có nhỏ hơn hoặc bằng giá trị định trước hay không (bước Su_1). Ở đây, khi xác định rằng kích cỡ biến đổi nhỏ hơn hoặc bằng giá trị được xác định (Có trong bước Su_1), bộ biến đổi 106 thực hiện biến đổi trực giao của phần dư dự đoán của khối hiện tại nhờ sử dụng loại biến đổi được chứa trong nhóm loại biến đổi thứ nhất (bước Su_2). Tiếp theo, bộ biến đổi 106 xuất ra thông tin mà chỉ báo loại biến đổi cần được sử dụng trong số ít nhất một loại biến đổi được chứa trong nhóm loại biến đổi thứ nhất tới bộ mã hóa entropy 110, để cho phép bộ mã hóa entropy 110 mã hóa thông tin (bước Su_3). Mặt khác, khi

xác định rằng kích cỡ biến đổi không nhỏ hơn hoặc bằng giá trị định trước (Không trong bước Su_1), bộ biến đổi 106 thực hiện biến đổi trực giao của phần dư dự đoán của khối hiện tại nhờ sử dụng nhóm loại biến đổi thứ hai (bước Su_4). Giá trị được xác định có thể là giá trị ngưỡng, và có thể là giá trị định trước.

Trong bước Su_3, thông tin mà chỉ báo loại biến đổi để sử dụng trong biến đổi trực giao có thể là thông tin mà chỉ báo kết hợp của loại biến đổi cần được áp dụng theo chiều dọc trong khối hiện tại và loại biến đổi cần được áp dụng theo chiều ngang trong khối hiện tại. Nhóm loại thứ nhất có thể bao gồm chỉ một loại biến đổi, và thông tin mà chỉ báo loại biến đổi để sử dụng trong biến đổi trực giao có thể không được mã hóa. Nhóm loại biến đổi thứ hai có thể bao gồm nhiều loại biến đổi, và thông tin mà chỉ báo loại biến đổi để sử dụng trong biến đổi trực giao trong số một hoặc nhiều loại biến đổi được chứa trong nhóm loại biến đổi thứ hai có thể được mã hóa.

Ngoài ra, loại biến đổi có thể được chỉ báo dựa trên kích cỡ biến đổi thông tin chỉ báo loại biến đổi. Cần lưu ý rằng các việc xác định này không bị giới hạn ở việc xác định đối với việc kích cỡ biến đổi có nhỏ hơn hoặc bằng giá trị được xác định hay không, và các xử lý khác cũng có thể khả dụng để xác định loại biến đổi để sử dụng trong biến đổi trực giao dựa trên kích cỡ biến đổi.

(Bộ lượng tử hóa)

Bộ lượng tử hóa 108 lượng tử hóa các hệ số biến đổi được xuất ra từ bộ biến đổi 106. Cụ thể hơn, bộ lượng tử hóa 108 quét, theo thứ tự quét được xác định, các hệ số biến đổi của khối hiện tại, và lượng tử hóa các hệ số biến đổi được quét dựa trên các tham số lượng tử hóa (QP-quantization parameter) tương ứng với các hệ số biến đổi. Bộ lượng tử hóa 108 sau đó xuất các hệ số biến đổi được lượng tử hóa (sau đây cũng được gọi là các hệ số được lượng tử hóa) của khối hiện tại tới bộ mã hóa entropy 110 và bộ lượng tử hóa ngược 112. Thứ tự quét được xác định có thể được xác định trước.

Thứ tự quét được xác định là thứ tự để lượng tử hóa/lượng tử hóa ngược các hệ số biến đổi. Ví dụ, thứ tự quét được xác định có thể được định nghĩa là thứ tự

tăng dần của tần số (từ tần số thấp đến cao) hoặc thứ tự giảm dần của tần số (từ tần số cao đến thấp).

Tham số lượng tử hóa (QP) là tham số mà xác định bước lượng tử hóa (độ rộng lượng tử hóa). Ví dụ, khi giá trị của tham số lượng tử hóa tăng lên, bước lượng tử hóa cũng tăng lên. Nói cách khác, khi giá trị của tham số lượng tử hóa tăng lên, sai số trong các hệ số lượng tử (sai số lượng tử hóa) tăng lên.

Ngoài ra, ma trận lượng tử hóa có thể được sử dụng cho việc lượng tử hóa. Ví dụ, một vài loại ma trận lượng tử hóa có thể được sử dụng một cách tương ứng tới các kích cỡ biến đổi tần số như 4×4 và 8×8 , các chế độ dự đoán như việc nội dự đoán và việc dự đoán liên đới, và các thành phần điểm ảnh như các thành phần điểm ảnh độ chói và sắc độ. Cần lưu ý rằng việc lượng tử có nghĩa là số hóa các giá trị được lấy mẫu tại các khoảng cách được xác định một cách tương ứng thành các mức được xác định. Trong lĩnh vực kỹ thuật này, việc lượng tử hóa có thể được viện dẫn nhờ sử dụng các biểu diễn khác, như làm tròn và biến đổi tỷ lệ, và có thể áp dụng làm tròn và biến đổi tỷ lệ. Các khoảng cách được xác định và các mức được xác định có thể được xác định trước.

Các phương pháp nhờ sử dụng các ma trận lượng tử hóa có thể bao gồm phương pháp mà sử dụng ma trận lượng tử hóa mà đã được thiết lập một cách trực tiếp tại phía bộ mã hóa 100, và phương pháp mà sử dụng ma trận lượng tử hóa mà đã được thiết lập là mặc định (ma trận mặc định). Tại phía bộ mã hóa 100, ma trận lượng tử hóa thích hợp đối với các đặc điểm của ảnh có thể được thiết lập bằng cách thiết lập trực tiếp ma trận lượng tử hóa. Tuy nhiên, trường hợp này có thể có nhược điểm là làm tăng lượng mã để mã hóa ma trận lượng tử hóa. Cần lưu ý rằng ma trận lượng tử hóa cần được sử dụng để lượng tử hóa khối hiện tại có thể được tạo ra dựa trên ma trận lượng tử hóa mặc định hoặc ma trận lượng tử hóa được mã hóa, thay vì sử dụng trực tiếp ma trận lượng tử hóa mặc định hoặc ma trận lượng tử hóa được mã hóa.

Có phương pháp để lượng tử hóa hệ số tần số cao và hệ số tần số thấp mà không sử dụng ma trận lượng tử hóa. Cần lưu ý rằng phương pháp này có thể được

xem là tương đương với phương pháp sử dụng ma trận lượng tử hóa (ma trận phẳng) mà các hệ số của nó có cùng giá trị.

Ma trận lượng tử hóa có thể được mã hóa, ví dụ, tại mức chuỗi, mức ảnh, mức lát, mức viên, hoặc mức CTU. Ma trận lượng tử hóa có thể được chỉ rõ nhờ sử dụng, ví dụ, tập hợp tham số chuỗi (SPS) hoặc tập hợp tham số ảnh (PPS). SPS bao gồm tham số mà được sử dụng đối với chuỗi, và PPS bao gồm tham số mà được sử dụng đối với ảnh. Mỗi SPS và PPS có thể được gọi đơn giản là tập hợp tham số.

Khi sử dụng ma trận lượng tử hóa, bộ lượng tử hóa 108 biến đổi tỷ lệ, đối với mỗi hệ số biến đổi, ví dụ độ rộng lượng tử hóa mà có thể được tính toán dựa trên tham số lượng tử hóa, v.v., nhờ sử dụng giá trị của ma trận lượng tử hóa. Xử lý lượng tử hóa được thực hiện mà không sử dụng ma trận lượng tử hóa có thể là xử lý để lượng tử hóa các hệ số biến đổi dựa trên độ rộng lượng tử hóa được tính toán dựa trên tham số lượng tử hóa, v.v. Cần lưu ý rằng, trong xử lý lượng tử hóa được thực hiện mà không sử dụng bất kỳ ma trận lượng tử hóa, độ rộng lượng tử hóa có thể được nhân với giá trị được xác định mà là chung đối với tất cả các hệ số biến đổi trong khối. Giá trị được xác định có thể được xác định trước.

FIG.18 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lượng tử hóa theo phương án của sáng chế. Ví dụ, bộ lượng tử hóa 108 bao gồm bộ tạo tham số lượng tử hóa chênh lệch 108a, bộ tạo tham số lượng tử hóa được dự đoán 108b, bộ tạo tham số lượng tử hóa 108c, bộ lưu trữ tham số lượng tử hóa 108d, và bộ thực thi lượng tử hóa 108e.

FIG.19 là lưu đồ minh họa một ví dụ về xử lý lượng tử hóa được thực hiện bởi bộ lượng tử hóa 108, và sẽ được mô tả nhằm thuận tiện có viện dẫn tới các Fig.7 và Fig.18.

Theo một ví dụ, bộ lượng tử hóa 108 có thể thực hiện việc lượng tử hóa đối với mỗi CU dựa trên lưu đồ được minh họa trong FIG.19. Cụ thể hơn, bộ tạo tham số lượng tử hóa 108c xác định rằng có thực hiện việc lượng tử hóa hay không (bước Sv_1). Ở đây, khi xác định để thực hiện việc lượng tử hóa (Có trong bước Sv_1), bộ tạo tham số lượng tử hóa 108c tạo ra tham số lượng tử hóa đối với khối hiện tại (bước

Sv_2), và lưu trữ tham số lượng tử hóa tới bộ lưu trữ tham số lượng tử hóa 108d (bước Sv_3).

Tiếp theo, bộ thực thi lượng tử hóa 108e lượng tử hóa các hệ số biến đổi của khối hiện tại nhờ sử dụng tham số lượng tử hóa được tạo ra trong bước Sv_2 (bước Sv_4). Sau đó, bộ tạo tham số lượng tử hóa được dự đoán 108b thu nhận tham số lượng tử hóa đối với đơn xử lý khác với khối hiện tại từ bộ lưu trữ tham số lượng tử hóa 108d (bước Sv_5). Bộ tạo tham số lượng tử hóa được dự đoán 108b tạo ra tham số lượng tử hóa được dự đoán của khối hiện tại dựa trên tham số lượng tử hóa thu được (bước Sv_6). Bộ tạo tham số lượng tử hóa chênh lệch 108a tính toán độ chênh lệch giữa tham số lượng tử hóa của khối hiện tại được tạo ra bởi bộ tạo tham số lượng tử hóa 108c và tham số lượng tử hóa được dự đoán của khối hiện tại được tạo ra bởi bộ tạo tham số lượng tử hóa được dự đoán 108b (bước Sv_7). Tham số lượng tử hóa chênh lệch có thể được tạo ra bằng cách tính toán độ chênh lệch. Bộ tạo tham số lượng tử hóa chênh lệch 108a xuất ra tham số lượng tử hóa chênh lệch tới bộ mã hóa entropy 110, để cho phép bộ mã hóa entropy 110 mã hóa tham số lượng tử hóa chênh lệch (bước Sv_8).

Cần lưu ý rằng tham số lượng tử hóa chênh lệch có thể được mã hóa, ví dụ, tại mức chuỗi, mức ảnh, mức lát, mức viên, hoặc mức CTU. Ngoài ra, giá trị khởi tạo của tham số lượng tử hóa có thể được mã hóa tại mức chuỗi, mức ảnh, mức lát, mức viên, hoặc mức CTU. Tại lúc khởi tạo, tham số lượng tử hóa có thể được tạo ra nhờ sử dụng giá trị khởi tạo của tham số lượng tử hóa và tham số lượng tử hóa chênh lệch.

Cần lưu ý rằng bộ lượng tử hóa 108 có thể bao gồm nhiều bộ lượng tử hóa, và có thể áp dụng việc lượng tử hóa phụ thuộc trong đó các hệ số biến đổi được lượng tử hóa nhờ sử dụng phương pháp lượng tử hóa được lựa chọn từ nhiều phương pháp lượng tử hóa.

(Bộ mã hóa entropy)

FIG.20 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ mã hóa entropy 110 theo phương án của sáng chế, và sẽ được mô tả nhằm thuận tiện có viện dẫn tới FIG.7. Bộ mã hóa entropy 110 tạo ra dòng bằng cách mã hóa entropy

các hệ số được lượng tử hóa được đưa vào từ bộ lượng tử hóa 108 và tham số dự đoán được đưa vào từ bộ tạo tham số dự đoán 130. Ví dụ, việc mã hóa số học nhị phân thích nghi dựa trên ngữ cảnh (CABAC-context-based adaptive binary arithmetic coding) được sử dụng như là mã hóa entropy. Cụ thể hơn, bộ mã hóa entropy 110 như được minh họa bao gồm bộ nhị phân hóa 110a, bộ điều khiển ngữ cảnh 110b, và bộ mã hóa số học nhị phân 110c. Bộ nhị phân hóa 110a thực hiện việc nhị phân hóa trong đó các tín hiệu đa mức như các hệ số được lượng tử hóa và tham số dự đoán được biến đổi thành các tín hiệu nhị phân. Các ví dụ của các phương pháp nhị phân hóa bao gồm nhị phân hóa Rice cắt, các mã Golomb theo số mũ, và nhị phân hóa độ dài cố định. Bộ điều khiển ngữ cảnh 110b thu nhận giá trị ngữ cảnh theo đặc tính hoặc trạng thái xung quanh của phần tử cú pháp, mà là xác suất xuất hiện của tín hiệu nhị phân. Các ví dụ của các phương pháp để thu nhận giá trị ngữ cảnh bao gồm đường vòng, mà viện dẫn tới phần tử cú pháp, viện dẫn tới các khối liền kề phía trên và bên trái, viện dẫn tới thông tin phân cấp, v.v. bộ mã hóa số học nhị phân 110c mã hóa số học tín hiệu nhị phân nhờ sử dụng ngữ cảnh thu được.

FIG.21 là sơ đồ khái niệm để minh họa dòng xử lý CABAC ví dụ trong bộ mã hóa entropy 110. Đầu tiên, việc khởi tạo được thực hiện trong CABAC trong bộ mã hóa entropy 110. Trong việc khởi tạo, việc khởi tạo trong bộ mã hóa số học nhị phân 110c và thiết lập của giá trị ngữ cảnh khởi tạo được thực hiện. Ví dụ, bộ nhị phân hóa 110a và bộ mã hóa số học nhị phân 110c có thể thực thi việc nhị phân hóa và mã hóa số học của các hệ số lượng tử hóa trong CTU một cách tuần tự. Bộ điều khiển ngữ cảnh 110b có thể cập nhật giá trị ngữ cảnh mỗi lần việc mã hóa số học được thực hiện. Sau đó, bộ điều khiển ngữ cảnh 110b có thể lưu giá trị ngữ cảnh như là hậu xử lý. Giá trị ngữ cảnh được lưu có thể được sử dụng, ví dụ, để khởi tạo giá trị ngữ cảnh đối với CTU tiếp theo.

(Bộ lượng tử hóa ngược)

Bộ lượng tử hóa ngược 112 lượng tử hóa ngược các hệ số được lượng tử hóa mà đã được đưa vào từ bộ lượng tử hóa 108. Cụ thể hơn, bộ lượng tử hóa ngược 112 lượng tử hóa ngược, theo thứ tự quét được xác định, các hệ số được lượng tử hóa của khối hiện tại. Bộ lượng tử hóa ngược 112 sau đó xuất các hệ số biến đổi được lượng

tử hóa ngược của khối hiện tại tới bộ biến đổi ngược 114. Thứ tự quét được xác định có thể được xác định trước.

(Bộ biến đổi ngược)

Bộ biến đổi ngược 114 khôi phục các phần dư dự đoán bằng cách biến đổi ngược các hệ số biến đổi, mà được đưa vào từ bộ lượng tử hóa ngược 112. Cụ thể hơn, bộ biến đổi ngược 114 lưu trữ các phần dư dự đoán của khối hiện tại bằng cách thực hiện biến đổi ngược tương ứng với biến đổi được áp dụng tới các hệ số biến đổi bởi bộ biến đổi 106. Bộ biến đổi ngược 114 sau đó xuất các phần dư dự đoán được khôi phục tới bộ cộng 116.

Cần lưu ý rằng do thông tin thường bị mất trong khi lượng tử hóa, các phần dư dự đoán được khôi phục không khớp với các phần dư dự đoán được tính toán bởi bộ trừ 104. Nói cách khác, các phần dư dự đoán được khôi phục thường bao gồm các sai số lượng tử hóa.

(Bộ cộng)

Bộ cộng 116 khôi phục khối hiện tại bằng cách cộng các phần dư dự đoán mà được đưa vào từ bộ biến đổi ngược 114 và các ảnh dự đoán mà được đưa vào từ bộ điều khiển dự đoán 128. Do đó, ảnh được khôi phục được tạo ra. Bộ cộng 116 sau đó xuất ảnh được khôi phục tới bộ nhớ khối 118 và bộ lọc vòng 120. Khối được khôi phục cũng có thể được gọi là khối được giải mã cục bộ.

(Bộ nhớ khối)

Bộ nhớ khối 118 là bộ lưu trữ để lưu trữ các khối trong ảnh hiện tại, ví dụ, để sử dụng trong việc nội dự đoán. Cụ thể hơn, bộ nhớ khối 118 lưu trữ các ảnh được khôi phục được xuất ra từ bộ cộng 116.

(Bộ nhớ khung)

Bộ nhớ khung 122, ví dụ, là bộ lưu trữ để lưu trữ các ảnh tham chiếu được sử dụng trong dự đoán liên đới, và cũng được gọi là bộ đệm khung. Cụ thể hơn, bộ nhớ khung 122 lưu trữ các ảnh được khôi phục được lọc bởi bộ lọc vòng 120.

(Bộ lọc vòng)

Bộ lọc vòng 120 áp dụng các lọc vòng tới ảnh được khôi phục được xuất ra bởi bộ cộng 116, và xuất ra ảnh được khôi phục được lọc tới bộ nhớ khung 122. Bộ lọc vòng là bộ lọc được sử dụng trong vòng mã hóa (bộ lọc nội vòng). Các ví dụ của các bộ lọc vòng bao gồm, ví dụ, bộ lọc vòng thích nghi (ALF-adaptive loop filter), bộ lọc giải khối (DB hoặc DBF), bộ lọc dịch thích nghi mẫu (SAO-sample adaptive offset), v.v.

FIG.22 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lọc vòng 120 theo phương án của sáng chế. Ví dụ, như được minh họa trong FIG.22, bộ lọc vòng 120 bao gồm bộ thực thi lọc giải khối 120a, bộ thực thi SAO 120b, và bộ thực thi ALF 120c. Bộ thực thi lọc giải khối 120a thực hiện xử lý lọc giải khối trên ảnh được khôi phục. Bộ thực thi SAO 120b thực hiện xử lý SAO trên ảnh được khôi phục sau khi được đưa vào xử lý lọc giải khối. Bộ thực thi ALF 120c thực hiện xử lý ALF trên ảnh được khôi phục sau khi được đưa vào xử lý SAO. Xử lý ALF và giải khối được mô tả chi tiết sau đây. Xử lý SAO là xử lý để nâng cao chất lượng ảnh bằng cách làm giảm độ rung (hiện tượng trong đó các giá trị điểm ảnh bị méo như các sóng xung quanh biên) và hiệu chỉnh độ lệch trong giá trị điểm ảnh. Các ví dụ của xử lý SAO bao gồm xử lý dịch biên và xử lý dịch băng. Cần lưu ý rằng bộ lọc vòng 120, trong một vài phương án, có thể không bao gồm tất cả các thành phần cấu thành được bộc lộ trong FIG.22, và có thể bao gồm một vài thành phần cấu thành, và có thể bao gồm các thành phần bổ sung. Ngoài ra, bộ lọc vòng 120 có thể có cấu trúc để thực hiện các xử lý nêu trên trong thứ tự xử lý khác với từ thứ tự được bộc lộ trong FIG.22, có thể không thực hiện tất cả các xử lý này, v.v.

(Bộ lọc vòng > Bộ lọc vòng thích nghi)

Trong ALF, bộ lọc sai số bình phương tối thiểu để loại bỏ các méo nén được áp dụng. Ví dụ, một bộ lọc được lựa chọn từ nhiều bộ lọc dựa trên chiều và hoạt động của các gradien cục bộ được áp dụng đối với mỗi khối con 2×2 điểm ảnh trong khối hiện tại.

Cụ thể hơn, đầu tiên, mỗi khối con (ví dụ, mỗi khối con 2×2 điểm ảnh) được phân loại thành một trong số nhiều lớp (ví dụ, 15 hoặc 25 lớp). Việc phân loại khối con có thể được dựa trên, ví dụ, chiều và độ hoạt động gradien. Theo ví dụ của sáng

chế, chỉ số hạng mục C (ví dụ, $C = 5D + A$) được tính toán hoặc được xác định dựa trên chiều gradient D (ví dụ, 0 đến 2 hoặc 0 đến 4) và độ hoạt động gradient A (ví dụ, 0 đến 4). Sau đó, dựa trên chỉ số phân loại C, mỗi khối con được phân loại thành một trong số nhiều lớp.

Ví dụ, chiều gradient D được tính toán bằng cách so sánh các gradient của các chiều (ví dụ, chiều ngang, chiều dọc, và hai chiều đường chéo). Ngoài ra, ví dụ, độ hoạt động gradient A được tính toán bằng cách cộng các gradient của các chiều và lượng tử hóa kết quả cộng này.

Bộ lọc cần được sử dụng cho mỗi khối con có thể được xác định từ trong số nhiều bộ lọc dựa trên kết quả của việc phân loại này.

Dạng bộ lọc cần được sử dụng trong ALF, ví dụ, là dạng bộ lọc đối xứng vòng tròn. FIG.23A đến FIG.23C là các sơ đồ khái niệm để minh họa các ví dụ của các dạng lọc được sử dụng trong các ALF. FIG.23A minh họa bộ lọc dạng kim cương 5×5 , FIG.23B minh họa bộ lọc dạng kim cương 7×7 , và FIG.23C minh họa bộ lọc dạng kim cương 9×9 . Thông tin mà chỉ báo dạng bộ lọc thường được báo hiệu tại cấp ảnh. Lưu ý rằng việc báo hiệu thông tin này mà chỉ báo dạng bộ lọc không cần được thực hiện tại cấp ảnh, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp lớp, cấp lát, cấp CTU, hoặc cấp CU).

Việc cho phép hoặc không cho phép ALF có thể được xác định tại cấp ảnh hoặc cấp CU. Ví dụ, việc quyết định có áp dụng hay không ALF tới độ chói có thể được thực hiện tại cấp CU, và việc quyết định có áp dụng hay không ALF tới sắc độ có thể được thực hiện tại cấp ảnh. Thông tin mà chỉ báo việc cho phép hoặc không cho phép của ALF thường được báo hiệu tại cấp ảnh hoặc cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin mà chỉ báo việc cho phép hoặc không cho phép ALF không cần thiết được thực hiện tại cấp ảnh hoặc cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp lát, cấp ô, hoặc cấp CTU).

Ngoài ra, như được mô tả nêu trên, một bộ lọc được lựa chọn từ nhiều bộ lọc, và xử lý ALF của khối con được thực hiện. Tập hợp của các hệ số cần được sử dụng đối với mỗi bộ lọc (ví dụ, tối đa tới bộ lọc thứ mười lăm hoặc hai mươi lăm) thường được báo hiệu tại mức ảnh. Cần lưu ý rằng việc báo hiệu của tập hợp hệ số được

thiết lập không cần được thực hiện tại cấp ảnh, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp lớp, cấp lát, cấp CTU, cấp CU, hoặc cấp khối con).

(Bộ lọc vòng > Bộ lọc vòng thích nghi thành phần chéo)

FIG.23D là sơ đồ khái niệm để minh họa dòng ví dụ của ALF thành phần chéo (CC-ALF). FIG.23E là sơ đồ khái niệm để minh họa ví dụ về dạng bộ lọc được sử dụng trong CC-ALF, như CC-ALF của FIG.23D. Ví dụ CC-ALF của các FIG.23D và FIG.23E hoạt động bằng cách áp dụng bộ lọc dạng kim cương tuyến tính tới kênh độ chói đối với mỗi thành phần sắc độ. Các hệ số lọc, ví dụ, có thể được truyền trong APS, được biến đổi tỷ lệ bởi hệ số 2^{10} , và được làm tròn đối với biểu diễn điểm cố định. Ví dụ, trong FIG.23D, Y mẫu (thành phần thứ nhất) được sử dụng cho CCALF đối với Cb và CCALF đối với Cr (các thành phần khác với thành phần thứ nhất).

Việc áp dụng các bộ lọc có thể được điều khiển trên kích cỡ khối biến thiên và được báo hiệu bởi cờ được mã hóa ngữ cảnh thu được đối với mỗi khối của các mẫu. Kích cỡ khối cùng với cờ cho phép CC-ALF có thể được thu tại mức lát đối với mỗi thành phần sắc độ. CC-ALF có thể hỗ trợ các kích cỡ khối khác nhau, ví dụ (trong các mẫu sắc độ) 16x16 điểm ảnh, 32x32 điểm ảnh, 64x64 điểm ảnh, 128x128 điểm ảnh.

(Bộ lọc vòng > Bộ lọc vòng thích nghi thành phần chéo sắc độ liên kết)

Một ví dụ về CCALF sắc độ liên kết, được minh họa trong các Fig.23F và Fig.23G. FIG.23F là sơ đồ khái niệm để minh họa dòng xử lý ví dụ của CCALF sắc độ liên kết. FIG.23G là Bảng minh họa các ứng viên chỉ số trọng số ví dụ. Như được minh họa, một bộ lọc CCALF được sử dụng để tạo ra một đầu ra được lọc CCALF như là tín hiệu tinh chỉnh sắc độ đối với một thành phần màu, trong khi phiên bản có trọng số của cùng tín hiệu tinh chỉnh sắc độ được áp dụng tới thành phần màu khác. Theo cách này, độ phức tạp của CCALF hiện tại được làm giảm xuống khoảng một nửa. Giá trị trọng số có thể được mã hóa thành cờ dấu và chỉ số trọng số. Chỉ số trọng số (được ký hiệu là `weight_index`) có thể được mã hóa thành 3 bit, và chỉ rõ độ lớn của trọng số JC-CCALF `JcCcWeight`, mà là độ lớn không phải 0. Độ lớn của `JcCcWeight` có thể, ví dụ, được xác định như sau:

Nếu `weight_index` nhỏ hơn hoặc bằng 4, `JcCcWeight` bằng `weight_index >> 2`;

Nếu không phải, `JcCcWeight` bằng $4 / (\text{weight_index} - 4)$.

Việc điều khiển bật/tắt mức khối của việc lọc ALF đối với `Cb` và `Cr` có thể được tách biệt. Điều này là tương tự trong CCALF, và hai tập hợp riêng biệt của các cờ điều khiển bật/tắt mức khối có thể được mã hóa. Khác với CCALF, ở đây, các kích cỡ khối điều khiển bật/tắt `Cb`, `Cr` là giống nhau, và do đó, chỉ một kích cỡ khối biến thiên có thể được mã hóa.

(Bộ lọc vòng > Bộ lọc giải khối)

Trong xử lý lọc giải khối, bộ lọc vòng 120 thực hiện xử lý lọc trên biên khối trong ảnh được khôi phục để làm giảm độ méo mà xảy ra tại biên khối.

FIG.24 là sơ đồ khối minh họa một ví dụ về cấu hình cụ thể của bộ thực thi lọc giải khối 120a của bộ lọc vòng 120 (xem các Fig.7 và Fig.22) mà thực hiện chức năng là bộ lọc giải khối.

Bộ thực thi lọc giải khối 120a bao gồm: bộ xác định biên 1201; bộ xác định lọc 1203; bộ thực thi lọc 1205; bộ xác định xử lý 1208; bộ xác định đặc tính lọc 1207; và các bộ chuyển đổi 1202, 1204, và 1206.

Bộ xác định biên 1201 xác định rằng điểm ảnh cần được lọc giải khối (tức là, điểm ảnh hiện tại) được hiện diện xung quanh biên khối. Sau đó, bộ xác định biên 1201 xuất ra kết quả xác định tới bộ chuyển đổi 1202 và bộ xác định xử lý 1208.

Trong trường hợp trong đó bộ xác định biên 1201 đã xác định rằng điểm ảnh hiện tại được hiện diện xung quanh biên khối, bộ chuyển đổi 1202 xuất ra ảnh không được lọc tới bộ chuyển đổi 1204. Trong trường hợp ngược lại trong đó bộ xác định biên 1201 đã xác định rằng không có điểm ảnh hiện tại được hiện diện xung quanh biên khối, bộ chuyển đổi 1202 xuất ra ảnh không được lọc tới bộ chuyển đổi 1206. Cần lưu ý rằng ảnh không được lọc là ảnh được cấu hình với điểm ảnh hiện tại và ít nhất một điểm ảnh xung quanh được bố trí xung quanh điểm ảnh hiện tại.

Bộ xác định lọc 1203 xác định rằng có thực hiện việc lọc giải khối của điểm ảnh hiện tại hay không, dựa trên giá trị điểm ảnh của ít nhất một điểm ảnh xung

quanh được bố trí xung quanh điểm ảnh hiện tại. Sau đó, bộ xác định lọc 1203 xuất ra kết quả xác định tới bộ chuyển đổi 1204 và bộ xác định xử lý 1208.

Trong trường hợp trong đó bộ xác định lọc 1203 đã xác định để thực hiện việc lọc giải khối của điểm ảnh hiện tại, bộ chuyển đổi 1204 xuất ra ảnh không được lọc thu được thông qua bộ chuyển đổi 1202 tới bộ thực thi lọc 1205. Trong trường hợp ngược lại trong đó bộ xác định lọc 1203 đã xác định không thực hiện việc lọc giải khối của điểm ảnh hiện tại, bộ chuyển đổi 1204 xuất ra ảnh không được lọc thu được thông qua bộ chuyển đổi 1202 tới bộ chuyển đổi 1206.

Khi thu nhận ảnh không được lọc thông qua các bộ chuyển đổi 1202 và 1204, bộ thực thi lọc 1205 thực thi, đối với điểm ảnh hiện tại, việc lọc giải khối với đặc tính lọc được xác định bởi bộ xác định đặc tính lọc 1207. Sau đó, bộ thực thi lọc 1205 xuất ra điểm ảnh được lọc tới bộ chuyển đổi 1206.

Dưới sự điều khiển của bộ xác định xử lý 1208, bộ chuyển đổi 1206 xuất ra một cách chọn lọc một trong số điểm ảnh mà không được lọc giải khối và điểm ảnh mà được lọc giải khối bởi bộ thực thi lọc 1205.

Bộ xác định xử lý 1208 điều khiển bộ chuyển đổi 1206 dựa trên các kết quả của việc xác định được thực hiện bởi bộ xác định biên 1201 và bộ xác định lọc 1203. Nói cách khác, bộ xác định xử lý 1208 làm cho bộ chuyển đổi 1206 xuất ra điểm ảnh mà được lọc giải khối khi bộ xác định biên 1201 xác định rằng điểm ảnh hiện tại được hiện diện xung quanh biên khối và khi bộ xác định lọc 1203 xác định để thực hiện việc lọc giải khối của điểm ảnh hiện tại. Ngoài ra, ngoài trường hợp nêu trên, bộ xác định xử lý 1208 làm cho bộ chuyển đổi 1206 xuất ra điểm ảnh mà không được lọc giải khối. Ảnh được lọc được xuất ra từ bộ chuyển đổi 1206 bằng cách lặp lại đầu ra của điểm ảnh theo cách này. Cần lưu ý rằng cấu hình được minh họa trong FIG.24 là một ví dụ của cấu hình trong bộ thực thi lọc giải khối 120a. Bộ thực thi lọc giải khối 120a có thể có các cấu hình khác.

FIG.25 là sơ đồ khái niệm để minh họa ví dụ về bộ lọc giải khối mà có đặc tính lọc đối xứng đối với biên khối.

Trong xử lý lọc giải khối, một trong số hai bộ lọc giải khối mà có các đặc tính khác nhau, tức là, bộ lọc mạnh và bộ lọc yếu, có thể được lựa chọn nhờ sử dụng các

giá trị điểm ảnh và các tham số lượng tử hóa. Trong trường hợp của bộ lọc mạnh, khi các điểm ảnh p_0 đến p_2 và các điểm ảnh q_0 đến q_2 được hiện diện đi qua biên khối như được minh họa trong FIG.25, các giá trị điểm ảnh của điểm ảnh tương ứng q_0 đến q_2 được thay đổi thành các giá trị điểm ảnh q'_0 đến q'_2 bằng cách thực hiện, ví dụ, các tính toán theo biểu thức dưới đây.

$$q'_0 = (p_1 + 2 \times p_0 + 2 \times q_0 + 2 \times q_1 + q_2 + 4) / 8$$

$$q'_1 = (p_0 + q_0 + q_1 + q_2 + 2) / 4$$

$$q'_2 = (p_0 + q_0 + q_1 + 3 \times q_2 + 2 \times q_3 + 4) / 8$$

Cần lưu ý rằng, trong các biểu thức nêu trên, p_0 đến p_2 và q_0 đến q_2 là các giá trị điểm ảnh của các điểm ảnh tương ứng p_0 đến p_2 và các điểm ảnh q_0 đến q_2 . Ngoài ra, q_3 là giá trị điểm ảnh của điểm ảnh lân cận q_3 được bố trí tại phía đối diện của điểm ảnh q_2 so với biên khối. Ngoài ra, trong vế phải của mỗi biểu thức, các hệ số mà được nhân với các giá trị điểm ảnh tương ứng của các điểm ảnh cần được sử dụng để lọc giải khối là các hệ số lọc.

Ngoài ra, trong việc lọc giải khối, việc cắt có thể được thực hiện sao cho các giá trị điểm ảnh được tính toán không được thay đổi lớn hơn giá trị ngưỡng. Ví dụ, trong xử lý cắt của các giá trị điểm ảnh được tính toán theo các biểu thức nêu trên có thể được cắt thành giá trị thu được theo “giá trị điểm ảnh tính toán $\pm 2 \times$ giá trị ngưỡng” nhờ sử dụng giá trị ngưỡng được xác định dựa trên tham số lượng tử hóa. Theo cách này, có thể ngăn ngừa việc san bằng quá mức.

FIG.26 là sơ đồ khái niệm để minh họa biên khối mà trên đó xử lý lọc giải khối được thực hiện. FIG.27 là sơ đồ khái niệm để minh họa các ví dụ về các giá trị cường độ biên (Bs-Boundary strength).

Biên khối mà trên đó xử lý lọc giải khối được thực hiện là, ví dụ, biên giữa các CU, Pu, hoặc TU mà có các khối 8×8 điểm ảnh như được minh họa trong FIG.26. Xử lý lọc giải khối có thể được thực hiện, ví dụ, trong các đơn vị của bốn hàng hoặc bốn cột. Đầu tiên, các giá trị cường độ biên (Bs) được xác định như được chỉ báo trong FIG.27 đối với khối P và khối Q được minh họa trong FIG.26.

Theo các giá trị Bs trong FIG.27, việc có thực hiện hay không các xử lý lọc giải khối của các biên khối mà thuộc về cùng ảnh sử dụng các cường độ khác nhau có thể được xác định. Xử lý lọc giải khối đối với tín hiệu sắc độ được thực hiện khi giá trị Bs là 2. Xử lý lọc giải khối đối với tín hiệu độ chói được thực hiện khi giá trị Bs là 1 hoặc lớn hơn và điều kiện được xác định được thỏa mãn. Điều kiện được xác định có thể được xác định trước. Cần lưu ý rằng các điều kiện để xác định các giá trị Bs không bị giới hạn ở những gì được thể hiện trong FIG.27, và giá trị Bs có thể được xác định dựa trên tham số khác.

(Bộ dự đoán (bộ nội dự đoán, bộ dự đoán liên đới, bộ điều khiển dự đoán))

FIG.28 là lưu đồ minh họa một ví dụ của xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa 100. Cần lưu ý rằng bộ dự đoán bao gồm tất cả hoặc một phần của các thành phần cấu thành sau đây: bộ nội dự đoán 124; bộ dự đoán liên đới 126; và bộ điều khiển dự đoán 128. Bộ thực thi dự đoán bao gồm, ví dụ, bộ nội dự đoán 124 và bộ dự đoán liên đới 126.

Bộ dự đoán tạo ra ảnh dự đoán của khối hiện tại (bước Sb_1). Ảnh dự đoán này có thể cũng được gọi là tín hiệu dự đoán hoặc khối dự đoán. Cần lưu ý rằng tín hiệu dự đoán là, ví dụ, ảnh nội dự đoán (tín hiệu dự đoán ảnh) hoặc ảnh dự đoán liên đới (tín hiệu dự đoán liên đới). Bộ dự đoán tạo ra ảnh dự đoán của khối hiện tại nhờ sử dụng ảnh được khôi phục mà đã thu được thông qua thông qua khối khác thông qua việc tạo ra ảnh dự đoán, việc tạo ra phần dư dự đoán, việc tạo ra các hệ số được lượng tử hóa, việc khôi phục của phần dư dự đoán, và việc bổ sung ảnh dự đoán.

Ảnh được khôi phục có thể là, ví dụ, ảnh trong ảnh tham chiếu, hoặc ảnh của khối được mã hóa (tức là, khối khác được mô tả nêu trên) trong ảnh hiện tại mà là ảnh bao gồm khối hiện tại. Khối được mã hóa trong ảnh hiện tại là, ví dụ, khối lân cận của khối hiện tại.

FIG.29 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa 100.

Bộ dự đoán tạo ra ảnh dự đoán nhờ sử dụng phương pháp thứ nhất (bước Sc_1a), tạo ra ảnh dự đoán nhờ sử dụng phương pháp thứ hai (bước Sc_1b), và tạo ra ảnh dự đoán nhờ sử dụng phương pháp thứ ba (bước Sc_1c). Phương pháp thứ

nhất, phương pháp thứ hai, và phương pháp thứ ba có thể là các phương pháp khác nhau để tạo ra ảnh dự đoán. Mỗi phương pháp thứ nhất đến thứ ba có thể là phương pháp dự đoán liên đới, phương pháp nội dự đoán, hoặc phương pháp dự đoán khác. Ảnh được khôi phục nêu trên có thể được sử dụng trong các phương pháp dự đoán này.

Tiếp theo, bộ xử lý dự đoán đánh giá các ảnh dự đoán được tạo ra trong các bước Sc_1a, Sc_1b, và Sc_1c (bước Sc_2). Ví dụ, bộ dự đoán tính toán các giá trị C đối với các ảnh dự đoán được tạo ra trong bước Sc_1a, Sc_1b, và Sc_1c, và đánh giá các ảnh dự đoán bằng cách so sánh các giá trị C của các ảnh dự đoán. Cần lưu ý rằng giá trị C có thể được tính toán, ví dụ, theo biểu thức mô hình tối ưu hóa R-D, ví dụ, $C = D + \lambda \times R$. Trong biểu thức này, D chỉ báo các độ méo nén của ảnh dự đoán, và được biểu diễn như là, ví dụ, tổng của các độ chênh lệch tuyệt đối giữa giá trị điểm ảnh của khối hiện tại và giá trị điểm ảnh của ảnh dự đoán. Ngoài ra, R chỉ báo tốc độ bit của dòng. Ngoài ra, λ chỉ báo, ví dụ, số nhân theo phương pháp nhân tử Lagrange.

Sau đó, bộ dự đoán lựa chọn một trong số các ảnh dự đoán được tạo ra trong các bước Sc_1a, Sc_1b, và Sc_1c (bước Sc_3). Nói cách khác, bộ dự đoán lựa chọn phương pháp hoặc chế độ để thu nhận ảnh dự đoán cuối cùng. Ví dụ, bộ dự đoán lựa chọn ảnh dự đoán mà có giá trị C nhỏ nhất, dựa trên các giá trị C được tính toán đối với các ảnh dự đoán. Ngoài ra, việc đánh giá trong bước Sc_2 và việc lựa chọn của ảnh dự đoán trong bước Sc_3 có thể được thực hiện dựa trên tham số mà được sử dụng trong xử lý mã hóa. Bộ mã hóa 100 có thể biến đổi thông tin để nhận dạng ảnh dự đoán được lựa chọn, phương pháp, hoặc chế độ vào dòng. Thông tin này có thể là, ví dụ, cờ hoặc loại tương tự. Theo cách này, bộ giải mã 200 có thể tạo ra ảnh dự đoán theo phương pháp hoặc chế độ được lựa chọn bởi bộ mã hóa 100, dựa trên thông tin này. Cần lưu ý rằng, trong ví dụ được minh họa trong FIG.29, bộ dự đoán lựa chọn bất kỳ trong số các ảnh dự đoán sau khi các ảnh dự đoán được tạo ra nhờ sử dụng các phương pháp tương ứng. Tuy nhiên, bộ dự đoán có thể lựa chọn phương pháp hoặc chế độ dựa trên tham số để sử dụng trong xử lý mã hóa nêu trên trước khi tạo ra các ảnh dự đoán, và có thể tạo ra ảnh dự đoán theo phương pháp hoặc chế độ được lựa chọn.

Ví dụ, phương pháp thứ nhất và phương pháp thứ hai có thể là việc nội dự đoán và việc dự đoán liên đới, một cách lần lượt, và bộ dự đoán có thể lựa chọn ảnh dự đoán cuối cùng đối với khối hiện tại từ các ảnh dự đoán được tạo ra theo các phương pháp dự đoán.

FIG.30 là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ mã hóa 100.

Đầu tiên, bộ dự đoán tạo ra ảnh dự đoán nhờ sử dụng việc nội dự đoán (bước Sd_1a), và tạo ra ảnh dự đoán nhờ sử dụng việc dự đoán liên đới (bước Sd_1b). Cần lưu ý rằng ảnh dự đoán được tạo ra bởi việc nội dự đoán cũng được gọi là ảnh nội dự đoán, và ảnh dự đoán được tạo ra bởi việc dự đoán liên đới cũng được gọi là ảnh dự đoán liên đới.

Tiếp theo, bộ dự đoán đánh giá mỗi ảnh nội dự đoán và ảnh dự đoán liên đới (bước Sd_2). Giá trị C được mô tả nêu trên có thể được sử dụng trong việc đánh giá. Sau đó, bộ dự đoán có thể lựa chọn ảnh dự đoán mà giá trị C nhỏ nhất đã được tính toán trong số ảnh nội dự đoán và ảnh dự đoán liên đới, như là ảnh dự đoán cuối cùng đối với khối hiện tại (bước Sd_3). Nói cách khác, phương pháp dự đoán hoặc chế độ tạo ra ảnh dự đoán đối với khối hiện tại được lựa chọn.

Sau đó, bộ xử lý dự đoán lựa chọn ảnh dự đoán mà giá trị C nhỏ nhất đã được tính toán trong số ảnh nội dự đoán và ảnh dự đoán liên đới, như là ảnh dự đoán cuối cùng đối với khối hiện tại (bước Sd_3). Nói cách khác, phương pháp dự đoán hoặc chế độ tạo ra ảnh dự đoán đối với khối hiện tại được lựa chọn.

(Bộ nội dự đoán)

Bộ nội dự đoán 124 tạo ra tín hiệu dự đoán (tức là, ảnh nội dự đoán) bằng cách thực hiện việc nội dự đoán (cũng được gọi là dự đoán khung nội bộ) của khối hiện tại bằng cách viện dẫn tới khối hoặc các khối trong ảnh hiện tại và được lưu trữ trong bộ nhớ khối 118. Cụ thể hơn, bộ nội dự đoán 124 tạo ra ảnh nội dự đoán bằng cách thực hiện việc nội dự đoán bằng cách viện dẫn tới các giá trị điểm ảnh (ví dụ, các giá trị độ chói và/hoặc sắc độ) của khối hoặc các khối mà lân cận khối hiện tại, và sau đó xuất ra ảnh nội dự đoán tới bộ điều khiển dự đoán 128.

Ví dụ, bộ nội dự đoán 124 thực hiện việc nội dự đoán bằng cách sử dụng một chế độ từ trong số các chế độ nội dự đoán mà đã được xác định. Các chế độ dự đoán trong thường bao gồm một hoặc nhiều chế độ dự đoán vô hướng và nhiều chế độ dự đoán có hướng. Các chế độ định trước có thể được xác định trước.

Một hoặc nhiều chế độ dự đoán vô hướng bao gồm, ví dụ, chế độ dự đoán phẳng và chế độ dự đoán DC được định nghĩa trong tiêu chuẩn H.265/Mã hóa video hiệu quả cao (HEVC).

Các chế độ dự đoán có hướng bao gồm, ví dụ, 33 chế độ dự đoán có hướng được định nghĩa trong tiêu chuẩn H.265/HEVC. Cần lưu ý rằng các chế độ dự đoán có hướng có thể còn bao gồm 32 chế độ dự đoán có hướng ngoài 33 chế độ dự đoán có hướng (đối với tổng cộng 65 chế độ dự đoán có hướng). FIG.31 là sơ đồ khái niệm để minh họa tổng 67 chế độ nội dự đoán mà có thể được sử dụng trong việc nội dự đoán (hai chế độ dự đoán vô hướng và 65 chế độ dự đoán có hướng). Các mũi tên liền nét biểu diễn 33 chiều được xác định trong tiêu chuẩn H.265/HEVC, và các mũi tên đứt nét biểu diễn 32 chiều bổ sung (hai chế độ dự đoán vô hướng không được minh họa trong FIG.31).

Trong các loại ví dụ xử lý khác nhau, khối độ chói có thể được viện dẫn tới trong việc nội dự đoán của khối sắc độ. Nói cách khác, thành phần màu của khối hiện tại có thể được dự đoán dựa trên thành phần màu của khối hiện tại. Việc dự đoán trong cũng được gọi là dự đoán mô hình tuyến tính thành phần chéo (CCLM-cross-component linear model). Chế độ nội dự đoán đối với khối sắc độ trong đó khối độ chói này được viện dẫn tới (cũng được gọi là, ví dụ, chế độ CCLM) có thể được bổ sung như là một trong số các chế độ nội dự đoán đối với các khối sắc độ.

Bộ nội dự đoán 124 có thể hiệu chỉnh các giá trị điểm ảnh được nội dự đoán dựa trên các gradien điểm ảnh tham chiếu dọc/ngang. Việc dự đoán trong kèm theo bởi loại hiệu chỉnh này cũng được gọi là kết hợp dự đoán trong phụ thuộc vị trí (PDPC-position dependent intra prediction combination). Thông tin mà chỉ báo áp dụng PDPC hoặc không (được gọi là, ví dụ, cờ PDPC) thường được báo hiệu tại cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin này không cần được thực hiện tại

cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp ảnh, cấp lớp, cấp lát, hoặc cấp CTU).

FIG.32 là lưu đồ minh họa một ví dụ của xử lý được thực hiện bởi bộ nội dự đoán 124.

Bộ nội dự đoán 124 lựa chọn một chế độ nội dự đoán từ các chế độ nội dự đoán (bước Sw_1). Sau đó, bộ nội dự đoán 124 tạo ra ảnh dự đoán theo chế độ nội dự đoán được lựa chọn (bước Sw_2). Tiếp theo, bộ nội dự đoán 124 xác định các chế độ có khả năng xảy ra nhất (MPM-most probable mode) (bước Sw_3). Các MPM bao gồm, ví dụ, sáu chế độ nội dự đoán. Ví dụ, hai chế độ trong số sáu chế độ nội dự đoán có thể là chế độ phẳng và chế độ dự đoán DC, và bốn chế độ còn lại có thể là các chế độ dự đoán có hướng. Bộ nội dự đoán 124 xác định rằng chế độ nội dự đoán được lựa chọn trong bước Sw_1 có được chứa trong các MPM hay không (bước Sw_4).

Ở đây, khi xác định rằng chế độ nội dự đoán được lựa chọn trong bước Sw_1 được chứa trong các MPM (Có trong bước Sw_4), bộ nội dự đoán 124 thiết lập cờ MPM thành 1 (bước Sw_5), và tạo ra thông tin mà chỉ báo chế độ nội dự đoán được lựa chọn trong số các MPM (bước Sw_6). Cần lưu ý rằng cờ MPM được thiết lập là 1 và thông tin mà chỉ báo chế độ nội dự đoán có thể được mã hóa như là các tham số dự đoán bởi bộ mã hóa entropy 110.

Khi xác định rằng chế độ nội dự đoán được lựa chọn không được chứa trong các MPM (Không trong bước Sw_4), bộ nội dự đoán 124 thiết lập cờ MPM thành 0 (bước Sw_7). Ngoài ra, bộ nội dự đoán 124 không thiết lập bất kỳ cờ MPM. Sau đó, bộ nội dự đoán 124 tạo ra thông tin mà chỉ báo chế độ nội dự đoán được lựa chọn trong số ít nhất một chế độ nội dự đoán mà không được chứa trong các MPM (bước Sw_8). Cần lưu ý rằng cờ MPM được thiết lập là 0 và thông tin mà chỉ báo chế độ nội dự đoán có thể được mã hóa như là các tham số dự đoán bởi bộ mã hóa entropy 110. Thông tin mà chỉ báo chế độ nội dự đoán chỉ báo, ví dụ, bất kỳ một trong số 0 đến 60.

(Bộ nội dự đoán)

Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán (ảnh dự đoán liên đới) bằng cách thực hiện việc dự đoán liên đới (cũng được gọi là dự đoán khung liên đới) của khối hiện tại bằng cách viển dẫn tới khối hoặc các khối trong ảnh tham chiếu, mà khác với ảnh hiện tại và được lưu trữ trong bộ nhớ khung 122. Việc dự đoán liên đới được thực hiện trong các đơn vị của khối hiện tại hoặc khối con hiện tại (ví dụ, trên khối 4×4) trong khối hiện tại. Khối con được chứa trong khối và là đơn vị nhỏ hơn khối. Kích cỡ của khối con có thể nằm trong dạng của lát, viên, ảnh, v.v.

Ví dụ, bộ dự đoán liên đới 126 thực hiện ước lượng chuyển động trong ảnh tham chiếu đối với khối hiện tại hoặc khối con hiện tại, và tìm thấy khối tham chiếu hoặc khối con tham chiếu mà khớp nhất với khối hiện tại hoặc khối con hiện tại. Sau đó, bộ dự đoán liên đới 126 thu nhận thông tin chuyển động (ví dụ, vector chuyển động) mà bù chuyển động hoặc thay đổi từ khối tham chiếu hoặc khối con tham chiếu tới khối hiện tại hoặc khối con. Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán liên đới của khối hiện tại hoặc khối con bằng cách thực hiện việc bù chuyển động (hoặc dự đoán chuyển động) dựa trên thông tin chuyển động. Bộ dự đoán liên đới 126 xuất ra ảnh dự đoán liên đới được tạo ra tới bộ điều khiển dự đoán 128.

Thông tin chuyển động được sử dụng trong việc bù chuyển động có thể được báo hiệu như là các tín hiệu dự đoán liên đới trong các dạng khác nhau. Ví dụ, vector chuyển động có thể được báo hiệu. Theo ví dụ khác, độ chênh lệch giữa vector chuyển động và phần dự đoán vector chuyển động có thể được báo hiệu.

(Danh sách ảnh tham chiếu)

FIG.33 là sơ đồ khái niệm để minh họa các ví dụ về các ảnh tham chiếu. FIG.34 là sơ đồ khái niệm để minh họa các ví dụ về các danh sách ảnh tham chiếu. Danh sách ảnh tham chiếu là danh sách mà chỉ báo ít nhất một ảnh tham chiếu được lưu trữ trong bộ nhớ khung 122. Cần lưu ý rằng, trong FIG.33, mỗi hình chữ nhật chỉ báo ảnh, mỗi mũi tên chỉ báo quan hệ tham chiếu ảnh, trực ngang chỉ báo thời gian, I, P, và B trong các hình chữ nhật chỉ báo ảnh nội dự đoán, ảnh dự đoán đơn hướng, và ảnh dự đoán hai hướng, một cách lần lượt, và các số trong các hình chữ nhật chỉ báo thứ tự giải mã. Như được minh họa trong FIG.33, thứ tự giải mã của các ảnh là thứ tự I0, P1, B2, B3, và B4, và thứ tự hiển thị của các ảnh là thứ tự I0,

B3, B2, B4, và P1. Như được minh họa trong FIG.34, danh sách ảnh tham chiếu là danh sách mà biểu diễn các ứng viên ảnh tham chiếu. Ví dụ, một ảnh (hoặc lát) có thể bao gồm ít nhất một danh sách ảnh tham chiếu. Ví dụ, một danh sách ảnh tham chiếu được sử dụng khi ảnh hiện tại là ảnh dự đoán đơn hướng, và hai danh sách ảnh tham chiếu được sử dụng khi ảnh hiện tại là ảnh dự đoán hai hướng. Trong các ví dụ của các Fig.33 và Fig.34, ảnh B3 mà là ảnh hiện tại currPic có hai danh sách ảnh tham chiếu mà là danh sách L0 và danh sách L1. Khi ảnh hiện tại currPic là ảnh B3, các ứng viên ảnh tham chiếu đối với ảnh hiện tại currPic là I0, P1, và B2, và các danh sách ảnh tham chiếu (mà là danh sách L0 và danh sách L1) chỉ báo các ảnh này. Bộ dự đoán liên đới 126 hoặc bộ điều khiển dự đoán 128 chỉ rõ ảnh nào trong mỗi danh sách ảnh tham chiếu được viện dẫn thực tế trong dạng của chỉ số ảnh tham chiếu refIdxLx. Trong FIG.34, các ảnh tham chiếu P1 và B2 được chỉ rõ bởi các chỉ số ảnh tham chiếu refIdxL0 và refIdxL1.

Danh sách ảnh tham chiếu này có thể được tạo ra đối với mỗi đơn vị như chuỗi, ảnh, lát, viên, CTU, hoặc CU. Ngoài ra, trong số các ảnh tham chiếu được chỉ báo trong các danh sách ảnh tham chiếu, chỉ số ảnh tham chiếu mà chỉ báo ảnh tham chiếu được viện dẫn tới trong việc dự đoán liên đới có thể được báo hiệu tại mức chuỗi, mức ảnh, mức lát, mức viên, mức CTU, hoặc mức CU. Ngoài ra, danh sách ảnh tham chiếu chung có thể được sử dụng trong các chế độ dự đoán liên đới.

(Dòng xử lý cơ bản của việc dự đoán liên đới)

FIG.35 là lưu đồ minh họa dòng xử lý cơ bản ví dụ của xử lý dự đoán liên đới.

Đầu tiên, bộ dự đoán liên đới 126 tạo ra tín hiệu dự đoán (Các bước Se_1 đến Se_3). Tiếp theo, bộ trừ 104 tạo ra độ chênh lệch giữa khối hiện tại và ảnh dự đoán như là phần dư dự đoán (bước Se_4).

Ở đây, trong việc tạo ra ảnh dự đoán, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán thông qua việc xác định vector chuyển động (MV) của khối hiện tại (các bước Se_1 và Se_2) và việc bù chuyển động (bước Se_3). Ngoài ra, trong việc xác định MV, bộ dự đoán liên đới 126 xác định MV thông qua việc lựa chọn ứng viên vector chuyển động (ứng viên MV) (bước Se_1) và thu nhận MV (bước Se_2). Việc lựa

chọn của ứng viên MV được thực hiện bởi, ví dụ, bộ dự đoán liên đới 126 mà tạo ra danh sách ứng viên MV và lựa chọn ít nhất một ứng viên MV từ danh sách ứng viên MV. Cần lưu ý rằng các MV thu được trong quá khứ có thể được thêm vào danh sách ứng viên MV. Ngoài ra, trong việc thu nhận MV, bộ dự đoán liên đới 126 có thể còn lựa chọn ít nhất một ứng viên MV từ ít nhất một ứng viên MV, và xác định ít nhất một ứng viên MV được lựa chọn như là MV đối với khối hiện tại. Ngoài ra, bộ dự đoán liên đới 126 có thể xác định MV đối với khối hiện tại bằng cách thực hiện việc ước lượng trong vùng ảnh tham chiếu được chỉ rõ bởi mỗi ít nhất một ứng viên MV được lựa chọn. Cần lưu ý rằng việc ước lượng trong vùng ảnh tham chiếu có thể được gọi là ước lượng chuyển động.

Ngoài ra, mặc dù các bước Se_1 đến Se_3 được thực hiện bởi bộ dự đoán liên đới 126 trong ví dụ nêu trên, xử lý mà là ví dụ bước Se_1, bước Se_2, hoặc loại tương tự có thể được thực hiện bởi thành phần cấu thành khác được chứa trong bộ mã hóa 100.

Cần lưu ý rằng danh sách ứng viên MV có thể được tạo ra đối với mỗi xử lý trong chế độ dự đoán liên đới, hoặc danh sách ứng viên MV chung có thể được sử dụng trong các chế độ dự đoán liên đới. Các xử lý trong các bước Se_3 và Se_4 tương ứng với các bước Sa_3 và Sa_4 được minh họa trong FIG.9, một cách lần lượt. Xử lý trong bước Se_3 tương ứng với xử lý trong bước Sd_1b trong FIG.30.

(Dòng xử lý thu nhận vector chuyển động)

FIG.36 là lưu đồ minh họa một ví dụ về xử lý thu nhận của các vector chuyển động.

Bộ dự đoán liên đới 126 có thể thu nhận MV của khối hiện tại trong chế độ để mã hóa thông tin chuyển động (ví dụ, MV). Trong trường hợp này, ví dụ, thông tin chuyển động có thể được mã hóa như là tham số dự đoán, và có thể được báo hiệu. Nói cách khác, thông tin chuyển động được mã hóa được chứa trong dòng.

Ngoài ra, bộ dự đoán liên đới 126 có thể thu nhận MV trong chế độ trong đó thông tin chuyển động không được mã hóa. Trong trường hợp này, không có thông tin chuyển động được chứa trong dòng.

Ở đây, các chế độ thu nhận MV có thể bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ FRUC, chế độ afin, v.v. mà được mô tả sau đây. Các chế độ trong đó thông tin chuyển động được mã hóa trong số các chế độ bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ afin (cụ thể, chế độ liên đới afin và chế độ hợp nhất afin), v.v. Cần lưu ý rằng thông tin chuyển động có thể bao gồm không chỉ MV mà còn thông tin lựa chọn bộ dự đoán vector chuyển động mà được mô tả sau đây. Các chế độ trong đó không có thông tin chuyển động được mã hóa bao gồm chế độ FRUC, v.v. bộ dự đoán liên đới 126 lựa chọn chế độ để thu nhận MV của khối hiện tại từ các chế độ, và thu nhận MV của khối hiện tại nhờ sử dụng chế độ được lựa chọn.

FIG.37 là lưu đồ minh họa một ví dụ khác về xử lý thu nhận của các vector chuyển động.

Bộ dự đoán liên đới 126 có thể thu nhận MV đối với khối hiện tại trong chế độ trong đó độ chênh lệch MV được mã hóa. Trong trường hợp này, ví dụ, độ chênh lệch MV có thể được mã hóa như là tham số dự đoán, và có thể được báo hiệu. Nói cách khác, độ chênh lệch MV được mã hóa được chứa trong dòng. Độ chênh lệch MV là độ chênh lệch giữa MV của khối hiện tại và bộ dự đoán MV. Cần lưu ý rằng bộ dự đoán MV là bộ dự đoán vector chuyển động.

Ngoài ra, bộ dự đoán liên đới 126 có thể thu nhận MV trong chế độ trong đó không có độ chênh lệch MV được mã hóa. Trong trường hợp này, không có độ chênh lệch MV được mã hóa được chứa trong dòng.

Ở đây, như được mô tả nêu trên, các chế độ thu nhận MV bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ FRUC, chế độ afin, v.v. mà được mô tả sau đây. Các chế độ trong đó độ chênh lệch MV được mã hóa trong số các chế độ bao gồm chế độ liên đới thường, chế độ afin (cụ thể, chế độ liên đới afin), v.v. Các chế độ trong đó không có độ chênh lệch MV được mã hóa bao gồm chế độ FRUC, chế độ hợp nhất thường, chế độ afin (cụ thể, chế độ hợp nhất afin), v.v. bộ dự đoán liên đới 126 lựa chọn chế độ để thu nhận MV của khối hiện tại từ các chế độ, và thu nhận MV của khối hiện tại nhờ sử dụng chế độ được lựa chọn.

(Các chế độ thu nhận vector chuyển động)

Các Fig.38A và Fig.38B là các sơ đồ khái niệm để minh họa hạng mục ví dụ của các chế độ để thu nhận MV. Ví dụ, như được minh họa trong FIG.38A, các chế độ thu nhận MV được phân loại cơ bản thành ba chế độ theo việc có mã hóa hay không thông tin chuyển động và có mã hóa hay không các độ chênh lệch MV. Ba chế độ là chế độ liên đới, chế độ hợp nhất, và chế độ chuyển đổi lên tốc độ khung (FRUC-frame rate up-conversion). Chế độ liên đới là chế độ trong đó ước lượng chuyển động được thực hiện, và trong đó thông tin chuyển động và độ chênh lệch MV được mã hóa. Ví dụ, như được minh họa trong FIG.38B, chế độ liên đới bao gồm chế độ liên đới afin và chế độ liên đới thường. Chế độ hợp nhất là chế độ trong đó no ước lượng chuyển động được thực hiện, và trong đó MV được lựa chọn từ khối xung quanh được mã hóa và MV đối với khối hiện tại được thu nhận nhờ sử dụng MV. Chế độ hợp nhất là chế độ trong đó, về cơ bản, thông tin chuyển động được mã hóa và không có độ chênh lệch MV được mã hóa. Ví dụ, như được minh họa trong FIG.38B, các chế độ hợp nhất bao gồm chế độ hợp nhất thường (cũng được gọi là chế độ hợp nhất thường hoặc chế độ hợp nhất đều), hợp nhất với chế độ độ chênh lệch vector chuyển động (MMVD), chế độ liên đới kết hợp / chế độ nội dự đoán (CIIP), chế độ tam giác, chế độ ATMVP, và chế độ hợp nhất afin. Ở đây, độ chênh lệch MV được mã hóa ngoại trừ trong chế độ MMVD trong số các chế độ được chứa trong các chế độ hợp nhất. Cần lưu ý rằng chế độ hợp nhất afin và chế độ liên đới afin là các chế độ được chứa trong các chế độ afin. Chế độ afin là chế độ để thu nhận, như là MV của khối hiện tại, MV của mỗi khối con được chứa trong khối hiện tại, giả định biến đổi afin. Chế độ FRUC là chế độ mà để thu nhận MV của khối hiện tại bằng cách thực hiện việc ước lượng giữa các vùng được mã hóa, và trong đó không có thông tin chuyển động và bất kỳ độ chênh lệch MV được mã hóa. Cần lưu ý rằng các chế độ tương ứng sẽ được mô tả chi tiết hơn sau đây.

Cần lưu ý rằng các phân loại của các chế độ được minh họa trong các Fig.38A và Fig.38B là các ví dụ, và việc phân loại không bị giới hạn ở đây. Ví dụ, khi độ chênh lệch MV được mã hóa trong chế độ CIIP, chế độ CIIP được phân loại thành các chế độ liên đới.

(Thu nhận MV > chế độ liên đới thường)

Chế độ liên đới thường là chế độ dự đoán liên đới để thu nhận MV của khối hiện tại dựa trên khối tương tự với ảnh của khối hiện tại từ vùng ảnh tham chiếu được chỉ rõ bởi ứng viên MV. Trong chế độ liên đới thường này, độ chênh lệch MV được mã hóa.

FIG.39 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới trong chế độ liên đới thường.

Đầu tiên, bộ dự đoán liên đới 126 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được mã hóa theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sg_1). Nói cách khác, bộ dự đoán liên đới 126 tạo ra danh sách ứng viên MV.

Tiếp theo, bộ dự đoán liên đới 126 tách N (số nguyên bằng 2 hoặc lớn hơn) ứng viên MV từ các ứng viên MV thu được trong bước Sg_1, như là các ứng viên bộ dự đoán vector chuyển động (cũng được gọi là các ứng viên bộ dự đoán MV) theo thứ tự ưu tiên được xác định (bước Sg_2). Cần lưu ý rằng thứ tự ưu tiên có thể được xác định trước đối với mỗi N ứng viên MV.

Tiếp theo, bộ dự đoán liên đới 126 lựa chọn một ứng viên bộ dự đoán vector chuyển động từ N ứng viên bộ dự đoán vector chuyển động, như là bộ dự đoán vector chuyển động (cũng được gọi là bộ dự đoán MV) của khối hiện tại (bước Sg_3). Tại thời điểm này, bộ dự đoán liên đới 126 mã hóa, trong dòng, thông tin lựa chọn bộ dự đoán vector chuyển động để nhận dạng bộ dự đoán vector chuyển động được lựa chọn. Nói cách khác, bộ dự đoán liên đới 126 xuất ra thông tin lựa chọn bộ dự đoán MV như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Tiếp theo, bộ dự đoán liên đới 126 thu nhận MV của khối hiện tại bằng cách viện dẫn tới ảnh tham chiếu được mã hóa (bước Sg_4). Lúc này, bộ dự đoán liên đới 126 còn mã hóa, trong dòng, giá trị chênh lệch giữa MV được thu nhận và bộ dự đoán vector chuyển động như là độ chênh lệch MV. Nói cách khác, bộ dự đoán liên đới 126 xuất ra độ chênh lệch MV như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130. Cần lưu ý rằng ảnh tham chiếu được mã hóa là ảnh bao gồm các khối mà được khôi phục sau khi được mã hóa.

Cuối cùng, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được thu nhận và ảnh tham chiếu được mã hóa (bước Sg_5). Các xử lý trong các bước Sg_1 đến Sg_5 được thực thi trên mỗi khối. Ví dụ, khi các xử lý trong các bước Sg_1 đến Sg_5 được thực thi trên tất cả các khối trong lát, việc dự đoán liên đới của lát sử dụng chế độ liên đới thường hoàn thành. Ví dụ, khi các xử lý trong các bước Sg_1 đến Sg_5 được thực thi trên tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ liên đới thường hoàn thành. Cần lưu ý rằng không phải tất cả các khối được chứa trong lát các xử lý có thể được thực hiện trong các bước Sg_1 đến Sg_5, và việc dự đoán liên đới của lát sử dụng chế độ liên đới thường có thể hoàn thiện khi một phần của khối được đưa vào các xử lý này. Điều này cũng áp dụng tới các xử lý trong các bước Sg_1 đến Sg_5. Việc dự đoán liên đới của ảnh sử dụng chế độ liên đới thường có thể hoàn thành khi các xử lý được thực thi trên một phần của các khối trong ảnh.

Cần lưu ý rằng ảnh dự đoán là tín hiệu dự đoán liên đới như được mô tả nêu trên. Ngoài ra, thông tin mà chỉ báo chế độ dự đoán liên đới (chế độ liên đới thường trong ví dụ nêu trên) được sử dụng để tạo ra ảnh dự đoán, ví dụ, được mã hóa như là tham số dự đoán trong tín hiệu được mã hóa.

Cần lưu ý rằng danh sách ứng viên MV có thể cũng được sử dụng như là danh sách để sử dụng trong chế độ khác. Ngoài ra, các xử lý liên quan đến danh sách ứng viên MV có thể được áp dụng tới các xử lý liên quan đến danh sách để sử dụng trong chế độ khác. Các xử lý liên quan đến danh sách ứng viên MV bao gồm, ví dụ, việc tách hoặc lựa chọn ứng viên MV từ danh sách ứng viên MV, sắp xếp lại các ứng viên MV, hoặc xóa bỏ ứng viên MV.

(Thu nhận MV > chế độ hợp nhất thường)

Chế độ hợp nhất thường là chế độ dự đoán liên đới để lựa chọn ứng viên MV từ danh sách ứng viên MV như là MV của khối hiện tại, nhờ đó thu nhận MV. Cần lưu ý rằng chế độ hợp nhất thường là loại của chế độ hợp nhất và có thể được gọi đơn giản là chế độ hợp nhất. Trong phương án này, chế độ hợp nhất thường và chế

độ hợp nhất được phân biệt, và chế độ hợp nhất được sử dụng trong ý nghĩa rộng hơn.

FIG.40 là lưu đồ minh họa ví dụ về việc dự đoán liên đới trong chế độ hợp nhất thường.

Đầu tiên, bộ dự đoán liên đới 126 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được mã hóa theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sh_1). Nói cách khác, bộ dự đoán liên đới 126 tạo ra danh sách ứng viên MV.

Tiếp theo, bộ dự đoán liên đới 126 lựa chọn một ứng viên MV từ các ứng viên MV thu được trong bước Sh_1, nhờ đó thu nhận MV của khối hiện tại (bước Sh_2). Lúc này, bộ dự đoán liên đới 126 mã hóa, trong dòng, thông tin lựa chọn MV để nhận dạng ứng viên MV được lựa chọn. Nói cách khác, bộ dự đoán liên đới 126 xuất ra thông tin lựa chọn MV như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Cuối cùng, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được thu nhận và ảnh tham chiếu được mã hóa (bước Sh_3). Các xử lý trong các bước Sh_1 đến Sh_3 được thực thi, ví dụ, trên mỗi khối. Ví dụ, khi các xử lý trong các bước Sh_1 đến Sh_3 được thực thi trên tất cả các khối trong lát, việc dự đoán liên đới của lát sử dụng chế độ hợp nhất thường hoàn thành. Ngoài ra, khi các xử lý trong các bước Sh_1 đến Sh_3 được thực thi trên tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ hợp nhất thường hoàn thành. Cần lưu ý rằng không phải tất cả các khối được chứa trong lát có thể được đưa vào xử lý trong các Sh_1 đến Sh_3, và việc dự đoán liên đới của lát sử dụng chế độ hợp nhất thường có thể hoàn thành khi một phần của các khối được đưa vào các xử lý. Điều này cũng áp dụng tới các xử lý trong các bước Sh_1 đến Sh_3. Việc dự đoán liên đới của ảnh sử dụng chế độ hợp nhất thường có thể hoàn thành khi các xử lý được thực thi trên một phần của các khối trong ảnh.

Ngoài ra, thông tin mà chỉ báo chế độ dự đoán liên đới (chế độ hợp nhất thường trong ví dụ nêu trên) được sử dụng để tạo ra ảnh dự đoán và được chứa trong tín hiệu được mã hóa, ví dụ, được mã hóa như là tham số dự đoán trong dòng.

FIG.41 là sơ đồ khái niệm để minh họa một ví dụ của xử lý thu nhận vector chuyển động của ảnh hiện tại bởi chế độ hợp nhất thường.

Đầu tiên, bộ dự đoán liên đới 126 tạo ra danh sách ứng viên MV trong đó các ứng viên MV được ghi. Các ví dụ của các ứng viên MV bao gồm: các ứng viên MV lân cận theo không gian mà là các MV của các khối được mã hóa nằm xung quanh về mặt không gian khối hiện tại; các ứng viên MV lân cận về mặt thời gian mà là các MV của các khối xung quanh mà trên đó vị trí của khối hiện tại trong ảnh tham chiếu được mã hóa được đưa ra; các ứng viên MV được kết hợp mà là các MV được tạo ra bằng cách kết hợp giá trị MV của bộ dự đoán MV lân cận về mặt không gian và giá trị MV của bộ dự đoán MV lân cận về mặt thời gian; và ứng viên MV 0 mà là MV có giá trị 0.

Tiếp theo, bộ dự đoán liên đới 126 lựa chọn một ứng viên MV từ các ứng viên MV được ghi trong danh sách ứng viên MV, và xác định ứng viên MV như là MV của khối hiện tại.

Ngoài ra, bộ mã hóa entropy 110 ghi và mã hóa, trong dòng, `merge_idx` mà là tín hiệu chỉ báo ứng viên MV nào được lựa chọn.

Cần lưu ý rằng các ứng viên MV được ghi trong danh sách ứng viên MV được mô tả trong FIG.41 là các ví dụ. Số lượng ứng viên MV có thể khác với số lượng ứng viên MV trong sơ đồ, danh sách ứng viên MV có thể được cấu hình trong cách thức mà một vài loại của các ứng viên MV trong sơ đồ có thể không được bao gồm, hoặc một hoặc nhiều ứng viên MV khác ngoài các loại ứng viên MV trong sơ đồ được bao gồm.

MV cuối cùng có thể được xác định bằng cách thực hiện việc làm mới vector chuyển động linh động (DMVR) được mô tả sau đây nhờ sử dụng MV của khối hiện tại thu được bởi chế độ hợp nhất thường. Cần lưu ý rằng, trong chế độ hợp nhất thường, thông tin chuyển động được mã hóa và không có độ chênh lệch MV được mã hóa. Trong chế độ MMVD, một ứng viên MV được lựa chọn từ danh sách ứng

viên MV như trong trường hợp của chế độ hợp nhất thường, độ chênh lệch MV được mã hóa. Như được minh họa trong FIG.38B, MMVD có thể được phân loại thành các chế độ hợp nhất cùng với chế độ hợp nhất thường. Cần lưu ý rằng độ chênh lệch MV trong chế độ MMVD không cần luôn giống với độ chênh lệch MV để sử dụng trong chế độ liên đới. Ví dụ, việc thu nhận độ chênh lệch MV trong chế độ MMVD có thể là xử lý mà yêu cầu lượng xử lý nhỏ hơn lượng xử lý được yêu cầu cho việc thu nhận độ chênh lệch MV trong chế độ liên đới.

Ngoài ra, chế độ hợp nhất liên đới kết hợp / nội dự đoán (CIIP) có thể được thực hiện. Chế độ này là để xếp chồng ảnh dự đoán được tạo ra trong việc dự đoán liên đới và ảnh dự đoán được tạo ra trong việc nội dự đoán để tạo ra ảnh dự đoán đối với khối hiện tại.

Cần lưu ý rằng danh sách ứng viên MV có thể được gọi là danh sách ứng viên. Ngoài ra, merge_idx là thông tin lựa chọn MV.

(Thu nhận MV > chế độ HMVP)

FIG.42 là sơ đồ khái niệm để minh họa một ví dụ về xử lý thu nhận MV đối với ảnh hiện tại sử dụng chế độ hợp nhất HMVP.

Trong chế độ hợp nhất thường, MV đối với, ví dụ, CU mà là khối hiện tại được xác định bằng cách lựa chọn một ứng viên MV từ danh sách MV được tạo ra bằng cách viên dẫn tới khối được mã hóa (ví dụ, CU). Ở đây, ứng viên MV khác có thể được ghi trong danh sách ứng viên MV. Chế độ trong đó ứng viên MV khác này được ghi được gọi là chế độ HMVP.

Trong chế độ HMVP, các ứng viên MV được quản lý nhờ sử dụng máy chủ vào trước ra trước (FIFO-first-in first-out) đối với HMVP, một cách riêng biệt từ danh sách ứng viên MV đối với chế độ hợp nhất thường.

Trong bộ đệm FIFO, thông tin chuyển động như các MV của các khối được xử lý trong quá khứ được lưu trữ mới nhất trước. Trong quản lý của bộ đệm FIFO, mỗi khi một khối được xử lý, MV đối với khối mới nhất (mà là CU được xử lý ngay trước) được lưu trữ trong bộ đệm FIFO, và MV của CU cũ nhất (tức là, CU được xử lý sớm nhất) được xóa khỏi bộ đệm FIFO. Trong ví dụ được minh họa trong

FIG.42, HMVP1 là MV đối với khối mới nhất, và HMVP5 là MV đối với MV cũ nhất.

Sau đó, bộ dự đoán liên đới 126, ví dụ, kiểm tra rằng mỗi MV được quản lý trong bộ đệm FIFO có phải là MV khác với tất cả ứng viên MV mà đã được ghi trong danh sách ứng viên MV đối với chế độ hợp nhất thường bắt đầu từ HMVP1 hay không. Khi xác định rằng MV khác với tất cả ứng viên MV, bộ dự đoán liên đới 126 có thể thêm MV được quản lý trong bộ đệm FIFO trong danh sách ứng viên MV đối với chế độ hợp nhất thường như là ứng viên MV. Lúc này, một hoặc nhiều ứng viên MV trong bộ đệm FIFO có thể được ghi (được thêm vào danh sách ứng viên MV).

Bằng cách sử dụng chế độ HMVP theo cách này, có thể thêm không chỉ MV của khối mà lân cận khối hiện tại theo không gian hoặc thời gian mà còn MV đối với khối được xử lý trong quá khứ. Kết quả là, biến thể của các ứng viên MV đối với chế độ hợp nhất thường được mở rộng, mà làm tăng xác suất mà hiệu quả mã hóa có thể được tăng lên.

Cần lưu ý rằng MV có thể là thông tin chuyển động. Nói cách khác, thông tin được lưu trữ trong danh sách ứng viên MV và bộ đệm FIFO có thể bao gồm không chỉ các giá trị MV mà còn thông tin ảnh tham chiếu, các chiều tham chiếu, các số lượng ảnh, v.v. Ngoài ra, khối có thể là, ví dụ, CU.

Cần lưu ý rằng danh sách ứng viên MV và bộ đệm FIFO được minh họa trong FIG.42 là các ví dụ. Danh sách ứng viên MV và bộ đệm FIFO có thể khác nhau về kích cỡ so với những gì trong FIG.42, hoặc có thể được cấu hình để ghi các ứng viên MV trong thứ tự khác với thứ tự trong FIG.42. Ngoài ra, xử lý được mô tả ở đây có thể dùng chung giữa bộ mã hóa 100 và bộ giải mã 200.

Cần lưu ý rằng chế độ HMVP có thể được áp dụng đối với các chế độ khác ngoài chế độ hợp nhất thường. Ví dụ, cũng có thể rằng thông tin chuyển động như các MV của các khối được xử lý trong chế độ afin trong quá khứ có thể được lưu trữ mới nhất đầu tiên, và có thể được sử dụng như là các ứng viên MV, mà có thể hỗ trợ hiệu quả tốt hơn. Chế độ thu được bằng cách áp dụng chế độ HMVP tới chế độ afin có thể được gọi là chế độ afin lịch sử.

(Thu nhận MV > chế độ FRUC)

Thông tin chuyển động có thể được thu nhận tại phía bộ giải mã mà không được báo hiệu từ phía bộ mã hóa. Ví dụ, thông tin chuyển động có thể được thu nhận bằng cách thực hiện ước lượng chuyển động tại phía bộ giải mã 200. Trong phương án, tại phía bộ giải mã, ước lượng chuyển động được thực hiện mà không sử dụng bất kỳ giá trị điểm ảnh trong khối hiện tại. Các chế độ thực hiện ước lượng chuyển động tại phía bộ giải mã 200 mà không sử dụng bất kỳ giá trị điểm ảnh trong khối hiện tại bao gồm chế độ chuyển đổi lên tốc độ khung (FRUC), chế độ thu nhận vector chuyển động được so khớp mẫu (PMMVD), v.v.

Một ví dụ của xử lý FRUC trong dạng của lưu đồ được minh họa trong FIG.43. Đầu tiên, danh sách mà chỉ báo, như là các ứng viên MV, các MV đối với các khối được mã hóa mà mỗi chúng lân cận khối hiện tại về mặt không gian hoặc thời gian bằng cách viện dẫn tới các MV (danh sách có thể là danh sách ứng viên MV, và cũng được sử dụng là danh sách ứng viên MV đối với chế độ hợp nhất thường) (bước Si_1).

Tiếp theo, ứng viên MV tốt nhất được lựa chọn từ các ứng viên MV được ghi trong danh sách ứng viên MV (bước Si_2). Ví dụ, các giá trị đánh giá của các ứng viên MV tương ứng được chứa trong danh sách ứng viên MV được tính toán, và một ứng viên MV được lựa chọn dựa trên các giá trị đánh giá. Dựa trên các ứng viên vector chuyển động được lựa chọn, vector chuyển động đối với khối hiện tại được thu nhận sau đó (bước Si_4). Cụ thể hơn, ví dụ, ứng viên vector chuyển động được lựa chọn (ứng viên MV tốt nhất) được thu nhận một cách trực tiếp như là vector chuyển động đối với khối hiện tại. Ngoài ra, ví dụ, vector chuyển động đối với khối hiện tại có thể được thu nhận sử dụng so khớp khuôn mẫu trong vùng xung quanh của vị trí trong ảnh tham chiếu trong đó vị trí trong ảnh tham chiếu tương ứng với ứng viên vector chuyển động được lựa chọn. Nói cách khác, việc ước lượng sử dụng so khớp khuôn mẫu và các giá trị đánh giá có thể được thực hiện trong vùng xung quanh của ứng viên MV tốt nhất, và khi có MV mà có giá trị đánh giá tốt hơn, ứng viên MV tốt nhất có thể được cập nhật thành MV mà có giá trị đánh giá tốt hơn, và MV được cập nhật có thể được xác định như là MV cuối cùng đối với khối hiện tại. Trong một vài phương án, việc cập nhật vector chuyển động mà có giá trị đánh giá tốt hơn có thể không được thực hiện.

Cuối cùng, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được thu nhận và ảnh tham chiếu được mã hóa (bước Si_5). Các xử lý trong các bước Si_1 đến Si_5 được thực thi, ví dụ, trên mỗi khối. Ví dụ, khi các xử lý trong các bước Si_1 đến Si_5 được thực thi trên tất cả các khối trong lát, việc dự đoán liên đới của lát sử dụng chế độ FRUC hoàn thành. Ví dụ, khi các xử lý trong các bước Si_1 đến Si_5 được thực thi trên tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ FRUC hoàn thành. Cần lưu ý rằng không phải tất cả các khối được chứa trong lát có thể được đưa vào xử lý trong các bước Si_1 đến Si_5, và việc dự đoán liên đới của lát sử dụng chế độ FRUC có thể hoàn thành khi một phần của các khối được đưa vào các xử lý. Khi các xử lý trong các bước Si_1 đến Si_5 được thực thi trên một phần của các khối được chứa trong ảnh trong cách thức tương tự, việc dự đoán liên đới của ảnh sử dụng chế độ FRUC có thể hoàn thành.

Xử lý tương tự có thể được thực hiện trong các đơn vị của khối con.

Các giá trị đánh giá có thể được tính toán theo các loại phương pháp khác nhau. Ví dụ, việc so sánh được thực hiện giữa ảnh được khôi phục trong vùng trong ảnh tham chiếu tương ứng với vector chuyển động, và ảnh được khôi phục trong vùng được xác định (vùng có thể là, ví dụ, vùng trong ảnh tham chiếu khác hoặc vùng trong khối lân cận của ảnh hiện tại, như được thể hiện dưới đây). Vùng được xác định có thể được xác định trước.

Độ chênh lệch giữa các giá trị điểm ảnh của hai ảnh được khôi phục có thể được sử dụng đối với giá trị đánh giá của các vector chuyển động. Cần lưu ý rằng giá trị đánh giá có thể được tính toán nhờ sử dụng thông tin khác ngoài giá trị của độ chênh lệch.

Tiếp theo, ví dụ về việc so khớp khuôn mẫu được mô tả chi tiết. Đầu tiên, một ứng viên MV được chứa trong danh sách ứng viên MV (ví dụ, danh sách hợp nhất) được lựa chọn như là điểm bắt đầu để ước lượng bằng cách so khớp khuôn mẫu. Ví dụ, đối với việc so khớp khuôn mẫu, việc so khớp khuôn mẫu thứ nhất hoặc việc so khớp khuôn mẫu thứ hai có thể được sử dụng. Việc so khớp khuôn mẫu thứ

nhất và so khớp khuôn mẫu thứ hai có thể cũng được gọi là so khớp hai chiều và so khớp mẫu, một cách lần lượt.

(Thu nhận $MV > FRUC > \text{So khớp hai chiều}$)

Trong so khớp khuôn mẫu thứ nhất, việc so khớp khuôn mẫu được thực hiện giữa hai khối mà được bố trí dọc theo đường chuyển động của khối hiện tại và được chứa trong hai ảnh tham chiếu khác nhau. Do đó, trong so khớp khuôn mẫu thứ nhất, vùng trong ảnh tham chiếu khác dọc theo đường chuyển động của khối hiện tại được sử dụng như là vùng được xác định để tính toán giá trị đánh giá của ứng viên nêu trên. Vùng được xác định có thể được xác định trước.

FIG.44 là sơ đồ khái niệm để minh họa một ví dụ về so khớp khuôn mẫu thứ nhất (so khớp hai chiều) giữa hai khối trong hai ảnh tham chiếu dọc theo đường chuyển động. Như được minh họa trong FIG.44, trong so khớp khuôn mẫu thứ nhất, hai vectơ chuyển động (MV_0 , MV_1) được thu nhận bằng cách ước lượng cặp mà khớp nhất trong số các cặp trong hai khối được chứa trong hai ảnh tham chiếu khác nhau (Ref_0 , Ref_1) và được bố trí dọc theo đường chuyển động của khối hiện tại (khối Cur). Cụ thể hơn, độ chênh lệch giữa ảnh được khôi phục tại vị trí được cụ thể hóa trong ảnh tham chiếu được mã hóa thứ nhất (Ref_0) được chỉ rõ bởi ứng viên MV , và ảnh được khôi phục tại vị trí được chỉ rõ trong ảnh tham chiếu được mã hóa thứ hai (Ref_1) được chỉ rõ bởi MV đối xứng thu được bằng cách biến đổi tỷ lệ ứng viên MV tại khoảng thời gian hiển thị được thu nhận đối với khối hiện tại, và giá trị đánh giá được tính toán nhờ sử dụng giá trị của độ chênh lệch thu được. Có thể lựa chọn, như là MV cuối cùng, ứng viên MV mà có giá trị đánh giá tốt nhất trong số các ứng viên MV , và có thể tạo ra các kết quả tốt.

Trong giả thiết của đường chuyển động liên tục, các vectơ chuyển động (MV_0 , MV_1) mà chỉ rõ hai khối tham chiếu tỷ lệ với các khoảng cách theo thời gian (TD_0 , TD_1) giữa ảnh hiện tại (Cur Pic) và hai ảnh tham chiếu (Ref_0 , Ref_1). Ví dụ, khi ảnh hiện tại được bố trí theo thời gian giữa hai ảnh tham chiếu và các khoảng cách theo thời gian từ ảnh hiện tại tới hai ảnh tham chiếu tương ứng bằng nhau, các vectơ chuyển động hai chiều đối xứng phản chiếu được thu nhận trong so khớp khuôn mẫu thứ nhất.

(Thu nhận MV > FRUC > So khớp mẫu)

Trong so khớp khuôn mẫu thứ hai (so khớp mẫu), việc so khớp khuôn mẫu được thực hiện giữa khối trong ảnh tham chiếu và mẫu trong ảnh hiện tại (mẫu là khối lân cận khối hiện tại trong ảnh hiện tại (khối lân cận là, ví dụ, các khối lân cận phía trên và/hoặc bên trái). Do đó, trong việc so khớp khuôn mẫu thứ hai, khối mà lân cận khối hiện tại trong ảnh hiện tại được sử dụng như là vùng được xác định để tính toán giá trị đánh giá của ứng viên MV nêu trên.

FIG.45 là sơ đồ khái niệm để minh họa một ví dụ về so khớp mẫu (so khớp khuôn mẫu) giữa mẫu trong ảnh hiện tại và khối trong ảnh tham chiếu. Như được minh họa trong FIG.45, trong việc so khớp khuôn mẫu thứ hai, vectơ chuyển động của khối hiện tại (khối Cur) được thu nhận bằng cách ước lượng, trong ảnh tham chiếu (Ref0), khối mà khớp nhất với khối lân cận khối hiện tại trong ảnh hiện tại (Ảnh Cur). Cụ thể hơn, độ chênh lệch giữa ảnh được khôi phục trong vùng được mã hóa mà lân cận với cả bên trái và phía trên hoặc bên trái hoặc phía trên và ảnh được khôi phục mà nằm trong vùng tương ứng trong ảnh tham chiếu được mã hóa (Ref0) và được chỉ rõ bởi ứng viên MV được thu nhận, và giá trị đánh giá được tính toán nhờ sử dụng giá trị của độ chênh lệch thu được. Ứng viên MV mà có giá trị đánh giá tốt nhất trong số các ứng viên MV có thể được lựa chọn như là ứng viên MV tốt nhất.

Thông tin mà chỉ báo rằng áp dụng chế độ FRUC hoặc không (được gọi là, ví dụ, cờ FRUC) có thể được báo hiệu tại cấp CU. Ngoài ra, khi chế độ FRUC được áp dụng (ví dụ, khi cờ FRUC là đúng), thông tin mà chỉ báo phương pháp so khớp khuôn mẫu có thể áp dụng (ví dụ, so khớp khuôn mẫu thứ nhất hoặc so khớp khuôn mẫu thứ hai) có thể được báo hiệu tại cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin này không cần được thực hiện tại cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp ảnh, cấp lát, cấp ô, cấp CTU hoặc cấp khối con).

(Thu nhận MV > chế độ affin)

Chế độ affin là chế độ để tạo ra MV nhờ sử dụng biến đổi affin. Ví dụ, MV có thể được thu nhận trong các đơn vị của khối con dựa trên các vectơ chuyển động của các khối lân cận. Chế độ này cũng được gọi là chế độ dự đoán bù chuyển động affin.

FIG.46A là sơ đồ khái niệm để minh họa một ví dụ về thu nhận MV trong các đơn vị của khối con dựa trên các vector chuyển động của các khối lân cận. Trong FIG.46A, khối hiện tại bao gồm, ví dụ, 16 khối con 4×4 . Ở đây, vector chuyển động V_0 tại điểm điều khiển góc phía trên bên trái trong khối hiện tại được thu nhận dựa trên vector chuyển động của khối lân cận, và tương tự, vector chuyển động V_1 tại điểm điều khiển góc phía trên bên phải trong khối hiện tại được thu nhận dựa trên vector chuyển động của khối con lân cận. Hai vector chuyển động v_0 và v_1 có thể được chiếu theo biểu thức (1A) được thể hiện dưới đây, và các vector chuyển động (v_x, v_y) đối với các khối con tương ứng trong khối hiện tại có thể được thu nhận.

[Công thức toán 1]

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w}x - \frac{(v_{1y} - v_{0y})}{w}y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w}x - \frac{(v_{1x} - v_{0x})}{w}y + v_{0y} \end{cases} \quad (1A)$$

Ở đây, x và y chỉ báo vị trí theo chiều ngang và vị trí theo chiều dọc của khối con, một cách lần lượt, và w chỉ báo hệ số trọng số được xác định. Hệ số trọng số được xác định có thể được xác định trước.

Thông tin chỉ báo chế độ afin (ví dụ, được gọi là cờ afin) này có thể được báo hiệu tại mức CU. Cần lưu ý rằng việc báo hiệu của thông tin chỉ báo chế độ afin không cần được thực hiện tại cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp chuỗi, cấp ảnh, cấp lát, cấp ô, cấp CTU hoặc cấp khối con).

Ngoài ra, chế độ afin có thể bao gồm một vài chế độ đối với các phương pháp khác nhau để thu nhận các vector chuyển động tại các điểm điều khiển phía trên bên trái và phía trên bên phải. Ví dụ, chế độ afin bao gồm hai chế độ mà là chế độ liên đới afin (cũng được gọi là chế độ liên đới thường afin) và chế độ hợp nhất afin.

(Thu nhận MV > chế độ afin)

FIG.46B là sơ đồ khái niệm để minh họa một ví dụ về việc thu nhận MV trong các đơn vị của khối con trong chế độ afin trong đó ba điểm điều khiển được sử dụng. Trong FIG.46B, khối hiện tại bao gồm, ví dụ, 16 khối 4×4 . Ở đây, vector chuyển động V_0 tại điểm điều khiển góc phía trên-bên trái trong khối hiện tại được

thu nhận dựa trên vector chuyển động của khối lân cận. Ở đây, vector chuyển động V_1 tại điểm điều khiển góc phía trên-bên phải trong khối hiện tại được thu nhận dựa trên vector chuyển động của khối lân cận, và tương tự vector chuyển động V_2 tại điểm điều khiển góc phía dưới-bên trái đối với khối hiện tại được thu nhận dựa trên vector chuyển động của khối lân cận. Ba vector chuyển động v_0 , v_1 , và v_2 có thể được chiếu theo biểu thức (1B) được thể hiện dưới đây, và các vector chuyển động (v_x , v_y) đối với các khối con tương ứng trong khối hiện tại có thể được thu nhận.

[Công thức toán 2]

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w}x - \frac{(v_{2x} - v_{0x})}{h}y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w}x - \frac{(v_{2y} - v_{0y})}{h}y + v_{0y} \end{cases} \quad (1B)$$

Ở đây, x và y chỉ báo vị trí theo chiều ngang và vị trí theo chiều dọc của khối con, một cách lần lượt, và w và h có thể là các hệ số trọng số, mà có thể là các hệ số trọng số được xác định trước. Theo phương án của sáng chế, w có thể chỉ báo độ rộng của khối hiện tại, và h có thể chỉ báo độ cao của khối hiện tại.

Các chế độ affin trong đó các số lượng điểm điều khiển khác nhau (ví dụ, hai và ba điểm điều khiển) được sử dụng có thể được chuyển đổi và được báo hiệu tại mức CU. Cần lưu ý rằng thông tin mà chỉ báo số lượng điểm điều khiển trong chế độ affin được sử dụng tại mức CU có thể được báo hiệu tại mức khác (ví dụ, mức chuỗi, mức ảnh, mức lát, mức ô, mức CTU, hoặc mức khối con).

Ngoài ra, chế độ affin này trong đó ba điểm điều khiển được sử dụng có thể bao gồm các phương pháp khác nhau để thu nhận các vector chuyển động tại các điểm điều khiển phía trên-bên trái, phía trên-bên phải, và phía dưới-bên trái. Ví dụ, các chế độ affin trong đó ba điểm điều khiển được sử dụng có thể bao gồm hai chế độ mà là chế độ liên đới affin và chế độ hợp nhất affin, như trong trường hợp của các chế độ affin trong đó hai điểm điều khiển được sử dụng.

Cần lưu ý rằng, trong các chế độ affin, kích cỡ của mỗi khối con được chứa trong khối hiện tại có thể không bị giới hạn ở 4×4 điểm ảnh, và có thể là kích cỡ khác. Ví dụ, kích cỡ của mỗi khối con có thể là 8×8 điểm ảnh.

(Thu nhận MV > chế độ affin > Điểm điều khiển)

FIG.47A, FIG.47B, và FIG.47C là các sơ đồ khái niệm để minh họa các ví dụ về việc thu nhận MV tại các điểm điều khiển trong chế độ affin.

Như được minh họa trong FIG.47A, trong chế độ affin, ví dụ, các bộ dự đoán vectơ chuyển động tại các điểm điều khiển tương ứng của khối hiện tại được tính toán dựa trên các vectơ chuyển động tương ứng với các khối được mã hóa theo chế độ affin trong số khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được mã hóa mà lân cận khối hiện tại. Cụ thể hơn, khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được mã hóa được kiểm tra trong thứ tự được liệt kê, và khối có hiệu lực đầu tiên được mã hóa theo chế độ affin được nhận dạng. Các bộ dự đoán vectơ chuyển động tại các điểm điều khiển của khối hiện tại được tính toán dựa trên các vectơ chuyển động tương ứng với khối được nhận dạng.

Ví dụ, như được minh họa trong FIG.47B, khi khối A mà lân cận với bên trái của khối hiện tại đã được mã hóa theo chế độ affin trong đó hai điểm điều khiển được sử dụng, các vectơ chuyển động v_3 và v_4 được chiếu tại vị trí góc phía trên-bên trái và vị trí góc phía trên-bên phải của khối được mã hóa bao gồm khối A được thu nhận. Vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại và sau đó vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại được tính toán từ các vectơ chuyển động được thu nhận v_3 và v_4 .

Ví dụ, như được minh họa trong FIG.47C, khi khối A mà lân cận với bên trái của khối hiện tại đã được mã hóa theo chế độ affin trong đó ba điểm điều khiển được sử dụng, các vectơ chuyển động v_3 , v_4 , và v_5 được chiếu tại vị trí góc phía trên-bên trái, vị trí góc phía trên-bên phải, và vị trí góc phía dưới-bên trái của khối được mã hóa bao gồm khối A được thu nhận. Vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại, vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại, và sau đó vectơ chuyển động v_2 tại điểm điều khiển góc phía dưới-bên trái của khối hiện tại được tính toán từ các vectơ chuyển động được thu nhận v_3 , v_4 , và v_5 .

Các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C có thể được sử dụng trong việc thu nhận MV tại mỗi điểm điều khiển đối với khối hiện tại trong bước Sk_1 được minh họa trong FIG.50, hoặc có thể được sử dụng để thu nhận bộ dự đoán MV tại mỗi điểm điều khiển đối với khối hiện tại trong bước Sj_1 được minh họa trong FIG.51 được mô tả sau đây.

Các FIG.48A và FIG.48B là các sơ đồ khái niệm để minh họa các ví dụ về việc thu nhận MV tại các điểm điều khiển trong chế độ affin.

FIG.48A là sơ đồ khái niệm để minh họa chế độ affin ví dụ trong đó hai điểm điều khiển được sử dụng.

Trong chế độ affin, như được minh họa trong FIG.48A, MV được lựa chọn từ các MV tại khối A, khối B, và khối C được mã hóa mà lân cận khối hiện tại được sử dụng như là vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái đối với khối hiện tại. Tương tự, MV được lựa chọn từ các MV của khối D và khối E được mã hóa mà lân cận khối hiện tại được sử dụng như là vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải đối với khối hiện tại.

FIG.48B là sơ đồ khái niệm để minh họa chế độ affin ví dụ trong đó ba điểm điều khiển được sử dụng.

Trong chế độ affin, như được minh họa trong FIG.48B, MV được lựa chọn từ các MV tại khối A, khối B, và khối C được mã hóa mà lân cận khối hiện tại được sử dụng như là vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái đối với khối hiện tại. Tương tự, MV được lựa chọn từ các MV của khối D và khối E được mã hóa mà lân cận khối hiện tại được sử dụng như là vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải đối với khối hiện tại. Ngoài ra, MV được lựa chọn từ các MV của khối F và khối G được mã hóa mà lân cận khối hiện tại được sử dụng như là vectơ chuyển động v_2 tại điểm điều khiển góc phía dưới-bên trái đối với khối hiện tại.

Cần lưu ý rằng các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B có thể được sử dụng trong việc thu nhận MV tại mỗi điểm điều khiển đối với khối hiện tại trong bước Sk_1 được minh họa trong FIG.50 được mô tả sau đây, hoặc có thể được sử dụng để thu nhận bộ dự đoán MV tại mỗi điểm điều

khíên đối với khối hiện tại trong bước Sj_1 được minh họa trong FIG.51 được mô tả sau đây.

Ở đây, khi các chế độ affin trong đó các số lượng điểm điều khiển khác nhau (ví dụ, hai và ba điểm điều khiển) được sử dụng có thể được chuyển đổi và được báo hiệu tại mức CU, số lượng điểm điều khiển đối với khối được mã hóa và số lượng điểm điều khiển đối với khối hiện tại có thể khác nhau.

Các Fig.49A và Fig.49B là các sơ đồ khái niệm để minh họa các ví dụ về phương pháp thu nhận MV tại các điểm điều khiển khi số lượng điểm điều khiển đối với khối được mã hóa và số lượng điểm điều khiển đối với khối hiện tại là khác nhau.

Ví dụ, như được minh họa trong FIG.49A, khối hiện tại có ba điểm điều khiển tại góc phía trên-bên trái, góc phía trên-bên phải, và góc phía dưới-bên trái, và khối A mà lân cận với bên trái của khối hiện tại đã được mã hóa theo chế độ affin trong đó hai điểm điều khiển được sử dụng. Trong trường hợp này, các vectơ chuyển động v_3 và v_4 được chiếu tại vị trí góc phía trên-bên trái và vị trí góc phía trên-bên phải trong khối được mã hóa bao gồm khối A được thu nhận. Vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái và sau đó vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải đối với khối hiện tại được tính toán từ các vectơ chuyển động được thu nhận v_3 và v_4 . Ngoài ra, vectơ chuyển động v_2 tại điểm điều khiển góc phía dưới-bên trái được tính toán từ các vectơ chuyển động được thu nhận v_0 và v_1 .

Ví dụ, như được minh họa trong FIG.49B, khối hiện tại có hai điểm điều khiển tại góc phía trên-bên trái và góc phía trên-bên phải, và khối A mà lân cận với bên trái của khối hiện tại đã được mã hóa theo chế độ affin trong đó ba điểm điều khiển được sử dụng. Trong trường hợp này, các vectơ chuyển động v_3 , v_4 , và v_5 được chiếu tại vị trí góc phía trên-bên trái trong khối được mã hóa bao gồm khối A, vị trí góc phía trên-bên phải trong khối được mã hóa, và vị trí góc phía dưới-bên trái trong khối được mã hóa được thu nhận. Vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái đối với khối hiện tại và sau đó vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải đối với khối hiện tại được tính toán từ các vectơ chuyển động được thu nhận v_3 , v_4 và v_5 .

Cần lưu ý rằng các phương pháp thu nhận MV được minh họa trong các Fig.49A và Fig.49B có thể được sử dụng trong việc thu nhận MV tại mỗi điểm điều khiển đối với khối hiện tại trong bước Sk_1 được minh họa trong FIG.50 được mô tả sau đây, hoặc có thể được sử dụng để thu nhận bộ dự đoán MV tại mỗi điểm điều khiển đối với khối hiện tại trong bước Sj_1 được minh họa trong FIG.51 được mô tả sau đây.

(Thu nhận MV > chế độ affin > chế độ hợp nhất affin)

FIG.50 là lưu đồ minh họa một ví dụ về xử lý trong chế độ hợp nhất affin.

Trong chế độ hợp nhất affin như được minh họa, đầu tiên, bộ dự đoán liên đới 126 thu nhận các MV tại các điểm điều khiển tương ứng đối với khối hiện tại (bước Sk_1). Các điểm điều khiển là điểm góc phía trên-bên trái của khối hiện tại và điểm góc phía trên-bên phải của khối hiện tại như được minh họa trong FIG.46A, hoặc điểm góc phía trên-bên trái của khối hiện tại, điểm góc phía trên-bên phải của khối hiện tại, và điểm góc phía dưới-bên trái của khối hiện tại như được minh họa trong FIG.46B. Bộ dự đoán liên đới 126 có thể mã hóa thông tin lựa chọn MV để nhận hai hoặc ba MV được thu nhận trong dòng.

Ví dụ, khi các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C được sử dụng, như được minh họa trong FIG.47A, bộ dự đoán liên đới 126 kiểm tra khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được mã hóa trong thứ tự được liệt kê, và nhận dạng khối có hiệu lực thứ nhất được mã hóa theo chế độ affin.

Bộ dự đoán liên đới 126 thu nhận MV tại điểm điều khiển sử dụng khối có hiệu lực thứ nhất được nhận dạng được mã hóa theo chế độ affin được nhận dạng. Ví dụ, khi khối A được nhận dạng và khối A có hai điểm điều khiển, như được minh họa trong FIG.47B, bộ dự đoán liên đới 126 tính toán vector chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại và vector chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại từ các vector chuyển động v_3 và v_4 tại góc phía trên-bên trái của khối được mã hóa bao gồm khối A và góc phía trên-bên phải của khối được mã hóa. Ví dụ, bộ dự đoán liên đới 126 tính toán vector chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại và vector

chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại bằng cách chiếu các vector chuyển động v_3 và v_4 tại góc phía trên-bên trái và góc phía trên-bên phải của khối được mã hóa lên trên khối hiện tại.

Ngoài ra, khi khối A được nhận dạng và khối A có ba điểm điều khiển, như được minh họa trong FIG.47C, bộ dự đoán liên đới 126 tính toán vector chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại, vector chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại, và vector chuyển động v_2 tại điểm điều khiển góc phía dưới-bên trái của khối hiện tại từ các vector chuyển động v_3 , v_4 , và v_5 tại góc phía trên-bên trái của khối được mã hóa bao gồm khối A, góc phía trên-bên phải của khối được mã hóa, và góc phía dưới-bên trái của khối được mã hóa. Ví dụ, bộ dự đoán liên đới 126 tính toán vector chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại, vector chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại, và vector chuyển động v_2 tại điểm điều khiển góc phía dưới-bên trái của khối hiện tại bằng cách chiếu các vector chuyển động v_3 , v_4 , và v_5 tại góc phía trên-bên trái, góc phía trên-bên phải, và góc phía dưới-bên trái của khối được mã hóa lên trên khối hiện tại.

Cần lưu ý rằng, như được minh họa trong FIG.49A được mô tả nêu trên, các MV tại ba điểm điều khiển có thể được tính toán khi khối A được nhận dạng và khối A có hai điểm điều khiển, và như được minh họa trong FIG.49B được mô tả nêu trên, các MV tại hai điểm điều khiển có thể được tính toán khi khối A được nhận dạng và khối A có ba điểm điều khiển.

Tiếp theo, bộ dự đoán liên đới 126 thực hiện việc bù chuyển động của mỗi khối con được chứa trong khối hiện tại. Nói cách khác, bộ dự đoán liên đới 126 tính toán MV đối với mỗi khối con như là MV afin, ví dụ sử dụng hai vector chuyển động v_0 và v_1 và biểu thức (1A) nêu trên hoặc ba vector chuyển động v_0 , v_1 , và v_2 và biểu thức (1B) nêu trên (bước Sk_2). Sau đó, bộ dự đoán liên đới 126 thực hiện việc bù chuyển động của các khối con sử dụng các MV afin này và các ảnh tham chiếu được mã hóa (bước Sk_3). Khi các xử lý trong các bước Sk_2 và Sk_3 được thực thi đối với mỗi tất cả khối con được chứa trong khối hiện tại, xử lý để tạo ra ảnh dự đoán sử dụng chế độ hợp nhất afin đối với khối hiện tại hoàn thành. Nói cách khác, việc

bù chuyển động của khối hiện tại được thực hiện để tạo ra ảnh dự đoán của khối hiện tại.

Cần lưu ý rằng danh sách ứng viên MV nêu trên có thể được tạo ra trong bước Sk_1. Danh sách ứng viên MV có thể là, ví dụ, danh sách bao gồm các ứng viên MV thu được nhờ sử dụng các phương pháp thu nhận MV đối với mỗi điểm điều khiển. Các phương pháp thu nhận MV có thể là, ví dụ, bất kỳ kết hợp của các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C, các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B, các phương pháp thu nhận MV được minh họa trong các Fig.49A và Fig.49B, và các phương pháp thu nhận MV khác.

Cần lưu ý rằng các danh sách ứng viên MV có thể bao gồm các ứng viên MV trong chế độ trong đó việc dự đoán được thực hiện trong các đơn vị của khối con, ngoài chế độ afin.

Cần lưu ý rằng, ví dụ, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra như là danh sách ứng viên MV. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra một cách riêng biệt. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong một trong số chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra. Các ứng viên MV có thể là, ví dụ, các MV đối với khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được mã hóa, hoặc MV đối với khối có hiệu lực trong số các khối.

Cần lưu ý rằng chỉ số mà chỉ báo một trong số các MV trong danh sách ứng viên MV có thể được truyền như là thông tin lựa chọn M.

(Thu nhận MV > chế độ afin > chế độ liên đới afin)

FIG.51 là lưu đồ minh họa một ví dụ về xử lý trong chế độ liên đới afin.

Trong chế độ liên đới afin, đầu tiên, bộ dự đoán liên đới 126 thu nhận các bộ dự đoán liên đới MV (v_0, v_1) hoặc (v_0, v_1, v_2) của hai hoặc ba điểm điều khiển tương ứng đối với khối hiện tại (bước Sj_1). Các điểm điều khiển có thể là, ví dụ, điểm góc phía trên-bên trái đối với khối hiện tại, điểm góc phía trên-bên phải của khối hiện tại, và điểm góc phía trên-bên phải đối với khối hiện tại như được minh họa trong FIG.46A hoặc FIG.46B.

Ví dụ, khi các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B được sử dụng, bộ dự đoán liên đới 126 thu nhận các bộ dự đoán MV (v_0, v_1) hoặc (v_0, v_1, v_2) tại hai hoặc ba điểm điều khiển tương ứng đối với khối hiện tại bằng cách lựa chọn các MV của bất kỳ khối trong số các khối được mã hóa trong lân cận của các điểm điều khiển tương ứng đối với khối hiện tại được minh họa trong FIG.48A hoặc FIG.48B. Lúc này, bộ dự đoán liên đới 126 mã hóa, trong dòng, thông tin lựa chọn bộ dự đoán MV để nhận dạng hai hoặc ba bộ dự đoán MV được lựa chọn.

Ví dụ, bộ dự đoán liên đới 126 có thể xác định, nhờ sử dụng việc đánh giá giá trị hoặc loại tương tự, khối mà từ đó MV như là bộ dự đoán MV tại điểm điều khiển được lựa chọn từ trong số các khối được mã hóa mà lân cận khối hiện tại, và có thể ghi, trong dòng bit, cờ mà chỉ báo bộ dự đoán MV nào được lựa chọn. Nói cách khác, bộ dự đoán liên đới 126 xuất ra, như là tham số dự đoán, thông tin lựa chọn bộ dự đoán MV như cờ tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Tiếp theo, bộ dự đoán liên đới 126 thực hiện ước lượng chuyển động (bước Sj_3 và Sj_4) trong khi cập nhật bộ dự đoán MV được lựa chọn hoặc được thu nhận trong bước Sj_1 (bước Sj_2). Nói cách khác, bộ dự đoán liên đới 126 tính toán, như là MV afin, MV của mỗi khối con mà tương ứng với bộ dự đoán MV được cập nhật, sử dụng biểu thức (1A) hoặc biểu thức (1B) được mô tả nêu trên (bước Sj_3). Sau đó, bộ dự đoán liên đới 126 thực hiện việc bù chuyển động của các khối con sử dụng các MV afin này và các ảnh tham chiếu được mã hóa (bước Sj_4). Các xử lý trong bước Sj_3 và Sj_4 được thực thi trên tất cả các khối trong khối hiện tại khi bộ dự đoán MV được cập nhật trong bước Sj_2. Kết quả là, ví dụ, bộ dự đoán liên đới 126

xác định bộ dự đoán MV mà có giá trị nhỏ nhất như là MV tại điểm điều khiển trong vòng ước lượng chuyển động (bước Sj_5). Lúc này, bộ dự đoán liên đới 126 còn mã hóa, trong dòng, giá trị chênh lệch giữa MV được xác định và các bộ dự đoán MV như là độ chênh lệch MV. Nói cách khác, bộ dự đoán liên đới 126 xuất ra độ chênh lệch MV như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Cuối cùng, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện bù chuyển động của khối hiện tại sử dụng MV được xác định và ảnh tham chiếu được mã hóa (bước Sj_6).

Cần lưu ý rằng danh sách ứng viên MV nêu trên có thể được tạo ra trong bước Sj_1. Danh sách ứng viên MV có thể là, ví dụ, danh sách bao gồm các ứng viên MV thu được nhờ sử dụng các phương pháp thu nhận MV đối với mỗi điểm điều khiển. Các phương pháp thu nhận MV có thể là, ví dụ, bất kỳ kết hợp của các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C, các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B, các phương pháp thu nhận MV được minh họa trong các Fig.49A và Fig.49B, và các phương pháp thu nhận MV khác.

Cần lưu ý rằng các danh sách ứng viên MV có thể bao gồm các ứng viên MV trong chế độ trong đó việc dự đoán được thực hiện trong các đơn vị của khối con, ngoài chế độ afin.

Cần lưu ý rằng, ví dụ, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ liên đới afin trong đó hai điểm điều khiển được sử dụng và chế độ liên đới afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra như là danh sách ứng viên MV. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ liên đới afin trong đó hai điểm điều khiển được sử dụng và danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ liên đới afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra một cách riêng biệt. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong một trong số chế độ liên đới afin trong đó hai điểm điều khiển được sử dụng và chế độ liên đới afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra. Các ứng viên MV có thể là, ví dụ, các MV

đối với khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được mã hóa, hoặc MV đối với khối có hiệu lực trong số các khối.

Cần lưu ý rằng chỉ số mà chỉ báo một trong số các ứng viên MV trong danh sách ứng viên MV có thể được truyền như là thông tin lựa chọn bộ dự đoán MV.

(Thu nhận MV > chế độ tam giác)

Bộ dự đoán liên đới 126 tạo ra một ảnh dự đoán hình chữ nhật đối với khối hình chữ nhật hiện tại trong ví dụ nêu trên. Tuy nhiên, bộ dự đoán liên đới 126 có thể tạo ra các ảnh dự đoán mỗi chúng có dạng khác với hình chữ nhật đối với khối chữ nhật hiện tại, và có thể kết hợp các ảnh dự đoán để tạo ra ảnh dự đoán chữ nhật cuối cùng. Dạng khác với hình chữ nhật có thể là, ví dụ, hình tam giác.

FIG.52A là sơ đồ khái niệm để minh họa việc tạo ra hai ảnh dự đoán tam giác.

Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán tam giác bằng cách thực hiện việc bù chuyển động của phân vùng thứ nhất mà có dạng tam giác trong khối hiện tại bằng cách sử dụng MV thứ nhất của phân vùng thứ nhất, để tạo ra ảnh dự đoán tam giác. Tương tự, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán tam giác bằng cách thực hiện việc bù chuyển động của phân vùng thứ hai mà có dạng tam giác trong khối hiện tại bằng cách sử dụng MV thứ hai của phân vùng thứ hai, để tạo ra ảnh dự đoán tam giác. Sau đó, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán mà có cùng dạng chữ nhật như dạng chữ nhật của khối hiện tại bằng cách kết hợp các ảnh dự đoán này.

Cần lưu ý rằng ảnh dự đoán thứ nhất mà có dạng chữ nhật tương ứng với khối hiện tại có thể được tạo ra như là ảnh dự đoán đối với phân vùng thứ nhất, nhờ sử dụng MV thứ nhất. Ngoài ra, ảnh dự đoán thứ hai mà có dạng chữ nhật tương ứng với khối hiện tại có thể được tạo ra như là ảnh dự đoán đối với phân vùng thứ hai, nhờ sử dụng MV thứ hai. Ảnh dự đoán đối với khối hiện tại có thể được tạo ra bằng cách thực hiện việc cộng có trọng số của ảnh dự đoán thứ nhất và ảnh dự đoán thứ hai. Cần lưu ý rằng phần mà được đưa vào cộng có trọng số có thể là một phần vùng đi qua biên giữa phân vùng thứ nhất và phân vùng thứ hai.

FIG.52B là sơ đồ khái niệm để minh họa các ví dụ về phần thứ nhất của phân vùng thứ nhất mà chồng lấn với phân vùng thứ hai, và các tập hợp của các mẫu thứ nhất và thứ hai mà có thể được lấy trọng số như là một phần của xử lý hiệu chỉnh. Phần thứ nhất có thể là, ví dụ, một phần tư của độ rộng hoặc độ cao của phân vùng thứ nhất. Theo ví dụ khác, phần thứ nhất có thể có độ rộng tương ứng với N mẫu lân cận với biên của phân vùng thứ nhất, trong đó N là số nguyên lớn hơn 0, ví dụ, N có thể là số nguyên 2. Như được minh họa, ví dụ bên trái của FIG.52B thể hiện phân vùng chữ nhật mà có phần chữ nhật với độ rộng mà là một phần tư của độ rộng của phân vùng thứ nhất, với tập hợp thứ nhất của các mẫu bao gồm các mẫu phía ngoài của phần thứ nhất và các mẫu bên trong của phần thứ nhất, và tập hợp thứ hai của các mẫu bao gồm các mẫu bên trong phần thứ nhất. Ví dụ ở giữa của FIG.52B thể hiện phân vùng chữ nhật mà có phần chữ nhật với độ cao mà là một phần tư của độ cao của phân vùng thứ nhất, với tập hợp thứ nhất của các mẫu bao gồm các mẫu phía ngoài của phần thứ nhất và các mẫu bên trong của phần thứ nhất, và tập hợp thứ hai của các mẫu bao gồm các mẫu bên trong phần thứ nhất. Ví dụ bên phải của FIG.52B thể hiện phân vùng tam giác mà có phần đa giác với độ cao mà tương ứng với hai mẫu, với tập hợp thứ nhất của các mẫu bao gồm các mẫu bên ngoài của phần thứ nhất và các mẫu bên trong của phần thứ nhất, và tập hợp thứ hai của các mẫu bao gồm các mẫu bên trong phần thứ nhất.

Phần thứ nhất có thể là một phần của phân vùng thứ nhất mà chồng lên phân vùng liền kề. FIG.52C là sơ đồ khái niệm để minh họa phần thứ nhất của phân vùng thứ nhất, mà là phần của phân vùng thứ nhất mà chồng lấn với phần của phân vùng liền kề. Nhằm dễ dàng minh họa, phân vùng chữ nhật mà có phần xếp chồng với phân vùng chữ nhật liền kề về mặt không gian được thể hiện. Các phân vùng mà có dạng khác, như các phân vùng tam giác, có thể được sử dụng, và các phần chồng lấn có thể chồng lấn với phân vùng lân cận về mặt không gian hoặc thời gian.

Ngoài ra, mặc dù ví dụ được đưa ra trong đó ảnh dự đoán được tạo ra đối với mỗi hai phân vùng sử dụng việc dự đoán liên đới, ảnh dự đoán có thể được tạo ra đối với ít nhất một phân vùng sử dụng việc nội dự đoán.

FIG.53 là lưu đồ minh họa một ví dụ về xử lý trong chế độ tam giác.

Đầu tiên, trong chế độ tam giác, bộ dự đoán liên đới 126 phân chia khối hiện tại thành phân vùng thứ nhất và phân vùng thứ hai (bước Sx_1). Lúc này, bộ dự đoán liên đới 126 có thể mã hóa, trong dòng, thông tin phân vùng mà là thông tin liên quan đến việc phân chia thành các phân vùng như là tham số dự đoán. Nói cách khác, bộ dự đoán liên đới 126 có thể xuất ra thông tin phân vùng như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Đầu tiên, bộ dự đoán liên đới 126 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được mã hóa theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sx_2). Nói cách khác, bộ dự đoán liên đới 126 tạo ra danh sách ứng viên MV.

Sau đó, bộ dự đoán liên đới 126 lựa chọn ứng viên MV đối với phân vùng thứ nhất và ứng viên MV đối với phân vùng thứ hai như là MV thứ nhất và MV thứ hai, một cách lần lượt, từ các ứng viên MV thu được trong bước Sx_1 (bước Sx_3). Lúc này, bộ dự đoán liên đới 126 mã hóa, trong dòng, thông tin lựa chọn MV để nhận dạng ứng viên MV được lựa chọn như là tham số dự đoán. Nói cách khác, bộ dự đoán liên đới 126 xuất ra thông tin lựa chọn MV như là tham số dự đoán tới bộ mã hóa entropy 110 thông qua bộ tạo tham số dự đoán 130.

Tiếp theo, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán thứ nhất bằng cách thực hiện việc bù chuyển động nhờ sử dụng MV thứ nhất được lựa chọn và ảnh tham chiếu được mã hóa (bước Sx_4). Tương tự, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán thứ hai bằng cách thực hiện việc bù chuyển động nhờ sử dụng MV thứ hai được lựa chọn và ảnh tham chiếu được mã hóa (bước Sx_5).

Cuối cùng, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc cộng có trọng số của ảnh dự đoán thứ nhất và ảnh dự đoán thứ hai (bước Sx_6).

Cần lưu ý rằng, mặc dù phân vùng thứ nhất và phân vùng thứ hai là các hình tam giác trong ví dụ được minh họa trong FIG.52A, phân vùng thứ nhất và phân vùng thứ hai có thể là các hình thang, hoặc các dạng khác khác nhau. Ngoài ra, mặc dù khối hiện tại bao gồm hai phân vùng trong các ví dụ được minh họa trong các Fig.52A và Fig.52C, khối hiện tại có thể bao gồm ba phân vùng hoặc lớn hơn.

Ngoài ra, phân vùng thứ nhất và phân vùng thứ hai có thể chồng lấn nhau. Nói cách khác, phân vùng thứ nhất và phân vùng thứ hai có thể bao gồm cùng vùng điểm ảnh. Trong trường hợp này, ảnh dự đoán đối với khối hiện tại có thể được tạo ra nhờ sử dụng ảnh dự đoán trong phân vùng thứ nhất và ảnh dự đoán trong phân vùng thứ hai.

Ngoài ra, mặc dù ví dụ trong đó ảnh dự đoán được tạo ra đối với mỗi hai phân vùng sử dụng việc dự đoán liên đới đã được minh họa, ảnh dự đoán có thể được tạo ra đối với ít nhất một phân vùng sử dụng việc nội dự đoán.

Cần lưu ý rằng danh sách ứng viên MV để lựa chọn MV thứ nhất và danh sách ứng viên MV để lựa chọn MV thứ hai có thể khác nhau, hoặc danh sách ứng viên MV để lựa chọn MV thứ nhất có thể cũng được sử dụng như là danh sách ứng viên MV để lựa chọn MV thứ hai.

Cần lưu ý rằng thông tin phân vùng có thể bao gồm chỉ số mà chỉ báo chiều phân chia trong đó ít nhất khối hiện tại được phân chia thành nhiều phân vùng. Thông tin lựa chọn MV có thể bao gồm chỉ số mà chỉ báo MV thứ nhất được lựa chọn và chỉ số mà chỉ báo MV thứ hai được lựa chọn. Một chỉ số có thể chỉ báo nhiều đoạn thông tin. Ví dụ, một chỉ số mà chỉ báo chung một phần hoặc toàn bộ thông tin phân vùng và một phần hoặc toàn bộ thông tin lựa chọn MV có thể được mã hóa.

(Thu nhận MV > Chế độ ATMVP)

FIG.54 là sơ đồ khái niệm để minh họa một ví dụ về chế độ dự đoán vector chuyển động theo thời gian cải tiến (ATMVP-Advanced Temporal Motion Vector Prediction) trong đó MV được thu nhận trong các đơn vị của khối con.

Chế độ ATMVP là chế độ được phân loại thành chế độ hợp nhất. Ví dụ, trong chế độ ATMVP, ứng viên MV đối với mỗi khối con được ghi trong danh sách ứng viên MV để sử dụng trong chế độ hợp nhất thường.

Cụ thể hơn, trong chế độ ATMVP, đầu tiên, như được minh họa trong FIG.54, khối tham chiếu MV theo thời gian được kết hợp với khối hiện tại được nhận dạng trong ảnh tham chiếu được mã hóa được chỉ rõ bởi MV (MV0) của khối lân cận nằm

tại vị trí phía dưới-bên trái so với khối hiện tại. Tiếp theo, trong mỗi khối con trong khối hiện tại, MV được sử dụng để mã hóa vùng tương ứng với khối con trong khối tham chiếu MV theo thời gian được nhận dạng. MV được nhận dạng theo cách này được chứa trong danh sách ứng viên MV như là ứng viên MV đối với khối con trong khối hiện tại. Khi ứng viên MV đối với mỗi khối con được lựa chọn từ danh sách ứng viên MV, khối con được bù chuyển động trong đó ứng viên MV được sử dụng như là MV đối với khối con. Theo cách này, ảnh dự đoán đối với mỗi khối con được tạo ra.

Mặc dù khối nằm tại vị trí phía dưới-bên trái so với khối hiện tại được sử dụng như là khối tham chiếu MV xung quanh trong ví dụ được minh họa trong FIG.54, cần lưu ý rằng khối khác có thể được sử dụng. Ngoài ra, kích cỡ của khối con có thể là 4×4 điểm ảnh, 8×8 điểm ảnh, hoặc kích cỡ khác. Kích cỡ của khối con có thể được chuyển đổi đối với đơn vị như lát, viên, ảnh, v.v.

(Thu nhận MV > DMVR)

FIG.55 là lưu đồ minh họa quan hệ giữa chế độ hợp nhất và tinh chỉnh vector chuyển động bộ giải mã (DMVR-Decoder Motion Vector Refinement).

Bộ dự đoán liên đới 126 thu nhận vector chuyển động đối với khối hiện tại theo chế độ hợp nhất (bước S1_1). Tiếp theo, bộ dự đoán liên đới 126 xác định rằng có thực hiện hay không việc ước lượng của vector chuyển động, tức là, ước lượng chuyển động (bước S1_2). Ở đây, khi xác định không thực hiện ước lượng chuyển động (Không trong bước S1_2), bộ dự đoán liên đới 126 xác định vector chuyển động thu được trong bước S1_1 như là vector chuyển động cuối cùng đối với khối hiện tại (bước S1_4). Nói cách khác, trong trường hợp này, vector chuyển động của khối hiện tại được xác định theo chế độ hợp nhất.

Khi xác định thực hiện ước lượng chuyển động trong bước S1_1 (Có trong bước S1_2), bộ dự đoán liên đới 126 thu nhận vector chuyển động cuối cùng đối với khối hiện tại bằng cách ước lượng vùng xung quanh của ảnh tham chiếu được chỉ rõ bởi vector chuyển động thu được trong bước S1_1 (bước S1_3). Nói cách khác, trong trường hợp này, vector chuyển động của khối hiện tại được xác định theo DMVR.

FIG.56 là sơ đồ khái niệm để minh họa một ví dụ về xử lý DMVR để xác định MV.

Đầu tiên, trong chế độ hợp nhất ví dụ, các ứng viên MV (L0 và L1) được lựa chọn đối với khối hiện tại. Điểm ảnh tham chiếu được nhận dạng từ ảnh tham chiếu thứ nhất (L0) mà là ảnh được mã hóa trong danh sách L0 theo ứng viên MV (L0). Tương tự, điểm ảnh tham chiếu được nhận dạng từ ảnh tham chiếu thứ hai (L1) mà là ảnh được mã hóa trong danh sách L1 theo ứng viên MV (L1). Khuôn mẫu được tạo ra bằng cách tính toán giá trị trung bình của các điểm ảnh tham chiếu này.

Tiếp theo, mỗi vùng xung quanh của các ứng viên MV của ảnh tham chiếu thứ nhất (L0) và ảnh tham chiếu thứ hai (L1) được ước lượng nhờ sử dụng khuôn mẫu, và MV mà có giá trị nhỏ nhất được xác định là MV cuối cùng. Cần lưu ý rằng giá trị có thể được tính toán, ví dụ, sử dụng giá trị chênh lệch giữa mỗi các giá trị điểm ảnh trong khuôn mẫu và một trong số các giá trị điểm ảnh tương ứng trong vùng ước lượng, các giá trị của các ứng viên MV, v.v.

Các xử lý tương tự được mô tả ở đây không cần luôn được thực hiện. Các xử lý khác để thực hiện việc thu nhận MV cuối cùng bằng cách ước lượng trong các vùng xung quanh của các ứng viên MV có thể được sử dụng.

FIG.57 là sơ đồ khái niệm để minh họa ví dụ khác về DMVR để xác định MV. Không giống như ví dụ về DMVR được minh họa trong FIG.56, trong ví dụ được minh họa trong FIG.57, các giá trị được tính toán mà không tạo ra khuôn mẫu.

Đầu tiên, bộ dự đoán liên đới 126 ước lượng vùng xung quanh của khối tham chiếu được chứa trong mỗi ảnh tham chiếu trong danh sách L0 và danh sách L1, dựa trên MV khởi tạo mà là ứng viên MV thu được từ mỗi danh sách ứng viên MV. Ví dụ, như được minh họa trong FIG.57, MV khởi tạo tương ứng với khối tham chiếu trong danh sách L0 là `InitMV_L0`, và MV khởi tạo tương ứng với khối tham chiếu trong danh sách L1 là `InitMV_L1`. Trong ước lượng chuyển động, bộ dự đoán liên đới 126 đầu tiên thiết lập vị trí tìm kiếm đối với ảnh tham chiếu trong danh sách L0. Dựa trên vị trí được chỉ báo bởi độ chênh lệch vector mà chỉ báo vị trí tìm kiếm cần được thiết lập, cụ thể, MV khởi tạo (tức là, `InitMV_L0`, độ chênh lệch vector tới vị trí tìm kiếm là `MVd_L0`. Sau đó, bộ dự đoán liên đới 126 xác định vị trí ước lượng

trong ảnh tham chiếu trong danh sách L1. Vị trí tìm kiếm này được chỉ báo bởi độ chênh lệch vector tới vị trí tìm kiếm từ vị trí được chỉ báo bởi MV khởi tạo (tức là, InitMV_L1). Cụ thể hơn, bộ dự đoán liên đới 126 xác định độ chênh lệch vector như là MVd_L1 bằng cách phản chiếu của MVd_L0. Nói cách khác, bộ dự đoán liên đới 126 xác định vị trí mà đối xứng với vị trí được chỉ báo bởi MV khởi tạo là vị trí tìm kiếm trong mỗi ảnh tham chiếu trong danh sách L0 và danh sách L1. Bộ dự đoán liên đới 126 tính toán, đối với mỗi vị trí tìm kiếm, tổng của các độ chênh lệch tuyệt đối (SAD) giữa các giá trị của các điểm ảnh tại các vị trí tìm kiếm trong các khối như là giá trị, và tìm ra vị trí tìm kiếm mà có giá trị nhỏ nhất.

FIG.58A là sơ đồ khái niệm để minh họa một ví dụ về ước lượng chuyển động trong DMVR, và FIG.58B là lưu đồ minh họa một ví dụ về xử lý ước lượng chuyển động.

Đầu tiên, trong bước 1, bộ dự đoán liên đới 126 tính toán giá trị giữa vị trí tìm kiếm (cũng được gọi là điểm bắt đầu) được chỉ báo bởi MV khởi tạo và tám vị trí tìm kiếm xung quanh. Sau đó, bộ dự đoán liên đới 126 xác định rằng giá trị tại mỗi vị trí tìm kiếm khác ngoài điểm bắt đầu có phải là nhỏ nhất hay không. Ở đây, khi xác định rằng giá trị tại vị trí tìm kiếm khác ngoài điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 126 thay đổi đích tới vị trí tìm kiếm mà tại đó giá trị nhỏ nhất được thu nhận, và thực hiện xử lý trong bước 2. Khi giá trị tại điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 126 bỏ qua xử lý trong bước 2 và thực hiện xử lý trong bước 3.

Trong bước 2, bộ dự đoán liên đới 126 thực hiện việc tìm kiếm tương tự xử lý trong bước 1, xem xét, như là điểm bắt đầu mới, vị trí tìm kiếm sau khi thay đổi đích theo kết quả của xử lý trong bước 1. Sau đó, bộ dự đoán liên đới 126 xác định rằng giá trị tại mỗi vị trí tìm kiếm khác ngoài điểm bắt đầu có phải là nhỏ nhất hay không. Ở đây, khi xác định rằng giá trị tại vị trí tìm kiếm khác ngoài điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 126 thực hiện xử lý trong bước 4. Khi giá trị tại điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 126 thực hiện xử lý trong bước 3.

Trong bước 4, bộ dự đoán liên đới 126 xem xét vị trí tìm kiếm tại điểm bắt đầu như là vị trí tìm kiếm cuối cùng, và xác định độ chênh lệch giữa vị trí được chỉ báo bởi MV khởi tạo và vị trí tìm kiếm cuối cùng là độ chênh lệch vector.

Trong bước 3, bộ dự đoán liên đới 126 xác định vị trí điểm ảnh tại độ chính xác điểm ảnh con mà tại đó giá trị nhỏ nhất được thu nhận, dựa trên các giá trị tại bốn điểm nằm tại các vị trí phía trên, phía dưới, bên trái, và bên phải so với điểm bắt đầu trong bước 1 hoặc bước 2, và xem xét vị trí điểm ảnh như là vị trí tìm kiếm cuối cùng. Vị trí điểm ảnh tại độ chính xác điểm ảnh con được xác định bằng cách thực hiện việc cộng có trọng số của mỗi bốn vector phía trên, phía dưới, bên trái, và bên phải $((0, 1), (0, -1), (-1, 0), \text{ và } (1, 0))$, sử dụng, như là trọng số, giá trị tại một trong số bốn vị trí tìm kiếm tương ứng. Sau đó, bộ dự đoán liên đới 126 xác định độ chênh lệch giữa vị trí được chỉ báo bởi MV khởi tạo và vị trí tìm kiếm cuối cùng là độ chênh lệch vector.

(Bù chuyển động > BIO/OBMC/LIC)

Bù chuyển động bao gồm chế độ để tạo ra ảnh dự đoán, và hiệu chỉnh ảnh dự đoán. Chế độ này là, ví dụ, dòng quang hai chiều (BIO), bù chuyển động khối chồng lấn (OBMC), bù độ chói cục bộ (LIC), được mô tả sau đây, v.v.

FIG.59 là lưu đồ minh họa một ví dụ về xử lý tạo ra ảnh dự đoán.

Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán (bước Sm_1), và hiệu chỉnh ảnh dự đoán, ví dụ, theo, ví dụ, bất kỳ chế độ được mô tả nêu trên (bước Sm_2).

FIG.60 là lưu đồ minh họa ví dụ khác về xử lý tạo ra ảnh dự đoán.

Bộ dự đoán liên đới 126 xác định vector chuyển động của khối hiện tại (bước Sn_1). Tiếp theo, bộ dự đoán liên đới 126 tạo ra ảnh dự đoán nhờ sử dụng vector chuyển động (bước Sn_2), và xác định có thực hiện xử lý hiệu chỉnh hay không (bước Sn_3). Ở đây, khi xác định thực hiện xử lý hiệu chỉnh (Có trong bước Sn_3), bộ dự đoán liên đới 126 tạo ra ảnh dự đoán cuối cùng bằng cách hiệu chỉnh ảnh dự đoán (bước Sn_4). Cần lưu ý rằng, trong LIC được mô tả sau đây, độ chói và sắc độ có thể được hiệu chỉnh trong bước Sn_4. Khi xác định không thực hiện xử lý hiệu chỉnh (Không trong bước Sn_3), bộ dự đoán liên đới 126 xuất ra ảnh dự đoán như là ảnh dự đoán cuối cùng mà không hiệu chỉnh ảnh dự đoán (bước Sn_5).

(Bù chuyển động > OBMC)

Cần lưu ý rằng ảnh dự đoán liên đới có thể được tạo ra nhờ sử dụng thông tin chuyển động đối với khối lân cận ngoài thông tin chuyển động đối với khối hiện tại thu được bằng cách ước lượng chuyển động. Cụ thể hơn, ảnh dự đoán liên đới có thể được tạo ra đối với mỗi khối con trong khối hiện tại bằng cách thực hiện việc cộng có trọng số của ảnh dự đoán dựa trên thông tin chuyển động thu được bằng cách ước lượng chuyển động (trong ảnh tham chiếu) và ảnh dự đoán dựa trên thông tin chuyển động của khối lân cận (trong ảnh hiện tại). Việc dự đoán liên đới này (bù chuyển động) cũng được gọi là bù chuyển động khối chồng lấn (OBMC-overlapped block motion compensation) hoặc chế độ OBMC.

Trong chế độ OBMC, thông tin mà chỉ báo kích cỡ khối con đối với OBMC (được gọi là, ví dụ, kích cỡ khối OBMC) có thể được báo hiệu tại cấp chuỗi. Ngoài ra, thông tin mà chỉ báo rằng áp dụng chế độ OBMC hoặc không (được gọi là, ví dụ, cờ OBMC) có thể được báo hiệu tại cấp CU. Cần lưu ý rằng việc báo hiệu của thông tin này không cần được thực hiện tại cấp chuỗi và cấp CU, và có thể được thực hiện tại cấp khác (ví dụ, tại cấp ảnh, cấp lát, cấp viên, cấp CTU hoặc cấp khối con).

Chế độ OBMC sẽ được mô tả chi tiết hơn. Các Fig.61 và Fig.62 là lưu đồ và sơ đồ khái niệm để minh họa tổng quát của xử lý hiệu chỉnh ảnh dự đoán được thực hiện bởi OBMC.

Đầu tiên, như được minh họa trong FIG.62, ảnh dự đoán (Pred) bằng cách bù chuyển động thường được thu nhận nhờ sử dụng MV được gán tại khối hiện tại. Trong FIG.62, mũi tên “MV” trỏ tới ảnh tham chiếu, và chỉ báo khối hiện tại nào của ảnh hiện tại viên dẫn tới để thu nhận ảnh dự đoán.

Tiếp theo, ảnh dự đoán (Pred_L) được thu nhận bằng cách áp dụng vector chuyển động (MV_L) mà đã được thu nhận đối với khối được mã hóa mà lân cận với bên trái của khối hiện tại tới khối hiện tại (sử dụng lại vector chuyển động đối với khối hiện tại). Vector chuyển động (MV_L) được chỉ báo bởi mũi tên “MV_L” mà chỉ báo ảnh tham chiếu từ khối hiện tại. Việc hiệu chỉnh thứ nhất của ảnh dự đoán được thực hiện bằng cách xếp chồng hai ảnh dự đoán Pred và Pred_L. Điều này mang lại hiệu quả trộn lẫn biên giữa các khối lân cận.

Tương tự, ảnh dự đoán (Pred_U) được thu nhận bằng cách áp dụng MV (MV_U) mà đã được thu nhận đối với khối được mã hóa mà lân cận phía trên khối hiện tại tới khối hiện tại (tái sử dụng MV đối với khối hiện tại). MV (MV_U) được chỉ báo bởi mũi tên “MV_U” mà chỉ báo ảnh tham chiếu từ khối hiện tại. Việc hiệu chỉnh thứ hai của ảnh dự đoán được thực hiện bằng cách xếp chồng ảnh dự đoán Pred_U tới các ảnh dự đoán (ví dụ, Pred và Pred_L) mà trên đó việc hiệu chỉnh thứ nhất đã được thực hiện. Điều này mang lại hiệu quả trộn lẫn biên giữa các khối lân cận. Ảnh dự đoán thu được bởi việc hiệu chỉnh thứ hai là ảnh trong đó biên giữa các khối lân cận đã được trộn (được san bằng), và do đó là ảnh dự đoán cuối cùng của khối hiện tại.

Mặc dù ví dụ nêu trên là phương pháp hiệu chỉnh hai đường sử dụng các khối lân cận bên trái và phía trên, cần lưu ý rằng phương pháp hiệu chỉnh có thể là phương pháp hiệu chỉnh ba đường hoặc nhiều hơn nhờ sử dụng cũng khối lân cận bên phải và/hoặc khối lân cận phía dưới.

Cần lưu ý rằng vùng trong đó việc xếp chồng này được thực hiện có thể chỉ là một phần của vùng gần biên khối thay vì vùng điểm ảnh của toàn bộ khối.

Cần lưu ý rằng xử lý hiệu chỉnh ảnh dự đoán theo OBMC để thu nhận một ảnh dự đoán Pred từ một ảnh tham chiếu bằng cách xếp chồng ảnh dự đoán bổ sung Pred_L và Pred_U đã được mô tả nêu trên. Tuy nhiên, khi ảnh dự đoán được hiệu chỉnh dựa trên nhiều ảnh tham chiếu, xử lý tương tự có thể được áp dụng tới mỗi ảnh tham chiếu. Trong trường hợp này, sau khi các ảnh dự đoán được hiệu chỉnh được thu nhận từ các ảnh tham chiếu tương ứng bằng cách thực hiện hiệu chỉnh ảnh OBMC dựa trên các ảnh tham chiếu, các ảnh dự đoán được hiệu chỉnh thu được được xếp chồng tiếp để thu nhận ảnh dự đoán cuối cùng.

Cần lưu ý rằng, trong OBMC, đơn vị khối hiện tại có thể là PU hoặc đơn vị khối con thu được bằng cách phân chia tiếp PU.

Một ví dụ về phương pháp để xác định rằng có áp dụng OBMC hay không là phương pháp để sử dụng obmc_flag mà là tín hiệu mà chỉ báo rằng có áp dụng OBMC hay không. Theo một ví dụ cụ thể, bộ mã hóa 100 có thể xác định rằng khối hiện tại có thuộc về vùng mà có chuyển động phức tạp hay không. Bộ mã hóa 100 thiết lập

obmc_flag là giá trị bằng “1” khi khối thuộc về vùng mà có chuyển động phức tạp và áp dụng OBMC khi mã hóa, và thiết lập obmc_flag thành giá trị bằng “0” khi khối không thuộc về vùng mà có chuyển động phức tạp và mã hóa khối mà không áp dụng OBMC. Bộ giải mã 200 chuyển đổi giữa việc áp dụng và không áp dụng OBMC bằng cách giải mã obmc_flag được ghi trong dòng.

(Bù chuyển động > BIO)

Tiếp theo, phương pháp thu nhận MV được mô tả. Đầu tiên, chế độ thu nhận MV dựa trên mô hình giả định chuyển động tuyến tính đồng đều được mô tả. Chế độ này cũng được gọi là chế độ dòng quang hai chiều (BIO). Ngoài ra, dòng quang hai chiều này có thể được ghi là BDOF thay vì BIO.

FIG.63 là sơ đồ khái niệm để minh họa mô hình giả định chuyển động tuyến tính đồng đều. Trong FIG.63, (v_x, v_y) chỉ báo vector vận tốc, và τ_0 và τ_1 ký hiệu các khoảng cách theo thời gian giữa ảnh hiện tại (Cur Pic) và hai ảnh tham chiếu (Ref_0 , Ref_1). (MV_{x0}, MV_{y0}) chỉ báo các MV tương ứng với ảnh tham chiếu Ref_0 , và (MV_{x1}, MV_{y1}) chỉ báo các MV tương ứng với ảnh tham chiếu Ref_1 .

Ở đây, dưới giả định của chuyển động tuyến tính đồng đều được biểu diễn bởi vector vận tốc (v_x, v_y) , (MV_{x0}, MV_{y0}) và (MV_{x1}, MV_{y1}) được biểu diễn là $(v_x\tau_0, v_y\tau_0)$ và $(-v_x\tau_1, -v_y\tau_1)$, một cách lần lượt, và công thức dòng quang sau đây (2) được đưa ra.

[Công thức toán 3]

$$\partial I^{(k)} / \partial t + v_x \partial I^{(k)} / \partial x + v_y \partial I^{(k)} / \partial y = 0 \quad (2)$$

Ở đây, $I(k)$ chỉ báo giá trị độ chói được bù chuyển động của ảnh tham chiếu k ($k = 0, 1$) sau khi bù chuyển động. Công thức dòng quang này thể hiện rằng tổng của (i) đạo hàm thời gian của giá trị độ sáng, (ii) tích của vận tốc ngang và thành phần theo chiều ngang của gradient không gian của ảnh tham chiếu, và (iii) tích của vận tốc dọc và thành phần theo chiều dọc của gradient không gian của ảnh tham chiếu, bằng 0 (zero). Vector chuyển động của mỗi khối thu được từ, ví dụ, danh sách ứng viên MV có thể được hiệu chỉnh trong các đơn vị của điểm ảnh dựa trên kết hợp của phương trình dòng quang và phép nội suy Hermite.

Cần lưu ý rằng vector chuyển động có thể thu được trên phía bộ giải mã 200 nhờ sử dụng phương pháp khác ngoài thu nhận vector chuyển động dựa trên mô hình giả định chuyển động tuyến tính đồng đều. Ví dụ, vector chuyển động có thể được thu nhận trong các đơn vị của khối con dựa trên các vector chuyển động của các khối lân cận.

FIG.64 là lưu đồ minh họa một ví dụ về xử lý dự đoán liên đới theo BIO. FIG.65 là sơ đồ khối chức năng minh họa một ví dụ về cấu trúc chức năng của bộ dự đoán liên đới 126 mà có thể thực hiện việc dự đoán liên đới theo BIO.

Như được minh họa trong FIG.65, bộ dự đoán liên đới 126 bao gồm, ví dụ, bộ nhớ 126a, bộ thu nhận ảnh được nội suy 126b, bộ thu nhận ảnh gradient 126c, bộ thu nhận dòng quang 126d, bộ thu nhận giá trị hiệu chỉnh 126e, và bộ hiệu chỉnh ảnh dự đoán 126f. Cần lưu ý rằng bộ nhớ 126a có thể là bộ nhớ khung 122.

Bộ dự đoán liên đới 126 thu nhận hai vector chuyển động (M_0 , M_1), nhờ sử dụng hai ảnh tham chiếu (Ref_0 , Ref_1) khác với ảnh ($Cur\ Pic$) bao gồm khối hiện tại. Sau đó, bộ dự đoán liên đới 126 thu nhận ảnh dự đoán đối với khối hiện tại nhờ sử dụng hai vector chuyển động (M_0 , M_1) (bước Sy_1). Cần lưu ý rằng vector chuyển động M_0 là vector chuyển động (MV_{x0} , MV_{y0}) tương ứng với ảnh tham chiếu Ref_0 , và vector chuyển động M_1 là vector chuyển động (MV_{x1} , MV_{y1}) tương ứng với ảnh tham chiếu Ref_1 .

Tiếp theo, bộ thu nhận ảnh được nội suy 126b thu nhận ảnh được nội suy I^0 đối với khối hiện tại, nhờ sử dụng vector chuyển động M_0 và ảnh tham chiếu L_0 bằng cách viền dẫn tới bộ nhớ 126a. Tiếp theo, bộ thu nhận ảnh được nội suy 126b thu nhận ảnh được nội suy I^1 đối với khối hiện tại, nhờ sử dụng vector chuyển động M_1 và ảnh tham chiếu L_1 bằng cách viền dẫn tới bộ nhớ 126a (bước Sy_2). Ở đây, ảnh được nội suy I^0 là ảnh được chứa trong ảnh tham chiếu Ref_0 và được thu nhận đối với khối hiện tại, và ảnh được nội suy I^1 là ảnh được chứa trong ảnh tham chiếu Ref_1 và được thu nhận đối với khối hiện tại. Mỗi ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể có cùng kích cỡ như khối hiện tại. Ngoài ra, mỗi ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể là ảnh lớn hơn khối hiện tại. Ngoài ra, ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể bao gồm ảnh dự đoán thu được bằng cách sử dụng các

vector chuyển động (M_0, M_1) và các ảnh tham chiếu (L_0, L_1) và áp dụng bộ lọc bù chuyển động.

Ngoài ra, bộ thu nhận ảnh gradient 126c thu nhận các ảnh gradient ($I_x^0, I_x^1, I_y^0, I_y^1$) của khối hiện tại, từ ảnh được nội suy I^0 và ảnh được nội suy I^1 (bước Sy_3). Cần lưu ý rằng các ảnh gradient trong chiều ngang là (I_x^0, I_x^1), và các ảnh gradient trong chiều dọc là (I_y^0, I_y^1). Bộ thu nhận ảnh gradient 126c có thể thu nhận mỗi ảnh gradient bằng cách, ví dụ, áp dụng lọc gradient tới các ảnh được nội suy. Ảnh gradient có thể chỉ báo lượng thay đổi không gian trong giá trị điểm ảnh theo chiều ngang, theo chiều dọc, hoặc cả hai.

Tiếp theo, bộ thu nhận dòng quang 126d thu nhận, đối với mỗi khối con của khối hiện tại, dòng quang (v_x, v_y) mà là vector vận tốc, nhờ sử dụng các ảnh được nội suy (I^0, I^1) và các ảnh gradient ($I_x^0, I_x^1, I_y^0, I_y^1$) (bước Sy_4). Dòng quang chỉ báo các hệ số để hiệu chỉnh lượng di chuyển điểm ảnh không gian, và có thể được gọi là giá trị ước lượng chuyển động cục bộ, vector chuyển động được hiệu chỉnh, hoặc vector trọng số được hiệu chỉnh. Theo một ví dụ, khối con có thể là CU con 4×4 điểm ảnh. Cần lưu ý rằng việc thu nhận dòng quang có thể được thực hiện đối với mỗi đơn vị điểm ảnh, hoặc loại tương tự, thay vì được thực hiện đối với mỗi khối con.

Tiếp theo, bộ dự đoán liên đới 126 hiệu chỉnh ảnh dự đoán đối với khối hiện tại nhờ sử dụng dòng quang (v_x, v_y). Ví dụ, bộ thu nhận giá trị hiệu chỉnh 126e thu nhận giá trị hiệu chỉnh đối với giá trị của điểm ảnh được chứa trong khối hiện tại, nhờ sử dụng dòng quang (v_x, v_y) (bước Sy_5). Sau đó, bộ hiệu chỉnh ảnh dự đoán 126f có thể hiệu chỉnh ảnh dự đoán đối với khối hiện tại nhờ sử dụng giá trị hiệu chỉnh (bước Sy_6). Cần lưu ý rằng giá trị hiệu chỉnh có thể được thu nhận trong các đơn vị của điểm ảnh, hoặc có thể được thu nhận trong các đơn vị của các điểm ảnh hoặc trong các đơn vị của khối con.

Cần lưu ý rằng dòng xử lý BIO không bị giới hạn ở xử lý được bộc lộ trong FIG.64. Ví dụ, chỉ một phần của các xử lý được bộc lộ trong FIG.64 có thể được thực hiện, hoặc xử lý khác có thể được bổ sung hoặc được sử dụng như là sự thay thế, hoặc các xử lý có thể được thực thi trong thứ tự xử lý khác, v.v.

(Bù chuyển động > LIC)

Tiếp theo, một ví dụ của chế độ tạo ra ảnh dự đoán (dự đoán) sử dụng xử lý bù độ chói cục bộ (LIC-local illumination compensation) được mô tả.

FIG.66A là sơ đồ khái niệm để minh họa một ví dụ về xử lý của phương pháp tạo ảnh dự đoán nhờ sử dụng xử lý hiệu chỉnh độ chói được thực hiện bởi LIC. FIG.66B là lưu đồ minh họa một ví dụ về xử lý của phương pháp tạo ảnh dự đoán nhờ sử dụng LIC.

Đầu tiên, bộ dự đoán liên đới 126 thu nhận MV từ ảnh tham chiếu được mã hóa, và thu nhận ảnh tham chiếu tương ứng với khối hiện tại (bước Sz_1).

Tiếp theo, bộ dự đoán liên đới 126 tách, đối với khối hiện tại, thông tin mà chỉ báo giá trị độ chói đã thay đổi như thế nào giữa khối hiện tại và ảnh tham chiếu (bước Sz_2). Việc tách này được thực hiện dựa trên các giá trị điểm ảnh độ chói của vùng tham chiếu lân cận bên trái được mã hóa (vùng tham chiếu xung quanh) và vùng tham chiếu lân cận phía trên được mã hóa (vùng tham chiếu xung quanh) trong ảnh hiện tại, và các giá trị điểm ảnh độ chói tại các vị trí tương ứng trong ảnh tham chiếu được chỉ rõ bởi các MV được thu nhận. Bộ dự đoán liên đới 126 tính toán tham số hiệu chỉnh độ chói, nhờ sử dụng thông tin mà chỉ báo giá trị độ chói đã thay đổi như thế nào (bước Sz_3).

Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện xử lý hiệu chỉnh độ chói trong đó tham số hiệu chỉnh độ chói được áp dụng tới ảnh tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV (bước Sz_4). Nói cách khác, ảnh dự đoán mà là ảnh tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV được đưa vào hiệu chỉnh dựa trên tham số hiệu chỉnh độ chói. Trong việc hiệu chỉnh này, độ chói có thể được hiệu chỉnh, hoặc sắc độ có thể được hiệu chỉnh, hoặc cả hai. Nói cách khác, tham số hiệu chỉnh sắc độ có thể được tính toán nhờ sử dụng thông tin mà chỉ báo sắc độ đã thay đổi thế nào, và xử lý hiệu chỉnh sắc độ có thể được thực hiện.

Cần lưu ý rằng dạng của vùng tham chiếu xung quanh được minh họa trong FIG.66A là một ví dụ; dạng khác có thể được sử dụng.

Ngoài ra, mặc dù xử lý trong đó ảnh dự đoán được tạo ra từ một ảnh tham chiếu đã được mô tả ở đây, các trường hợp trong đó ảnh dự đoán được tạo ra từ các ảnh tham chiếu có thể được mô tả trong cùng cách thức. Ảnh dự đoán có thể được tạo ra sau khi thực hiện xử lý hiệu chỉnh độ chói của các ảnh tham chiếu thu được từ các ảnh tham chiếu trong cùng cách thức như được mô tả nêu trên.

Một ví dụ của phương pháp xác định rằng có áp dụng LIC hay không là phương pháp sử dụng `lic_flag` mà là tín hiệu chỉ báo rằng có áp dụng LIC hay không. Theo một ví dụ cụ thể, bộ mã hóa 100 xác định rằng khối hiện tại có thuộc về vùng mà có sự thay đổi độ chói hay không. Bộ mã hóa 100 thiết lập `lic_flag` thành giá trị bằng “1” khi khối thuộc về vùng mà có sự thay đổi độ chói và áp dụng LIC khi mã hóa, và thiết lập `lic_flag` thành giá trị bằng “0” khi khối không thuộc về vùng mà có sự thay đổi độ chói và thực hiện việc mã hóa mà không áp dụng LIC. Bộ giải mã 200 có thể giải mã `lic_flag` được ghi trong dòng và giải mã khối hiện tại bằng cách chuyển đổi giữa việc áp dụng và không áp dụng LIC theo giá trị cờ.

Một ví dụ của phương pháp khác để xác định rằng có áp dụng xử lý LIC hay không là phương pháp xác định theo việc xử lý LIC có được áp dụng tới khối xung quanh hay không. Theo một ví dụ cụ thể, khi khối hiện tại đã được xử lý trong chế độ hợp nhất, bộ dự đoán liên đới 126 xác định rằng khối xung quanh được mã hóa được lựa chọn trong việc thu nhận MV trong chế độ hợp nhất có được mã hóa sử dụng LIC hay không. Bộ dự đoán liên đới 126 thực hiện việc mã hóa bằng cách chuyển đổi giữa việc áp dụng và không áp dụng LIC theo kết quả này. Cần lưu ý rằng, cũng trong ví dụ này, các xử lý tương tự được áp dụng trong các xử lý tại phía bộ giải mã 200.

Xử lý hiệu chỉnh độ chói (LIC) đã được mô tả có viện dẫn tới các Fig.66A và Fig.66B, và được mô tả tiếp dưới đây.

Đầu tiên, bộ dự đoán liên đới 126 thu nhận MV để thu nhận ảnh tham chiếu tương ứng với khối hiện tại cần được mã hóa từ ảnh tham chiếu mà là ảnh được mã hóa.

Tiếp theo, bộ dự đoán liên đới 126 tách thông tin mà chỉ báo giá trị độ chói của ảnh tham chiếu đã thay đổi như thế nào thành giá trị độ chói của ảnh hiện tại, sử

dụng các giá trị điểm ảnh độ chói của các vùng tham chiếu xung quanh được mã hóa mà lân cận với bên trái và phía trên khối hiện tại và các giá trị độ chói trong các vị trí tương ứng trong các ảnh tham chiếu được chỉ rõ bởi các MV, và tính toán tham số hiệu chỉnh độ chói. Ví dụ, giả thiết rằng giá trị điểm ảnh độ chói của điểm ảnh định trước trong vùng tham chiếu xung quanh trong ảnh hiện tại là p_0 , và giá trị điểm ảnh độ chói của điểm ảnh tương ứng với điểm ảnh định trước trong vùng tham chiếu xung quanh trong ảnh tham chiếu là p_1 . Bộ dự đoán liên đới 126 tính toán các hệ số A và B để tối ưu hóa $A \times p_1 + B = p_0$ như là tham số hiệu chỉnh độ chói đối với các điểm ảnh trong vùng tham chiếu xung quanh.

Tiếp theo, bộ dự đoán liên đới 126 thực hiện xử lý hiệu chỉnh độ chói sử dụng tham số hiệu chỉnh độ chói đối với ảnh tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV, để tạo ra ảnh dự đoán đối với khối hiện tại. Ví dụ, giả thiết rằng giá trị điểm ảnh độ chói trong ảnh tham chiếu là p_2 , và giá trị điểm ảnh độ chói được hiệu chỉnh độ chói của ảnh dự đoán là p_3 . Bộ dự đoán liên đới 126 tạo ra ảnh dự đoán sau khi được đưa vào xử lý hiệu chỉnh độ chói bằng cách tính toán $A \times p_2 + B = p_3$ đối với mỗi điểm ảnh trong ảnh tham chiếu.

Ví dụ, vùng mà có số lượng điểm ảnh được xác định được tách từ mỗi điểm ảnh lân cận phía trên và điểm ảnh lân cận bên trái có thể được sử dụng như là vùng tham chiếu xung quanh. Ngoài ra, vùng tham chiếu xung quanh không bị giới hạn ở vùng mà lân cận khối hiện tại, và có thể là vùng mà không lân cận khối hiện tại. Trong ví dụ được minh họa trong FIG.66A, vùng tham chiếu xung quanh trong ảnh tham chiếu có thể là vùng được chỉ rõ bởi MV khác trong ảnh hiện tại, từ vùng tham chiếu xung quanh trong ảnh hiện tại. Ví dụ, MV khác có thể là MV trong vùng tham chiếu xung quanh trong ảnh hiện tại.

Mặc dù các hoạt động được thực hiện bởi bộ mã hóa 100 đã được mô tả ở đây, cần lưu ý rằng bộ giải mã 200 thực hiện các hoạt động tương tự.

Cần lưu ý rằng LIC có thể được áp dụng không chỉ tới độ chói mà còn được áp dụng tới sắc độ. Lúc này, tham số hiệu chỉnh có thể được thu nhận một cách riêng biệt đối với mỗi Y , Cb , và Cr , hoặc tham số hiệu chỉnh chung có thể được sử dụng đối với bất kỳ trong số Y , Cb , và Cr .

Ngoài ra, xử lý LIC có thể được áp dụng trong các đơn vị của khối con. Ví dụ, tham chiếu hiệu chỉnh có thể được thu nhận nhờ sử dụng vùng tham chiếu xung quanh trong khối con hiện tại và vùng tham chiếu xung quanh trong khối con tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV của khối con hiện tại.

(Bộ điều khiển dự đoán)

Bộ điều khiển dự đoán 128 lựa chọn một trong số tín hiệu nội dự đoán (ảnh hoặc tín hiệu được xuất ra từ bộ nội dự đoán 124) và tín hiệu dự đoán liên đới (ảnh hoặc tín hiệu được xuất ra từ bộ dự đoán liên đới 126), và xuất ra ảnh dự đoán được lựa chọn tới bộ trừ 104 và bộ cộng 116 như là tín hiệu dự đoán.

(Bộ tạo tham số dự đoán)

Bộ tạo tham số dự đoán 130 có thể xuất ra thông tin liên quan đến việc nội dự đoán, việc dự đoán liên đới, việc lựa chọn ảnh dự đoán trong bộ điều khiển dự đoán 128, v.v. Như là tham số dự đoán tới bộ mã hóa entropy 110. Bộ mã hóa entropy 110 có thể tạo ra dòng, dựa trên tham số dự đoán mà được đưa vào từ bộ tạo tham số dự đoán 130 và các hệ số được lượng tử hóa mà được đưa vào từ bộ lượng tử hóa 108. Tham số dự đoán có thể được sử dụng trong bộ giải mã 200. Bộ giải mã 200 có thể thu và giải mã dòng, và thực hiện các xử lý tương tự như các xử lý dự đoán được thực hiện bởi bộ nội dự đoán 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128. Tham số dự đoán có thể bao gồm, ví dụ, (i) tín hiệu dự đoán lựa chọn (ví dụ, MV, loại dự đoán, hoặc chế độ dự đoán được sử dụng bởi bộ nội dự đoán 124 hoặc bộ dự đoán liên đới 126), hoặc (ii) chỉ số tùy chọn, cờ, hoặc giá trị mà dựa trên xử lý dự đoán được thực hiện trong mỗi bộ nội dự đoán 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128, hoặc chỉ báo xử lý dự đoán.

(Bộ giải mã)

Tiếp theo, bộ giải mã 200 có thể giải mã dòng được xuất ra từ bộ mã hóa 100 được mô tả nêu trên được mô tả. FIG.67 là sơ đồ khối minh họa cấu trúc chức năng của bộ giải mã 200 theo phương án này. Bộ giải mã 200 là thiết bị mà giải mã dòng mà là ảnh được mã hóa trong các đơn vị của khối.

Như được minh họa trong FIG.67, bộ giải mã 200 bao gồm bộ giải mã entropy 202, bộ lượng tử hóa ngược 204, bộ biến đổi ngược 206, bộ cộng 208, bộ nhớ khối 210, bộ lọc vòng 212, bộ nhớ khung 214, bộ nội dự đoán 216, bộ dự đoán liên đới 218, bộ điều khiển dự đoán 220, bộ tạo tham số dự đoán 222, và bộ xác định phân chia 224. Cần lưu ý rằng bộ nội dự đoán 216 và bộ dự đoán liên đới 218 có cấu trúc như là một phần của bộ thực thi dự đoán.

(Ví dụ bố trí của bộ giải mã)

FIG.68 là sơ đồ khối chức năng minh họa ví dụ bố trí của bộ giải mã 200. Bộ giải mã 200 bao gồm bộ xử lý b1 và bộ nhớ b2. Ví dụ, các thành phần cấu thành của bộ giải mã 200 được minh họa trong FIG.67 được bố trí trên bộ xử lý b1 và bộ nhớ b2 được minh họa trong FIG.68.

Bộ xử lý b1 là mạch mà thực hiện việc xử lý thông tin và được ghép nối tới bộ nhớ b2. Ví dụ, bộ xử lý b1 là mạch điện tử chung hoặc dành riêng mà giải mã dòng. Bộ xử lý b1 có thể là bộ xử lý như CPU. Ngoài ra, bộ xử lý b1 có thể là kết hợp của nhiều mạch điện tử. Ngoài ra, ví dụ, bộ xử lý b1 có thể đóng vai trò của hai thành phần cấu thành hoặc nhiều hơn ngoài thành phần cấu thành để lưu trữ thông tin trong số các thành phần cấu thành của bộ giải mã 200 được minh họa trong FIG.67, v.v.

Bộ nhớ b2 là bộ nhớ chung hoặc dành riêng để lưu trữ thông tin mà được sử dụng bởi bộ xử lý b1 để giải mã dòng. Bộ nhớ b2 có thể là mạch điện tử, và có thể được kết nối tới bộ xử lý b1. Ngoài ra, bộ nhớ b2 có thể được chứa trong bộ xử lý b1. Ngoài ra, bộ nhớ b2 có thể là kết hợp của nhiều mạch điện tử. Ngoài ra, bộ nhớ b2 có thể là đĩa từ, đĩa quang, hoặc loại tương tự, hoặc có thể được biểu diễn như là bộ lưu trữ, vật ghi, hoặc loại tương tự. Ngoài ra, bộ nhớ b2 có thể là bộ nhớ bất biến, hoặc bộ nhớ khả biến.

Ví dụ, bộ nhớ b2 có thể lưu trữ ảnh hoặc dòng. Ngoài ra, bộ nhớ b2 có thể lưu trữ chương trình để làm cho bộ xử lý b1 giải mã dòng.

Ngoài ra, ví dụ, bộ nhớ b2 có thể đóng vai trò của hai thành phần cấu thành hoặc nhiều hơn để lưu trữ thông tin ngoài các thành phần cấu thành của bộ giải mã 200 được minh họa trong FIG.67, v.v. Cụ thể hơn, bộ nhớ b2 có thể đóng vai trò của

bộ nhớ khối 210 và bộ nhớ khung 214 được minh họa trong FIG.67. Cụ thể hơn, bộ nhớ b2 có thể lưu trữ ảnh được khôi phục (cụ thể, khối được khôi phục, ảnh được khôi phục, hoặc loại tương tự).

Cần lưu ý rằng, trong bộ giải mã 200, không phải tất cả các thành phần cấu thành được minh họa trong FIG.67, v.v. có thể được thực hiện, và không phải tất cả xử lý được mô tả ở đây có thể được thực hiện. Một phần của các thành phần cấu thành được thể hiện trong FIG.67, v.v. có thể được chứa trong thiết bị khác, hoặc một phần của các xử lý được mô tả ở đây có thể được thực hiện bởi thiết bị khác.

Sau đây, dòng xử lý chung được thực hiện bởi bộ giải mã 200 được mô tả, và sau đó mỗi thành phần cấu thành được chứa trong bộ giải mã 200 được mô tả. Cần lưu ý rằng, một vài thành phần cấu thành được chứa trong bộ giải mã 200 thực hiện các xử lý tương tự như được thực hiện bởi một vài bộ mã hóa 100, và do đó các xử lý tương tự không được mô tả lặp lại chi tiết. Ví dụ, bộ lượng tử hóa ngược 204, bộ biến đổi ngược 206, bộ cộng 208, bộ nhớ khối 210, bộ nhớ khung 214, bộ nội dự đoán 216, bộ dự đoán liên đới 218, bộ điều khiển dự đoán 220, và bộ lọc vòng 212 được chứa trong bộ giải mã 200 thực hiện các xử lý tương tự như được thực hiện bởi bộ lượng tử hóa ngược 112, bộ biến đổi ngược 114, bộ cộng 116, bộ nhớ khối 118, bộ nhớ khung 122, bộ nội dự đoán 124, bộ dự đoán liên đới 126, bộ điều khiển dự đoán 128, và bộ lọc vòng 120 được chứa trong bộ giải mã 200, một cách lần lượt.

(Dòng xử lý giải mã chung)

FIG.69 là lưu đồ minh họa một ví dụ về xử lý giải mã chung được thực hiện bởi bộ giải mã 200.

Đầu tiên, bộ xác định phân chia 224 trong bộ giải mã 200 xác định mẫu phân chia của mỗi khối có kích cỡ cố định (128×128 điểm ảnh) được chứa trong ảnh, dựa trên tham số mà được đưa vào từ bộ giải mã entropy 202 (bước Sp_1). Mẫu phân chia là mẫu phân chia được lựa chọn bởi bộ mã hóa 100. Sau đó, bộ giải mã 200 thực hiện các xử lý của bước Sp_2 đến Sp_6 đối với mỗi khối của mẫu phân chia.

Bộ giải mã entropy 202 giải mã (cụ thể, giải mã entropy) các hệ số được lượng tử hóa được mã hóa và tham số dự đoán của khối hiện tại (bước Sp_2).

Tiếp theo, bộ lượng tử hóa ngược 204 thực hiện việc lượng tử hóa ngược của các hệ số được lượng tử hóa và bộ biến đổi ngược 206 thực hiện việc biến đổi ngược của kết quả, để khôi phục các phần dư dự đoán (tức là, khối chênh lệch) (bước Sp_3).

Tiếp theo, bộ thực thi dự đoán bao gồm tất cả hoặc một phần của bộ nội dự đoán 216, bộ dự đoán liên đới 218, và bộ điều khiển dự đoán 220 tạo ra tín hiệu dự đoán của khối hiện tại (bước Sp_4).

Tiếp theo, bộ cộng 208 cộng ảnh dự đoán với phần dư dự đoán để tạo ra ảnh được khôi phục (cũng được gọi là khối ảnh được giải mã) của khối hiện tại (bước Sp_5).

Khi ảnh được khôi phục được tạo ra, bộ lọc vòng 212 thực hiện việc lọc của ảnh được khôi phục (bước Sp_6).

Sau đó bộ giải mã 200 xác định rằng việc giải mã của toàn bộ ảnh đã được hoàn thành hay chưa (bước Sp_7). Khi xác định rằng việc giải mã chưa được hoàn thành (Không trong bước Sp_7), bộ giải mã 200 lặp lại xử lý bắt đầu với bước Sp_1.

Cần lưu ý rằng các xử lý của các bước này Sp_1 đến Sp_7 có thể được thực hiện tuần tự bởi bộ giải mã 200, hoặc hai xử lý hoặc nhiều hơn có thể được thực hiện một cách song song. Thứ tự xử lý của hai xử lý hoặc nhiều hơn có thể được cải biến.

(Bộ xác định phân chia)

FIG.70 là sơ đồ khái niệm để minh họa quan hệ giữa bộ xác định phân chia 224 và các thành phần cấu thành khác trong phương án. Bộ xác định phân chia 224 có thể thực hiện các xử lý khác nhau như là các ví dụ.

Ví dụ, bộ xác định phân chia 224 thu thập thông tin khối từ bộ nhớ khối 210 hoặc bộ nhớ khung 214, và còn thu nhận tham số từ bộ giải mã entropy 202. Sau đó, bộ xác định phân chia 224 có thể xác định mẫu phân chia của khối có kích cỡ cố định, dựa trên thông tin khối và tham số. Sau đó, bộ xác định phân chia 224 có thể xuất ra thông tin mà chỉ báo mẫu phân chia được xác định tới bộ biến đổi ngược 206, bộ nội dự đoán 216, và bộ dự đoán liên đới 218. Bộ biến đổi ngược 206 có thể thực hiện việc biến đổi ngược của các hệ số biến đổi, dựa trên mẫu phân chia được chỉ báo bởi thông tin từ bộ xác định phân chia 224. Bộ nội dự đoán 216 và bộ dự đoán

liên đới 218 có thể tạo ra ảnh dự đoán, dựa trên mẫu phân chia được chỉ báo bởi thông tin từ bộ xác định phân chia 224.

(Bộ giải mã entropy)

FIG.71 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ giải mã entropy 202.

Bộ giải mã entropy 202 tạo ra các hệ số được lượng tử hóa, tham số dự đoán, và tham số liên quan đến mẫu phân chia, bằng cách giải mã entropy dòng. Ví dụ, CABAC được sử dụng trong việc giải mã entropy. Cụ thể hơn, bộ giải mã entropy 202 bao gồm, ví dụ, bộ giải mã số học nhị phân 202a, bộ điều khiển ngữ cảnh 202b, và bộ giải nhị phân 202c. Bộ giải mã số học nhị phân 202a giải mã số học dòng sử dụng giá trị ngữ cảnh thu được bởi bộ điều khiển ngữ cảnh 202b thành tín hiệu nhị phân. Bộ điều khiển ngữ cảnh 202b thu nhận giá trị ngữ cảnh theo đặc tính hoặc trạng thái xung quanh của phần tử cú pháp, mà là xác suất xuất hiện của tín hiệu nhị phân, trong cùng cách thức như được thực hiện bởi bộ điều khiển ngữ cảnh 110b của bộ mã hóa 100. Bộ giải nhị phân 202c thực hiện việc giải nhị phân để biến đổi tín hiệu nhị phân được xuất ra từ bộ giải mã số học nhị phân 202a thành tín hiệu đa mức mà chỉ báo các hệ số được lượng tử hóa như được mô tả nêu trên. Việc nhị phân hóa có thể được thực hiện theo phương pháp nhị phân hóa được mô tả nêu trên.

Theo đó, bộ giải mã entropy 202 xuất các hệ số được lượng tử hóa của mỗi khối tới bộ lượng tử hóa ngược 204. Bộ giải mã entropy 202 có thể xuất ra tham số dự đoán được chứa trong dòng (xem FIG.1) tới bộ nội dự đoán 216, bộ dự đoán liên đới 218, và bộ điều khiển dự đoán 220. Bộ nội dự đoán 216, bộ dự đoán liên đới 218, và bộ điều khiển dự đoán 220 có thể thực thi các xử lý dự đoán tương tự như được thực hiện bởi bộ nội dự đoán 124, bộ dự đoán liên đới 126, và bộ điều khiển dự đoán 128 tại phía bộ mã hóa 100.

FIG.72 là sơ đồ khái niệm để minh họa dòng xử lý CABAC ví dụ trong bộ giải mã entropy 202.

Đầu tiên, việc khởi tạo được thực hiện trong CABAC trong bộ giải mã entropy 202. Trong việc khởi tạo, việc khởi tạo trong bộ giải mã số học nhị phân 202a và thiết lập của giá trị ngữ cảnh khởi tạo được thực hiện. Bộ giải mã số học nhị phân

202a và bộ giải nhị phân 202c sau đó thực thi việc giải mã số học và giải nhị phân của, ví dụ, dữ liệu được mã hóa của CTU. Lúc này, bộ điều khiển ngữ cảnh 202b cập nhật giá trị ngữ cảnh mỗi khi việc giải mã số học được thực hiện. Sau đó bộ điều khiển ngữ cảnh 202b lưu giá trị ngữ cảnh như là hậu xử lý. Giá trị ngữ cảnh được lưu được sử dụng, ví dụ, để khởi tạo giá trị ngữ cảnh đối với CTU tiếp theo.

(Bộ lượng tử hóa ngược)

Bộ lượng tử hóa ngược 204 lượng tử hóa ngược các hệ số được lượng tử hóa của khối hiện tại mà là các đầu vào từ bộ giải mã entropy 202. Cụ thể hơn, bộ lượng tử hóa ngược 204 lượng tử hóa ngược các hệ số được lượng tử hóa của khối hiện tại dựa trên các tham số lượng tử hóa tương ứng với các hệ số được lượng tử hóa. Bộ lượng tử hóa ngược 204 sau đó xuất các hệ số biến đổi được lượng tử hóa ngược (tức là, các hệ số biến đổi) của khối hiện tại tới bộ biến đổi ngược 206.

FIG.73 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lượng tử hóa ngược 204.

Bộ lượng tử hóa ngược 204 bao gồm, ví dụ, bộ tạo tham số lượng tử hóa 204a, bộ tạo tham số lượng tử hóa được dự đoán 204b, bộ lưu trữ tham số lượng tử hóa 204d, và bộ thực thi lượng tử hóa ngược 204e.

FIG.74 là lưu đồ minh họa một ví dụ về xử lý lượng tử hóa ngược được thực hiện bởi bộ lượng tử hóa ngược 204.

Bộ lượng tử hóa ngược 204 có thể thực hiện xử lý lượng tử hóa ngược như là một ví dụ đối với mỗi CU dựa trên dòng xử lý được minh họa trong FIG.74. Cụ thể hơn, bộ tạo tham số lượng tử hóa 204a xác định rằng có thực hiện việc lượng tử hóa ngược hay không (bước Sv_11). Ở đây, khi xác định thực hiện việc lượng tử hóa ngược (Có trong bước Sv_11), bộ tạo tham số lượng tử hóa 204a thu nhận tham số lượng tử hóa chênh lệch đối với khối hiện tại từ bộ giải mã entropy 202 (bước Sv_12).

Tiếp theo, bộ tạo tham số lượng tử hóa được dự đoán 204b thu nhận tham số lượng tử hóa đối với đơn vị xử lý khác với khối hiện tại từ bộ lưu trữ tham số lượng tử hóa 204d (bước Sv_13). Bộ tạo tham số lượng tử hóa được dự đoán 204b tạo ra

tham số lượng tử hóa được dự đoán của khối hiện tại dựa trên tham số lượng tử hóa thu được (bước Sv_14).

Sau đó, bộ tạo tham số lượng tử hóa 204a tạo ra tham số lượng tử hóa đối với khối hiện tại dựa trên tham số lượng tử hóa chênh lệch đối với khối hiện tại thu được từ bộ giải mã entropy 202 và tham số lượng tử hóa được dự đoán đối với khối hiện tại được tạo ra bởi bộ tạo tham số lượng tử hóa được dự đoán 204b (bước Sv_15). Ví dụ, tham số lượng tử hóa chênh lệch đối với khối hiện tại thu được từ bộ giải mã entropy 202 và tham số lượng tử hóa được dự đoán đối với khối hiện tại được tạo ra bởi bộ tạo tham số lượng tử hóa được dự đoán 204b có thể được cộng với nhau để tạo ra tham số lượng tử hóa đối với khối hiện tại. Ngoài ra, bộ tạo tham số lượng tử hóa 204a lưu trữ tham số lượng tử hóa đối với khối hiện tại trong bộ lưu trữ tham số lượng tử hóa 204d (bước Sv_16).

Tiếp theo, bộ thực thi lượng tử hóa ngược 204e lượng tử hóa ngược các hệ số được lượng tử hóa của khối hiện tại thành các hệ số biến đổi, sử dụng tham số lượng tử hóa được tạo ra trong bước Sv_15 (bước Sv_17).

Cần lưu ý rằng tham số lượng tử hóa chênh lệch có thể được giải mã tại mức chuỗi bit, mức ảnh, mức lát, mức viên, hoặc mức CTU. Ngoài ra, giá trị khởi tạo của tham số lượng tử hóa có thể được giải mã tại mức chuỗi, mức ảnh, mức lát, mức viên, hoặc mức CTU. Lúc này, tham số lượng tử hóa có thể được tạo ra nhờ sử dụng giá trị khởi tạo của tham số lượng tử hóa và tham số lượng tử hóa chênh lệch.

Cần lưu ý rằng bộ lượng tử hóa ngược 204 có thể bao gồm các bộ lượng tử hóa ngược, và có thể lượng tử hóa ngược các hệ số được lượng tử hóa sử dụng phương pháp lượng tử hóa ngược được lựa chọn từ các phương pháp lượng tử hóa ngược.

(Bộ biến đổi ngược)

Bộ biến đổi ngược 206 khôi phục các phần dư dự đoán bằng cách biến đổi ngược các hệ số biến đổi mà là các đầu vào từ bộ lượng tử hóa ngược 204.

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng EMT hoặc AMT cần được áp dụng (ví dụ, khi cờ AMT là đúng), bộ biến đổi ngược 206 biến đổi ngược

các hệ số biến đổi của khối hiện tại dựa trên thông tin mà chỉ báo loại biến đổi được phân tích.

Ngoài ra, ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng NSST cần được áp dụng, bộ biến đổi ngược 206 áp dụng biến đổi ngược thứ cấp tới các hệ số biến đổi.

FIG.75 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi ngược 206.

Ví dụ, bộ biến đổi ngược 206 xác định rằng thông tin mà chỉ báo rằng không có biến đổi trực giao được thực hiện được hiện diện trong dòng (bước St_11). Ở đây, khi xác định rằng không có thông tin này được hiện diện (Không trong bước St_11) (ví dụ: không có mặt bất kỳ chỉ báo đối với việc biến đổi trực giao có được thực hiện hay không; có chỉ báo mà chỉ báo rằng biến đổi trực giao cần được thực hiện); bộ biến đổi ngược 206 thu nhận thông tin mà chỉ báo loại biến đổi được giải mã bởi bộ giải mã entropy 202 (bước St_12). Tiếp theo, dựa trên thông tin này, bộ biến đổi ngược 206 xác định loại biến đổi được sử dụng cho việc biến đổi trực giao trong bộ mã hóa 100 (bước St_13). Sau đó bộ biến đổi ngược 206 thực hiện biến đổi trực giao ngược sử dụng loại biến đổi được xác định (bước St_14). Như được minh họa trong FIG.75, khi xác định rằng thông tin mà chỉ báo rằng không có biến đổi trực giao được thực hiện được hiện diện (Có trong bước St_11) (ví dụ, chỉ báo biểu diễn rằng không có biến đổi trực giao được thực hiện; không có mặt chỉ báo biến đổi trực giao được thực hiện), không có biến đổi trực giao được thực hiện.

FIG.76 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ biến đổi ngược 206.

Ví dụ, bộ biến đổi ngược 206 xác định rằng kích cỡ biến đổi có nhỏ hơn hoặc bằng giá trị định trước hay không (bước Su_11). Giá trị được xác định có thể được xác định trước. Ở đây, khi xác định rằng kích cỡ biến đổi nhỏ hơn hoặc bằng giá trị được xác định (Có trong bước Su_11), bộ biến đổi ngược 206 thu nhận, từ bộ giải mã entropy 202, thông tin mà chỉ báo loại biến đổi nào được sử dụng bởi bộ mã hóa 100 trong số ít nhất một loại biến đổi được chứa trong nhóm loại biến đổi thứ nhất

(bước Su_12). Cần lưu ý rằng thông tin này được giải mã bởi bộ giải mã entropy 202 và được xuất ra tới bộ biến đổi ngược 206.

Dựa trên thông tin này, bộ biến đổi ngược 206 xác định loại biến đổi được sử dụng cho biến đổi trực giao trong bộ mã hóa 100 (bước Su_13). Sau đó, bộ biến đổi ngược 206 biến đổi trực giao ngược các hệ số biến đổi của khối hiện tại nhờ sử dụng loại biến đổi được xác định (bước Su_14). Khi xác định rằng kích cỡ biến đổi ngược không nhỏ hơn hoặc bằng giá trị được xác định (Không trong bước Su_11), bộ biến đổi ngược 206 biến đổi ngược các hệ số biến đổi của khối hiện tại nhờ sử dụng nhóm loại biến đổi thứ hai (bước Su_15).

Cần lưu ý rằng biến đổi trực giao ngược bởi bộ biến đổi ngược 206 có thể được thực hiện theo dòng xử lý được minh họa trong FIG.75 hoặc FIG.76 đối với mỗi TU như là một ví dụ. Ngoài ra, biến đổi trực giao ngược có thể được thực hiện bằng cách sử dụng loại biến đổi được xác định mà không giải mã thông tin mà chỉ báo loại biến đổi được sử dụng cho biến đổi trực giao. Loại biến đổi được xác định có thể là loại biến đổi được xác định trước hoặc loại biến đổi mặc định. Ngoài ra, loại biến đổi có thể một cách cụ thể là DST7, DCT8, hoặc loại tương tự. Trong biến đổi trực giao ngược, hàm cơ sở biến đổi ngược tương ứng với loại biến đổi được sử dụng.

(Bộ cộng)

Bộ cộng 208 khôi phục khối hiện tại bằng cách cộng phần dư dự đoán mà là đầu vào từ bộ biến đổi ngược 206 và ảnh dự đoán mà là đầu vào từ bộ điều khiển dự đoán 220. Nói cách khác, ảnh được khôi phục của khối hiện tại được tạo ra. Bộ cộng 208 sau đó xuất ảnh được khôi phục của khối hiện tại tới bộ nhớ khối 210 và bộ lọc vòng 212.

(Bộ nhớ khối)

Bộ nhớ khối 210 là bộ lưu trữ để là trữ khối mà được chứa trong ảnh hiện tại và có thể được viện dẫn tới trong việc nội dự đoán. Cụ thể hơn, bộ nhớ khối 210 lưu trữ ảnh được khôi phục được xuất ra từ bộ cộng 208.

(Bộ lọc vòng)

Bộ lọc vòng 212 áp dụng lọc vòng tới ảnh được khôi phục được tạo ra bởi bộ cộng 208, và xuất ra ảnh được khôi phục được lọc tới bộ nhớ khung 214 và đưa ra đầu ra của bộ giải mã 200, ví dụ, và được xuất ra tới thiết bị hiển thị, v.v.

Khi thông tin mà chỉ báo việc kích hoạt hoặc ngắt hoạt động ALF được phân tích từ dòng chỉ báo rằng ALF được kích hoạt, một bộ lọc từ trong số nhiều bộ lọc có thể được lựa chọn, ví dụ, dựa trên chiều và độ hoạt động của gradien cục bộ, và bộ lọc được lựa chọn được áp dụng tới ảnh được khôi phục.

FIG.77 là sơ đồ khối minh họa một ví dụ về cấu trúc chức năng của bộ lọc vòng 212. Cần lưu ý rằng bộ lọc vòng 212 có cấu trúc tương tự cấu trúc của bộ lọc vòng 120 của bộ mã hóa 100.

Ví dụ, như được minh họa trong FIG.77, bộ lọc vòng 212 bao gồm bộ thực thi lọc giải khối 212a, bộ thực thi SAO 212b, và bộ thực thi ALF 212c. Bộ thực thi lọc giải khối 212a thực hiện xử lý lọc giải khối trên ảnh được khôi phục. Bộ thực thi SAO 212b thực hiện xử lý SAO trên ảnh được khôi phục sau khi được đưa vào xử lý lọc giải khối. Bộ thực thi ALF 212c thực hiện xử lý ALF trên ảnh được khôi phục sau khi được đưa vào xử lý SAO. Cần lưu ý rằng bộ lọc vòng 212 không cần luôn bao gồm tất cả các thành phần cấu thành được bộc lộ trong FIG.77, và có thể bao gồm chỉ một phần của các thành phần cấu thành. Ngoài ra, bộ lọc vòng 212 có thể có cấu trúc để thực hiện các xử lý nêu trên trong thứ tự xử lý khác với từ thứ tự được bộc lộ trong FIG.77, có thể không thực hiện tất cả các xử lý được minh họa trong FIG.77, v.v.

(Bộ nhớ khung)

Bộ nhớ khung 214, ví dụ, là bộ lưu trữ để lưu trữ các ảnh tham chiếu được sử dụng trong dự đoán liên đới, và có thể cũng được gọi là bộ đệm khung. Cụ thể hơn, bộ nhớ khung 214 lưu trữ ảnh được khôi phục được lọc bởi bộ lọc vòng 212.

(Bộ dự đoán (bộ nội dự đoán, bộ dự đoán liên đới, bộ điều khiển dự đoán))

FIG.78 là lưu đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã 200. Cần lưu ý rằng bộ thực thi dự đoán có thể bao gồm tất cả hoặc một phần của các thành phần cấu thành sau đây: bộ nội dự đoán 216; bộ dự đoán liên

đới 218; và bộ điều khiển dự đoán 220. Bộ thực thi dự đoán bao gồm, ví dụ, bộ nội dự đoán 216 và bộ dự đoán liên đới 218.

Bộ dự đoán tạo ra ảnh dự đoán của khối hiện tại (bước Sq_1). Ảnh dự đoán này cũng được gọi là tín hiệu dự đoán hoặc khối dự đoán. Cần lưu ý rằng tín hiệu dự đoán là, ví dụ, tín hiệu nội dự đoán hoặc tín hiệu dự đoán liên đới. Cụ thể hơn, bộ dự đoán tạo ra ảnh dự đoán của khối hiện tại nhờ sử dụng ảnh được khôi phục mà đã thu được đối với khối khác thông qua việc tạo ra ảnh dự đoán, việc khôi phục phần dư dự đoán, và việc cộng ảnh dự đoán. Bộ dự đoán của bộ giải mã 200 tạo ra ảnh dự đoán tương tự như ảnh dự đoán được tạo ra bởi bộ dự đoán của bộ mã hóa 100. Nói cách khác, các ảnh dự đoán được tạo ra theo phương pháp chung giữa các bộ dự đoán hoặc các phương pháp tương ứng.

Ảnh được khôi phục có thể là, ví dụ, ảnh trong ảnh tham chiếu, hoặc ảnh của khối được giải mã (tức là, khối khác được mô tả nêu trên) trong ảnh hiện tại mà là ảnh bao gồm khối hiện tại. Khối được giải mã trong ảnh hiện tại là, ví dụ, khối lân cận của khối hiện tại.

FIG.79 là lưu đồ minh họa ví dụ xử lý khác được thực hiện bởi bộ dự đoán của bộ giải mã 200.

Bộ dự đoán xác định phương pháp hoặc chế độ tạo ra ảnh dự đoán (bước Sr_1). Ví dụ, phương pháp hoặc chế độ có thể được xác định dựa trên, ví dụ, tham số dự đoán, v.v.

Khi xác định phương pháp thứ nhất là chế độ tạo ra ảnh dự đoán, bộ dự đoán tạo ra ảnh dự đoán theo phương pháp thứ nhất (bước Sr_2a). Khi xác định phương pháp thứ hai là chế độ tạo ra ảnh dự đoán, bộ dự đoán tạo ra ảnh dự đoán theo phương pháp thứ hai (bước Sr_2b). Khi xác định phương pháp thứ ba là chế độ tạo ra ảnh dự đoán, bộ dự đoán tạo ra ảnh dự đoán theo phương pháp thứ ba (bước Sr_2c).

Phương pháp thứ nhất, phương pháp thứ hai, và phương pháp thứ ba có thể là các phương pháp khác nhau để tạo ra ảnh dự đoán. Mỗi phương pháp thứ nhất đến thứ ba có thể là phương pháp dự đoán liên đới, phương pháp nội dự đoán, hoặc phương pháp dự đoán khác. Ảnh được khôi phục nêu trên có thể được sử dụng trong các phương pháp dự đoán này.

Các Fig.80A đến Fig.80C (gọi chung là FIG.80) là lưu đồ minh họa ví dụ khác về xử lý được thực hiện bởi bộ dự đoán của bộ giải mã 200.

Bộ dự đoán có thể thực hiện xử lý dự đoán theo dòng xử lý được minh họa trong FIG.80 như là một ví dụ. Cần lưu ý rằng việc sao chép nội khối được minh họa trong FIG.80 là một chế độ mà thuộc về việc dự đoán liên đới, và trong đó khối được chứa trong ảnh hiện tại được gọi là ảnh tham chiếu hoặc khối tham chiếu. Nói cách khác, ảnh khác với ảnh hiện tại không được viện dẫn trong sao chép nội khối. Ngoài ra, chế độ PCM được minh họa trong FIG.80 là một chế độ mà thuộc về việc nội dự đoán, và trong đó không có việc biến đổi và lượng tử hóa được thực hiện.

(Bộ nội dự đoán)

Bộ nội dự đoán 216 thực hiện việc nội dự đoán bằng cách viện dẫn tới khối trong ảnh hiện tại được lưu trữ trong bộ nhớ khối 210, dựa trên chế độ nội dự đoán được phân tích từ dòng, để tạo ra ảnh dự đoán của khối hiện tại (tức là, khối nội dự đoán). Cụ thể hơn, bộ nội dự đoán 216 thực hiện việc nội dự đoán bằng cách viện dẫn tới các giá trị điểm ảnh (ví dụ, các giá trị độ chói và/hoặc sắc độ) của khối hoặc các khối mà lân cận khối hiện tại để tạo ra ảnh nội dự đoán, và sau đó xuất ra ảnh nội dự đoán tới bộ điều khiển dự đoán 220.

Cần lưu ý rằng khi chế độ nội dự đoán trong đó khối độ chói được viện dẫn tới trong việc nội dự đoán của khối sắc độ được lựa chọn, bộ nội dự đoán 216 có thể dự đoán thành phần sắc độ của khối hiện tại dựa trên thành phần độ chói của khối hiện tại.

Ngoài ra, khi thông tin được phân tích từ dòng chỉ báo rằng PDPC cần được áp dụng, bộ nội dự đoán 216 hiệu chỉnh các giá trị điểm ảnh được nội dự đoán dựa trên các gradien điểm ảnh tham chiếu theo chiều dọc/ngang.

FIG.81 là sơ đồ minh họa một ví dụ về xử lý được thực hiện bởi bộ nội dự đoán 216 của bộ giải mã 200.

Đầu tiên, bộ nội dự đoán 216 xác định rằng MPM có cần được sử dụng hay không. Như được minh họa trong FIG.81, bộ nội dự đoán 216 xác định rằng cờ

MPM mà chỉ báo 1 có được hiện diện trong dòng hay không (bước Sw_11). Ở đây, khi xác định rằng có MPM mà chỉ báo 1 được hiện diện (Có trong bước Sw_11), bộ nội dự đoán 216 thu nhận, từ bộ giải mã entropy 202, thông tin mà chỉ báo chế độ nội dự đoán được lựa chọn trong bộ mã hóa 100 trong số các MPM. Cần lưu ý rằng thông tin này được giải mã bởi bộ giải mã entropy 202 và được xuất ra tới bộ nội dự đoán 216. Tiếp theo, bộ nội dự đoán 216 xác định các MPM (bước Sw_13). Các MPM bao gồm, ví dụ, sáu chế độ nội dự đoán. Sau đó, bộ nội dự đoán 216 xác định chế độ nội dự đoán mà được chứa trong các chế độ nội dự đoán được chứa trong các MPM và được chỉ báo bởi thông tin thu được trong bước Sw_12 (bước Sw_14).

Khi xác định rằng không có cờ MPM mà chỉ báo 1 được hiện diện (Không trong bước Sw_11), bộ nội dự đoán 216 thu nhận thông tin mà chỉ báo chế độ nội dự đoán được lựa chọn trong bộ mã hóa 100 (bước Sw_15). Nói cách khác, bộ nội dự đoán 216 thu nhận, từ bộ giải mã entropy 202, thông tin mà chỉ báo chế độ nội dự đoán được lựa chọn trong bộ mã hóa 100 từ trong số ít nhất một chế độ nội dự đoán mà không được chứa trong các MPM. Cần lưu ý rằng thông tin này được giải mã bởi bộ giải mã entropy 202 và được xuất ra tới bộ nội dự đoán 216. Sau đó, bộ nội dự đoán 216 xác định chế độ nội dự đoán mà không được chứa trong các chế độ nội dự đoán được chứa trong các MPM và được chỉ báo bởi thông tin thu được trong bước Sw_15 (bước Sw_17).

Bộ nội dự đoán 216 tạo ra ảnh dự đoán theo chế độ nội dự đoán được xác định trong bước Sw_14 hoặc bước Sw_17 (bước Sw_18).

(Bộ dự đoán liên đới)

Bộ dự đoán liên đới 218 dự đoán khối hiện tại bằng cách viện dẫn tới ảnh tham chiếu được lưu trữ trong bộ nhớ khung 214. Việc dự đoán được thực hiện trong các đơn vị của khối hiện tại hoặc khối con hiện tại trong khối hiện tại. Cần lưu ý rằng khối con được chứa trong khối và là đơn vị nhỏ hơn khối. Kích cỡ của khối con có thể là 4×4 điểm ảnh, 8×8 điểm ảnh, hoặc kích cỡ khác. Kích cỡ của khối con có thể được chuyển đổi đối với đơn vị như lát, viên, ảnh, v.v.

Ví dụ, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán liên đới của khối hiện tại hoặc khối con hiện tại bằng cách thực hiện việc bù chuyển động nhờ sử dụng thông

tin chuyển động (ví dụ, MV) được phân tích từ dòng (ví dụ, tham số dự đoán được xuất ra từ bộ giải mã entropy 202), và xuất ra ảnh dự đoán liên đới tới bộ điều khiển dự đoán 220.

Khi thông tin được phân tích từ dòng chỉ báo rằng chế độ OBMC cần được áp dụng, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán liên đới sử dụng thông tin chuyển động của khối lân cận ngoài thông tin chuyển động của khối hiện tại thu được thông qua ước lượng chuyển động.

Ngoài ra, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ FRUC cần được áp dụng, bộ dự đoán liên đới 218 thu nhận thông tin chuyển động bằng cách thực hiện ước lượng chuyển động theo phương pháp so khớp mẫu (ví dụ, so khớp hai chiều hoặc so khớp khuôn mẫu) được phân tích từ dòng. Bộ dự đoán liên đới 218 sau đó thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng thông tin chuyển động được thu nhận.

Ngoài ra, khi chế độ BIO cần được áp dụng, bộ dự đoán liên đới 218 thu được MV dựa trên mô hình mà giả định chuyển động tuyến tính đồng đều. Ngoài ra, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ afin cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV đối với mỗi khối con, dựa trên các MV của các khối lân cận.

(Dòng xử lý thu nhận MV)

FIG.82 là lưu đồ minh họa một ví dụ về xử lý thu nhận MV trong bộ giải mã 200.

Bộ dự đoán liên đới 218 xác định, ví dụ, rằng có giải mã thông tin chuyển động (ví dụ, MV) hay không. Ví dụ, bộ dự đoán liên đới 218 có thể thực hiện việc xác định theo chế độ dự đoán được chứa trong dòng, hoặc có thể thực hiện việc xác định dựa trên thông tin khác được chứa trong dòng. Ở đây, khi xác định để giải mã thông tin chuyển động, bộ dự đoán liên đới 218 thu nhận MV đối với khối hiện tại trong chế độ trong đó thông tin chuyển động được giải mã. Khi xác định không giải mã thông tin chuyển động, bộ dự đoán liên đới 218 thu nhận MV trong chế độ trong đó không có thông tin chuyển động được giải mã.

Ở đây, các chế độ thu nhận MV bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ FRUC, chế độ afin, v.v. mà được mô tả sau đây. Các chế độ trong đó thông tin chuyển động được giải mã trong số các chế độ bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ afin (cụ thể, chế độ liên đới afin và chế độ hợp nhất afin), v.v. Cần lưu ý rằng thông tin chuyển động có thể bao gồm không chỉ MV mà còn thông tin lựa chọn bộ dự đoán MV mà được mô tả sau đây. Các chế độ trong đó không có thông tin chuyển động được giải mã bao gồm chế độ FRUC, v.v. bộ dự đoán liên đới 218 lựa chọn chế độ để thu nhận MV đối với khối hiện tại từ các chế độ, và thu nhận MV đối với khối hiện tại nhờ sử dụng chế độ được lựa chọn.

FIG.83 là lưu đồ minh họa một ví dụ về xử lý thu nhận MV trong bộ giải mã 200.

Ví dụ, bộ dự đoán liên đới 218 có thể xác định rằng có giải mã hay không độ chênh lệch MV, tức là ví dụ, có thể thực hiện việc xác định theo chế độ dự đoán được chứa trong dòng, hoặc có thể thực hiện việc xác định dựa trên thông tin khác được chứa trong dòng. Ở đây, khi xác định giải mã độ chênh lệch MV, bộ dự đoán liên đới 218 có thể thu nhận MV đối với khối hiện tại trong chế độ trong đó độ chênh lệch MV được giải mã. Trong trường hợp này, ví dụ, độ chênh lệch MV được chứa trong dòng được giải mã như là tham số dự đoán.

Khi xác định không giải mã bất kỳ độ chênh lệch MV, bộ dự đoán liên đới 218 thu nhận MV trong chế độ trong đó không có độ chênh lệch MV được giải mã. Trong trường hợp này, không có độ chênh lệch MV được mã hóa được chứa trong dòng.

Ở đây, như được mô tả nêu trên, các chế độ thu nhận MV bao gồm chế độ liên đới thường, chế độ hợp nhất thường, chế độ FRUC, chế độ afin, v.v. mà được mô tả sau đây. Các chế độ trong đó độ chênh lệch MV được mã hóa trong số các chế độ bao gồm chế độ liên đới thường, và chế độ afin (cụ thể, chế độ liên đới afin), v.v. Các chế độ trong đó không có độ chênh lệch MV được mã hóa bao gồm chế độ FRUC, chế độ hợp nhất thường, chế độ afin (cụ thể, chế độ hợp nhất afin), v.v. bộ

dự đoán liên đới 218 lựa chọn chế độ để thu nhận MV đối với khối hiện tại từ các chế độ, và thu nhận MV đối với khối hiện tại nhờ sử dụng chế độ được lựa chọn.

(Thu nhận MV > chế độ liên đới thường)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ liên đới thường cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV dựa trên thông tin được phân tích từ dòng và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV.

FIG.84 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ liên đới thường trong bộ giải mã 200.

Bộ dự đoán liên đới 218 của bộ giải mã 200 thực hiện việc bù chuyển động đối với mỗi khối. Đầu tiên, bộ dự đoán liên đới 218 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được giải mã theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sg_11). Nói cách khác, bộ dự đoán liên đới 218 tạo ra danh sách ứng viên MV.

Tiếp theo, bộ dự đoán liên đới 218 tách N (số nguyên bằng 2 hoặc lớn hơn) ứng viên MV từ các ứng viên MV thu được trong bước Sg_11, như là các ứng viên bộ dự đoán vector chuyển động (cũng được gọi là các ứng viên bộ dự đoán MV) theo các hạng được xác định trong thứ tự ưu tiên (bước Sg_12). Cần lưu ý rằng các hạng trong thứ tự ưu tiên có thể được xác định trước đối với N ứng viên bộ dự đoán MV tương ứng và có thể được xác định trước.

Tiếp theo, bộ dự đoán liên đới 218 giải mã thông tin lựa chọn bộ dự đoán MV từ dòng đầu vào, và lựa chọn một ứng viên bộ dự đoán MV từ N ứng viên bộ dự đoán MV như là bộ dự đoán MV đối với khối hiện tại sử dụng thông tin lựa chọn bộ dự đoán MV được giải mã (bước Sg_13).

Tiếp theo, bộ dự đoán liên đới 218 giải mã độ chênh lệch MV từ dòng đầu vào, và thu nhận MV đối với khối hiện tại bằng cách cộng giá trị chênh lệch mà là độ chênh lệch MV được giải mã và bộ dự đoán MV được lựa chọn (bước Sg_14).

Cuối cùng, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được

thu nhận và ảnh tham chiếu được giải mã (bước Sg_15). Các xử lý trong các bước Sg_11 đến Sg_15 được thực thi trên mỗi khối. Ví dụ, khi các xử lý trong các bước Sg_11 đến Sg_15 được thực thi trên mỗi tất cả khối trong lát, việc dự đoán liên đới của lát mà sử dụng chế độ liên đới thường hoàn thành. Ví dụ, khi các xử lý trong các bước Sg_11 đến Sg_15 được thực thi trên mỗi tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ liên đới thường hoàn thành. Cần lưu ý rằng không phải tất cả các khối được chứa trong lát có thể được đưa vào xử lý trong các Sg_11 đến Sg_15, và việc dự đoán liên đới của lát sử dụng chế độ liên đới thường có thể hoàn thành khi một phần của các khối được đưa vào các xử lý. Điều này cũng áp dụng tới các ảnh trong các bước Sg_11 đến Sg_15. Việc dự đoán liên đới của ảnh sử dụng chế độ liên đới thường có thể hoàn thành khi các xử lý được thực thi trên một phần của các khối trong ảnh.

(Thu nhận MV > chế độ hợp nhất thường)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ hợp nhất thường cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV.

FIG.85 là lưu đồ minh họa ví dụ về xử lý của việc dự đoán liên đới bởi chế độ hợp nhất thường trong bộ giải mã 200.

Đầu tiên, bộ dự đoán liên đới 218 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được giải mã theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sh_11). Nói cách khác, bộ dự đoán liên đới 218 tạo ra danh sách ứng viên MV.

Tiếp theo, bộ dự đoán liên đới 218 lựa chọn một ứng viên MV từ các ứng viên MV thu được trong bước Sh_11, thu nhận MV đối với khối hiện tại (bước Sh_12). Cụ thể hơn, bộ dự đoán liên đới 218 thu nhận thông tin lựa chọn MV được bao gồm như là tham số dự đoán trong dòng, và lựa chọn ứng viên MV được nhận dạng bởi thông tin lựa chọn MV như là MV đối với khối hiện tại.

Cuối cùng, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được thu nhận và ảnh tham chiếu được giải mã (bước Sh_13). Các xử lý trong các bước

Sh_11 đến Sh_13 được thực thi, ví dụ, trên mỗi khối. Ví dụ, khi các xử lý trong các bước Sh_11 đến Sh_13 được thực thi trên mỗi tất cả khối trong lát, việc dự đoán liên đới của lát sử dụng chế độ hợp nhất thường hoàn thành. Ngoài ra, khi các xử lý trong các bước Sh_11 đến Sh_13 được thực thi trên mỗi tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ hợp nhất thường hoàn thành. Cần lưu ý rằng không phải tất cả các khối được chứa trong lát được đưa vào xử lý trong các Sh_11 đến Sh_13, và việc dự đoán liên đới của lát sử dụng chế độ hợp nhất thường có thể hoàn thành khi một phần của các khối được đưa vào các xử lý. Điều này cũng áp dụng tới các ảnh trong các bước Sh_11 đến Sh_13. Việc dự đoán liên đới của ảnh sử dụng chế độ hợp nhất thường có thể hoàn thành khi các xử lý được thực thi trên một phần của các khối trong ảnh.

(Thu nhận MV > chế độ FRUC)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ FRUC cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV trong chế độ FRUC và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV. Trong trường hợp này, thông tin chuyển động được thu nhận tại phía bộ giải mã 200 mà không được báo hiệu từ phía bộ mã hóa 100. Ví dụ, bộ giải mã 200 có thể thu nhận thông tin chuyển động bằng cách thực hiện ước lượng chuyển động. Trong trường hợp này, bộ giải mã 200 thực hiện ước lượng chuyển động mà không sử dụng bất kỳ các giá trị điểm ảnh trong khối hiện tại.

FIG.86 là lưu đồ minh họa ví dụ về xử lý dự đoán liên đới bởi chế độ FRUC trong bộ giải mã 200.

Đầu tiên, bộ dự đoán liên đới 218 tạo ra danh sách mà chỉ báo các MV của các khối được giải mã theo không gian hoặc thời gian lân cận khối hiện tại bằng cách viển dẫn tới các MV như là các ứng viên MV (danh sách này là danh sách ứng viên MV, và có thể, ví dụ, cũng được sử dụng như là danh sách ứng viên MV đối với chế độ hợp nhất thường (bước Si_11). Tiếp theo, ứng viên MV tốt nhất được lựa chọn từ các ứng viên MV được ghi trong danh sách ứng viên MV (bước Si_12). Ví dụ, bộ dự đoán liên đới 218 tính toán giá trị đánh giá của mỗi ứng viên MV được chứa trong danh sách ứng viên MV, và lựa chọn một trong số các ứng viên MV như là ứng viên

MV tốt nhất dựa trên các giá trị đánh giá. Dựa trên các ứng viên MV tốt nhất được lựa chọn, sau đó bộ dự đoán liên đới 218 thu nhận MV đối với khối hiện tại (bước Si_14). Cụ thể hơn, ví dụ, các ứng viên MV tốt nhất được lựa chọn được thu nhận trực tiếp như là MV đối với khối hiện tại. Ngoài ra, ví dụ, MV đối với khối hiện tại có thể được thu nhận nhờ sử dụng so khớp mẫu trong vùng xung quanh của vị trí mà được chứa trong ảnh tham chiếu và tương ứng với ứng viên MV tốt nhất được lựa chọn. Nói cách khác, việc ước lượng sử dụng so khớp khuôn mẫu trong ảnh tham chiếu và các giá trị đánh giá có thể được thực hiện trong vùng xung quanh của ứng viên MV tốt nhất, và khi có MV mà có giá trị đánh giá tốt hơn, ứng viên MV tốt nhất có thể được cập nhật thành MV mà có giá trị đánh giá tốt hơn, và MV được cập nhật có thể được xác định như là MV cuối cùng đối với khối hiện tại. Trong phương án của sáng chế, việc cập nhật tới MV mà có giá trị đánh giá tốt hơn có thể không được thực hiện.

Cuối cùng, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc bù chuyển động của khối hiện tại nhờ sử dụng MV được thu nhận và ảnh tham chiếu được giải mã (bước Si_15). Các xử lý trong các bước Si_11 đến Si_15 được thực thi, ví dụ, trên mỗi khối. Ví dụ, khi các xử lý trong các bước Si_11 đến Si_15 được thực thi trên mỗi tất cả các khối trong lát, việc dự đoán liên đới của lát sử dụng chế độ FRUC hoàn thành. Ví dụ, khi các xử lý trong các bước Si_11 đến Si_15 được thực thi trên mỗi tất cả các khối trong ảnh, việc dự đoán liên đới của ảnh sử dụng chế độ FRUC hoàn thành. Mỗi khối con có thể được xử lý tương tự như trường hợp của mỗi khối.

(Thu nhận $MV > \text{chế độ FRUC}$)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ hợp nhất affin cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV trong chế độ hợp nhất affin và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV.

FIG.87 là lưu đồ minh họa ví dụ về xử lý của việc dự đoán liên đới bởi chế độ hợp nhất affin trong bộ giải mã 200.

Trong chế độ hợp nhất affin, đầu tiên, bộ dự đoán liên đới 218 thu nhận các MV tại các điểm điều khiển tương ứng đối với khối hiện tại (bước Sk_11). Các điểm

điều khiển là điểm góc phía trên-bên trái của khối hiện tại và điểm góc phía trên-bên phải của khối hiện tại như được minh họa trong FIG.46A, hoặc điểm góc phía trên-bên trái của khối hiện tại, điểm góc phía trên-bên phải của khối hiện tại, và điểm góc phía dưới-bên trái của khối hiện tại như được minh họa trong FIG.46B.

Ví dụ, khi các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C được sử dụng, như được minh họa trong FIG.47A, bộ dự đoán liên đới 218 kiểm tra khối A (bên trái), khối B (phía trên), khối C (phía trên-bên phải), khối D (phía dưới-bên trái), và khối E (phía trên-bên trái) được giải mã trong thứ tự này, và nhận dạng khối có hiệu lực thứ nhất được giải mã theo chế độ affin. Bộ dự đoán liên đới 218 thu nhận MV tại điểm điều khiển sử dụng khối có hiệu lực thứ nhất được nhận dạng được giải mã theo chế độ affin. Ví dụ, khi khối A được nhận dạng và khối A có hai điểm điều khiển, như được minh họa trong FIG.47B, bộ dự đoán liên đới 218 tính toán vectơ chuyển động v_0 tại điểm điều khiển góc phía trên-bên trái của khối hiện tại và vectơ chuyển động v_1 tại điểm điều khiển góc phía trên-bên phải của khối hiện tại từ các vectơ chuyển động v_3 và v_4 tại góc phía trên-bên trái và góc phía trên-bên phải của khối được giải mã bao gồm khối A. Theo cách này, MV tại mỗi điểm điều khiển được thu nhận.

Cần lưu ý rằng, như được minh họa trong FIG.49A, các MV tại ba điểm điều khiển có thể được tính toán khi khối A được nhận dạng và khối A có hai điểm điều khiển, và như được minh họa trong FIG.49B, các MV tại hai điểm điều khiển có thể được tính toán khi khối A được nhận dạng và khi khối A có ba điểm điều khiển.

Ngoài ra, khi thông tin lựa chọn MV được bao gồm như là tham số dự đoán trong dòng, bộ dự đoán liên đới 218 có thể thu nhận MV tại mỗi điểm điều khiển đối với khối hiện tại nhờ sử dụng thông tin lựa chọn MV.

Tiếp theo, bộ dự đoán liên đới 218 thực hiện việc bù chuyển động của mỗi khối con được chứa trong khối hiện tại. Nói cách khác, bộ dự đoán liên đới 218 tính toán MV đối với khối con như là MV affin, sử dụng hai vectơ chuyển động v_0 và v_1 và biểu thức (1A) nêu trên hoặc ba vectơ chuyển động v_0 , v_1 , và v_2 và biểu thức (1B) nêu trên (bước Sk_12). Sau đó, bộ dự đoán liên đới 218 thực hiện việc bù chuyển động của các khối con sử dụng các MV affin này và các ảnh tham chiếu được

giải mã (bước Sk_13). Khi các xử lý trong các bước Sk_12 và Sk_13 được thực thi đối với mỗi khối con được chứa trong khối hiện tại, việc dự đoán liên đới nhờ sử dụng chế độ hợp nhất afin đối với khối hiện tại hoàn thành. Nói cách khác, việc bù chuyển động của khối hiện tại được thực hiện để tạo ra ảnh dự đoán của khối hiện tại.

Cần lưu ý rằng danh sách ứng viên MV nêu trên có thể được tạo ra trong bước Sk_11. Danh sách ứng viên MV có thể là, ví dụ, danh sách bao gồm các ứng viên MV thu được nhờ sử dụng các phương pháp thu nhận MV đối với mỗi điểm điều khiển. Các phương pháp thu nhận MV có thể là, ví dụ, bất kỳ kết hợp của các phương pháp thu nhận MV được minh họa trong các Fig.47A đến Fig.47C, các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B, các phương pháp thu nhận MV được minh họa trong các Fig.49A và Fig.49B, và các phương pháp thu nhận MV khác.

Cần lưu ý rằng danh sách ứng viên MV có thể bao gồm các ứng viên MV trong chế độ trong đó việc dự đoán được thực hiện trong các đơn vị của khối con, ngoài chế độ afin.

Cần lưu ý rằng, ví dụ, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra như là danh sách ứng viên MV. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và danh sách ứng viên MV bao gồm các ứng viên MV trong chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra một cách riêng biệt. Ngoài ra, danh sách ứng viên MV bao gồm các ứng viên MV trong một trong số chế độ hợp nhất afin trong đó hai điểm điều khiển được sử dụng và chế độ hợp nhất afin trong đó ba điểm điều khiển được sử dụng có thể được tạo ra.

(Thu nhận MV > chế độ liên đới afin)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ liên đới afin cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV trong chế độ liên đới afin và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV.

FIG.88 là lưu đồ minh họa ví dụ về xử lý của việc dự đoán liên đới bởi chế độ liên đới afin trong bộ giải mã 200.

Trong chế độ liên đới afin, đầu tiên, bộ dự đoán liên đới 218 thu nhận các bộ dự đoán MV (v_0 , v_1) hoặc (v_0 , v_1 , v_2) của hai hoặc ba điểm điều khiển tương ứng đối với khối hiện tại (bước Sj_11). Các điểm điều khiển là điểm góc phía trên-bên trái của khối hiện tại, điểm góc phía trên-bên phải của khối hiện tại, và điểm góc phía dưới-bên trái của khối hiện tại như được minh họa trong FIG.46A hoặc FIG.46B.

Bộ dự đoán liên đới 218 thu nhận thông tin lựa chọn bộ dự đoán MV được bao gồm như là tham số dự đoán trong dòng, và thu nhận bộ dự đoán MV tại mỗi điểm điều khiển đối với khối hiện tại nhờ sử dụng MV được nhận dạng bởi thông tin lựa chọn bộ dự đoán MV. Ví dụ, khi các phương pháp thu nhận MV được minh họa trong các Fig.48A và Fig.48B được sử dụng, bộ dự đoán liên đới 218 thu nhận các bộ dự đoán vectơ chuyển động (v_0 , v_1) hoặc (v_0 , v_1 , v_2) tại các điểm điều khiển đối với khối hiện tại bằng cách lựa chọn MV của khối được nhận dạng bởi thông tin lựa chọn bộ dự đoán MV trong số các khối được mã hóa trong lân cận của các điểm điều khiển tương ứng đối với khối hiện tại được minh họa trong FIG.48A hoặc FIG.48B.

Tiếp theo, bộ dự đoán liên đới 218 thu nhận mỗi độ chênh lệch MV được bao gồm như là tham số dự đoán trong dòng, và cộng bộ dự đoán MV tại mỗi điểm điều khiển đối với khối hiện tại và độ chênh lệch MV tương ứng với bộ dự đoán MV (bước Sj_12). Theo cách này, MV tại mỗi điểm điều khiển đối với khối hiện tại được thu nhận.

Tiếp theo, bộ dự đoán liên đới 218 thực hiện việc bù chuyển động của mỗi khối con được chứa trong khối hiện tại. Nói cách khác, bộ dự đoán liên đới 218 tính toán MV đối với khối con như là MV afin, sử dụng hai vectơ chuyển động v_0 và v_1 và biểu thức (1A) nêu trên hoặc ba vectơ chuyển động v_0 , v_1 , và v_2 và biểu thức (1B) nêu trên (bước Sj_13). Sau đó, bộ dự đoán liên đới 218 thực hiện việc bù chuyển động của các khối con sử dụng các MV afin này và các ảnh tham chiếu được giải mã (bước Sj_14). Khi các xử lý trong các bước Sj_13 và Sj_14 được thực thi đối với mỗi khối con được chứa trong khối hiện tại, việc dự đoán liên đới nhờ sử dụng chế độ hợp nhất afin đối với khối hiện tại hoàn thành. Nói cách khác, việc bù

chuyển động của khối hiện tại được thực hiện để tạo ra ảnh dự đoán của khối hiện tại.

Cần lưu ý rằng danh sách ứng viên MV nêu trên có thể được tạo ra trong bước Sj_11 như trong bước Sk_11.

(Thu nhận $MV > \text{chế độ tam giác}$)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng chế độ tam giác cần được áp dụng, bộ dự đoán liên đới 218 thu nhận MV trong chế độ tam giác và thực hiện việc bù chuyển động (dự đoán) nhờ sử dụng MV.

FIG.89 là lưu đồ minh họa ví dụ về xử lý của việc dự đoán liên đới bởi chế độ tam giác trong bộ giải mã 200.

Đầu tiên, trong chế độ tam giác, bộ dự đoán liên đới 218 phân chia khối hiện tại thành phân vùng thứ nhất và phân vùng thứ hai (bước Sx_11). Ví dụ, bộ dự đoán liên đới 218 có thể thu nhận, từ dòng, thông tin phân chia mà là thông tin liên quan đến việc phân chia như là tham số dự đoán. Sau đó, bộ dự đoán liên đới 218 có thể phân chia khối hiện tại thành phân vùng thứ nhất và phân vùng thứ hai theo thông tin phân vùng.

Tiếp theo, bộ dự đoán liên đới 218 thu nhận các ứng viên MV đối với khối hiện tại dựa trên thông tin như các MV của các khối được giải mã theo thời gian hoặc không gian xung quanh khối hiện tại (bước Sx_12). Nói cách khác, bộ dự đoán liên đới 218 tạo ra danh sách ứng viên MV.

Sau đó, bộ dự đoán liên đới 218 lựa chọn ứng viên MV đối với phân vùng thứ nhất và ứng viên MV đối với phân vùng thứ hai như là MV thứ nhất và MV thứ hai, một cách lần lượt, từ các ứng viên MV thu được trong bước Sx_11 (bước Sx_13). Lúc này, bộ dự đoán liên đới 218 có thể thu nhận từ dòng, thông tin lựa chọn MV để nhận dạng mỗi ứng viên MV được lựa chọn như là tham số dự đoán. Sau đó, bộ dự đoán liên đới 218 có thể lựa chọn MV thứ nhất và MV thứ hai theo thông tin lựa chọn MV.

Tiếp theo, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán thứ nhất bằng cách thực hiện việc bù chuyển động nhờ sử dụng MV thứ nhất được lựa chọn và ảnh tham

chiếu được giải mã (bước Sx_14). Tương tự, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán thứ hai bằng cách thực hiện việc bù chuyển động nhờ sử dụng MV thứ hai được lựa chọn và ảnh tham chiếu được giải mã (bước Sx_15).

Cuối cùng, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện việc cộng có trọng số của ảnh dự đoán thứ nhất và ảnh dự đoán thứ hai (bước Sx_16).

(Ước lượng $MV > DMVR$)

Ví dụ, thông tin được phân tích từ dòng chỉ báo rằng DMVR cần được áp dụng, bộ dự đoán liên đới 218 thực hiện ước lượng chuyển động nhờ sử dụng DMVR.

FIG.90 là lưu đồ minh họa ví dụ về xử lý ước lượng chuyển động bởi DMVR trong bộ giải mã 200.

Bộ dự đoán liên đới 218 thu nhận MV đối với khối hiện tại theo chế độ hợp nhất (bước Sl_11). Tiếp theo, bộ dự đoán liên đới 218 thu nhận MV cuối cùng đối với khối hiện tại bằng cách tìm kiếm vùng xung quanh ảnh tham chiếu được chỉ báo bởi MV được thu nhận trong Sl_11 (bước Sl_12). Nói cách khác, trong trường hợp này, MV của khối hiện tại được xác định theo DMVR.

FIG.91 là lưu đồ minh họa ví dụ về xử lý ước lượng chuyển động bởi DMVR trong bộ giải mã 200, và tương tự như FIG.58B.

Đầu tiên, trong bước 1 được minh họa trên Fig.58A, bộ dự đoán liên đới 218 tính toán giá trị giữa vị trí tìm kiếm (cũng được gọi là điểm bắt đầu) được chỉ báo bởi MV khởi tạo và tám vị trí tìm kiếm xung quanh. Sau đó, bộ dự đoán liên đới 218 xác định rằng giá trị tại mỗi vị trí tìm kiếm khác ngoài điểm bắt đầu có phải là nhỏ nhất hay không. Ở đây, khi xác định rằng giá trị tại một trong số các vị trí tìm kiếm khác ngoài điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 218 thay đổi đích tới vị trí tìm kiếm mà tại đó giá trị nhỏ nhất được thu nhận, và thực hiện xử lý trong bước 2 được minh họa trên FIG.58. Khi giá trị tại điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 218 bỏ qua xử lý trong bước 2 được minh họa trên FIG.58A và thực hiện xử lý trong bước 3.

Trong bước 2 được minh họa trong FIG.58A, bộ dự đoán liên đới 218 thực hiện việc tìm kiếm tương tự xử lý trong bước 1, mà xem xét vị trí tìm kiếm sau khi thay đổi đích như là điểm bắt đầu mới theo kết quả của xử lý trong bước 1. Sau đó, bộ dự đoán liên đới 218 xác định rằng giá trị tại mỗi vị trí tìm kiếm khác ngoài điểm bắt đầu có phải là nhỏ nhất hay không. Ở đây, khi xác định rằng giá trị tại một trong số các vị trí tìm kiếm khác ngoài điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 218 thực hiện xử lý trong bước 4. Khi giá trị tại điểm bắt đầu là nhỏ nhất, bộ dự đoán liên đới 218 thực hiện xử lý trong bước 3.

Trong bước 4, bộ dự đoán liên đới 218 xem xét vị trí tìm kiếm tại điểm bắt đầu như là vị trí tìm kiếm cuối cùng, và xác định độ chênh lệch giữa vị trí được chỉ báo bởi MV khởi tạo và vị trí tìm kiếm cuối cùng là độ chênh lệch vector.

Trong bước 3 được minh họa trên FIG.58A, bộ dự đoán liên đới 218 xác định vị trí điểm ảnh tại độ chính xác điểm ảnh con mà tại đó giá trị nhỏ nhất được thu nhận, dựa trên các giá trị tại bốn điểm nằm tại các vị trí phía trên, phía dưới, bên trái, và bên phải so với điểm bắt đầu trong bước 1 hoặc bước 2, và xem xét vị trí điểm ảnh như là vị trí tìm kiếm cuối cùng.

Vị trí điểm ảnh tại độ chính xác điểm ảnh con được xác định bằng cách thực hiện việc cộng có trọng số của mỗi bốn vector phía trên, phía dưới, bên trái, và bên phải $((0, 1), (0, -1), (-1, 0), \text{ và } (1, 0))$, sử dụng, như là trọng số, giá trị tại một trong số bốn vị trí tìm kiếm tương ứng. Sau đó, bộ dự đoán liên đới 218 xác định độ chênh lệch giữa vị trí được chỉ báo bởi MV khởi tạo và vị trí tìm kiếm cuối cùng là độ chênh lệch vector.

(Bù chuyển động > BIO/OBMC/LIC)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng việc hiệu chỉnh ảnh dự đoán cần được thực hiện, sau khi tạo ra ảnh dự đoán, bộ dự đoán liên đới 218 hiệu chỉnh ảnh dự đoán dựa trên chế độ đối với hiệu chỉnh. Chế độ này là, ví dụ, một trong số BIO, OBMC, và LIC được mô tả nêu trên.

FIG.92 là lưu đồ minh họa một ví dụ về xử lý tạo ra ảnh dự đoán trong bộ giải mã 200.

Bộ dự đoán liên đới 218 tạo ra ảnh dự đoán (bước Sm_11), và hiệu chỉnh ảnh dự đoán theo bất kỳ chế độ được mô tả nêu trên (bước Sm_12).

FIG.93 là lưu đồ minh họa ví dụ khác về xử lý tạo ra ảnh dự đoán trong bộ giải mã 200.

Bộ dự đoán liên đới 218 thu nhận MV đối với khối hiện tại (bước Sn_11). Tiếp theo, bộ dự đoán liên đới 218 tạo ra ảnh dự đoán nhờ sử dụng MV (bước Sn_12), và xác định rằng có thực hiện xử lý hiệu chỉnh hay không (bước Sn_13). Ví dụ, bộ dự đoán liên đới 218 thu nhận tham số dự đoán được chứa trong dòng, và xác định rằng có thực hiện xử lý hiệu chỉnh dựa trên tham số dự đoán hay không. Tham số dự đoán này là, ví dụ, cờ mà chỉ báo rằng một hoặc nhiều chế độ nêu trên có cần được áp dụng hay không. Ở đây, khi xác định thực hiện xử lý hiệu chỉnh (Có trong bước Sn_13), bộ dự đoán liên đới 218 tạo ra ảnh dự đoán cuối cùng bằng cách hiệu chỉnh ảnh dự đoán (bước Sn_14). Cần lưu ý rằng, trong LIC, độ chói và sắc độ có thể được hiệu chỉnh trong bước Sn_14. Khi xác định không thực hiện xử lý hiệu chỉnh (Không trong bước Sn_13), bộ dự đoán liên đới 218 xuất ra ảnh dự đoán cuối cùng mà không hiệu chỉnh ảnh dự đoán (bước Sn_15).

(Bù chuyển động > OBMC)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng OBMC cần được thực hiện, sau khi tạo ra ảnh dự đoán, bộ dự đoán liên đới 218 hiệu chỉnh ảnh dự đoán theo OBMC.

FIG.94 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi OBMC trong bộ giải mã 200. Cần lưu ý rằng lưu đồ trong FIG.94 chỉ báo dòng xử lý hiệu chỉnh ảnh dự đoán nhờ sử dụng ảnh hiện tại và ảnh tham chiếu được minh họa trong FIG.62.

Đầu tiên, như được minh họa trong FIG.62, bộ dự đoán liên đới 218 thu nhận ảnh dự đoán (Pred) bằng cách bù chuyển động thường nhờ sử dụng MV được gán tới khối hiện tại.

Tiếp theo, bộ dự đoán liên đới 218 thu nhận ảnh dự đoán (Pred_L) bằng cách áp dụng vector chuyển động (MV_L) mà đã được thu nhận đối với đối tượng được

giải mã lân cận bên trái của khối hiện tại tới khối hiện tại (tái sử dụng vector chuyển động đối với khối hiện tại). Bộ dự đoán liên đới 218 sau đó thực hiện việc hiệu chỉnh thứ nhất của ảnh dự đoán bằng cách xếp chồng hai ảnh dự đoán $Pred$ và $Pred_L$. Điều này mang lại hiệu quả trộn lẫn biên giữa các khối lân cận.

Tương tự, bộ dự đoán liên đới 218 thu nhận ảnh dự đoán ($Pred_U$) bằng cách áp dụng MV (MV_U) mà đã được thu nhận đối với khối được giải mã lân cận phía trên khối hiện tại tới khối hiện tại (tái sử dụng vector chuyển động đối với khối hiện tại). Sau đó, bộ dự đoán liên đới 218 thực hiện việc hiệu chỉnh thứ hai của ảnh dự đoán bằng cách xếp chồng ảnh dự đoán $Pred_U$ tới các ảnh dự đoán (ví dụ, $Pred$ và $Pred_L$) mà trên đó việc hiệu chỉnh thứ nhất đã được thực hiện. Điều này mang lại hiệu quả trộn lẫn biên giữa các khối lân cận. Ảnh dự đoán thu được bởi việc hiệu chỉnh thứ hai là ảnh trong đó biên giữa các khối lân cận đã được trộn (được san bằng), và do đó là ảnh dự đoán cuối cùng của khối hiện tại.

(Bù chuyển động > BIO)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng BIO cần được thực hiện, sau khi tạo ra ảnh dự đoán, bộ dự đoán liên đới 218 hiệu chỉnh ảnh dự đoán theo BIO.

FIG.95 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi BIO trong bộ giải mã 200.

Như được minh họa trong FIG.63, bộ dự đoán liên đới 218 thu nhận hai vector chuyển động (M_0 , M_1), nhờ sử dụng hai ảnh tham chiếu (Ref_0 , Ref_1) khác với ảnh ($Cur\ Pic$) bao gồm khối hiện tại. Sau đó, bộ dự đoán liên đới 218 thu nhận ảnh dự đoán đối với khối hiện tại nhờ sử dụng hai vector chuyển động (M_0 , M_1) (bước Sy_11). Cần lưu ý rằng vector chuyển động M_0 là vector chuyển động (MV_{x0} , MV_{y0}) tương ứng với ảnh tham chiếu Ref_0 , và vector chuyển động M_1 là vector chuyển động (MV_{x1} , MV_{y1}) tương ứng với ảnh tham chiếu Ref_1 .

Tiếp theo, bộ dự đoán liên đới 218 thu nhận ảnh được nội suy I^0 đối với khối hiện tại nhờ sử dụng vector chuyển động M_0 và ảnh tham chiếu L_0 . Ngoài ra, bộ dự đoán liên đới 218 thu nhận ảnh được nội suy I^1 đối với khối hiện tại nhờ sử dụng vector chuyển động M_1 và ảnh tham chiếu L_1 (bước Sy_12). Ở đây, ảnh được nội suy

I^0 là ảnh được chứa trong ảnh tham chiếu Ref_0 và được thu nhận đối với khối hiện tại, và ảnh được nội suy I^1 là ảnh được chứa trong ảnh tham chiếu Ref_1 và được thu nhận đối với khối hiện tại. Mỗi ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể có cùng kích cỡ như khối hiện tại. Ngoài ra, mỗi ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể là ảnh lớn hơn khối hiện tại. Ngoài ra, ảnh được nội suy I^0 và ảnh được nội suy I^1 có thể bao gồm ảnh dự đoán thu được bằng cách sử dụng các vector chuyển động (M_0, M_1) và các ảnh tham chiếu (L_0, L_1) và áp dụng bộ lọc bù chuyển động.

Ngoài ra, bộ dự đoán liên đới 218 thu nhận các ảnh gradient (Ix^0, Ix^1, Iy^0, Iy^1) của khối hiện tại, từ ảnh được nội suy I^0 và ảnh được nội suy I^1 (bước Sy_13). Cần lưu ý rằng các ảnh gradient trong chiều ngang là (Ix^0, Ix^1), và các ảnh gradient trong chiều dọc là (Iy^0, Iy^1). Bộ dự đoán liên đới 218 có thể thu nhận các ảnh gradient bằng cách, ví dụ, áp dụng lọc gradient tới các ảnh được nội suy. Các ảnh gradient có thể các ảnh mà mỗi chúng chỉ báo lượng thay đổi không gian trong giá trị điểm ảnh theo chiều ngang hoặc lượng thay đổi không gian trong giá trị điểm ảnh theo chiều dọc.

Tiếp theo, bộ dự đoán liên đới 218 thu nhận, đối với mỗi khối con của khối hiện tại, dòng quang (vx, vy) mà là vector vận tốc, sử dụng các ảnh được nội suy (I^0, I^1) và các ảnh gradient (Ix^0, Ix^1, Iy^0, Iy^1) (bước Sy_14). Theo một ví dụ, khối con có thể là CU con 4×4 điểm ảnh.

Tiếp theo, bộ dự đoán liên đới 218 hiệu chỉnh ảnh dự đoán đối với khối hiện tại nhờ sử dụng dòng quang (vx, vy). Ví dụ, bộ dự đoán liên đới 218 thu nhận giá trị hiệu chỉnh đối với giá trị của điểm ảnh được chứa trong khối hiện tại, nhờ sử dụng dòng quang (vx, vy) (bước Sy_15). Sau đó, bộ dự đoán liên đới 218 có thể hiệu chỉnh ảnh dự đoán đối với khối hiện tại nhờ sử dụng giá trị hiệu chỉnh (bước Sy_16). Cần lưu ý rằng giá trị hiệu chỉnh có thể được thu nhận trong các đơn vị của điểm ảnh, hoặc có thể được thu nhận trong các đơn vị của các điểm ảnh hoặc trong các đơn vị của khối con, v.v.

Cần lưu ý rằng dòng xử lý BIO không bị giới hạn ở xử lý được bộc lộ trong FIG.95. Chỉ một phần của các xử lý được bộc lộ trong FIG.95 có thể được thực hiện, hoặc xử lý khác có thể được bổ sung hoặc được sử dụng như là sự thay thế, hoặc các xử lý có thể được thực thi trong thứ tự xử lý khác.

(Bù chuyển động > LIC)

Ví dụ, khi thông tin được phân tích từ dòng chỉ báo rằng LIC cần được thực hiện, sau khi tạo ra ảnh dự đoán, bộ dự đoán liên đới 218 hiệu chỉnh ảnh dự đoán theo LIC.

FIG.96 là lưu đồ minh họa ví dụ về xử lý hiệu chỉnh ảnh dự đoán bởi LIC trong bộ giải mã 200.

Đầu tiên, bộ dự đoán liên đới 218 thu nhận ảnh tham chiếu tương ứng với khối hiện tại từ ảnh tham chiếu được giải mã nhờ sử dụng MV (bước Sz_11).

Tiếp theo, bộ dự đoán liên đới 218 tách, đối với khối hiện tại, thông tin mà chỉ báo giá trị độ chói đã thay đổi như thế nào giữa ảnh hiện tại và ảnh tham chiếu (bước Sz_12). Việc tách có thể được thực hiện dựa trên các giá trị điểm ảnh độ chói đối với vùng tham chiếu lân cận bên trái được giải mã (vùng tham chiếu xung quanh) và vùng tham chiếu lân cận phía trên được giải mã (vùng tham chiếu xung quanh), và các giá trị điểm ảnh độ chói tại các vị trí tương ứng trong ảnh tham chiếu được chỉ rõ bởi các MV được thu nhận. Bộ dự đoán liên đới 218 tính toán tham số hiệu chỉnh độ chói, nhờ sử dụng thông tin mà chỉ báo giá trị độ chói đã thay đổi như thế nào (bước Sz_13).

Bộ dự đoán liên đới 218 tạo ra ảnh dự đoán đối với khối hiện tại bằng cách thực hiện xử lý hiệu chỉnh độ chói trong đó tham số hiệu chỉnh độ chói được áp dụng tới ảnh tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV (bước Sz_14). Nói cách khác, ảnh dự đoán mà là ảnh tham chiếu trong ảnh tham chiếu được chỉ rõ bởi MV được đưa vào hiệu chỉnh dựa trên tham số hiệu chỉnh độ chói. Trong việc hiệu chỉnh này, độ chói có thể được hiệu chỉnh, hoặc sắc độ có thể được hiệu chỉnh.

(Bộ điều khiển dự đoán)

Bộ điều khiển dự đoán 220 lựa chọn ảnh nội dự đoán hoặc ảnh dự đoán liên đới, và xuất ra ảnh được lựa chọn tới bộ cộng 208. Nói chung, các cấu trúc, chức năng, và xử lý của bộ điều khiển dự đoán 220, bộ nội dự đoán 216, và bộ dự đoán liên đới 218 tại phía bộ giải mã 200 có thể tương ứng với các cấu trúc, chức năng, và

các xử lý của bộ điều khiển dự đoán 128, bộ nội dự đoán 124, và bộ dự đoán liên đới 126 tại phía bộ mã hóa 100.

(Khía cạnh thứ nhất)

FIG.97 là lưu đồ của ví dụ về dòng xử lý 1000 để giải mã ảnh mà sử dụng xử lý CCALF (cross component adaptive loop filtering-lọc vòng thích nghi thành phần chéo) theo khía cạnh thứ nhất. Dòng xử lý 1000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong bước S1001, xử lý lọc được áp dụng tới các mẫu ảnh được khôi phục của thành phần thứ nhất. Thành phần thứ nhất có thể là, ví dụ, thành phần độ chói. Thành phần độ chói có thể được biểu diễn là thành phần Y. Các mẫu ảnh được khôi phục của độ chói có thể là các tín hiệu đầu ra của xử lý ALF. Các tín hiệu đầu ra của ALF có thể là các mẫu độ chói được khôi phục được tạo ra thông qua xử lý SAO. Trong một vài phương án, xử lý lọc này được thực hiện trong bước S1001 có thể được biểu diễn là xử lý CCALF. Các số lượng của các mẫu độ chói được khôi phục có thể tương tự như số lượng hệ số của bộ lọc cần được sử dụng trong xử lý CCALF. Trong các phương án khác, xử lý cắt có thể được thực hiện trên các mẫu độ chói được khôi phục được lọc.

Trong bước S1002, mẫu ảnh được khôi phục của thành phần thứ hai được cải biến. Thành phần thứ hai có thể là thành phần sắc độ. Thành phần sắc độ có thể được biểu diễn là thành phần Cb và/hoặc Cr. Các mẫu ảnh được khôi phục của sắc độ có thể là các tín hiệu đầu ra của xử lý ALF. Các tín hiệu đầu ra của ALF có thể là các mẫu sắc độ được khôi phục được tạo ra thông qua xử lý SAO. Mẫu ảnh được khôi phục được cải biến có thể là tổng của các mẫu được khôi phục của sắc độ và các mẫu được khôi phục được lọc của độ chói, mà là đầu ra của bước S1001. Nói cách khác, xử lý cải biến có thể được thực hiện bằng cách cộng giá trị được lọc của các mẫu độ chói được khôi phục được tạo ra bởi xử lý CCALF của bước S1001 với giá trị được lọc của các mẫu sắc độ được khôi phục được tạo ra bởi xử lý ALF. Trong một vài phương án, xử lý cắt có thể được thực hiện trên các mẫu sắc độ được khôi phục. Thành phần thứ nhất và thành phần thứ hai có thể thuộc về cùng khối hoặc có thể thuộc về các khối khác nhau.

Trong bước S1003, giá trị của mẫu ảnh được khôi phục được cải biến của thành phần sắc độ được cắt. Bằng cách thực hiện xử lý cắt, giá trị của các mẫu có thể được đảm bảo nằm trong phạm vi được xác định. Ngoài ra, việc cắt có thể hỗ trợ sự hội tụ tốt hơn trong xử lý tối ưu hóa bình phương tối thiểu, v.v., để tối thiểu hóa độ chênh lệch giữa phần dư (độ chênh lệch giữa giá trị mẫu gốc và giá trị mẫu được khôi phục) và giá trị được lọc của các mẫu sắc độ để xác định các hệ số lọc.

Trong bước S1004, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được cắt của thành phần sắc độ. Trong một vài phương án, bước S1003 không cần được thực hiện. Trong trường hợp này, ảnh được giải mã nhờ sử dụng mẫu sắc độ được khôi phục được cải biến mà không được cắt.

FIG.98 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế. Trong phương án này, xử lý cắt được áp dụng tới mẫu ảnh được khôi phục được cải biến của thành phần sắc độ, như trong bước S1003 của FIG.97. Ví dụ, mẫu ảnh được khôi phục được cải biến có thể được cắt để nằm trong đoạn từ $[0, 1023]$ đối với 10 bit đầu ra. Khi các mẫu ảnh được khôi phục được lọc của thành phần độ chói được tạo ra bởi xử lý CCALF được cắt, có thể không cần thiết cắt mẫu ảnh được khôi phục được cải biến của thành phần sắc độ trong một vài phương án.

FIG.99 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế. Trong phương án này, xử lý cắt được áp dụng tới mẫu ảnh được khôi phục được cải biến của thành phần sắc độ, như trong bước S1003 của FIG.97. Xử lý cắt không được áp dụng tới các mẫu độ chói được khôi phục được lọc được tạo ra bởi xử lý CCALF. Giá trị được lọc của các mẫu sắc độ được khôi phục được tạo ra bởi xử lý ALF không cần được cắt, như được thể hiện bởi chỉ dẫn “Không cắt” trong FIG.99. Nói cách khác, mẫu ảnh được khôi phục cần được cải biến được tạo ra nhờ sử dụng giá trị được lọc (sắc độ ALF) và giá trị chênh lệch (CCALF Cb/Cr), trong đó việc cắt không được áp dụng tới đầu ra của giá trị mẫu được tạo ra.

FIG.100 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo phương án của sáng chế. Trong phương án này, xử lý cắt được áp dụng tới

các mẫu độ chói được khôi phục được lọc được tạo ra bởi xử lý CCALF (“cắt các mẫu đầu ra”) và các mẫu ảnh được khôi phục được cải biến của thành phần sắc độ (“cắt sau khi cộng tổng”). Giá trị được lọc của các mẫu sắc độ được khôi phục được tạo ra bởi xử lý ALF không được cắt (“Không cắt”). Theo ví dụ của sáng chế, phạm vi được cắt được áp dụng tới mẫu ảnh được khôi phục được lọc của thành phần độ chói có thể là $[-2^{15}, 2^{15}-1]$ hoặc $[-2^7, 2^7-1]$.

FIG.101 thể hiện ví dụ khác trong đó xử lý cắt được áp dụng tới các mẫu độ chói được khôi phục được lọc được tạo ra bởi xử lý CCALF (“Cắt các mẫu đầu tới”), tới các mẫu ảnh được khôi phục được cải biến của thành phần sắc độ (“Cắt sau khi cộng tổng”), và tới các mẫu sắc độ được khôi phục được lọc được tạo ra bởi xử lý ALF (“Cắt”). Nói cách khác, các giá trị đầu ra từ xử lý CCALF và xử lý sắc độ ALF được cắt riêng biệt, và được cắt lần nữa sau khi chúng được cộng tổng. Trong phương án này, mẫu ảnh được khôi phục được cải biến của thành phần sắc độ không cần được cắt. Theo ví dụ của sáng chế, đầu ra cuối cùng từ xử lý sắc độ ALF có thể được cắt thành giá trị 10 bit. Theo ví dụ của sáng chế, phạm vi được cắt được áp dụng tới các mẫu ảnh được khôi phục được lọc của thành phần độ chói có thể là $[-2^{15}, 2^{15}-1]$ hoặc $[-2^7, 2^7-1]$. Phạm vi này có thể là cố định hoặc có thể được xác định thích nghi. Trong trường hợp này, phạm vi này có thể được báo hiệu trong thông tin tiêu đề, ví dụ, trong SPS (Sequence Parameter set-Tập hợp tham số chuỗi) hoặc APS (Adaptation Parameter set-Tập hợp tham số chuỗi thích nghi). Trong trường hợp khi ALF phi tuyến được sử dụng, các tham số cắt có thể được xác định đối với “Cắt sau khi cộng tổng” trong FIG.101.

Các mẫu ảnh được khôi phục của thành phần độ chói cần được lọc bởi xử lý CCALF có thể là các mẫu lân cận mà liền kề với mẫu ảnh được khôi phục hiện tại của thành phần sắc độ. Tức là, mẫu ảnh được khôi phục hiện tại được cải biến có thể được tạo ra bằng cách cộng giá trị được lọc của các mẫu ảnh lân cận của thành phần độ chói nằm liền kề mẫu ảnh hiện tại với giá trị được lọc của mẫu ảnh hiện tại của thành phần sắc độ. Giá trị được lọc của các mẫu ảnh của thành phần độ chói có thể được biểu diễn là giá trị chệch lệch.

Các xử lý được bọc lộ trong khía cạnh này có thể làm giảm kích cỡ bộ nhớ bên trong phần cứng được yêu cầu để lưu trữ các giá trị mẫu ảnh được lọc.

(Khía cạnh thứ hai)

FIG.102 là lưu đồ về ví dụ của dòng xử lý 2000 để giải mã ảnh mà áp dụng xử lý CCALF sử dụng thông tin được xác định, theo khía cạnh thứ hai. Dòng xử lý 2000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong bước S2001, tham số cắt được phân tích từ dòng bit. Tham số cắt có thể được phân tích từ VPS, APS, SPS, PPS, thông tin tiêu đề lát, tại mức CTU hoặc TU, như được mô tả trong các FIG.103A đến FIG.103F. FIG.103A-130F là các sơ đồ khái niệm thể hiện các vị trí của các tham số cắt. Tham số được mô tả trong mỗi FIG.103A-103F có thể được thay thế bởi loại tham số cắt, cờ, hoặc chỉ số khác. Hai tham số cắt hoặc nhiều hơn có thể được phân tích từ hai tập hợp tham số hoặc nhiều hơn trong dòng bit.

Trong bước S2002, độ chênh lệch được cắt nhờ sử dụng tham số cắt. Độ chênh lệch được tạo ra dựa trên các mẫu ảnh được khôi phục của thành phần thứ nhất (ví dụ, giá trị chênh lệch (CCALF Cb/Cr) trong các FIG.98-101). Theo ví dụ của sáng chế, thành phần thứ nhất là thành phần độ chói và độ chênh lệch là các mẫu độ chói được khôi phục được lọc được tạo ra bởi xử lý CCALF. Trong trường hợp này, xử lý cắt được áp dụng tới các mẫu độ chói được khôi phục được lọc nhờ sử dụng tham số cắt được phân tích.

Tham số cắt giới hạn giá trị nằm trong phạm vi mong muốn. Nếu phạm vi mong muốn là $[-3, 3]$, ví dụ, giá trị 5 được cắt thành 3 nhờ sử dụng toán tử clip($-3, 3, 5$). Trong ví dụ này, giá trị -3 là phạm vi phía dưới và giá trị 3 là phạm vi phía trên.

Tham số cắt có thể chỉ báo chỉ số để thu nhận phạm vi phía dưới và phạm vi phía trên, như được thể hiện trên FIG.104(i). Trong ví dụ này, `ccalf_luma_clip_idx[]` là chỉ số, `-range_array[]` là phạm vi phía dưới, và `range_array[]` là phạm vi phía trên. Trong ví dụ này, `range_array[]` là mảng phạm vi được xác định mà có thể khác với mảng phạm vi được sử dụng cho ALF. Mảng phạm vi được xác định có thể được xác định trước.

Tham số cắt có thể chỉ báo phạm vi phía dưới và phạm vi phía trên, như được thể hiện trên FIG.104(ii). Trong ví dụ này, `-ccalf_luma_clip_low_range[]` là phạm vi phía dưới, và `ccalf_luma_clip_up_range[]` là phạm vi phía trên.

Tham số cắt có thể chỉ báo phạm vi chung cho cả phạm vi phía dưới và phạm vi phía trên, như được thể hiện trên FIG.104(iii). Trong ví dụ này, `-ccalf_luma_clip_range` là phạm vi phía dưới, và `ccalf_luma_clip_range` là phạm vi phía trên.

Độ chênh lệch được tạo ra bằng cách nhân, chia, cộng hoặc trừ ít nhất hai mẫu ảnh được khôi phục của thành phần thứ nhất. Hai mẫu ảnh được khôi phục, ví dụ, có thể đến từ các mẫu ảnh hiện tại hoặc lân cận hoặc hai mẫu ảnh lân cận. Các vị trí của các mẫu ảnh hiện tại và lân cận có thể được xác định trước.

Trong bước S2003, mẫu ảnh được khôi phục của thành phần thứ hai khác với thành phần thứ nhất được cải biến nhờ sử dụng giá trị được cắt. Giá trị được cắt có thể là giá trị được cắt của mẫu ảnh được khôi phục của thành phần độ chói. Thành phần thứ hai có thể là thành phần sắc độ. Việc cải biến có thể bao gồm toán tử nhân, chia, cộng hoặc trừ giá trị được cắt đối với mẫu ảnh được khôi phục của thành phần thứ hai.

Trong bước S2004, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được cải biến.

Theo sáng chế, một hoặc nhiều tham số cắt để lọc vòng thích nghi thành phần chéo được báo hiệu trong dòng bit. Với báo hiệu này, cú pháp của việc lọc vòng thích nghi thành phần chéo và cú pháp của bộ lọc vòng thích nghi có thể được kết hợp để đơn giản hóa cú pháp. Ngoài ra, với báo hiệu này, thiết kế của việc lọc vòng thích nghi thành phần chéo có thể linh hoạt hơn để cải thiện hiệu quả mã hóa.

Các tham số cắt có thể được xác định hoặc được xác định trước đối với cả bộ mã hóa và bộ giải mã mà không được báo hiệu. Các tham số cắt này có thể cũng được thu nhận nhờ sử dụng thông tin độ chói mà không được báo hiệu. Ví dụ, các tham số cắt tương ứng với phạm vi cắt lớn có thể được thu nhận nếu gradien mạnh hoặc biên được phát hiện trong ảnh được khôi phục độ chói, và các tham số cắt tương

ứng với phạm vi cắt ngắn có thể được thu nhận nếu gradien yếu hoặc biên được phát hiện trong ảnh được khôi phục độ chói.

(Khía cạnh thứ ba)

FIG.105 là lưu đồ của ví dụ về dòng xử lý 3000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng hệ số lọc theo khía cạnh thứ ba. Dòng xử lý 3000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 của FIG.67, v.v. Hệ số lọc được sử dụng trong bước lọc của xử lý CCALF để tạo ra mẫu ảnh được khôi phục được lọc của thành phần độ chói.

Trong bước S3001, được xác định rằng hệ số lọc có nằm bên trong vùng đối xứng được xác định của bộ lọc hay không. Một cách tùy chọn, bước bổ sung để đánh giá rằng dạng của các hệ số lọc là đối xứng hay không có thể được thực hiện. Thông tin mà chỉ báo rằng mẫu của các hệ số lọc là đối xứng hay không có thể được mã hóa vào dòng bit. Nếu dạng là đối xứng, các vị trí của các hệ số mà nằm bên trong vùng đối xứng có thể được xác định hoặc được xác định trước.

Trong bước S3002, nếu hệ số lọc nằm bên trong vùng đối xứng được xác định (Có trong bước S3001), hệ số lọc được sao chép tới vị trí đối xứng và tập hợp của các hệ số lọc được tạo ra.

Trong bước S3003, các hệ số lọc được sử dụng để lọc các mẫu ảnh được khôi phục của thành phần thứ nhất. Thành phần thứ nhất có thể là thành phần độ chói.

Trong bước S3004, đầu ra của việc lọc được sử dụng để cải biến mẫu ảnh được khôi phục của thành phần thứ hai khác với thành phần thứ nhất. Thành phần thứ hai có thể là thành phần sắc độ.

Trong bước S3005, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được cải biến.

Nếu các hệ số lọc là không đối xứng (Không trong bước S3001), tất cả các hệ số lọc có thể được mã hóa từ dòng bit và tập hợp của các hệ số lọc có thể được tạo ra mà không sao chép.

Khía cạnh này có thể làm giảm lượng thông tin cần được mã hóa vào dòng bit. Tức là, chỉ một trong số các hệ số lọc mà là đối xứng có thể cần được mã hóa trong dòng bit.

Các FIG.106A-106H, 107A-107H, 108A-108H, 109A-109D, và 110A-110D là các sơ đồ khái niệm về các vị trí chỉ báo ví dụ của các hệ số lọc cần được sử dụng trong xử lý CCALF. Trong các ví dụ này, một vài hệ số được chứa trong tập hợp của các hệ số được báo hiệu, giả định tính đối xứng tồn tại.

Cụ thể, các FIG.106A, FIG.106B, FIG.106C, và FIG.106D chỉ báo các ví dụ trong đó một phần của tập hợp của các hệ số CCALF (được đánh dấu bởi các đường chéo và các mẫu hình chấm) nằm bên trong vùng đối xứng được xác định. Trong các ví dụ này, các vùng đối xứng có dạng đối xứng dòng. Chỉ một vài hệ số được đánh dấu (được đánh dấu bởi các đường chéo hoặc các mẫu hình chấm) và các hệ số có màu trắng có thể được mã hóa vào dòng bit và các hệ số khác có thể được tạo ra bằng cách sử dụng các hệ số được mã hóa. Theo các ví dụ khác, chỉ các hệ số được đánh dấu có thể được tạo ra và được sử dụng trong xử lý lọc. Các hệ số có màu trắng khác (không được đánh dấu bởi bất kỳ mẫu) không cần được sử dụng trong xử lý lọc.

Các FIG.106E, FIG.106F, FIG.106G, và FIG.106H thể hiện các ví dụ trong đó dạng của vùng đối xứng là theo chiều ngang, chiều dọc, đường chéo, dọc theo chiều, đối xứng điểm, hoặc đối xứng điểm với chiều.

Trong các hình vẽ này, chỉ một phần của các hệ số mà được đánh dấu bởi các đường chéo hoặc các mẫu hình chấm có thể cần được mã hóa. Các vị trí của các hệ số cần được mã hóa có thể được xác định hoặc được xác định trước. Ví dụ, các hệ số có thể được mã hóa trong thứ tự quét được xác định và hệ số xuất hiện trước có thể được mã hóa trước. Hệ số trong vùng đối xứng mà hệ số của nó không được mã hóa có thể được sao chép từ hệ số mà nằm tại vị trí đối xứng của nó. Trong một vài phương án, có thể không cần thiết để xử lý các hệ số dựa trên tính đối xứng. Ví dụ, khi được xác định rằng hệ số thứ i là tương tự hệ số thứ j trong thứ tự quét, xử lý có thể chỉ sao chép giá trị thứ j thành giá trị thứ i . Vị trí này có thể được xác định dựa trên các tham số khác.

Các FIG.107A, FIG.107B, FIG.107C, và FIG.107D chỉ báo các ví dụ trong đó một phần của tập hợp của các hệ số CCALF (được đánh dấu bởi các đường chéo và các mẫu hình chấm) nằm bên trong vùng đối xứng được xác định. Trong các ví dụ này, số lượng hệ số đối xứng có thể khác nhau. Số lượng hệ số đối xứng có thể được xác định, được xác định trước, hoặc có thể được báo hiệu tại mức ảnh, mức lát, hoặc mức khối.

Các FIG.107E, FIG.107F, FIG.107G, và FIG.106H chỉ báo các ví dụ trong đó một phần của tập hợp của các hệ số CCALF (được đánh dấu bởi các đường chéo và các mẫu hình chấm) nằm bên trong vùng đối xứng được xác định. Trong các ví dụ này, các hệ số đối xứng trong một phía đối xứng có thể khác với các hệ số tương ứng trong phía đối xứng còn lại, tức là, một vài hệ số (ví dụ, tập hợp của các hệ số) trong một phía là đối xứng với các giá trị hệ số khác nhau (ví dụ, tập hợp của các hệ số khác) trong phía còn lại. Theo ví dụ của sáng chế, chỉ một phần của các hệ số “mẫu hình chấm” trong một phía đối xứng có thể được mã hóa vào dòng bit và được sao chép để tạo ra các hệ số “mẫu dòng đường chéo” trong phía đối xứng còn lại.

Các FIG.108A, FIG.108B, FIG.108C, và FIG.108D thể hiện các ví dụ của các dạng lọc trong đó loại sắc độ đóng vai trò là khuôn dạng được xác định. Khuôn dạng được xác định có thể là YUV 420 Loại 0, chẳng hạn. Các hệ số được đánh dấu (các hệ số dòng đường chéo hoặc các hệ số mẫu hình chấm) là đối xứng về vị trí sắc độ của loại sắc độ. Dạng lọc có thể được thiết kế là đối xứng về vị trí sắc độ của các khuôn dạng YUV. Ví dụ, các dạng lọc này của các FIG.108A-FIG.108D có thể được sử dụng như là mặc định, và các dạng lọc khác có thể được xác định để được sử dụng trong xử lý lọc khi tham số được mã hóa trong dòng bit chỉ báo các khuôn dạng khác.

Các FIG.108E, FIG.108F, FIG.108G, và FIG.108H thể hiện các ví dụ của các dạng lọc trong đó loại sắc độ đóng vai trò là khuôn dạng được xác định. Khuôn dạng được xác định có thể là khuôn dạng sắc độ YUV khác với YUV 420 Loại 0. Các dạng lọc khác nhau có thể được sử dụng cho các khuôn dạng khác.

Các FIG.109A, FIG.109B, FIG.109C, và FIG.109D chỉ báo các ví dụ khác của các dạng lọc. Trong FIG.109A, số lượng hệ số đối xứng có thể là 0 và tất cả hệ

số được báo hiệu độc lập. Số lượng hệ số đối xứng không cần được mã hóa vào dòng bit. Trong FIG.109B, số lượng hệ số đối xứng có thể là một nửa của tất cả hệ số.

Các FIG.110A, FIG.110B, FIG.110C, và FIG.110D thể hiện các ví dụ khác của các dạng lọc và các tín hiệu cần được mã hóa với thứ tự quét được chỉ báo bởi các mũi tên. Trong FIG.110A, thứ tự quét mảnh được áp dụng tới các hệ số lọc bất kể loại đối xứng. Trong FIG.110B, thứ tự quét mảnh được áp dụng tới các hệ số lọc, bất kể loại đối xứng, và chỉ các hệ số có màu trắng và các hệ số có mẫu hình chấm được báo hiệu trong dòng bit theo thứ tự quét mảnh. Bộ giải mã có thể sử dụng LUT (Bảng tra cứu) để sao chép các hệ số có mẫu hình chấm để tạo ra các hệ số có mẫu dòng đường chéo. Trong FIG.110C, các hệ số mẫu hình chấm được bố trí trong vùng đối xứng được quét và được báo hiệu, và sau đó các hệ số có màu trắng được bố trí trong vùng bất đối xứng được quét và được báo hiệu. Trong FIG.110D, các hệ số được bố trí trong vùng bất đối xứng được quét và được báo hiệu, và sau đó các hệ số mẫu hình chấm được quét và được báo hiệu.

FIG.112 là các sơ đồ khái niệm của các ví dụ khác mà thể hiện các vị trí của các hệ số lọc cần được sử dụng trong xử lý CCALF. Trong các ví dụ này, các vị trí đối xứng, các vị trí, hoặc các số lượng của các hệ số trong tập hợp các hệ số lọc có thể thích ứng với loại sắc độ.

FIG.113 là sơ đồ khối minh họa cấu trúc chức năng của xử lý CCALF được thực hiện bởi bộ mã hóa và bộ giải mã theo phương án của sáng chế. Sau khi lọc ảnh độ chói nhờ sử dụng các hệ số lọc được tạo ra, các mẫu đầu ra được áp dụng trên ảnh sắc độ. Bộ lọc với các hệ số lọc được tạo ra được áp dụng trên ảnh đầu ra độ chói SAO. Các mẫu được lọc (CC ALF Cb và CC ALF Cr) sau đó được thêm vào ảnh đầu ra sắc độ ALF.

(Khía cạnh thứ tư)

FIG.114 là lưu đồ về ví dụ của dòng xử lý 4000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng bộ lọc được lựa chọn từ các bộ lọc theo khía cạnh thứ tư. Dòng xử lý 4000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 của FIG.67, v.v. Phương án này bộc lộ các phương pháp để cải biến các mẫu được khôi phục của thành phần nhờ sử dụng thông tin từ thành phần khác nhau.

Trong bước S4001, tham số được xác định. Tham số có thể được phân tích từ VPS, APS, SPS, PPS, thông tin tiêu đề lát, hoặc tại mức CTU hoặc TU, như được mô tả trong các FIG.103-103F. Tham số được phân tích từ dòng bit để chỉ rõ bộ lọc. Ví dụ, tham số có thể chỉ báo chỉ số để lựa chọn bộ lọc từ nhiều bộ lọc được xác định. Tham số có thể được phân tích từ dòng bit để chỉ báo khuôn dạng lấy mẫu con sắc độ là 4:4:4, 4:2:0, 4:2:2, hoặc 4:1:1. Tham số có thể được phân tích từ dòng bit để chỉ báo không gian màu là YCbCr hoặc RGB. Tham số có thể được phân tích từ dòng bit để chỉ báo độ phân giải ảnh là 4K, FHD, CIF, QCIF. Tham số có thể chỉ báo thành phần màu là Y, Cb, hoặc Cr. Tham số có thể cũng được thu nhận nhờ sử dụng thông tin độ chói mà không được báo hiệu. Ví dụ, tham số tương ứng với bộ lọc nhánh ngắn có thể được thu nhận nếu gradient mạnh hoặc biên được phát hiện trong ảnh được khôi phục độ chói, và tham số tương ứng với bộ lọc nhánh dài có thể được thu nhận nếu gradient yếu hoặc biên được phát hiện trong ảnh được khôi phục độ chói. Theo ví dụ khác, tham số tương ứng với bộ lọc nhánh dài có thể được thu nhận nếu gradient mạnh hoặc biên được phát hiện trong ảnh được khôi phục độ chói, và tham số tương ứng với bộ lọc nhánh ngắn có thể được thu nhận nếu gradient yếu hoặc biên được phát hiện trong ảnh được khôi phục độ chói.

Trong bước S4002, bộ lọc được lựa chọn từ nhiều bộ lọc dựa trên tham số. Các bộ lọc có thể có các dạng hoặc kích cỡ khác nhau. Các bộ lọc có thể có cùng dạng và có các giá trị hệ số khác nhau. Tham số có thể chỉ báo các giá trị hệ số cần được sử dụng để tạo ra tập hợp của các hệ số lọc.

FIG.115 thể hiện ví dụ về dòng xử lý để lựa chọn bộ lọc.

Trong bước S4011, được xác định rằng tham số có chỉ báo khuôn dạng được xác định hay không. Khuôn dạng có thể được xác định trước. Khuôn dạng được xác định có thể chỉ báo khuôn dạng lấy mẫu con như là 4:4:4, 4:2:0, 4:2:2, hoặc 4:1:1. Khuôn dạng được xác định có thể chỉ báo thành phần màu là Y, Cb, hoặc Cr. Khuôn dạng được xác định có thể chỉ báo không gian màu như là YCbCr hoặc RGB. Khuôn dạng được xác định có thể chỉ báo độ phân giải ảnh như là 4K, FHD, CIF, QCIF.

Trong bước S4012, nếu được xác định rằng tham số chỉ báo khuôn dạng được xác định rằng (Có trong bước S4011), bộ lọc thứ nhất từ các bộ lọc được lựa chọn.

Trong bước S4013, nếu được xác định rằng tham số không chỉ báo khuôn dạng được xác định (Không trong bước S4011), bộ lọc khác với bộ lọc thứ nhất được lựa chọn từ nhiều bộ lọc. Dạng, kích cỡ, hoặc các giá trị của các hệ số lọc có thể khác nhau giữa S4012 và S4013.

Các FIG.116 và FIG.117 minh họa một vài ví dụ của bộ lọc. Trong FIG.116 mà thể hiện các bộ lọc (1A)-(1I), tổng số hàng mà có số lượng hệ số lớn nhất là số chẵn (ví dụ 2, 4, hoặc 6). Trong FIG.117 mà thể hiện các bộ lọc (2A)-(2I), tổng số hàng mà có số lượng hệ số lớn nhất là số lẻ (ví dụ 1, 3, hoặc 5).

Ví dụ, bộ lọc từ FIG.116 có thể được lựa chọn nếu tham số chỉ báo rằng khuôn dạng lấy mẫu con sắc độ 4:2:0 được áp dụng, trong khi bộ lọc từ FIG.117 có thể được lựa chọn nếu tham số chỉ báo rằng khuôn dạng lấy mẫu con sắc độ 4:2:2, 4:4:4, hoặc 4:1:1 được áp dụng. Việc lựa chọn các bộ lọc từ FIG.116 và FIG.117 có thể được dành riêng.

Ví dụ, bộ lọc từ FIG.116 có thể được lựa chọn nếu tham số chỉ báo rằng Y được sử dụng để cải biến Cb hoặc Cr, trong khi bộ lọc từ FIG.117 có thể được lựa chọn nếu tham số chỉ báo rằng Cb được sử dụng để cải biến Cr, hoặc Cr được sử dụng để cải biến Cb. Việc lựa chọn các bộ lọc từ FIG.116 và FIG.117 có thể được dành riêng.

Ví dụ, bộ lọc từ FIG.116 có thể được lựa chọn nếu tham số chỉ báo rằng không gian màu YCbCr được áp dụng, trong khi bộ lọc từ FIG.117 có thể được lựa chọn nếu tham số chỉ báo rằng không gian màu RGB được áp dụng. Việc lựa chọn các bộ lọc từ FIG.116 và FIG.117 có thể được dành riêng.

Ví dụ, bộ lọc thứ nhất từ FIG.116 có thể được lựa chọn nếu tham số chỉ báo rằng độ phân giải ảnh là lớn (ví dụ 4K hoặc 8K), trong khi bộ lọc khác với bộ lọc thứ nhất từ FIG.116 có thể được lựa chọn nếu tham số chỉ báo rằng độ phân giải ảnh là nhỏ (ví dụ QCIF hoặc CIF). Kích cỡ của hai bộ lọc được lựa chọn có thể khác nhau. Ví dụ, bộ lọc (1a) có thể được lựa chọn đối với độ phân giải ảnh QCIF, bộ lọc (1c) có thể được lựa chọn đối với độ phân giải ảnh FHD, và bộ lọc (1e) có thể được lựa chọn đối với độ phân giải ảnh 8K.

Ví dụ, bộ lọc thứ nhất từ FIG.117 có thể được lựa chọn nếu tham số chỉ báo rằng độ phân giải ảnh là lớn (ví dụ 4K hoặc 8K), trong khi bộ lọc khác với bộ lọc thứ nhất từ FIG.117 có thể được lựa chọn nếu tham số chỉ báo rằng độ phân giải ảnh là nhỏ (ví dụ QCIF hoặc CIF). Kích cỡ của hai bộ lọc được lựa chọn có thể khác nhau. Ví dụ, bộ lọc (2a) có thể được lựa chọn đối với độ phân giải ảnh QCIF, bộ lọc (2c) có thể được lựa chọn đối với độ phân giải ảnh FHD, và bộ lọc (2e) có thể được lựa chọn đối với độ phân giải ảnh 8K.

Trong bước S4003, các mẫu ảnh được khôi phục của thành phần thứ nhất được lọc nhờ sử dụng bộ lọc được lựa chọn. Thành phần thứ nhất có thể là thành phần độ chói. Xử lý lọc chứa ít nhất toán tử nhân, chia, cộng hoặc trừ trên ít nhất hai mẫu ảnh được khôi phục của thành phần thứ nhất. Ví dụ, hai mẫu ảnh được khôi phục có thể đến từ các mẫu ảnh lân cận và hiện tại, hoặc có thể đến từ hai mẫu ảnh lân cận. Các vị trí của các mẫu ảnh hiện tại và lân cận có thể được xác định trước.

Trong bước S4004, mẫu ảnh được khôi phục của thành phần thứ hai khác với thành phần thứ nhất được cải biến nhờ sử dụng đầu ra của việc lọc. Thành phần thứ hai có thể là thành phần sắc độ. Việc cải biến bao gồm toán tử nhân, chia, cộng hoặc trừ đầu ra của việc lọc với mẫu ảnh được khôi phục.

Tại bước S4005, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được cải biến.

Sáng chế liên quan đến việc lựa chọn thích nghi một bộ lọc từ nhiều bộ lọc để lọc thành phần chéo. Các bộ lọc khác nhau có thể có các dạng hoặc kích cỡ khác nhau. Việc lựa chọn thích nghi của bộ lọc làm cho việc lọc thành phần chéo linh hoạt hơn để cải thiện hiệu quả mã hóa.

Nhiều hơn một tập hợp bộ lọc có thể được báo hiệu. Các tập hợp bộ lọc khác nhau có thể có các dạng và kích cỡ khác nhau. Bộ lọc nào cần được sử dụng có thể được phân tích hoặc được xác định sau đây (ví dụ từ `filter_id`).

(Khía cạnh thứ năm)

FIG.118 là lưu đồ về ví dụ của dòng xử lý 5000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ năm. Dòng xử lý 5000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong bước S5001, tham số thứ nhất được phân tích từ dòng bit. Tham số thứ nhất có thể được phân tích từ VPS, APS, SPS, PPS, thông tin tiêu đề lát, hoặc tại mức CTU hoặc TU (FIG.103A-103F, trong đó “tham số” tương ứng với “tham số thứ nhất”). Tham số thứ nhất có thể chỉ báo khuôn dạng lấy mẫu con sắc độ là 4:4:4, 4:2:0, 4:2:2, hoặc 4:1:1. Tham số thứ nhất có thể chỉ báo không gian màu như là YCbCr hoặc RGB. Tham số thứ nhất có thể chỉ báo độ phân giải ảnh là 4K, FHD, CIF, QCIF. Tham số thứ nhất có thể chỉ báo thành phần màu là Y, Cb, hoặc Cr.

Trong bước S5002, được xác định rằng tham số thứ nhất có bằng giá trị được xác định hay không. Giá trị được xác định có thể được xác định trước.

Trong bước S5003, nếu được xác định rằng tham số thứ nhất bằng giá trị được xác định (Có trong bước S5002), số lượng hệ số thứ nhất được phân tích từ dòng bit. Số lượng hệ số thứ nhất có thể được phân tích từ VPS, APS, SPS, PPS, thông tin tiêu đề lát, hoặc tại mức CTU hoặc TU (FIG.103A-103F, trong đó “tham số” tương ứng với “số lượng hệ số thứ nhất”).

Trong bước S5004, nếu được xác định rằng tham số thứ nhất không bằng giá trị được xác định (Không trong bước S5003), số lượng hệ số thứ hai không bằng số lượng hệ số thứ nhất được phân tích từ dòng bit. Số lượng hệ số thứ hai có thể được phân tích từ VPS, APS, SPS, PPS, thông tin tiêu đề lát, hoặc tại mức CTU hoặc mức TU (FIG.103A-103F, trong đó “tham số” tương ứng với “số lượng hệ số thứ hai”). Số lượng thứ nhất và số lượng thứ hai từ bước S5002 và bước S5003 có thể khác nhau.

Ví dụ, như được thể hiện trên FIG.119(i), số lượng hệ số (NumCoeff) khi tham số thứ nhất chỉ báo rằng khuôn dạng lấy mẫu con sắc độ 4:2:0 được áp dụng khác với số lượng hệ số khi tham số thứ nhất chỉ báo rằng khuôn dạng lấy mẫu con sắc độ 4:2:2, 4:4:4, hoặc 4:1:1 được áp dụng.

Ví dụ, như được thể hiện trên FIG.119 (ii), số lượng hệ số khi tham số thứ nhất chỉ báo rằng không gian màu YCbCr được áp dụng khác với số lượng hệ số khi tham số thứ nhất chỉ báo rằng không gian màu RGB được áp dụng.

Ví dụ, như được thể hiện trên FIG.119 (iii), số lượng hệ số khi tham số thứ nhất chỉ báo rằng Y được sử dụng để cải biến Cb hoặc Cr khác với số lượng hệ số khi tham số thứ nhất chỉ báo rằng Cb được sử dụng để cải biến Cr, hoặc Cr được sử dụng để cải biến Cb.

Ví dụ, như được thể hiện trên FIG.119 (iv), số lượng hệ số khi tham số thứ nhất chỉ báo rằng độ phân giải ảnh là lớn (ví dụ 4K hoặc 8K) khác với số lượng hệ số khi tham số thứ nhất chỉ báo rằng độ phân giải ảnh là nhỏ (ví dụ QCIF hoặc CIF).

Trong FIG.119 (i)-(iv), thông tin như khuôn dạng lấy mẫu con sắc độ và độ phân giải ảnh có thể được thu nhận nếu SPS_id được mã hóa trong APS.

Trong bước S5005, các mẫu ảnh được khôi phục của thành phần thứ nhất được lọc nhờ sử dụng các hệ số được phân tích. Xử lý lọc chứa ít nhất toán tử nhân, chia, cộng hoặc trừ trên ít nhất hai mẫu ảnh được khôi phục của thành phần thứ nhất. Ví dụ, hai mẫu ảnh được khôi phục có thể đến từ các mẫu ảnh lân cận và hiện tại, hoặc có thể đến từ hai mẫu ảnh lân cận. Các vị trí của các mẫu ảnh hiện tại và lân cận có thể được xác định trước.

Trong bước S5006, mẫu ảnh được khôi phục của thành phần khác với thành phần thứ nhất được cải biến nhờ sử dụng đầu ra của việc lọc. Việc cải biến bao gồm toán tử nhân, chia, cộng hoặc trừ đầu ra của việc lọc với mẫu ảnh được khôi phục.

Trong bước S5007, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được cải biến.

Sáng chế đề cập đến việc thu nhận thích nghi số lượng hệ số lọc để lọc thành phần chéo. Việc thu nhận thích nghi của số lượng hệ số lọc làm cho việc lọc thành phần chéo linh hoạt hơn để cải thiện hiệu quả mã hóa.

Nhiều hơn một tập hợp hệ số có thể được báo hiệu. Các tập hợp hệ số khác nhau có thể có số lượng hệ số khác nhau. Các tập hợp hệ số khác nhau có thể có cùng số lượng hệ số. Số lượng hệ số của các tập hợp hệ số có thể được cố định. Tập hợp

hệ số nào cần được sử dụng được phân tích hoặc được xác định sau đây (ví dụ từ `coeff_set_id`, hoặc `filter_id`).

(Khía cạnh thứ sáu)

FIG.120 là lưu đồ về ví dụ của dòng xử lý 6000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ sáu. Dòng xử lý 6000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong S6001, xử lý lựa chọn ít nhất tập hợp của các mẫu được khôi phục từ thành phần thứ nhất;

Trong S6002, xử lý thu nhận giá trị dựa trên tập hợp của các mẫu được khôi phục được lựa chọn;

Trong S6003, xử lý lọc các mẫu được khôi phục dựa trên giá trị thu được;

Trong S6004, xử lý cải biến mẫu ảnh được khôi phục của thành phần thứ hai nhờ sử dụng đầu ra của lọc;

Trong S6005, xử lý giải mã ảnh sử dụng mẫu ảnh được khôi phục được lọc.

Các FIG.121, FIG.122, và FIG.123 là các sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói (xem bước S6002) đối với mẫu sắc độ hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận. Nói cách khác, trong ví dụ này, giá trị CCALF của các mẫu độ chói đối với mẫu sắc độ được tạo ra bằng cách tính toán tổng có trọng số của các mẫu độ chói nằm trong vùng lân cận của mẫu sắc độ. Các mẫu độ chói bao gồm mẫu nằm tại liền kề với mẫu sắc độ.

Trong FIG.121, vị trí được chỉ báo bởi dạng kim cương là vị trí của mẫu sắc độ hiện tại. Ví dụ, giá trị tương ứng với vị trí (curr) đối với CCALF có thể được thu nhận bằng cách tính toán giá trị trung bình của các mẫu độ chói lân cận mà được đánh dấu bởi các mẫu hình chấm. Các mẫu độ chói có màu trắng không cần được sử dụng đối với xử lý tính giá trị trung bình. Nói cách khác, giá trị đối với CCALF có thể được thu nhận bằng cách viện dẫn tới giá trị mẫu của mẫu độ chói nằm liền kề với mẫu sắc độ hiện tại. Có hai mẫu độ chói trong ví dụ của FIG.121.

FIG.122 mô tả các phương trình mẫu để tính toán giá trị CCALF. Giá trị CCALF có thể được thu nhận bằng cách sử dụng các giá trị hệ số lọc và các giá trị mẫu độ chói. Giá trị hệ số lọc được nhân với kết quả của phép trừ của hai giá trị mẫu độ chói lân cận. Mẫu độ chói được sử dụng trong mỗi tính toán phép trừ có thể được bố trí liền kề với mẫu sắc độ hiện tại. Dạng của các công thức có thể tương tự như dạng được sử dụng trong xử lý lọc ALF. Trong một vài phương án, nếu giá trị hệ số lọc nhỏ hơn 64, giá trị hệ số có thể được thiết lập là 0.

Như được mô tả trong FIG.123, các số lượng mẫu độ chói khác nhau có thể được trung bình hóa, và số lượng mẫu độ chói lân cận được trung bình hóa có thể được xác định trước, hoặc được báo hiệu trong/tại mức ảnh, lát, hoặc khối. Các vị trí của các mẫu độ chói lân cận được trung bình hóa có thể được xác định trước, hoặc được báo hiệu trong/tại mức ảnh, lát, hoặc khối. Các trọng số của các mẫu độ chói lân cận được trung bình hóa có thể được xác định trước, hoặc được báo hiệu trong/tại mức ảnh, lát, hoặc khối.

Các FIG.124 và FIG.125 là các sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói đối với mẫu hiện tại bằng cách tính toán giá trị trung bình có trọng số của các mẫu lân cận, trong đó các vị trí của các mẫu lân cận được xác định thích nghi theo loại sắc độ. Nói cách khác, các vị trí của các mẫu độ chói cần được sử dụng trong tính toán trọng số được xác định dựa trên vị trí của mẫu sắc độ hiện tại.

Các mẫu được đánh dấu với các mẫu khác nhau có thể biểu diễn các trọng số khác nhau. Số lượng mẫu được trung bình hóa có thể thích ứng (có thể tương ứng với) loại sắc độ. Các trọng số được trung bình hóa có thể thích ứng (có thể tương ứng với) loại sắc độ.

Các FIG.126 và FIG.127 là các sơ đồ khái niệm minh họa các ví dụ về việc tạo ra giá trị CCALF của thành phần độ chói bằng cách áp dụng việc dịch bit tới giá trị đầu ra của việc tính toán trọng số. Nói cách khác, xử lý dịch chuyển biến đổi xuống được áp dụng tới giá trị được lọc của các mẫu độ chói trong cùng cách thức như trong xử lý ALF. Trong một vài phương án, nếu giá trị hệ số nhỏ hơn 64, giá trị hệ số có thể được thiết lập là 0.

Số lượng bit dịch đối với CCALF được biểu diễn là x . Giá trị x có thể được xác định như là cùng giá trị trong xử lý ALF. Trong một vài ví dụ giá trị x có thể được cố định là 10. Trong một vài ví dụ giá trị x có thể được cố định là 7.

(Khía cạnh thứ bảy)

FIG.128 là lưu đồ về ví dụ của dòng xử lý 7000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ bảy. Dòng xử lý 7000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 của FIG.67, v.v. Các phương pháp xác định các mẫu được khôi phục cần được lọc sử dụng một hoặc nhiều tham số được mô tả.

Trong bước S7001, một hoặc nhiều tham số được phân tích từ dòng bit. Một hoặc nhiều tham số có thể được mã hóa trong ít nhất một trong số APS, SPS, PPS, thông tin tiêu đề lát hoặc mức CTU, như được thể hiện trên FIG.129A-129E. FIG.130A-130D thể hiện các xử lý mẫu để truy hồi một hoặc nhiều tham số.

Một hoặc nhiều tham số có thể nằm trong SPS. Lát đầu tiên xác định vị trí PPS theo PPS_id mà được mã hóa trong lát. PPS sau đó xác định vị trí SPS theo SPS_id mà được mã hóa trong PPS. Thông qua điều này, lát có thể truy hồi một hoặc nhiều tham số trong SPS như được thể hiện trên FIG.130A.

Một hoặc nhiều tham số có thể nằm trong tập hợp tham số tại mức ảnh, ví dụ, trong PPS. Lát đầu tiên xác định vị trí PPS theo PPS_id mà được mã hóa trong lát. Thông qua điều này, lát có thể truy hồi một hoặc nhiều tham số trong PPS như được thể hiện trên FIG.130B.

Một hoặc nhiều tham số có thể nằm trong APS. Lát đầu tiên xác định vị trí APS theo APS_id mà được mã hóa trong lát. Thông qua điều này, lát có thể truy hồi một hoặc nhiều tham số trong APS như được thể hiện trên FIG.130C.

Một hoặc nhiều tham số có thể nằm trong lát (FIG.130D). Lát có thể thu nhận một hoặc nhiều tham số từ thông tin tiêu đề bên trong hoặc dữ liệu.

Một hoặc nhiều tham số có thể bao gồm tham số thứ nhất mà lựa chọn kích cỡ của các mẫu cần được cải biến. Một hoặc nhiều tham số có thể chỉ báo rằng xử lý CCALF có được kích hoạt hay không. Một hoặc nhiều tham số có thể bao gồm các

tham số mà chỉ báo rằng xử lý CCALF có được kích hoạt hay không và tham số mà chỉ báo các giá trị hệ số của bộ lọc cần được sử dụng.

Các mẫu có thể được nhóm lại trong dạng hình vuông mà có kích cỡ cụ thể như 4x4, 8x8, 16x16, 32x32, 64x64, hoặc 128x128 mẫu.

Tham số thứ nhất có thể được phân tích trước khi phân tích thông tin tiêu đề lát hoặc dữ liệu lát. Ví dụ, tham số thứ nhất có thể được phân tích từ APS, SPS, hoặc PPS.

Tham số thứ nhất có thể được phân tích từ thông tin tiêu đề lát.

Tham số thứ nhất có thể được phân tích từ dữ liệu đơn vị cây mã hóa (CTU).

Tham số thứ nhất có thể phụ thuộc vào loại lấy mẫu con sắc độ hoặc kích cỡ CTU hoặc cả hai. Nếu loại lấy mẫu con sắc độ là 4:4:4, kích cỡ khối được lựa chọn bởi tham số thứ nhất có thể là 4x4, 8x8, 16x16, 32x32, 64x64, hoặc 128x128. Nếu loại lấy mẫu con sắc độ là 4:2:2 hoặc 4:2:0, kích cỡ khối được lựa chọn bởi tham số thứ nhất có thể là 4x4, 8x8, 16x16, 32x32, hoặc 64x64.

Tham số thứ nhất có thể phụ thuộc vào kích cỡ CTU trong đó kích cỡ khối được lựa chọn không thể vượt quá kích cỡ CTU của nó.

Một hoặc nhiều tham số có thể bao gồm tham số thứ hai mà chỉ báo rằng khối của các mẫu cần được lọc.

Tham số thứ hai có thể là cờ với giá trị bằng 1 hoặc 0, trong đó 1 chỉ báo để cải biến các mẫu được khôi phục và 0 chỉ báo không cải biến các mẫu được khôi phục.

Tham số thứ hai có thể được phân tích trước khi phân tích dữ liệu đơn vị cây mã hóa (CTU-coding tree unit) thứ nhất. Ví dụ, tham số thứ hai có thể được phân tích từ APS, SPS, PPS, thông tin tiêu đề lát, hoặc dữ liệu lát.

Tham số thứ hai có thể được phân tích từ dữ liệu đơn vị cây mã hóa (CTU). FIG. 131A-131C thể hiện các giá trị mẫu của tham số thứ hai. Trong FIG.131A, tham số thứ hai = 1, có nghĩa là “sử dụng lọc đối với khối” chẳng hạn. Trong FIG.131B, tham số thứ hai = 0, có nghĩa là “không sử dụng lọc đối với khối” chẳng hạn. Trong FIG.131C, tham số thứ nhất lựa chọn kích cỡ khối 32x32, và tham số thứ hai = 1000

(trong giá trị nhị phân), khối 0 = 1, có nghĩa là “sử dụng lọc đối với khối 0,” khối 1 = 0, có nghĩa là “không sử dụng lọc đối với khối 1,” khối 2 = 0, có nghĩa là “không sử dụng lọc đối với khối 2,” và khối 3 = 0, có nghĩa là “không sử dụng lọc đối với khối 3,” chẳng hạn. Các tham số thứ hai có thể chỉ báo rằng các khối trong đơn vị cây mã hóa (CTU) mà có các kích cỡ cụ thể cần được cải biến.

Các tham số thứ hai có thể chỉ báo rằng việc cải biến của các mẫu được khôi phục có vị vô hiệu hóa trong ảnh hoặc chuỗi hay không. Nếu các tham số thứ hai chỉ báo rằng việc cải biến của các mẫu được khôi phục bị vô hiệu hóa, bước S7002 của FIG.128 sẽ chuyển sang bước S7005 một cách trực tiếp, mà tương ứng với nhánh “Không” trong FIG.128.

Một hoặc nhiều tham số có thể bao gồm tham số mà có thể được phân tích nhờ sử dụng mã hóa phi số học như mã hóa độ dài cố định, mã hóa Golomb theo số mũ, hoặc VLC.

Một hoặc nhiều tham số có thể bao gồm tham số mà có thể được phân tích nhờ sử dụng mã hóa số học như CAVLC hoặc CABAC.

Ví dụ, như được thể hiện trên FIG.132, tham số thứ hai có thể được phân tích nhờ sử dụng mã hóa số học trước khi phân tích dữ liệu đơn vị cây mã hóa thứ nhất trong lát, được theo sau bởi dữ liệu căn chỉnh byte hoặc dữ liệu căn chỉnh bit. FIG.132 thể hiện ví dụ về việc căn chỉnh byte và vị trí khởi tạo mã hóa số học (CABAC). Trong ví dụ này, việc khởi tạo mã hóa số học để phân tích các tham số sau tham số thứ hai trong cùng lát có thể được áp dụng.

Trong bước S7002, được xác định rằng bộ lọc có cần được sử dụng dựa trên các tham số được phân tích hay không.

Nếu bộ lọc cần được sử dụng, trong bước S7003, ít nhất mẫu được khôi phục từ thành phần thứ nhất được lọc. Thành phần thứ nhất có thể là các mẫu độ chói.

Trong bước S7004, các mẫu được khôi phục được cải biến nhờ sử dụng ít nhất một mẫu được lọc được khôi phục từ thành phần khác với thành phần thứ nhất. Thành phần khác với thành phần thứ nhất có thể là các mẫu sắc độ.

Trong bước S7005, khối của các mẫu ảnh được giải mã sử dụng các mẫu được khôi phục được cải biến.

Sáng chế minh họa các đặc tính của một hoặc nhiều tham số tại nhiều mức để lọc, bao gồm các phương pháp tạo lập, các chức năng và các phương pháp mã hóa. Nhờ sử dụng các tham số điều khiển này, thiết kế của việc lọc có thể được tối ưu hóa để tiết kiệm các bit mã hóa, nâng cao các thành phần tần số cao nhờ sử dụng các mẫu được cải biến, và làm giảm độ dư thừa giữa các kênh khác nhau, để nhờ đó cải biến chất lượng ảnh.

FIG.133 là sơ đồ khái niệm về biến thể của phương án này.

Một hoặc nhiều tham số có thể phụ thuộc vào phân vùng của đơn vị cây mã hóa. Khi phân vùng có kích cỡ khác với kích cỡ được chỉ báo bởi tham số thứ nhất, tham số thứ hai mà chỉ báo phân vùng có được lọc hay không không được mã hóa và việc lọc của phân vùng bị vô hiệu hóa. Trong ví dụ này, các bit được mã hóa của các tham số thứ hai được làm giảm.

Dạng hoặc các mẫu được mô tả trong khía cạnh thứ bảy có thể được thay thế bởi hình chữ nhật hoặc phân vùng dạng không phải chữ nhật. Các ví dụ về phân vùng dạng không phải chữ nhật có thể là ít nhất một trong số phân vùng dạng tam giác, phân vùng dạng chữ L, phân vùng dạng ngũ giác, phân vùng dạng lục giác và phân vùng dạng đa giác như được thể hiện trên FIG.133.

(Khía cạnh thứ tám)

FIG.134 là lưu đồ về ví dụ của dòng xử lý 8000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ tám. Dòng xử lý 8000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong bước S8001, được xác định rằng mẫu thứ nhất của thành phần thứ nhất có nằm ngoài biên ảo hay không.

Trong bước S8002, nếu được xác định rằng mẫu thứ nhất của thành phần thứ nhất nằm ngoài biên ảo, mẫu thứ hai của thành phần thứ nhất được sao chép tới mẫu thứ nhất, trong đó mẫu thứ hai nằm bên trong biên ảo.

Trong bước S8003, mẫu được khôi phục của thành phần thứ nhất mà bao gồm các mẫu thứ nhất và thứ hai được lọc.

Trong bước S8004, mẫu được khôi phục của thành phần khác với thành phần thứ nhất được cải biến nhờ sử dụng đầu ra của việc lọc.

Trong bước S8005, mẫu được khôi phục được cải biến được sử dụng để giải mã ảnh.

Phương pháp tạo đệm trong S8002 có thể là giống nhau bất kể khuôn dạng mẫu sắc độ. Ví dụ, việc tạo đệm đối xứng có thể được sử dụng. Phương pháp tạo đệm có thể được thay đổi phụ thuộc vào khuôn dạng mẫu sắc độ, giữa tạo đệm đối xứng và tạo đệm không đối xứng chẳng hạn.

Thành phần thứ nhất có thể các mẫu độ chói và thành phần khác mà các giá trị mẫu của chúng được cải biến có thể là các mẫu sắc độ. Sắc độ có thể là Cb, Cr, hoặc cả hai.

FIG.135 là lưu đồ về ví dụ của dòng xử lý 8100 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ tám. Dòng xử lý 8100 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong S8101, được xác định rằng loại mẫu sắc độ có phải loại thứ nhất hay không. FIG.136 thể hiện các vị trí ví dụ của các loại mẫu sắc độ 0 đến 5. Cụ thể, FIG.136 thể hiện các loại mẫu sắc độ và các vị trí của chúng so với các vị trí mẫu độ chói đối với 4:2:0, chẳng hạn.

Trong S8102, nếu được xác định rằng loại mẫu sắc độ là loại thứ nhất, mẫu thứ nhất và mẫu thứ hai được sử dụng trong việc lọc các mẫu được khôi phục của thành phần thứ nhất, trong đó mẫu thứ nhất được sao chép từ mẫu được khôi phục thứ hai.

Trong S8103, nếu được xác định rằng loại mẫu sắc độ không phải loại thứ nhất, mẫu thứ hai và mẫu thứ ba được sử dụng trong việc lọc các mẫu được khôi phục của thành phần thứ nhất, và mẫu thứ nhất được loại trừ, trong đó mẫu thứ ba khác với mẫu thứ nhất hoặc thứ hai.

Trong bước S8104, mẫu được khôi phục từ thành phần khác với thành phần thứ nhất được cải biến nhờ sử dụng đầu ra của việc lọc. Ví dụ, thành phần thứ nhất đang chỉ báo độ chói và đầu ra của việc lọc được thêm vào mẫu được khôi phục từ thành phần sắc độ. Theo ví dụ khác, thành phần thứ nhất đang chỉ báo sắc độ Cr và đầu ra của việc lọc được thêm vào mẫu được khôi phục từ sắc độ Cb. Theo ví dụ khác, đầu ra của việc lọc và mẫu được khôi phục có thể được cộng, trừ, nhân, chia, hoặc được đưa vào bất kỳ kết hợp của các xử lý toán học để thu nhận mẫu được khôi phục được cải biến.

Trong bước S8105, mẫu được khôi phục được cải biến được sử dụng để giải mã ảnh. Ví dụ, mẫu được khôi phục được cải biến được lưu trữ trong bộ đệm ảnh tham chiếu.

Các FIG.137A-137D, FIG.138, và FIG.139 là các sơ đồ khái niệm của các ví dụ về việc tạo đệm đối xứng.

Ví dụ, trong S8002 của FIG.134, nếu loại mẫu sắc độ bằng 0 hoặc 1 và biên ảo nằm giữa C0 và C2 như được thể hiện trên FIG.137A, giá trị mẫu được khôi phục của mẫu thứ hai (C15) được sao chép tới mẫu thứ nhất (C17). Tương tự, giá trị mẫu được khôi phục của mẫu thứ hai (C2) được sao chép tới mẫu thứ nhất (C0).

Theo ví dụ khác, trong S8002, nếu loại mẫu sắc độ bằng 0 hoặc 1 và biên ảo nằm giữa C15 và C17 như được thể hiện trên FIG.137B, giá trị mẫu được khôi phục của mẫu thứ hai (C2) được sao chép tới mẫu thứ nhất (C0). Tương tự, giá trị mẫu được khôi phục của mẫu thứ hai (C15) được sao chép tới mẫu thứ nhất (C17).

Theo ví dụ khác, trong S8002, nếu loại mẫu sắc độ bằng 0 hoặc 1 và biên ảo nằm giữa C2 và C6 như được thể hiện trên FIG.137C, các giá trị mẫu được khôi phục của các mẫu thứ hai (C10, C11 và C12) được sao chép tới các mẫu thứ nhất (C14, C15, C16 và C17). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0, C1, C2 và C3).

Theo ví dụ khác, trong S8002, nếu loại mẫu sắc độ bằng 0 hoặc 1 và biên ảo nằm giữa C11 và C15 như được thể hiện trên FIG.137D, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0,

C1, C2 và C3). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C10, C11 và C12) được sao chép tới các mẫu thứ nhất (C14, C15, C16 và C17).

Các FIG.138 và FIG.139 thể hiện các ví dụ về các mẫu mà được sao chép trong FIG.137A và FIG.137C, một cách lần lượt.

Sau khi sao chép, các mẫu được sao chép được sử dụng trong việc lọc của các mẫu được khôi phục của thành phần thứ nhất.

FIG.140A-140D, FIG.141, FIG.142, và FIG.143 là các sơ đồ khái niệm về các ví dụ của việc tạo đệm phi đối xứng.

Ví dụ, trong S8002, nếu loại mẫu sắc độ bằng 2 hoặc 3 và biên ảo nằm giữa C0 và C2 như được thể hiện trên FIG.140A, các giá trị mẫu được khôi phục của các mẫu thứ hai (C10, C11 và C12) được sao chép tới các mẫu thứ nhất (C14, C15, C16 hoặc C17). Tương tự, giá trị mẫu được khôi phục của mẫu thứ hai (C2) được sao chép tới mẫu thứ nhất (C0).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 2 hoặc 3 và biên ảo nằm giữa C15 và C17 như được thể hiện trên FIG.140B, giá trị mẫu được khôi phục của mẫu thứ hai (C15) được sao chép tới mẫu thứ nhất (C17).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 2 hoặc 3 và biên ảo nằm giữa C2 và C6 như được thể hiện trên FIG.140C, các giá trị mẫu được khôi phục của các mẫu thứ hai (C4, C5, C6, C7 và C8) được sao chép tới các mẫu thứ nhất (C9, C10, C11, C12, C13, C14, C15, C16 và C17). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0, C1, C2 và C3).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 2 hoặc 3 và biên ảo nằm giữa C11 và C15 như được thể hiện trên FIG.140D, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0, C1, C2 và C3).

Các FIG.141, FIG.142, và FIG.143 thể hiện các ví dụ về các mẫu mà được sao chép trong FIG.140A, FIG.140C và FIG.140D, một cách lần lượt.

Sau khi sao chép, các mẫu được sao chép được sử dụng trong việc lọc của các mẫu được khôi phục của thành phần thứ nhất.

Các FIG.144A-144D, FIG.145, FIG.146, và FIG.147 là các sơ đồ khái niệm về các ví dụ khác của việc tạo đệm phi đối xứng.

Trong ví dụ của S8002, nếu loại mẫu sắc độ bằng 4 hoặc 5 và biên ảo nằm giữa C0 và C2 như được thể hiện trên FIG.144A, giá trị mẫu được khôi phục của mẫu thứ hai (C2) được sao chép tới mẫu thứ nhất (C0).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 4 hoặc 5 và biên ảo nằm giữa C15 và C17 như được thể hiện trên FIG.144B, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0, C1, C2 và C3). Tương tự, giá trị mẫu được khôi phục của mẫu thứ hai (C15) được sao chép tới mẫu thứ nhất (C17).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 4 hoặc 5 và biên ảo nằm giữa C2 và C6 như được thể hiện trên FIG.144C, giá trị mẫu được khôi phục của mẫu thứ hai (C15) được sao chép tới mẫu thứ nhất (C17). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5, C6 và C7) được sao chép tới các mẫu thứ nhất (C0, C1, C2 và C3).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 4 hoặc 5 và biên ảo nằm giữa C11 và C15 như được thể hiện trên FIG.144D, các giá trị mẫu được khôi phục của các mẫu thứ hai (C9, C10, C11, C12 và C13) được sao chép tới các mẫu thứ nhất (C0, C1, C2, C3, C4, C5, C6, C7 và C8).

Các FIG.145, FIG.146, và FIG.147 thể hiện các ví dụ của các mẫu, mà được sao chép trong FIG.144B, FIG.144C và FIG.144D, một cách lần lượt.

Sau khi sao chép, các mẫu được sao chép được sử dụng trong việc lọc của các mẫu được khôi phục của thành phần thứ nhất.

Các FIG.148A-148D, FIG.149, và FIG.150 là các sơ đồ khái niệm về các ví dụ khác của việc tạo đệm đối xứng.

Trong ví dụ của S8002, nếu loại mẫu sắc độ bằng 0, 2 hoặc 4 và biên ảo nằm giữa C4 và C5 như được thể hiện trên FIG.148A, các giá trị mẫu được khôi phục của

các mẫu thứ hai (C7 và C12) được sao chép tới các mẫu thứ nhất (C8 và C13). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5 và C10) được sao chép tới các mẫu thứ nhất (C4 và C9).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 0, 2 hoặc 4 và biên ảo nằm giữa C7 và C8 như được thể hiện trên FIG.148B, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5 và C10) được sao chép tới các mẫu thứ nhất (C4 và C9). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C7 và C12) được sao chép tới các mẫu thứ nhất (C8 và C13).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 0, 2 hoặc 4 và biên ảo nằm giữa C5 và C6 như được thể hiện trên FIG.148C, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C1, C4, C5, C9, C10 và C14). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C3, C7, C8, C12, C13 và C16).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 0, 2 hoặc 4 và biên ảo nằm giữa C6 và C7 như được thể hiện trên FIG.148D, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C1, C4, C5, C9, C10 và C14). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C3, C7, C8, C12, C13 và C16).

FIG.149 và FIG.150 thể hiện các ví dụ của các mẫu mà được sao chép trong FIG.148A và FIG.148C, một cách lần lượt.

Sau khi sao chép, các mẫu được sao chép được sử dụng trong việc lọc của các mẫu được khôi phục của thành phần thứ nhất.

Các FIG.151A-151C, FIG.152, FIG.153, FIG.154, và FIG.155A-155L là các sơ đồ khái niệm của các ví dụ khác của việc tạo đệm phi đối xứng.

Trong ví dụ của S8002, nếu loại mẫu sắc độ bằng 1, 3 hoặc 5 và biên ảo nằm giữa C4 và C5 như được thể hiện trên FIG.151A, các giá trị mẫu được khôi phục của các mẫu thứ hai (C5 và C10) được sao chép tới các mẫu thứ nhất (C4 và C9).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 1, 3 hoặc 5 và biên ảo nằm giữa C7 và C8 như được thể hiện trên FIG.151B, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C1, C4, C5, C9, C10 và C14). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C7 và C12) được sao chép tới các mẫu thứ nhất (C8 và C13).

Theo ví dụ khác của S8002, nếu loại mẫu sắc độ bằng 1, 3 hoặc 5 và biên ảo nằm giữa C5 và C6 như được thể hiện trên FIG.151C, các giá trị mẫu được khôi phục của các mẫu thứ hai (C7 và C12) được sao chép tới các mẫu thứ nhất (C8 và C13). Tương tự, các giá trị mẫu được khôi phục của các mẫu thứ hai (C2, C6, C11 và C15) được sao chép tới các mẫu thứ nhất (C1, C4, C5, C9, C10 và C14).

Các FIG.152, FIG.153 và FIG.154 thể hiện các ví dụ của các mẫu, mà được sao chép trong FIG.151A, FIG.151B và FIG.151C, một cách lần lượt.

FIG.155A-155L thể hiện các ví dụ về đệm khác với biên ảo theo chiều dọc và ngang.

Sau khi sao chép, các mẫu được sao chép được sử dụng trong việc lọc của các mẫu được khôi phục của thành phần thứ nhất.

Sáng chế minh họa việc tạo đệm hoặc sao chép các mẫu được sử dụng trong bộ lọc dựa trên loại mẫu sắc độ và vị trí biên ảo trong bộ lọc. Phương pháp tạo đệm hoặc sao chép các mẫu này cải thiện chất lượng ảnh.

(Các biến thể)

Loại mẫu sắc độ có thể được thay thế bởi thông tin khác, mà chỉ báo quan hệ giữa thành phần thứ nhất và thành phần khác mà khác với thành phần thứ nhất.

Loại mẫu sắc độ có thể được thay thế bởi cờ, mà lựa chọn việc tạo đệm đối xứng hoặc phi đối xứng tại biên ảo, trong đó 0 lựa chọn tạo đệm đối xứng và 1 lựa chọn tạo đệm phi đối xứng, hoặc 1 lựa chọn tạo đệm đối xứng và 0 lựa chọn tạo đệm phi đối xứng.

Cờ có thể được báo hiệu từ dòng bit hoặc có thể được thu nhận.

Giá trị mặc định của cờ có thể là tạo đệm đối xứng tại biên ảo.

Giá trị mặc định của cờ có thể là tạo đệm phi đối xứng tại biên ảo.

Ví dụ, cờ có thể được thu nhận dựa trên trạng thái bật/tắt của bộ lọc khác. Ví dụ, nếu bộ lọc ALF là bật, cờ có thể lựa chọn tạo đệm đối xứng. Nếu bộ lọc ALF là tắt, cờ có thể lựa chọn tạo đệm phi đối xứng.

Theo ví dụ khác, nếu cờ ALF là bật, cờ có thể lựa chọn tạo đệm phi đối xứng. Nếu bộ lọc ALF là tắt, cờ có thể lựa chọn tạo đệm đối xứng.

Các bộ lọc khác mà có thể được sử dụng bao gồm LMCS, SAO, DBF, và các bộ lọc phía sau khác.

Trong một vài phương án, cờ có thể được thiết lập dựa trên mẫu.

Biên ảo có thể được thay thế bởi ảnh, lát, viên, ô, hoặc biên ảnh con.

FIG.156 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo ví dụ trong đó việc tạo đệm đối xứng được sử dụng trên các vị trí biên ảo đối với ALF, và việc tạo đệm đối xứng hoặc phi đối xứng được sử dụng trên các vị trí biên ảo đối với CC-ALF dựa trên loại mẫu sắc độ và vị trí biên ảo.

FIG.157 là sơ đồ khối minh họa cấu trúc chức năng của bộ mã hóa và bộ giải mã theo ví dụ khác trong đó đệm đối xứng được sử dụng trên các vị trí biên ảo đối với ALF và đệm một phía được sử dụng trên các vị trí biên ảo đối với CC-ALF.

FIG.158A-158H là các sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc hoặc ngang.

FIG.159A-159C là các sơ đồ khái niệm minh họa các ví dụ về đệm một phía với biên ảo theo chiều dọc và ngang.

Đầu vào tới CCALF (các mẫu được khôi phục của thành phần thứ nhất được sử dụng để lọc) không bị giới hạn ở đầu ra SAO. Đầu vào có thể là từ đầu ra của ánh xạ độ chói với biến đổi tỷ lệ sắc độ (LMCS), hai chiều/hadamard, hoặc lọc giải khối hoặc bất kỳ các kết hợp của bộ lọc sau.

CCALF có thể được chuyển đổi có hiệu lực/vô hiệu tại mỗi khối. Khối không cần xếp chồng giữa nhiều hơn một CTU. Do ALF có thể được chuyển đổi có hiệu

lực/vô hiệu tại CTU, CCALF có thể được chuyển đổi có hiệu lực đối với CTU trong đó ALF có hiệu lực.

Một hoặc nhiều khía cạnh được mô tả lại ở đây có thể được thực hiện kết hợp với ít nhất một phần của khía cạnh khác trong sáng chế. Ngoài ra, một hoặc nhiều khía cạnh được bộc lộ ở đây có thể được thực hiện bằng cách kết hợp, với các khía cạnh khác, một phần của các xử lý được thể hiện trong bất kỳ các lưu đồ theo các khía cạnh, một phần của cấu trúc của bất kỳ các thiết bị, một phần của các cú pháp, v.v. Các khía cạnh được mô tả có viện dẫn tới thành phần cấu thành của bộ mã hóa có thể được thực hiện tương tự bởi thành phần cấu thành tương ứng của bộ giải mã.

(Khía cạnh thứ chín)

FIG.160 là lưu đồ về ví dụ của dòng xử lý 9000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tham số theo khía cạnh thứ chín. Dòng xử lý 9000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong S9001, tham số của xử lý CCALF được giải mã. Trong xử lý này, tham số thứ nhất cần được sử dụng để xác định mô hình để giải mã entropy tham số thứ hai có thể được xác định. Tham số thứ nhất đối với thành phần C_b và tham số thứ nhất đối với thành phần C_r có thể có cùng giá trị. Theo ví dụ của sáng chế, tham số thứ nhất có thể là giá trị k mà chỉ báo thứ tự của các bit nhị phân của mã Golomb theo số mũ. Giá trị k là chỉ số tương ứng với mô hình của các giá trị báo hiệu trong xử lý giải mã entropy. Tham số thứ nhất có thể là tham số khác mà chỉ báo phương pháp giải mã entropy. Bộ giải mã có thể lựa chọn mô hình từ nhiều mô hình nhờ viện dẫn tới tham số thứ nhất. Theo ví dụ của sáng chế, tham số thứ hai có thể là tham số mà chỉ báo giá trị hệ số lọc của bộ lọc được sử dụng trong xử lý CCALF, tức là, tham số thứ hai có thể được giải mã với mã Golomb số mũ nhờ sử dụng các thứ tự được chỉ báo bởi giá trị k . Giá trị k giống nhau có thể được áp dụng tới tất cả hệ số lọc được chứa trong bộ lọc của CCALF.

Trong S9002, các mẫu ảnh được khôi phục của thành phần thứ nhất được lọc. Thành phần thứ nhất có thể là thành phần độ chói.

Trong S9003, mẫu ảnh được khôi phục của thành phần thứ hai được cải biến nhờ sử dụng giá trị được lọc. Thành phần thứ hai có thể là thành phần sắc độ bao gồm thành phần Cr và thành phần Cb.

Trong S9004, ảnh được giải mã nhờ sử dụng mẫu ảnh được khôi phục được lọc.

FIG.161 là sơ đồ khái niệm về ví dụ của bộ lọc cần được áp dụng trong xử lý CCALF. “Sum” có thể có giá trị tương ứng với giá trị của mẫu được mô tả như là mẫu dạng kim cương. “Sum” có thể có giá trị tương ứng với giá trị của mẫu khác. “Sum” có thể được thu nhận bằng cách áp dụng lọc tới các mẫu độ chói lân cận được đánh dấu bởi các mẫu hình chấm. Các mẫu lân cận có thể được bố trí liền kề với mẫu mà có cùng giá trị như “sum”. Các mẫu lân cận được đánh dấu trong màu trắng được bố trí liền kề với các mẫu được đánh dấu bởi các mẫu hình chấm không cần được sử dụng trong tính toán lọc.

FIG.162 là sơ đồ khái niệm của các công thức mẫu của xử lý lọc. Dạng của các công thức của xử lý CCALF có thể tương tự như dạng được sử dụng trong xử lý ALF. “Sum” có thể được thu nhận bằng cách thực hiện tính toán được mô tả trong FIG.162 nhờ sử dụng các giá trị hệ số lọc và các giá trị mẫu độ chói. Số lượng hệ số lọc có thể khác nhau.

FIG.163 là sơ đồ khái niệm về ví dụ của cú pháp liên quan đến CCALF. Các hệ số lọc $f[idx[k]]$ đối với Cb và Cr có thể được báo hiệu trong dòng bit là: `alf_cross_component_cb_coeff_abs`, `alf_cross_component_cb_coeff_sign`, `alf_cross_component_cr_coeff_abs`, và `alf_cross_component_cr_coeff_sign`, như được mô tả trong FIG.163.

FIG.164 là sơ đồ khái niệm về ví dụ của báo hiệu giá trị hệ số lọc sử dụng mã Golomb số mũ với bậc cố định k (được ký hiệu là EGk). Giá trị hệ số lọc được nhị phân hóa nhờ sử dụng các bit 0 và 1. Theo ví dụ của sáng chế, giá trị hệ số của Cb, `cb_coeff_abs` được mô tả là giá trị x có thể được báo hiệu tham chiếu tới mô hình được xác định bởi giá trị của k. Khi giá trị k là 1 (EG1) và giá trị x là 2, giá trị x được báo hiệu là 0100. Giá trị k có thể được kết hợp với quy tắc báo hiệu khác, ví dụ, giá trị x có thể được báo hiệu là 1011 khi giá trị k là 1 và giá trị x là 2.

FIG.165A, FIG.165B, FIG.165C, và FIG.165D là các sơ đồ khái niệm của các ví dụ của EGk được áp dụng tới các hệ số lọc. Theo ví dụ của sáng chế, các hệ số được đánh dấu bởi các mẫu hình chấm được báo hiệu nhờ sử dụng EG3, các hệ số được đánh dấu bởi các đường chéo được báo hiệu nhờ sử dụng EG2, và các hệ số mà không được đánh dấu (các hình tròn trắng) không được sử dụng trong xử lý lọc.

Trong FIG.165A, tất cả hệ số được báo hiệu nhờ sử dụng cùng giá trị EGk. Nói cách khác, tất cả hệ số được sử dụng cho xử lý lọc được báo hiệu trong cùng cách thức. Trong FIG.165B, chỉ một phần của các hệ số được báo hiệu nhờ sử dụng cùng giá trị EGk. Trong FIG.165C, chỉ một phần của các hệ số, mà được đánh dấu bởi các mẫu hình chấm, được sử dụng trong việc lọc và các hệ số được sử dụng trong xử lý lọc được báo hiệu nhờ sử dụng cùng giá trị EGk. Một phần của các hệ số được đánh dấu bởi các mẫu hình chấm có thể được báo hiệu. Trong FIG.165D, chỉ một phần của các hệ số, mà được đánh dấu bởi các mẫu hình chấm hoặc các mẫu đường chéo, được sử dụng trong xử lý lọc. Các hệ số được đánh dấu bởi các mẫu hình chấm và các hệ số được đánh dấu bởi các mẫu đường chéo được báo hiệu nhờ sử dụng các giá trị EGk khác nhau.

Khi điều kiện được thiết lập là thỏa mãn rằng tất cả hệ số cần được sử dụng trong xử lý lọc được báo hiệu trong cùng cách thức, các ví dụ được mô tả trong các FIG.165A và FIG.165C thỏa mãn điều kiện, và các ví dụ được mô tả trong các FIG.165B và FIG.165D không thỏa mãn điều kiện. Tức là, dưới điều kiện khi các hệ số lọc được sử dụng để thu nhận một giá trị “sum”, giá trị k mà chỉ báo thứ tự EG cần được sử dụng trong việc mã hóa và giải mã các giá trị của các hệ số lọc có thể chỉ có một giá trị.

Các FIG.166A và FIG.166B là các sơ đồ khái niệm của các ví dụ của EGk được áp dụng tới các hệ số lọc. Theo ví dụ của sáng chế, các hệ số được đánh dấu bởi các mẫu hình chấm được báo hiệu nhờ sử dụng EG3, và các hệ số được đánh dấu bởi các mẫu đường chéo được báo hiệu nhờ sử dụng EG2.

Trong FIG.166A, thứ tự EG của `cb_coeff_abs` có thể tương tự như `cr_coeff_abs`. Ví dụ, EG3 đối với `cb_coeff_abs` và EG3 đối với `cr_coeff_abs`.

Trong FIG.166B, thứ tự EG của `cb_coeff_abs` có thể khác với `cr_coeff_abs`. Ví dụ, EG3 đối với `cb_coeff_abs` nhưng EG2 đối với `cr_coeff_abs`.

Các FIG.167A và FIG.167B là các sơ đồ khái niệm của các ví dụ của EGk được áp dụng tới các hệ số lọc. Theo ví dụ của sáng chế, các hệ số được đánh dấu bởi các mẫu hình chấm được báo hiệu nhờ sử dụng EG3, các hệ số được đánh dấu bởi các mẫu đường chéo được báo hiệu nhờ sử dụng EG2, và các hệ số không được đánh dấu không được sử dụng trong xử lý lọc.

Trong FIG.167A, thứ tự EG của `cb_coeff_abs` có thể tương tự như `cr_coeff_abs`. Ví dụ, EG3 đối với `cb_coeff_abs` và EG3 đối với `cr_coeff_abs`.

Trong FIG.167B, thứ tự EG của `cb_coeff_abs` có thể khác với `cr_coeff_abs`. Ví dụ, EG3 đối với `cb_coeff_abs` nhưng EG2 đối với `cr_coeff_abs`.

Các FIG.168A và FIG.168B là các sơ đồ khái niệm của các ví dụ của EGk được áp dụng tới các hệ số lọc. Theo ví dụ của sáng chế, các hệ số được đánh dấu bởi các mẫu hình chấm được báo hiệu nhờ sử dụng EG3, các hệ số được đánh dấu bởi mẫu đường chéo được báo hiệu nhờ sử dụng EG2.

Trong FIG.168A, thứ tự EG của các CTU khác nhau có thể khác nhau. Ví dụ, EG3 đối với CTU1 nhưng EG2 đối với CTU2.

Trong FIG.168B, trong đó các mẫu được sử dụng để lọc, không phải tất cả hệ số sử dụng cùng EGk đối với CTU2.

Giá trị k mà chỉ báo thứ tự EG cần được sử dụng trong việc mã hóa và giải mã các hệ số lọc có thể được xác định như là một giá trị đối với mỗi CTU. Tức là, giá trị k có thể được xác định đối với mỗi CTU. Giá trị k có thể được xác định trong mức khác, như tại mức của CU, viên, lát, ô, ảnh con, ảnh, hoặc chuỗi. Giá trị k có thể được xác định trước hoặc được báo hiệu trong dòng bit.

Giá trị k cần được sử dụng để thu nhận giá trị mẫu của C_r và giá trị k cần được sử dụng để thu nhận giá trị mẫu của C_b có thể có cùng giá trị hoặc các giá trị khác nhau.

Các bộ lọc khác nhau có thể được áp dụng tới CU khác nhau, CTU hoặc lát của Cr hoặc Cb. Trong trường hợp này, giá trị k khác nhau có thể được sử dụng trong mỗi bộ lọc.

FIG.169 là sơ đồ khái niệm về ví dụ của cú pháp của các tham số được sử dụng trong xử lý ALF. Các giá trị hệ số đối với độ chói được biểu diễn là $\text{alf_luma_coeff_abs}$ và các giá trị hệ số đối với sắc độ được biểu diễn là $\text{alf_chroma_coeff_abs}$ có thể được báo hiệu nhờ sử dụng cùng giá trị k .

Trong FIG.169, $\text{alf_luma_coeff_abs}[sIdx][j]$ chỉ rõ giá trị tuyệt đối của hệ số thứ j của bộ lọc độ chói được báo hiệu được chỉ báo bởi $sIdx$. Khi $\text{alf_luma_coeff_abs}[sIdx][j]$ không được hiện diện, tham số này được xem là bằng 0.

Thứ tự k của nhị phân hóa Golomb số mũ $\text{uek}(v)$ được thiết lập bằng 3.

$\text{alf_chroma_coeff_abs}[altIdx][j]$ chỉ rõ giá trị tuyệt đối của hệ số lọc sắc độ thứ j đối với bộ lọc sắc độ khác với chỉ số $altIdx$. Khi $\text{alf_chroma_coeff_abs}[altIdx][j]$ không được hiện diện, tham số được xem là bằng 0. Yêu cầu của tuân theo dòng bit là các giá trị của $\text{alf_chroma_coeff_abs}[altIdx][j]$ sẽ nằm trong đoạn từ 0 đến $2^7 - 1$.

Thứ tự k của nhị phân hóa Golomb số mũ $\text{uek}(v)$ được thiết lập bằng 3.

FIG.170 là sơ đồ khái niệm về ví dụ của cú pháp của các tham số được sử dụng trong xử lý CCALF. Các giá trị hệ số đối với Cb được biểu diễn là $\text{alf_cross_component_cb_coeff_abs}$ và các giá trị hệ số đối với Cr được biểu diễn là $\text{alf_cross_component_cr_coeff_abs}$ có thể được báo hiệu nhờ sử dụng cùng giá trị k . Tức là, tham số giống nhau để mã hóa entropy các tham số CCALF có thể được áp dụng tới Cb và Cr.

Trong FIG.170, $\text{alf_cross_component_cb_coeff_abs}[j]$ chỉ rõ giá trị tuyệt đối của hệ số thứ j của bộ lọc thành phần chéo được báo hiệu Cb. Khi $\text{alf_cross_component_cb_coeff_abs}[j]$ không được hiện diện, tham số này được xem là bằng 0.

Thứ tự k của mã Golomb số mũ được thiết lập bằng 3.

$\text{alf_cross_component_cr_coeff_abs}[j]$ chỉ rõ giá trị tuyệt đối của hệ số thứ j của bộ lọc thành phần chéo được báo hiệu Cr. Khi $\text{alf_cross_component_cr_coeff_abs}[j]$ không được hiện diện, tham số này được xem là bằng 0.

Thứ tự k của mã Golomb số mũ được thiết lập bằng 3.

Trong các FIG.169 và FIG.170, cả hai hệ số lọc ALF và CCALF được báo hiệu nhờ sử dụng Egk cố định. Tức là, thứ tự EG k được cố định. Thứ tự EG k được cố định đối với ALF và CCALF có thể khác nhau. Ví dụ, 3 đối với ALF nhưng 4 đối với CCALF cb_coeff_abs và cr_coeff_abs . Thứ tự EG k được cố định đối với CCALF cb_coeff_a và cr_coeff_abs có thể khác nhau. Ví dụ, 3 đối với cb_coeff_abs nhưng 4 đối với cr_coeff_abs .

(Khía cạnh thứ mười)

FIG.171 là lưu đồ về ví dụ của dòng xử lý 10000 để giải mã ảnh mà áp dụng xử lý CCALF nhờ sử dụng tập hợp của các hệ số theo khía cạnh thứ mười. Dòng xử lý 10000 có thể được thực hiện, ví dụ, bởi bộ giải mã 200 trên FIG.67, v.v.

Trong S10001, tập hợp của các hệ số được phân tích. Số lượng hệ số được chứa trong tập hợp các hệ số bằng N .

Trong bước S10002, các mẫu ảnh được khôi phục của thành phần thứ nhất được lọc nhờ sử dụng tập hợp của các hệ số. Số lượng mẫu ảnh được khôi phục bằng N . Thành phần thứ nhất có thể là thành phần Y. Các mẫu ảnh được khôi phục được lọc có thể là đầu ra của tập hợp của các bộ lọc vòng thích nghi được áp dụng tới các mẫu độ chói được khôi phục được tạo ra sau xử lý SAO;

Trong S10003, mẫu ảnh được khôi phục của thành phần thứ hai được cải biến nhờ sử dụng giá trị được lọc. Thành phần thứ hai khác với thành phần thứ nhất. Thành phần thứ hai có thể là thành phần Cb/Cr. Mẫu ảnh được khôi phục được cải biến có thể là tổng của mẫu được lọc của thành phần thứ hai sau xử lý ALF và các mẫu được khôi phục được lọc của thành phần thứ nhất;

Trong S10004, mẫu ảnh được khôi phục được cải biến của thành phần thứ hai được giải mã. Mẫu ảnh được khôi phục có thể được cắt. Xử lý cắt được áp dụng

tới mẫu ảnh được khôi phục được cải biến; ví dụ, đối với 10 bit đầu ra, mẫu ảnh được khôi phục được cải biến có thể được cắt trong đoạn từ $[0, 1023]$.

Mẫu ảnh được khôi phục cần được cải biến được tạo ra nhờ sử dụng giá trị được lọc và giá trị chênh lệch. Việc cắt có thể không được áp dụng tới đầu ra của giá trị mẫu được tạo ra.

Trong xử lý ALF đối với sắc độ, đầu ra cuối cùng có thể được cắt thành giá trị 10 bit. Xử lý cắt có thể không được thực hiện trên đầu ra cuối cùng và xử lý cắt có thể được thực hiện trên mẫu ảnh được khôi phục được cải biến.

FIG.172A, FIG.172B, FIG.172C, và FIG.172D là các sơ đồ khái niệm của các ví dụ về dạng của các tập hợp của các hệ số lọc.

FIG.172A và FIG.172C thể hiện các ví dụ về bộ lọc CCALF 8 nhánh mà có các dạng kim cương 3×4 . Các vị trí của các hệ số lọc tương ứng với các vị trí của các mẫu ảnh được khôi phục của thành phần thứ nhất. Mẫu ảnh được khôi phục của thành phần thứ hai cần được cải biến có thể nằm tại vị trí được đánh dấu là 2. Mẫu ảnh được khôi phục của thành phần thứ hai cần được cải biến có thể có vị trí nằm giữa 2 và 5. Giá trị hệ số thứ nhất có thể được tạo ra bằng cách thu nhận độ chênh lệch giữa giá trị của mẫu ảnh được khôi phục thứ nhất nằm tại 2 và các giá trị của các mẫu ảnh được khôi phục thứ nhất nằm tại 0, 1, 3, 4, 5, 6, và 7.

FIG.173A, FIG.173B, FIG.173C, và FIG.173D là các sơ đồ khái niệm của các ví dụ về các vị trí của các mẫu được khôi phục của các thành phần thứ nhất và thành phần thứ hai..

Khi mẫu ảnh được khôi phục của thành phần thứ hai nằm tại (x, y) , vị trí của các mẫu được khôi phục của các thành phần thứ nhất được đưa ra trong FIG.173A, FIG.173B, FIG.173C, và FIG.173D.

Trong FIG.173A, số lượng hệ số lọc là 8. Mỗi hệ số có thể có giá trị khác nhau. Khi tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$. Giá trị hệ số thứ nhất được tạo ra bằng cách nhân giá trị hệ số lọc của bộ lọc CCALF với kết quả của phép trừ của giá

trị của một trong số các mẫu ảnh được khôi phục thứ nhất $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$ và giá trị của mẫu ảnh được khôi phục thứ nhất nằm tại $(2x, 2y)$. Xử lý thu nhận của giá trị hệ số thứ nhất có thể được thực hiện trong cùng cách thức như được mô tả trong phương án thứ sáu.

Trong FIG.173B, số lượng hệ số lọc là 6.

Trong FIG.173C, hệ số thứ nhất và cuối cùng có cùng giá trị. Có 7 hệ số.

Trong FIG.173D, các cấu trúc đối xứng được sử dụng để làm giảm số lượng hệ số.

Lưu ý rằng biên ảo như biên ảo ALF hoặc biên ảo CCALF có thể được áp dụng tới FIG.173A, FIG.173C, và FIG.173D. Biên ảo có thể không được áp dụng tới FIG.173B.

Lưu ý rằng CCALF có thể không được áp dụng tới vùng CU khi mã hóa không tồn hao được kích hoạt hoặc được sử dụng.

Xử lý biên ảo là quy tắc điều khiển để làm giảm số lượng hàng của các mẫu ảnh tham chiếu cần được sử dụng tới xử lý lọc phía xa biên, ví dụ, biên ô, biên lát, hoặc biên CTU, v.v. Theo một ví dụ, khi biên ảo được xác định là 2 hàng điểm ảnh, các hàng của các mẫu ảnh tham chiếu cần được tham chiếu qua biên là 2 hàng điểm ảnh và xử lý lọc được thực hiện nhờ sử dụng các điểm ảnh được tạo ra bởi xử lý tạo đệm. Theo ví dụ khác, khi dạng lọc là một trong số các dạng được mô tả trong FIG.173A, FIG.173C, hoặc FIG.173D, điểm ảnh lân cận liền kề với phía trên của điểm ảnh lân cận (x, y) được tham chiếu cho xử lý lọc và xử lý biên ảo được kích hoạt. Khi dạng lọc là dạng được mô tả trong FIG.173B, xử lý lọc có thể được thực hiện mà không vi phạm dẫn tới điểm ảnh lân cận liền kề với phía trên của điểm ảnh hiện tại và xử lý biên ảo có thể được vô hiệu hóa để làm giảm lượng xử lý.

Chế độ mã hóa không tồn hao là chế độ bỏ qua xử lý biến đổi và xử lý lượng tử hóa. Ảnh được mã hóa và ảnh được giải mã của CU là tương tự như ảnh đầu vào (không tồn hao) khi CU được xử lý nhờ sử dụng chế độ mã hóa không tồn hao. Xử lý CCALF có thể bị vô hiệu hóa trong chế độ không tồn hao. Khi xử lý CCALF bị

vô hiệu hóa trong chế độ mã hóa không tổn hao, việc cải biến giá trị của điểm ảnh bằng xử lý lọc bị vô hiệu hóa và có thể đảm bảo sự không tổn hao.

(Các phương án thực hiện và các ứng dụng)

Như được mô tả trong mỗi phương án nêu trên, mỗi khối chức năng hoặc hoạt động có thể thường được thực hiện như là MPU (micro processing unit-Bộ vi xử lý) và bộ nhớ, chẳng hạn. Ngoài ra, các xử lý được thực hiện bởi mỗi khối chức năng có thể được thực hiện như là bộ thực thi chương trình, như bộ xử lý mà đọc và thực thi phần mềm (chương trình) được ghi trên vật ghi như ROM. Phần mềm có thể được phân phối. Phần mềm có thể được ghi trên các vật ghi khác nhau như bộ nhớ bán dẫn. Lưu ý rằng mỗi khối chức năng có thể cũng được thực hiện như là phần cứng (mạch dành riêng). Các kết hợp khác nhau của phần cứng và phần mềm có thể được sử dụng.

Xử lý được mô tả trong mỗi phương án có thể được thực hiện thông qua xử lý được tích hợp nhờ sử dụng một thiết bị (hệ thống), và, ngoài ra, có thể được thực hiện thông qua xử lý phi tập trung nhờ sử dụng nhiều thiết bị. Ngoài ra, bộ xử lý mà thực thi chương trình nêu trên có thể có bộ một bộ xử lý hoặc nhiều bộ xử lý. Nói cách khác, việc xử lý được tích hợp có thể được thực hiện, và, ngoài ra, xử lý phi tập trung có thể được thực hiện.

Các phương án của sáng chế không bị giới hạn ở các phương án ví dụ nêu trên; các cải biến khác nhau có thể được thực hiện đối với các phương án ví dụ, các kết quả của chúng cũng nằm trong phạm vi của các phương án theo sáng chế.

Tiếp theo, các ví dụ ứng dụng của phương pháp mã hóa ảnh động (phương pháp mã hóa ảnh) và phương pháp giải mã ảnh động (phương pháp giải mã ảnh) được mô tả trong mỗi phương án nêu trên sẽ được mô tả, cũng như các hệ thống khác nhau mà thực hiện các ví dụ ứng dụng. Hệ thống này có thể được đặc trưng là bao gồm bộ mã hóa ảnh mà sử dụng phương pháp mã hóa ảnh, bộ giải mã ảnh mà sử dụng phương pháp giải mã ảnh, hoặc bộ mã hóa-bộ giải mã ảnh mà bao gồm cả hai bộ mã hóa ảnh và bộ giải mã ảnh. Các cấu trúc khác của hệ thống này có thể được cải biến trên cơ sở từng trường hợp.

(Các ví dụ sử dụng)

FIG.174 minh họa cấu trúc chung của hệ thống cung cấp nội dung ex100 thích hợp để thực hiện dịch vụ phân phối nội dung. Vùng trong đó dịch vụ truyền thông được cung cấp được chia thành các tế bào có các kích cỡ mong muốn, và các trạm gốc ex106, ex107, ex108, ex109, và ex110, mà là các trạm không dây cố định trong ví dụ được minh họa, được bố trí trong các tế bào tương ứng.

Trong hệ thống cung cấp nội dung ex100, các thiết bị bao gồm máy tính ex111, thiết bị trò chơi ex112, camera ex113, thiết bị gia đình ex114, và điện thoại thông minh ex115 được kết nối tới internet ex101 thông qua nhà cung cấp dịch vụ internet ex102 hoặc mạng truyền thông ex104 và các trạm gốc ex106 đến ex110. Hệ thống cung cấp nội dung ex100 có thể kết hợp và kết nối bất kỳ kết hợp của các thiết bị nêu trên. Theo các cách thức thực hiện khác nhau, các thiết bị này có thể được kết nối trực tiếp hoặc gián tiếp với nhau thông qua mạng điện thoại hoặc truyền thông trường gần, ngoài thông qua các trạm gốc ex106 đến ex110. Ngoài ra, máy chủ dòng ex103 có thể được kết nối tới các thiết bị bao gồm máy tính ex111, thiết bị trò chơi ex112, camera ex113, thiết bị gia đình ex114, và điện thoại thông minh ex115 thông qua, ví dụ, internet ex101. Máy chủ dòng ex103 có thể cũng được kết nối tới, ví dụ, thiết bị đầu cuối trong điểm truy nhập (hotspot) trong máy bay ex117 thông qua vệ tinh ex116.

Lưu ý rằng thay vì các trạm gốc ex106 đến ex110, các điểm truy nhập không dây hoặc hotspot có thể được sử dụng. Máy chủ dòng ex103 có thể được kết nối tới mạng truyền thông ex104 một cách trực tiếp thay vì thông qua internet ex101 hoặc nhà cung cấp dịch vụ internet ex102, và có thể được kết nối tới máy bay ex117 một cách trực tiếp thay vì thông qua vệ tinh ex116.

Camera ex113 có thể là thiết bị có khả năng chụp ảnh tĩnh và video, như camera số. Điện thoại thông minh ex115 có thể là thiết bị điện thoại thông minh, điện thoại tế bào, hoặc điện thoại hệ thống truyền thông cầm tay cá nhân (PHS-personal handyphone system) mà có thể hoạt động dưới các tiêu chuẩn hệ thống truyền thông di động của các hệ thống 2G, 3G, 3,9G, và 4G, cũng như hệ thống thế hệ tiếp theo 5G.

Thiết bị gia đình ex114, ví dụ, là tủ lạnh hoặc thiết bị nằm trong hệ thống đồng phát pin nhiên liệu gia đình.

Trong hệ thống cung cấp nội dung ex100, thiết bị đầu cuối bao gồm chức năng chụp ảnh và/hoặc video có khả năng, ví dụ, phát tạo dòng trực tiếp bằng cách kết nối tới máy chủ dòng ex103 thông qua, ví dụ, trạm gốc ex106. Khi phát tạo dòng trực tiếp, thiết bị đầu cuối (ví dụ, máy tính ex111, thiết bị trò chơi ex112, camera ex113, thiết bị gia đình ex114, điện thoại thông minh ex115, hoặc thiết bị đầu cuối trong máy bay ex117) có thể thực hiện xử lý mã hóa được mô tả trong các phương án nêu trên trên ảnh tĩnh hoặc nội dung video được thu bởi người dùng thông qua thiết bị đầu cuối, có thể ghép kênh dữ liệu video thu được thông qua việc mã hóa và dữ liệu audio thu được bằng cách mã hóa audio tương ứng với video, và có thể truyền dữ liệu thu được tới máy chủ dòng ex103. Nói cách khác, thiết bị đầu cuối thực hiện chức năng như bộ mã hóa ảnh theo một khía cạnh của sáng chế.

Máy chủ dòng ex103 tạo dòng dữ liệu nội dung được truyền tới các máy khách mà yêu cầu dòng. Các ví dụ máy khách bao gồm máy tính ex111, thiết bị trò chơi ex112, camera ex113, thiết bị gia đình ex114, điện thoại thông minh ex115, và các thiết bị đầu cuối bên trong máy bay ex117, mà có khả năng giải mã dữ liệu được mã hóa được mô tả nêu trên. Các thiết bị mà thu dữ liệu được tạo dòng có thể giải mã và tái tạo dữ liệu thu được. Nói cách khác, mỗi thiết bị này có thể thực thể chức năng như là bộ giải mã ảnh, theo một khía cạnh của sáng chế.

(Xử lý phi tập trung)

Máy chủ dòng ex103 có thể được thực hiện như là các máy chủ hoặc các máy tính mà giữa chúng các tác vụ như xử lý, ghi, và tạo dòng dữ liệu được phân chia. Ví dụ, máy chủ dòng ex103 có thể được thực hiện như là mạng phân phát nội dung (CDN-content delivery network) mà tạo dòng nội dung thông qua mạng mà kết nối các máy chủ biên nằm trên khắp thế giới. Trong CDN, máy chủ biên nằm gần về mặt vật lý với máy khách có thể được gán động tới máy khách. Nội dung được lưu trữ đệm và được tạo dòng tới máy chủ biên để làm giảm thời gian tải. Trong trường hợp của, ví dụ, một vài loại lỗi hoặc thay đổi về kết nối, ví dụ, do sự tăng vọt lưu lượng, có thể tạo dòng dữ liệu ổn định tại các tốc độ cao, do có thể tránh được các

phần bị ảnh hưởng của mạng, ví dụ, bằng cách phân chia xử lý giữa các máy chủ biên, hoặc chuyển đổi các nghĩa vụ tạo dòng tới máy chủ biên khác và tiếp tục tạo dòng.

Việc phi tập trung không bị giới hạn ở chỉ việc phân chia xử lý để tạo dòng; việc mã hóa dữ liệu thu được có thể được phân chia giữa và được thực hiện bởi các thiết bị đầu cuối, trên phía máy chủ, hoặc cả hai. Trong một ví dụ, trong việc mã hóa thông thường, xử lý được thực hiện trong hai vòng. Vòng thứ nhất là để dò tìm ảnh phức tạp như thế nào dựa trên cơ sở theo từng khung hoặc từng cảnh, hoặc dò tìm tải mã hóa. Vòng thứ hai là cho việc xử lý mà duy trì chất lượng ảnh và cải thiện hiệu quả mã hóa. Ví dụ, có thể làm giảm tải xử lý của các thiết bị đầu cuối và cải thiện chất lượng và hiệu quả mã hóa của nội dung bằng cách bằng cách để cho các thiết bị đầu cuối thực hiện vòng thứ nhất để mã hóa và để cho phía máy chủ mà đã thu nội dung thực hiện vòng thứ hai để mã hóa. Trong trường hợp này, sau khi thu yêu cầu giải mã, có khả năng để cho dữ liệu được mã hóa thu được từ vòng thứ nhất được thực hiện bởi một thiết bị đầu cuối cần được thu và được tái tạo trên thiết bị đầu cuối khác trong xấp xỉ thời gian thực. Điều này làm cho có thể thực hiện việc tạo dòng theo thời gian thực, mượt mà.

Trong ví dụ khác, camera ex113 hoặc loại tương tự tách lượng đặc tính (lượng đặc tính hoặc đặc trưng) từ ảnh, nén dữ liệu liên quan đến lượng đặc tính như là siêu dữ liệu, và truyền siêu dữ liệu được nén tới máy chủ. Ví dụ, máy chủ xác định sự quan trọng của đối tượng dựa trên lượng đặc tính và thay đổi độ chính xác lượng tử hóa một cách tương ứng để thực hiện việc nén thích hợp đối với ý nghĩa (hoặc sự quan trọng nội dung) của ảnh. Dữ liệu lượng đặc tính là đặc biệt hiệu quả trong việc cải thiện độ chính xác và hiệu quả của việc dự đoán vector chuyển động trong việc thông qua nén thứ hai được thực hiện bởi máy chủ. Ngoài ra, việc mã hóa mà có tải xử lý tương đối thấp, như mã hóa độ dài biến thiên (VLC-variable length coding), có thể được xử lý bởi thiết bị đầu cuối, và việc mã hóa mà có tải xử lý tương đối cao, như mã hóa số học nhị phân thích nghi ngữ cảnh (CABAC-context-adaptive binary arithmetic coding), có thể được xử lý bởi máy chủ.

Trong ví dụ khác, có các trường hợp trong đó các video của gần như cùng cảnh hình được thu bởi các thiết bị đầu cuối trong, ví dụ, sân vận động, trung tâm mua sắm, hoặc nhà máy. Trong trường hợp này, ví dụ, việc mã hóa có thể được phi tập trung hóa bằng cách phân chia các tác vụ xử lý giữa các thiết bị đầu cuối mà thu các video và, nếu cần thiết, các thiết bị đầu cuối khác mà không thu các video, và máy chủ, trên trên cơ sở đơn vị. Các đơn vị có thể là, ví dụ, các nhóm của các ảnh (GOP-groups of pictures), các ảnh, hoặc các lát thu được từ việc phân chia ảnh. Điều này làm cho có thể làm giảm thời gian tải và thực hiện việc tạo dạng mà gần hơn với thời gian thực.

Do các video là của gần như cùng cảnh hình, việc quản lý và/hoặc các lệnh có thể được thực hiện bởi máy chủ sao cho các video được thu bởi các thiết bị đầu cuối có thể được tham chiếu chéo. Ngoài ra, máy chủ có thể thu dữ liệu được mã hóa từ các thiết bị đầu cuối, thay đổi quan hệ tham chiếu giữa các mục dữ liệu, hoặc hiệu chỉnh hoặc thay thế bản thân các ảnh, và sau đó thực hiện việc mã hóa. Điều này làm cho có thể tạo ra dòng có hiệu quả và chất lượng được tăng lên đối với các mục dữ liệu riêng biệt.

Ngoài ra, máy chủ có thể tạo dòng dữ liệu video sau khi thực hiện việc chuyển mã để chuyển đổi khuôn dạng mã hóa của dữ liệu video. Ví dụ, máy chủ có thể chuyển đổi khuôn dạng mã hóa từ MPEG thành VP (ví dụ, VP9), có thể chuyển đổi H.264 thành H.265, v.v.

Theo cách này, việc mã hóa có thể được thực hiện bởi thiết bị đầu cuối hoặc một hoặc nhiều máy chủ. Do đó, mặc dù thiết bị mà thực hiện việc mã hóa được gọi là "máy chủ" hoặc "thiết bị đầu cuối" trong phần mô tả sau đây, một vài hoặc tất cả xử lý được thực hiện bởi máy chủ có thể được thực hiện bởi thiết bị đầu cuối, và tương tự một vài hoặc tất cả xử lý được thực hiện bởi thiết bị đầu cuối có thể được thực hiện bởi máy chủ. Điều này cũng áp dụng tới các xử lý giải mã.

(3D, Đa góc)

Có sự gia tăng về việc sử dụng các ảnh hoặc video được kết hợp từ các ảnh hoặc video của các cảnh hình khác nhau được thu một cách đồng thời, hoặc của cùng cảnh hình được thu từ các góc khác nhau, bởi các thiết bị đầu cuối như camera ex1 13

và/hoặc điện thoại thông minh ex115. Các video thu được bởi các thiết bị đầu cuối có thể được kết hợp dựa trên, ví dụ, quan hệ vị trí tương đối thu được một cách riêng biệt giữa các thiết bị đầu cuối, hoặc các vùng trong video mà có các điểm đặc tính tương hợp.

Ngoài việc mã hóa của các ảnh động hai chiều, máy chủ có thể mã hóa ảnh tĩnh dựa trên phân tích cảnh hình của ảnh động, ví dụ một cách tự động hoặc tại thời điểm được chỉ rõ bởi người dùng, và truyền ảnh tĩnh được mã hóa tới thiết bị đầu cuối thu. Ngoài ra, khi máy chủ có thể thu nhận quan hệ vị trí tương đối giữa các thiết bị đầu cuối thu video, ngoài các ảnh động hai chiều, máy chủ có thể tạo ra hình ba chiều của cảnh hình dựa trên video của cùng cảnh hình thu được từ các góc khác nhau. Máy chủ có thể mã hóa riêng biệt dữ liệu ba chiều được tạo ra từ, ví dụ, mô hình điểm đám mây và, dựa trên kết quả nhận biết hoặc dõi theo người hoặc đối tượng sử dụng dữ liệu ba chiều, có thể lựa chọn hoặc khôi phục và tạo ra video cần được truyền tới thiết bị đầu cuối thu, từ các video thu được bởi các thiết bị đầu cuối.

Điều này cho phép người dùng thưởng thức cảnh hình bằng cách lựa chọn tự do các video tương ứng với các thiết bị đầu cuối thu video, và cho phép người dùng thưởng thức nội dung thu được bằng cách tách video tại điểm nhìn được lựa chọn từ dữ liệu ba chiều được khôi phục từ các ảnh hoặc video. Ngoài ra, như đối với video, âm thanh có thể được ghi từ các góc khác nhau tương đối, và máy chủ có thể ghép kênh audio từ góc hoặc không gian cụ thể với video tương ứng, và truyền video và audio được ghép kênh.

Trong những năm gần đây, nội dung mà là kết hợp của thế giới thực và thế giới ảo, như nội dung thực tế ảo (VR-virtual reality) và thực tế tăng cường (AR), cũng đã trở nên phổ biến. Trong trường hợp của các ảnh VR, máy chủ có thể tạo ra các ảnh từ các điểm nhìn của cả mắt trái và mắt phải, và thực hiện việc mã hóa mà cho phép sự tham chiếu giữa hai ảnh điểm nhìn, như mã hóa đa cảnh nhìn (MVC-multi-view coding), và, ngoài ra, có thể mã hóa các ảnh như là các dòng riêng biệt mà không có tham chiếu. Khi các ảnh được giải mã như là các dòng riêng biệt, các dòng có thể được đồng bộ hóa khi được tái tạo, để tạo ra lại không gian ba chiều ảo theo điểm nhìn của người dùng.

Trong trường hợp của các ảnh AR, máy chủ có thể xếp chồng thông tin đối tượng ảo tồn tại trong không gian ảo lên thông tin camera mà biểu diễn không thể giới thực, ví dụ dựa trên vị trí ba chiều hoặc chuyển động từ góc nhìn của người dùng. Bộ giải mã có thể thu nhận hoặc lưu trữ thông tin đối tượng ảo và dữ liệu ba chiều, tạo ra các ảnh hai chiều dựa trên chuyển động từ góc nhìn của người dùng, và sau đó tạo ra dữ liệu được xếp chồng bằng cách kết nối liền mạch các ảnh. Ngoài ra, bộ giải mã có thể truyền, tới máy chủ, chuyển động từ góc nhìn của người dùng ngoài yêu cầu đối với thông tin đối tượng ảo. Máy chủ có thể tạo ra dữ liệu được xếp chồng dựa trên dữ liệu ba chiều được lưu trữ trong máy chủ theo chuyển động thu được, và mã hóa và tạo dòng dữ liệu được xếp chồng được tạo ra tới bộ giải mã. Lưu ý rằng dữ liệu được xếp chồng thường bao gồm, ngoài các giá trị RGB, giá trị α mà chỉ báo độ trong suốt, và máy chủ thiết lập giá trị α đối với các phân vùng khác ngoài đối tượng được tạo ra từ dữ liệu ba chiều thành, ví dụ, 0, và có thể thực hiện việc mã hóa trong khi các phân vùng này là trong suốt. Ngoài ra, máy chủ có thể thiết lập nền thành giá trị RGB được xác định, như khóa màu, và tạo ra dữ liệu trong đó các vùng khác ngoài đối tượng được thiết lập như là nền. Giá trị RGB được xác định có thể được xác định trước.

Việc giải mã dữ liệu được tạo dòng tương tự có thể được thực hiện bởi máy khách (ví dụ, các thiết bị đầu cuối), trên phía máy chủ, hoặc được phân chia giữa chúng. Trong một ví dụ, một thiết bị đầu cuối có thể truyền yêu cầu thu tới máy chủ, nội dung được yêu cầu có thể được thu và được giải mã bởi thiết bị đầu cuối khác, và tín hiệu được giải mã có thể được truyền tới thiết bị mà có màn hình. Có thể tái tạo dữ liệu chất lượng ảnh cao bằng xử lý phi tập trung và lựa chọn một cách thích hợp nội dung bất kể khả năng xử lý của bản thân thiết bị đầu cuối truyền thông. Trong ví dụ khác, trong khi TV, ví dụ, đang thu dữ liệu ảnh mà có kích cỡ lớn, vùng của ảnh, như lát thu được bằng cách phân chia ảnh, có thể được giải mã và được hiển thị trên thiết bị đầu cuối cá nhân hoặc các thiết bị đầu cuối của người xem hoặc các người xem TV. Điều này làm cho các người xem có thể chia sẻ cảnh nhìn ảnh lớn cũng như làm cho mỗi người dùng có thể kiểm tra vùng được gán, hoặc kiểm tra vùng một cách chi tiết gần hơn.

Trong các trường hợp trong đó các kết nối không dây là khả dụng trên các khoảng cách gần, trung bình, và xa, trong nhà hoặc ngoài nhà, có thể thu không ngắt quãng nội dung nhờ sử dụng tiêu chuẩn hệ thống tạo dòng như MPEG-DASH. Người dùng có thể chuyển đổi giữa dữ liệu trong thời gian thực trong khi lựa chọn tự do bộ giải mã hoặc thiết bị hiển thị bao gồm thiết bị đầu cuối của người dùng, các màn hình được bố trí trong nhà hoặc ngoài nhà, v.v. Ngoài ra, sử dụng, ví dụ, thông tin về vị trí của người dùng, việc giải mã có thể được thực hiện trong khi chuyển đổi thiết bị đầu cuối nào mà xử lý việc giải mã và thiết bị đầu cuối nào mà xử lý việc hiển thị nội dung. Điều này làm cho có thể ánh xạ và hiển thị thông tin, trong khi người dùng đang di chuyển theo tuyến tới đích, trên tường của tòa nhà ở gần trong đó thiết bị có khả năng hiển thị nội dung được gắn vào, hoặc trên một phần của mặt đất. Ngoài ra, cũng có thể chuyển đổi tốc độ bit của dữ liệu thu được dựa trên khả năng truy nhập tới dữ liệu được mã hóa trên mạng, như khi dữ liệu được mã hóa được lưu trữ đệm trên máy chủ có thể truy nhập nhanh chóng từ thiết bị đầu cuối thu, hoặc khi dữ liệu được mã hóa được sao chép tới máy chủ biên trong dịch vụ phân phát nội dung.

(Tối ưu hóa trang mạng)

FIG.175 minh họa ví dụ về màn hiển thị của trang mạng trên máy tính ex111, chẳng hạn. FIG.176 minh họa ví dụ về màn hiển thị của trang mạng trên điện thoại thông minh ex115, chẳng hạn. Như được minh họa trong FIG.175 và FIG.176, trang mạng có thể bao gồm nhiều liên kết ảnh mà là các liên kết tới nội dung ảnh, và sự xuất hiện của trang mạng có thể khác nhau phụ thuộc vào thiết bị được sử dụng để xem trang mạng. Khi các liên kết ảnh có thể xem được trên màn ảnh, cho đến khi người dùng lựa chọn rõ ràng liên kết ảnh, hoặc cho đến khi liên kết ảnh nằm trong xấp xỉ trung tâm của màn ảnh hoặc toàn bộ liên kết ảnh vừa vắn trong màn ảnh, thiết bị hiển thị (bộ giải mã) có thể hiển thị, như là các liên kết ảnh, các ảnh tĩnh được chứa trong nội dung hoặc các ảnh I; có thể hiển thị video như gif động nhờ sử dụng các ảnh tĩnh hoặc các ảnh I; hoặc có thể thu chỉ lớp cơ sở, và giải mã và hiển thị video.

Khi liên kết ảnh được lựa chọn bởi người dùng, thiết bị hiển thị thực hiện việc giải mã trong khi, ví dụ gán mức ưu tiên cao nhất tới lớp cơ sở. Lưu ý rằng nếu

có thông tin trong mã HTML của trang mạng mà chỉ báo rằng nội dung có thể mở rộng, thiết bị hiển thị có thể giải mã tới đa tới lớp nâng cao. Ngoài ra, để hỗ trợ sự tái tạo thời gian thực, trước khi việc lựa chọn được thực hiện hoặc khi băng thông bị giới hạn nghiêm ngặt, thiết bị hiển thị có thể làm giảm độ trễ giữa thời điểm mà tại đó ảnh đầu được giải mã và thời điểm mà tại đó ảnh được giải mã được hiển thị (tức là, độ trễ giữa lúc bắt đầu của việc giải mã nội dung để lúc hiển thị nội dung) bằng cách giải mã và hiển thị chỉ các ảnh tham chiếu chuyển tiếp (ảnh I, ảnh P, ảnh tham chiếu chuyển tiếp B). Ngoài ra, thiết bị hiển thị có thể bỏ qua có mục đích quan hệ tham chiếu giữa các ảnh, và giải mã thô tất cả ảnh B và P như là các ảnh tham chiếu chuyển tiếp, và sau đó thực hiện việc giải mã bình thường khi số lượng ảnh thu được theo thời gian tăng lên.

(Lái xe tự trị)

Khi truyền và thu ảnh tĩnh hoặc dữ liệu video như thông tin bản đồ hai chiều hoặc ba chiều để lái xe tự trị hoặc lái xe hỗ trợ của xe ô tô, thiết bị đầu cuối thu có thể thu, ngoài dữ liệu ảnh thuộc về một hoặc nhiều lớp, thông tin về, ví dụ, thời tiết hoặc công trình đường xá như là siêu dữ liệu, và kết hợp siêu dữ liệu với dữ liệu ảnh sau khi giải mã. Lưu ý rằng siêu dữ liệu có thể được gán trên lớp và, ngoài ra, có thể được ghép kênh một cách đơn giản với dữ liệu ảnh.

Trong trường hợp này, do ô tô, thiết bị bay không người lái, máy bay, v.v, chứa thiết bị đầu cuối thu là di động, thiết bị đầu cuối thu có thể thu không gián đoạn và thực hiện việc giải mã trong khi chuyển đổi giữa các trạm gốc trong số các trạm gốc ex106 đến ex110 bằng cách truyền thông tin mà chỉ báo vị trí của thiết bị đầu cuối thu. Ngoài ra, theo việc lựa chọn được thực hiện bởi người dùng, tình huống của người dùng, và/hoặc băng thông của kết nối, thiết bị đầu cuối thu có thể lựa chọn động ở mức nào siêu dữ liệu được thu nhận, hoặc ở mức độ nào thông tin bản đồ, ví dụ, được cập nhật.

Trong hệ thống cung cấp nội dung ex100, máy khách có thể thu, giải mã, và tái tạo, trong thời gian thực, thông tin được mã hóa được truyền bởi người dùng.

(Tạo dòng nội dung riêng biệt)

Trong hệ thống cung cấp nội dung ex100, ngoài chất lượng ảnh cao, nội dung dài được phân phối bởi thực thể phân phối video, việc tạo dòng đơn hướng hoặc đa hướng của chất lượng ảnh thấp, và nội dung ngắn từ cá nhân cũng khả dụng. Nội dung từ các cá nhân này có thể tăng phổ biến hơn nữa. Máy chủ có thể đầu tiên thực hiện xử lý chỉnh sửa về nội dung trước xử lý mã hóa, để tinh chỉnh nội dung cá nhân. Điều này có thể đạt được nhờ sử dụng cấu trúc sau đây, chẳng hạn.

Trong thời gian thực trong khi thu nội dung video hoặc ảnh, hoặc sau khi nội dung được thu và được tích lũy, máy chủ thực hiện xử lý nhận biết dựa trên dữ liệu thô hoặc dữ liệu được mã hóa, như xử lý lỗi thu, xử lý tìm kiếm cảnh hình, phân tích ý nghĩa, và/hoặc xử lý dò tìm đối tượng. Sau đó, dựa trên kết quả của xử lý nhận biết, máy chủ - ví dụ khi được gọi ý hoặc một cách tự động - biên tập nội dung, các ví dụ của việc này bao gồm: hiệu chỉnh như tập trung và/hoặc hiệu chỉnh độ mờ chuyển động; loại bỏ các cảnh hình có mức ưu tiên thấp như các cảnh hình mà có độ sáng thấp so với các ảnh khác, hoặc nằm ngoài dải tập trung; điều chỉnh biên đối tượng; và điều chỉnh tông màu. Máy chủ mã hóa dữ liệu được biên tập dựa trên kết quả biên tập. Đã được biết rằng các video quá dài có xu hướng thu hút ít người xem hơn. Do đó, để giữ nội dung trong thời gian cụ thể mà tỷ lệ với độ dài của video gốc, máy chủ có thể, ngoài các cảnh hình có mức ưu tiên thấp được mô tả nêu trên, tự động cắt các cảnh hình có sự di chuyển thấp, dựa trên kết quả xử lý ảnh. Ngoài ra, máy chủ có thể tạo ra và mã hóa thông báo video dựa trên kết quả phân tích ý nghĩa của cảnh hình.

Có thể có các trường hợp trong đó nội dung cá nhân có thể bao gồm nội dung mà xâm phạm quyền tác giả, quyền tinh thần, các quyền hình tượng, v.v. Trường hợp này có thể dẫn đến trường hợp bất lợi cho người sáng tạo, như khi nội dung được chia sẻ ngoài phạm vi được dự định bởi người sáng tạo. Do đó, trước khi mã hóa, máy chủ có thể, ví dụ, biên tập các ảnh để làm mờ các khuôn mặt của người trong biên ngoài của màn ảnh hoặc làm mờ phần bên trong của ngôi nhà, chẳng hạn. Ngoài ra, máy chủ có thể có cấu trúc để nhận biết các khuôn mặt của người ngoài người được đăng ký trong các ảnh cần được mã hóa, và khi các khuôn mặt này xuất hiện trong ảnh, có thể áp dụng bộ lọc khảm, ví dụ, tối khuôn mặt của người. Ngoài ra, theo xử lý trước hoặc sau để mã hóa, người dùng có thể chỉ rõ, đối với các lý do

quyền tác giả, vùng của ảnh bao gồm người hoặc vùng của nền cần được xử lý. Máy chủ có thể xử lý vùng được chỉ rõ bằng cách, ví dụ, thay thế vùng này bằng ảnh khác, hoặc làm mờ vùng này. Nếu vùng này bao gồm người, người này có thể được đổi theo trong ảnh động, và vùng đầu của người có thể được thay thế bằng ảnh khác khi người này di chuyển.

Do có nhu cầu đối với việc xem thời gian thực của nội dung được sản xuất bởi các cá nhân, mà có xu hướng có kích cỡ dữ liệu nhỏ, bộ giải mã có thể đầu tiên thu lớp cơ sở như là mức ưu tiên cao nhất, và thực hiện việc giải mã và tái tạo, mặc dù điều này có thể khác nhau phụ thuộc vào băng thông. Khi nội dung được tái tạo hai lần hoặc nhiều hơn, như khi bộ giải mã thu lớp nâng cao trong khi giải mã và tái tạo của lớp gốc, và tạo vòng việc tái tạo, bộ giải mã có thể tái tạo video chất lượng ảnh cao bao gồm lớp nâng cao. Nếu dòng được mã hóa nhờ sử dụng việc mã hóa có thể mở rộng này, video có thể có chất lượng thấp khi trong trạng thái không được lựa chọn hoặc tại lúc bắt đầu video, nhưng có thể mang lại trải nghiệm trong đó chất lượng ảnh của dòng tăng lên dần theo cách thức thông minh. Điều này không bị giới hạn bởi chỉ việc mã hóa có thể mở rộng; trải nghiệm tương tự có thể được đề xuất bằng cách cấu hình một dòng từ dòng chất lượng thấp được tái tạo trong lần đầu tiên và dòng thứ hai được mã hóa nhờ sử dụng dòng thứ nhất như là tham chiếu.

(Cách thức thực hiện khác và các ví dụ ứng dụng)

Việc mã hóa và giải mã có thể được thực hiện bởi LSI (large scale integration circuitry-Mạch tích hợp cỡ lớn) ex500 (xem FIG.174), mà thường được chứa trong mỗi thiết bị đầu cuối. LSI ex500 có thể được cấu thành từ một chip hoặc nhiều chip. Phần mềm để mã hóa và giải mã các ảnh động có thể được tích hợp vào một vài loại vật ghi (như CD-ROM, đĩa mềm, hoặc đĩa cứng) mà có thể đọc được bởi, ví dụ, máy tính ex111, và việc mã hóa và giải mã có thể được thực hiện nhờ sử dụng phần mềm. Ngoài ra, khi điện thoại thông minh ex115 được trang bị camera, dữ liệu video thu được bởi camera có thể được truyền. Trong trường hợp này, dữ liệu video có thể được mã hóa bởi LSI ex500 được chứa trong điện thoại thông minh ex115.

Lưu ý rằng LSI ex500 có thể được cấu hình để tải xuống và kích hoạt ứng dụng. Trong trường hợp này, thiết bị đầu cuối đầu tiên xác định rằng có tương thích

với phương pháp được sử dụng để mã hóa nội dung hay không, hoặc có thể thực hiện dịch vụ cụ thể hay không. Khi thiết bị đầu cuối không tương thích với phương pháp mã hóa của nội dung, hoặc khi thiết bị đầu cuối không có khả năng thực hiện dịch vụ cụ thể, thiết bị đầu cuối có thể đầu tiên tải xuống phần mềm mã hóa-giải mã hoặc ứng dụng và sau đó thu nhận và tái tạo nội dung.

Ngoài ví dụ về hệ thống cung cấp nội dung ex100 mà sử dụng internet ex101, ít nhất bộ mã hóa ảnh động (bộ mã hóa ảnh) hoặc bộ giải mã ảnh động (bộ giải mã ảnh) được mô tả trong các phương án nêu trên có thể được thực hiện trong hệ thống quảng bá số. Xử lý mã hóa và xử lý giải mã giống nhau có thể được áp dụng để truyền và thu các sóng vô tuyến quảng bá được xếp chồng với dữ liệu audio và video được ghép kênh nhờ sử dụng, ví dụ, vệ tinh, mặc dù điều này hướng tới phát đa hướng, trong khi phát đơn hướng là dễ dàng hơn đối với hệ thống cung cấp nội dung ex100.

(Cấu trúc phần cứng)

FIG.177 minh họa các chi tiết khác của điện thoại thông minh ex115 ví dụ được thể hiện trên FIG.174. FIG.178 minh họa ví dụ cấu trúc chức năng của điện thoại thông minh ex115. Điện thoại thông minh ex115 bao gồm anten ex450 để truyền và thu các sóng vô tuyến tới và từ trạm gốc ex110, camera ex465 có khả năng thu video và các ảnh tĩnh, và màn hình ex458 mà hiển thị dữ liệu được giải mã, như video được thu bởi camera ex465 và video được thu bởi anten ex450. Điện thoại thông minh ex115 còn bao gồm giao diện người dùng ex466 như panel chạm, bộ xuất audio ex457 như loa để xuất lời nói hoặc audio khác, bộ nhập audio ex456 như microphôn để nhập audio, bộ nhớ ex467 có khả năng lưu trữ dữ liệu được giải mã như video hoặc các ảnh tĩnh được chụp, audio được ghi, video hoặc các ảnh tĩnh được thu nhận, và thư, cũng như dữ liệu được giải mã, và khe ex464 mà là giao diện cho SIM ex468 để ủy quyền truy nhập tới mạng và dữ liệu khác nhau. Lưu ý rằng bộ nhớ ngoài có thể được sử dụng thay vì hoặc ngoài bộ nhớ ex467.

Bộ điều khiển chính ex460, mà có thể điều khiển toàn diện màn hình ex458 và giao diện người dùng ex466, mạch cấp điện ex461, bộ điều khiển đầu vào giao diện người dùng ex462, bộ xử lý tín hiệu video ex455, giao diện camera ex463, bộ

điều khiển hiển thị ex459, bộ điều chế/bộ giải điều chế ex452, bộ ghép kênh/bộ giải ghép kênh ex453, bộ xử lý tín hiệu audio ex454, khe ex464, và bộ nhớ ex467 được kết nối thông qua kênh truyền ex470.

Khi người dùng bật nút nguồn của mạch cấp điện ex461, điện thoại thông minh ex115 được bật thành trạng thái có thể hoạt động, và mỗi thành phần được cấp điện, ví dụ, từ bộ pin.

Điện thoại thông minh ex115 thực hiện xử lý để, ví dụ, gọi và truyền dữ liệu, dựa trên điều khiển được thực hiện bởi bộ điều khiển chính ex460, mà bao gồm CPU, ROM, và RAM. Khi thực hiện cuộc gọi, tín hiệu audio được ghi bởi bộ nhập audio ex456 được chuyển đổi thành tín hiệu audio số bởi bộ xử lý tín hiệu audio ex454, mà xử lý trải phổ được áp dụng tới bởi bộ điều chế/bộ giải điều chế ex452 và chuyển đổi số-tương tự, và xử lý chuyển đổi tần số được áp dụng bởi bộ truyền/bộ thu ex451, và tín hiệu cuối cùng được truyền thông qua antenna ex450. Dữ liệu thu được được khuếch đại, được chuyển đổi tần số, và được chuyển đổi tương tự-số, được xử lý trải phổ ngược bởi bộ điều chế/bộ giải điều chế ex452, được chuyển đổi thành tín hiệu audio tương tự bởi bộ xử lý tín hiệu audio ex454, và sau đó được xuất ra từ bộ xuất audio ex457.

Trong chế độ truyền dữ liệu, văn bản, ảnh tĩnh, hoặc dữ liệu video có thể được truyền dưới sự điều khiển của bộ điều khiển chính ex460 thông qua bộ điều khiển đầu vào giao diện người dùng ex462 dựa trên hoạt động của giao diện người dùng ex466 của khối chính, chẳng hạn. Xử lý truyền và thu tương tự được thực hiện. Trong chế độ truyền dữ liệu, khi gửi video, ảnh tĩnh, hoặc video và audio, bộ xử lý tín hiệu video ex455 mã hóa nén, thông qua phương pháp mã hóa ảnh động được mô tả trong các phương án nêu trên, tín hiệu video được lưu trữ trong bộ nhớ ex467 hoặc tín hiệu video được đưa vào từ camera ex465, và truyền dữ liệu video được mã hóa tới bộ ghép kênh/bộ giải ghép kênh ex453. Bộ xử lý tín hiệu audio ex454 mã hóa tín hiệu audio được ghi bởi bộ nhập audio ex456 trong khi camera ex465 đang thu video hoặc ảnh tĩnh, và truyền dữ liệu audio được mã hóa tới bộ ghép kênh/bộ giải ghép kênh ex453. Bộ ghép kênh/bộ giải ghép kênh ex453 ghép kênh dữ liệu video được mã hóa và dữ liệu audio được mã hóa nhờ sử dụng phương pháp

được xác định, điều chế và chuyển đổi dữ liệu nhờ sử dụng bộ điều chế/bộ giải điều chế (mạch điều chế/mạch giải điều chế) ex452 và bộ truyền/bộ thu ex451, và truyền kết quả thông qua anten ex450. Phương pháp được xác định có thể được xác định trước.

Khi video được kèm trong thư điện tử hoặc cuộc nói chuyện, hoặc video được liên kết từ trang mạng, được thu nhận, ví dụ, để giải mã dữ liệu được ghép kênh thu được thông qua anten ex450, bộ ghép kênh/bộ giải ghép kênh ex453 giải ghép kênh dữ liệu được ghép kênh để phân chia dữ liệu được ghép kênh thành dòng bit của dữ liệu video và dòng bit của dữ liệu audio, cấp dữ liệu video được mã hóa tới bộ xử lý tín hiệu video ex455 thông qua kênh truyền đồng bộ ex470, và cấp dữ liệu audio được mã hóa tới bộ xử lý tín hiệu audio ex454 thông qua kênh truyền đồng bộ ex470. Bộ xử lý tín hiệu video ex455 giải mã tín hiệu video nhờ sử dụng phương pháp giải mã ảnh động tương ứng với phương pháp mã hóa ảnh động được mô tả trong các phương án nêu trên, và video hoặc ảnh tĩnh được chứa trong tệp ảnh động được liên kết được hiển thị trên màn hình ex458 thông qua bộ điều khiển hiển thị ex459. Bộ xử lý tín hiệu audio ex454 giải mã tín hiệu audio và xuất audio từ bộ xuất audio ex457. Do việc tạo dòng thời gian thực đang trở nên phổ biến hơn, có thể có các trường hợp trong đó việc tái tạo audio có thể không thích hợp về mặt xã hội, phụ thuộc vào môi trường của người dùng. Do đó, đối với giá trị khởi tạo, cấu hình trong đó chỉ dữ liệu video được tái tạo, ví dụ, tín hiệu audio không được tái tạo, có thể được ưu tiên; audio có thể được đồng bộ và được tái tạo chỉ khi đầu vào, như khi người dùng nhấn dữ liệu video, được thu nhận.

Mặc dù điện thoại thông minh ex115 được sử dụng trong ví dụ nêu trên, các cách thức thực hiện khác có thể được đề xuất: thiết bị đầu cuối thu phát bao gồm cả bộ mã hóa và bộ giải mã; thiết bị đầu cuối truyền bao gồm chỉ bộ mã hóa; và thiết bị đầu cuối thu bao gồm chỉ bộ giải mã. Trong phần mô tả của hệ thống quảng bá số, ví dụ được đưa ra trong đó dữ liệu được ghép kênh thu được như là kết quả của dữ liệu video được ghép kênh với dữ liệu audio được thu hoặc được truyền. Tuy nhiên, dữ liệu được ghép kênh, có thể là dữ liệu video được ghép kênh với dữ liệu khác ngoài dữ liệu audio, như dữ liệu văn bản liên quan đến video. Ngoài ra, bản thân dữ liệu video ngoài dữ liệu được ghép kênh có thể được thu hoặc được truyền.

Mặc dù bộ điều khiển chính ex460 bao gồm CPU được mô tả là điều khiển các xử lý mã hóa hoặc giải mã, các thiết bị đầu cuối khác nhau thường bao gồm các bộ xử lý đồ họa (GPU-graphics processing unit). Do đó, cấu hình có thể được chấp nhận trong đó vùng lớn được xử lý ngay lập tức bằng cách sử dụng khả năng thực hiện của GPU thông qua bộ nhớ được chia sẻ bởi CPU và GPU, hoặc bộ nhớ bao gồm địa chỉ mà được quản lý để cho phép sử dụng chung bởi CPU và GPU hoặc thông qua các bộ nhớ riêng biệt. Điều này làm cho có thể rút ngắn thời gian mã hóa, duy trì bản chất thời gian thực của dòng, và làm giảm độ trễ. Cụ thể, xử lý liên quan đến việc đánh giá chuyển động, lọc giải khối, dịch chuyển thích nghi mẫu (SAO-sample adaptive offset), và biến đổi/lượng tử hóa có thể được thực hiện hiệu quả bởi GPU thay vì CPU trong các đơn vị của các ảnh, ví dụ, tất cả trong một.

YÊU CẦU BẢO HỘ

1. Bộ mã hóa, bao gồm:

mạch; và

bộ nhớ được ghép nối tới mạch;

trong đó mạch, khi hoạt động:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng bộ lọc CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới các mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và

mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba:

trong đó trong bước áp dụng bộ lọc CCALF,

để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của các mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

2. Bộ mã hóa theo điểm 1, trong đó:

bộ lọc CCALF là bộ lọc 8 nhánh mà có dạng kim cương 3×4 .

3. Bộ mã hóa theo điểm 1, trong đó:

giá trị hệ số thứ nhất được tạo ra bằng cách nhân giá trị hệ số lọc của bộ lọc CCALF với kết quả của phép trừ của giá trị của một trong số các mẫu ảnh được khôi phục thứ nhất $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$ và giá trị của mẫu ảnh được khôi phục thứ nhất nằm tại $(2x, 2y)$.

4. Thiết bị mã hóa, bao gồm:

bộ phân chia khối, mà khi hoạt động, phân chia ảnh thứ nhất thành nhiều khối;

bộ nội dự đoán, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ nhất;

bộ dự đoán liên đới, mà khi hoạt động, dự đoán các khối được chứa trong ảnh thứ nhất, nhờ sử dụng các khối tham chiếu được chứa trong ảnh thứ hai khác với ảnh thứ nhất;

bộ lọc vòng, mà khi hoạt động, lọc các khối được chứa trong ảnh thứ nhất;

bộ biến đổi, mà khi hoạt động, biến đổi sai số dự đoán giữa tín hiệu gốc và tín hiệu dự đoán được tạo ra bởi bộ nội dự đoán hoặc bộ dự đoán liên đới, để tạo ra các hệ số biến đổi;

bộ lượng tử hóa, mà khi hoạt động, lượng tử hóa các hệ số biến đổi để tạo ra các hệ số được lượng tử hóa; và

bộ mã hóa entropy, mà khi hoạt động, mã hóa biến thiên các hệ số được lượng tử hóa để tạo ra dòng bit được mã hóa bao gồm các hệ số được lượng tử hóa và thông tin điều khiển được mã hóa,

trong đó bộ lọc vòng thực hiện:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng bộ lọc CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới các mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và

mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba:

trong đó trong bước áp dụng bộ lọc CCALF,

để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của các mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

5. Thiết bị mã hóa theo điểm 4, trong đó,

bộ lọc CCALF là bộ lọc 8 nhánh mà có dạng kim cương 3×4 .

6. Thiết bị mã hóa theo điểm 4, trong đó,

giá trị hệ số thứ nhất được tạo ra bằng cách nhân giá trị hệ số lọc của bộ lọc CCALF với kết quả của phép trừ của giá trị của một trong số các mẫu ảnh được khôi phục thứ nhất $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$ và giá trị của mẫu ảnh được khôi phục thứ nhất nằm tại $(2x, 2y)$.

7. Phương pháp mã hóa, bao gồm:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng bộ lọc CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới các mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và

mã hóa mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba:

trong đó trong bước áp dụng bộ lọc CCALF,

để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của các mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

8. Phương pháp giải mã, bao gồm:

tạo ra giá trị hệ số thứ nhất bằng cách áp dụng bộ lọc CCALF (cross component adaptive loop filtering-Lọc vòng thích nghi thành phần chéo) tới các mẫu ảnh được khôi phục thứ nhất của thành phần độ chói;

tạo ra giá trị hệ số thứ hai bằng cách áp dụng xử lý ALF (adaptive loop filtering-Lọc vòng thích nghi) tới mẫu ảnh được khôi phục thứ hai của thành phần sắc độ;

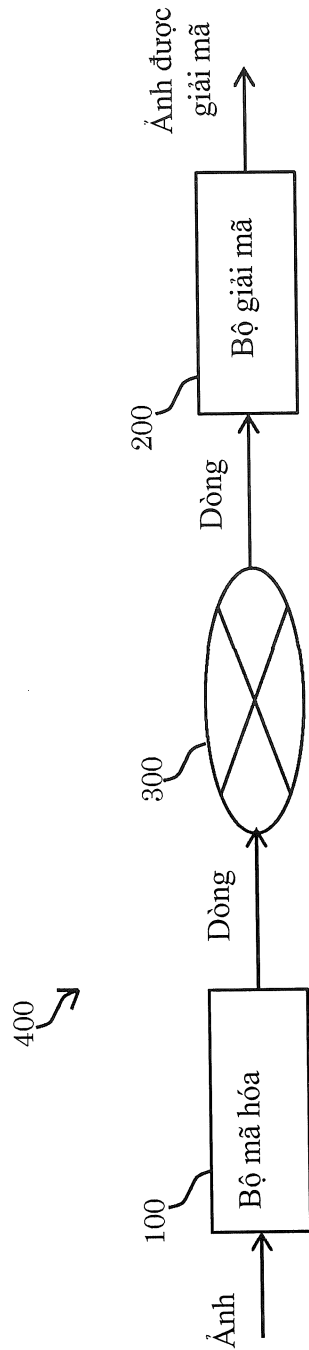
tạo ra giá trị hệ số thứ ba bằng cách cộng giá trị hệ số thứ nhất tới giá trị hệ số thứ hai, và

giải mã mẫu ảnh được khôi phục thứ ba của thành phần sắc độ nhờ sử dụng giá trị hệ số thứ ba:

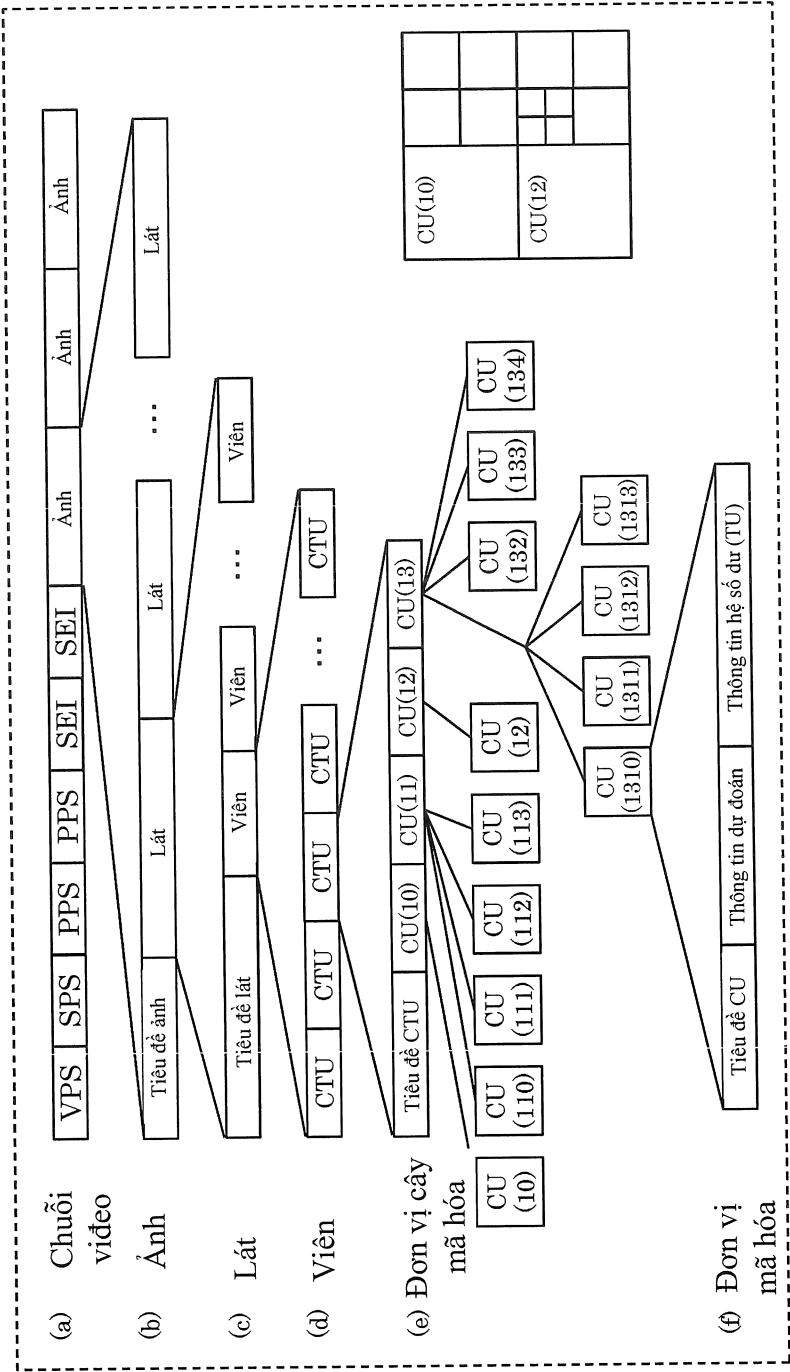
trong đó trong bước áp dụng bộ lọc CCALF,

để phản hồi lại tọa độ của mẫu ảnh được khôi phục thứ hai là (x, y) , các tọa độ của các mẫu ảnh được khôi phục thứ nhất là $(2x, 2y-1)$, $(2x-1, 2y)$, $(2x, 2y)$, $(2x+1, 2y)$, $(2x-1, 2y+1)$, $(2x, 2y+1)$, $(2x+1, 2y+1)$, và $(2x, 2y+2)$.

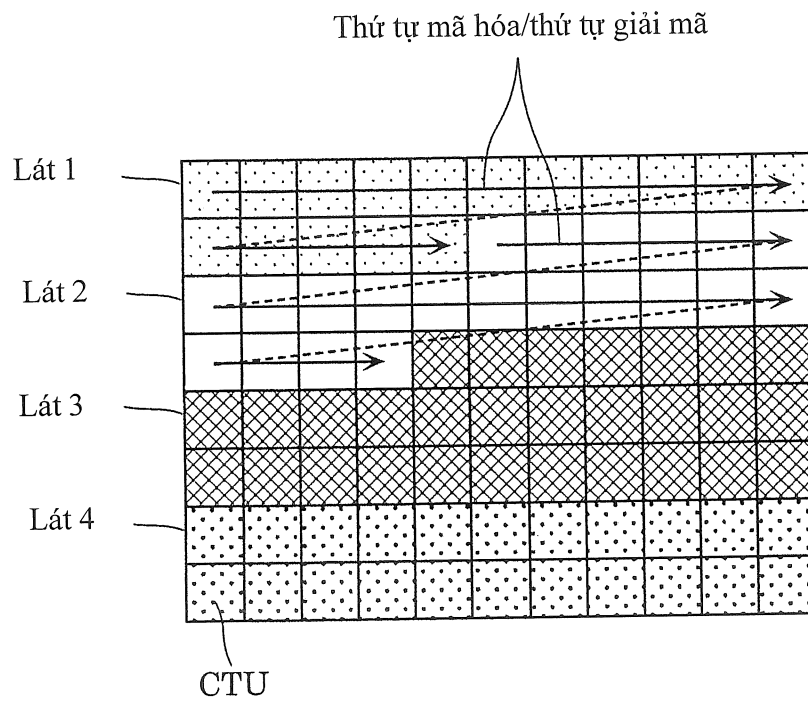
[Fig. 1]



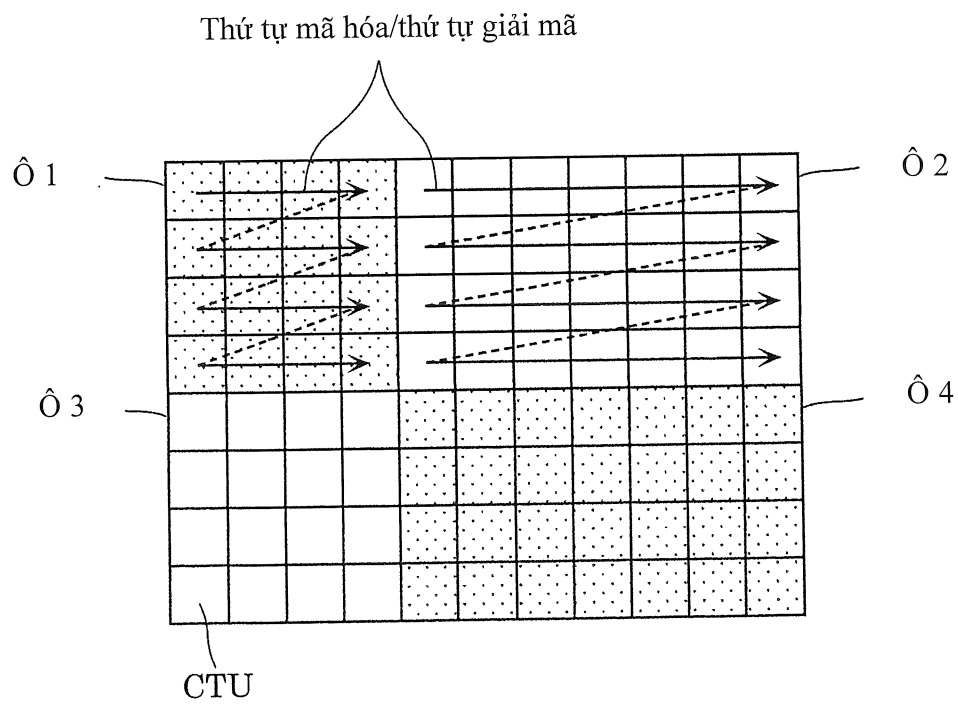
[Fig. 2]



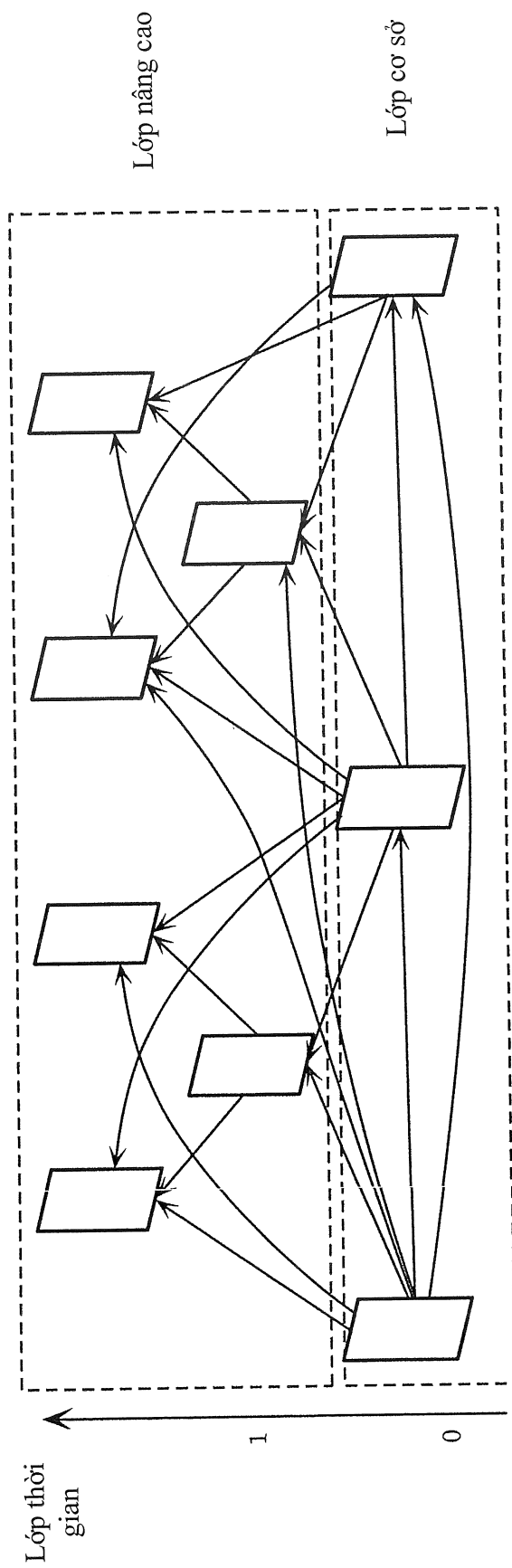
[Fig. 3]



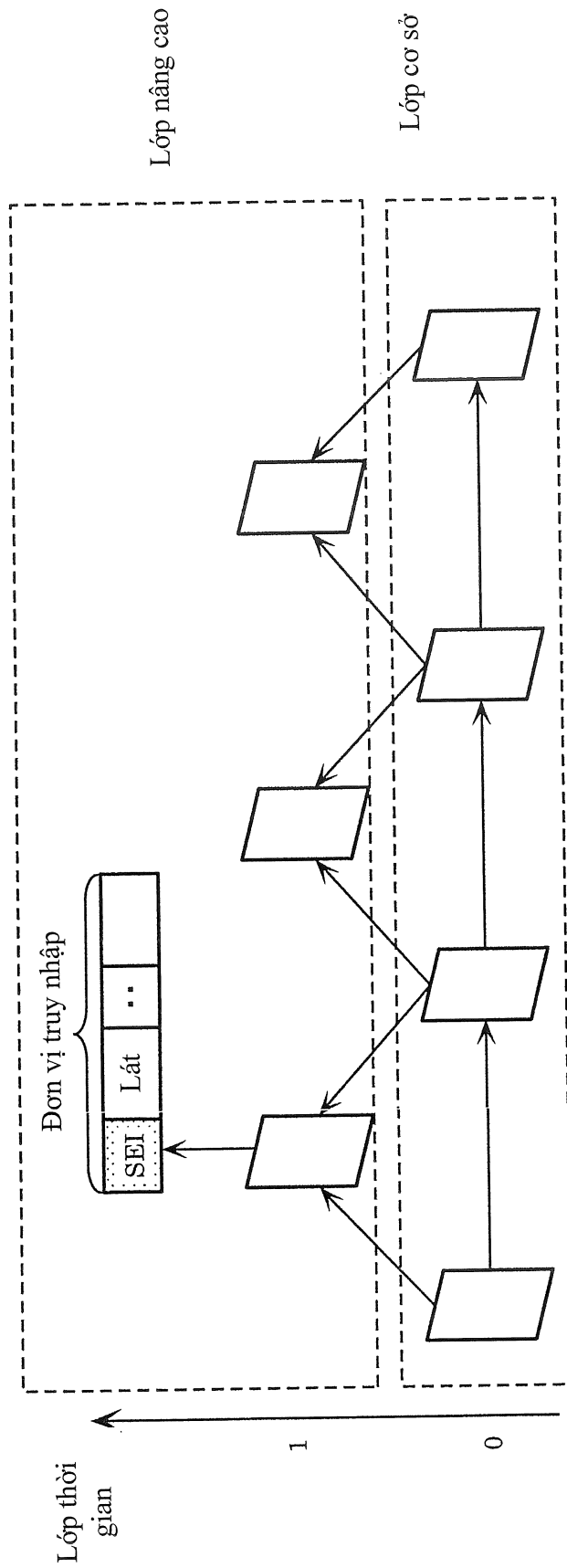
[Fig. 4]



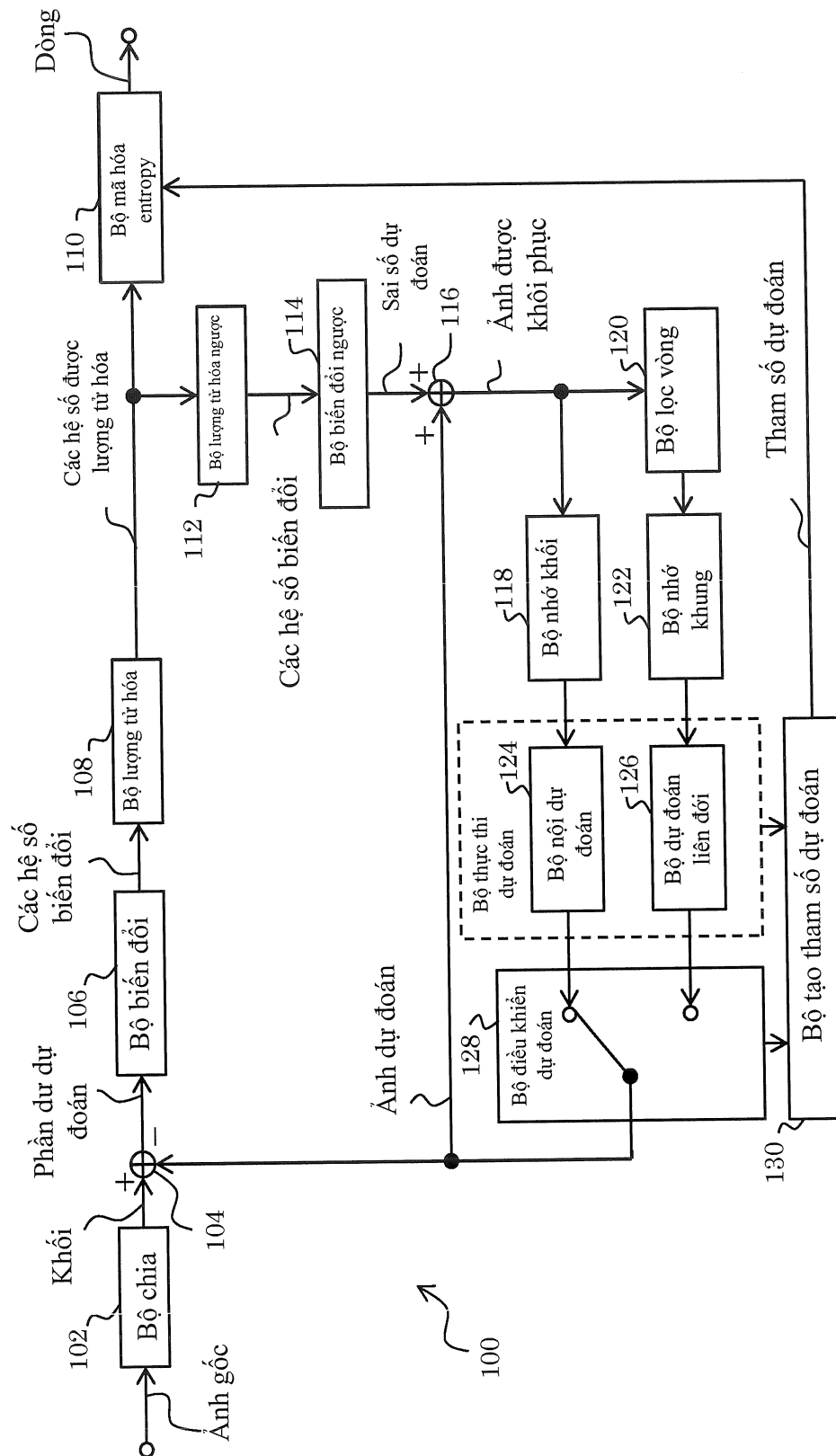
[Fig. 5]



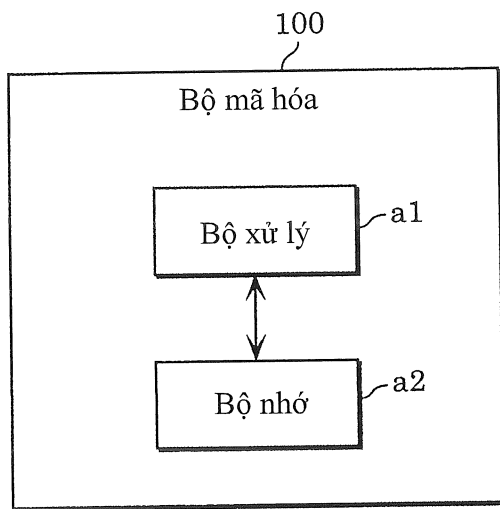
[Fig. 6]



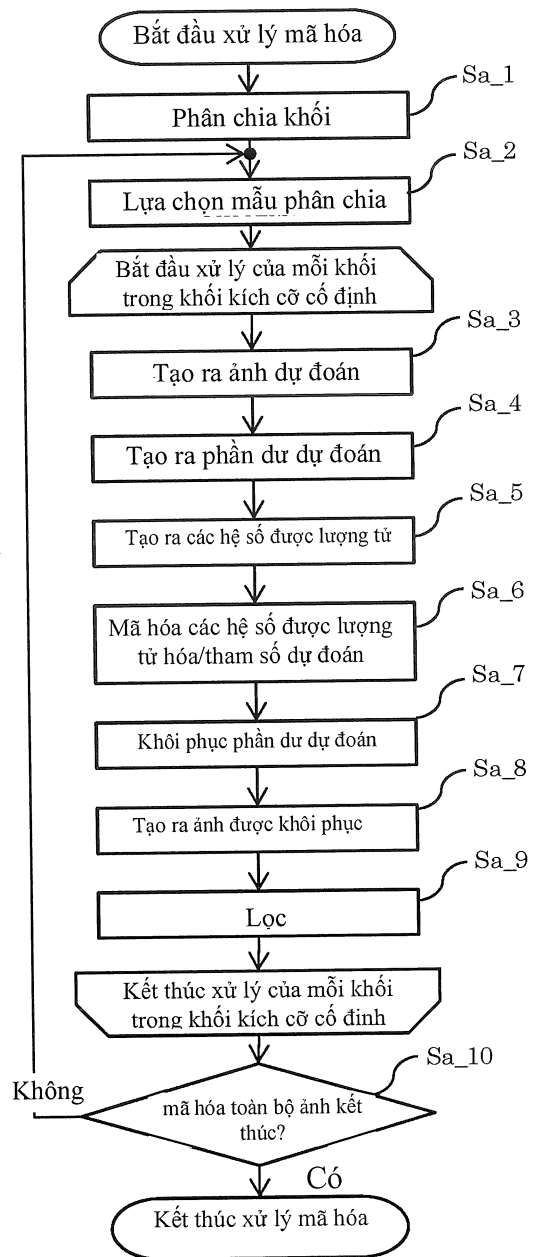
[Fig. 7]



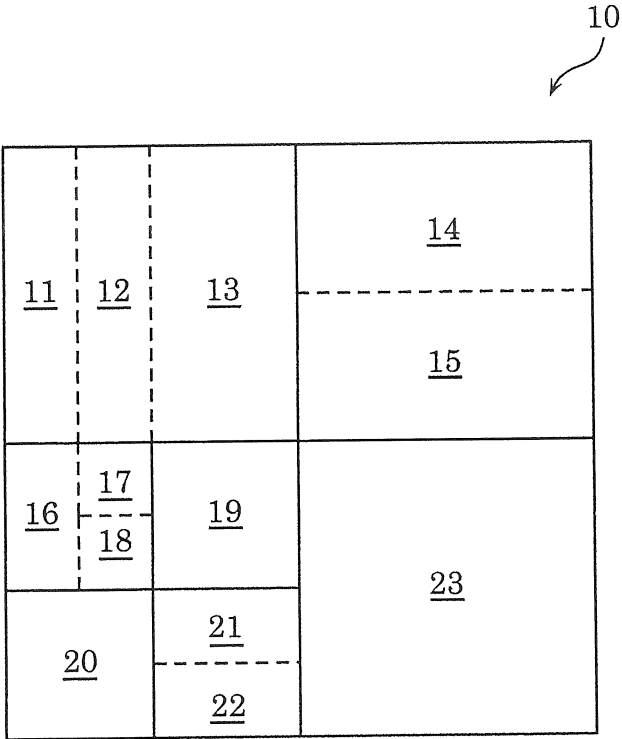
[Fig. 8]



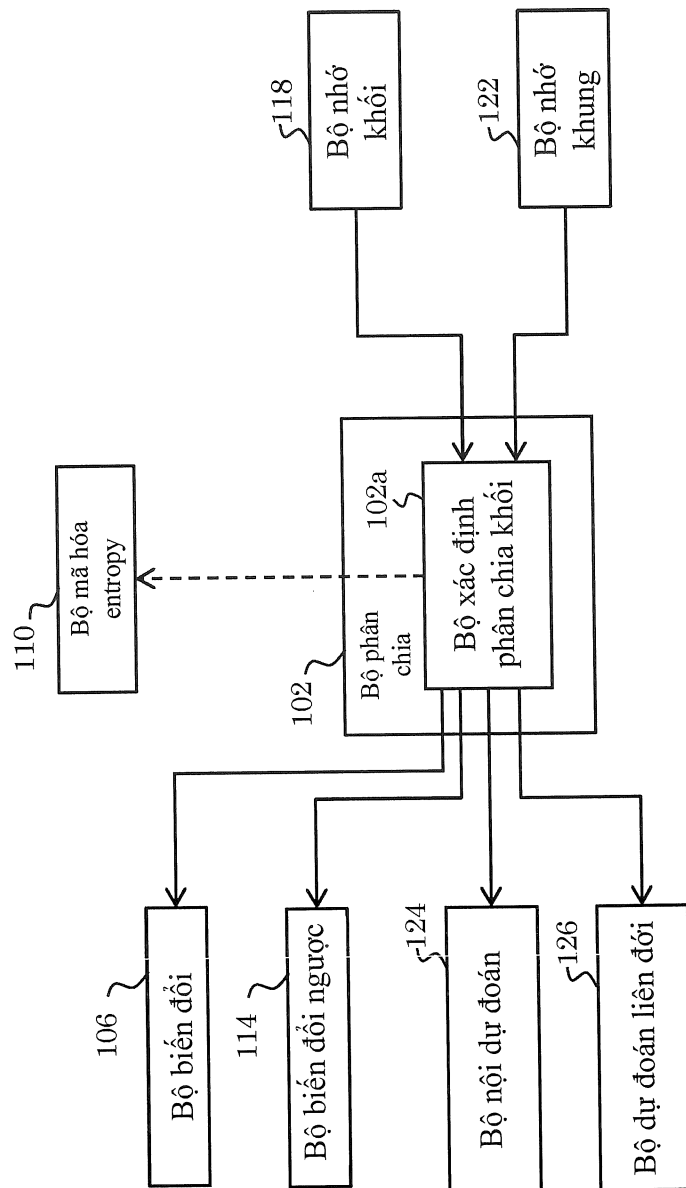
[Fig. 9]



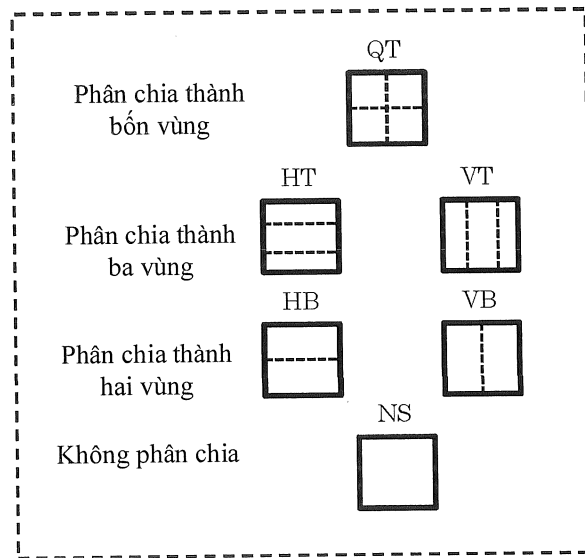
[Fig. 10]



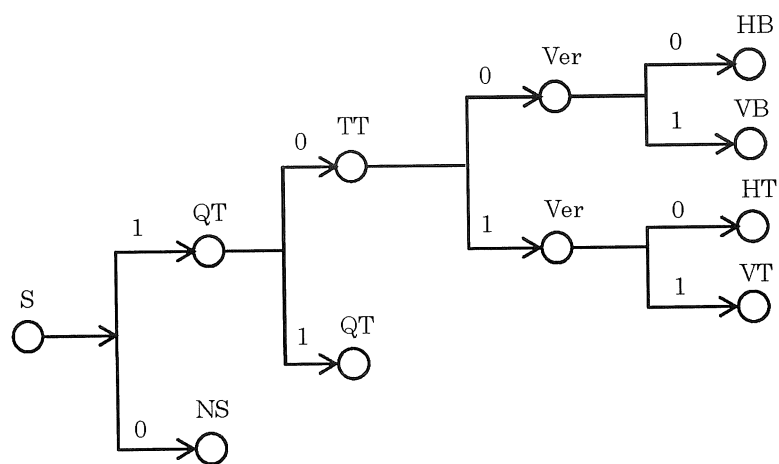
[Fig. 11]



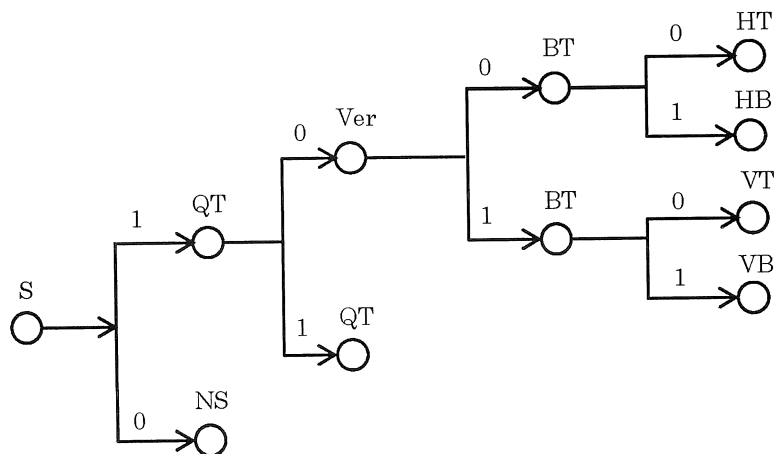
[Fig. 12]



[Fig. 13A]



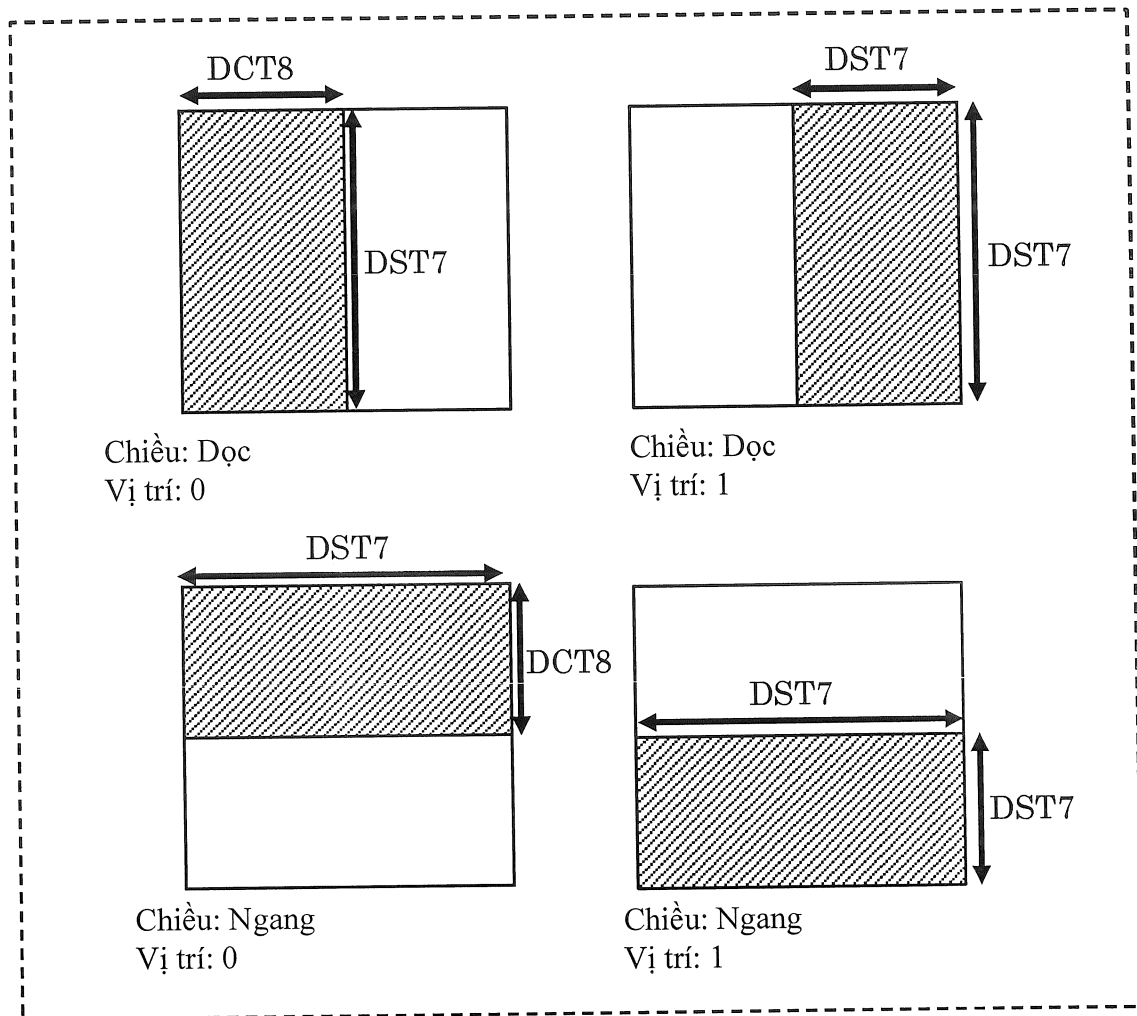
[Fig. 13B]



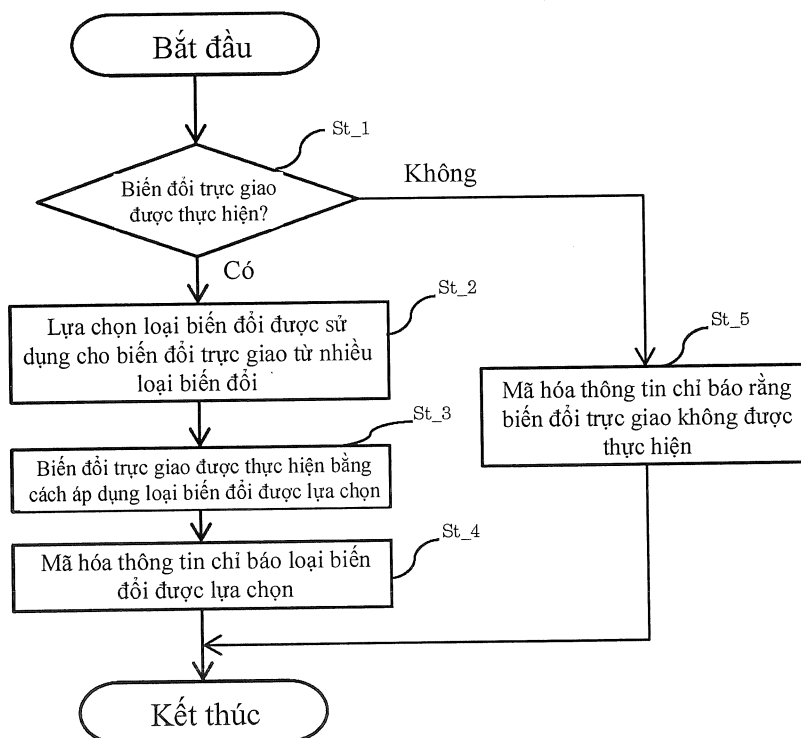
[Fig. 14]

Loại biến đổi	Hàm cơ sở $T_i(j)$, $i, j = 0, 1, \dots, N-1$
DCT - II	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos \left(\frac{\pi \cdot i \cdot (2j+1)}{2N} \right)$ trong đó $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i=0 \\ 1 & i \neq 0 \end{cases}$
DCT - V	$T_i(j) = \omega_0 \cdot \omega_1 \cdot \sqrt{\frac{2}{2N-1}} \cdot \cos \left(\frac{2\pi \cdot i \cdot j}{2N-1} \right)$ trong đó $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i=0 \\ 1 & i \neq 0 \end{cases}$, $\omega_1 = \begin{cases} \sqrt{\frac{2}{N}} & j=0 \\ 1 & j \neq 0 \end{cases}$
DCT - VIII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \cos \left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N+2} \right)$
DST - I	$T_i(j) = \sqrt{\frac{2}{N+1}} \cdot \sin \left(\frac{\pi \cdot (i+1) \cdot (j+1)}{N+1} \right)$
DST - VII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \sin \left(\frac{\pi \cdot (2i+1) \cdot (j+1)}{2N+1} \right)$

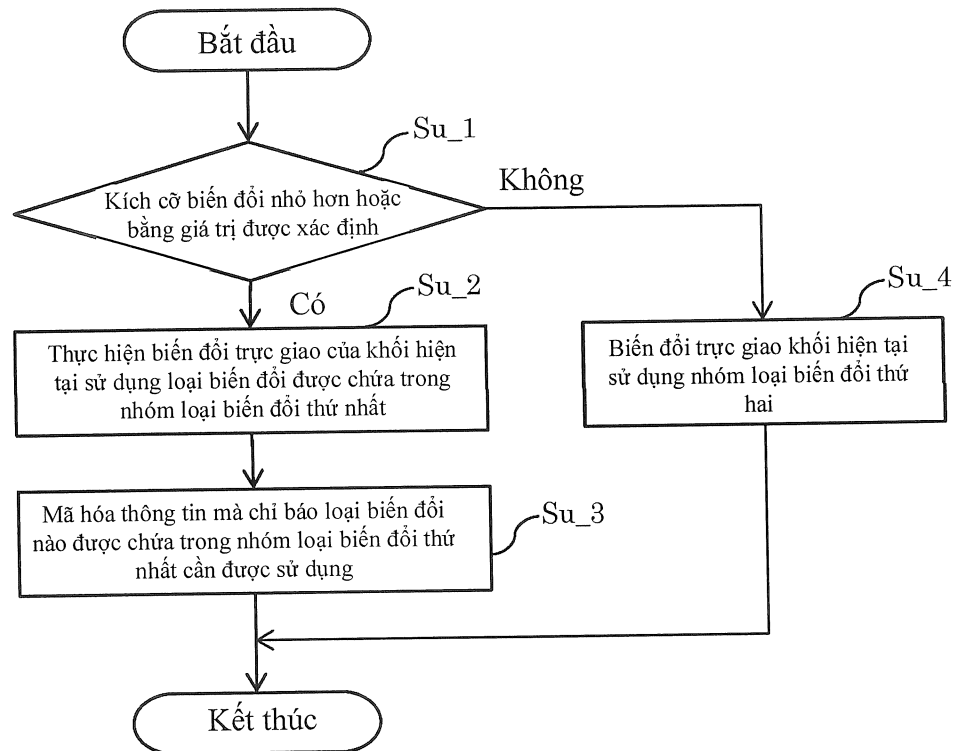
[Fig. 15]



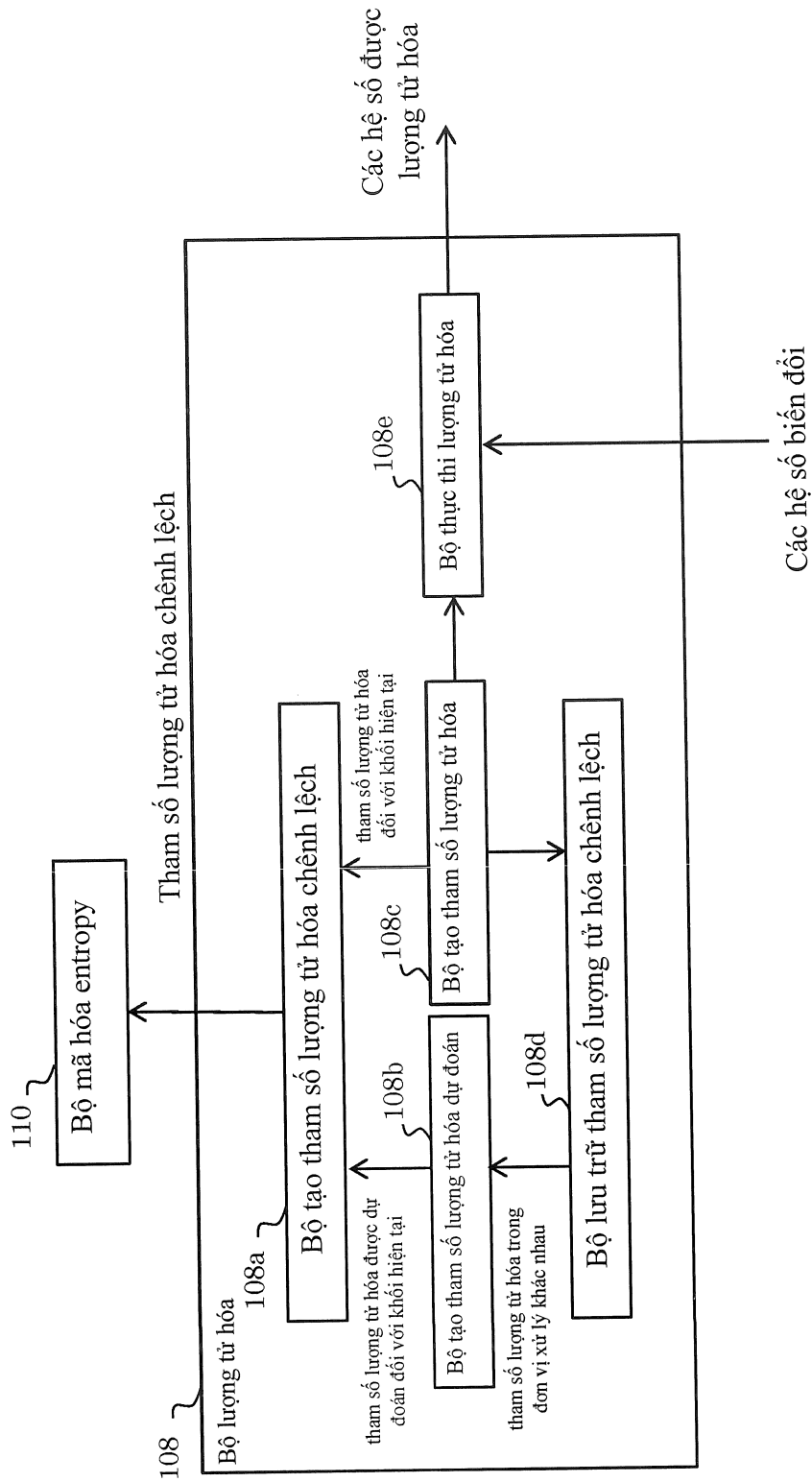
[Fig. 16]



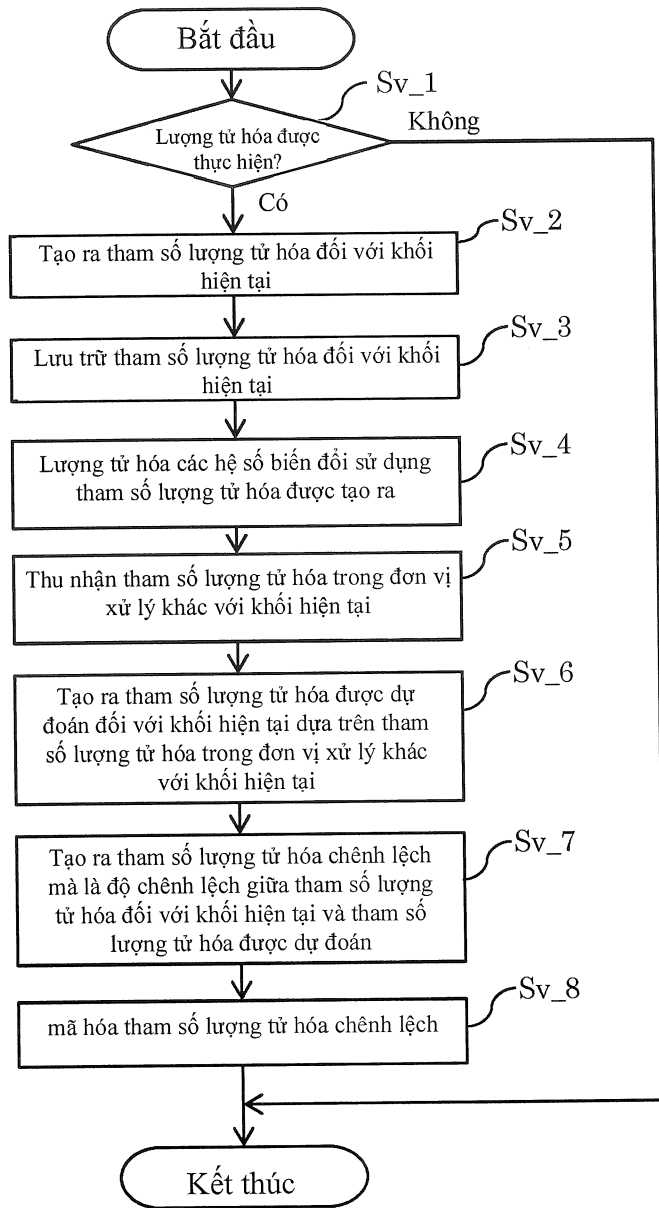
[Fig. 17]



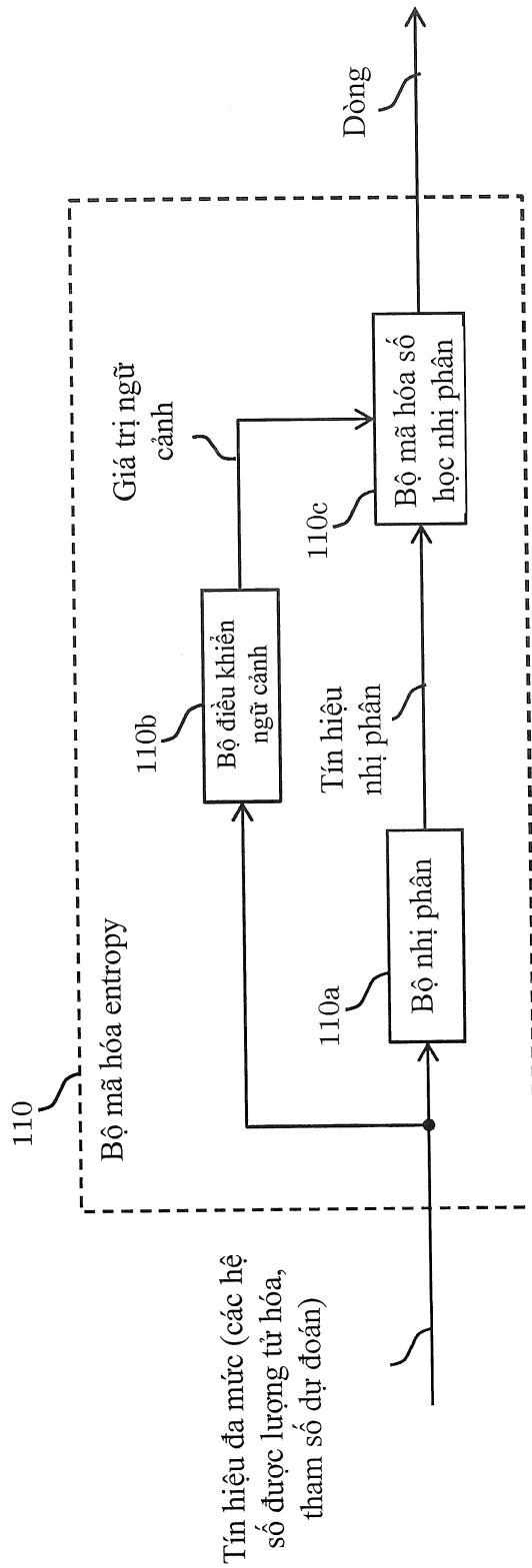
[Fig. 18]



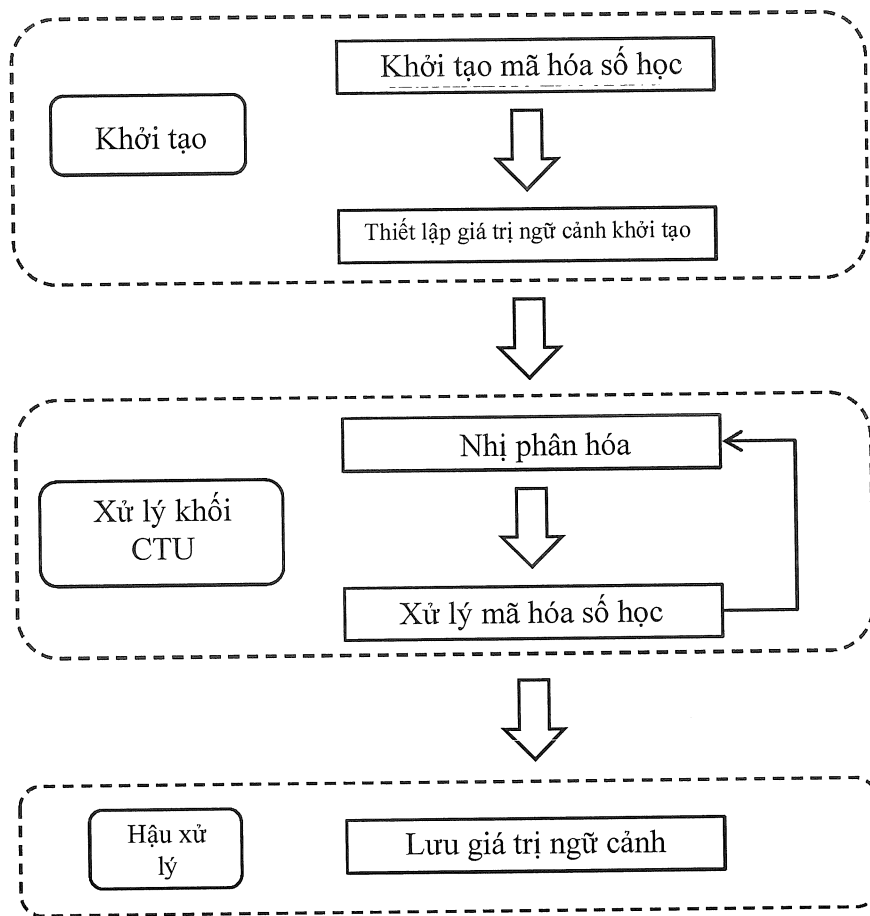
[Fig. 19]



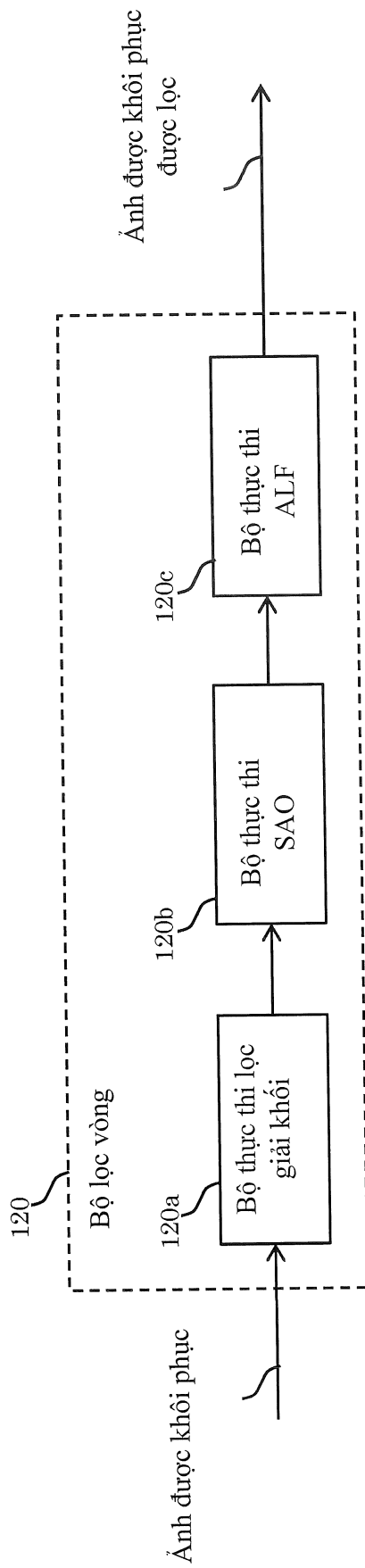
[Fig. 20]



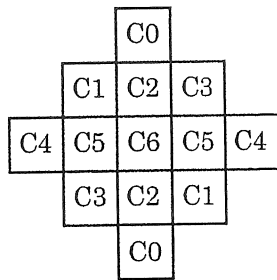
[Fig. 21]



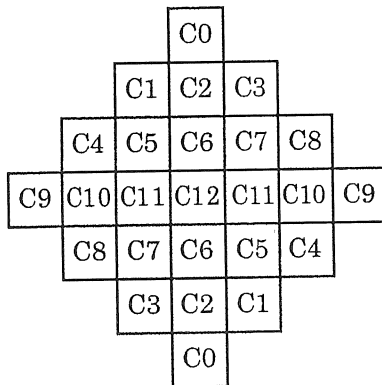
[Fig. 22]



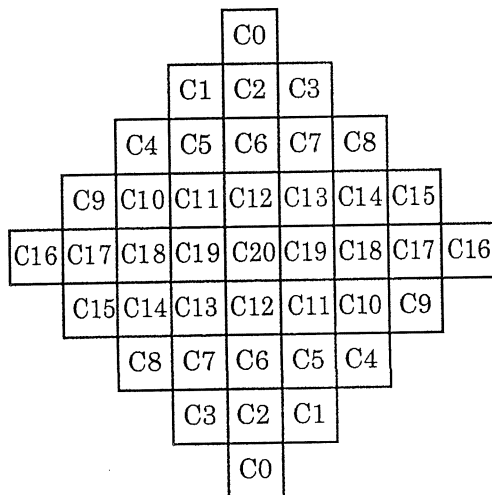
[Fig. 23A]



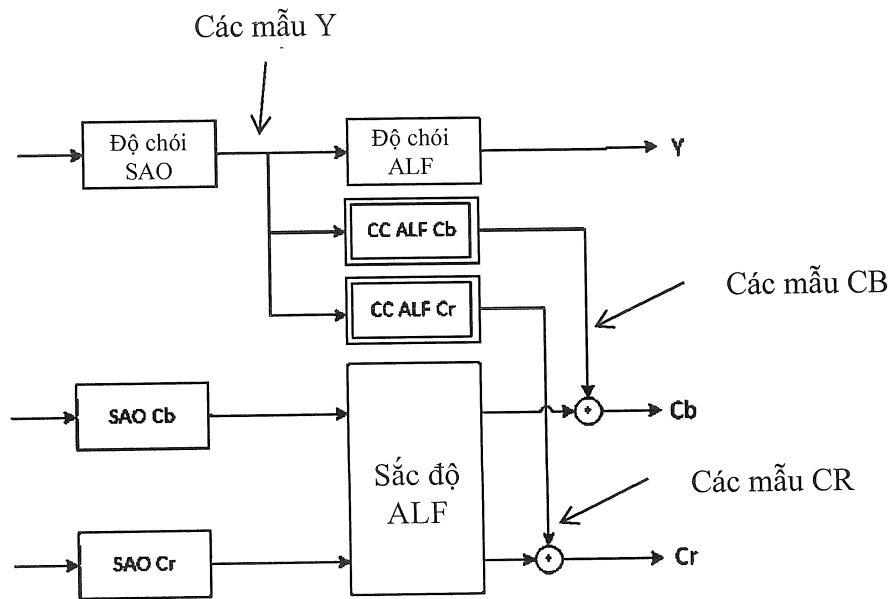
[Fig. 23B]



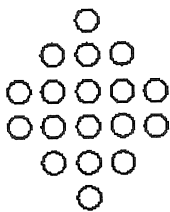
[Fig. 23C]



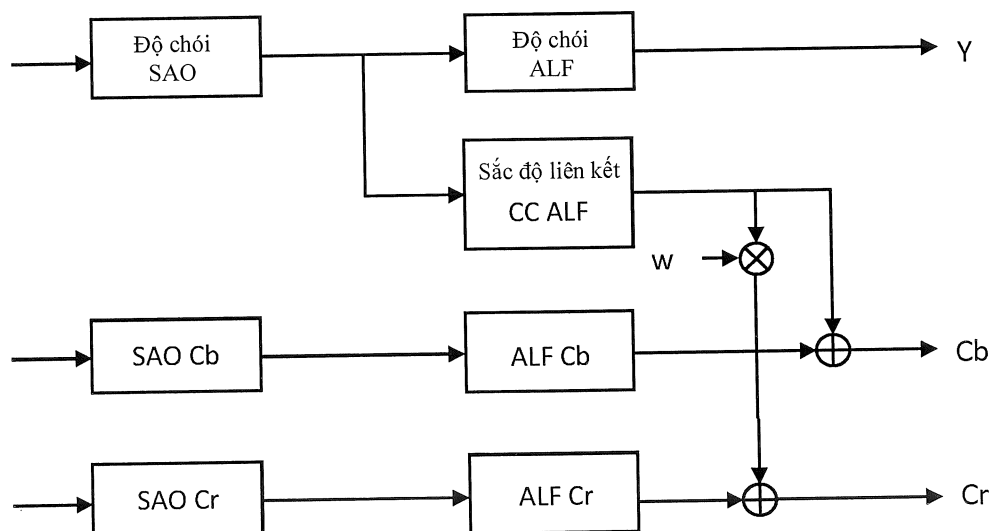
[Fig. 23D]



[Fig. 23E]



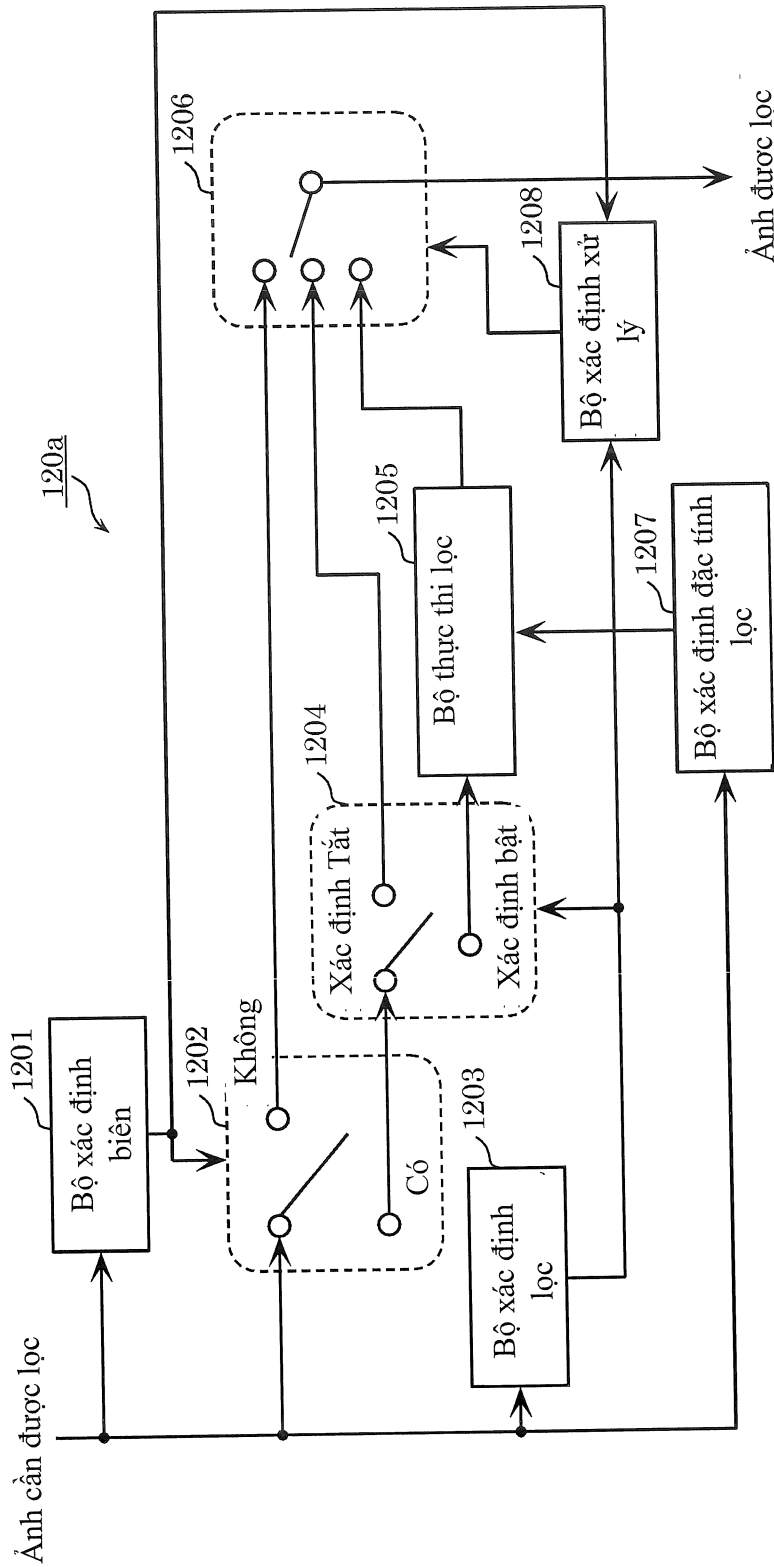
[Fig. 23F]



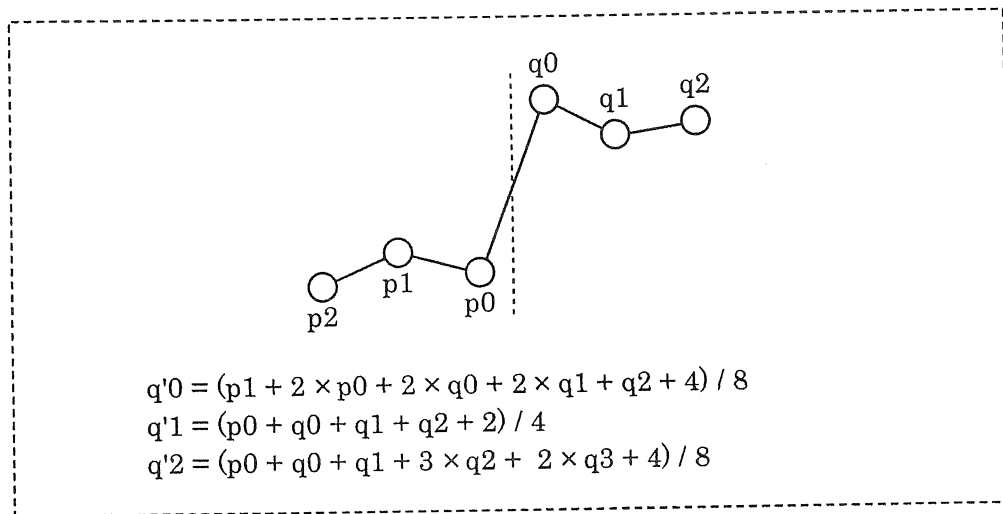
[Fig. 23G]

Chỉ số	1	2	3	4	5	6	7
$ w $	1/4	1/2	3/4	1	4	2	4/3

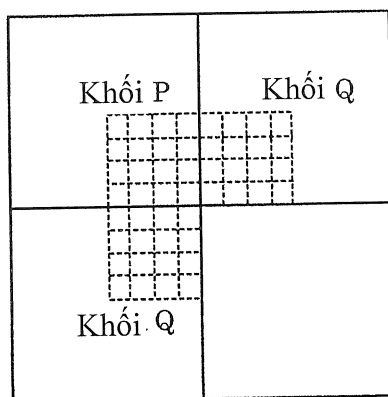
[Fig. 24]



[Fig. 25]



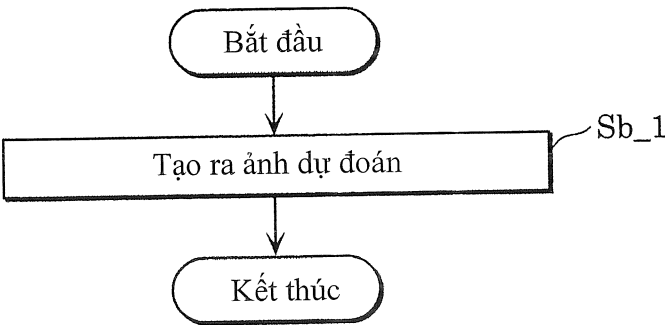
[Fig. 26]



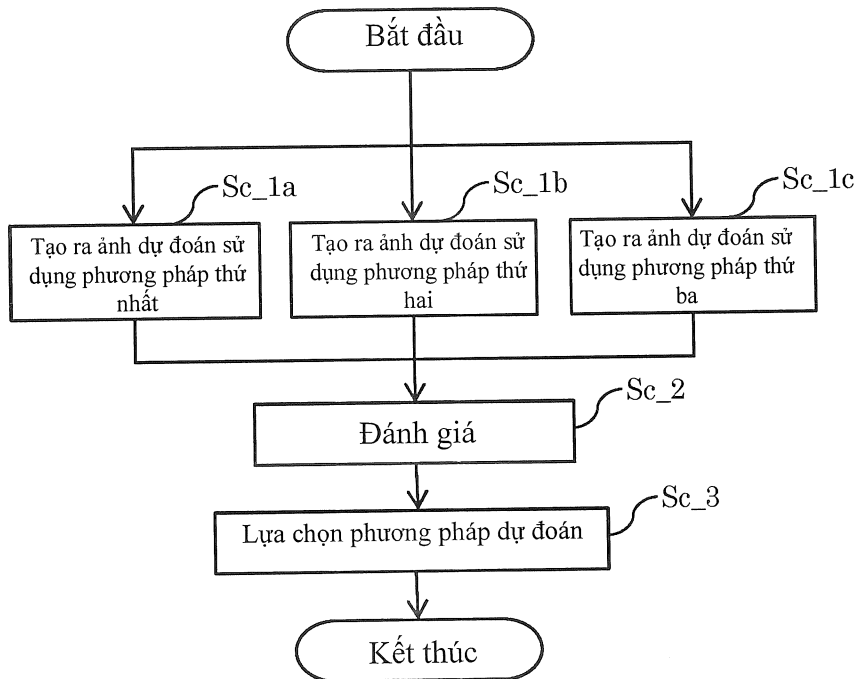
[Fig. 27]

Các điều kiện đối với các khối P và Q nằm qua biên mà để giá trị Bs được xác định	các giá trị Bs
- Ít nhất một trong số các khối là khối dự đoán nội bộ	2
- Ít nhất một trong số các khối bao gồm các hệ số biến đổi không phải 0 và có phía mà trùng với biên giữa các khối biến đổi trực giao	1
- Các giá trị tuyệt đối của độ chênh lệch giữa các vec tơ chuyển động của hai khối đi qua biên tương ứng với một hoặc nhiều điểm ảnh	1
- Các ảnh tham chiếu để bù chuyển động của hai khối qua biên hoặc các số lượng vectơ chuyển động của hai khối là khác nhau	1
- Khác	0

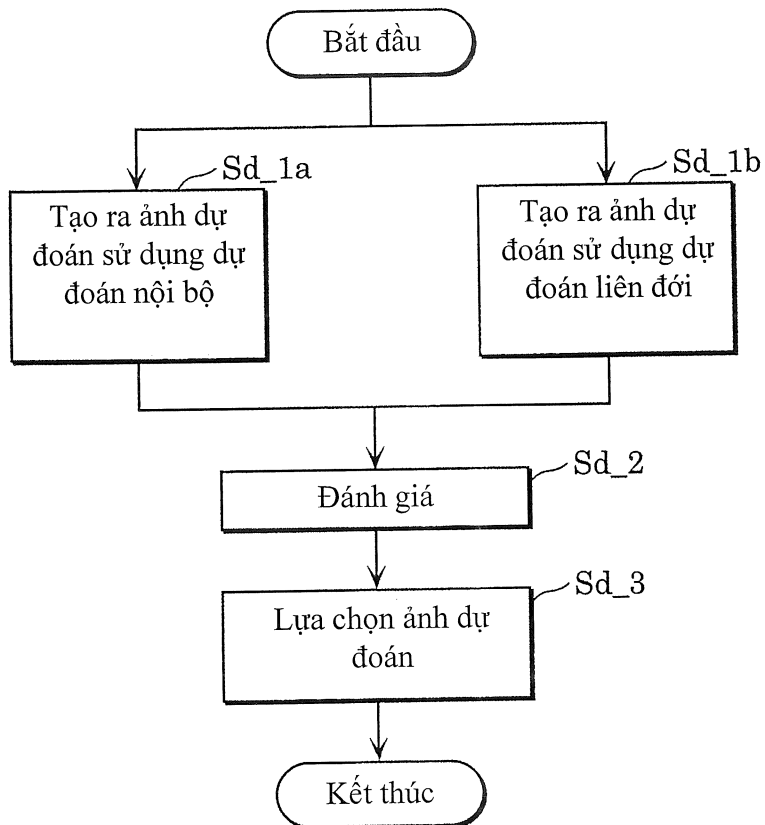
[Fig. 28]



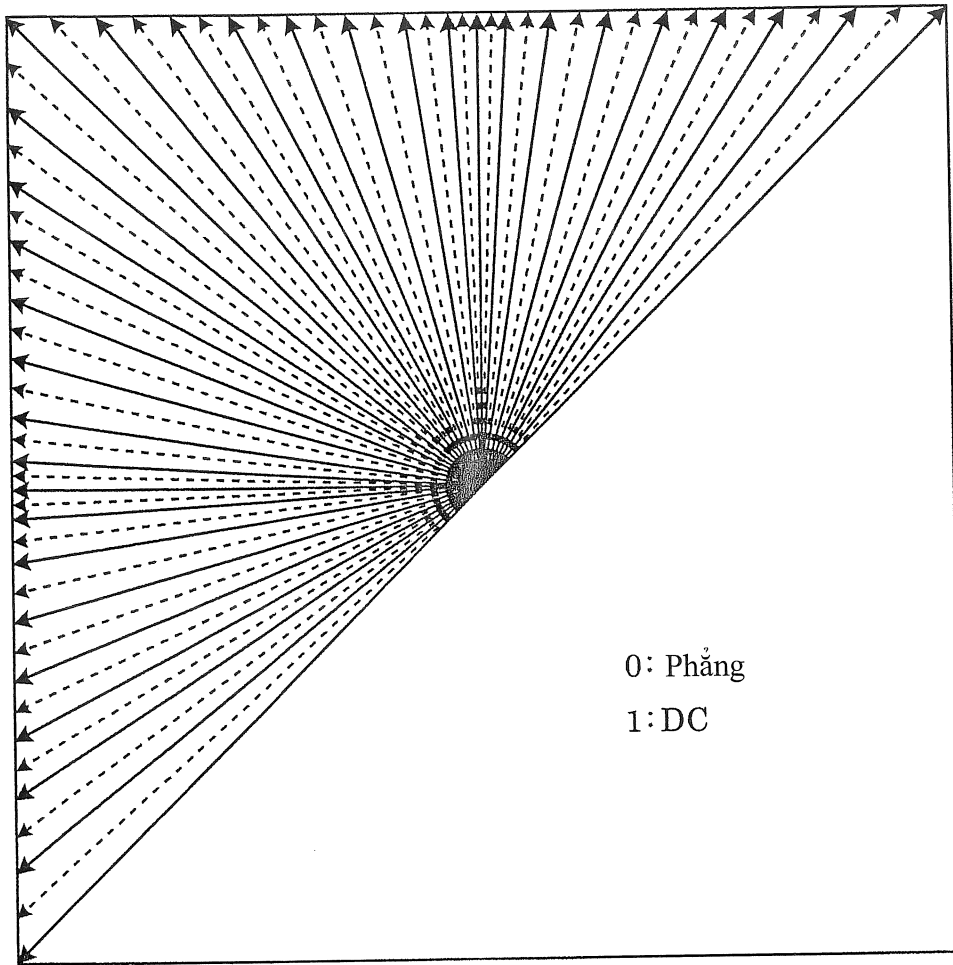
[Fig. 29]



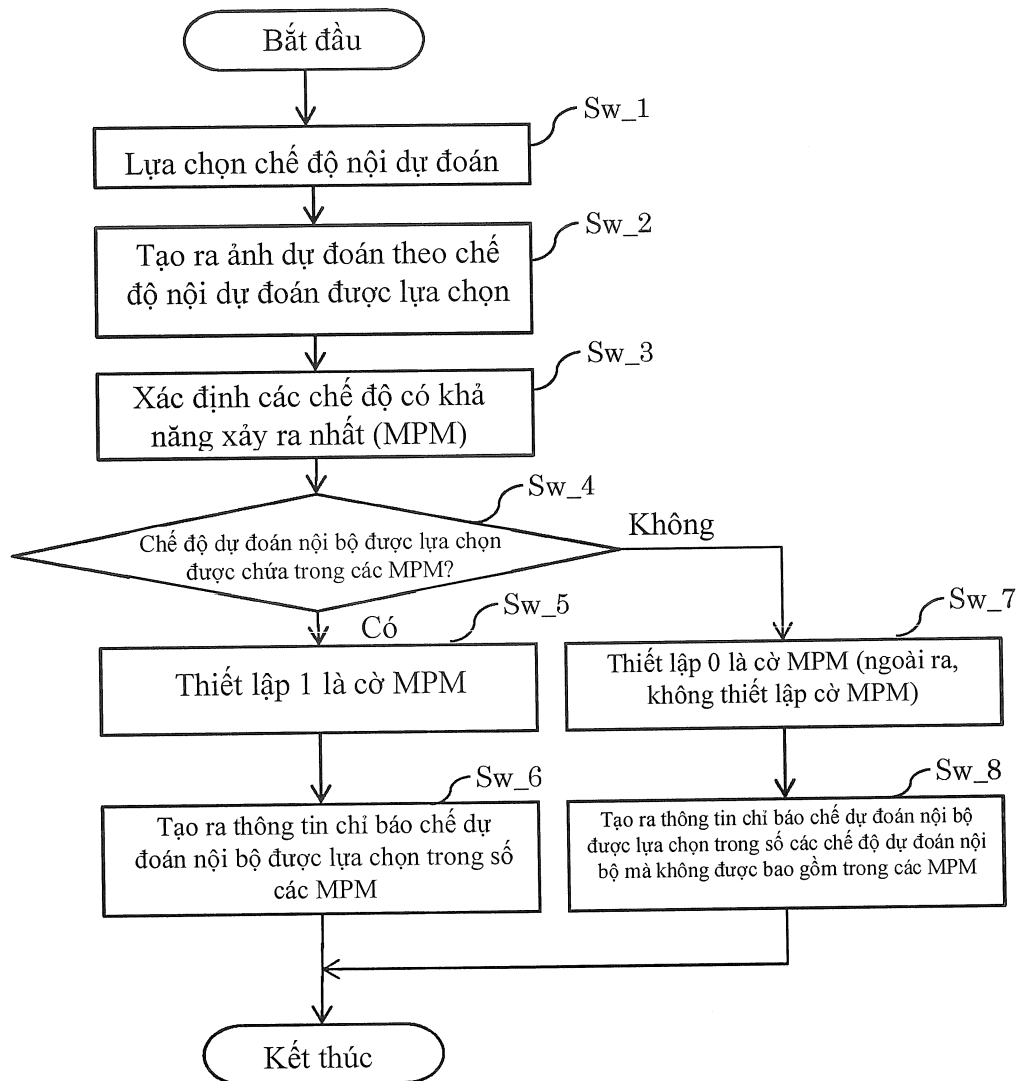
[Fig. 30]



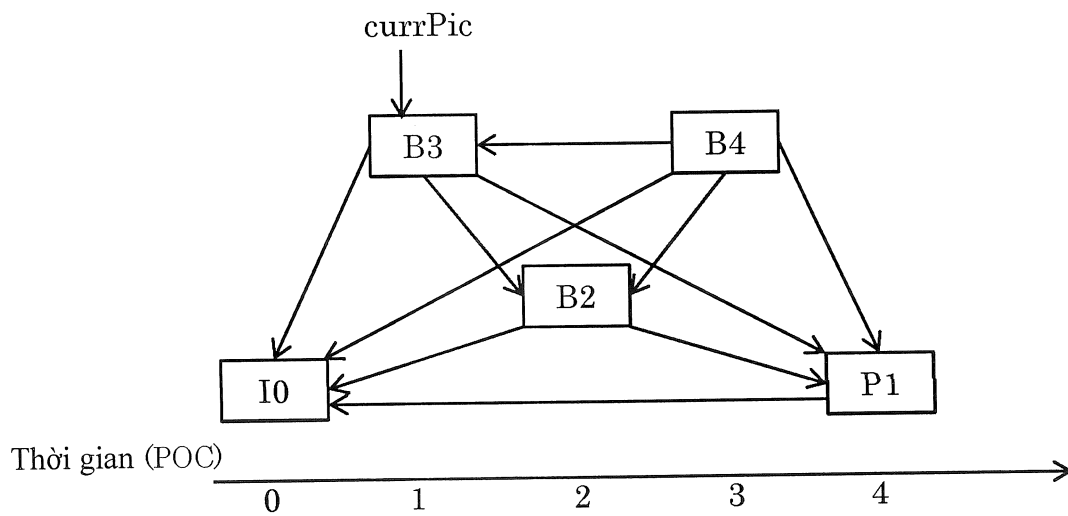
[Fig. 31]



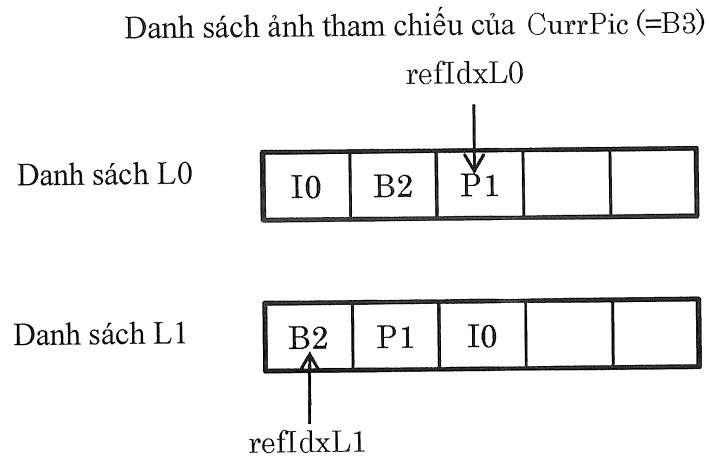
[Fig. 32]



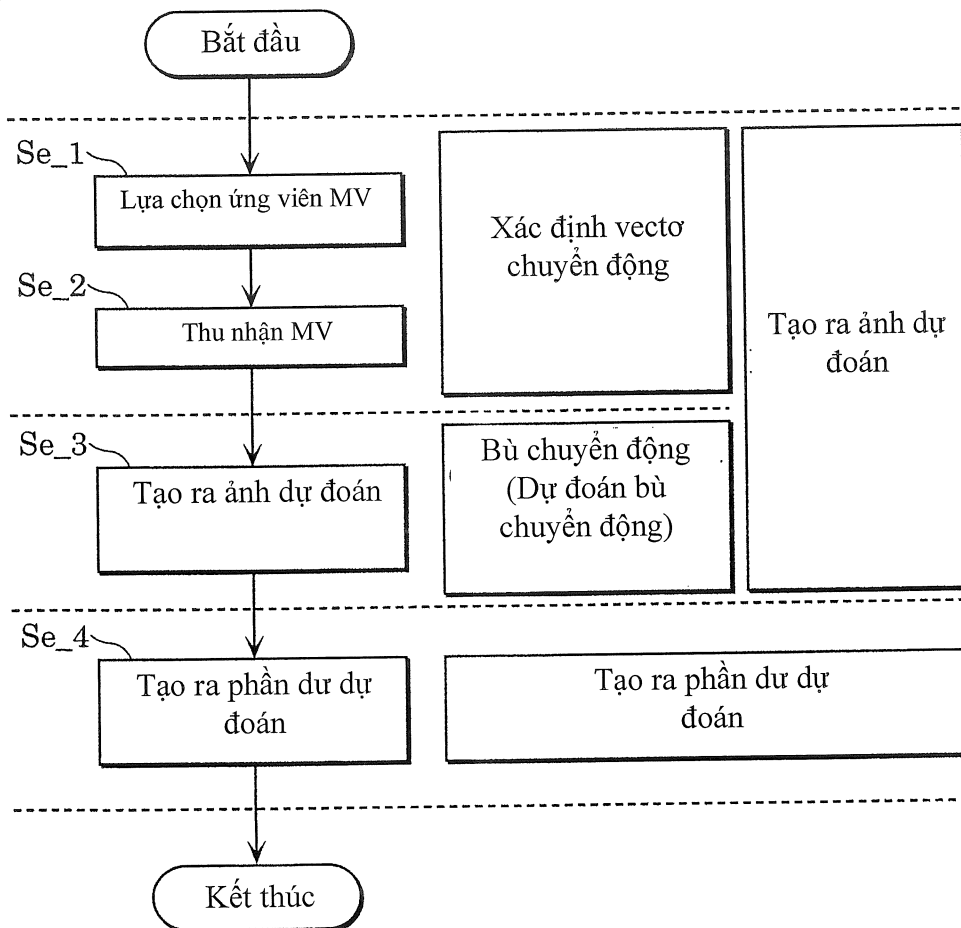
[Fig. 33]



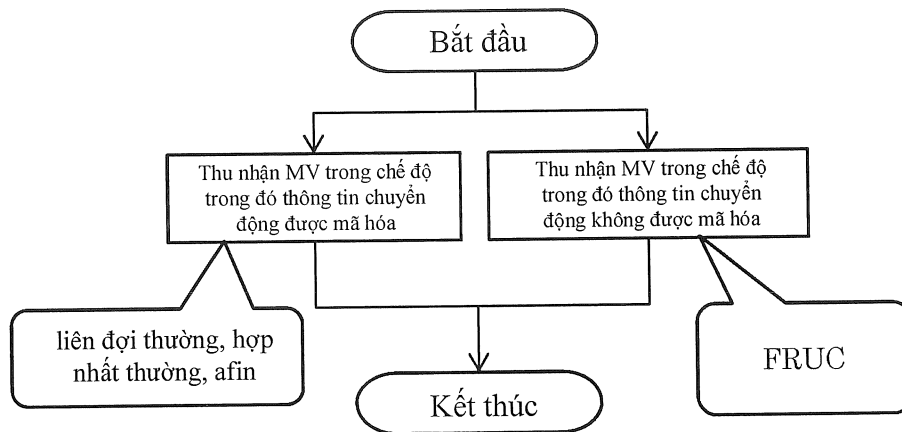
[Fig. 34]



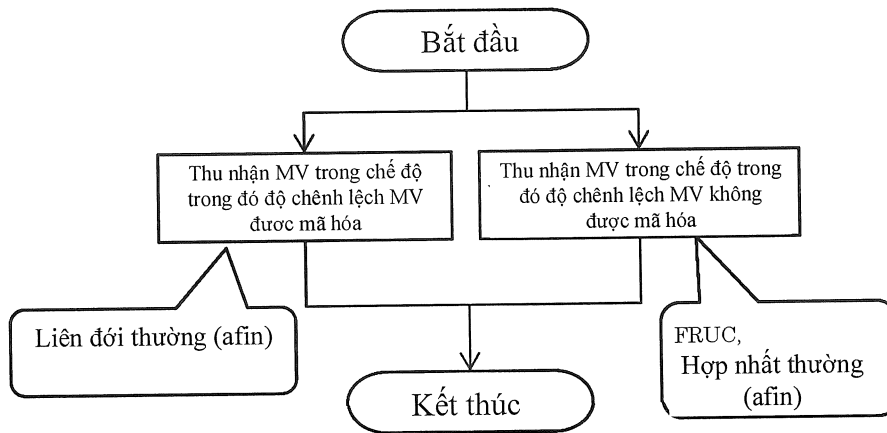
[Fig. 35]



[Fig. 36]



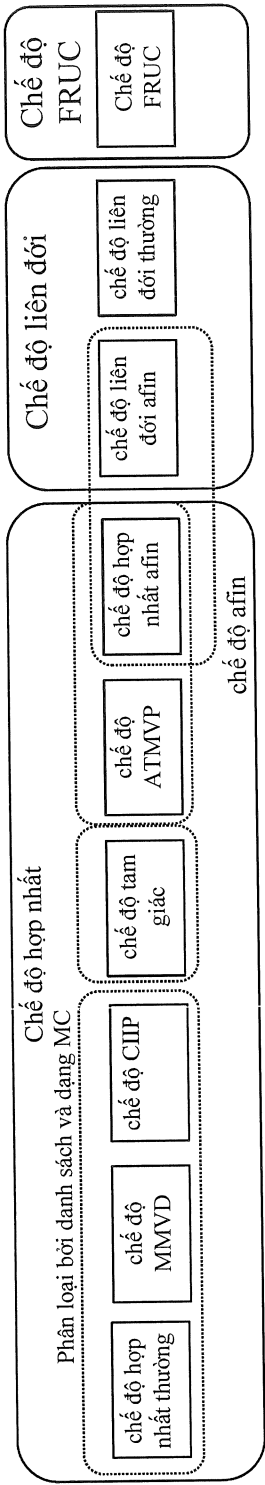
[Fig. 37]



[Fig. 38A]

Chế độ thu nhận MV		Thông tin chuyển động	Độ chênh lệch MV
Chế độ liên đới		○	○
Chế độ hợp nhất	Khác chế độ MMVD	○	×
	Chế độ MMVD	○	△
Chế độ FRUC		×	×

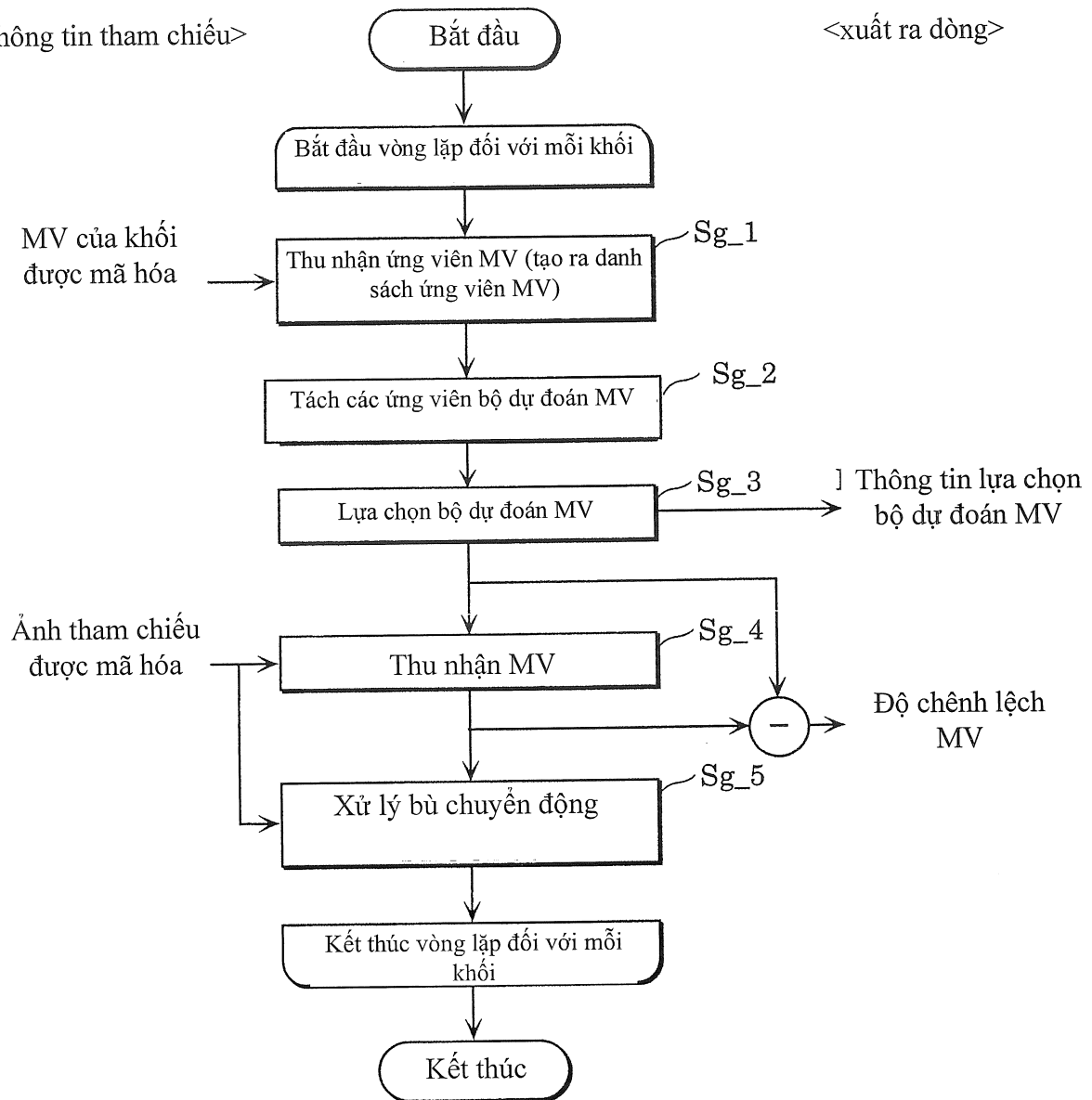
[Fig. 38B]



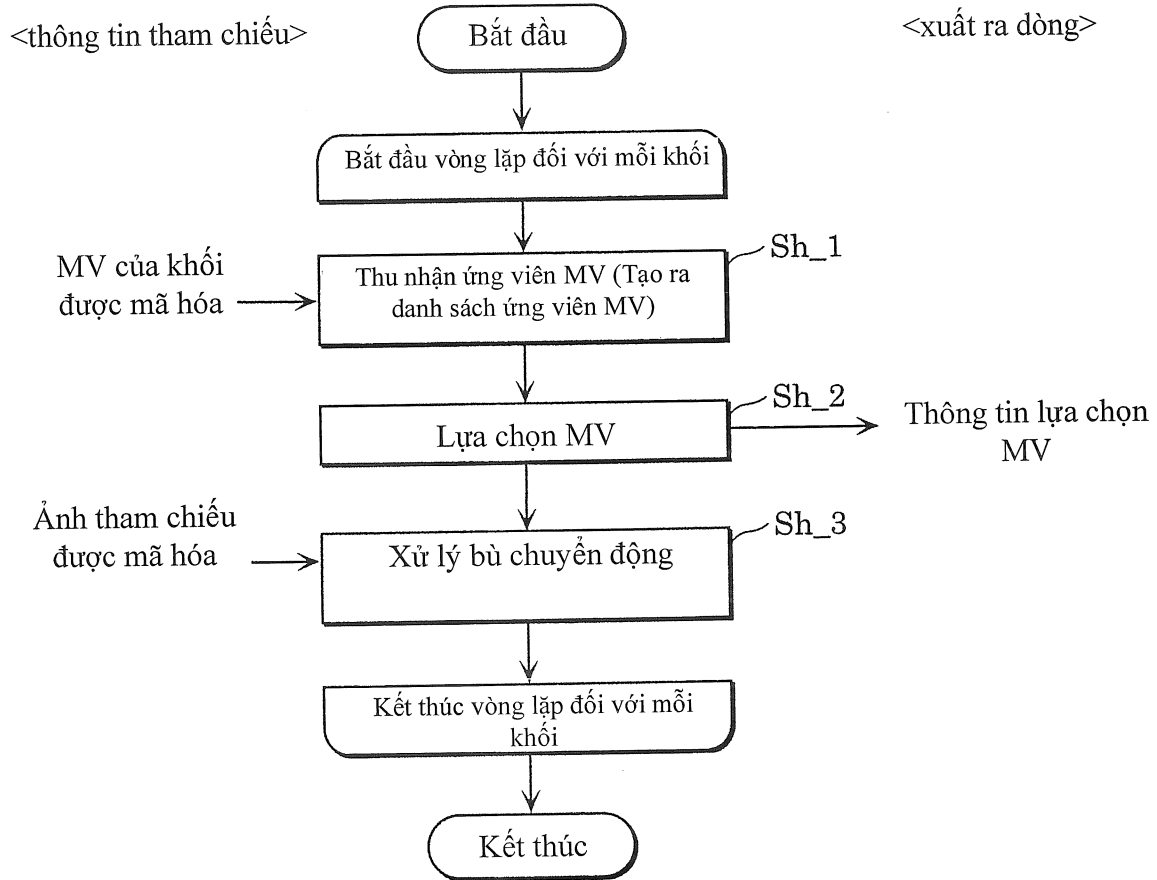
[Fig. 39]

<thông tin tham chiếu>

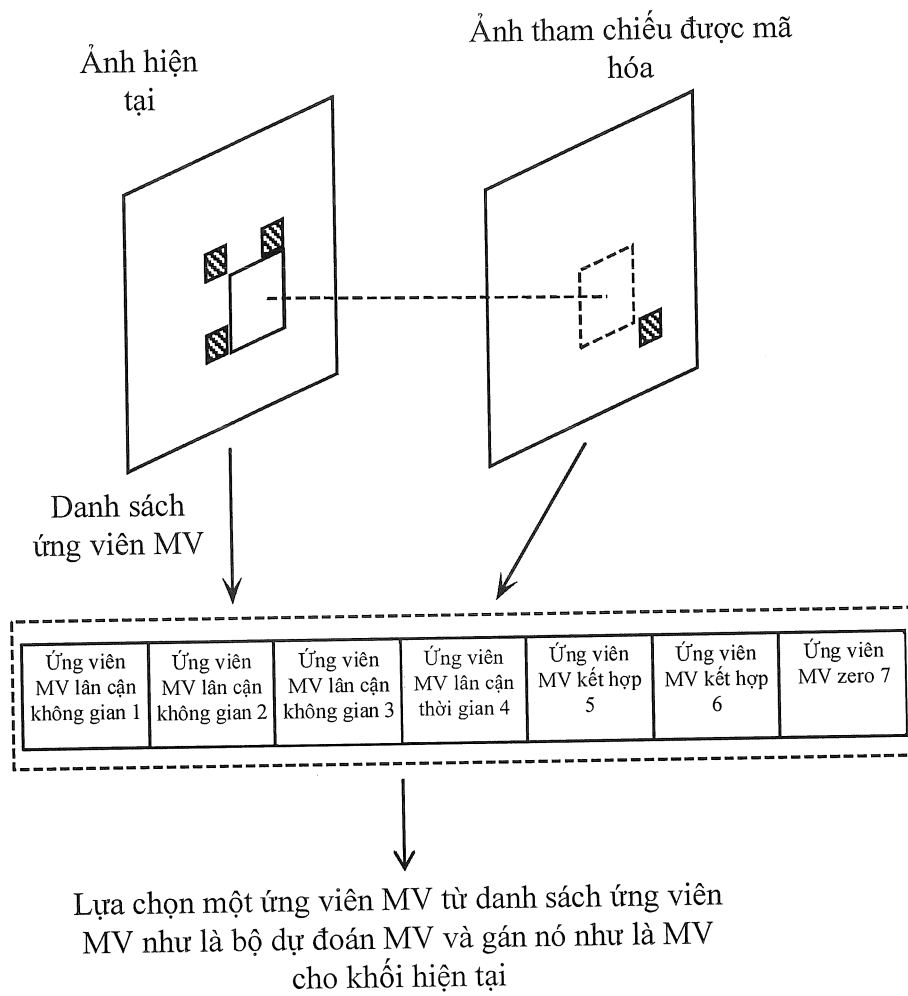
<xuất ra dòng>



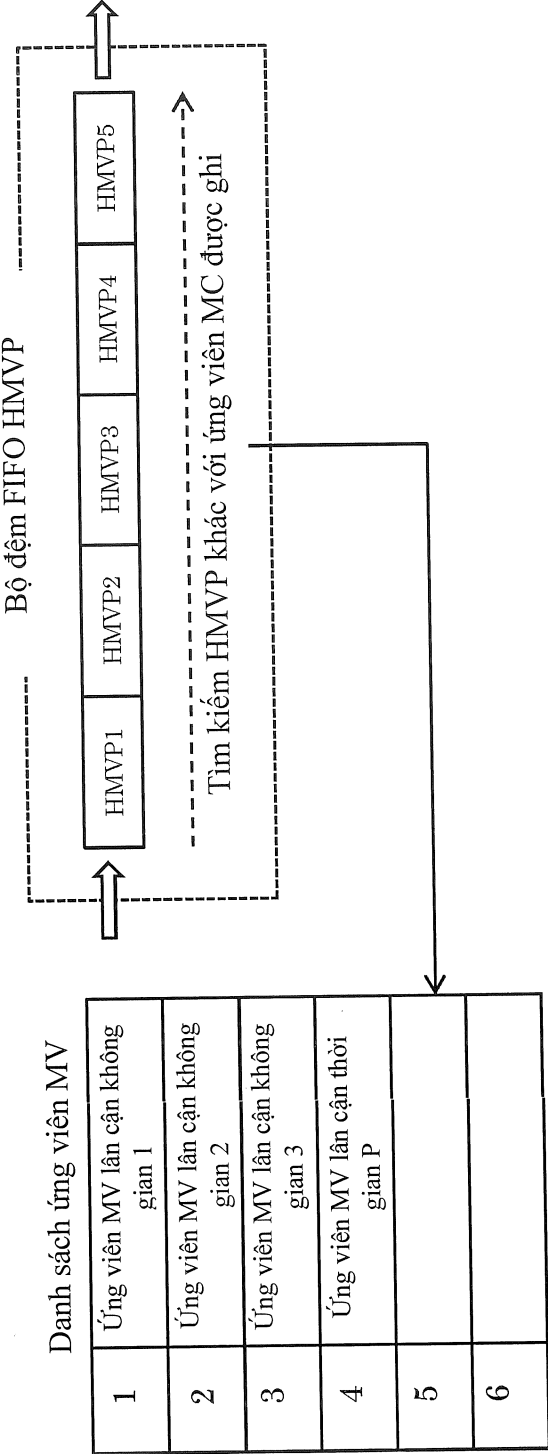
[Fig. 40]



[Fig. 41]

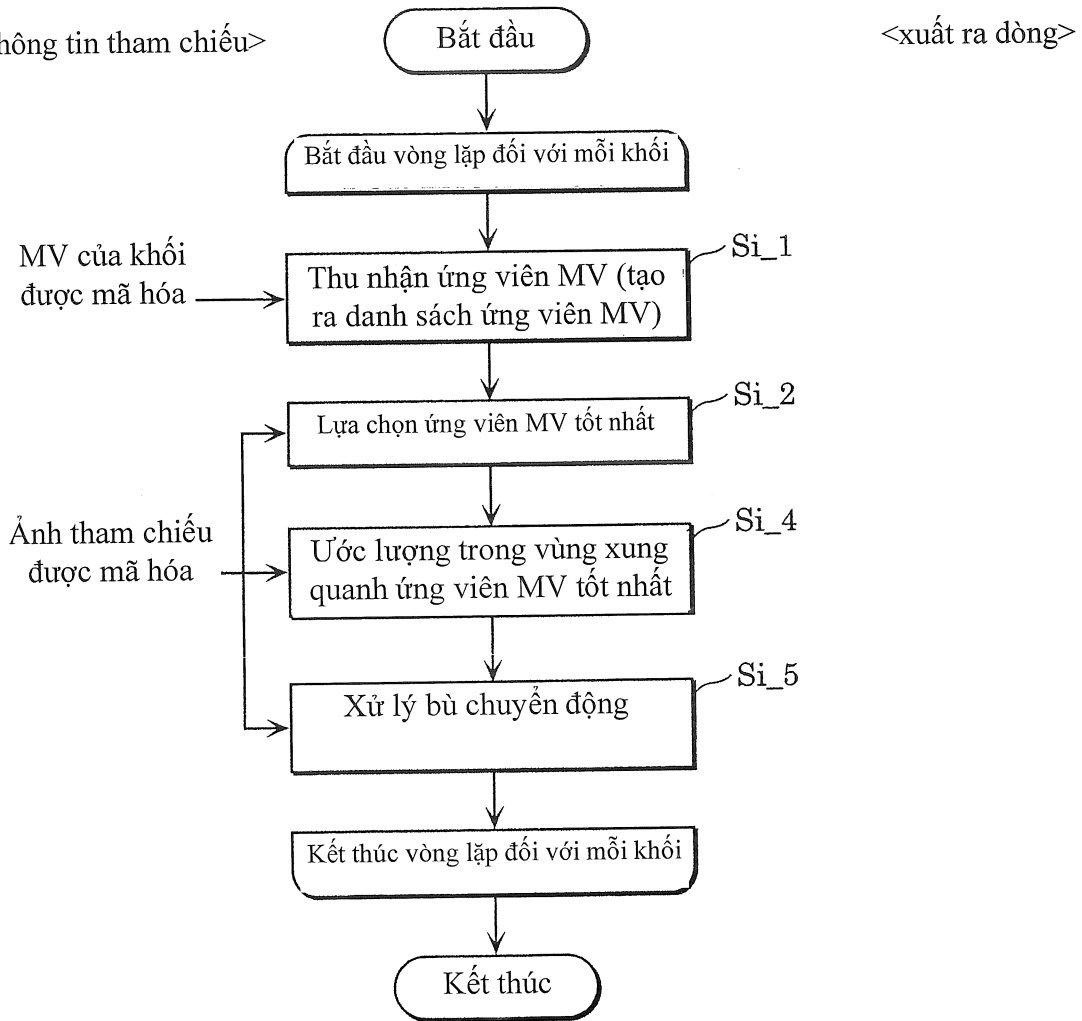


[Fig. 42]

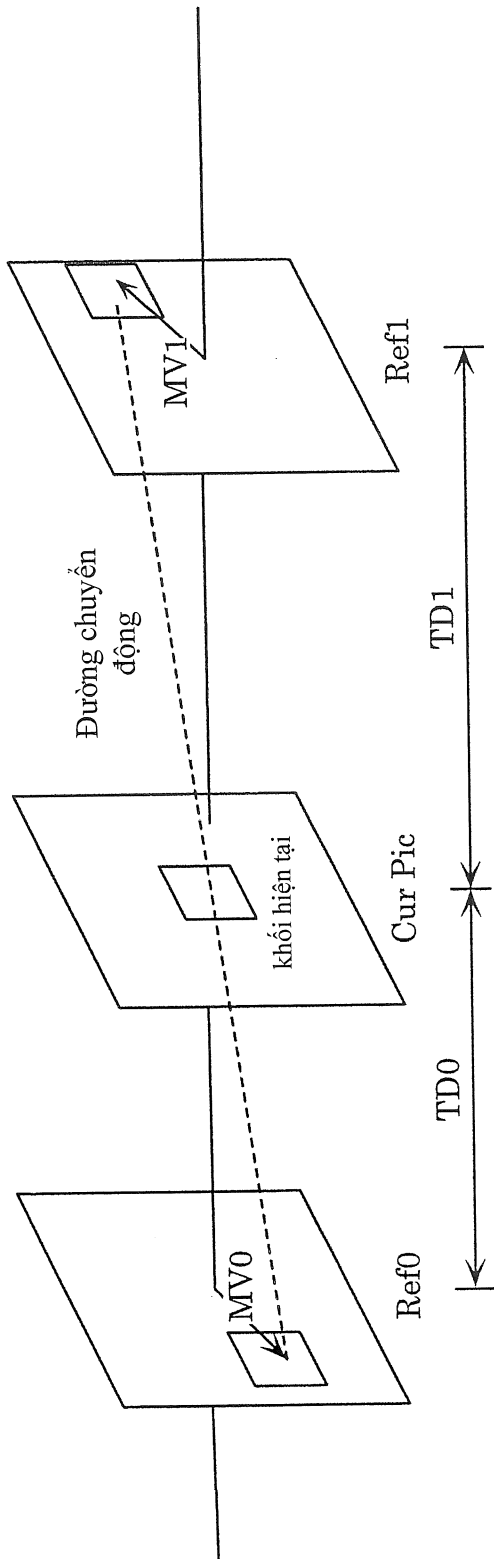


[Fig. 43]

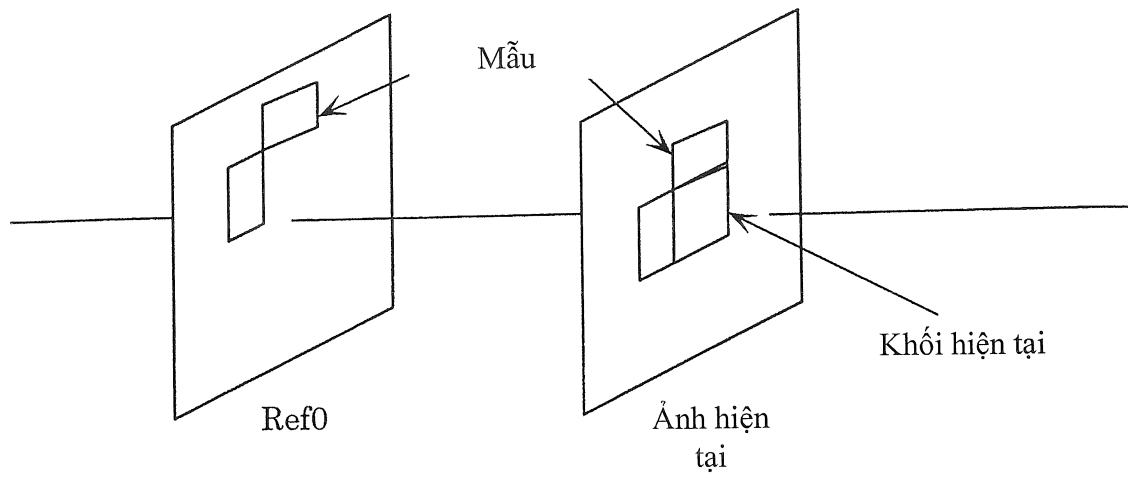
<thông tin tham chiếu>



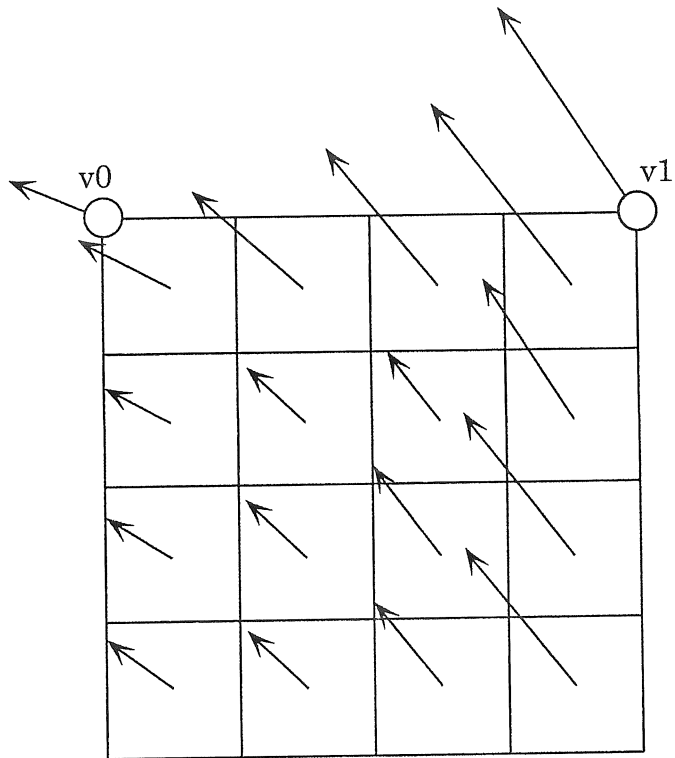
[Fig. 44]



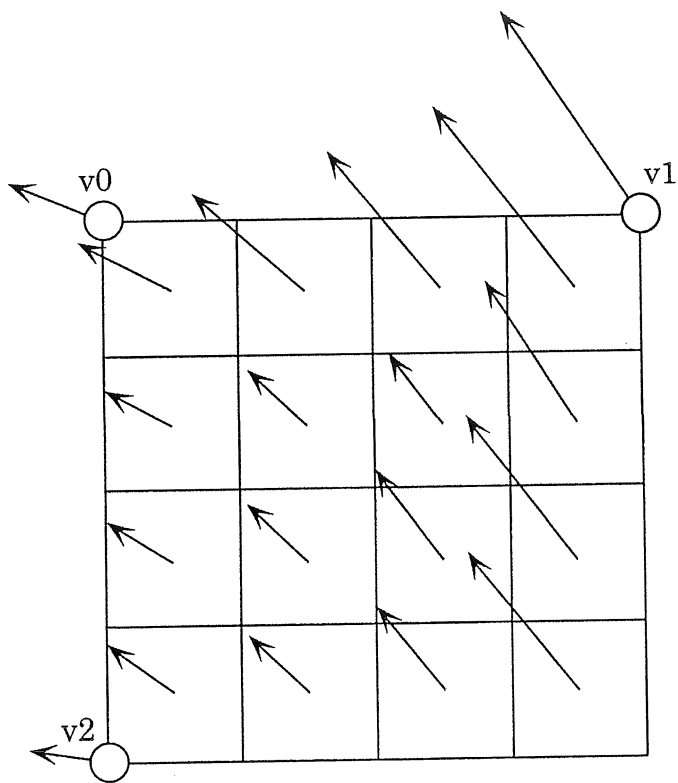
[Fig. 45]



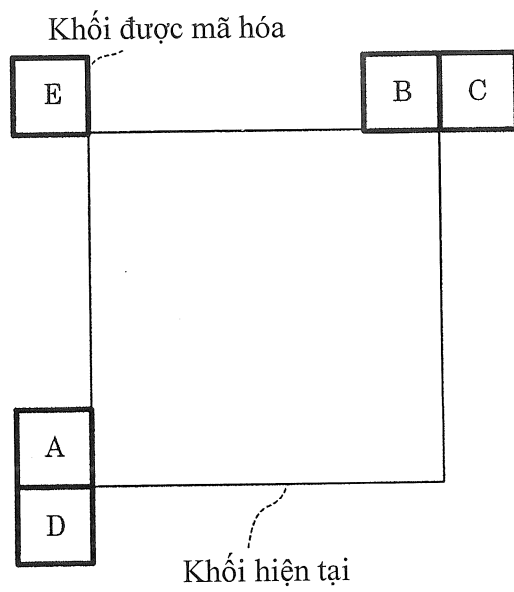
[Fig. 46A]



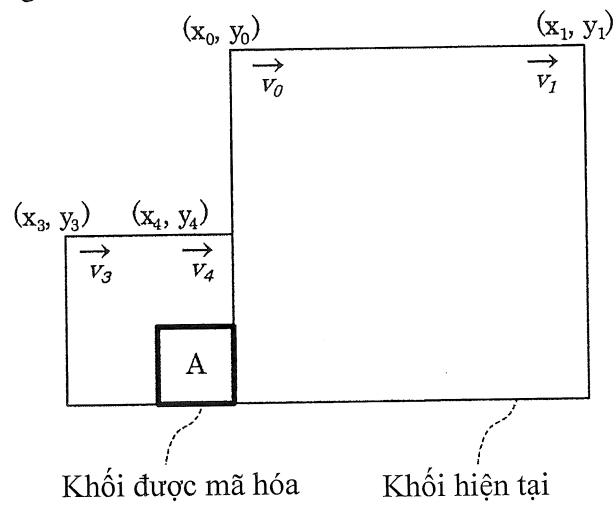
[Fig. 46B]



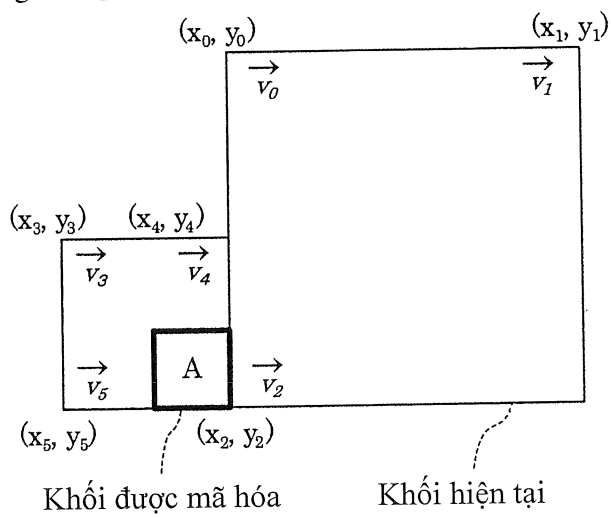
[Fig. 47A]



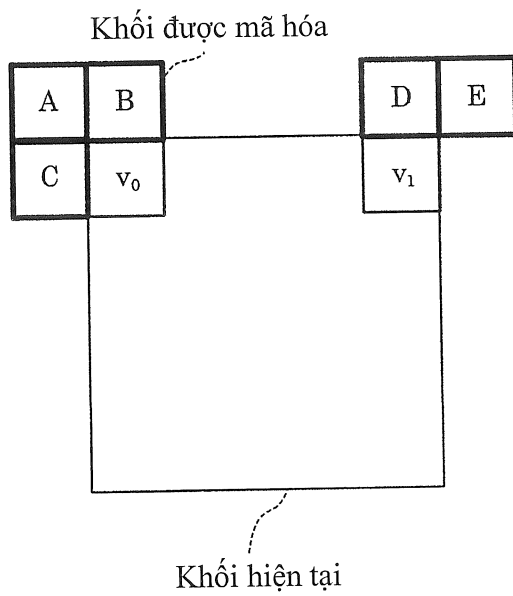
[Fig. 47B]



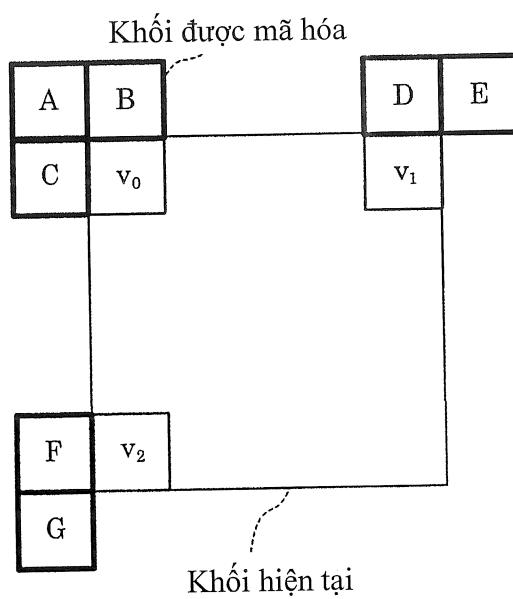
[Fig. 47C]



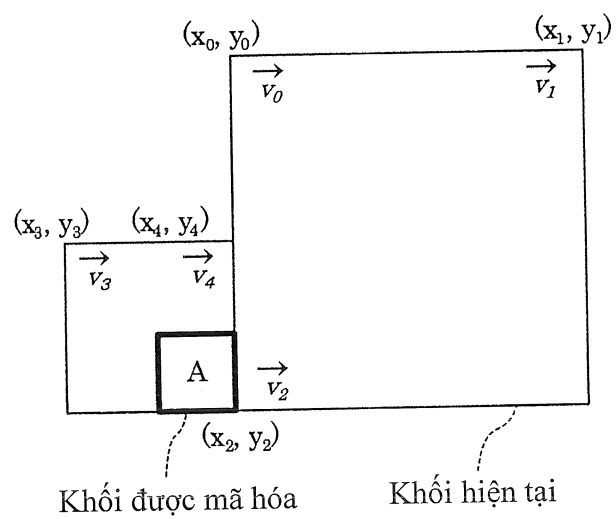
[Fig. 48A]



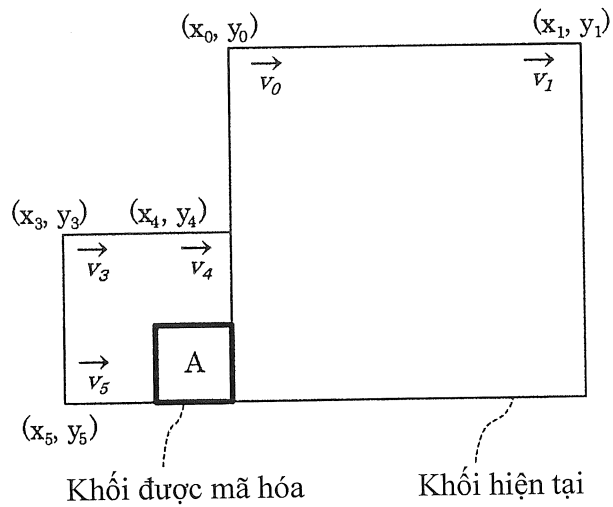
[Fig. 48B]



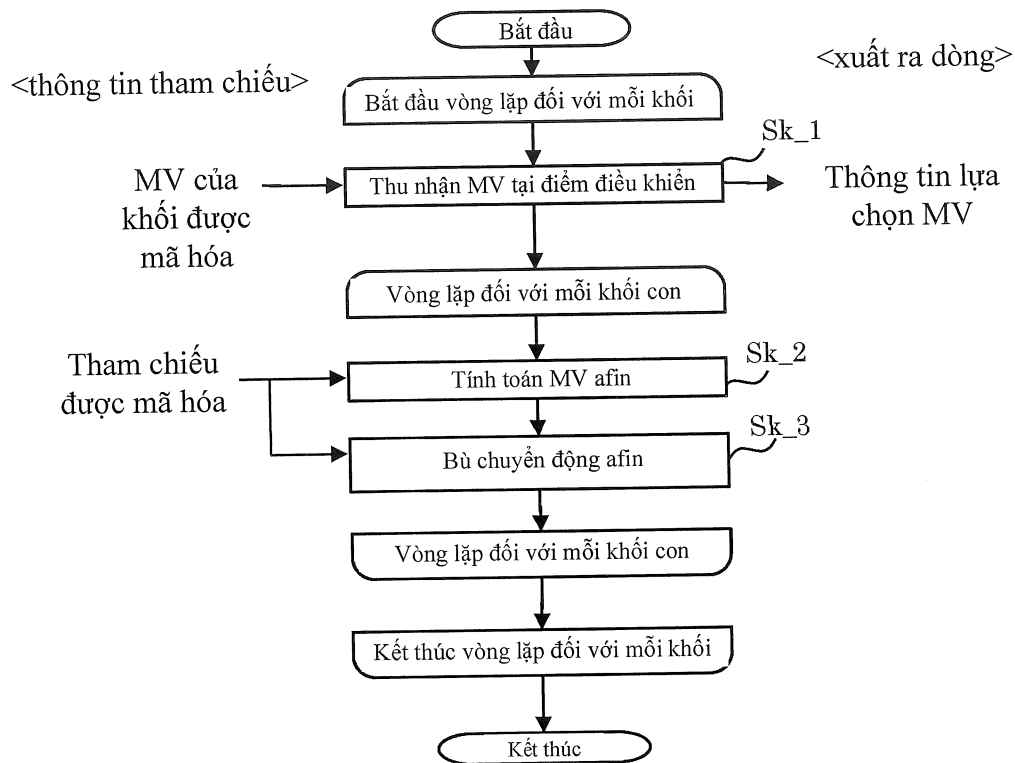
[Fig. 49A]



[Fig. 49B]



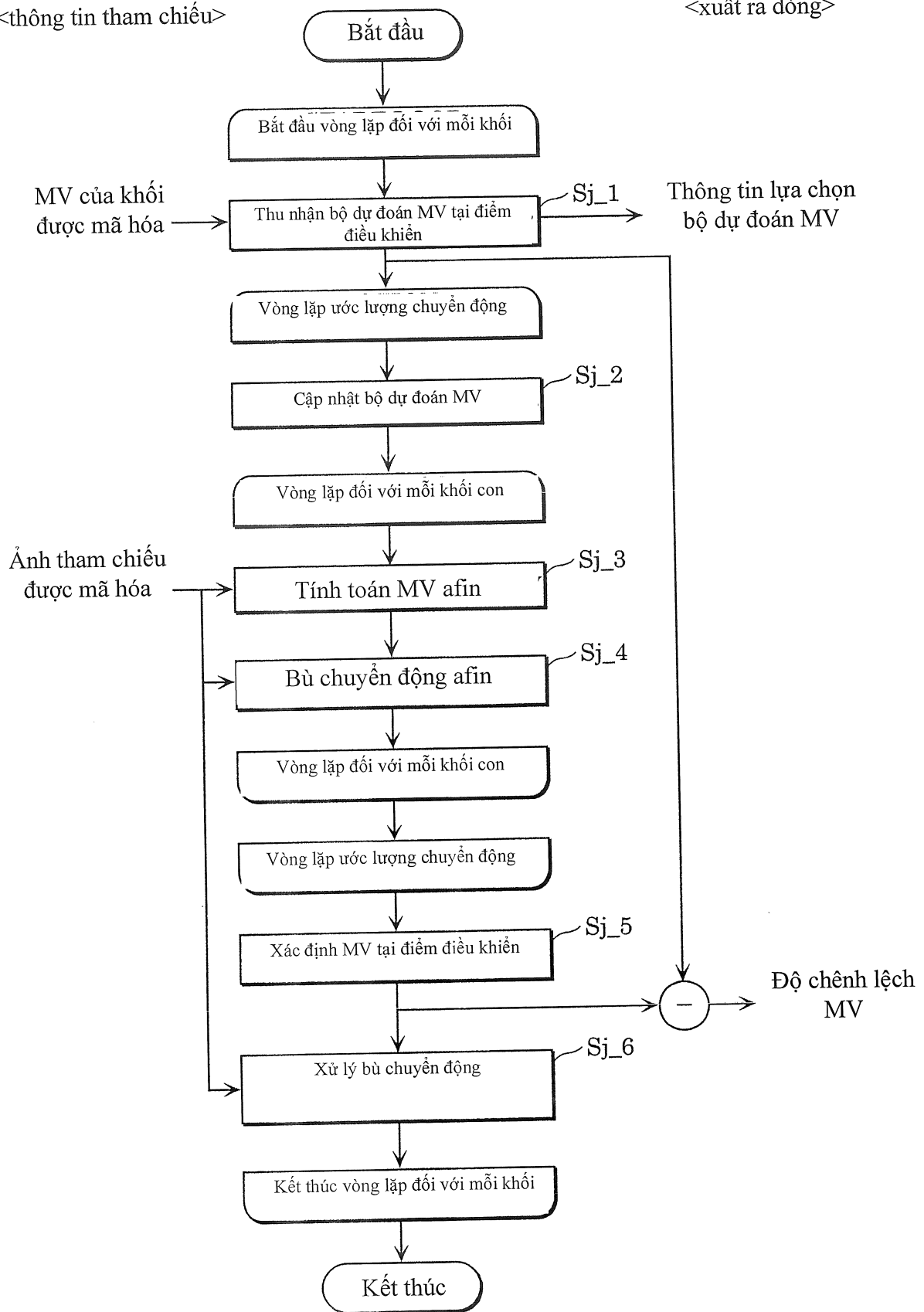
[Fig. 50]



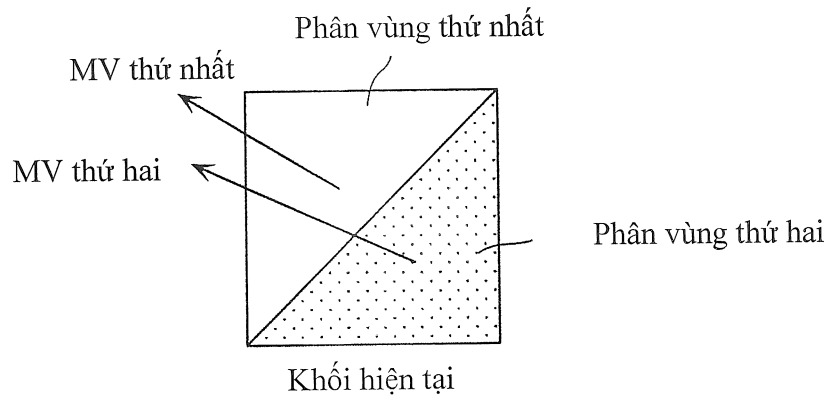
[Fig. 51]

<thông tin tham chiếu>

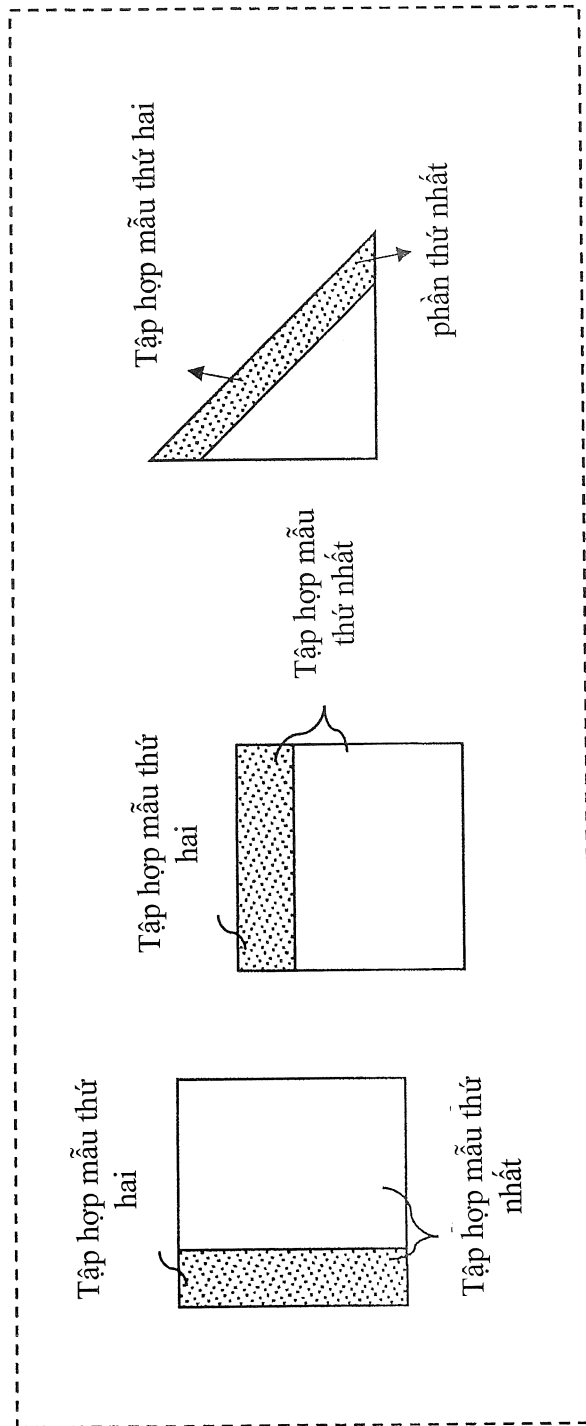
<xuất ra dòng>



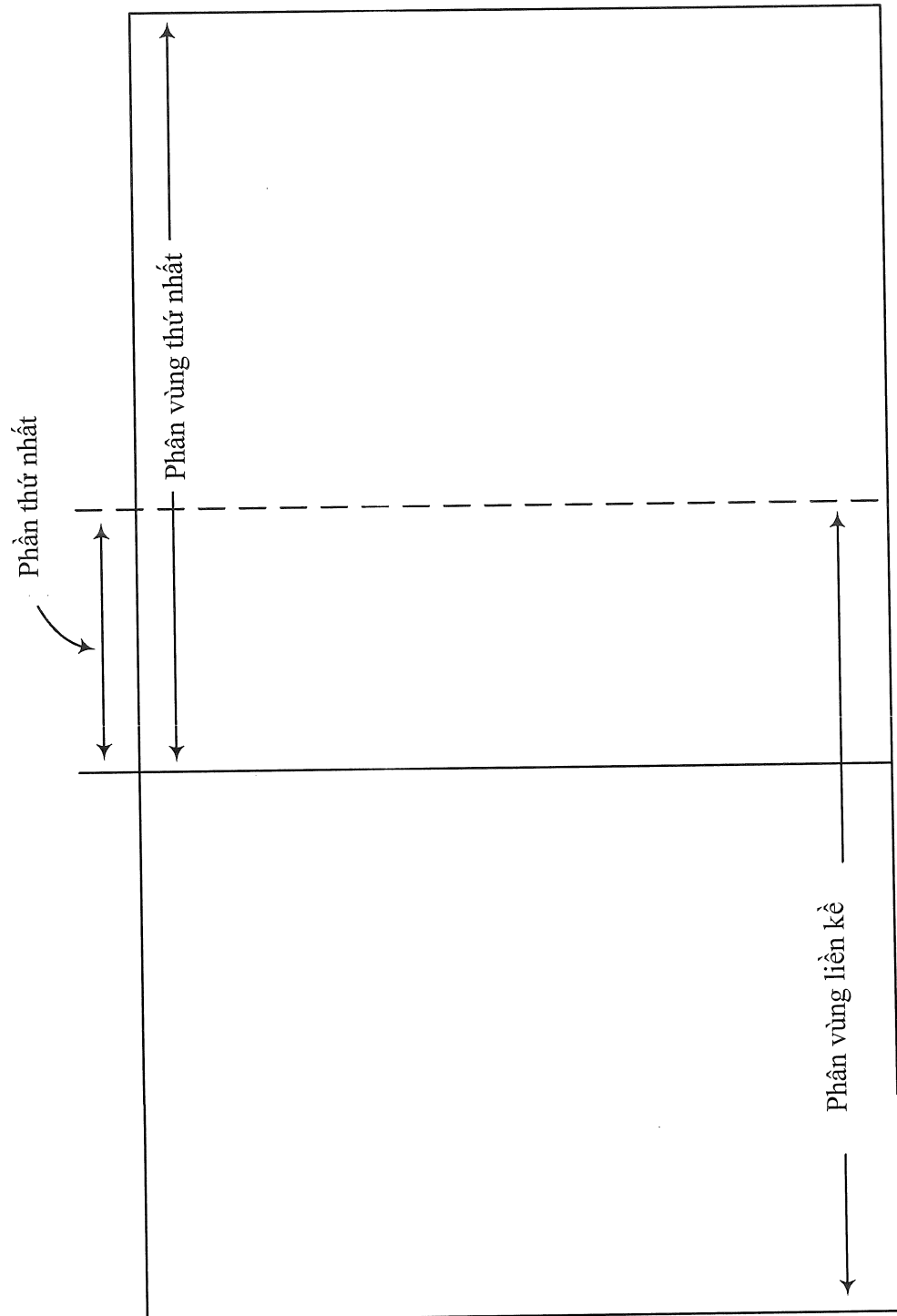
[Fig. 52A]



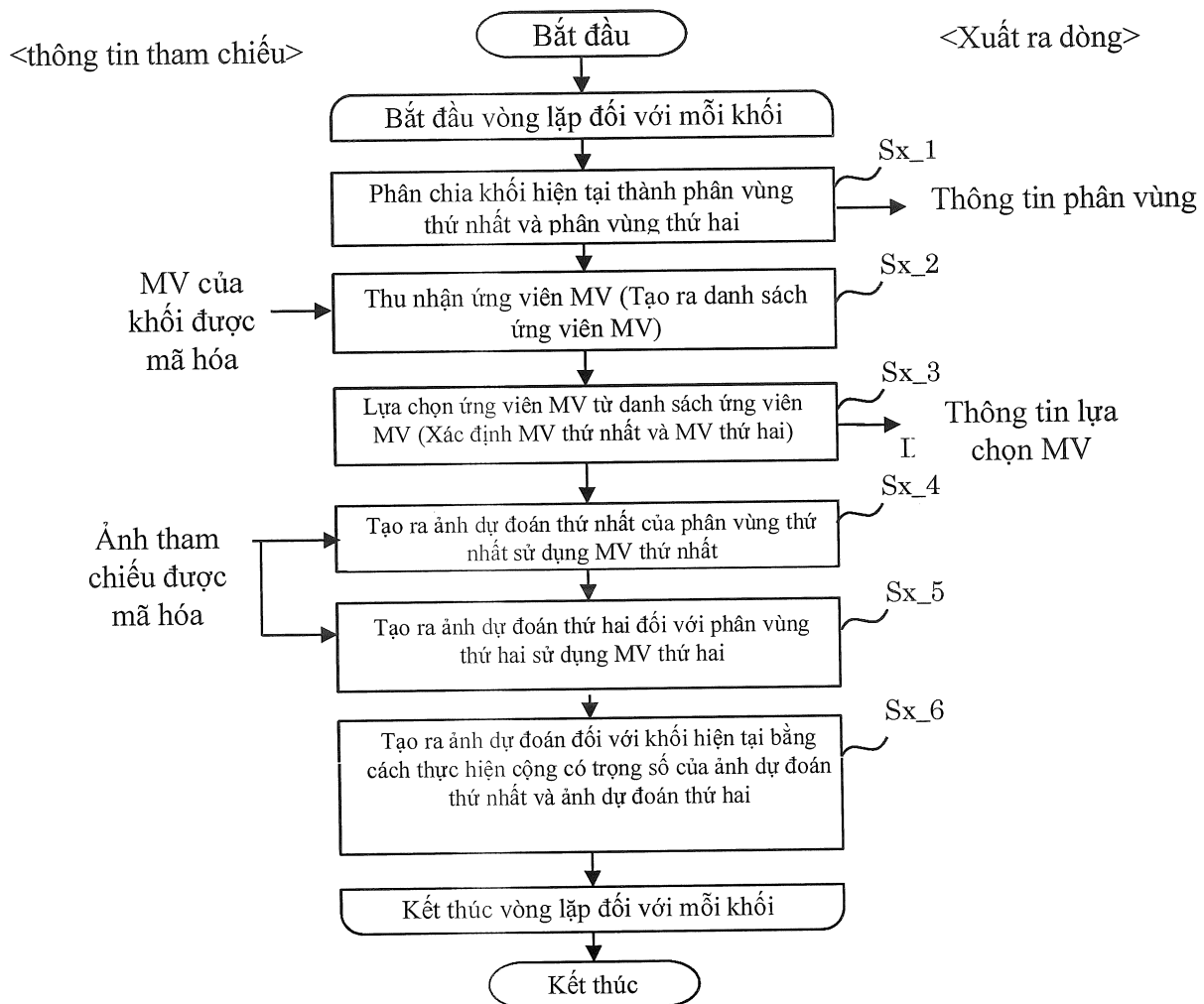
[Fig. 52B]



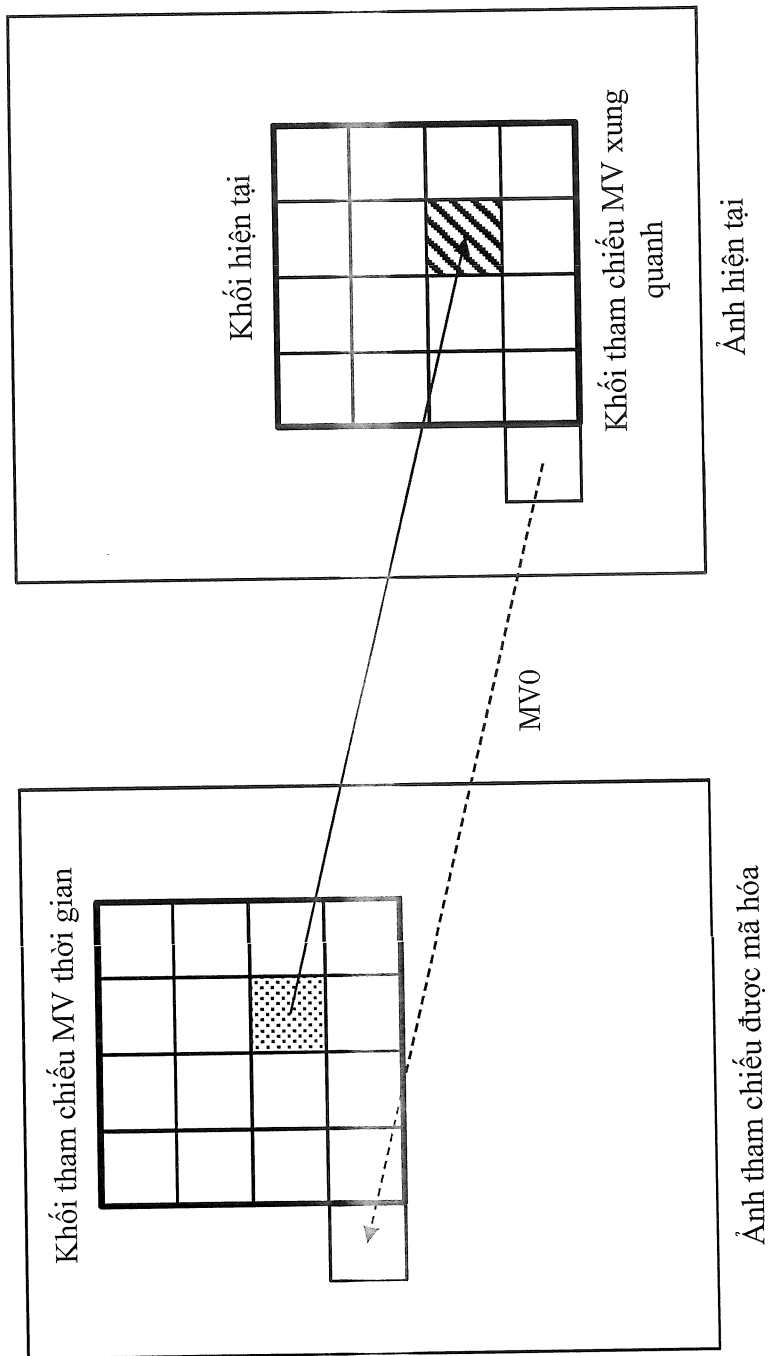
[Fig. 52C]



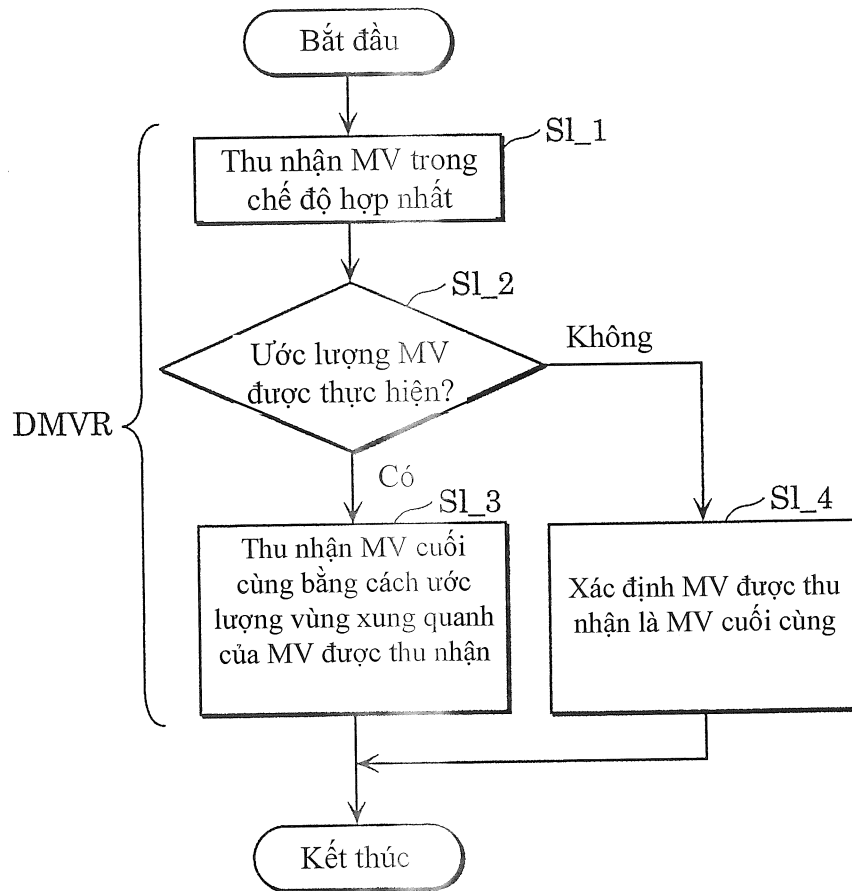
[Fig. 53]



[Fig. 54]

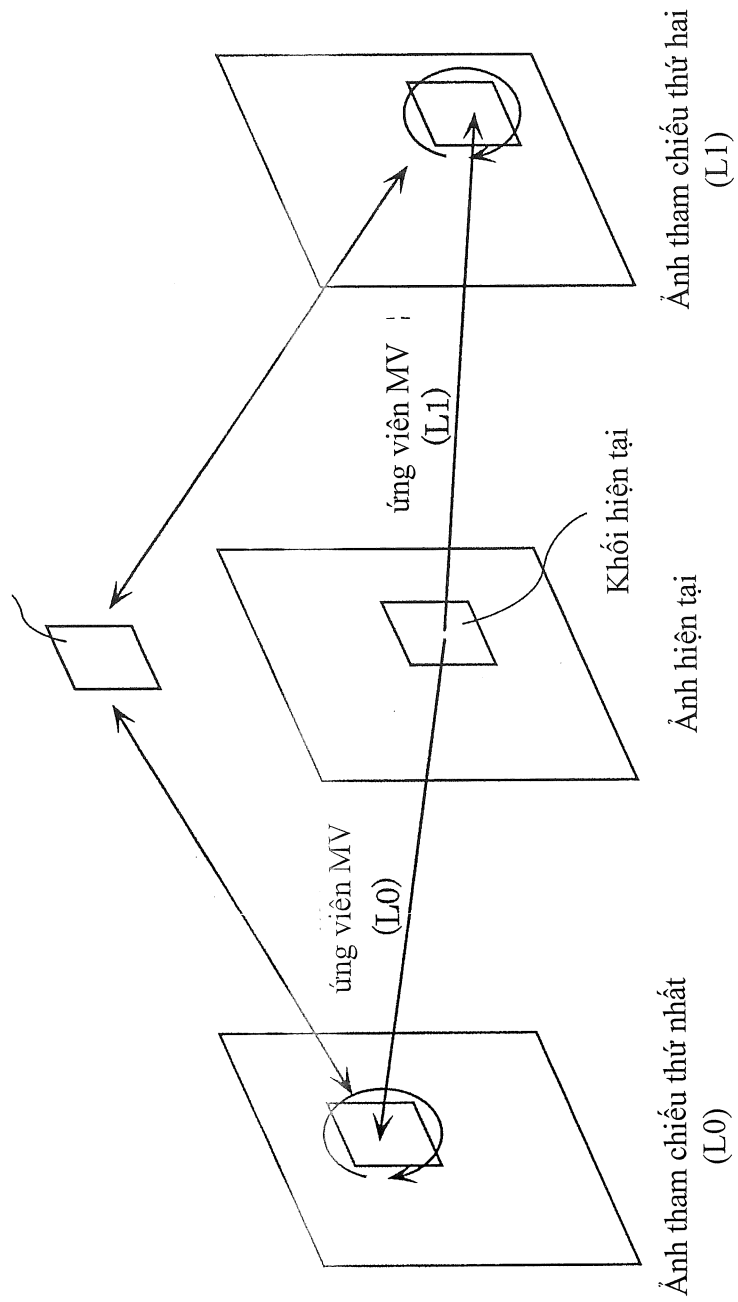


[Fig. 55]

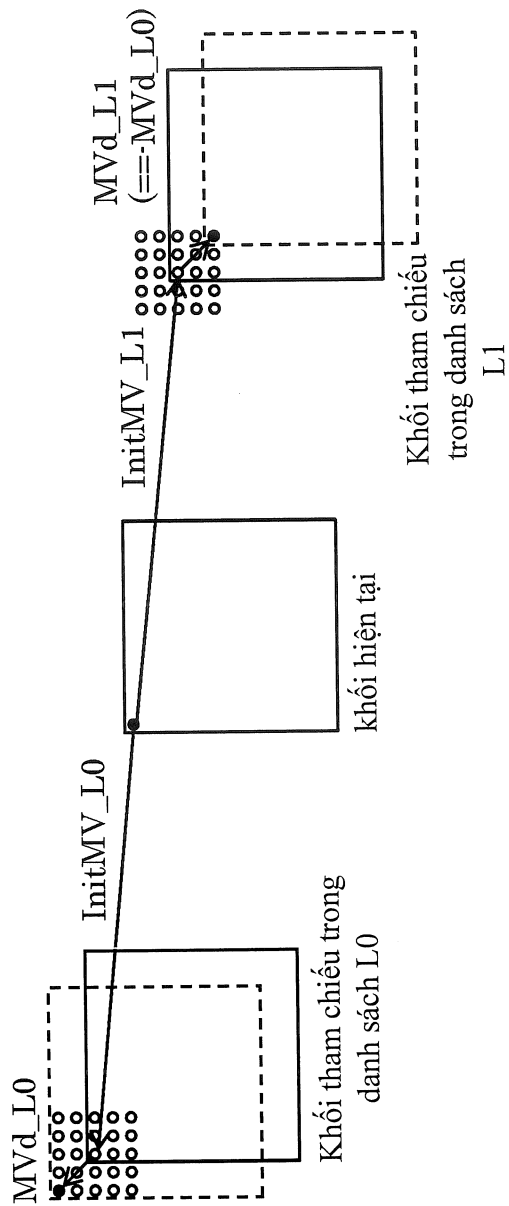


[Fig. 56]

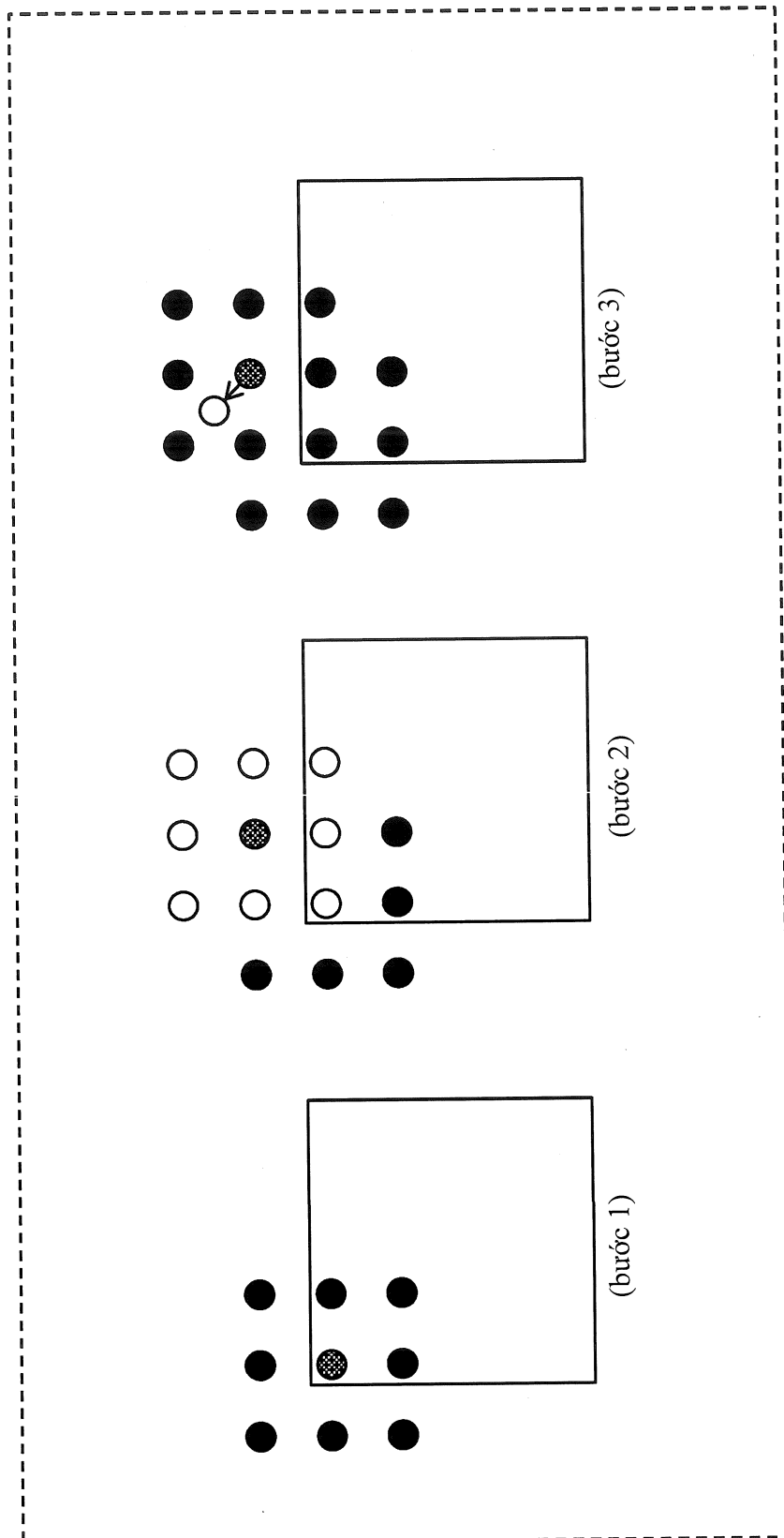
Mẫu được tạo ra dựa trên điểm ảnh tham chiếu của ứng viên MV (L0) và điểm ảnh tham chiếu của ứng viên MV (L1)



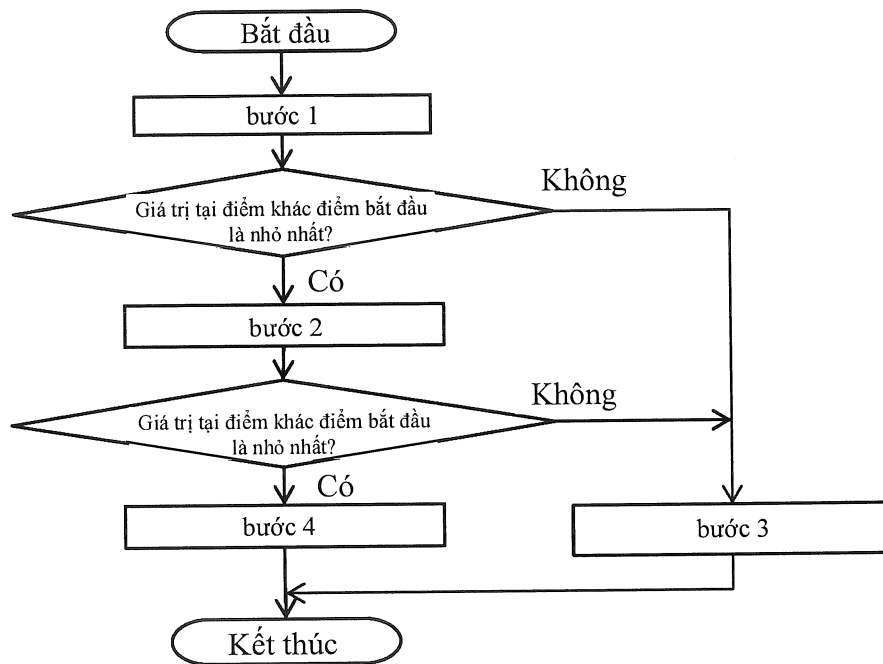
[Fig. 57]



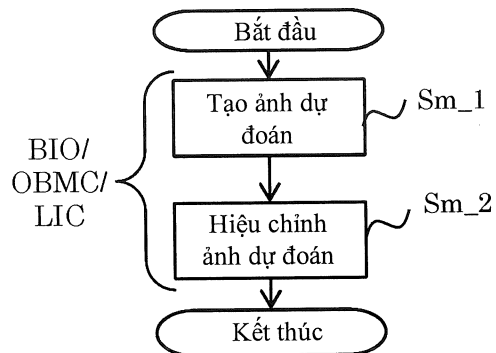
[Fig. 58A]



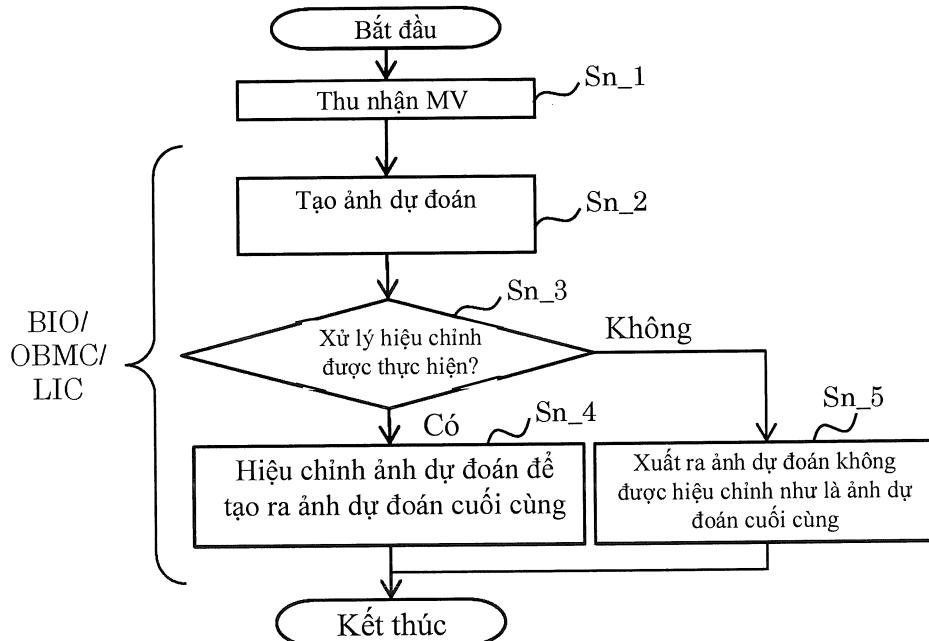
[Fig. 58B]



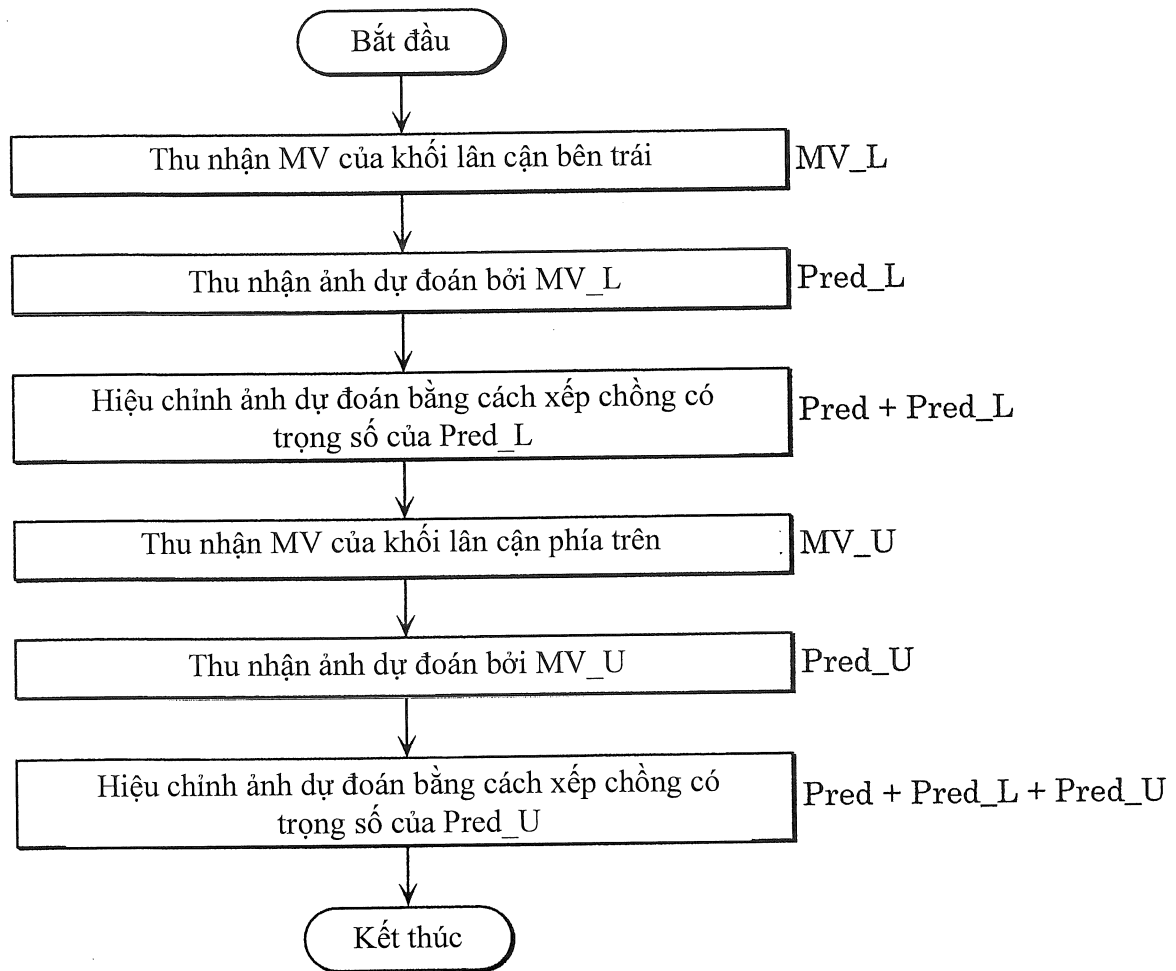
[Fig. 59]



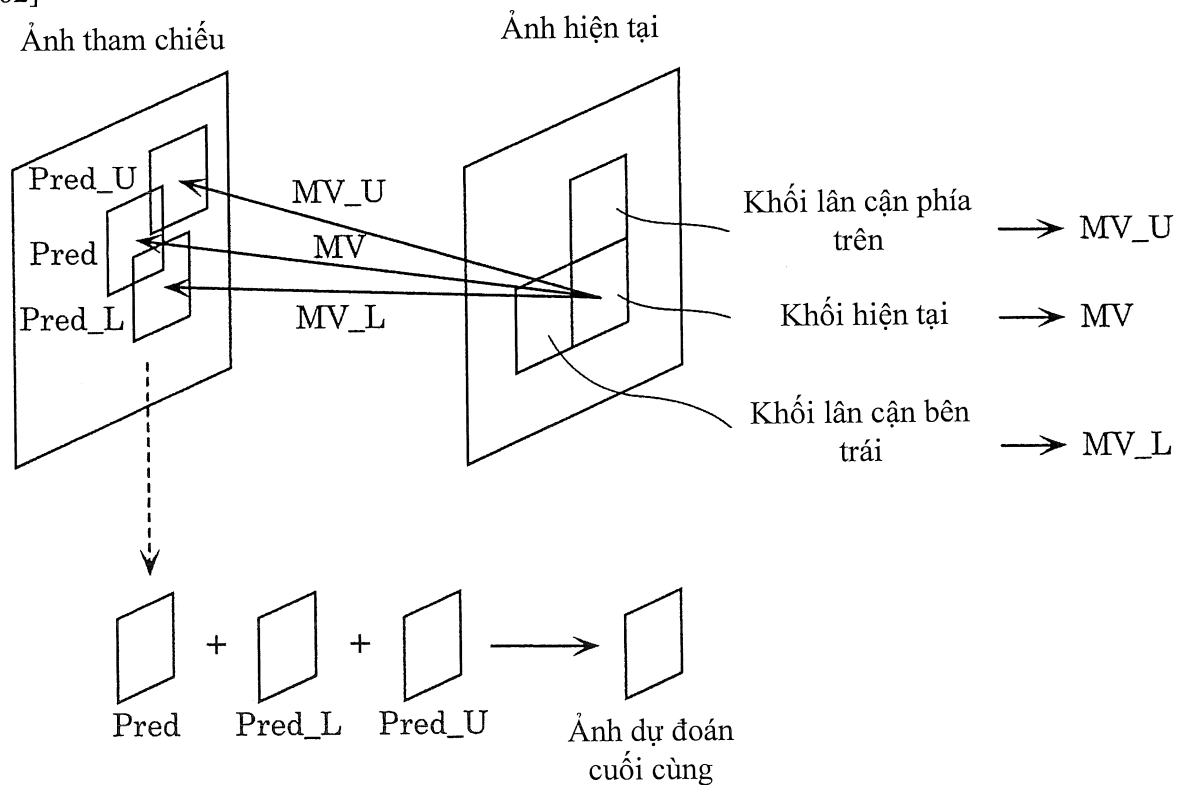
[Fig. 60]



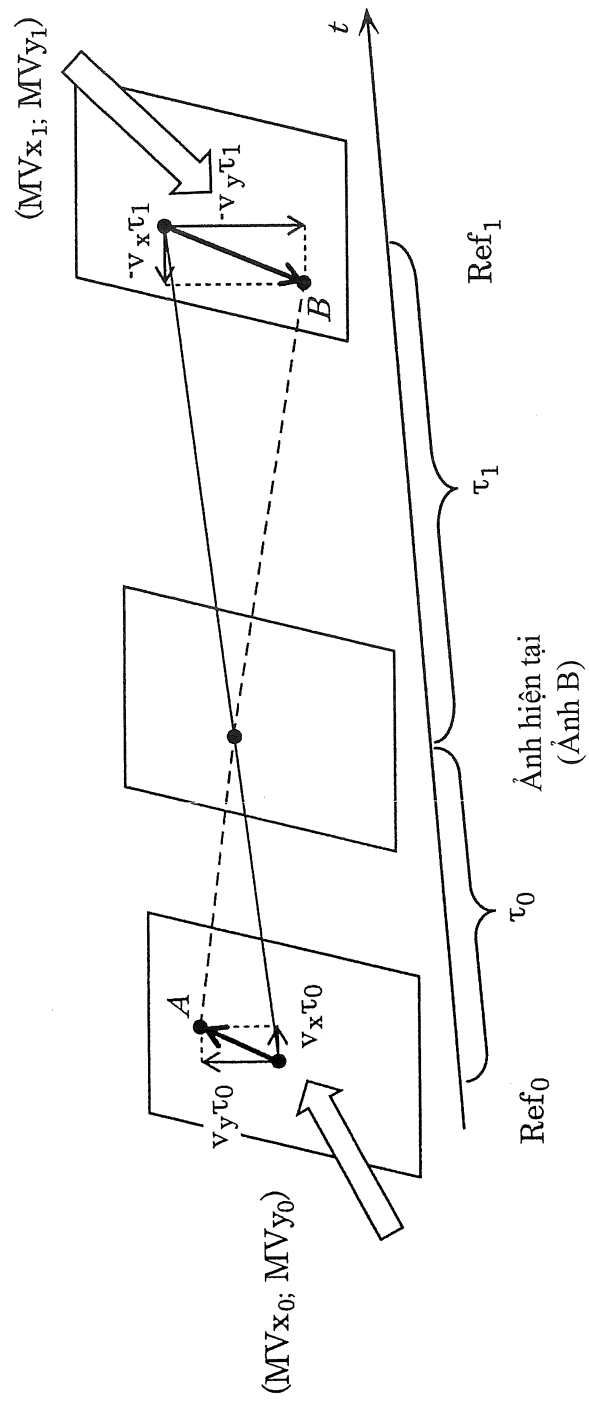
[Fig. 61]



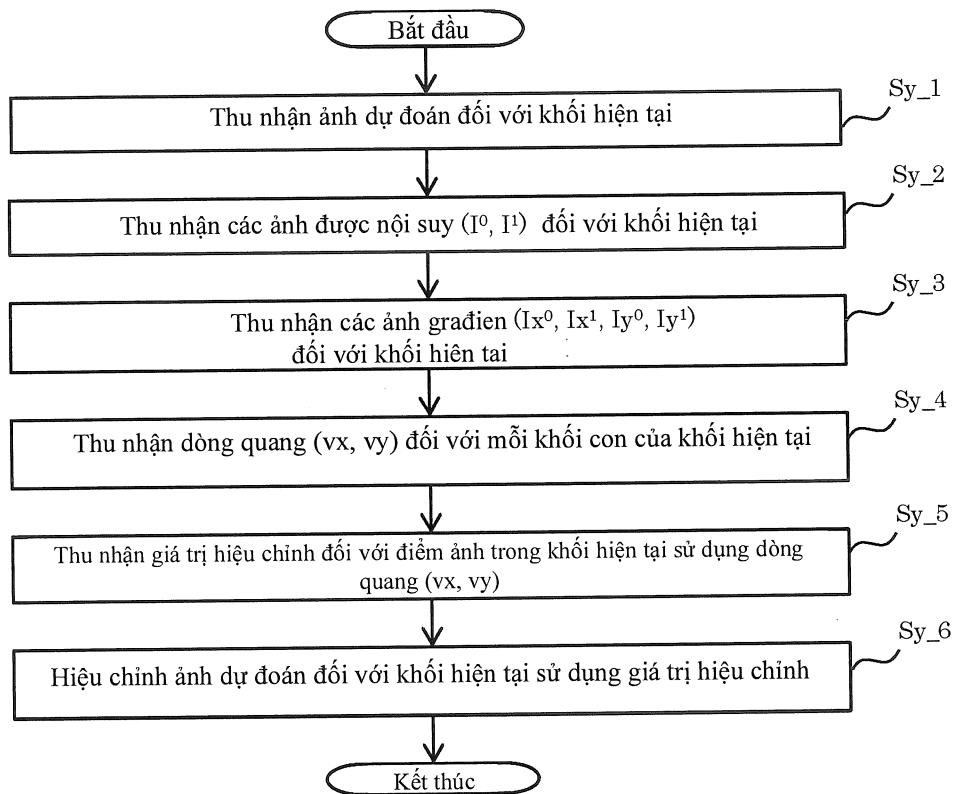
[Fig. 62]



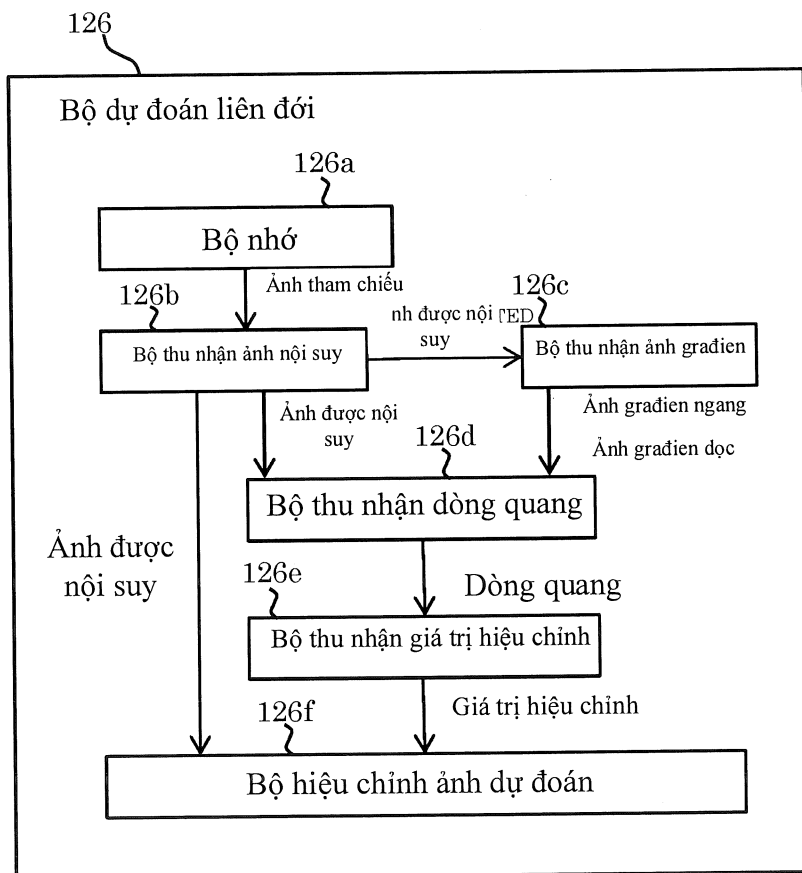
[Fig. 63]



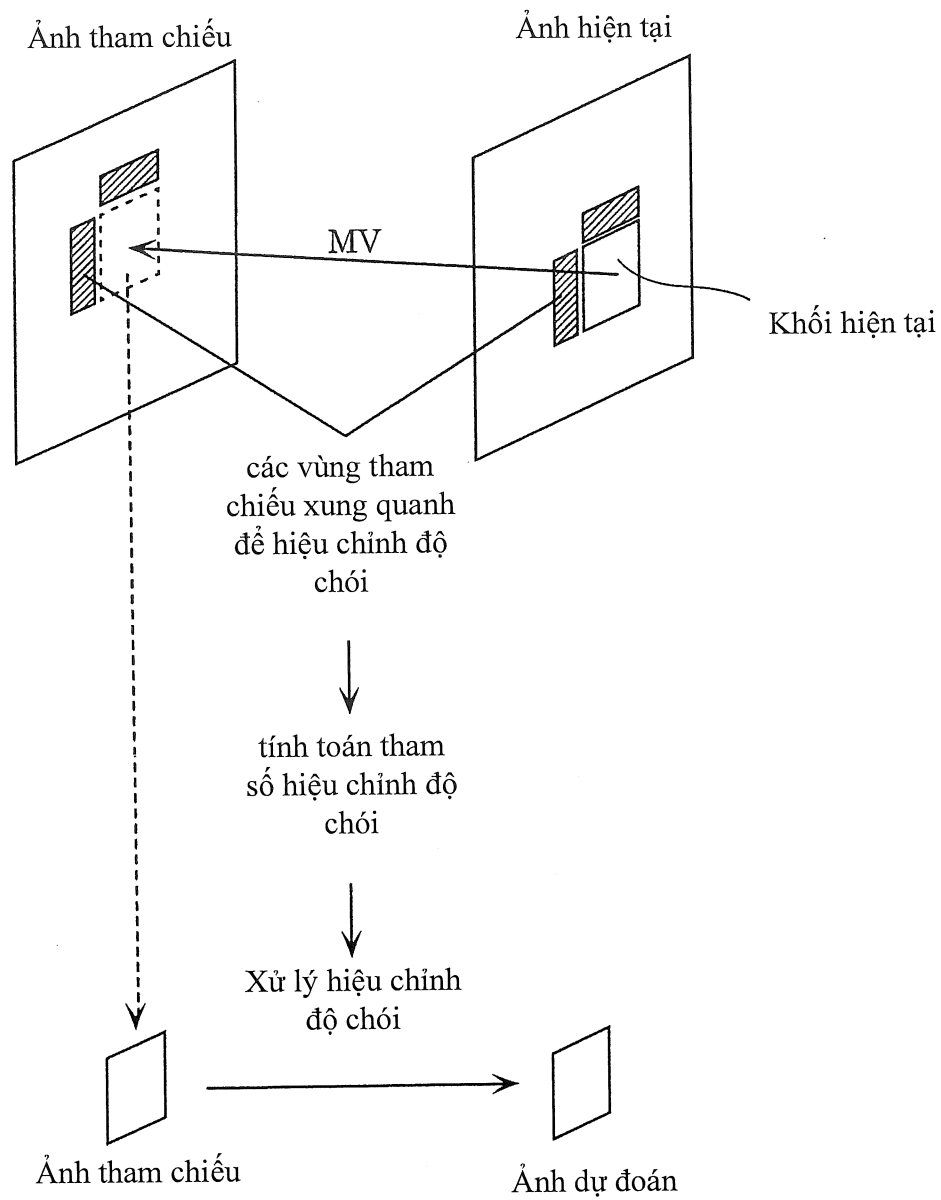
[Fig. 64]



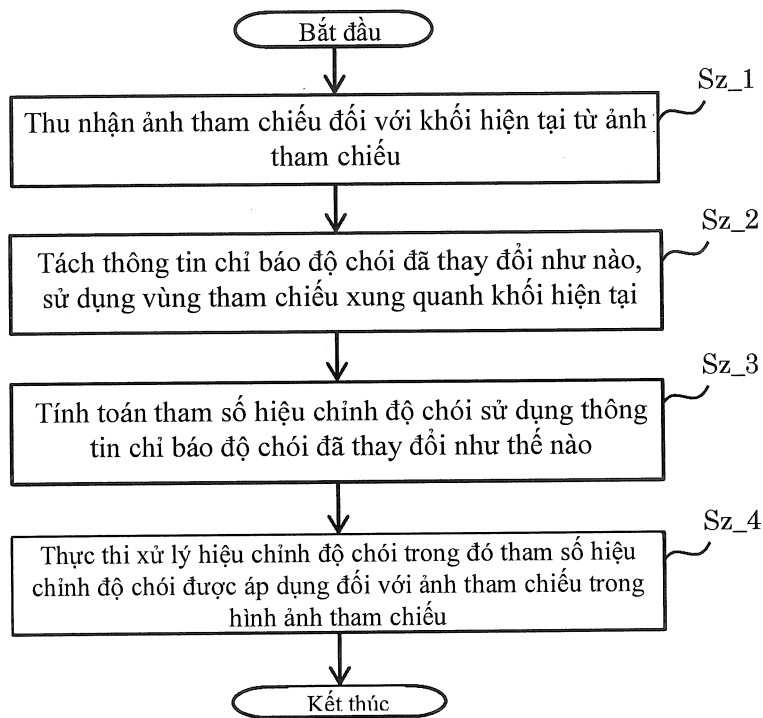
[Fig. 65]



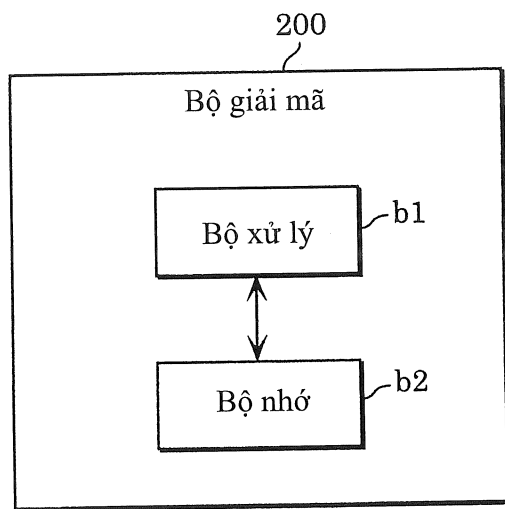
[Fig. 66A]



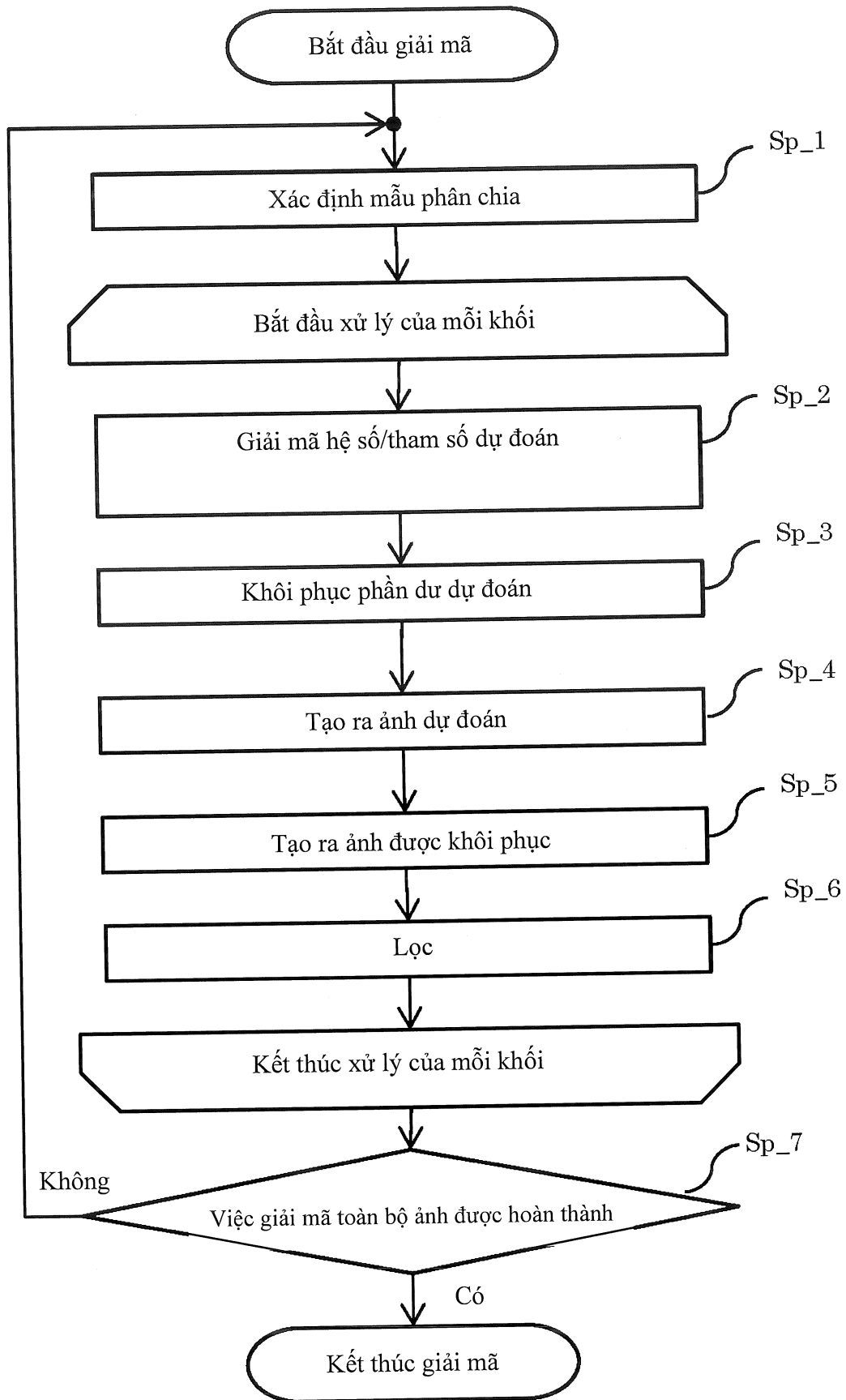
[Fig. 66B]



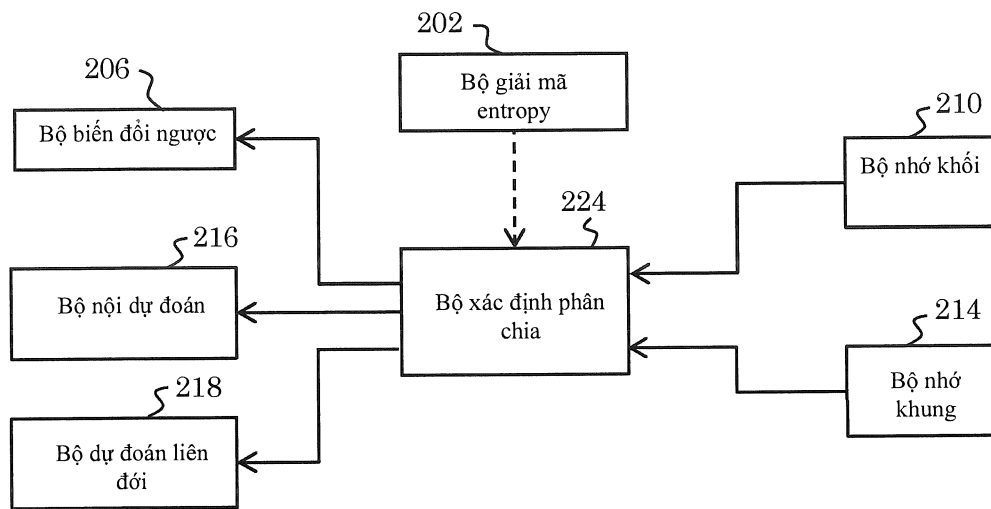
[Fig. 68]



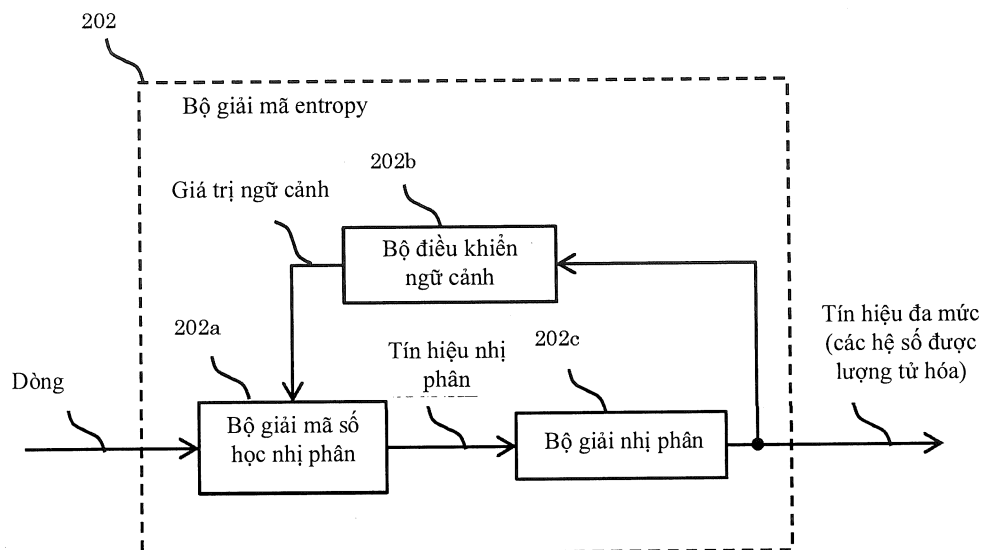
[Fig. 69]



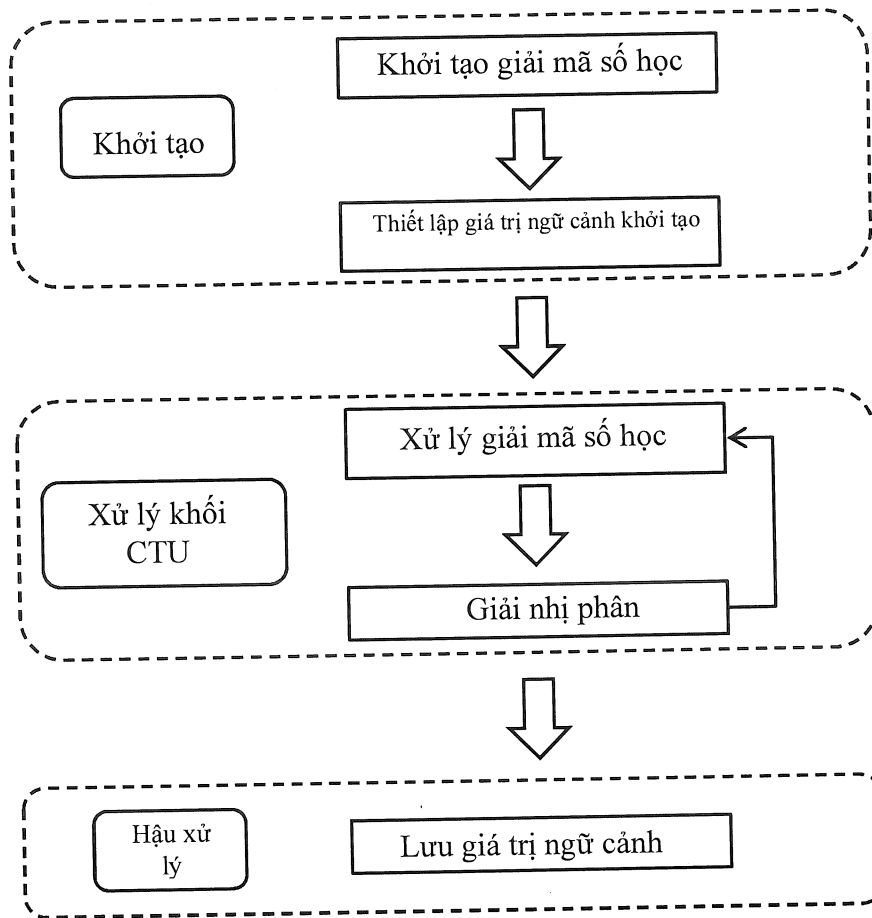
[Fig. 70]



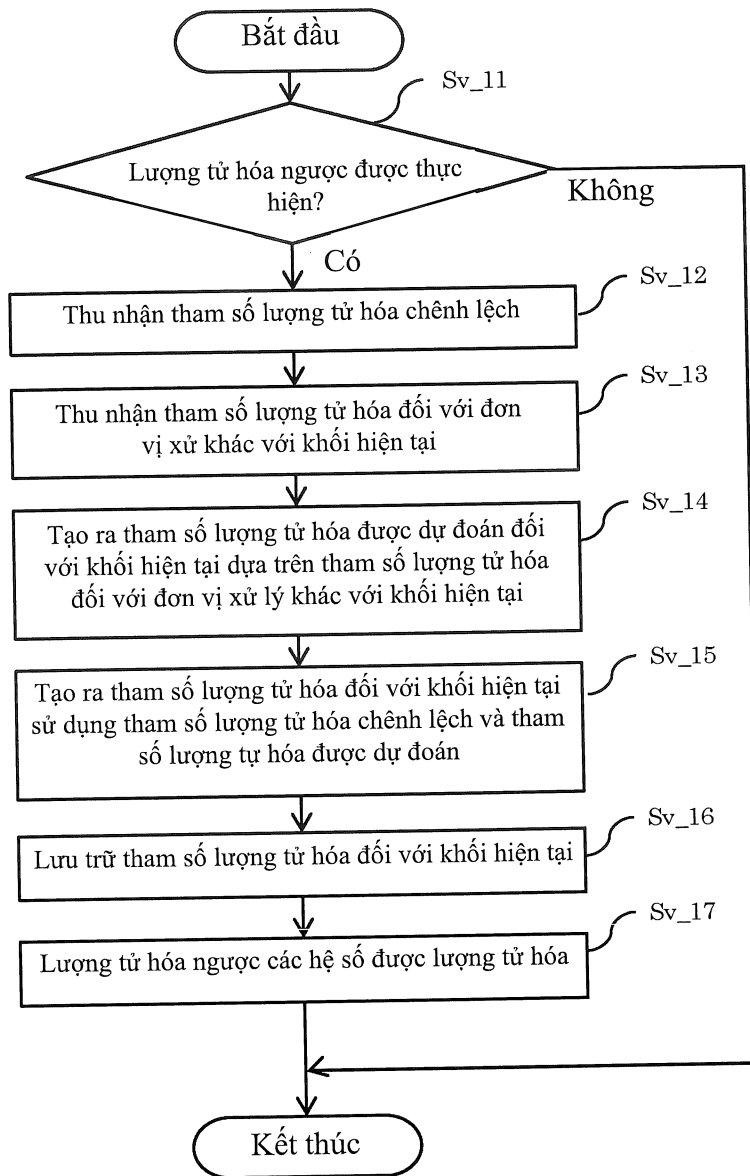
[Fig. 71]



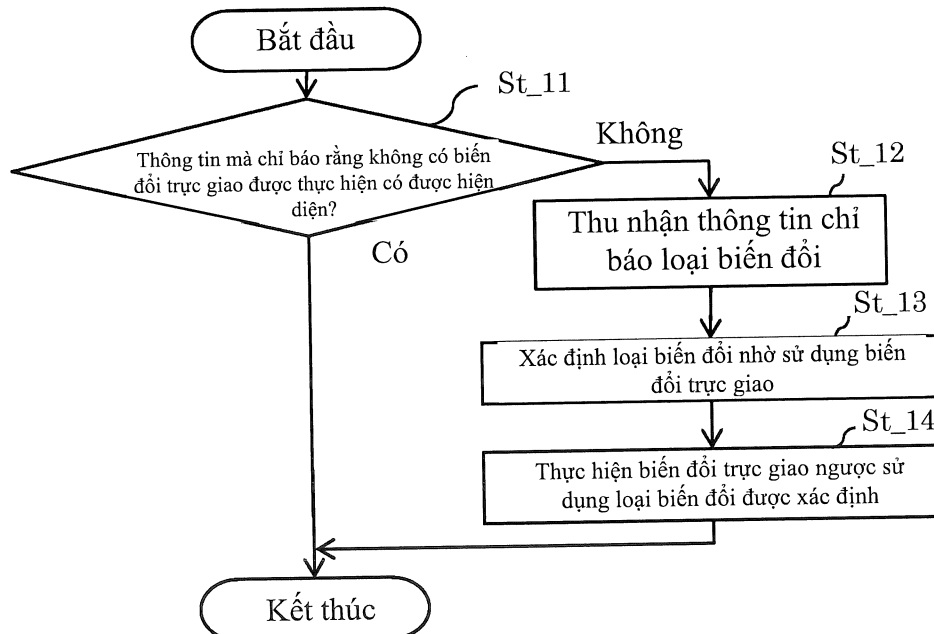
[Fig. 72]



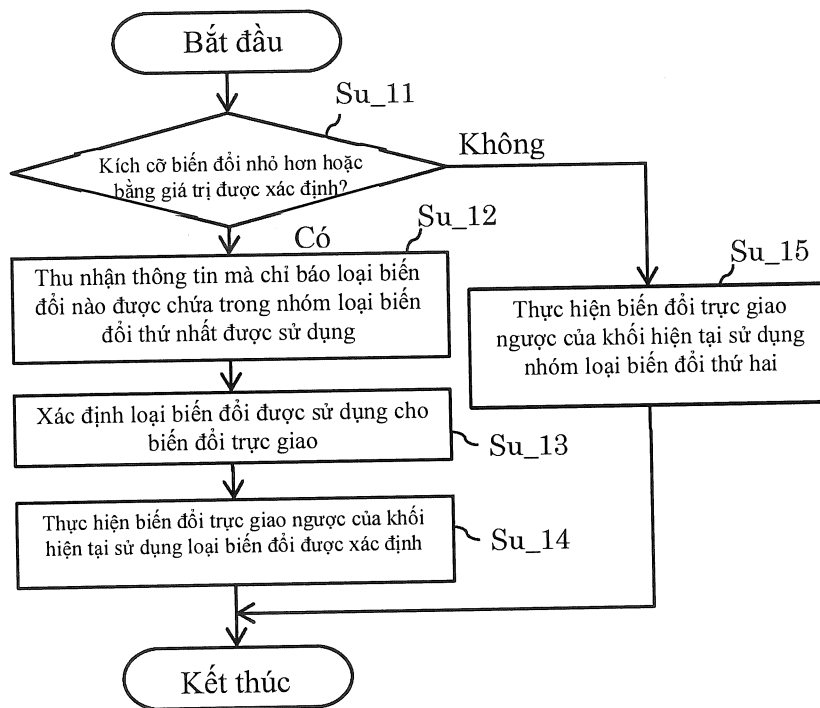
[Fig. 74]



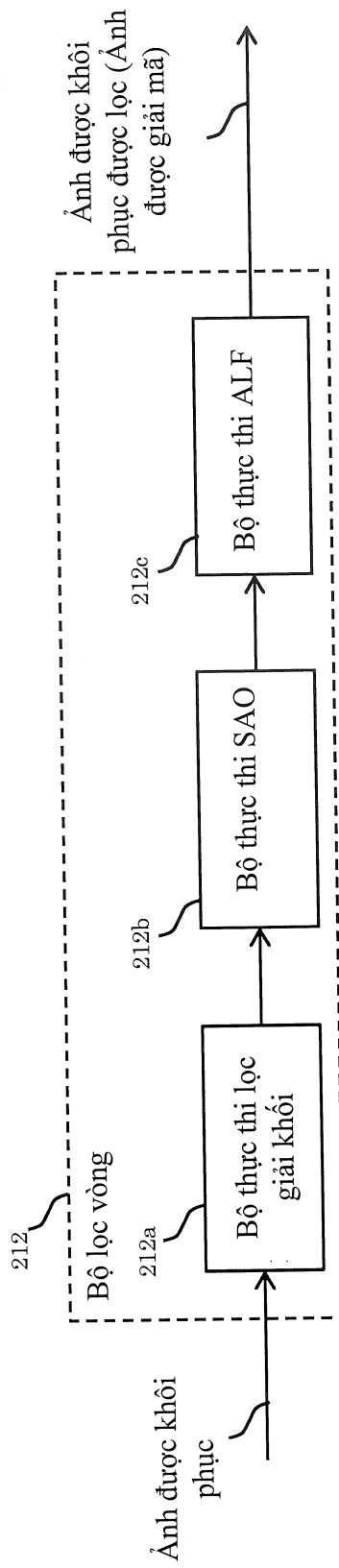
[Fig. 75]



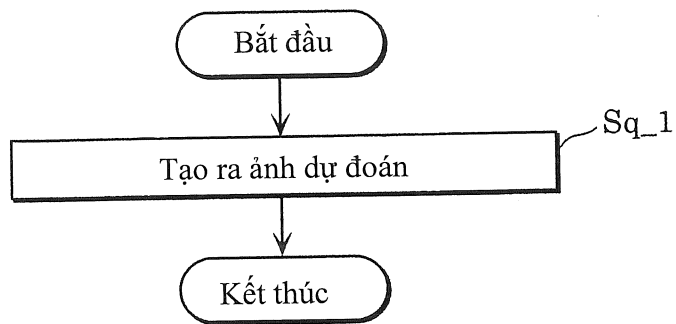
[Fig. 76]



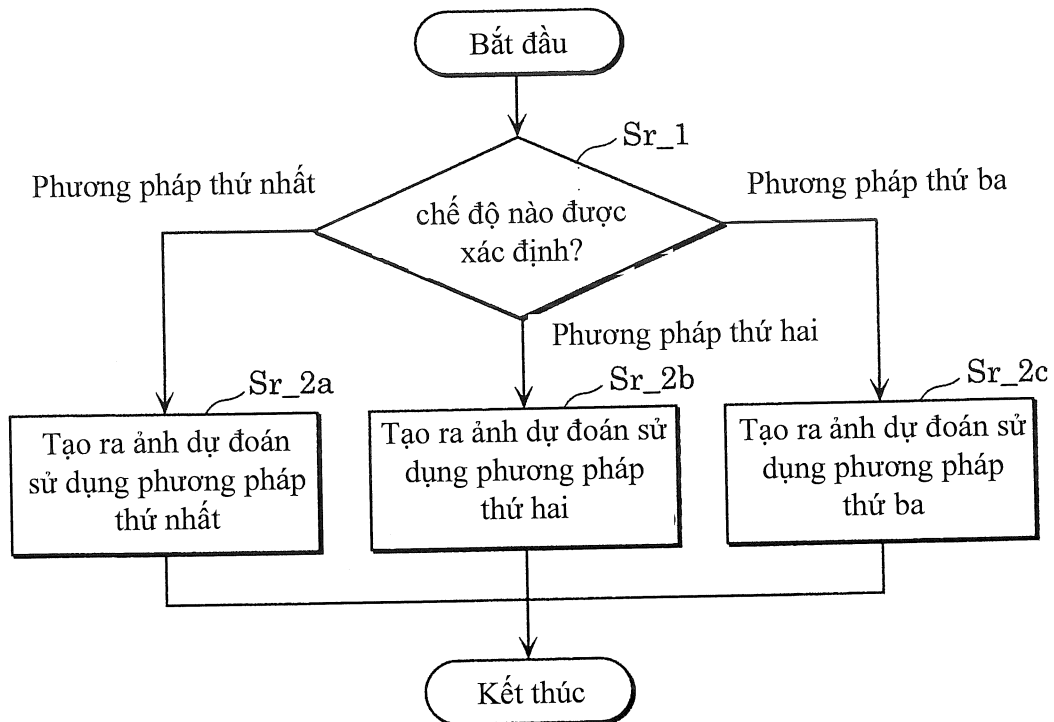
[Fig. 77]



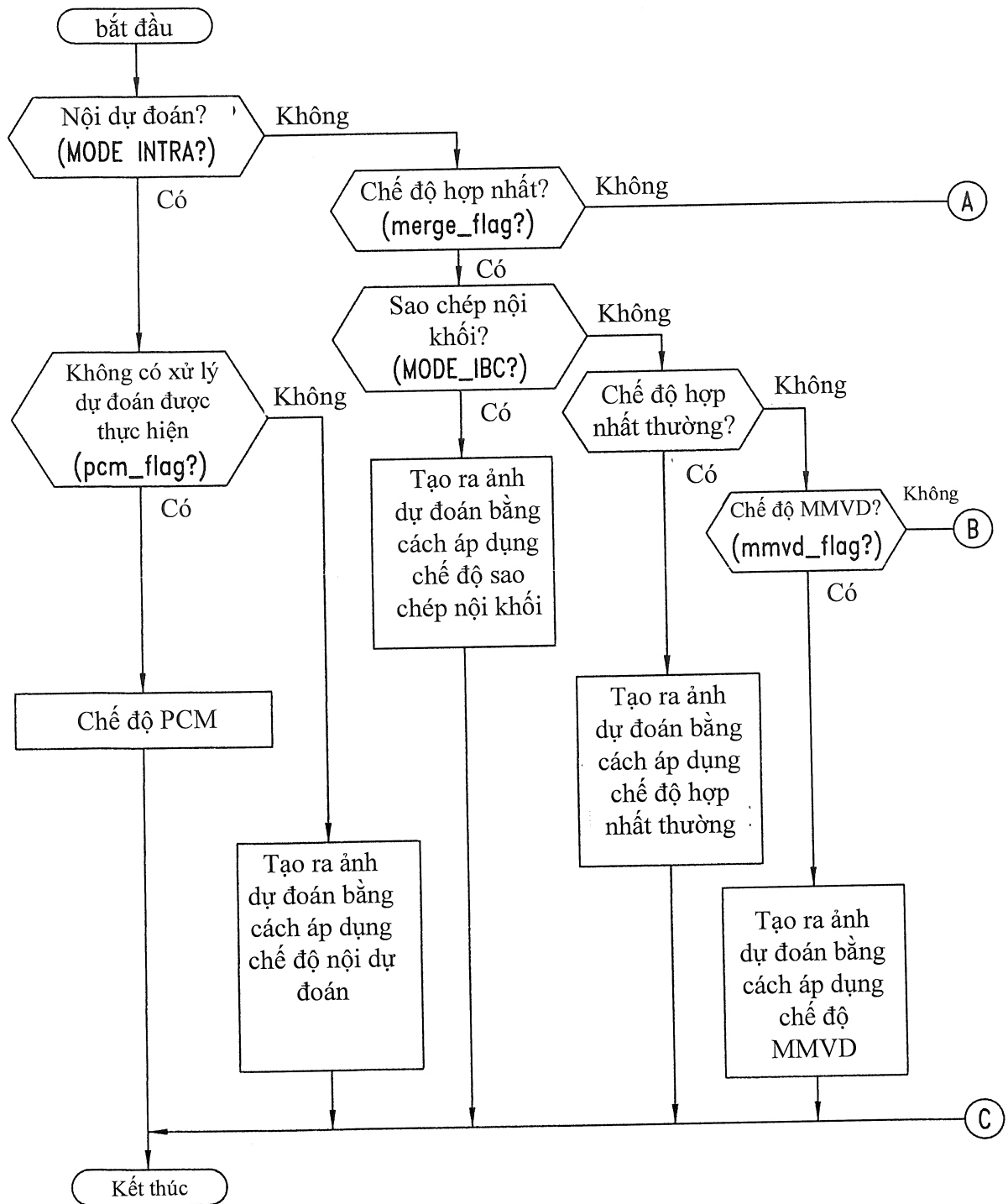
[Fig. 78]



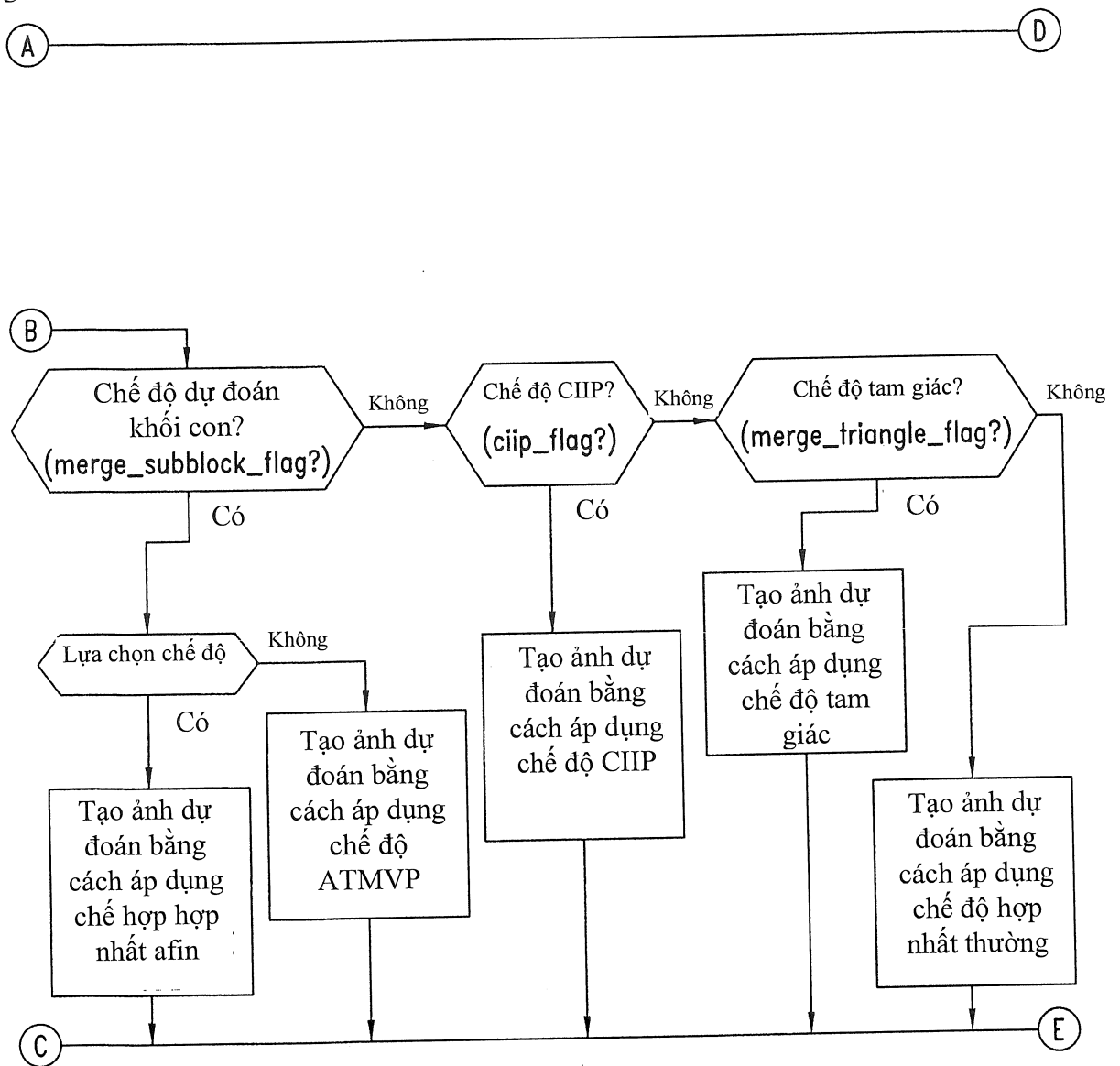
[Fig. 79]



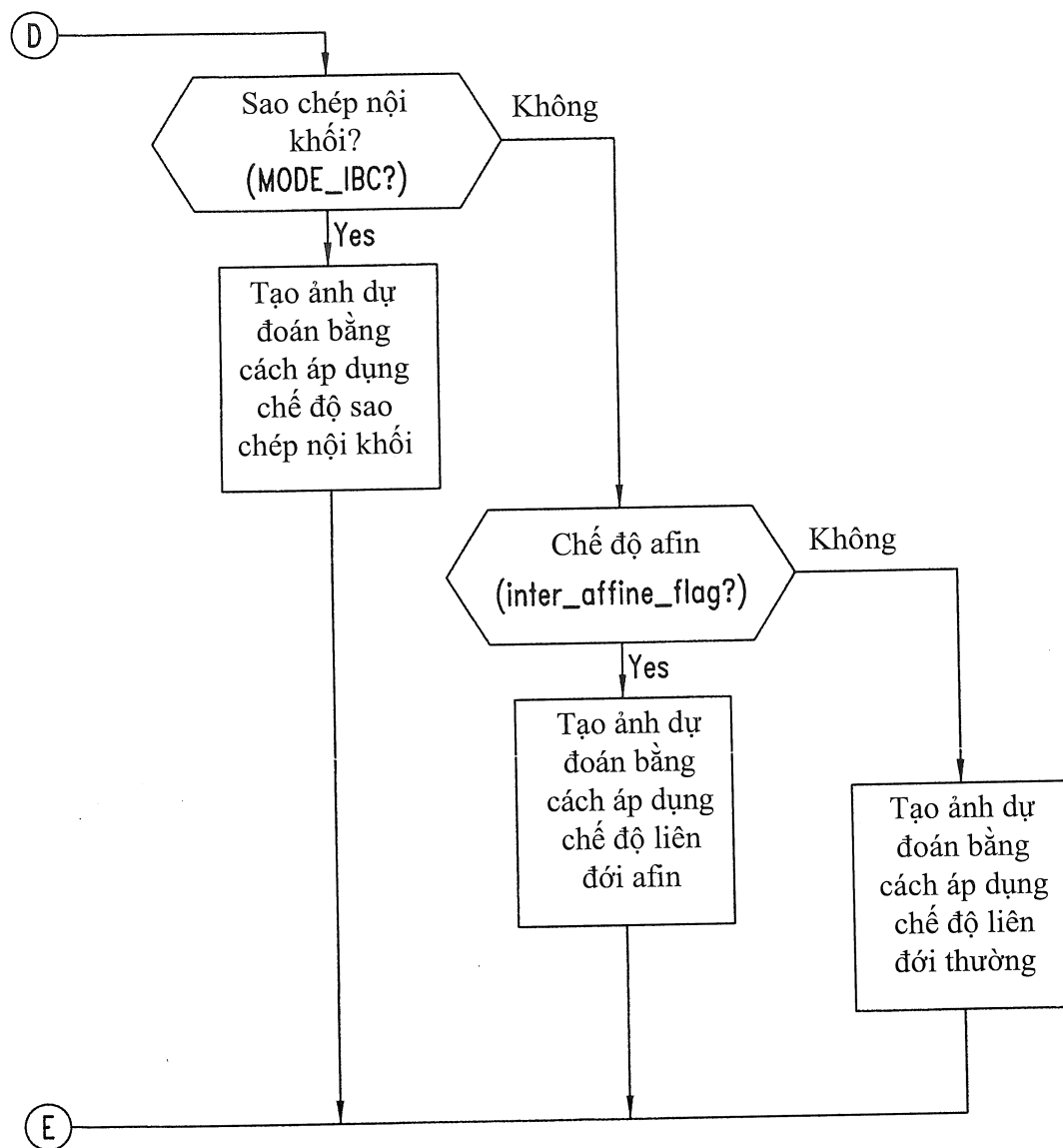
[Fig. 80A]



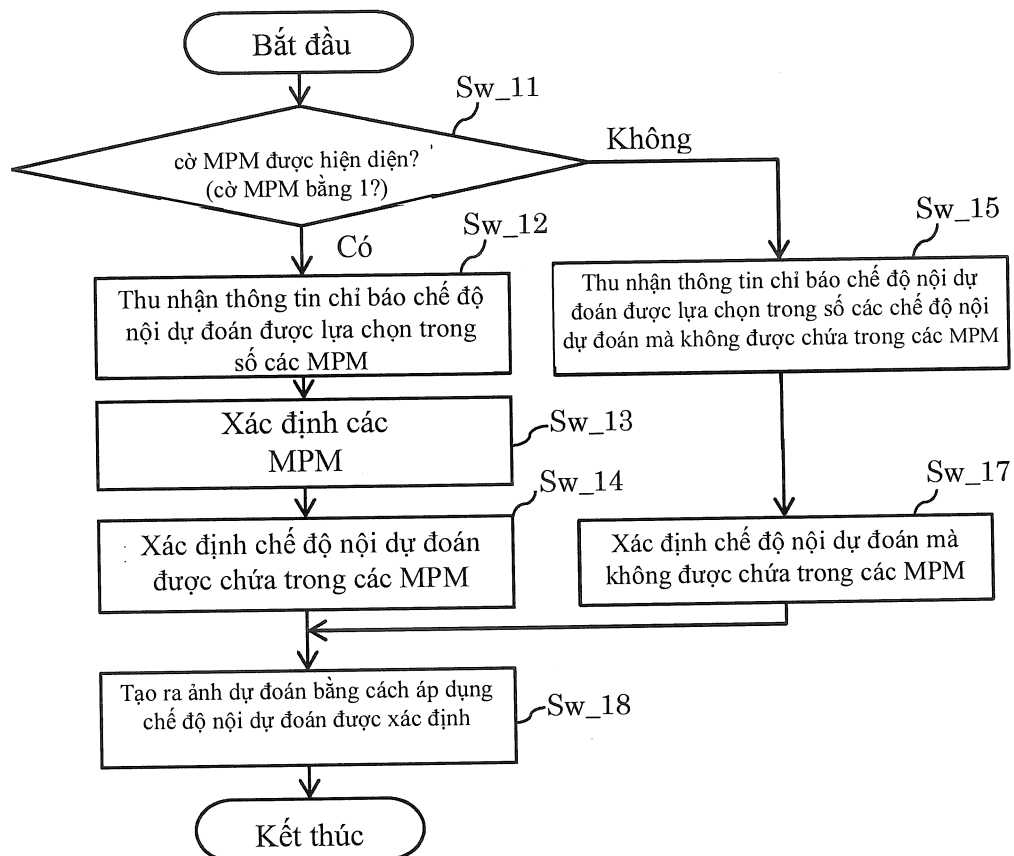
[Fig. 80B]



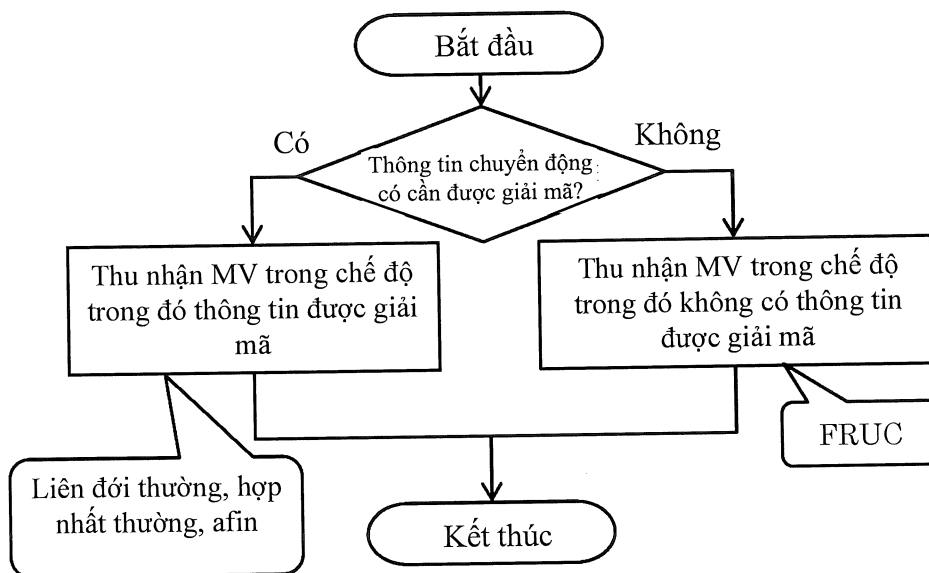
[Fig. 80C]



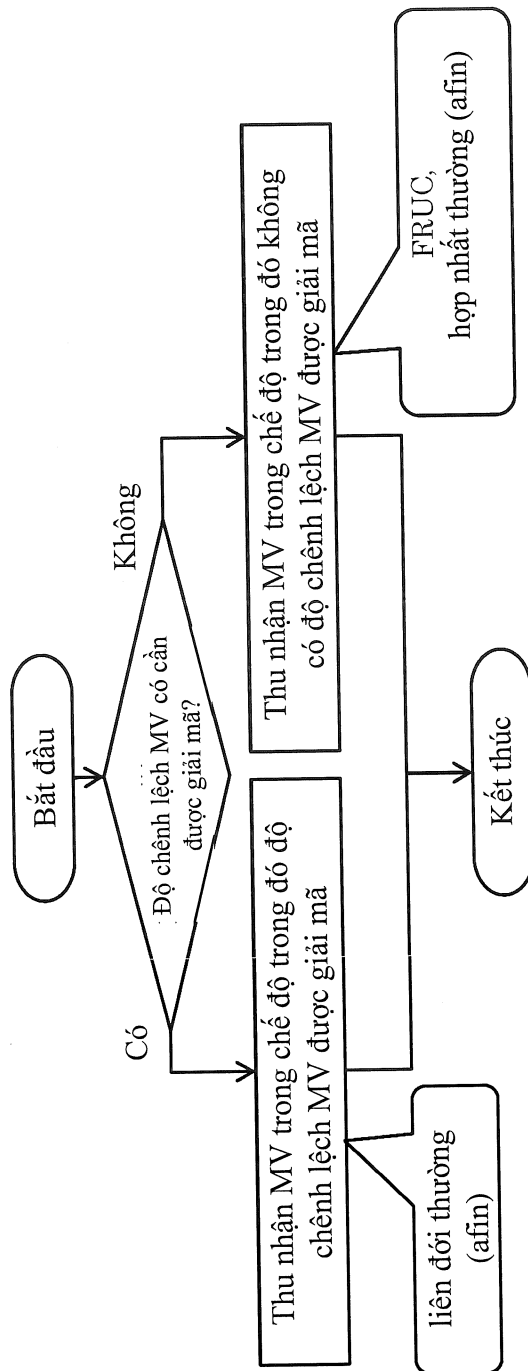
[Fig. 81]



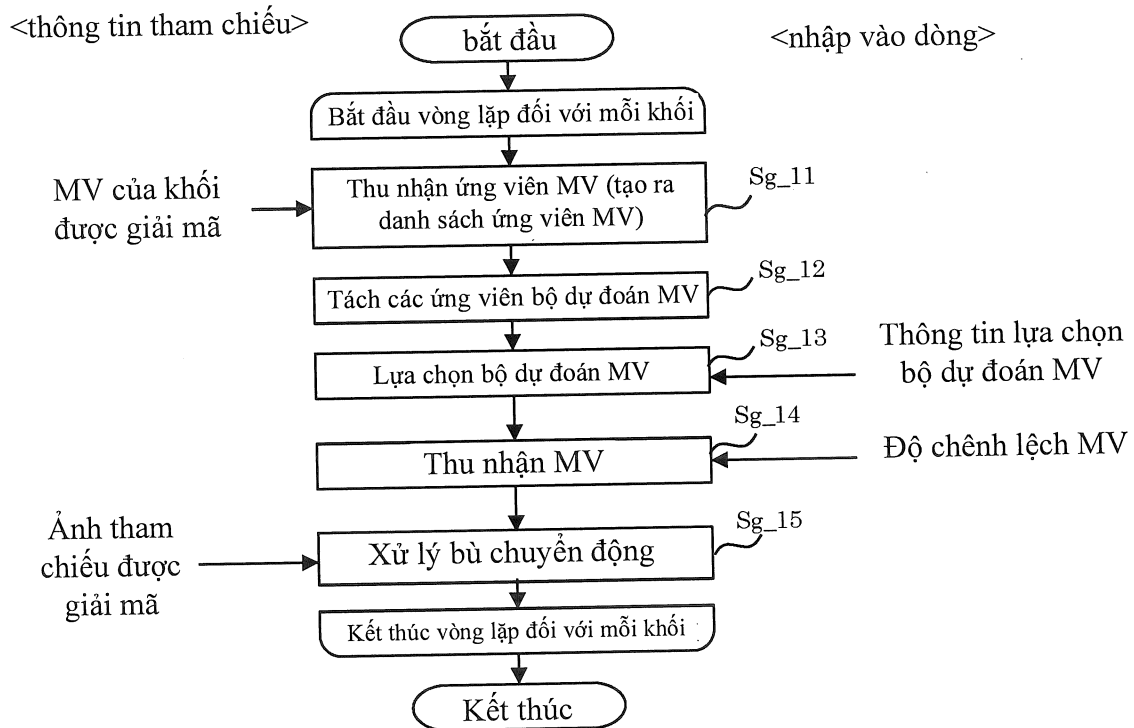
[Fig. 82]



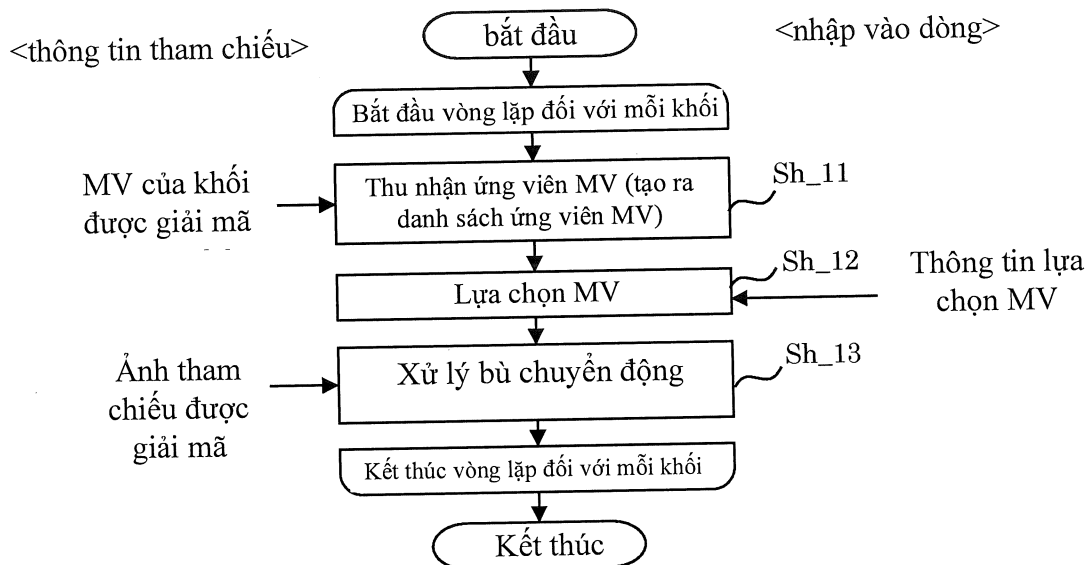
[Fig. 83]



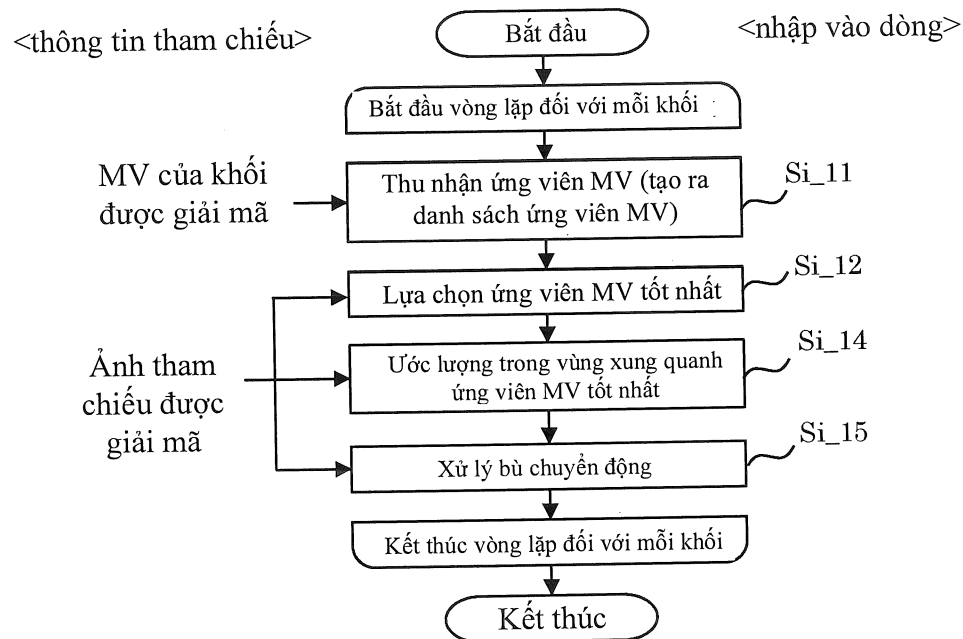
[Fig. 84]



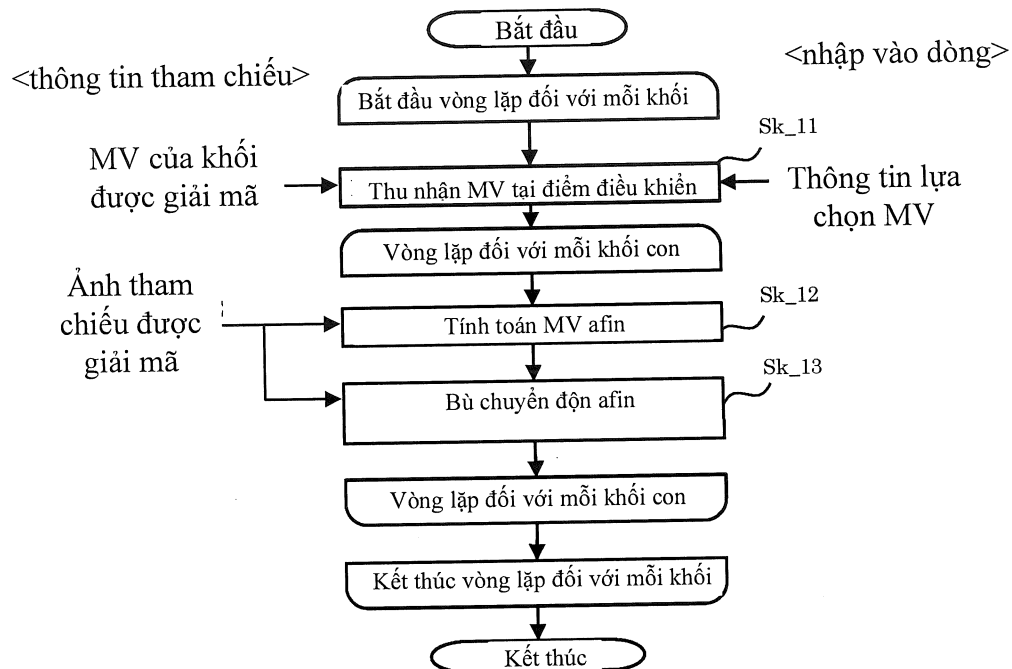
[Fig. 85]



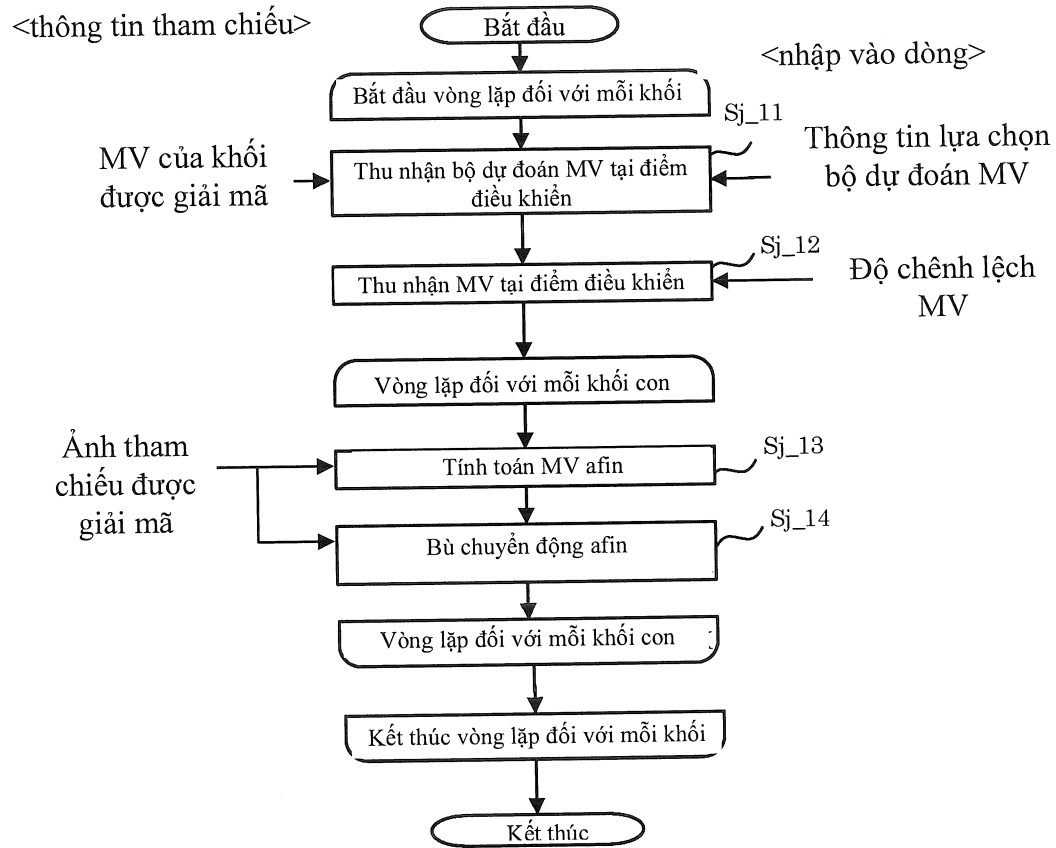
[Fig. 86]



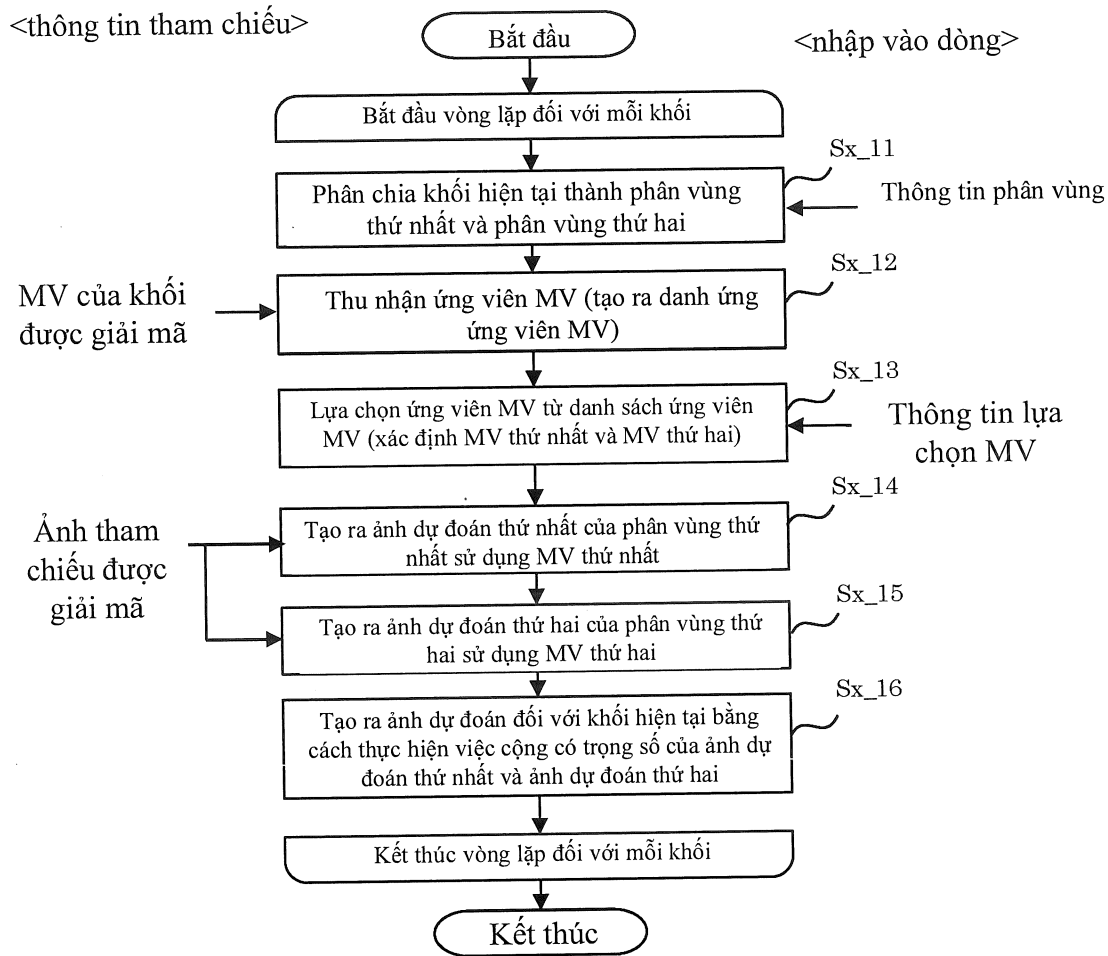
[Fig. 87]



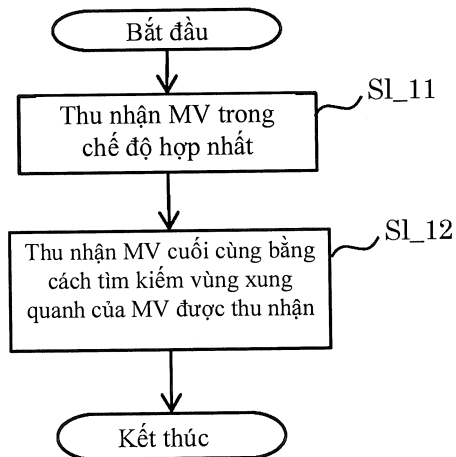
[Fig. 88]



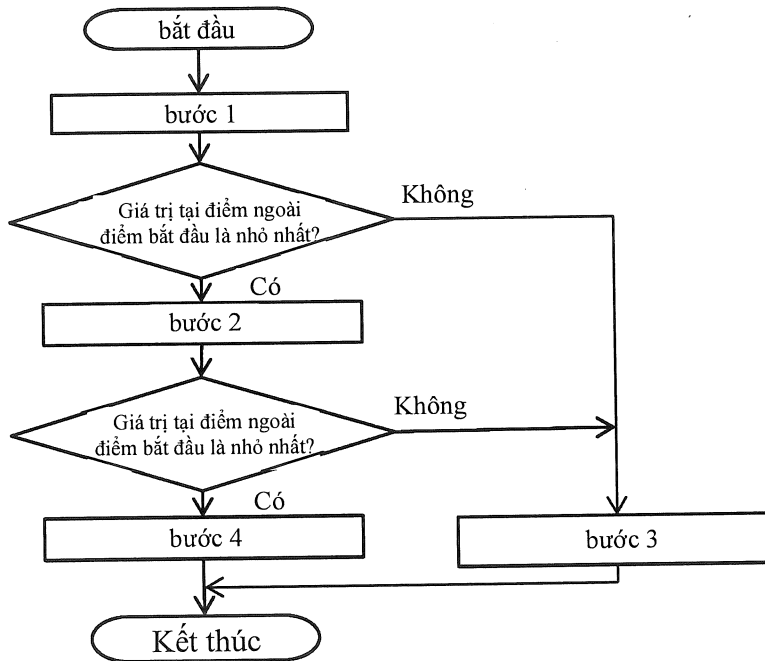
[Fig. 89]



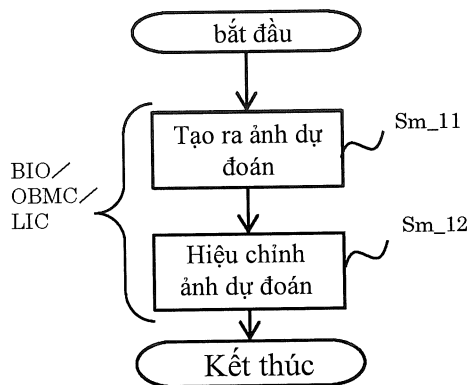
[Fig. 90]



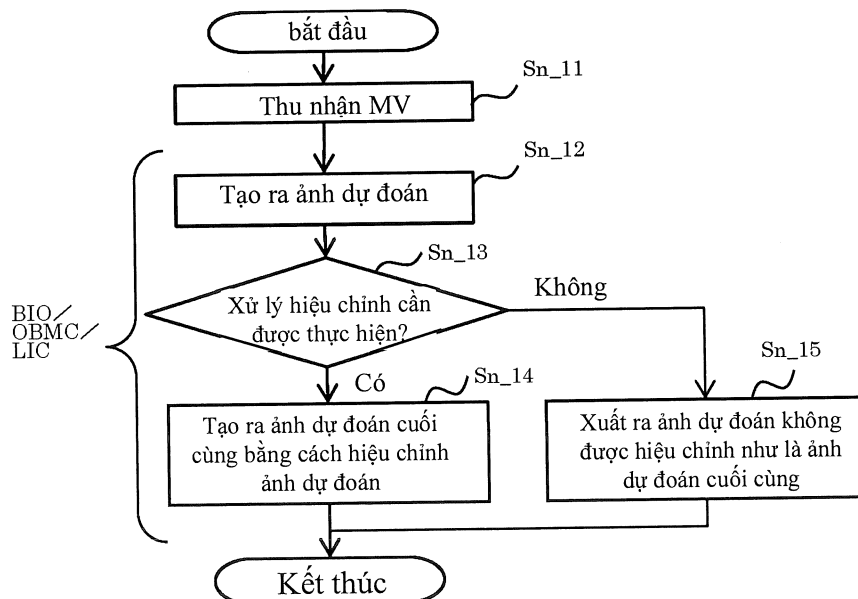
[Fig. 91]



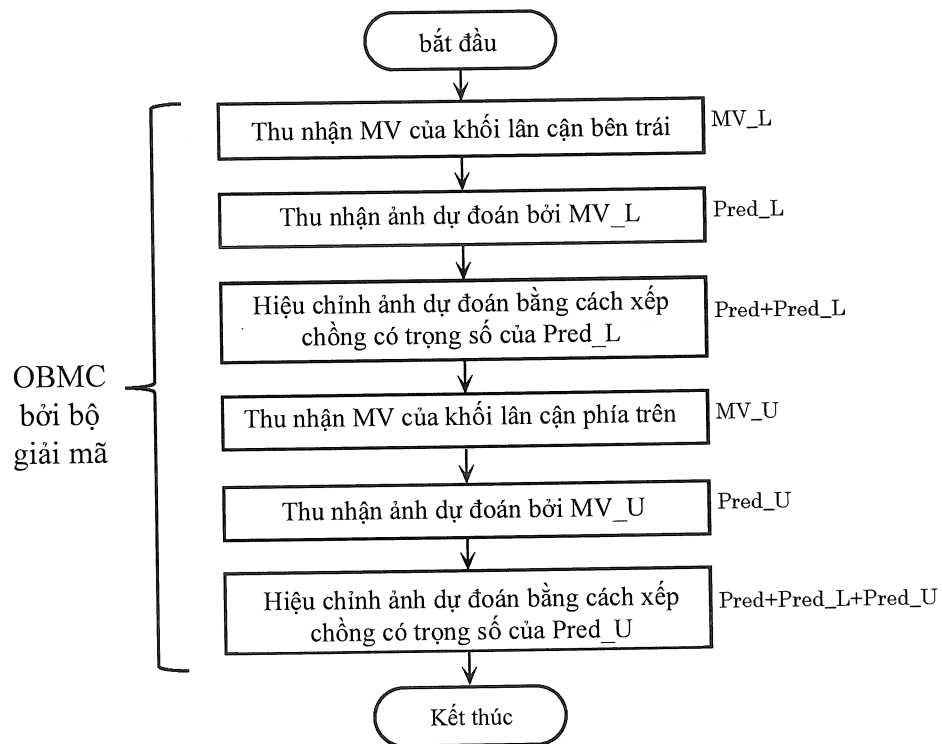
[Fig. 92]



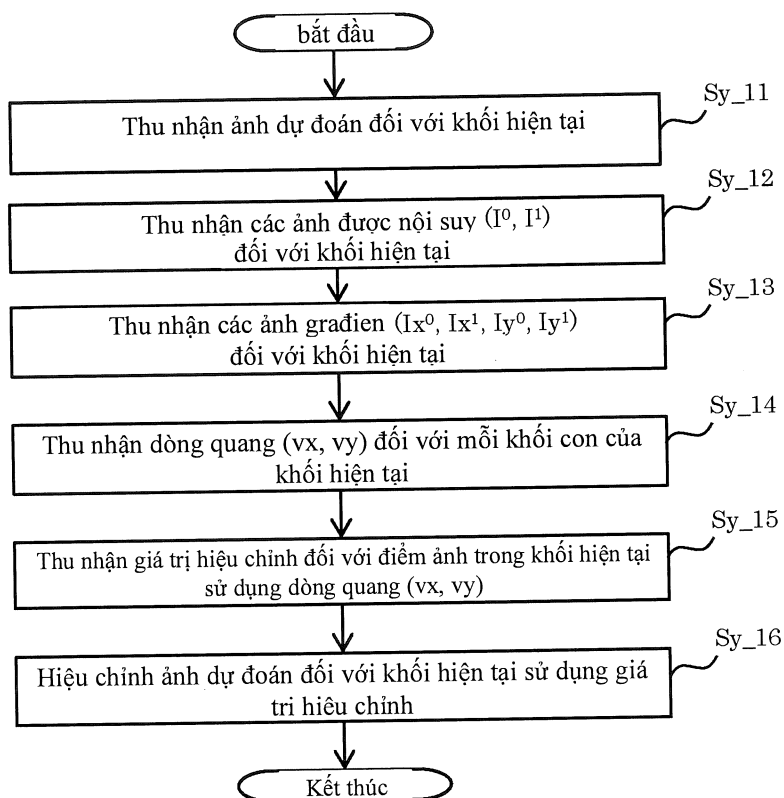
[Fig. 93]



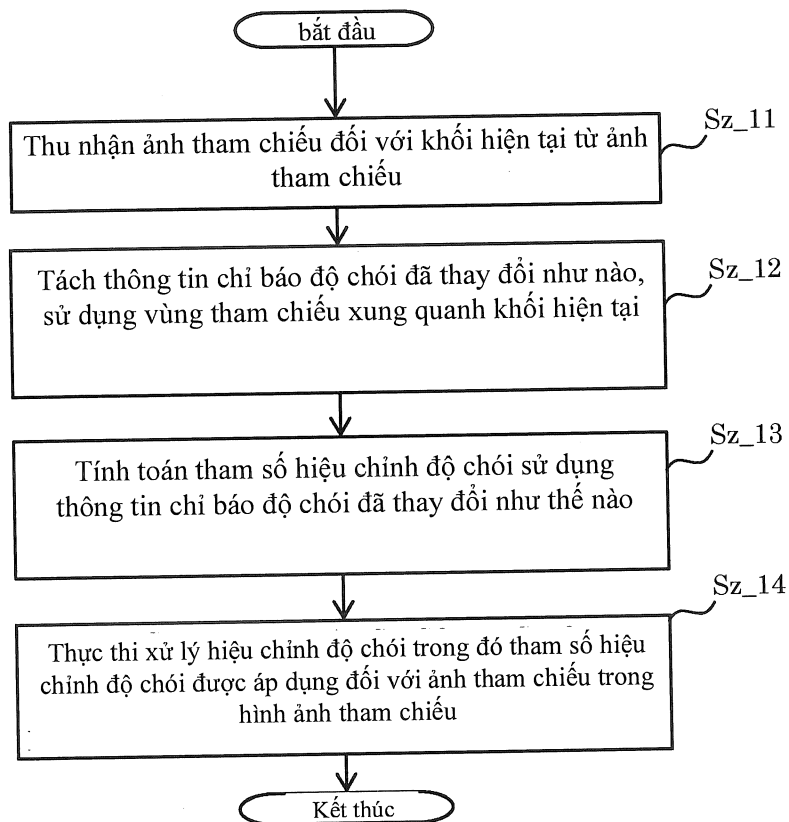
[Fig. 94]



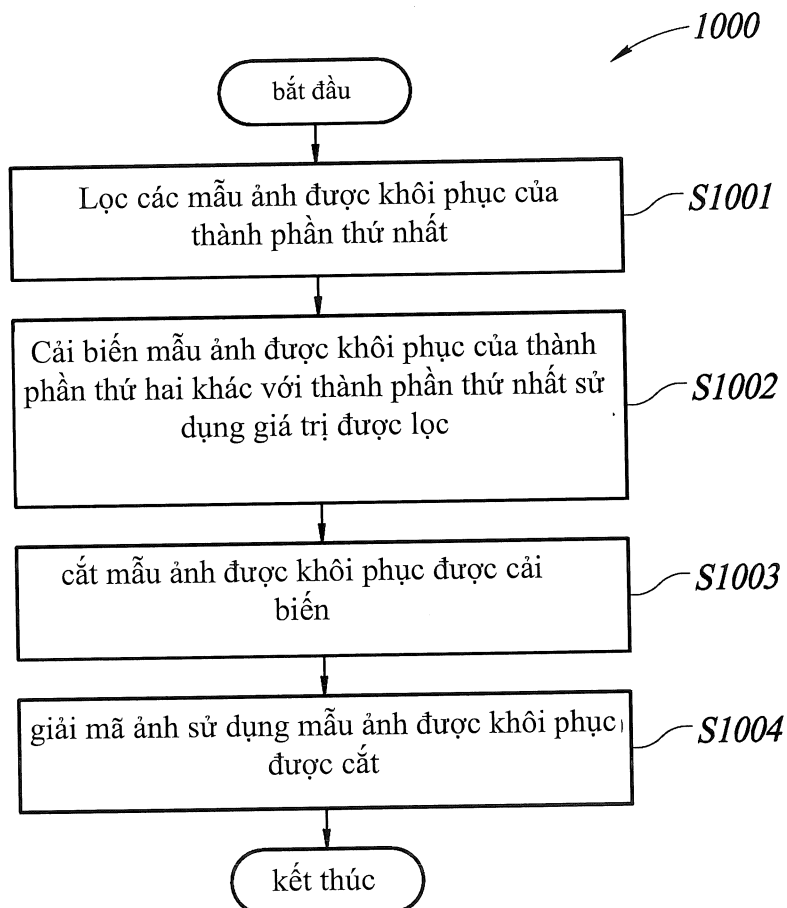
[Fig. 95]



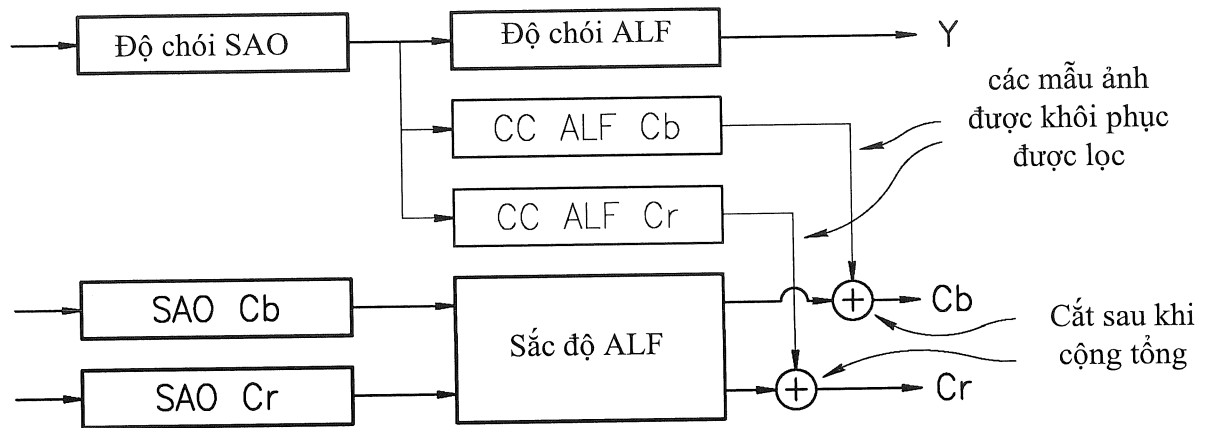
[Fig. 96]



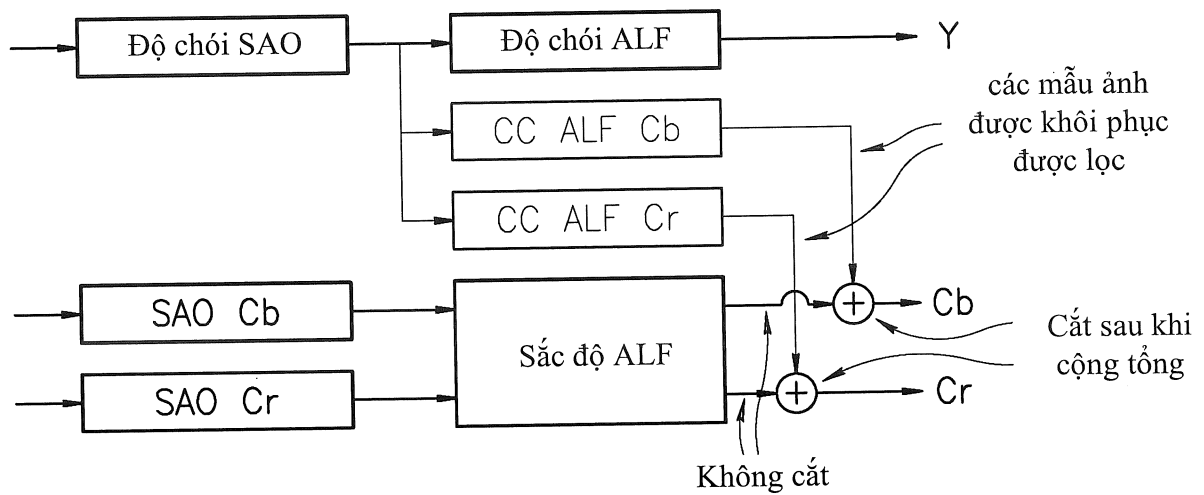
[Fig. 97]



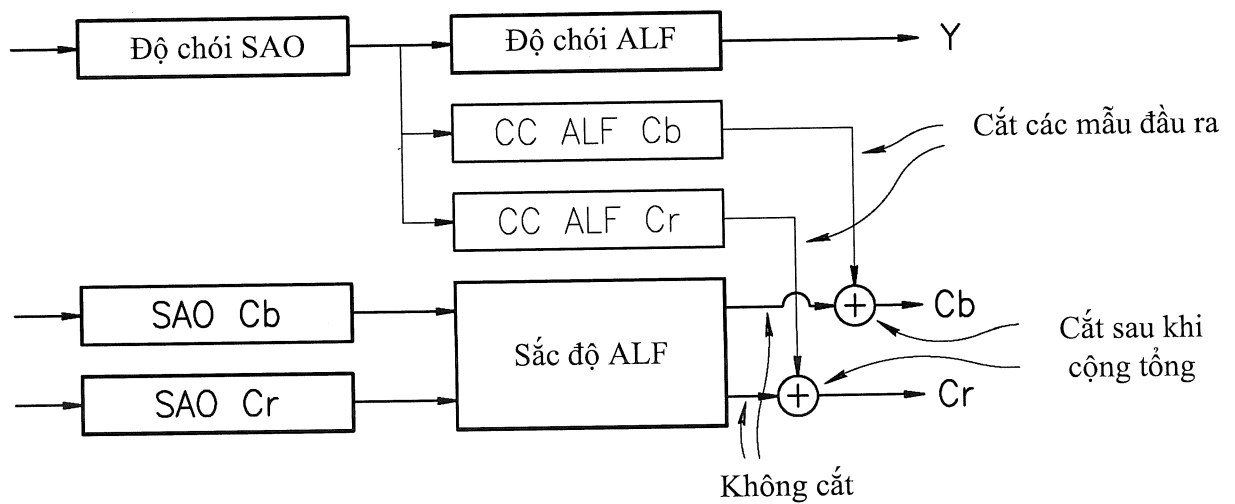
[Fig. 98]



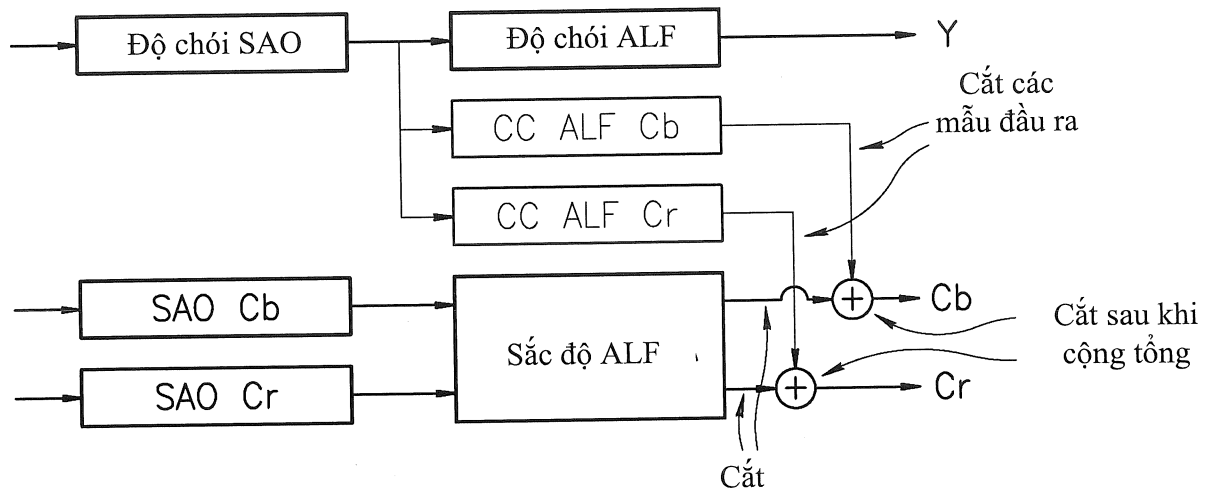
[Fig. 99]



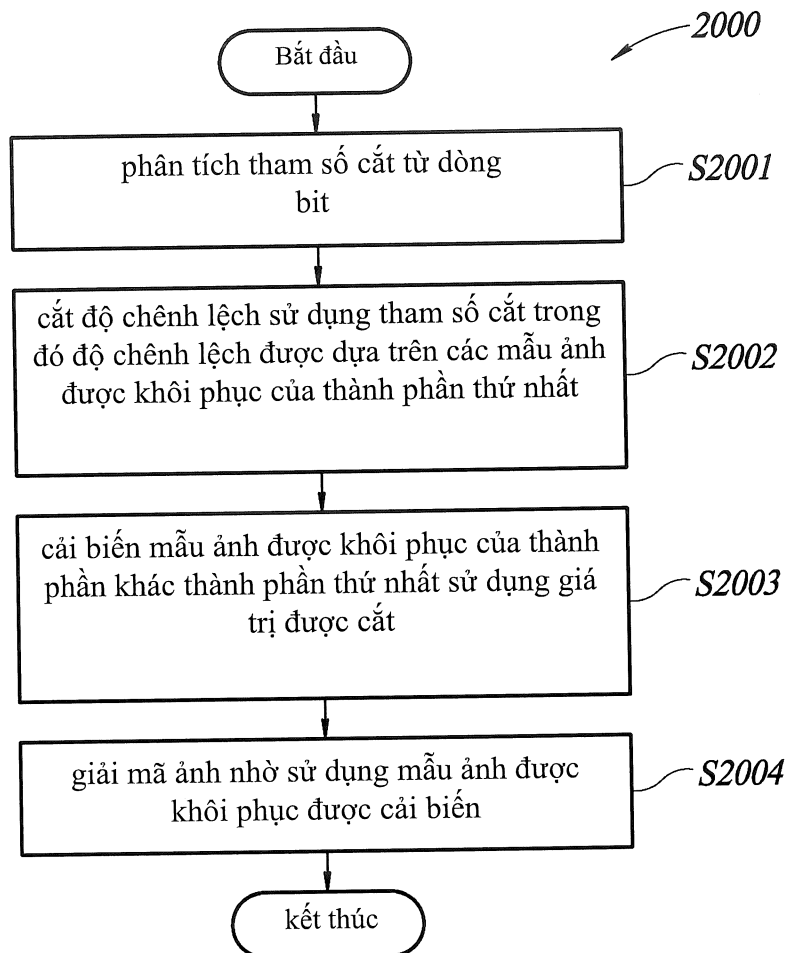
[Fig. 100]



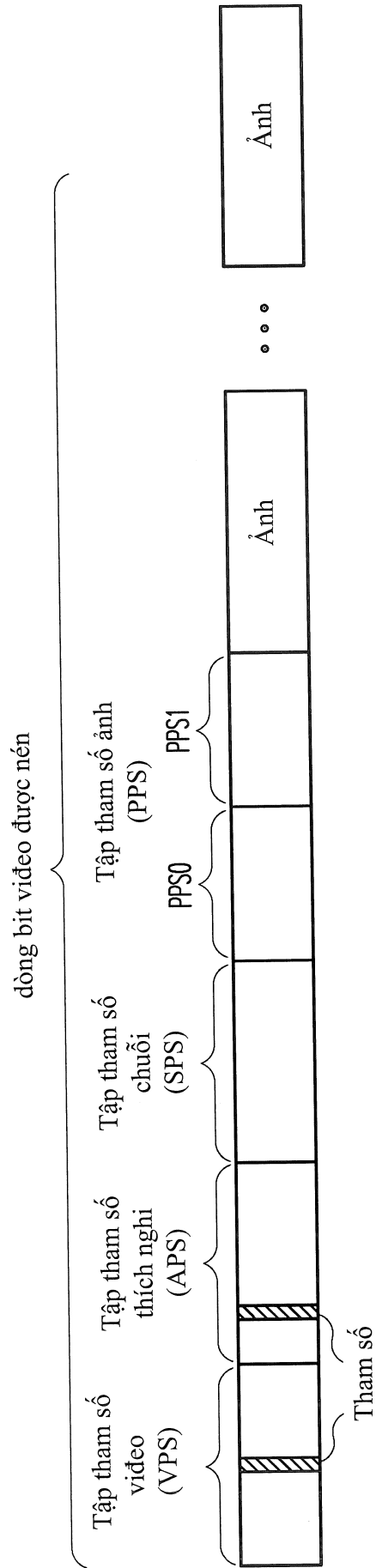
[Fig. 101]



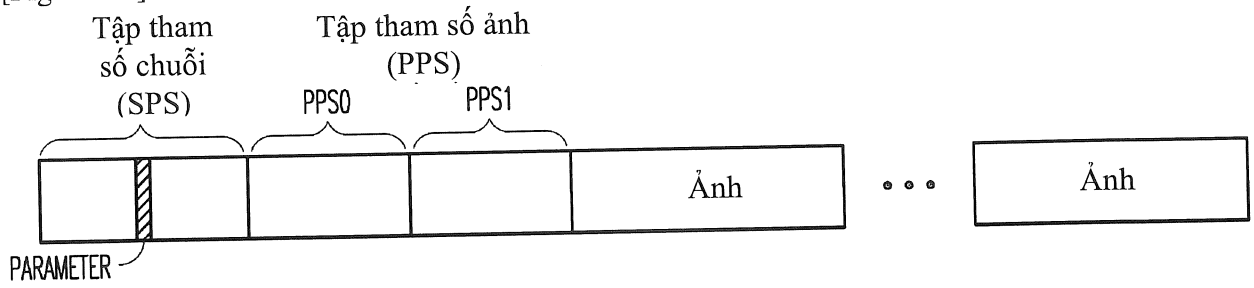
[Fig. 102]



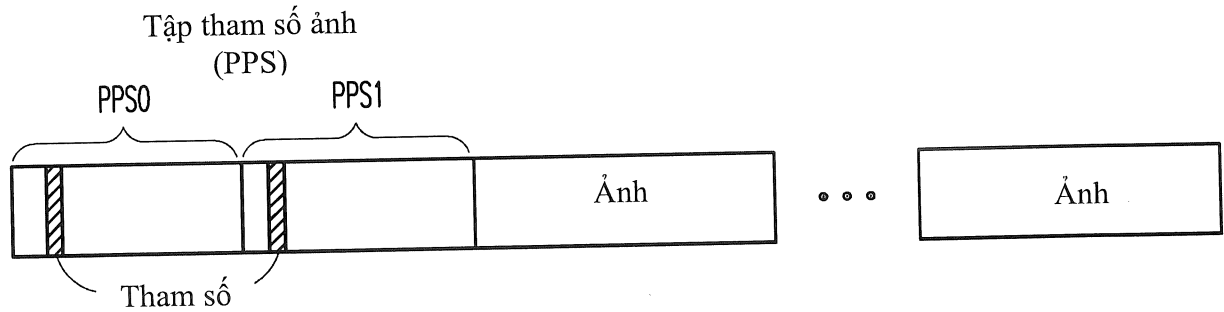
[Fig. 103A]



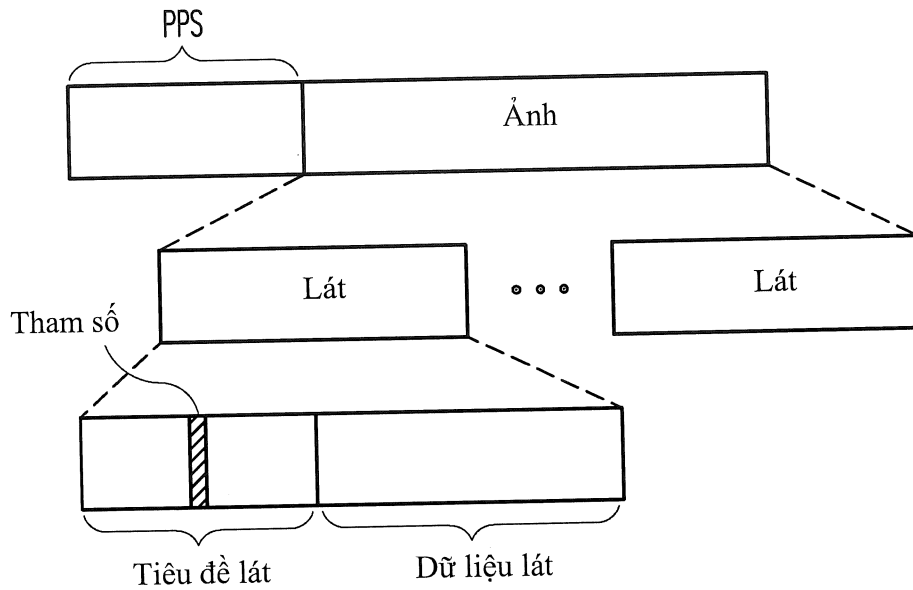
[Fig. 103B]



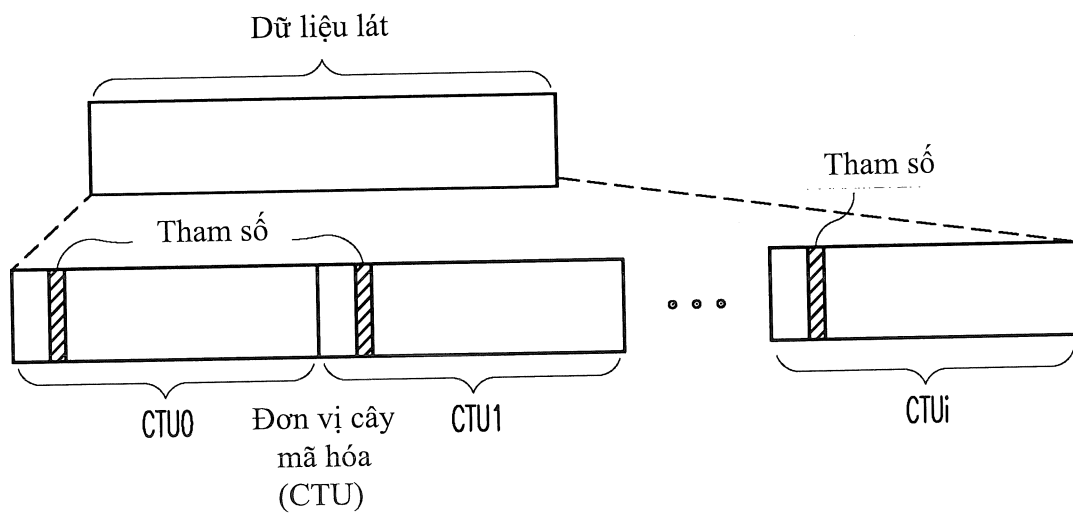
[Fig. 103C]



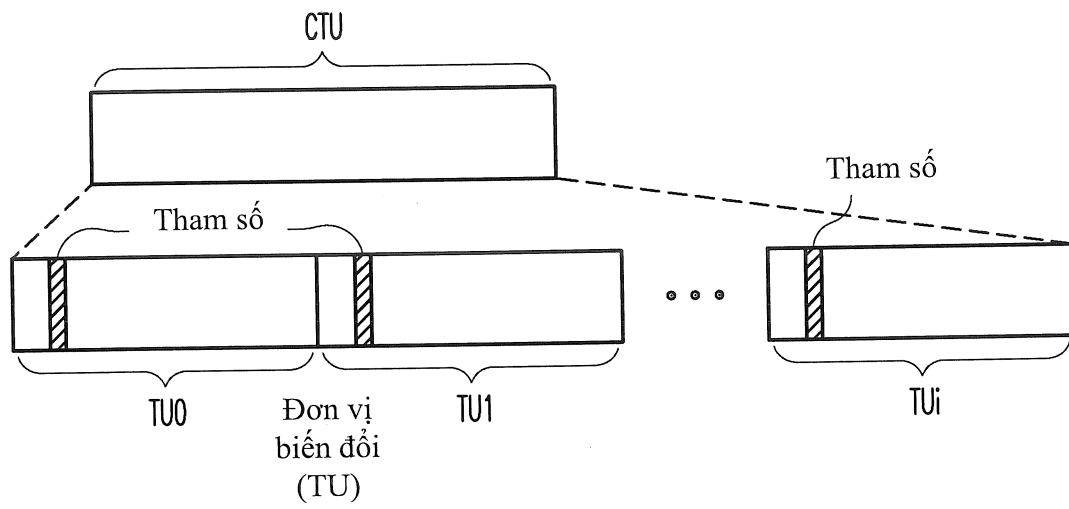
[Fig. 103D]



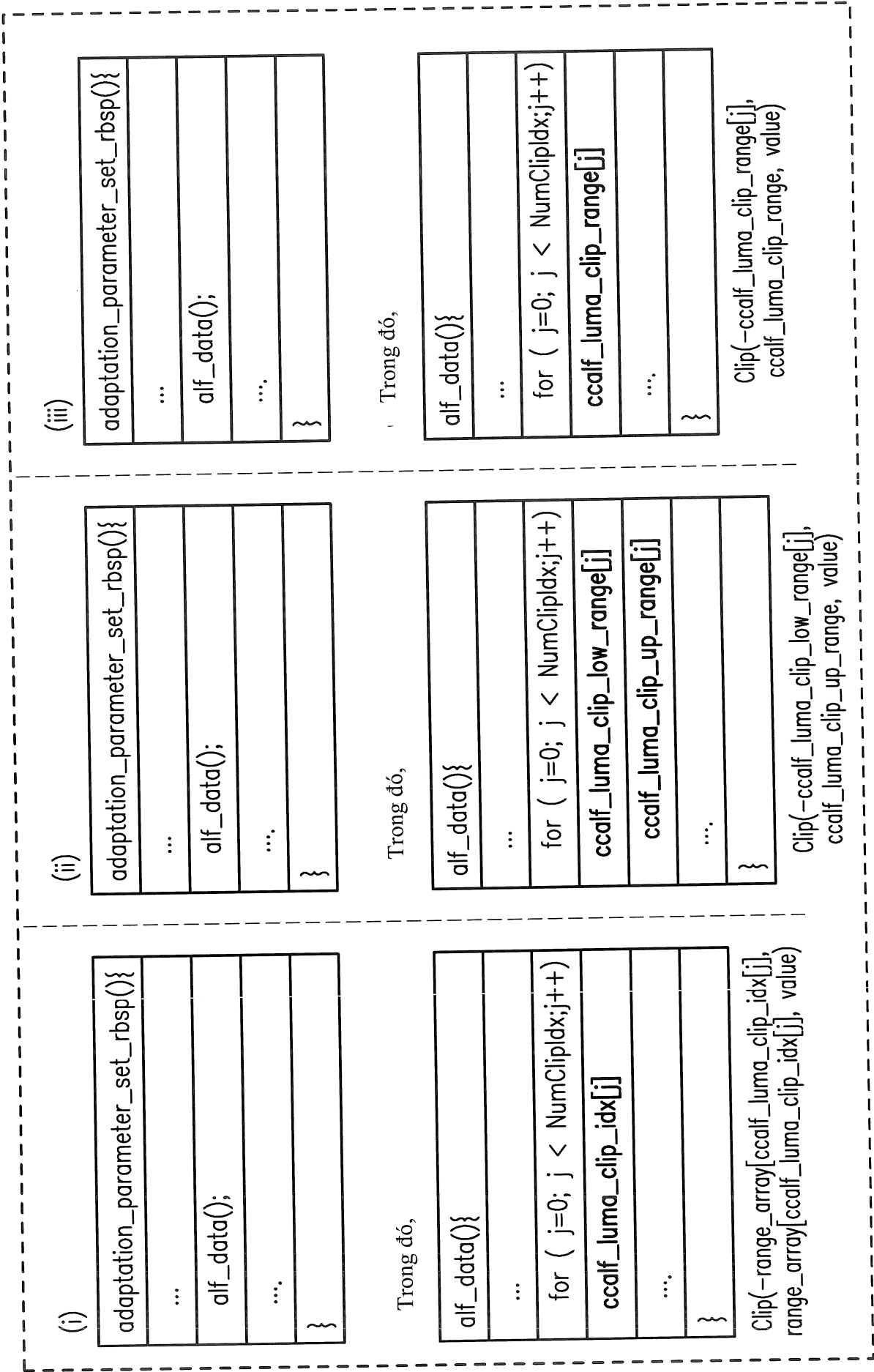
[Fig. 103E]



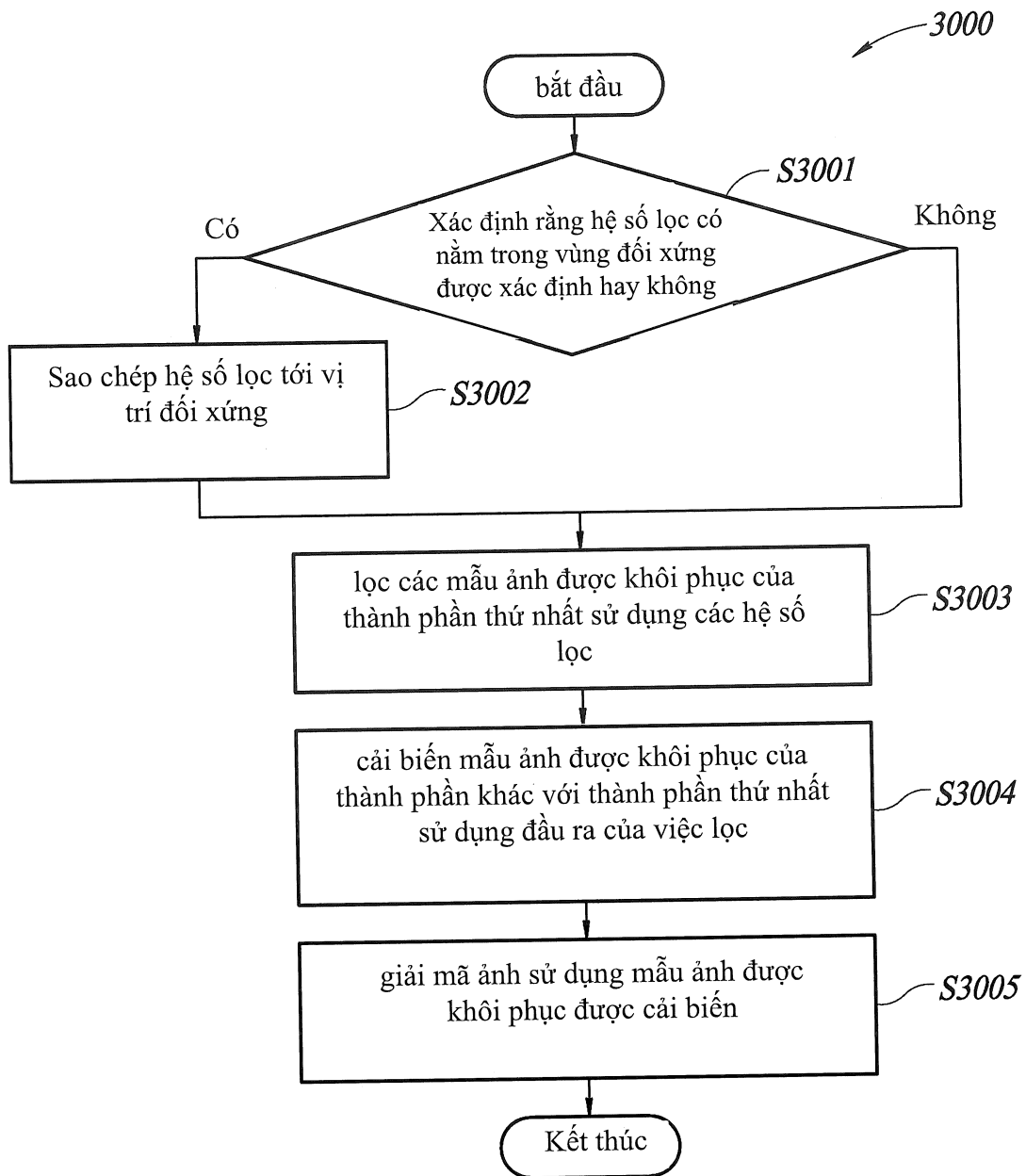
[Fig. 103F]



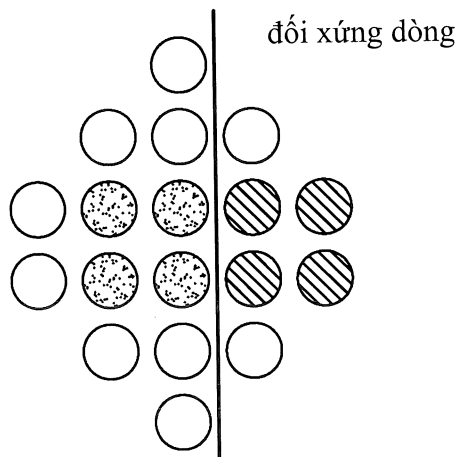
[Fig. 104]



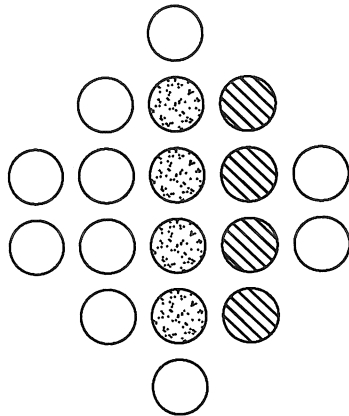
[Fig. 105]



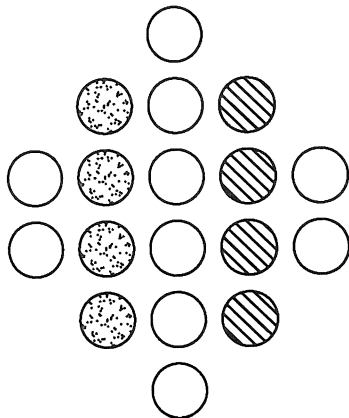
[Fig. 106A]



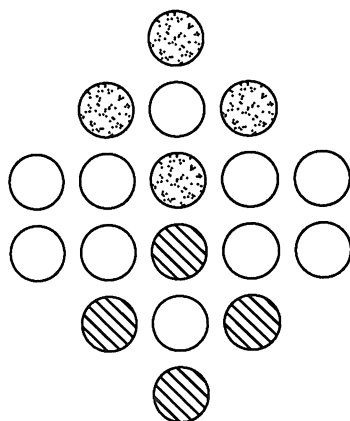
[Fig. 106B]



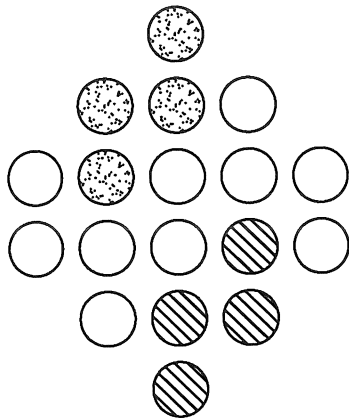
[Fig. 106C]



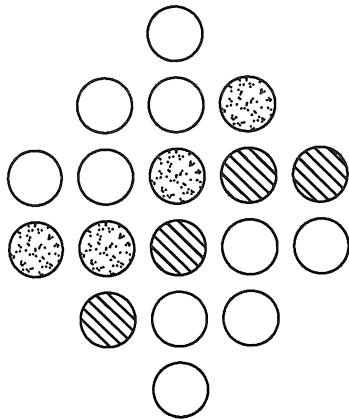
[Fig. 106D]



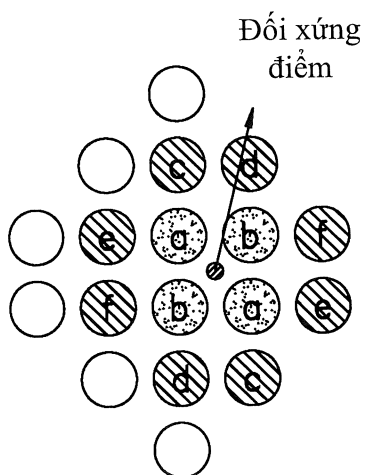
[Fig. 106E]



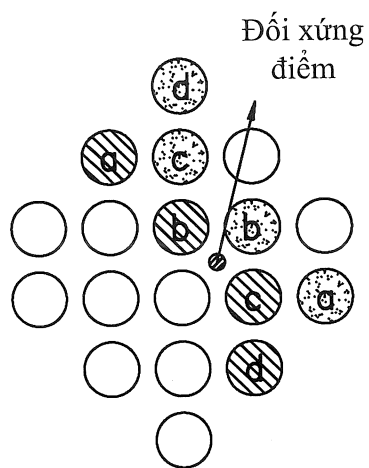
[Fig. 106F]



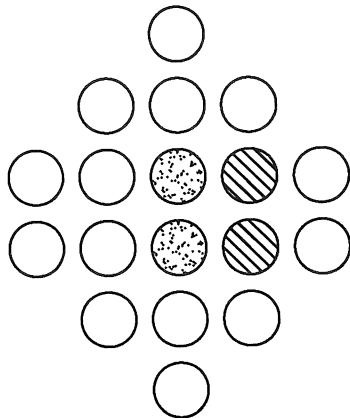
[Fig. 106G]



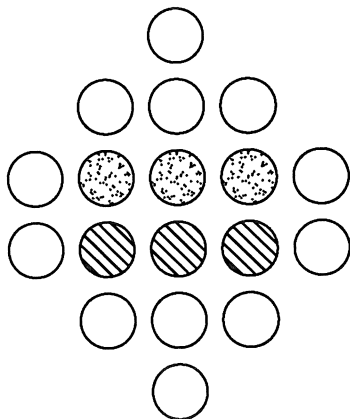
[Fig. 106H]



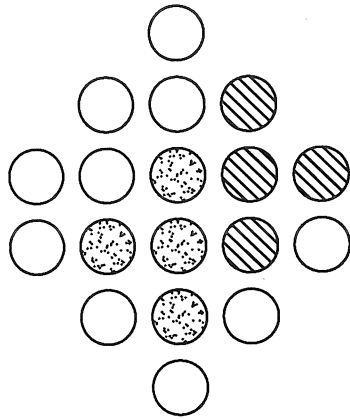
[Fig. 107A]



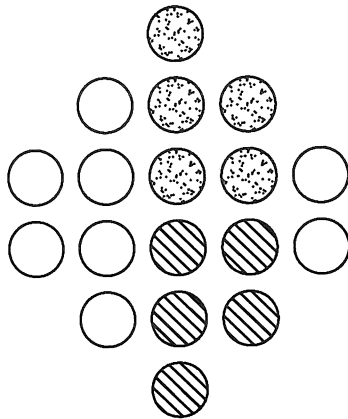
[Fig. 107B]



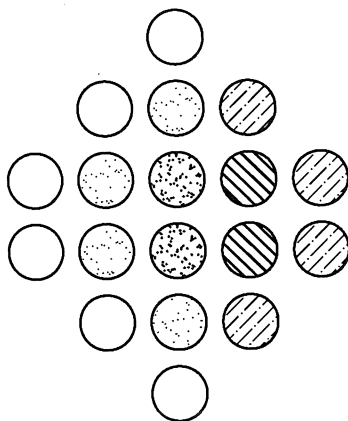
[Fig. 107C]



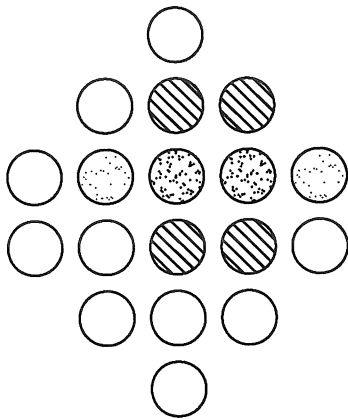
[Fig. 107D]



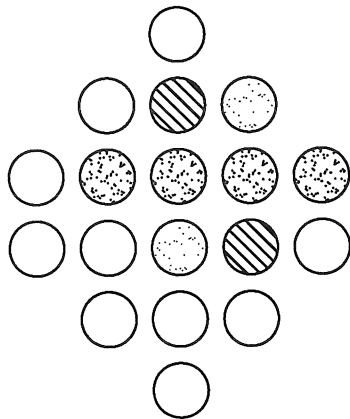
[Fig. 107E]



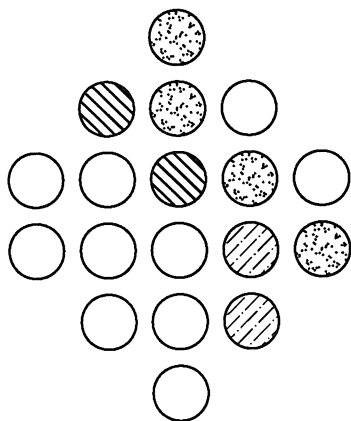
[Fig. 107F]



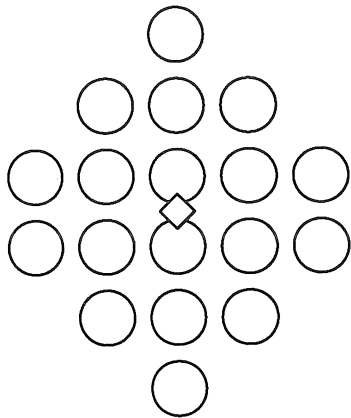
[Fig. 107G]



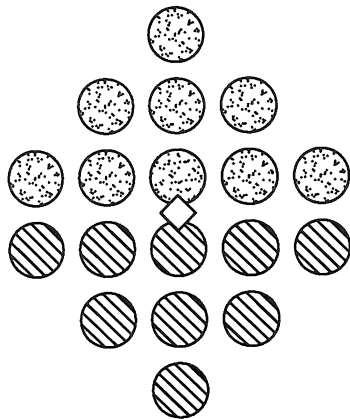
[Fig. 107H]



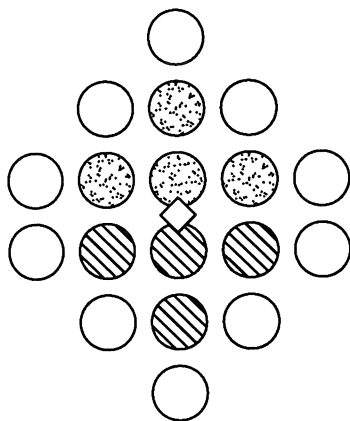
[Fig. 108A]



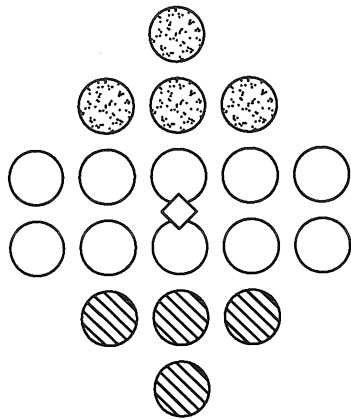
[Fig. 108B]



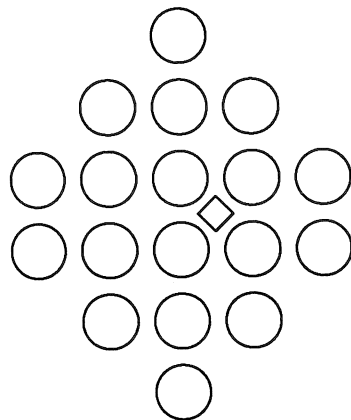
[Fig. 108C]



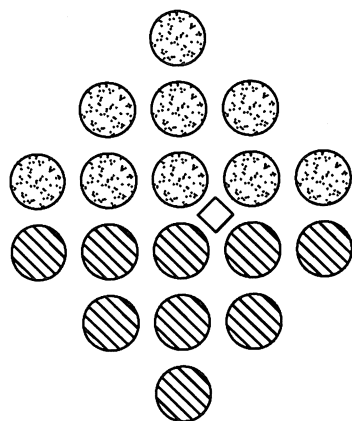
[Fig. 108D]



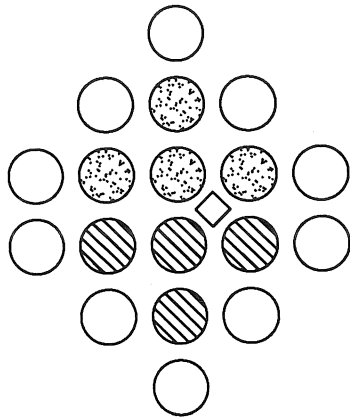
[Fig. 108E]



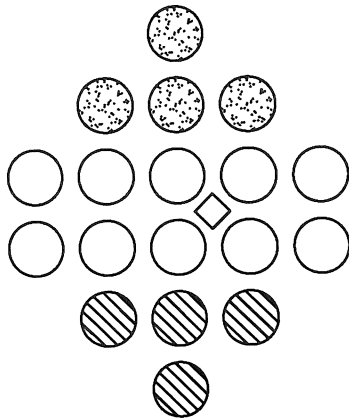
[Fig. 108F]



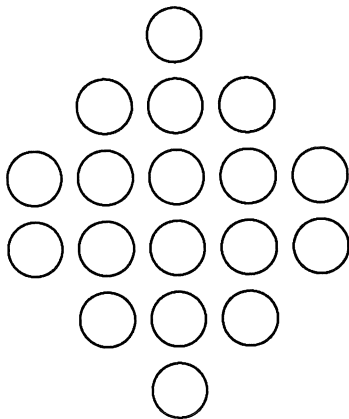
[Fig. 108G]



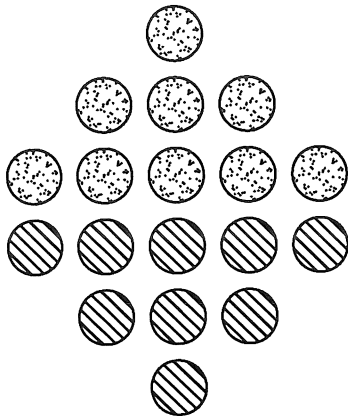
[Fig. 108H]



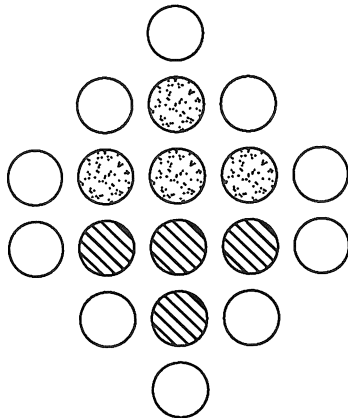
[Fig. 109A]



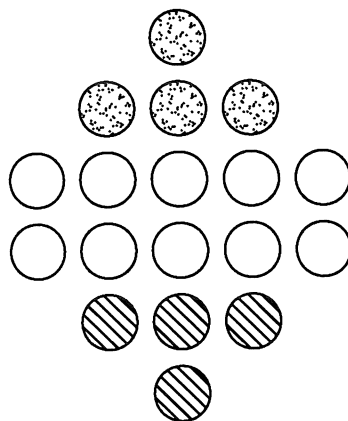
[Fig. 109B]



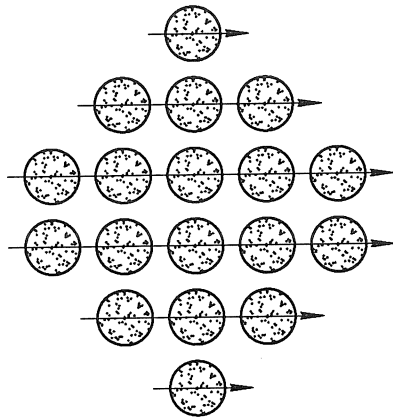
[Fig. 109C]



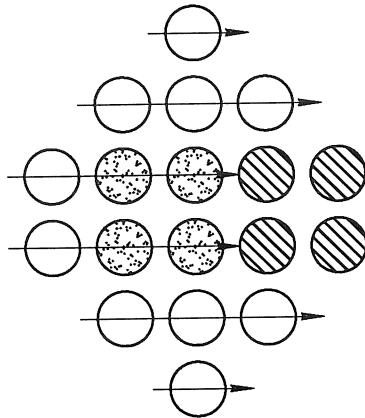
[Fig. 109D]



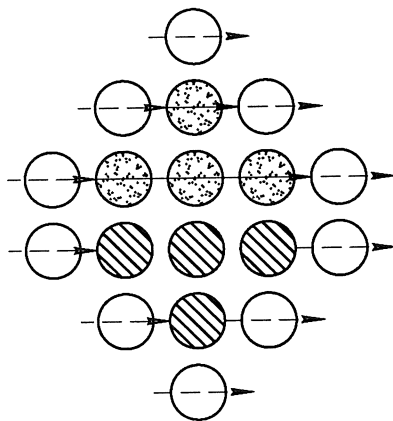
[Fig. 110A]



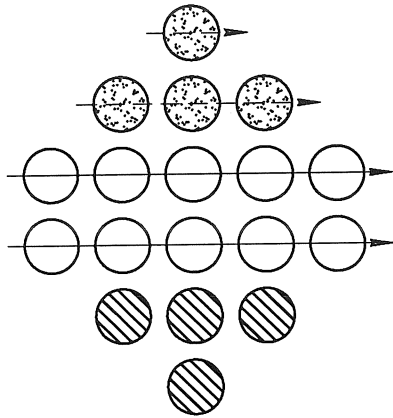
[Fig. 110B]



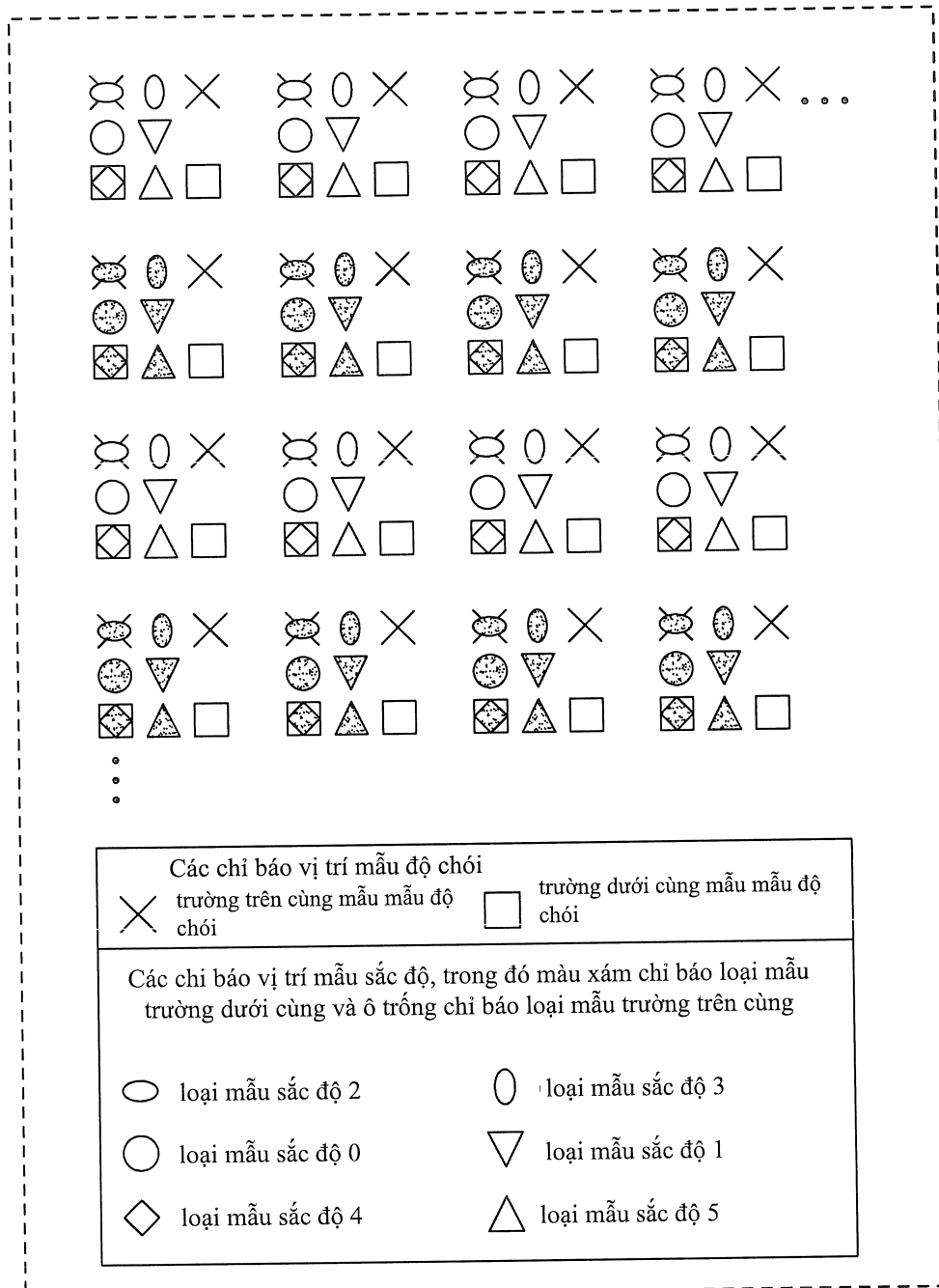
[Fig. 110C]



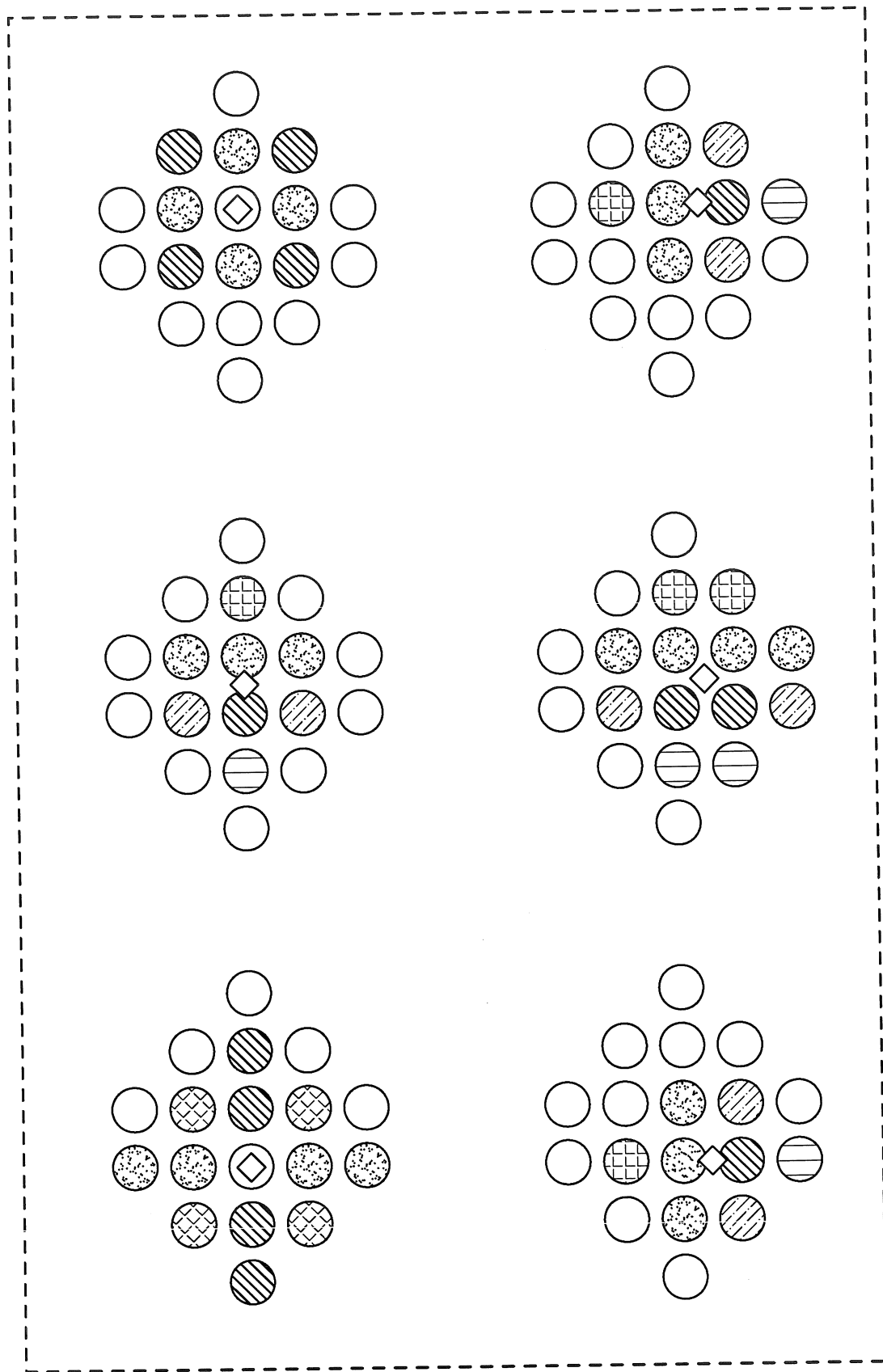
[Fig. 110D]



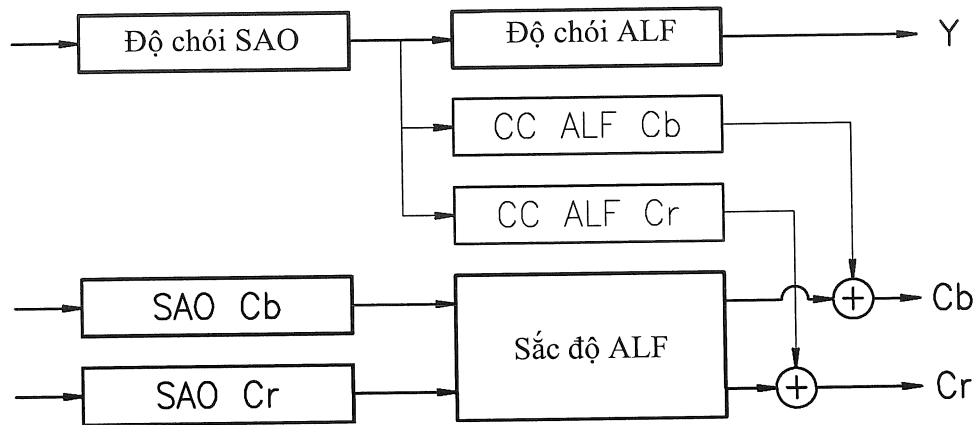
[Fig. 111]



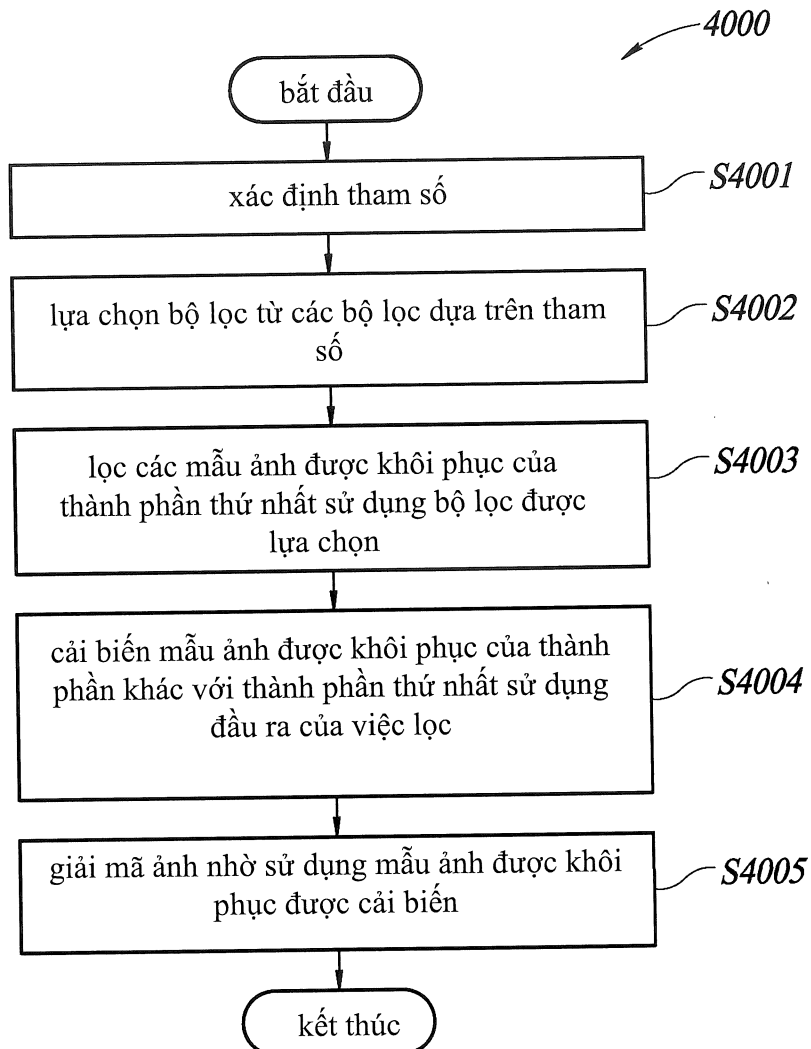
[Fig. 112]



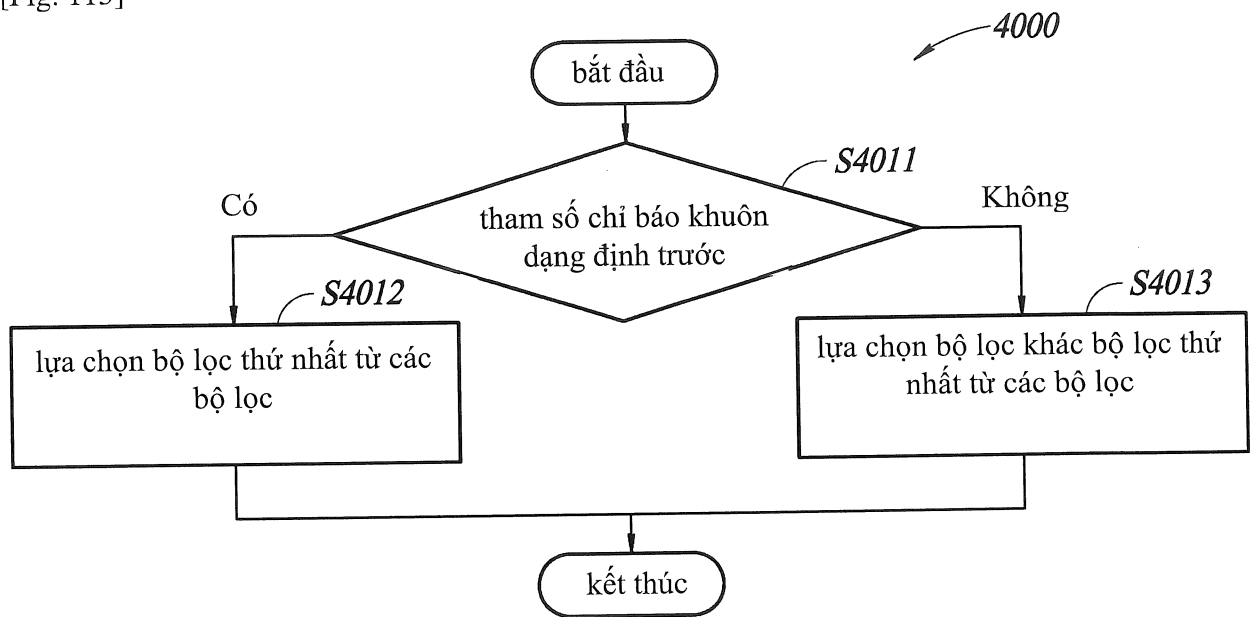
[Fig. 113]



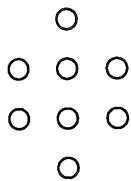
[Fig. 114]



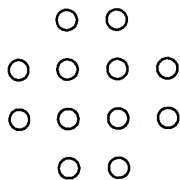
[Fig. 115]



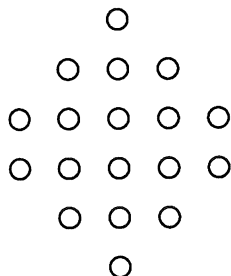
[Fig. 116-1A]



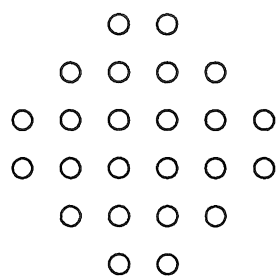
[Fig. 116-1B]



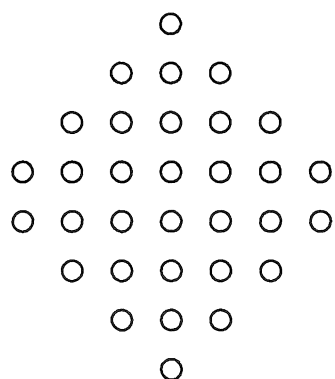
[Fig. 116-1C]



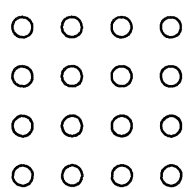
[Fig. 116-1D]



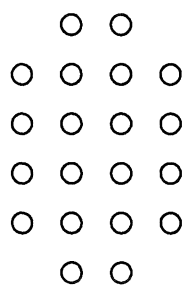
[Fig. 116-1E]



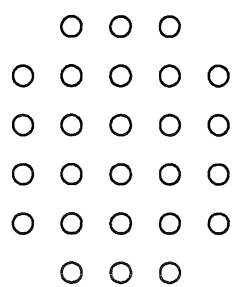
[Fig. 116-1F]



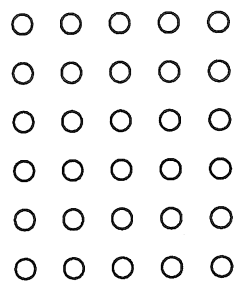
[Fig. 116-1G]



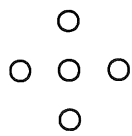
[Fig. 116-1H]



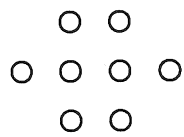
[Fig. 116-1I]



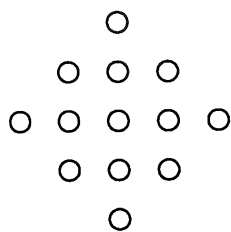
[Fig. 117-2A]



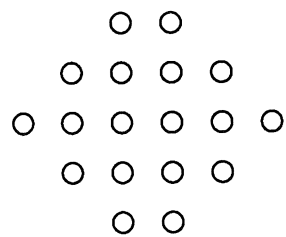
[Fig. 117-2B]



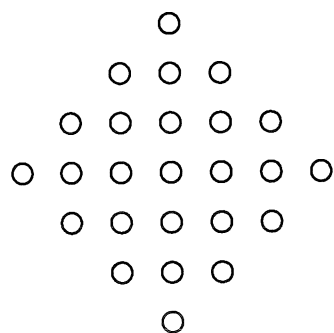
[Fig. 117-2C]



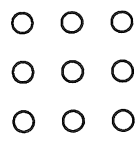
[Fig. 117-2D]



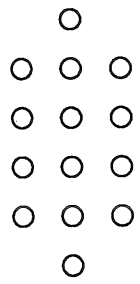
[Fig. 117-2E]



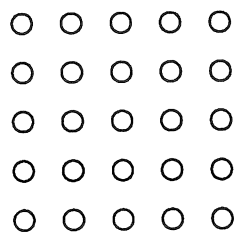
[Fig. 117-2F]



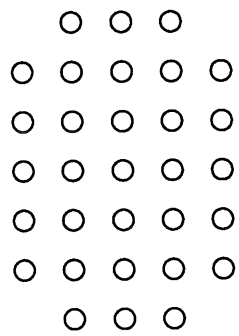
[Fig. 117-2G]



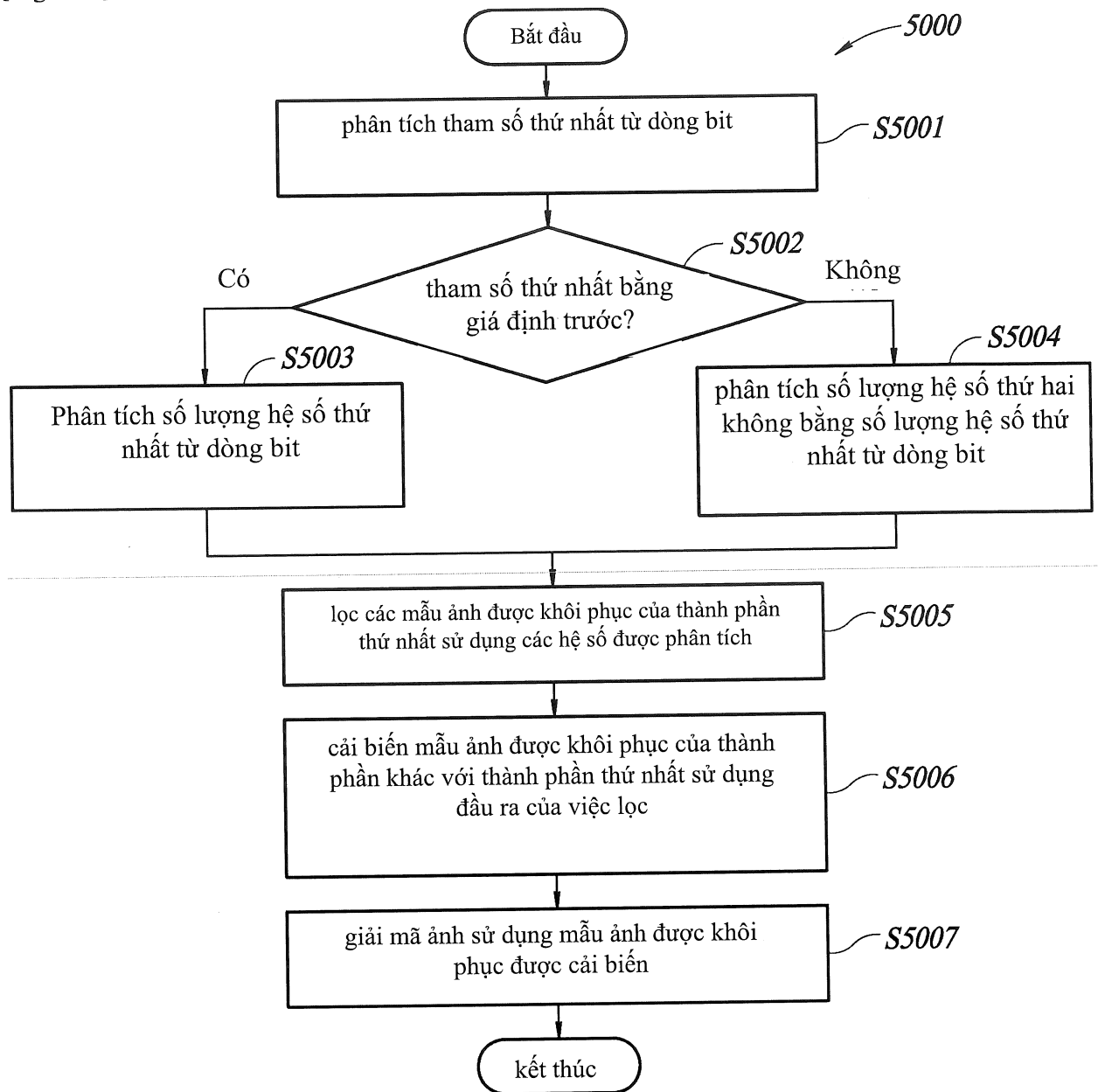
[Fig. 117-2H]



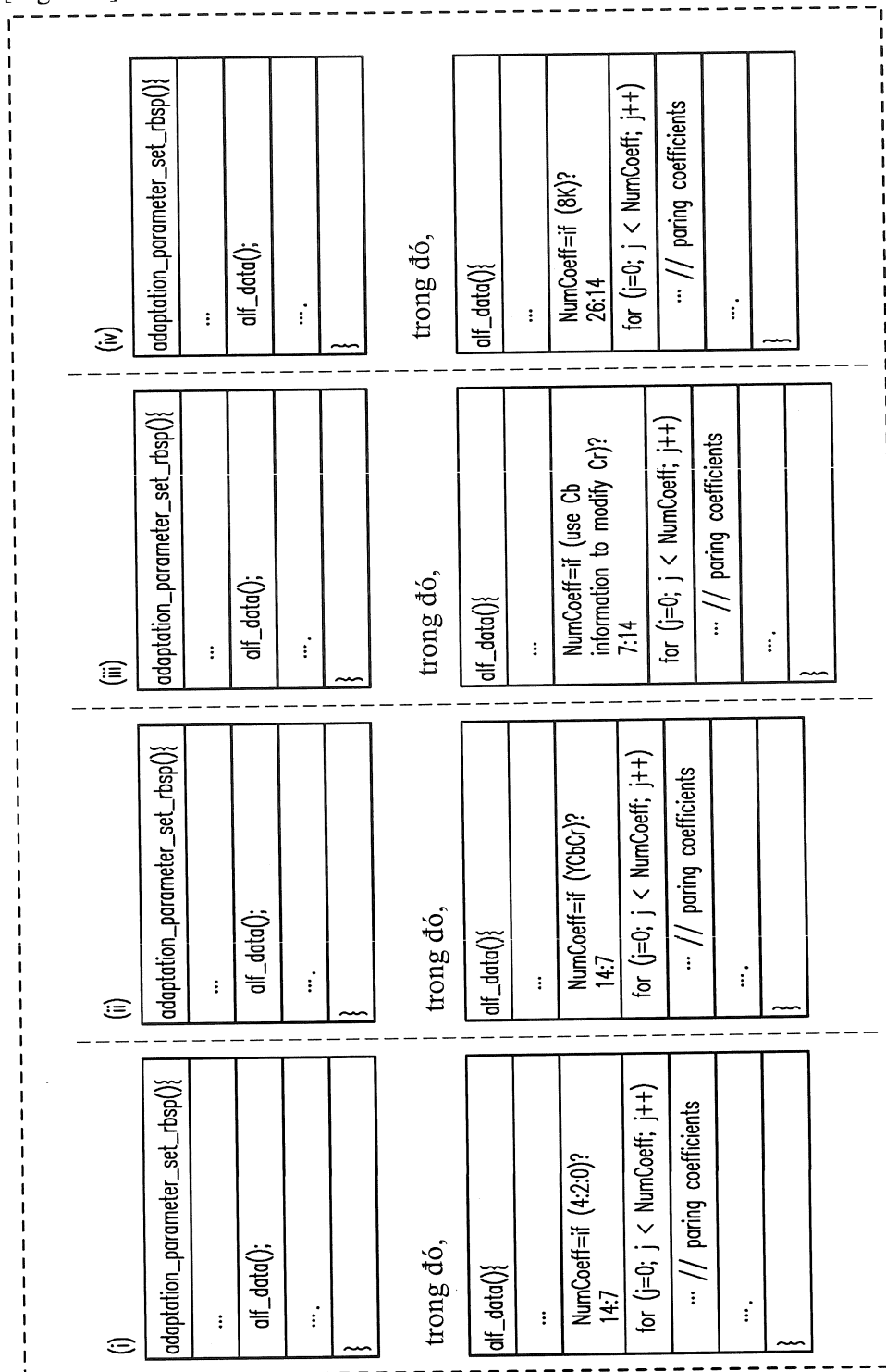
[Fig. 117-2I]



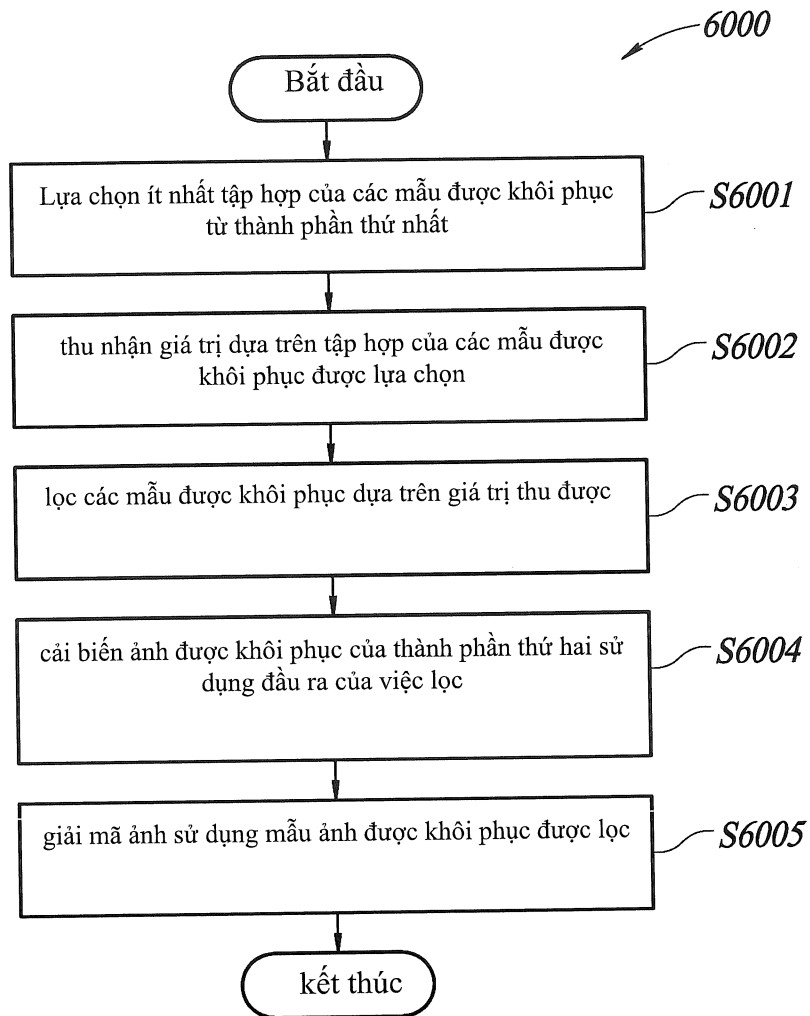
[Fig. 118]



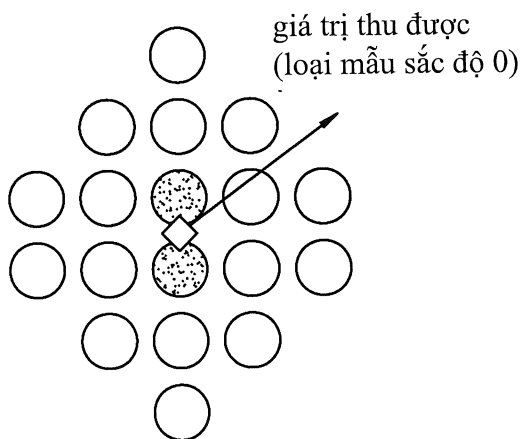
[Fig. 119]



[Fig. 120]



[Fig. 121]



[Fig. 122]

Ví dụ của công thức lọc độ chói ALF

- Tổng biến thiên được thu nhận như sau:

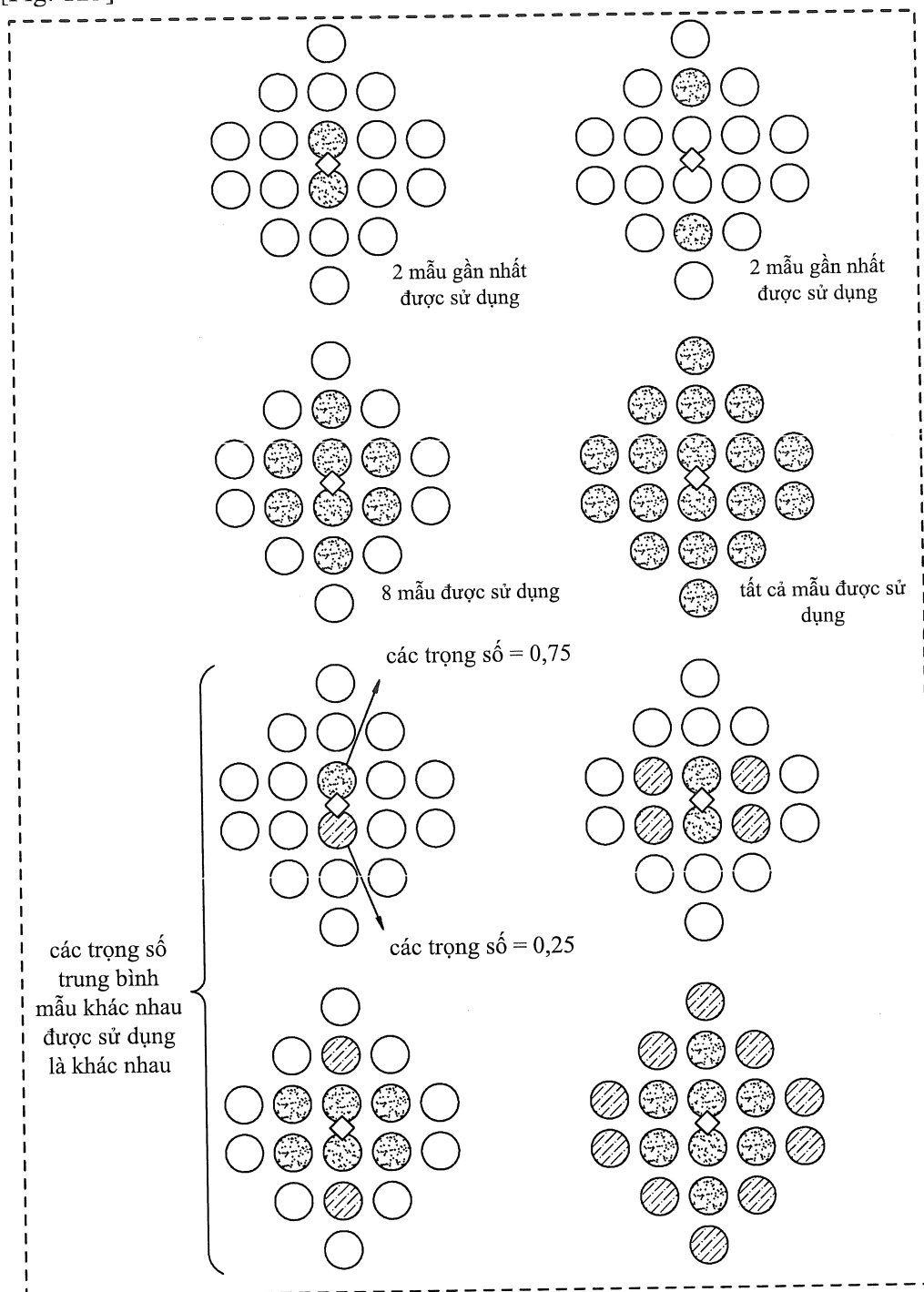
$$\begin{aligned}
 \text{sum} = & f[\text{idx}[0]] * (\text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_{v+r3}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_{v-r3}] - \text{curr})) + \\
 f[\text{idx}[1]] * & (\text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_{x+1}, v_{v+r2}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_{x-1}, v_{v-r2}] - \text{curr})) + \\
 f[\text{idx}[2]] * & (\text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_{v+r2}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_{v-r2}] - \text{curr})) + \\
 f[\text{idx}[3]] * & (\text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_{x-1}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_{x+1}, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[4]] * & (\text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_{x+2}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_{x-2}, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[5]] * & (\text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_{x+1}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_{x-1}, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[6]] * & (\text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[7]] * & (\text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_{x-1}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_{x+1}, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[8]] * & (\text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_{x-2}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_{x+2}, v_{v-r1}] - \text{curr})) + \\
 f[\text{idx}[9]] * & (\text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_{x+3}, v_v] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_{x-3}, v_v] - \text{curr})) + \\
 f[\text{idx}[10]] * & (\text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_{x+2}, v_v] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_{x-2}, v_v] - \text{curr})) + \\
 f[\text{idx}[11]] * & (\text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_{x+1}, v_v] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_{x-1}, v_v] - \text{curr})) + \\
 \text{sum} = & \boxed{\text{curr}} + ((\text{sum} + 64) >> 7) \quad (8-1188)
 \end{aligned}$$

Ví dụ của công thức lọc sắc độ ALF

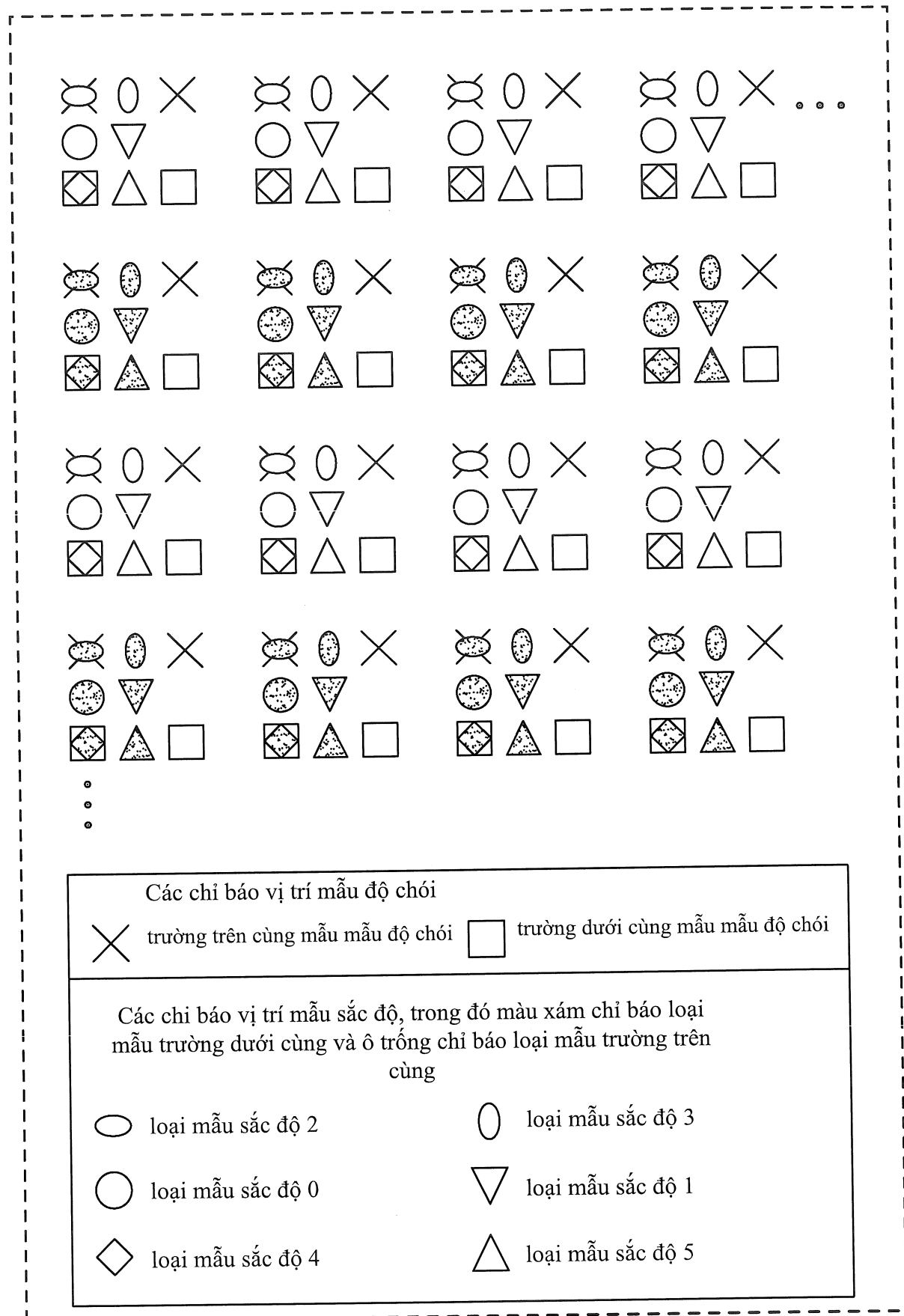
- Tổng biến thiên được thu nhận như sau:

$$\begin{aligned}
 \text{sum} = & f[0] * (\text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_{v+r2}] - \text{curr}) + \\
 & \text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_{v+r2}] - \text{curr})) + \\
 f[1] * & (\text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_{x+1}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_{x-1}, v_{v-r1}] - \text{curr})) + \\
 f[2] * & (\text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_{v-r1}] - \text{curr})) + \\
 f[3] * & (\text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_{x-1}, v_{v+r1}] - \text{curr}) + \\
 & \text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_{x+1}, v_{v-r1}] - \text{curr})) + \\
 f[4] * & (\text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_{x+2}, v_v] - \text{curr}) + \\
 & \text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_{x-2}, v_v] - \text{curr})) + \\
 f[5] * & (\text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_{x+1}, v_v] - \text{curr}) + \\
 & \text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_{x-1}, v_v] - \text{curr})) + \\
 \text{sum} = & \text{curr} + ((\text{sum} + 64) >> 7) \quad (8-1239)
 \end{aligned}$$

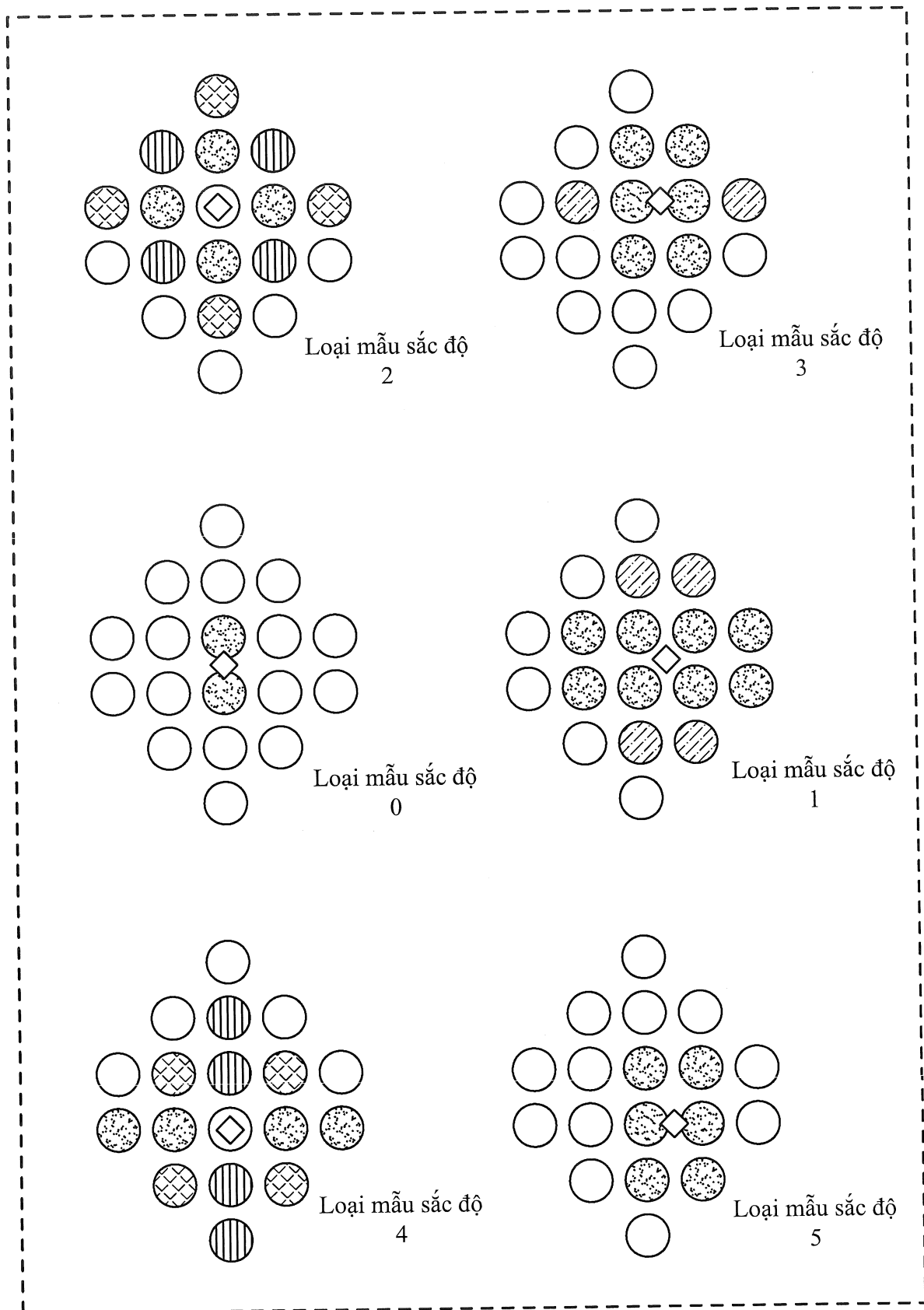
[Fig. 123]



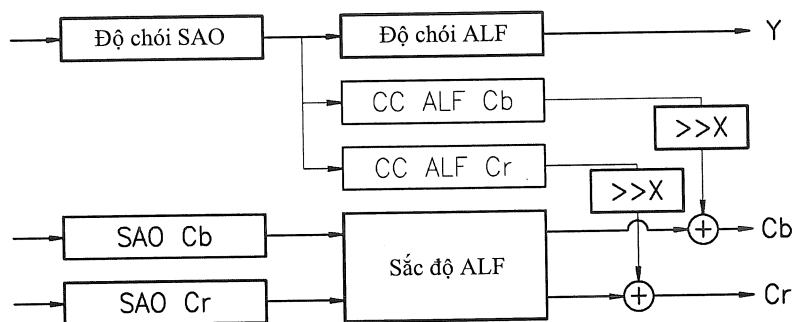
[Fig. 124]



[Fig. 125]



[Fig. 126]



[Fig. 127]

Ví dụ về công thức lọc độ chói ALF

Tổng biến thiên được thu nhận như sau:

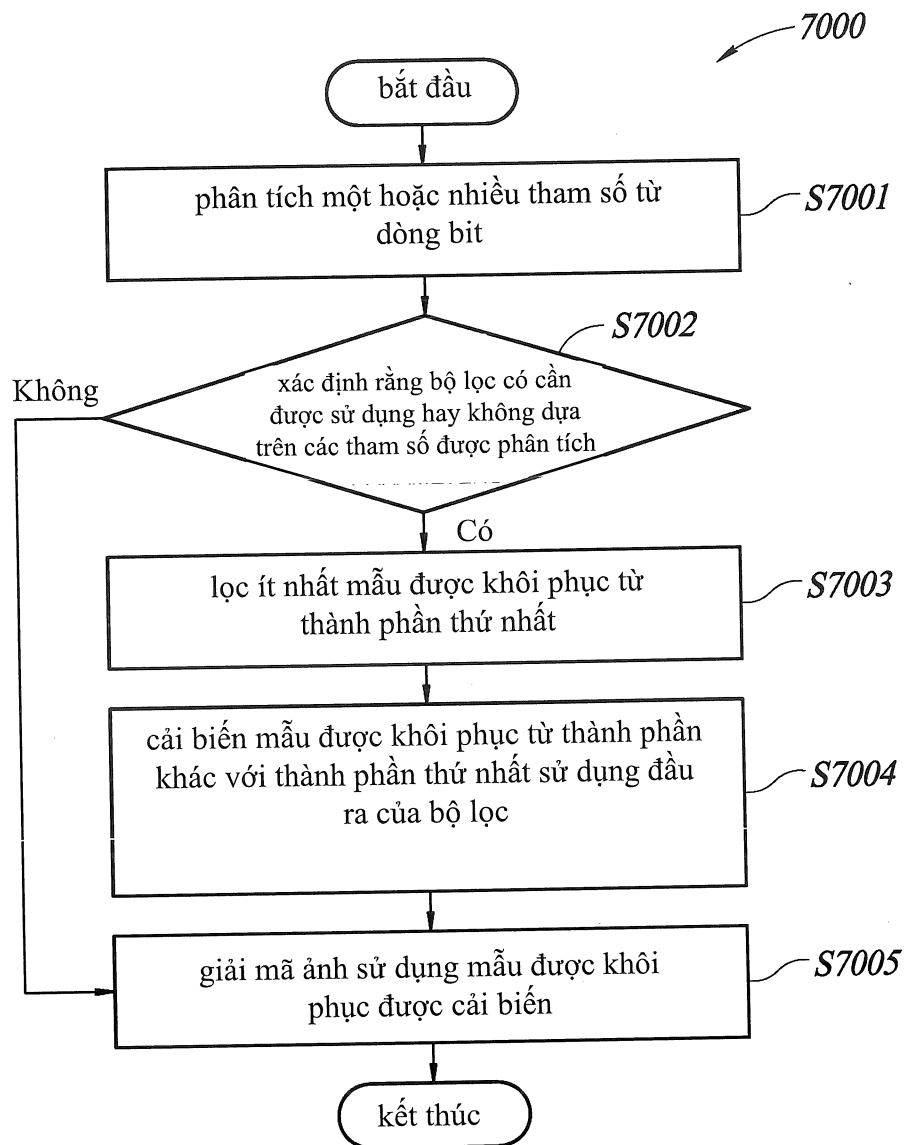
$$\begin{aligned}
 \text{sum} = & f[\text{idx}[0]] * (\text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_y+r3] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_y-r3] - \text{curr})) + \\
 & f[\text{idx}[1]] * (\text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_x+1, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_x-1, v_y-r2] - \text{curr})) + \\
 & f[\text{idx}[2]] * (\text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_y-r2] - \text{curr})) + \\
 & f[\text{idx}[3]] * (\text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_x-1, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_x+1, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[4]] * (\text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_x+2, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_x-2, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[5]] * (\text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_x+1, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_x-1, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[6]] * (\text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[7]] * (\text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_x-1, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_x+1, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[8]] * (\text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_x-2, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_x+2, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[9]] * (\text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_x+3, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_x-3, v_y] - \text{curr})) + \\
 & f[\text{idx}[10]] * (\text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_x+2, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_x-2, v_y] - \text{curr})) + \\
 & f[\text{idx}[11]] * (\text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_x+1, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_x-1, v_y] - \text{curr})) + \\
 & \text{sum} = \text{curr} + ((\text{sum} + 64) \gg 7) \quad (8-1188)
 \end{aligned}$$

Ví dụ về công thức lọc sắc độ ALF

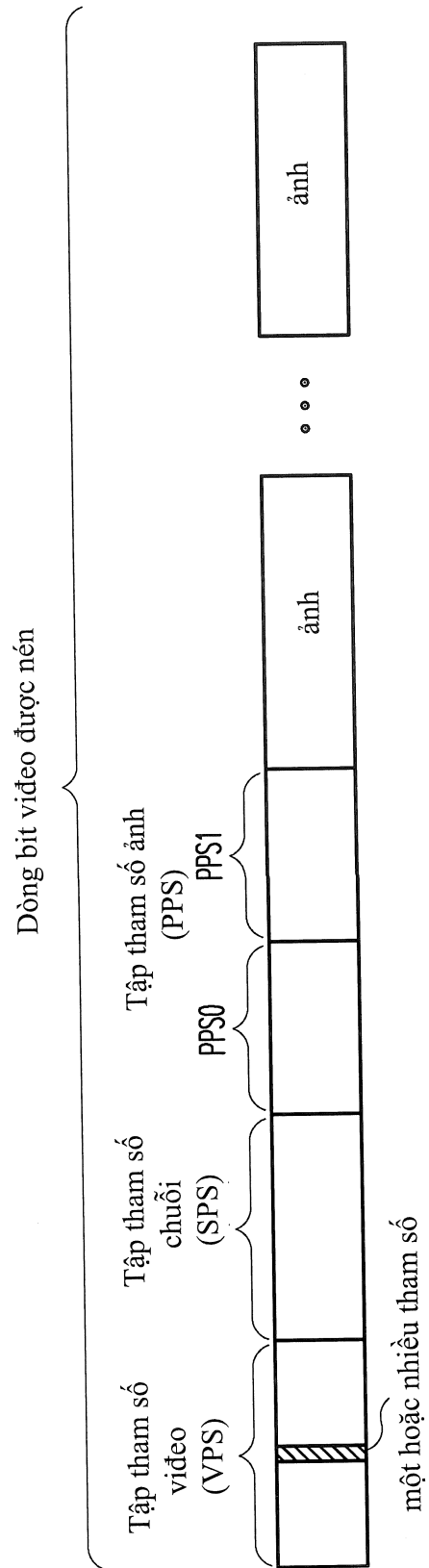
Tổng biến thiên được thu nhận như sau:

$$\begin{aligned}
 \text{sum} = & f[0] * (\text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_y+r2] - \text{curr})) + \\
 & f[1] * (\text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_x+1, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_x-1, v_y-r1] - \text{curr})) + \\
 & f[2] * (\text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_y-r1] - \text{curr})) + \\
 & f[3] * (\text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_x-1, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_x+1, v_y-r1] - \text{curr})) + \\
 & f[4] * (\text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_x+2, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_x-2, v_y] - \text{curr})) + \\
 & f[5] * (\text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_x+1, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_x-1, v_y] - \text{curr})) + \\
 & \text{sum} = \text{curr} + ((\text{sum} + 64) \gg 7) \quad (8-1239)
 \end{aligned}$$

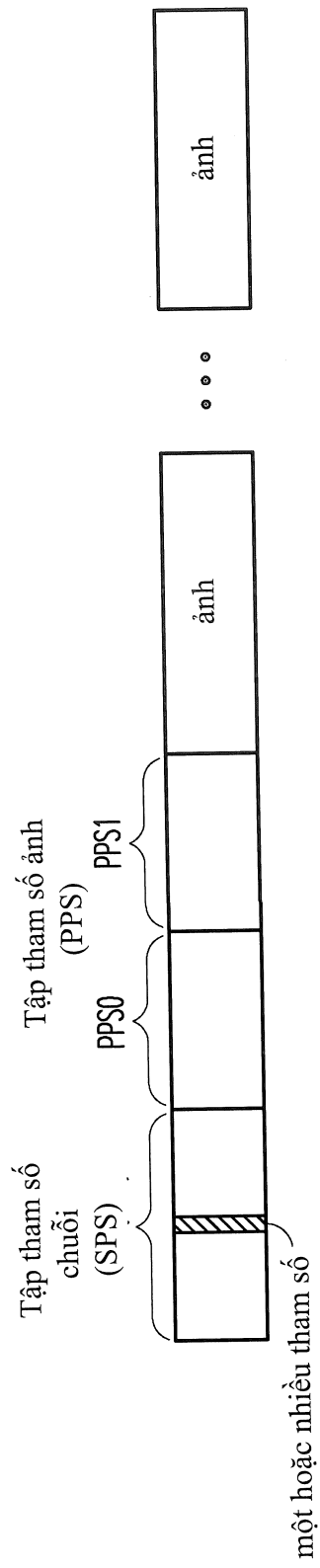
[Fig. 128]



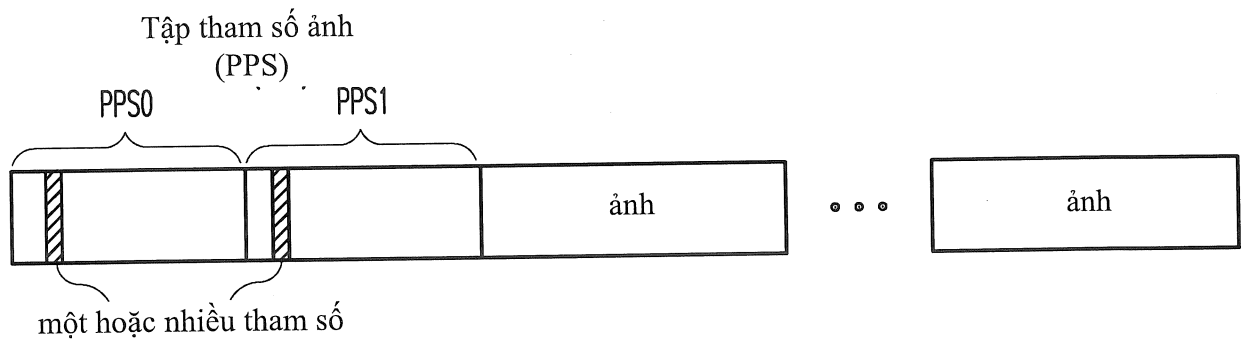
[Fig. 129A]



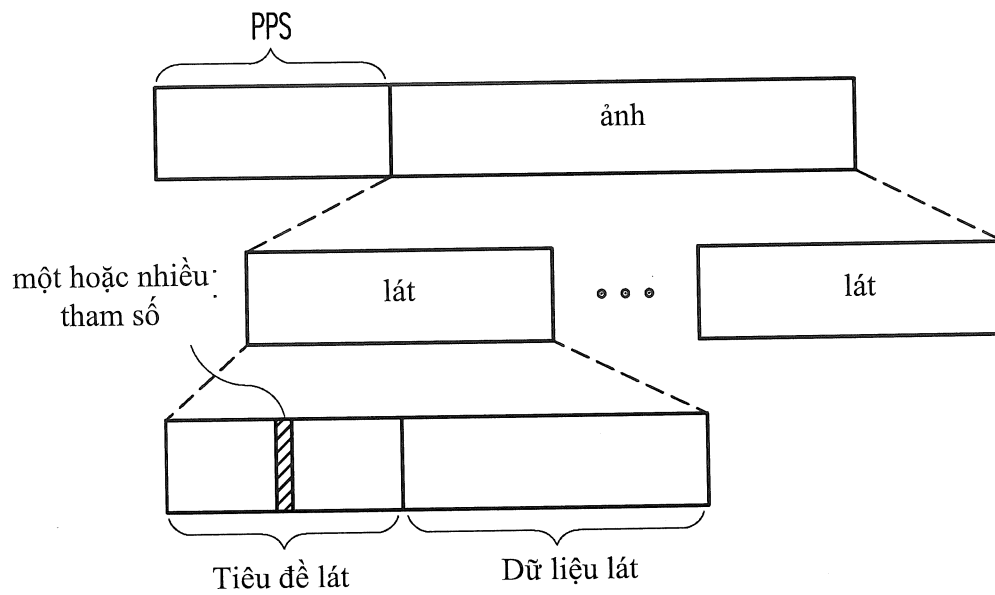
[Fig. 129B]



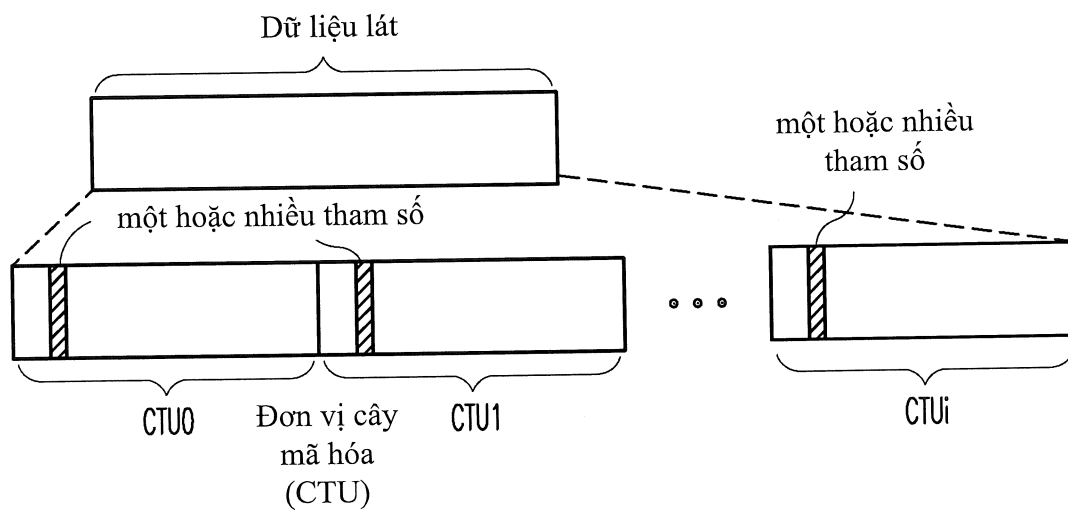
[Fig. 129C]



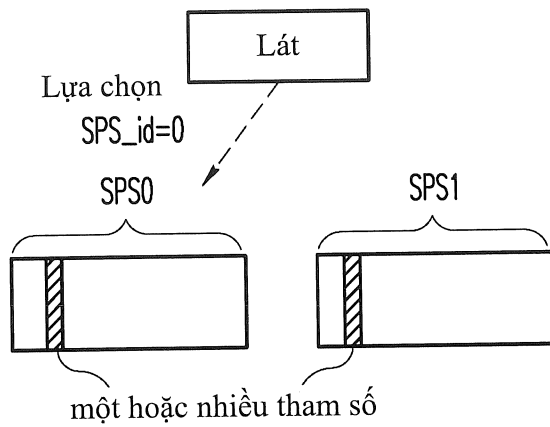
[Fig. 129D]



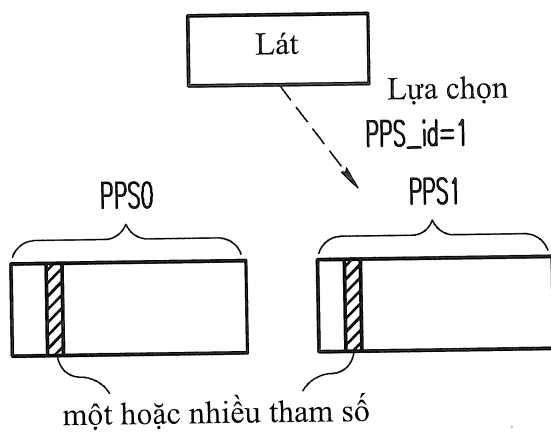
[Fig. 129E]



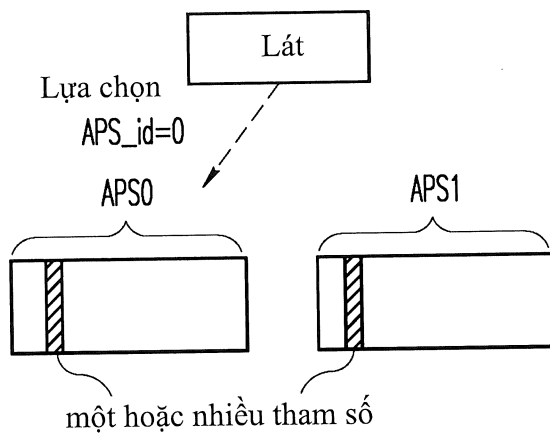
[Fig. 130A]



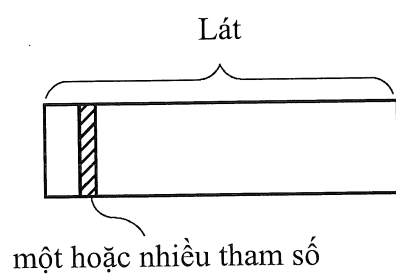
[Fig. 130B]



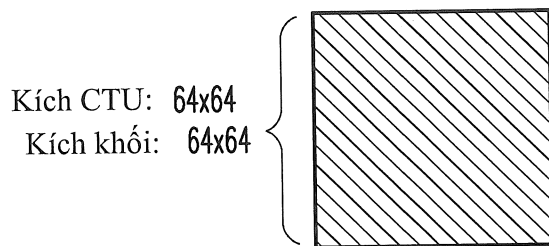
[Fig. 130C]



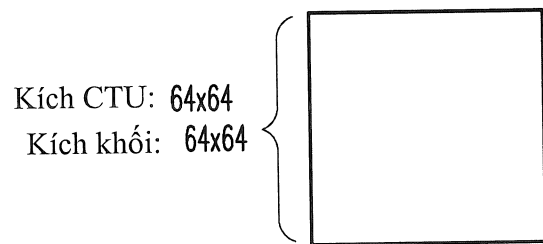
[Fig. 130D]



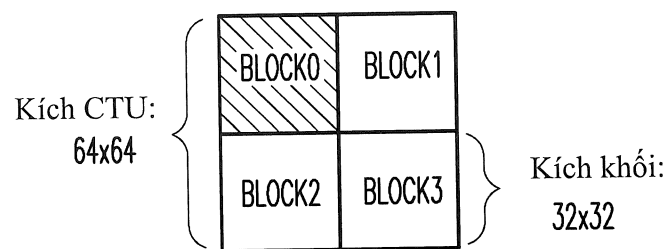
[Fig. 131A]



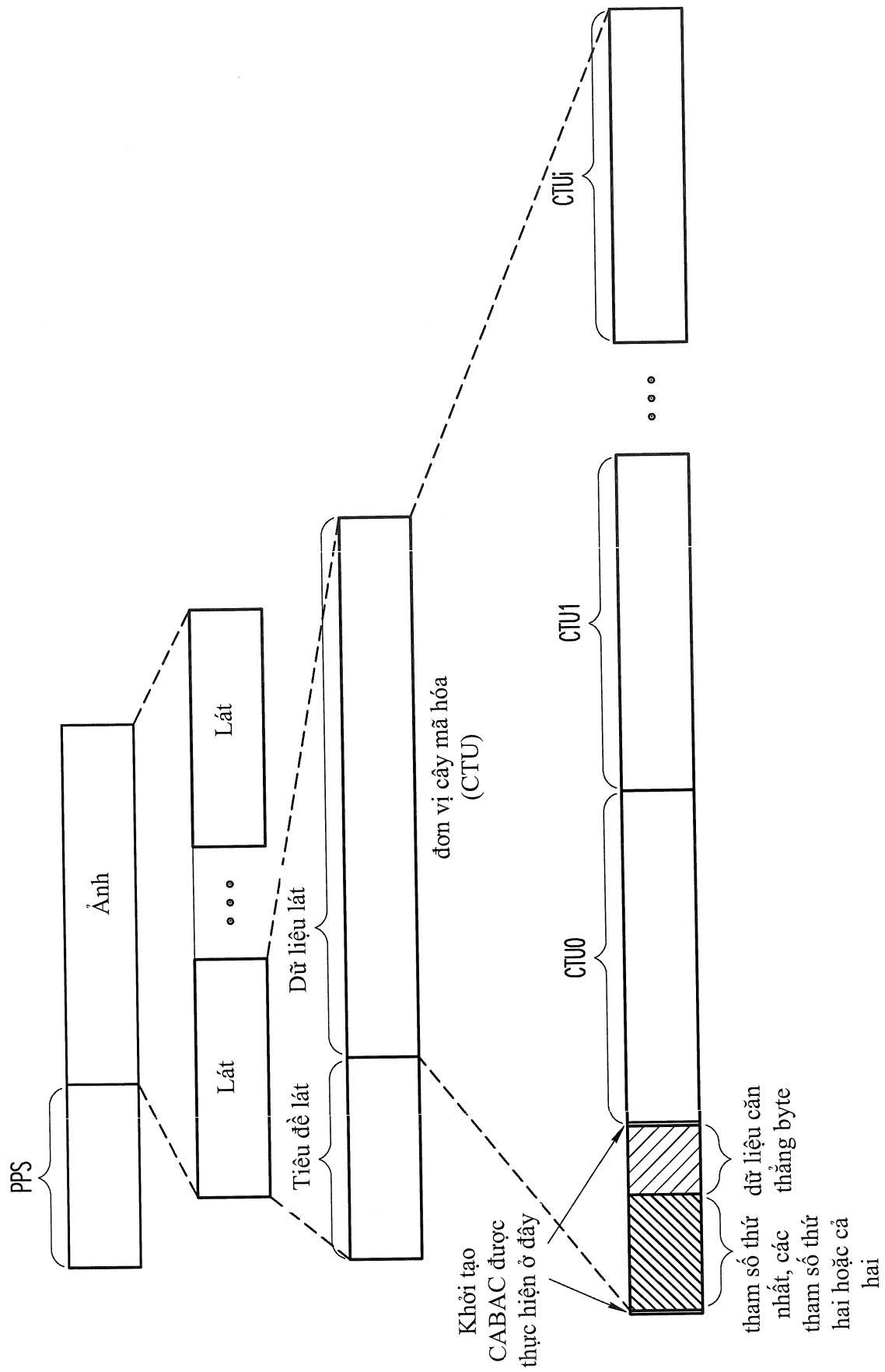
[Fig. 131B]



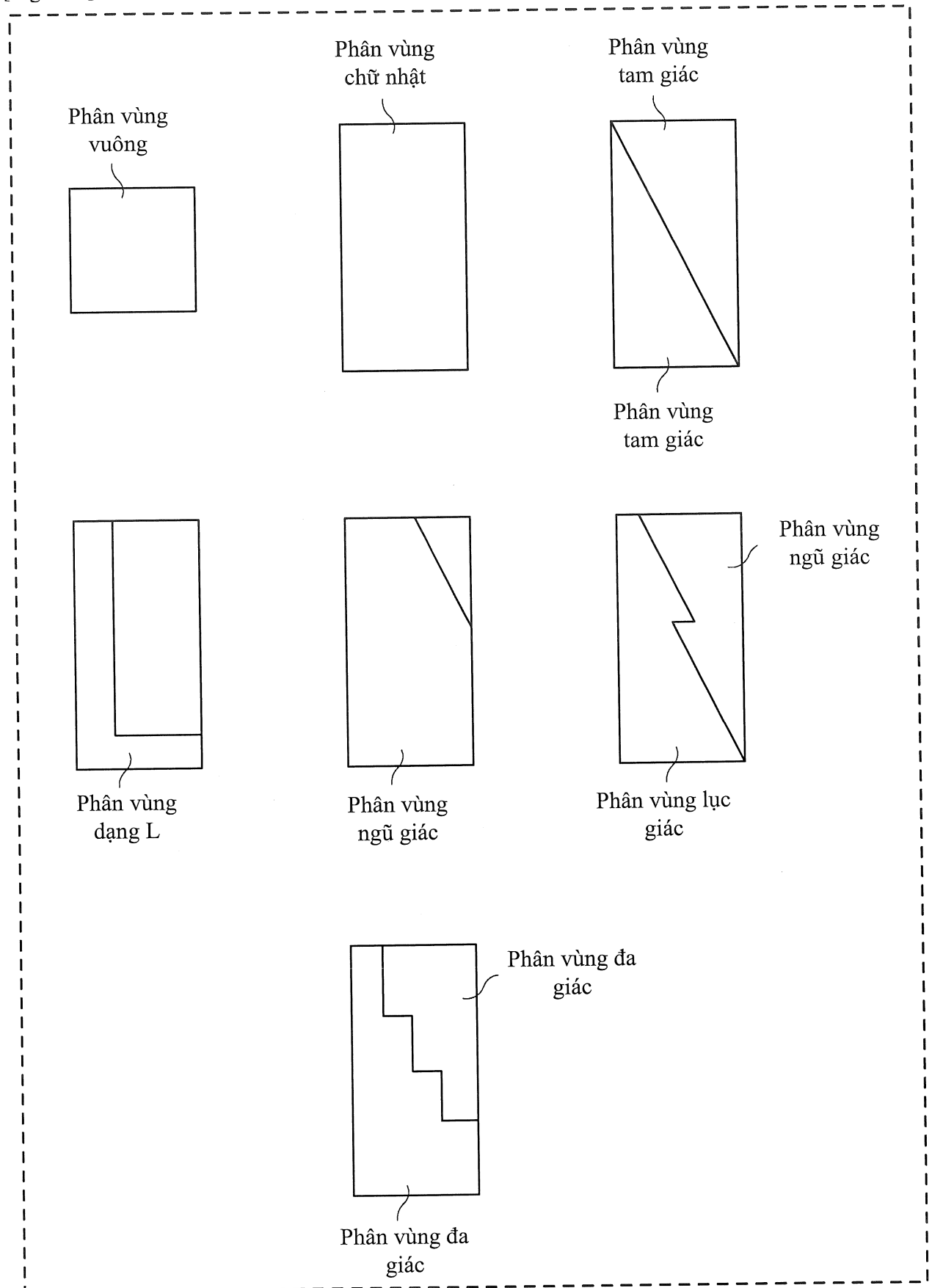
[Fig. 131C]



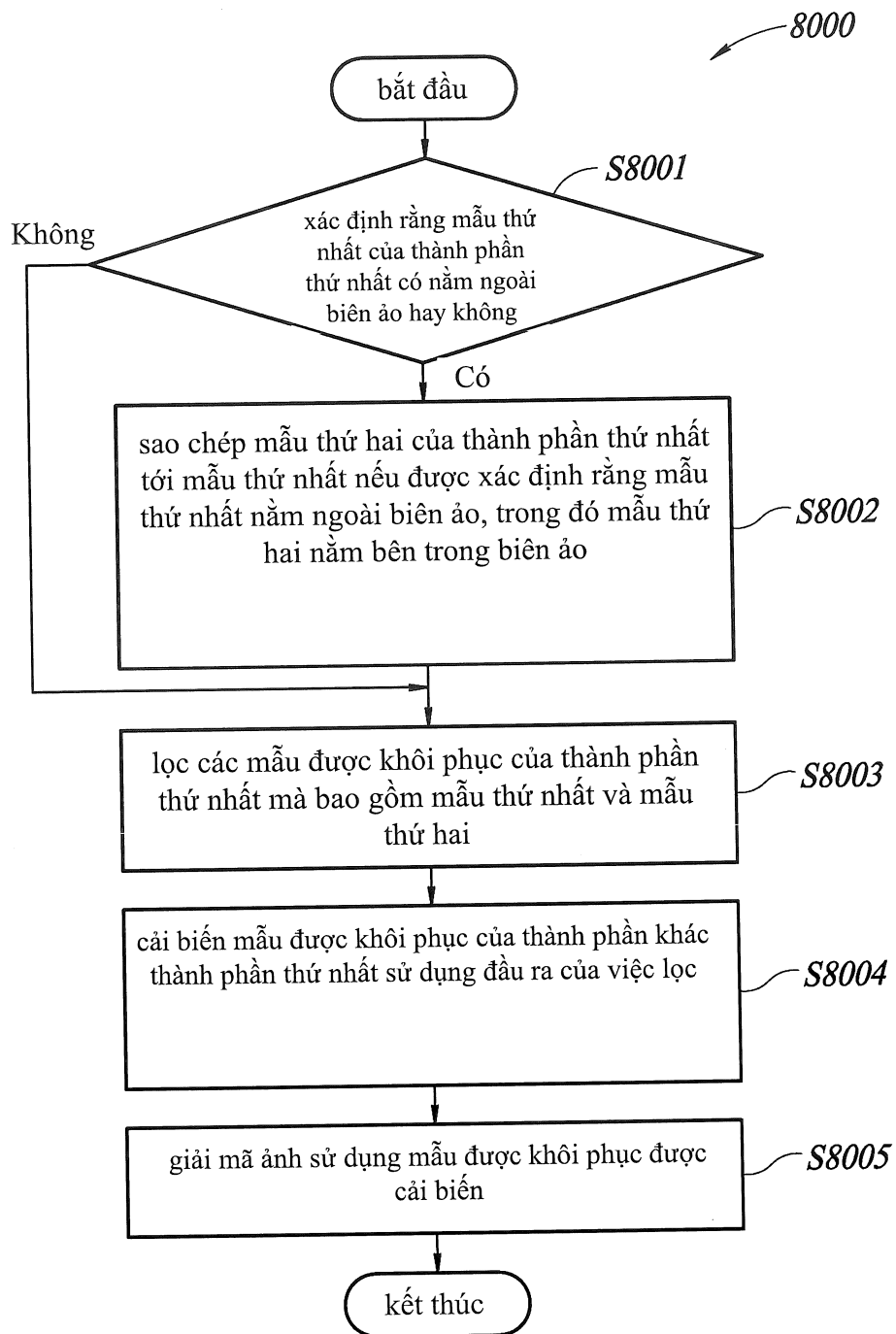
[Fig. 132]



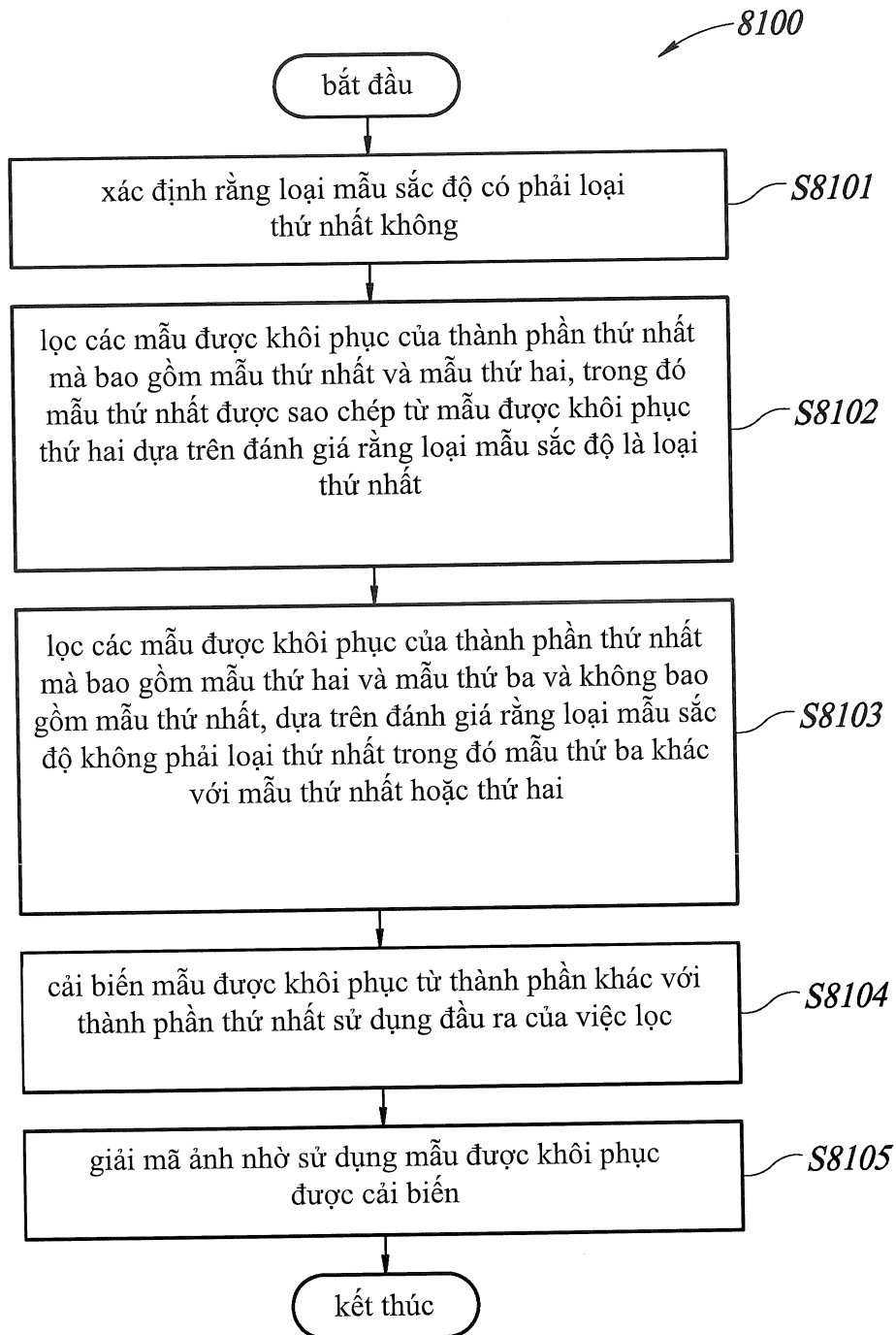
[Fig. 133]



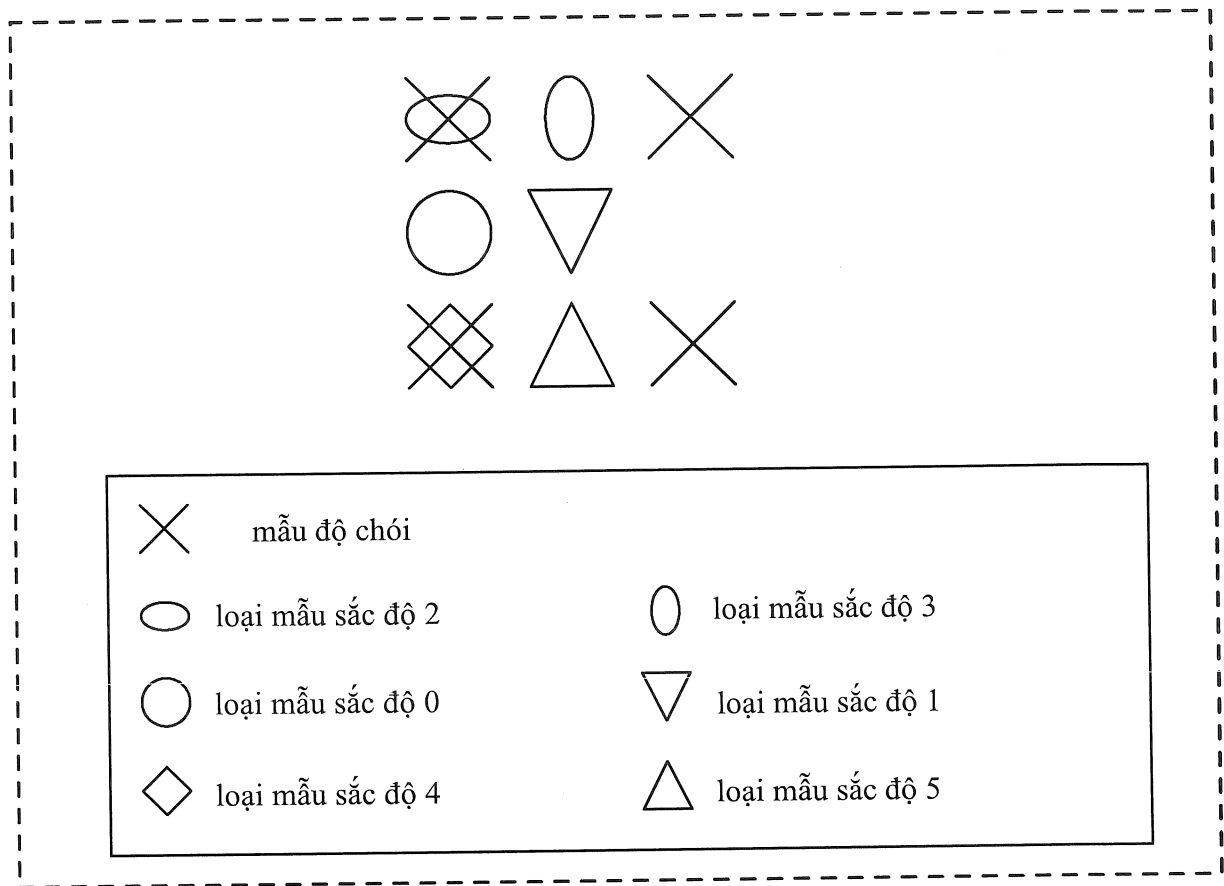
[Fig. 134]



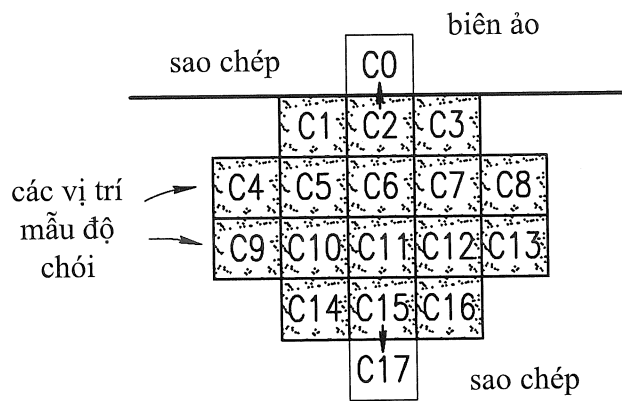
[Fig. 135]



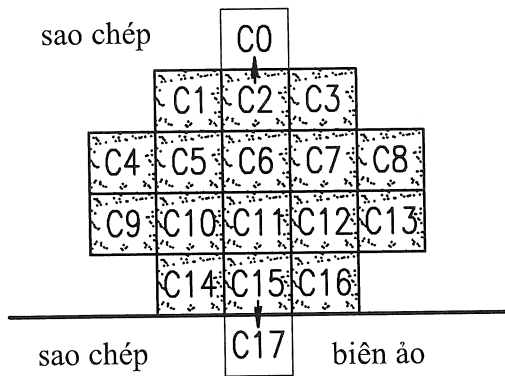
[Fig. 136]



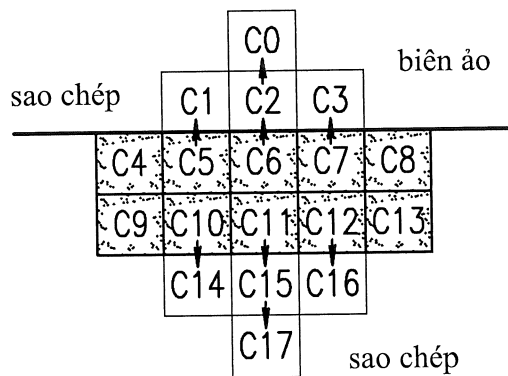
[Fig. 137A]



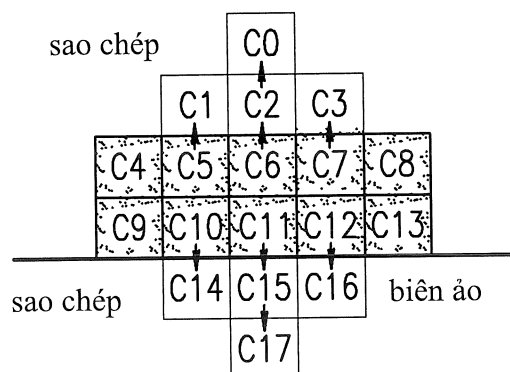
[Fig. 137B]



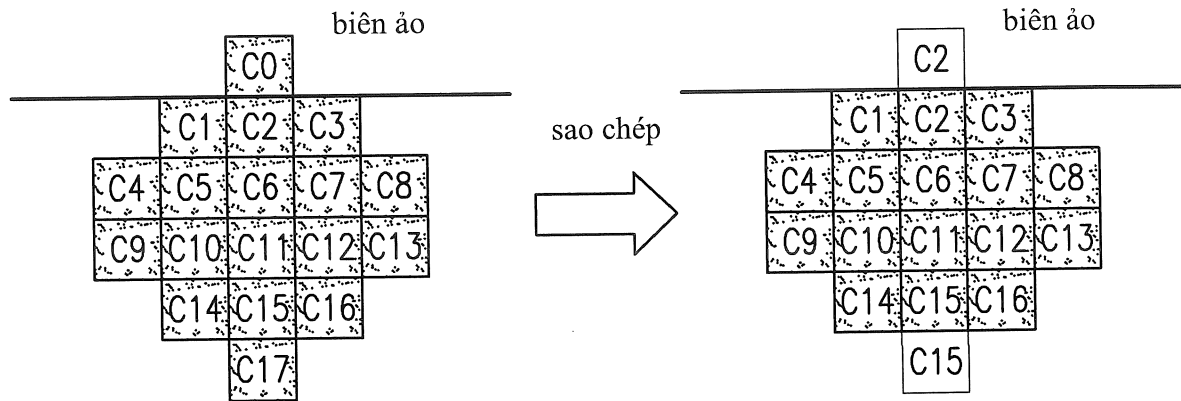
[Fig. 137C]



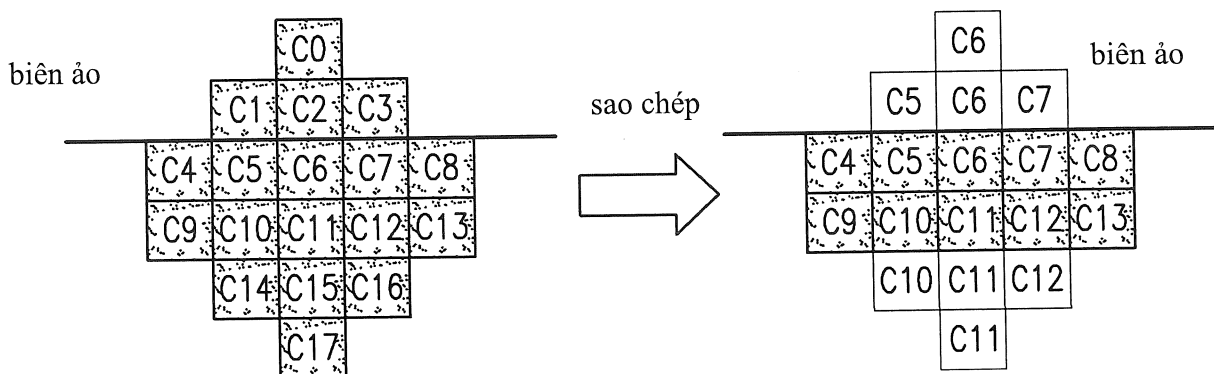
[Fig. 137D]



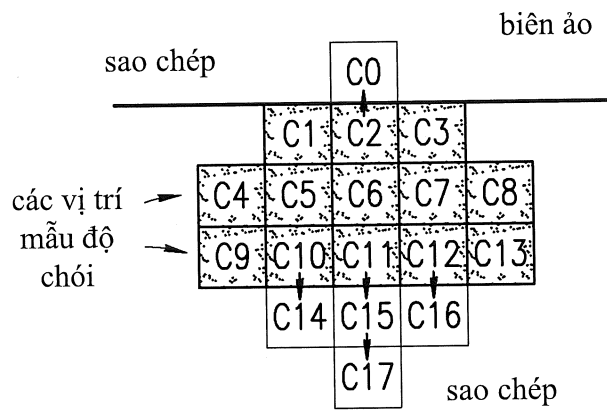
[Fig. 138]



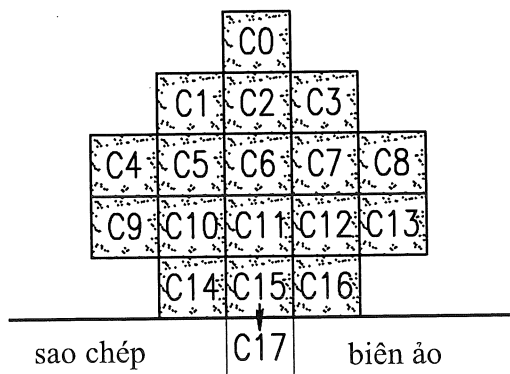
[Fig. 139]



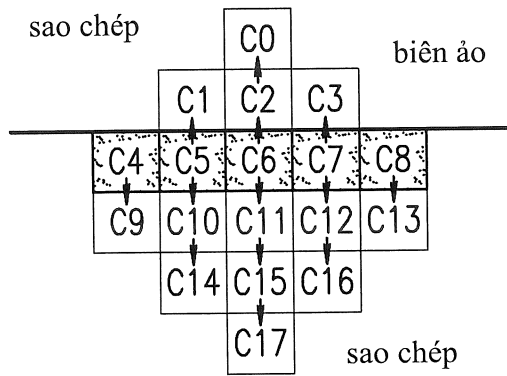
[Fig. 140A]



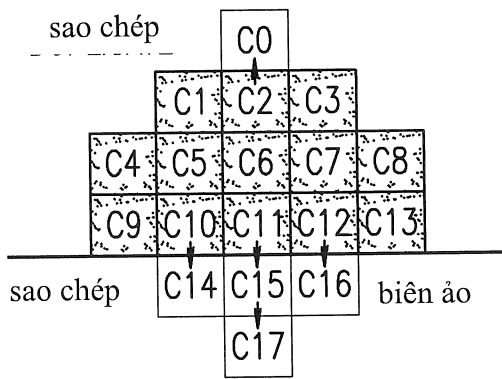
[Fig. 140B]



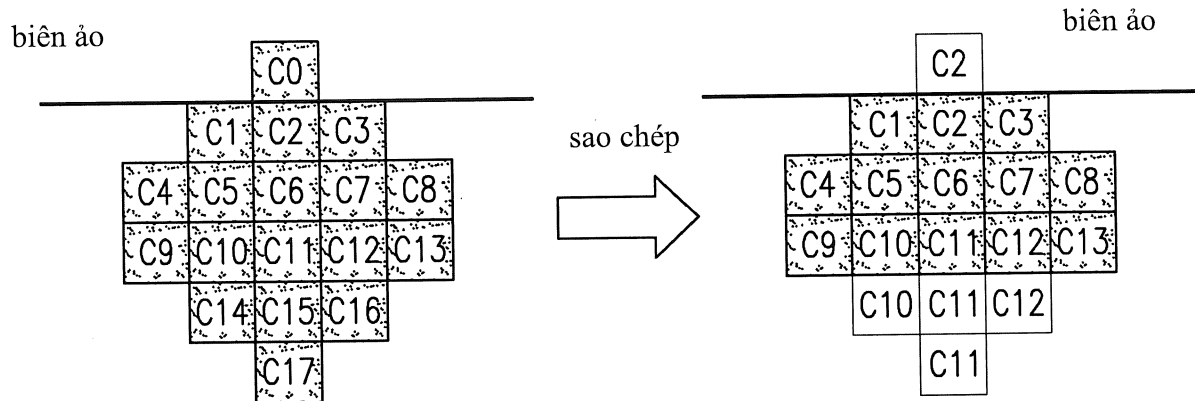
[Fig. 140C]



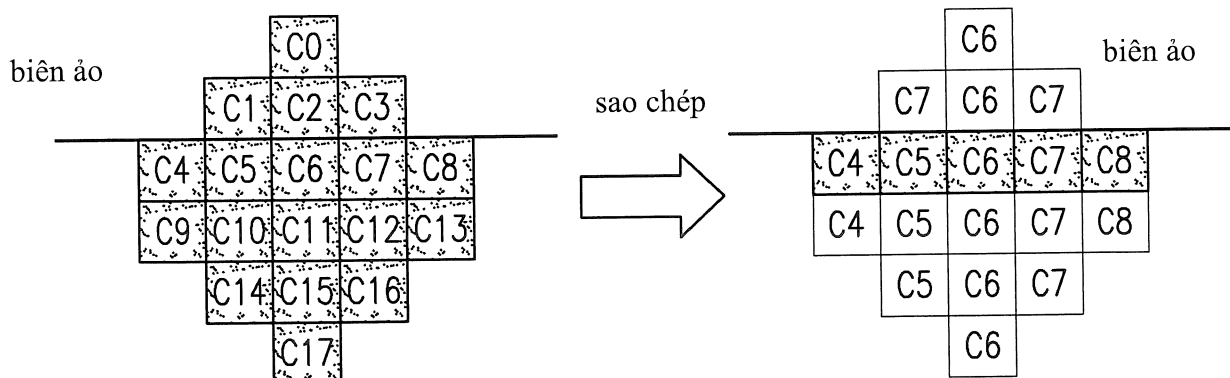
[Fig. 140D]



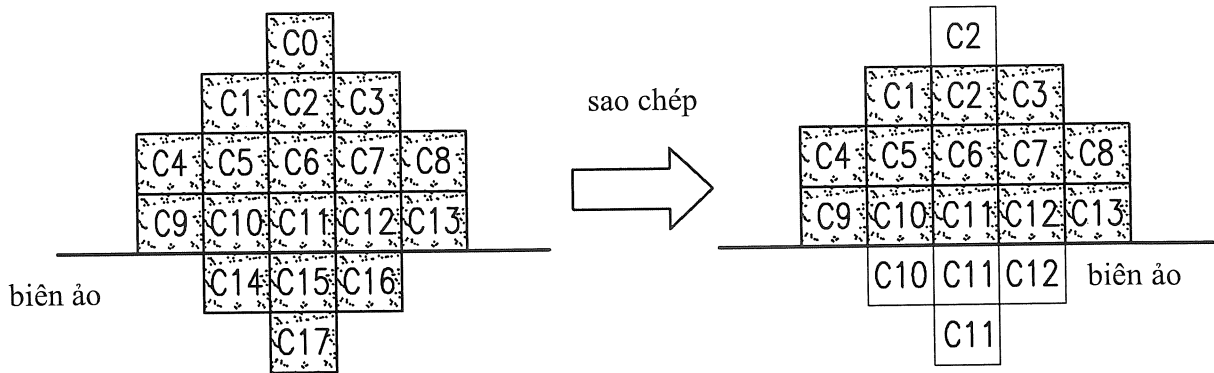
[Fig. 141]



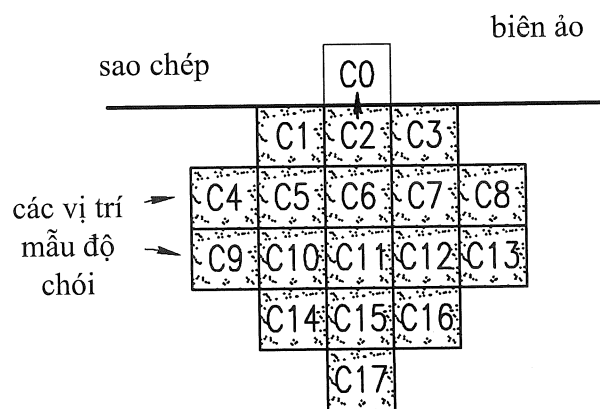
[Fig. 142]



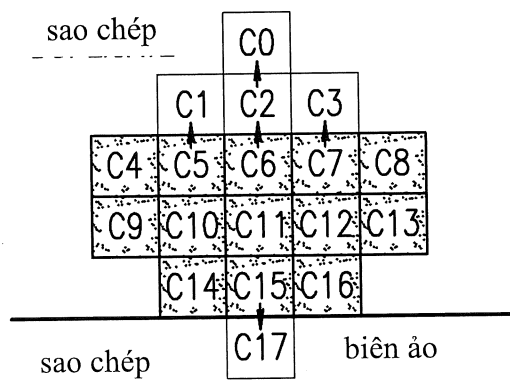
[Fig. 143]



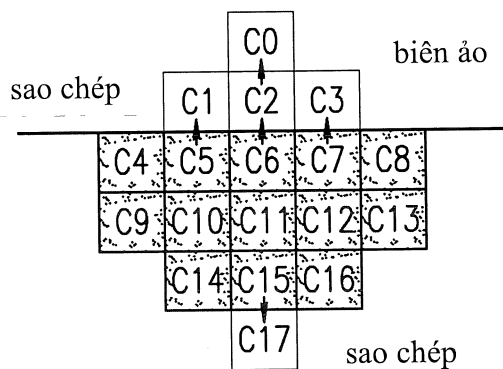
[Fig. 144A]



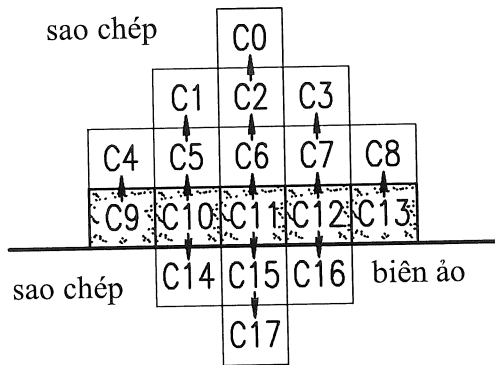
[Fig. 144B]



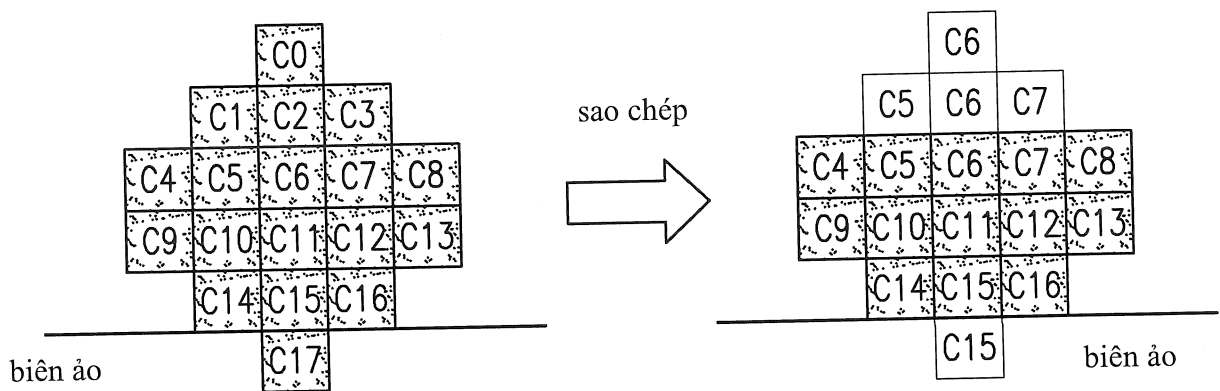
[Fig. 144C]



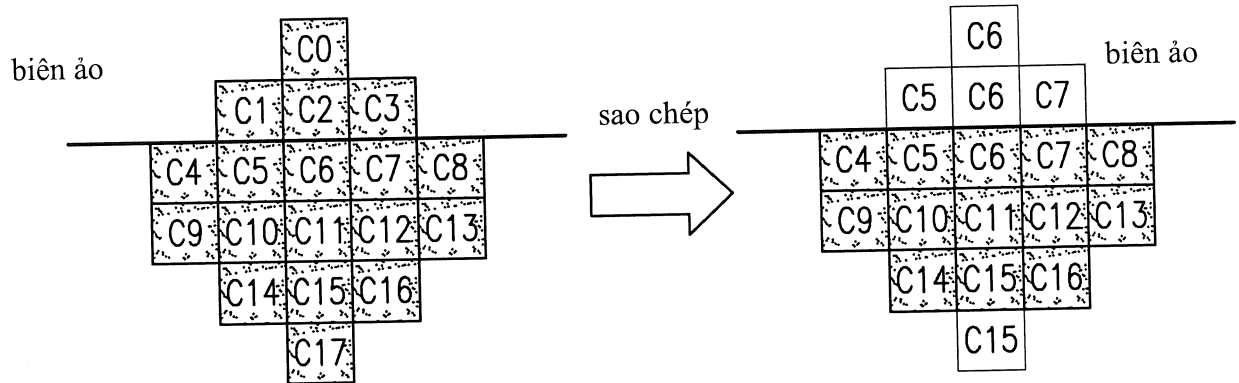
[Fig. 144D]



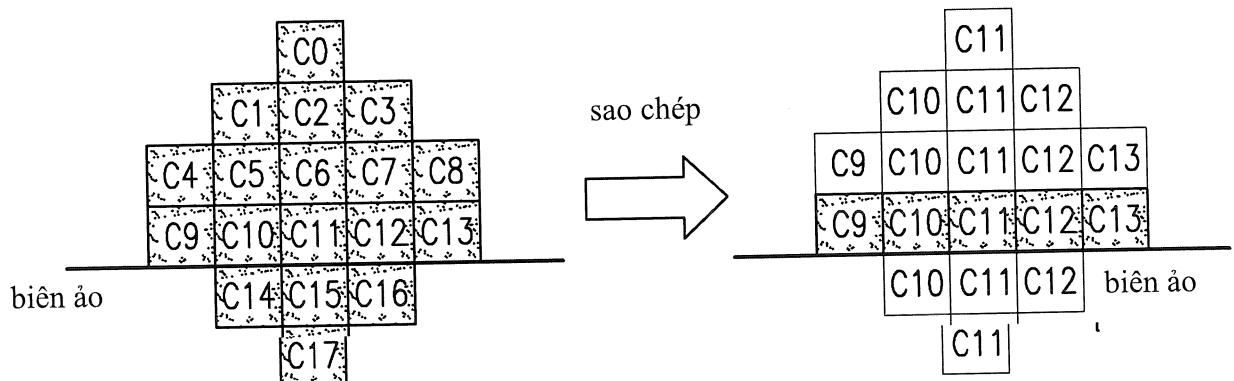
[Fig. 145]



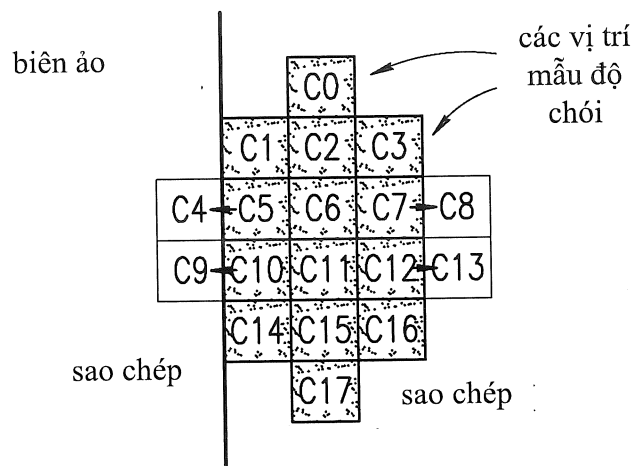
[Fig. 146]



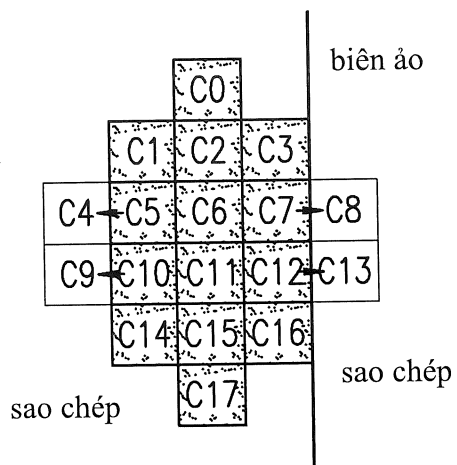
[Fig. 147]



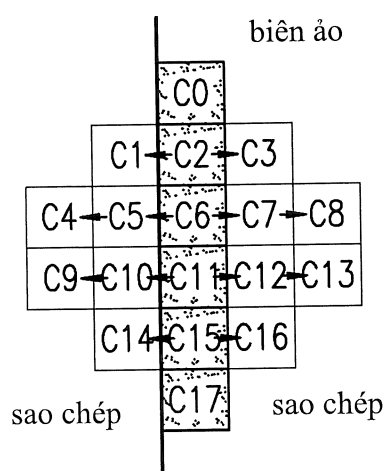
[Fig. 148A]



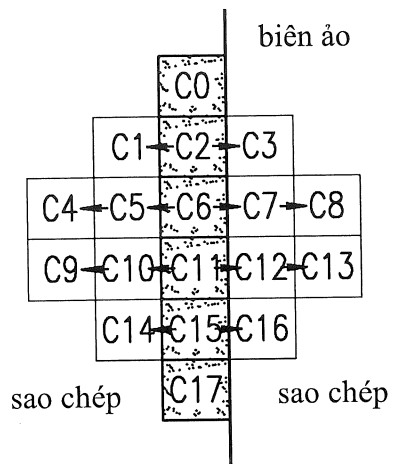
[Fig. 148B]



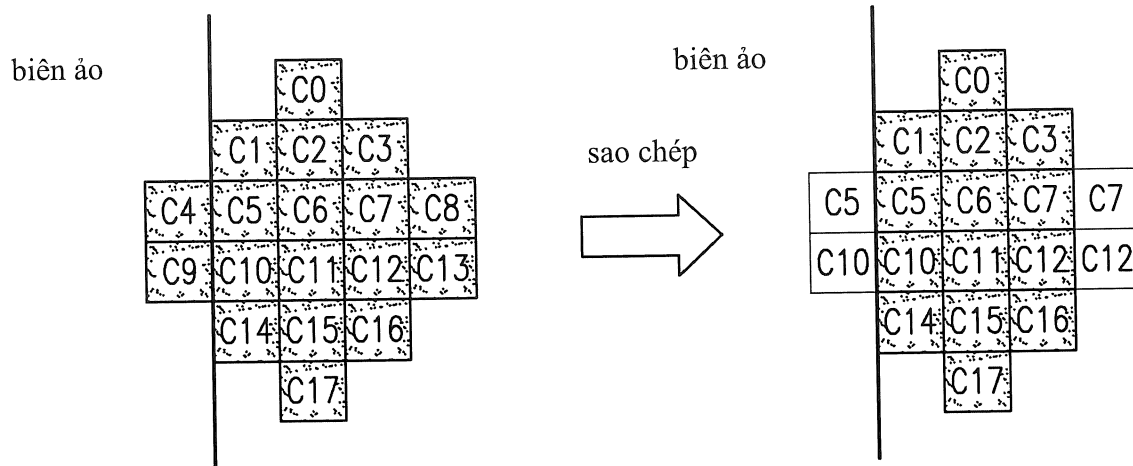
[Fig. 148C]



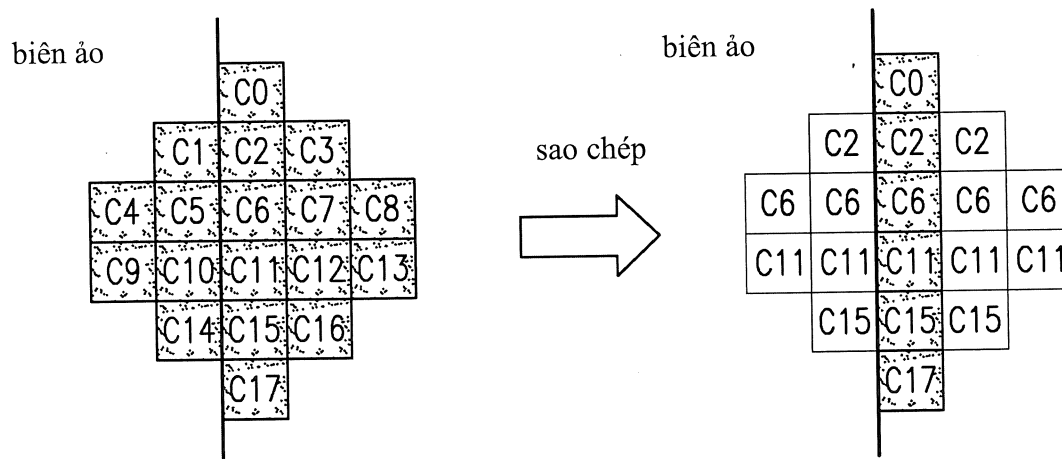
[Fig. 148D]



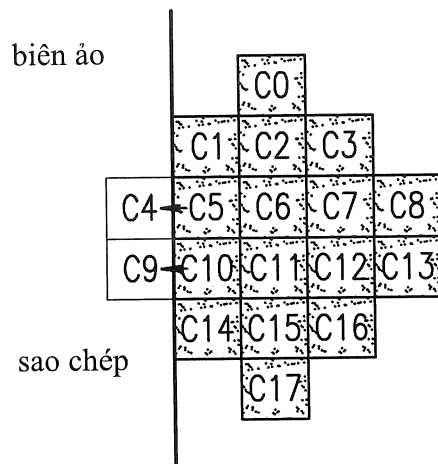
[Fig. 149]



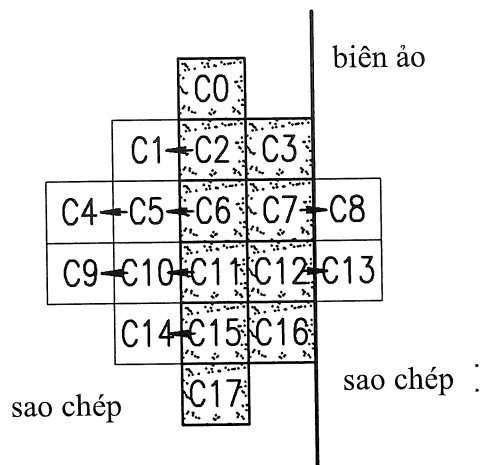
[Fig. 150]



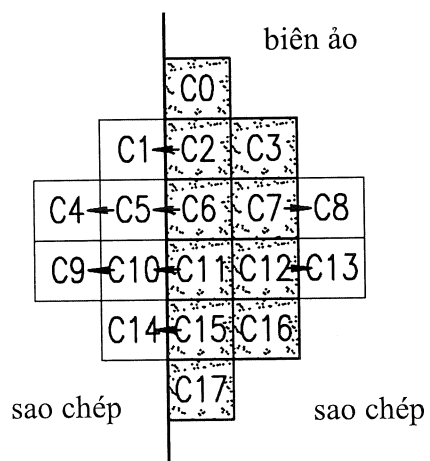
[Fig. 151A]



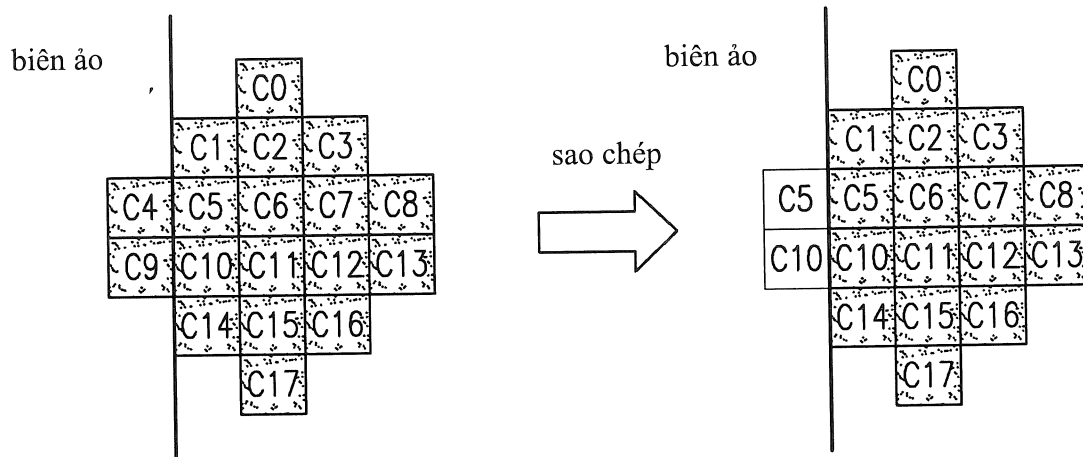
[Fig. 151B]



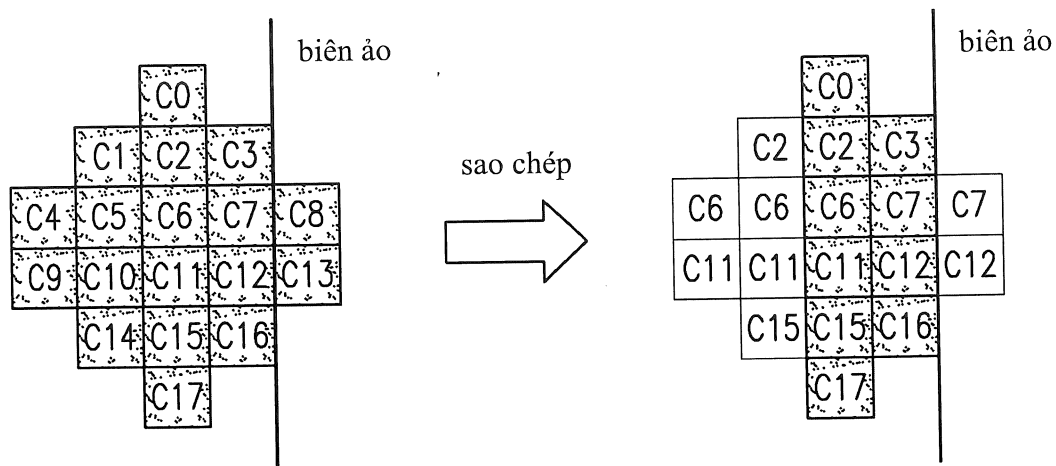
[Fig. 151C]



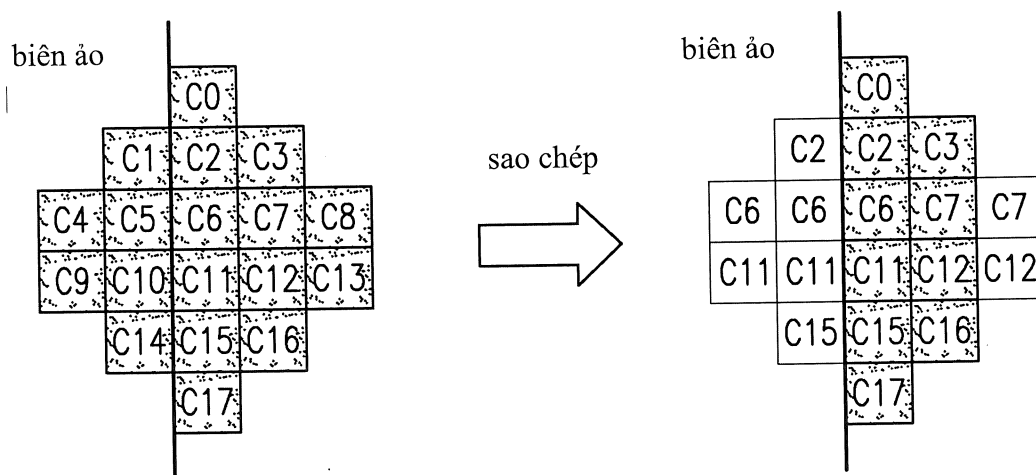
[Fig. 152]



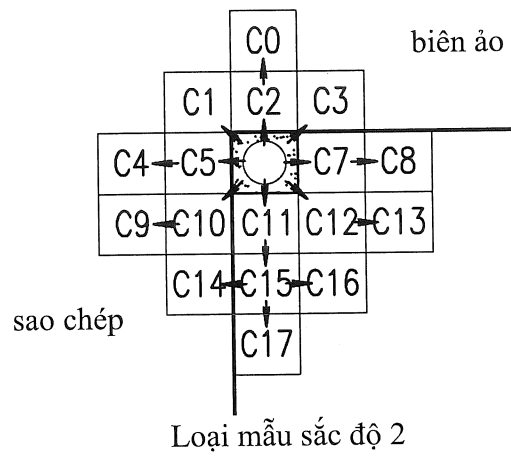
[Fig. 153]



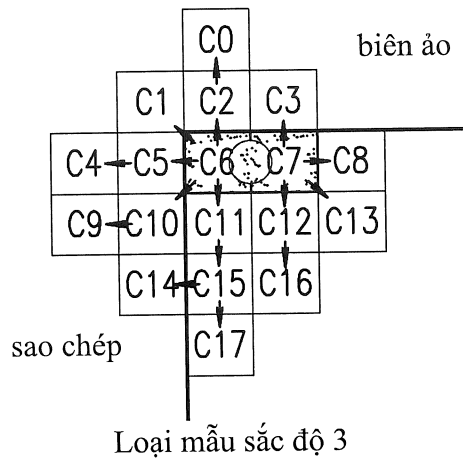
[Fig. 154]



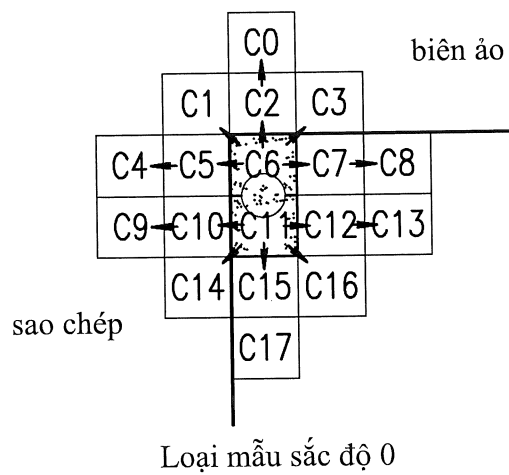
[Fig. 155A]



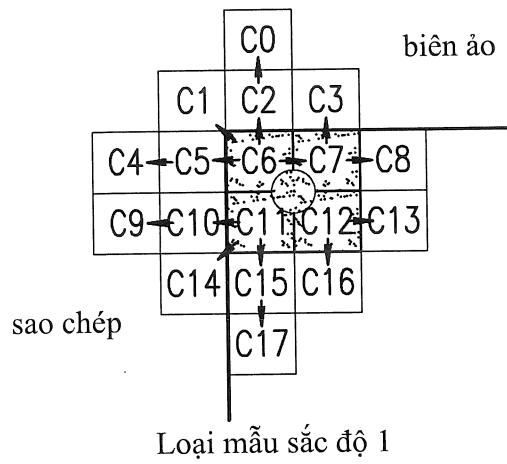
[Fig. 155B]



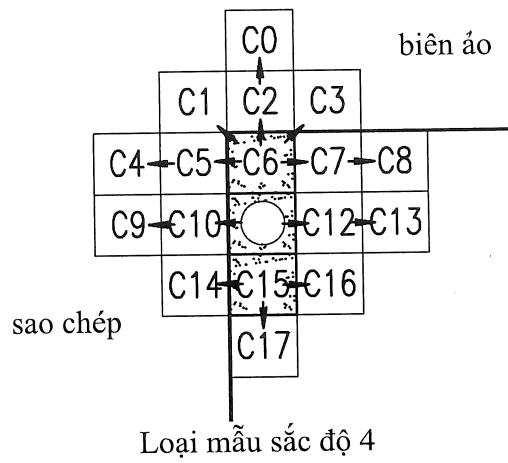
[Fig. 155C]



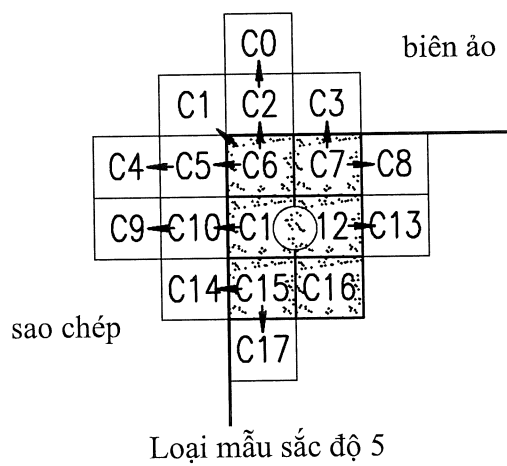
[Fig. 155D]



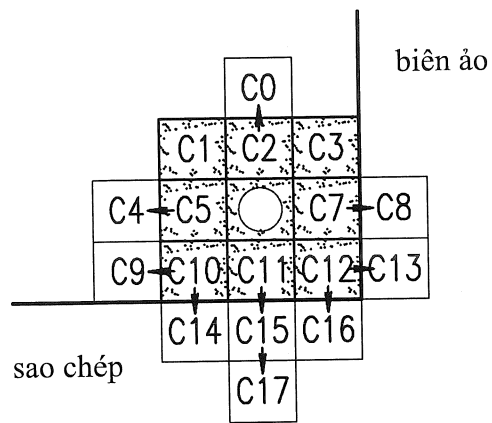
[Fig. 155E]



[Fig. 155F]

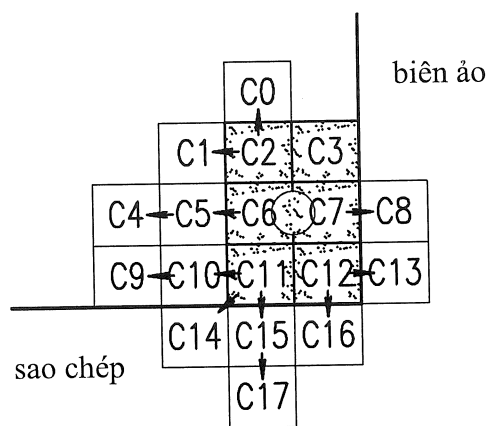


[Fig. 155G]



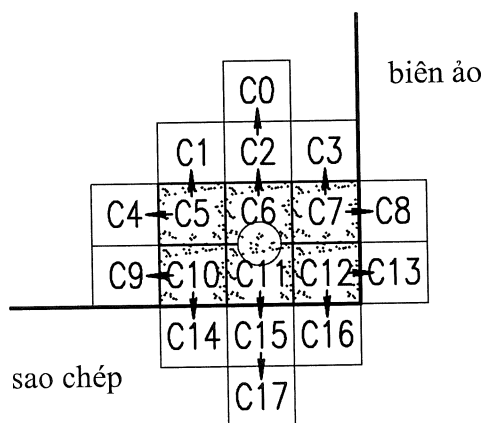
Loại mẫu sắc độ 2

[Fig. 155H]



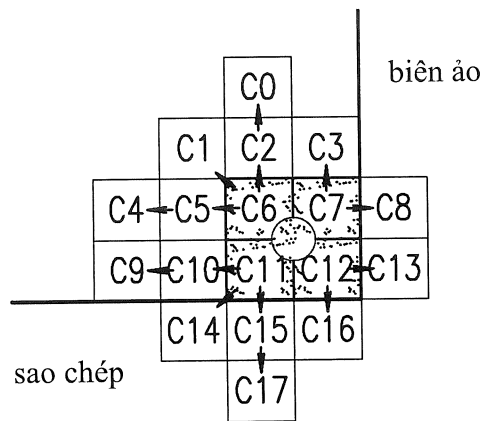
Loại mẫu sắc độ 3

[Fig. 155I]



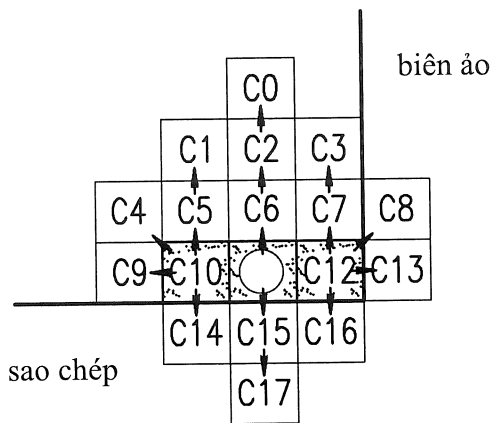
Loại mẫu sắc độ 0

[Fig. 155J]



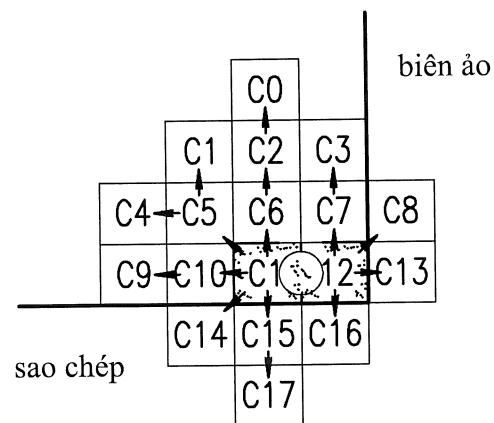
Loại mẫu sắc độ 1

[Fig. 155K]



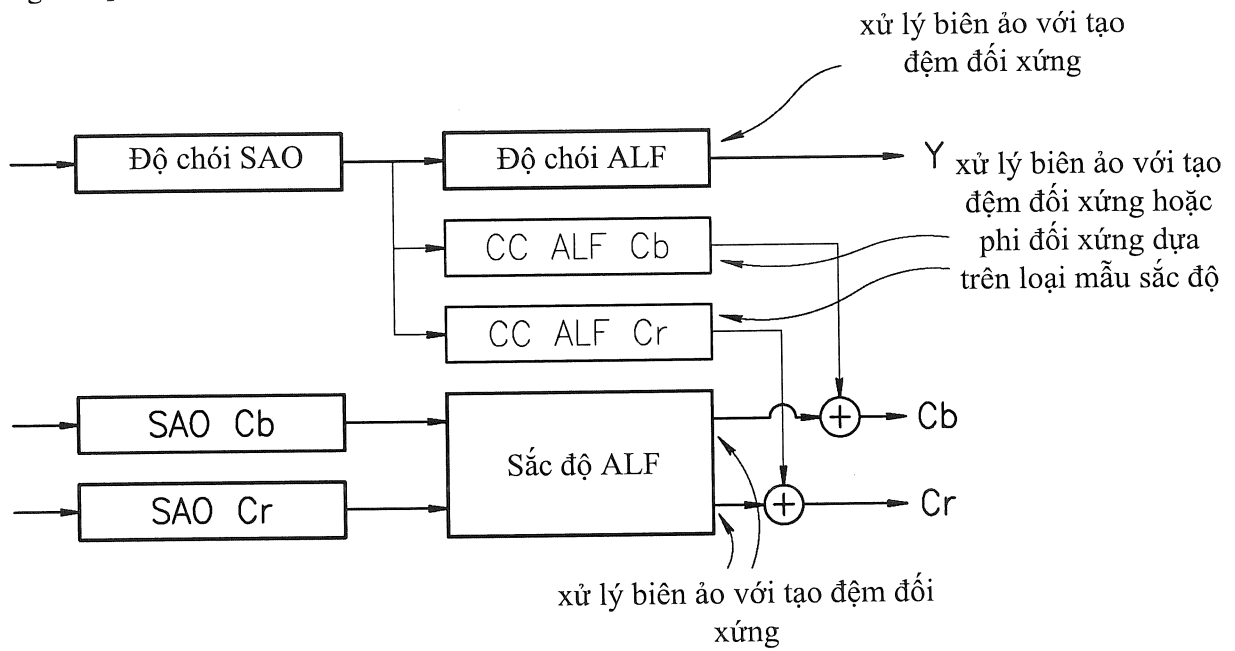
Loại mẫu sắc độ 4

[Fig. 155L]

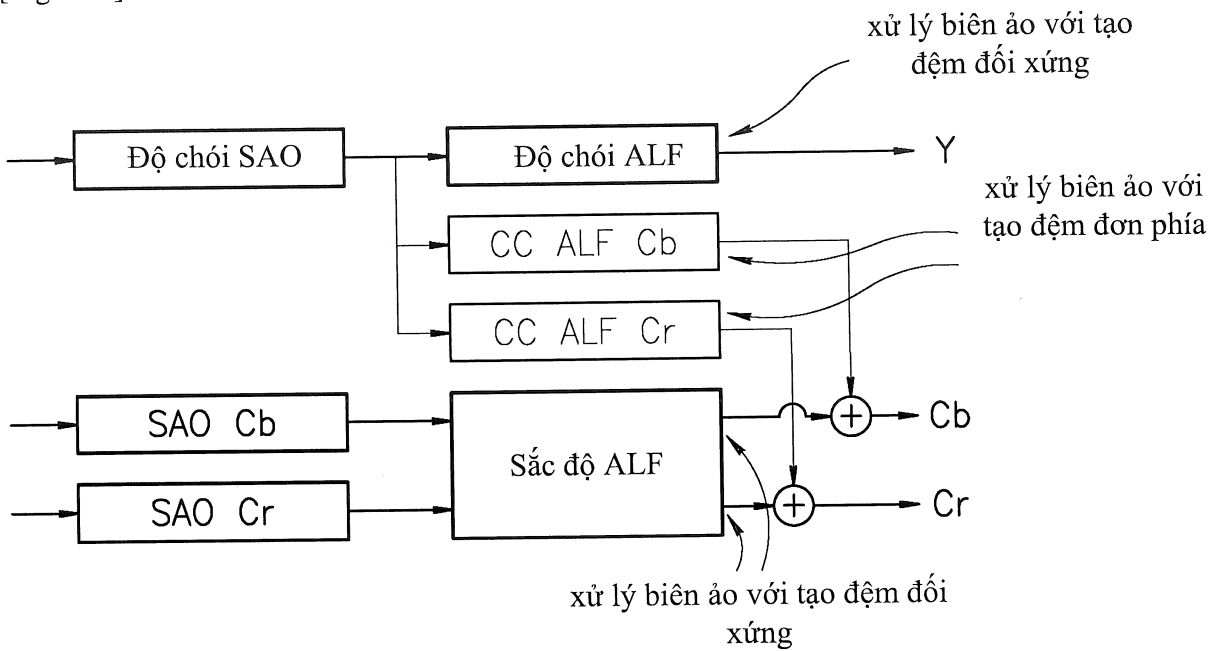


Loại mẫu sắc độ 5

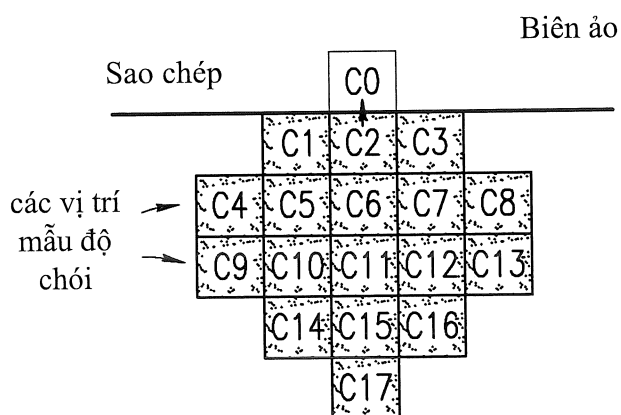
[Fig. 156]



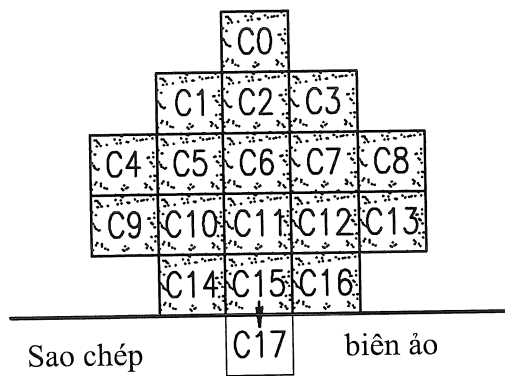
[Fig. 157]



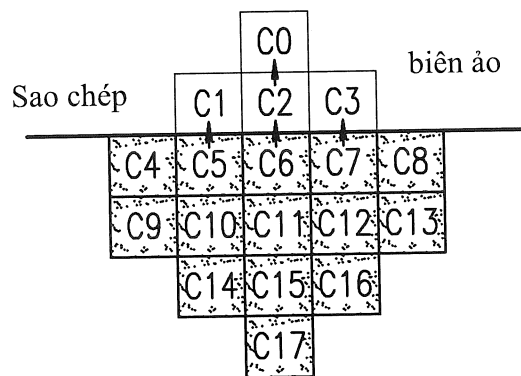
[Fig. 158A]



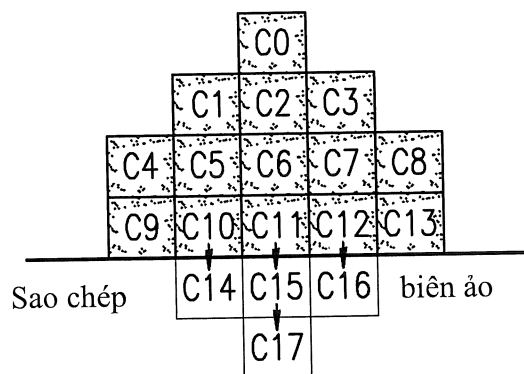
[Fig. 158B]



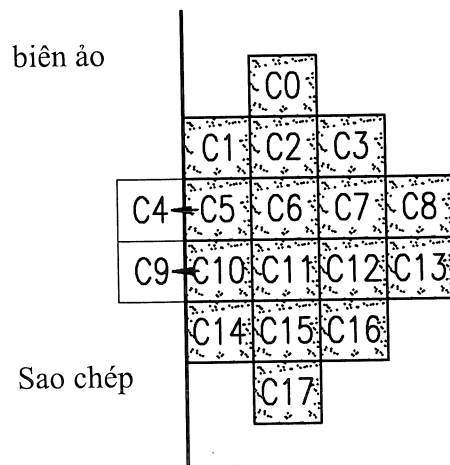
[Fig. 158C]



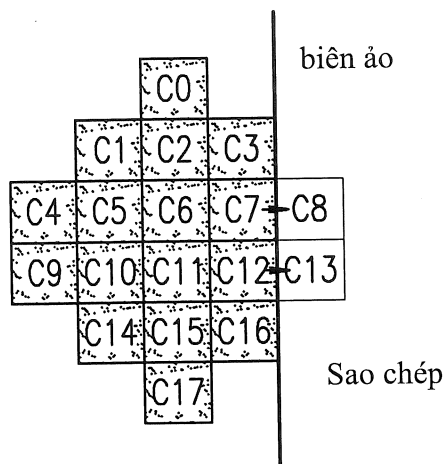
[Fig. 158D]



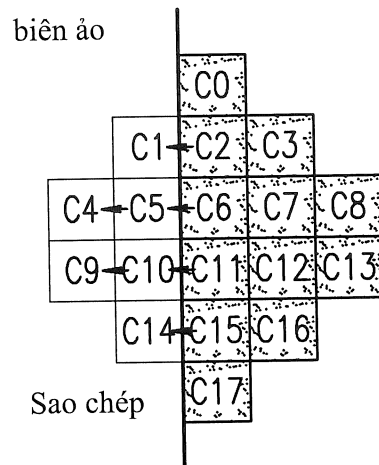
[Fig. 158E]



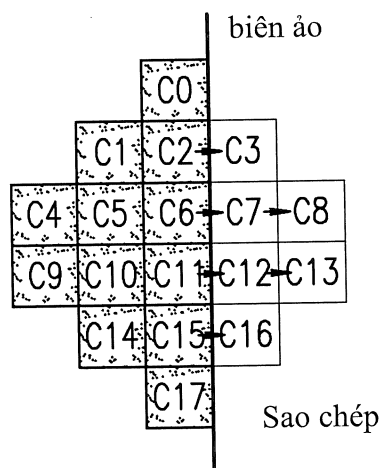
[Fig. 158F]



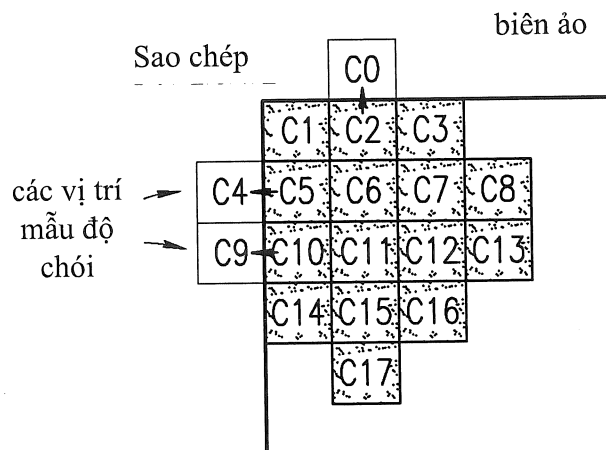
[Fig. 158G]



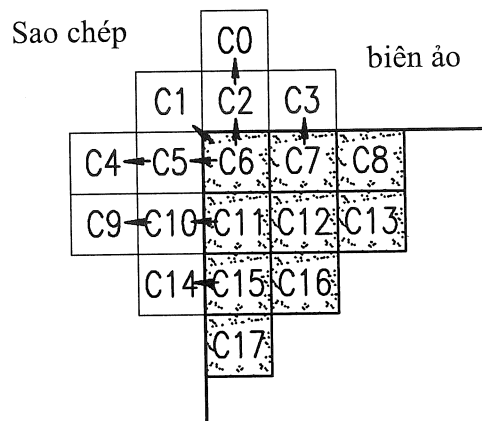
[Fig. 158H]



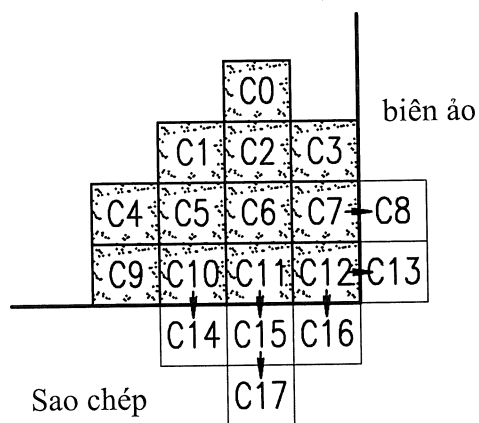
[Fig. 159A]



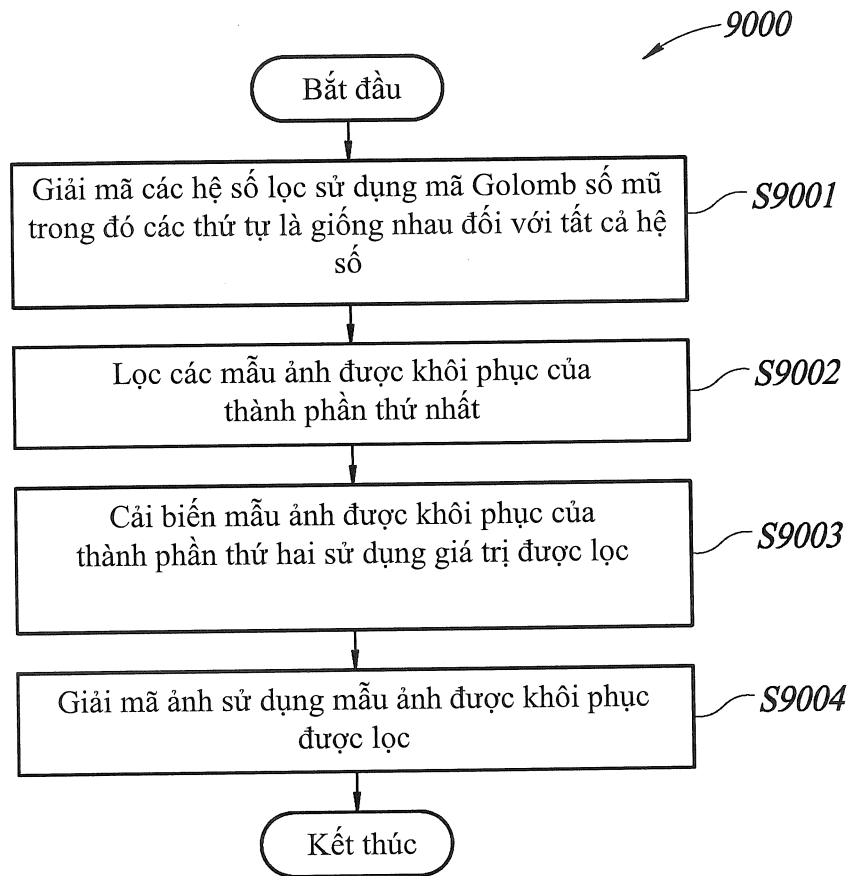
[Fig. 159B]



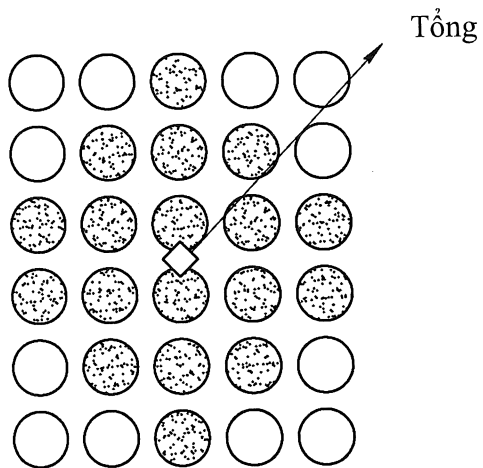
[Fig. 159C]



[Fig. 160]



[Fig. 161]



[Fig. 162]

JVET-02001-V9. Công thức lọc độ chói ALF

Tổng biến thiên được thu nhận như sau:

$$\begin{aligned}
 \text{sum} = & f[\text{idx}[0]] * (\text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_y+r3] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[0]], c[\text{idx}[0]], \text{recPicture}_L[h_x, v_y-r3] - \text{curr})) + \\
 & f[\text{idx}[1]] * (\text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_{x+1}, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[1]], c[\text{idx}[1]], \text{recPicture}_L[h_{x-1}, v_y-r2] - \text{curr})) + \\
 & f[\text{idx}[2]] * (\text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[2]], c[\text{idx}[2]], \text{recPicture}_L[h_x, v_y-r2] - \text{curr})) + \\
 & f[\text{idx}[3]] * (\text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_{x-1}, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[3]], c[\text{idx}[3]], \text{recPicture}_L[h_{x+1}, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[4]] * (\text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_{x+2}, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[4]], c[\text{idx}[4]], \text{recPicture}_L[h_{x-2}, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[5]] * (\text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_{x+1}, v_y+r] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[5]], c[\text{idx}[5]], \text{recPicture}_L[h_{x-1}, v_y-r] - \text{curr})) + \\
 & f[\text{idx}[6]] * (\text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[6]], c[\text{idx}[6]], \text{recPicture}_L[h_x, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[7]] * (\text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_{x-1}, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[7]], c[\text{idx}[7]], \text{recPicture}_L[h_{x+1}, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[8]] * (\text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_{x-2}, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[8]], c[\text{idx}[8]], \text{recPicture}_L[h_{x+2}, v_y-r1] - \text{curr})) + \\
 & f[\text{idx}[9]] * (\text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_{x+3}, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[9]], c[\text{idx}[9]], \text{recPicture}_L[h_{x-3}, v_y] - \text{curr})) + \\
 & f[\text{idx}[10]] * (\text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_{x+2}, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[10]], c[\text{idx}[10]], \text{recPicture}_L[h_{x-2}, v_y] - \text{curr})) + \\
 & f[\text{idx}[11]] * (\text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_{x+1}, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[\text{idx}[11]], c[\text{idx}[11]], \text{recPicture}_L[h_{x-1}, v_y] - \text{curr})) + \\
 & \text{sum} = \text{curr} + ((\text{sum} + 64) >> 7)
 \end{aligned} \tag{8-1187}$$

(8-1188)

JVET-02001-V9. Công thức lọc sắc độ ALF

Tổng biến thiên được thu nhận như sau:

$$\begin{aligned}
 \text{sum} = & f[0] * (\text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_y+r2] - \text{curr}) + \\
 & \text{Clip3}(-c[0], c[0], \text{recPicture}_L[h_x, v_y+r2] - \text{curr})) + \\
 & f[1] * (\text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_{x+1}, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[1], c[1], \text{recPicture}_L[h_{x-1}, v_y-r1] - \text{curr})) + \\
 & f[2] * (\text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[2], c[2], \text{recPicture}_L[h_x, v_y-r1] - \text{curr})) + \\
 & f[3] * (\text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_{x-1}, v_y+r1] - \text{curr}) + \\
 & \text{Clip3}(-c[3], c[3], \text{recPicture}_L[h_{x+1}, v_y-r1] - \text{curr})) + \\
 & f[4] * (\text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_{x+2}, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[4], c[4], \text{recPicture}_L[h_{x-2}, v_y] - \text{curr})) + \\
 & f[5] * (\text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_{x+1}, v_y] - \text{curr}) + \\
 & \text{Clip3}(-c[5], c[5], \text{recPicture}_L[h_{x-1}, v_y] - \text{curr})) + \\
 & \text{sum} = \text{curr} + ((\text{sum} + 64) >> 7)
 \end{aligned} \tag{8-1238}$$

(8-1239)

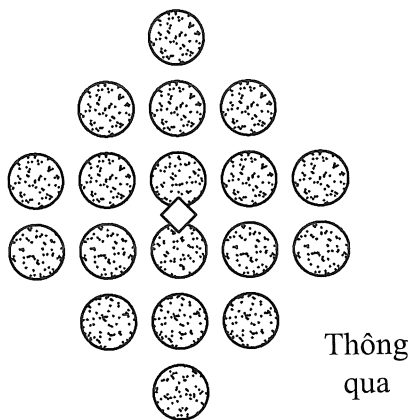
[Fig. 163]

for (j=0; j < 14;j++) {
alf_cross_component_cb_coeff_abs[j]
if(alf_cross_component_cb_coeff_abs[j])
alf_cross_component_cb_coeff_sign[j]
}
...
for (j=0; j < 14;j++) {
alf_cross_component_cr_coeff_abs[j]
if(alf_cross_component_cr_coeff_abs[j])
alf_cross_component_cr_coeff_sign[j]
}

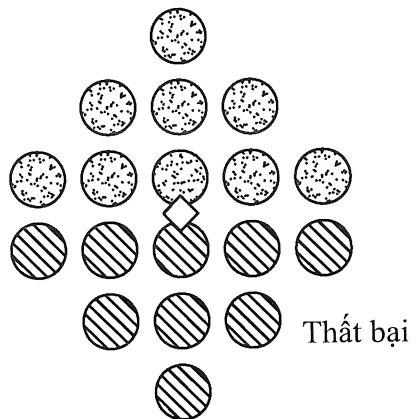
[Fig. 164]

X	k=0	k=1	k=2	k=3
0	1	1001	100	1000
1	010	1100	101	1001
2	011	01001011	110	1010
3	00100	01011010	111	1011
4	00101	0110...	01000	1100
5	00110	0111...	01001	1101
6	00111	001000	01010	1110
7	0001000	001001	01011	1111
8	0001001	001010	01100	010000
9	0001010	001011	01101	010001
10	0001011	001100	01110	010010
11	0001100	001101	01111	010011
12	0001101	001110	0010000	010100
13	0001110	001111	0010001	010101
14	0001111	0001000	0010010	010110
15	000010000	0001001	0010011	010111
16	000010001	00010010	0010100	011000
17	000010010	00010011	0010101	011001
18	000010011	00010100	0010110	011010
19	000010100	00010101	0010111	011011
20	000010101	00010110	0011000	011100
21	000010110	00010111	0011001	011101
22	000010111	00011000	0011010	011110
23	000011000	00011001	0011011	011111
24	000011001	00011010	0011100	00100000
25	000011010	00011011	0011101	00100001
26	000011011	00011100	0011110	00100010
27	000011100	00011101	0011111	00100011
28	000011101	00011110	000100000	00100100
29	000011110	00011111	000100001	00100101

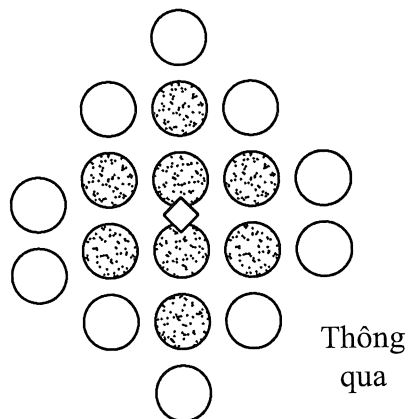
[Fig. 165A]



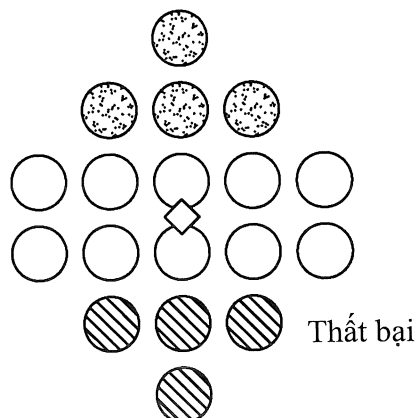
[Fig. 165B]



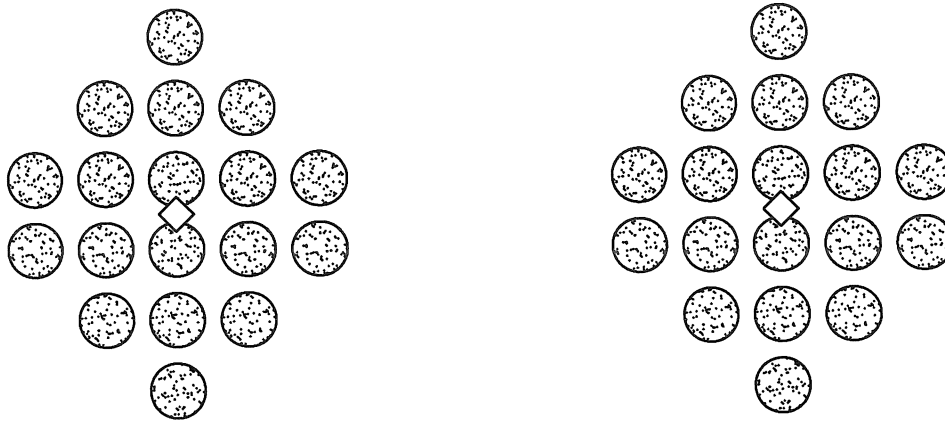
[Fig. 165C]



[Fig. 165D]

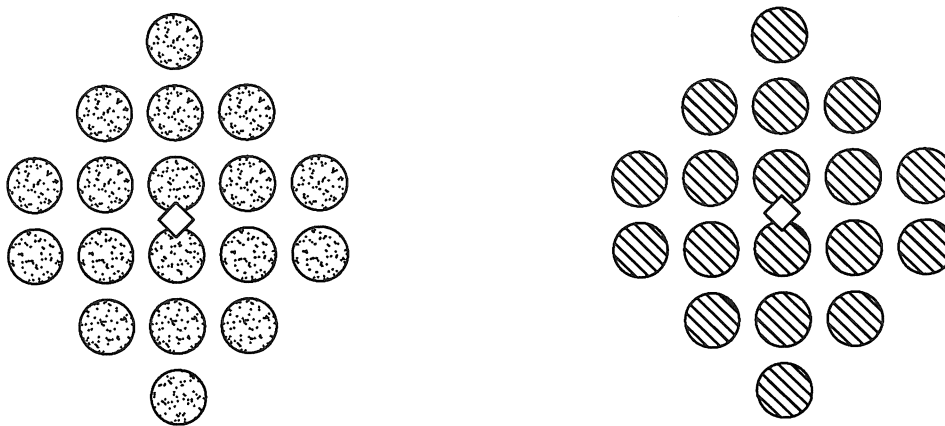


[Fig. 166A]



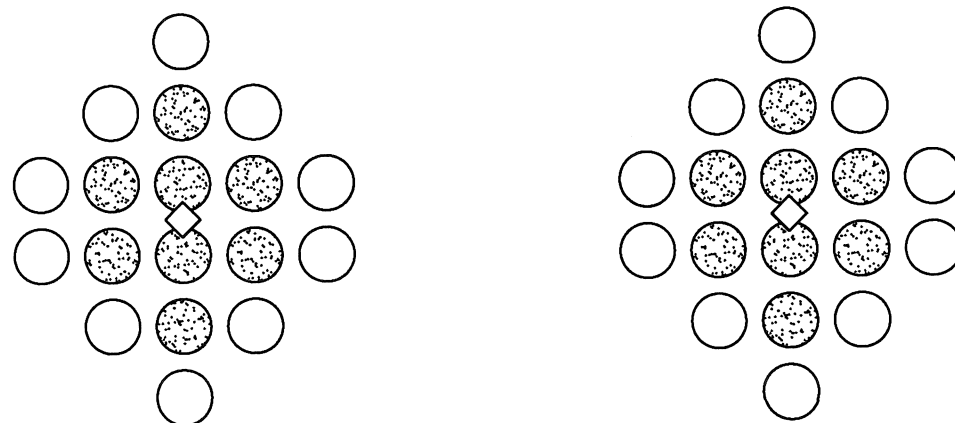
EG3 đối với cb_coeff_abs và EG3 đối với cr_coeff_abs

[Fig. 166B]



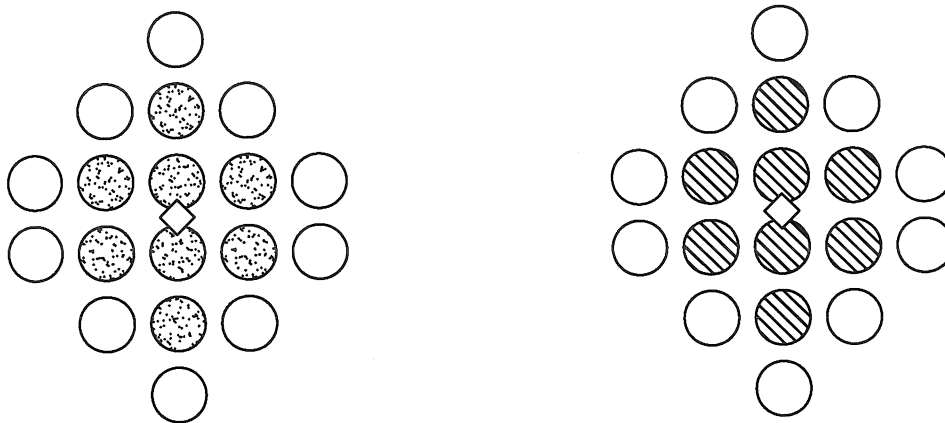
EG3 đối với cb_coeff_abs nhưng EG2 đối với cr_coeff_abs

[Fig. 167A]



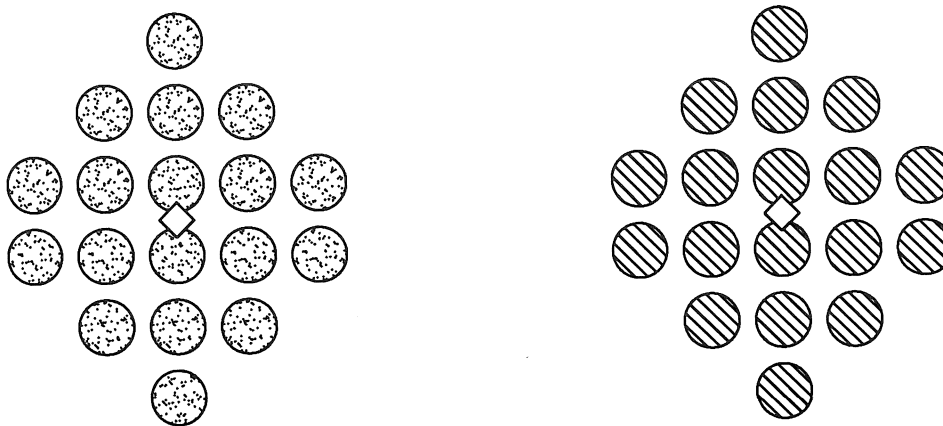
EG3 đối với cb_coeff_abs và EG3 đối với cr_coeff_abs

[Fig. 167B]



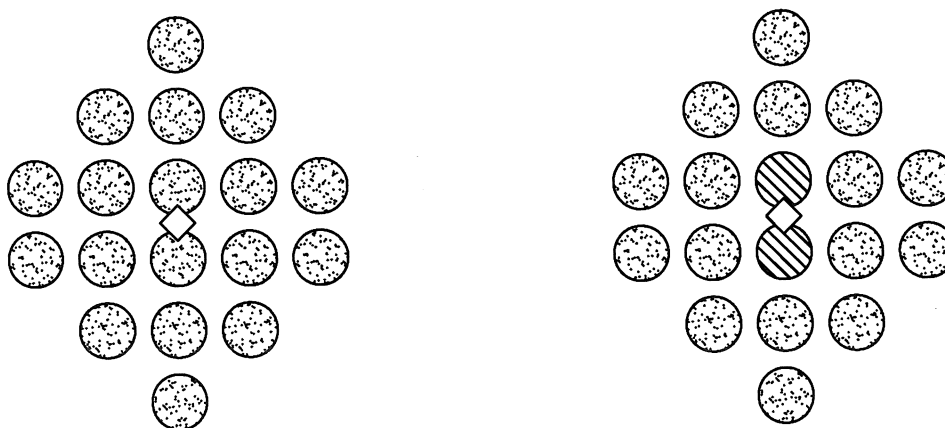
EG3 đối với cb_coeff_abs nhưng EG2 đối với cr_coeff_abs

[Fig. 168A]



EG3 đối với CTU1 nhưng EG2 đối với CTU2

[Fig. 168B]



thất bại đối với CTU 2

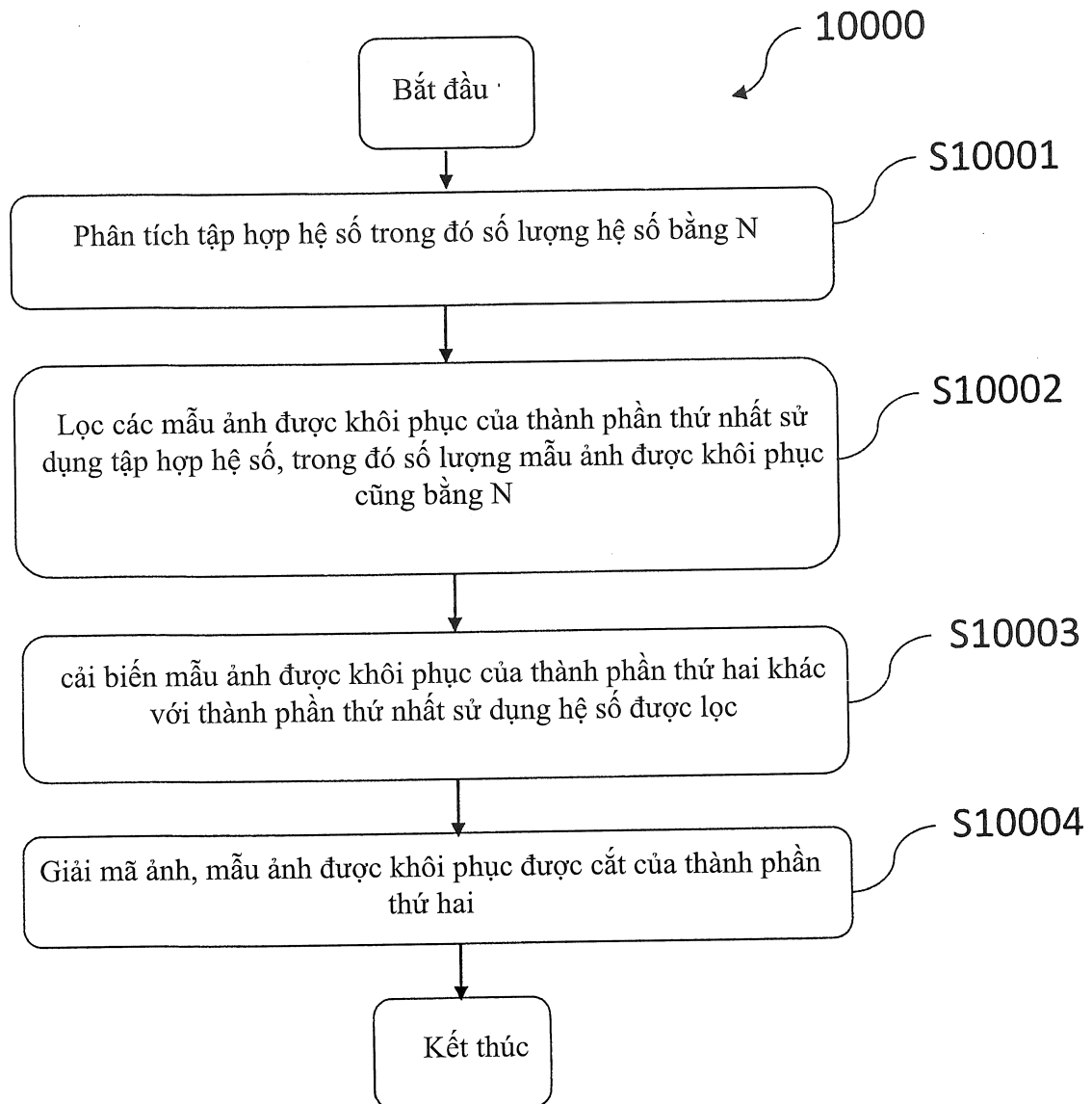
[Fig. 169]

for (j=0; j < 12;j++) {
alf_luma_abs[sfIdx][j]
if(alf_luma_abs[sfIdx][j])
alf_luma_coeff_sign[sfIdx][j]
}
...
for (j=0; j < 6;j++) {
alf_chroma_coeff_abs[altIdx[j]
if(alf_chroma_coeff_abs[altIdx][j] >0)
alf_chroma_coeff_sign[altIdx][j]
}

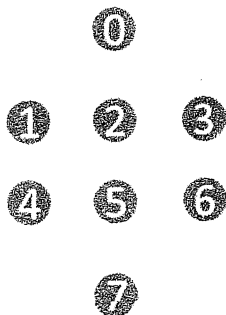
[Fig. 170]

for (j=0; j < 14;j++) {
alf_cross_component_cb_coeff_abs[j]
if(alf_cross_component_cb_coeff_abs[j])
alf_cross_component_cb_coeff_sign[j]
}
...
for (j=0; j < 14;j++) {
alf_cross_component_cr_coeff_abs[j]
if(alf_cross_component_cr_coeff_abs[j])
alf_cross_component_cr_coeff_sign[j]
}

[Fig. 171]



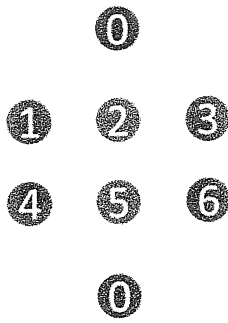
[Fig. 172A]



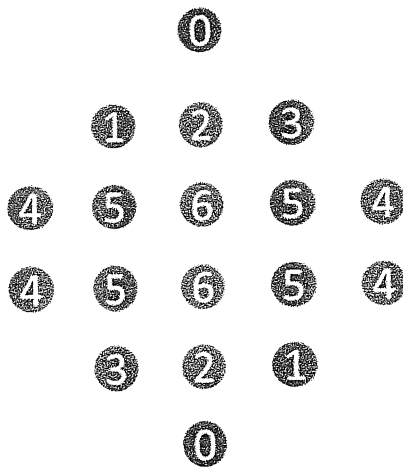
[Fig. 172B]



[Fig. 172C]



[Fig. 172D]



[Fig. 173A]

$$\begin{array}{ccccc}
 & & (2x, 2y-1) & & \\
 (2x-1, 2y) & (2x, 2y) & (2x+1, 2y) & & \\
 (2x-1, 2y+1) & (2x, 2y+1) & (2x+1, 2y+1) & & \\
 & & (2x, 2y+2) & &
 \end{array}$$

[Fig. 173B]

$$\begin{array}{ccccc}
 (2x-1, 2y) & (2x, 2y) & (2x+1, 2y) & & \\
 (2x-1, 2y+1) & (2x, 2y+1) & (2x+1, 2y+1) & &
 \end{array}$$

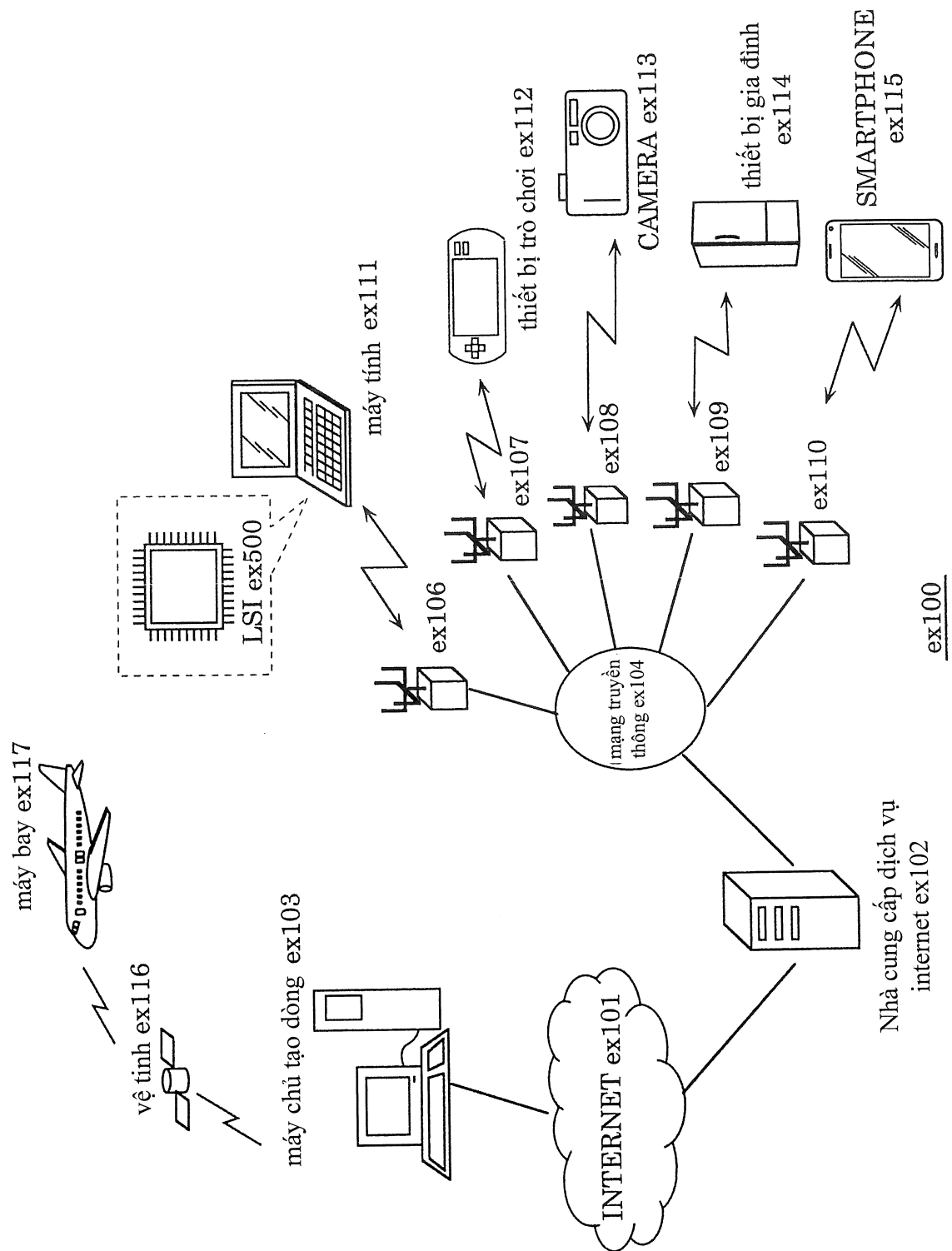
[Fig. 173C]

$$\begin{array}{ccccc}
 & & (2x, 2y-1) & & \\
 (2x-1, 2y) & (2x, 2y) & (2x+1, 2y) & & \\
 (2x-1, 2y+1) & (2x, 2y+1) & (2x+1, 2y+1) & & \\
 & & (2x, 2y+2) & &
 \end{array}$$

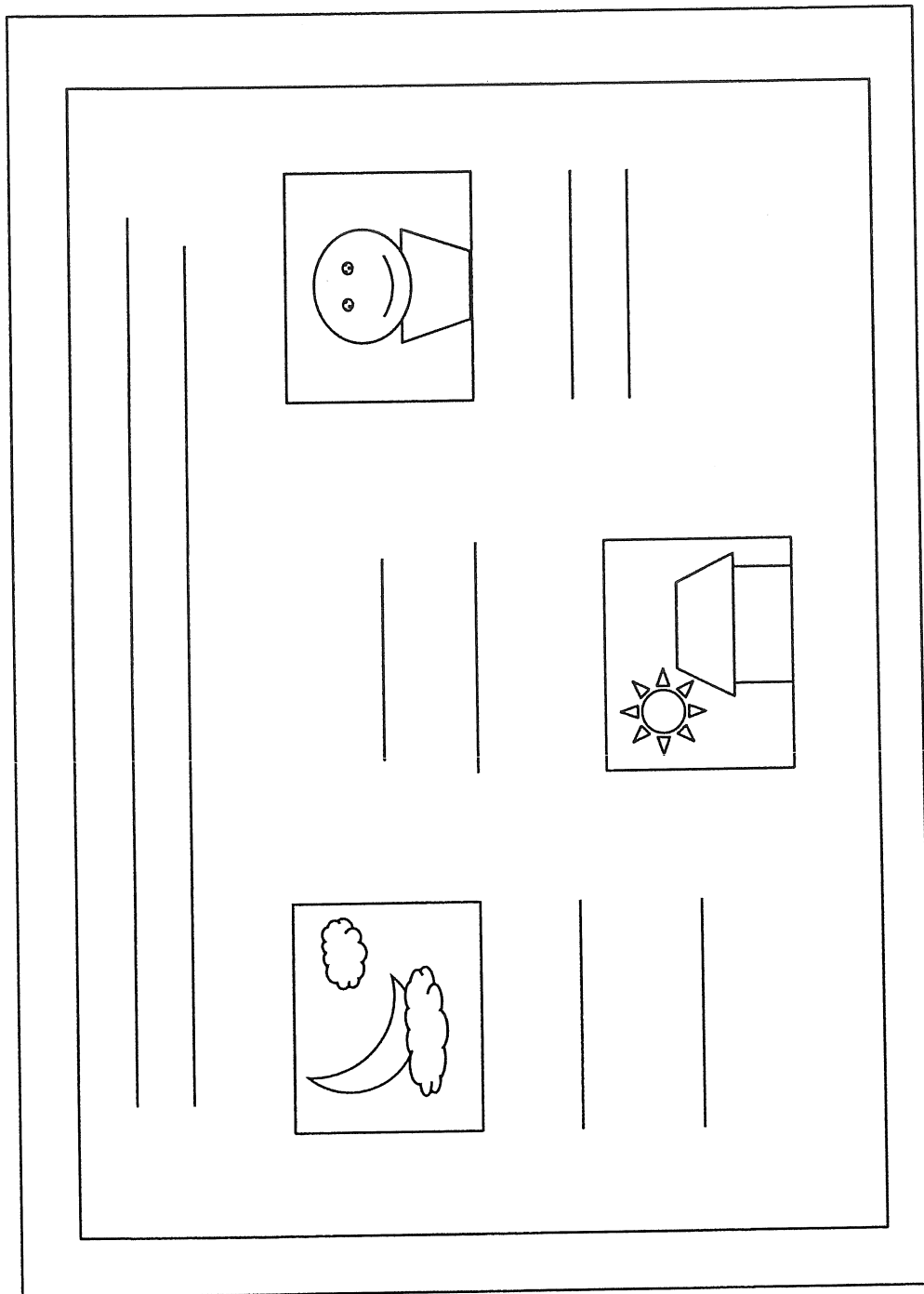
[Fig. 173D]

$$\begin{array}{ccccccc}
 & & & (2x, 2y-2) & & & \\
 & & (2x-1, 2y-1) & (2x, 2y-1) & (2x+1, 2y-1) & & \\
 (2x-2, 2y) & (2x-1, 2y) & (2x, 2y) & (2x+1, 2y) & (2x+2, 2y) & & \\
 (2x-2, 2y+1) & (2x-1, 2y+1) & (2x, 2y+1) & (2x+1, 2y+1) & (2x+2, 2y+1) & & \\
 & & (2x-1, 2y+2) & (2x, 2y+2) & (2x+1, 2y+2) & & \\
 & & & (2x, 2y+3) & & &
 \end{array}$$

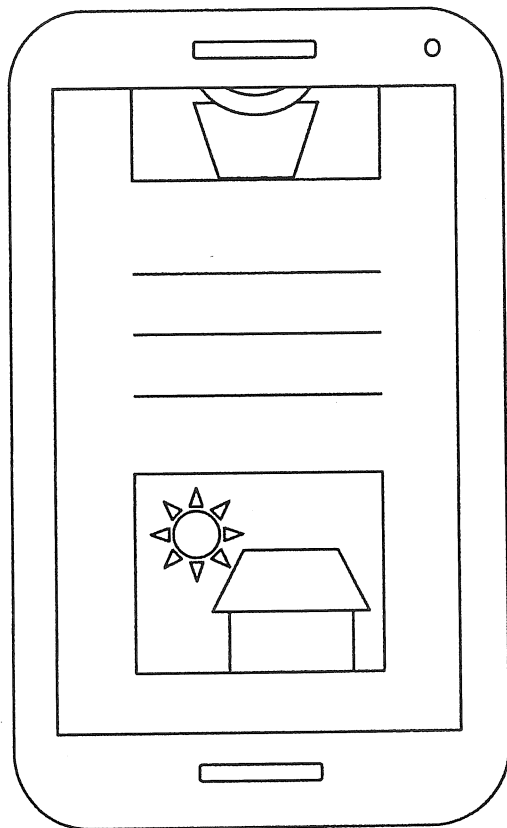
[Fig. 174]



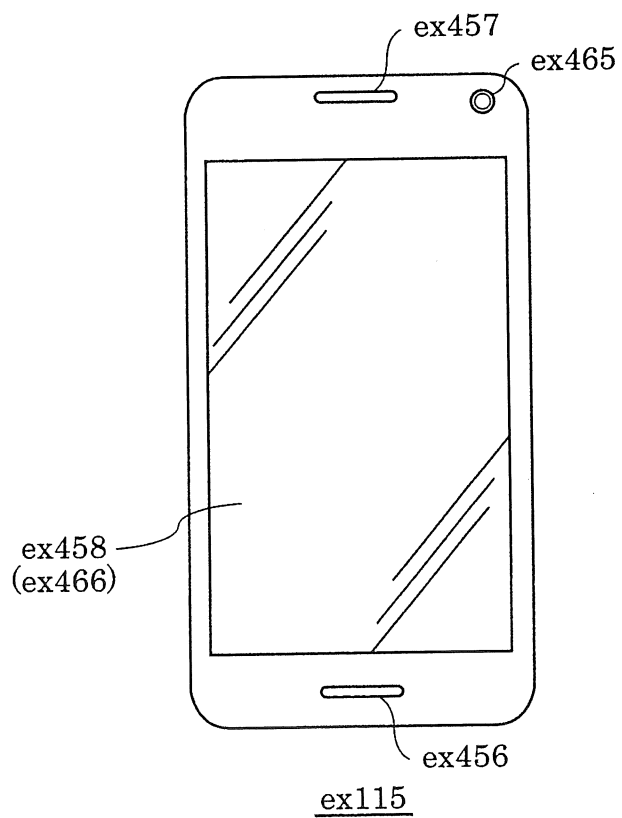
[Fig. 175]



[Fig. 176]



[Fig. 177]



[Fig. 178]

