



(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)



1-0048381

(51)^{2006.01} H04N 7/26

(13) B

(21) 1-2020-04317

(22) 20/02/2013

(62) 1-2014-02746

(86) PCT/JP2013/054126 20/02/2013

(87) WO 2013/129203 A1 06/09/2013

(30) 2012-044009 29/02/2012 JP

(45) 25/07/2025 448

(43) 26/10/2020 391A

(73) SONY CORPORATION (JP)

1-7-1, Konan, Minato-ku, Tokyo 108-0075, Japan

(72) TANAKA Junichi (JP); MORIGAMI Yoshitaka (JP).

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

(54) THIẾT BỊ XỬ LÝ ẢNH, PHƯƠNG PHÁP XỬ LÝ ẢNH VÀ VẬT GHI ĐỌC
ĐƯỢC BẰNG MÁY TÍNH

(21) 1-2020-04317

(57) Sáng chế đề cập đến thiết bị và phương pháp xử lý ảnh mà cho phép ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Thiết bị xử lý ảnh của sáng chế thiết lập hệ số nằm tại phần đầu của ma trận lượng tử hóa bằng cách cộng hệ số chênh lệch thay thế mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số nằm tại phần đầu của ma trận lượng tử hóa và hệ số nằm tại phần đầu của ma trận lượng tử hóa vào hệ số nằm tại phần đầu của ma trận lượng tử hóa; chuyển đổi lên ma trận lượng tử hóa được thiết lập; và giải lượng tử hóa dữ liệu được lượng tử hóa nhờ sử dụng ma trận lượng tử hóa được chuyển đổi lên trong đó hệ số nằm tại phần đầu của ma trận lượng tử hóa được chuyển đổi lên được thay thế bởi hệ số thay thế. Sáng chế có thể được áp dụng cho thiết bị xử lý ảnh.

scaling_list(ScalingList, sizeID, MatrixID) {	Phần mô tả
nextcoef = 8	
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
for(i=0; i < coefNum, i++) {	
scaling_list_delta_coef	se(v)
nextcoef = (nextcoef + scaling_list_delta_coef + 256) % 256	
ScalingList[sizeID][matrixID][i] = nextcoef	
}	
if(sizeID > 1)	
scaling_list_dc_coef_minus8[sizeID-2][matrixID]	se(v)
}	
A	
nextCoef = 8	
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID-2][matrixID]	se(v)
nextCoef =	
scaling_list_dc_coef_minus8[sizeID-2][matrixID] + 8	
}	
for(i = 0; i < coefNum; i++) {	se(v)
scaling_list_delta_coef	
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
ScalingList[sizeID][matrixID][i] = nextCoef	
}	
B	

FIG. 7

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến thiết bị và phương pháp xử lý ảnh.

Tình trạng kỹ thuật của sáng chế

Trong H.264/AVC (Advanced Video Coding, mã hóa video cải tiến), mà là một trong các đặc tả kỹ thuật tiêu chuẩn của các phương pháp mã hóa video, các lược sử của lược sử cao hoặc cao hơn cho phép việc lượng tử hóa của dữ liệu ảnh với các kích cỡ bước lượng tử hóa mà một thành phần của hệ số biến đổi trực giao khác với thành phần khác. Kích cỡ bước lượng tử hóa đối với mỗi thành phần của hệ số biến đổi trực giao có thể được thiết lập dựa trên giá trị bước tham chiếu và ma trận lượng tử hóa (cũng được gọi là danh sách chia tỷ lệ) được xác định bởi kích cỡ tương đương với đơn vị của biến đổi trực giao.

Giá trị được chỉ rõ của ma trận lượng tử hóa được bố trí cho mỗi chế độ dự đoán (chế độ dự đoán trong, chế độ dự đoán liên đới) và đối với mỗi kích cỡ đơn vị biến đổi (4x4, 8x8). Ngoài ra, những người dùng được phép chỉ rõ ma trận lượng tử hóa duy nhất khác với các giá trị được chỉ rõ trong tập tham số chuỗi hoặc tập tham số ảnh. Trong trường hợp mà không có các ma trận lượng tử hóa được sử dụng, các kích cỡ bước lượng tử hóa được sử dụng cho việc lượng tử hóa có giá trị bằng nhau đối với tất cả các thành phần.

Trong HEVC (High Efficiency Video Coding, mã hóa video hiệu quả cao), mà được chuẩn hóa là phương pháp mã hóa video thế hệ tiếp theo và là H.264/AVC kế tiếp, khái niệm của các đơn vị mã hóa (coding unit, CU) tương ứng với các khối macrô truyền thống được đưa ra (xem, ví dụ, tài liệu phi sáng chế 1). Phạm vi của các kích cỡ của các đơn vị mã hóa được chỉ rõ bởi tập của các giá trị mà các lũy thừa của 2, được gọi là đơn vị mã hóa lớn nhất (largest coding unit, LCU) và đơn vị mã hóa nhỏ nhất (smallest coding unit, SCU), trong tập tham số chuỗi. Ngoài ra, kích cỡ đơn vị mã hóa cụ thể trong phạm vi này được chỉ rõ bởi LCU và SCU được chỉ rõ nhờ sử dụng `split_flag`.

Trong HEVC, một đơn vị mã hóa có thể được chia thành một hoặc nhiều

đơn vị biến đổi trực giao, hoặc một hoặc nhiều đơn vị biến đổi (transform unit, TU). Kích cỡ đơn vị biến đổi khả dụng là bất kỳ trong số 4x4, 8x8, 16x16, và 32x32.

Trong khi, thành phần DC (cũng được gọi là thành phần dòng một chiều) của ma trận lượng tử hóa (danh sách chia tỷ lệ) được truyền làm dữ liệu khác với các thành phần AC (cũng được gọi là các thành phần dòng xoay chiều) của ma trận lượng tử hóa nhằm mục đích làm giảm lượng mã hóa trong khi truyền. Cụ thể, thành phần DC của danh sách chia tỷ lệ được truyền làm hệ số DC (cũng được gọi là hệ số dòng một chiều) khác với các hệ số AC (cũng được gọi là các hệ số dòng xoay chiều), mà là các thành phần AC của danh sách chia tỷ lệ.

Để làm giảm lượng mã hóa của hệ số DC trong khi truyền, có đề xuất rằng hằng số (ví dụ, 8) được trừ từ giá trị của hệ số DC và giá trị kết quả (scaling_list_dc_coef_minus8) được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu (xem, ví dụ, tài liệu phi sáng chế 1).

Danh mục tài liệu trích dẫn

Tài liệu phi sáng chế

Tài liệu phi sáng chế 1: Benjamin Bross, Fraunhofer HHI, Woo-Jin Han, đại học Gachon, Jens-Rainer Ohm, RWTH Aachen, Gary J. Sullivan, Microsoft, Thomas Wiegand, Fraunhofer HHI/TU Berlin, JCTVC-H1003, “High Efficiency Video Coding (HEVC) text specification draft 6 (bản thảo thứ 6 của đặc tả kỹ thuật mã hóa video hiệu quả cao)”, lần họp thứ 7 của nhóm hợp tác chung về mã hóa video (Joint Collaborative Team on Video Coding, JCT-VC) của ITU-T SG16 WP3 và ISO/IEC JTC1/SC29/WG11: Geneva, Thụy Sĩ, từ ngày 21 đến 30 tháng 11, năm 2011.

Bản chất kỹ thuật của sáng chế

Vấn đề kỹ thuật

Tuy nhiên, có vấn đề là phương pháp được mô tả trên đây không tạo ra đủ hiệu quả nén mặc dù nó hỗ trợ các xử lý.

Sáng chế được tạo ra để giải quyết nhược điểm nêu trên, và mục đích của sáng chế là cho phép ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Giải quyết vấn đề

Một khía cạnh của sáng chế đề xuất thiết bị xử lý ảnh bao gồm mạch được tạo cấu hình để:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng được sao chép, làm cú pháp mà ngữ nghĩa của nó được thiết đặt sao cho ma trận lượng tử hóa mặc định được tham chiếu tới điều kiện là giá trị bằng 0, giá trị chênh lệch giữa giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa tham chiếu và giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa dòng;

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch thay thế, mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số (0, 0) của ma trận lượng tử hóa 16x16 và hệ số (0, 0) của ma trận lượng tử hóa 8x8, ma trận lượng tử hóa 16x16 thu được nhờ sử dụng xử lý lân cận gần nhất dựa trên ma trận lượng tử hóa 8x8 dùng để chuyển đổi lên;

lượng tử hóa dữ liệu hệ số biến đổi thu được bằng cách biến đổi trực giao ảnh, và tạo dữ liệu được lượng tử hóa; và

mã hóa dữ liệu được lượng tử hóa đã tạo và hệ số chênh lệch thay thế để tạo dữ liệu được mã hóa, dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế và giá trị chênh lệch.

Khía cạnh khác của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch, mà là độ chênh lệch giữa mỗi hệ số của ma trận lượng tử hóa 8x8; và

tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch mà tạo nhóm hệ số chênh lệch thay thế và hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

mã hóa hệ số chênh lệch thay thế, và sau đó mã hóa hệ số chênh lệch; và tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch ban đầu, mà là độ chênh lệch giữa hệ số thay thế và giá trị ban đầu của hệ số được thiết đặt cho ma trận lượng tử hóa; và

tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch ban đầu.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

mã hóa hệ số chênh lệch ban đầu, và sau đó mã hóa nhóm hệ số chênh lệch; và

tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu và nhóm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

mã hóa hệ số chênh lệch ban đầu nhờ sử dụng mã hóa Golomb theo số mũ có dấu, và sau đó mã hóa nhóm hệ số chênh lệch nhờ sử dụng mã hóa Golomb theo số mũ có dấu; và

tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu được mã hóa và nhóm hệ số chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi; và

lượng tử hóa dữ liệu hệ số biến đổi đã tạo.

Khía cạnh khác nữa của sáng chế đề xuất thiết bị xử lý ảnh, trong đó mạch được tạo cấu hình để:

biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi nhờ sử dụng bộ chuyển đổi 16x16; và

lượng tử hóa dữ liệu hệ số biến đổi đã tạo nhờ sử dụng ma trận lượng tử hóa 16x16.

Khía cạnh khác của sáng chế đề xuất phương pháp xử lý ảnh bao gồm các

bước:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng được sao chép, làm cú pháp mà ngữ nghĩa của nó được thiết đặt sao cho ma trận lượng tử hóa mặc định được tham chiếu tới điều kiện là giá trị bằng 0, giá trị chênh lệch giữa giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa tham chiếu và giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa dòng;

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch thay thế, mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số (0, 0) của ma trận lượng tử hóa 16x16 và hệ số (0, 0) của ma trận lượng tử hóa 8x8, ma trận lượng tử hóa 16x16 thu được nhờ sử dụng xử lý lân cận gần nhất dựa trên ma trận lượng tử hóa 8x8 dùng để chuyển đổi lên;

lượng tử hóa dữ liệu hệ số biến đổi thu được bằng cách biến đổi trực giao ảnh, và tạo dữ liệu được lượng tử hóa; và

mã hóa, nhờ sử dụng mạch, dữ liệu được lượng tử hóa đã tạo và hệ số chênh lệch thay thế để tạo dữ liệu được mã hóa, dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế và giá trị chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, thiết đặt hệ số chênh lệch, mà là độ chênh lệch giữa mỗi hệ số của ma trận lượng tử hóa 8x8, và

bước mã hóa tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch mà tạo nhóm hệ số chênh lệch thay thế và hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước mã hóa mã hóa hệ số chênh lệch thay thế, và sau đó mã hóa hệ số chênh lệch, và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được

sao chép, thiết đặt hệ số chênh lệch ban đầu, mà là độ chênh lệch giữa hệ số thay thế và giá trị ban đầu của hệ số được thiết đặt cho ma trận lượng tử hóa, và

bước mã hóa tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch ban đầu.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước mã hóa mã hóa hệ số chênh lệch ban đầu, và sau đó mã hóa nhóm hệ số chênh lệch, và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu và nhóm hệ số chênh lệch.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước mã hóa mã hóa hệ số chênh lệch ban đầu nhờ sử dụng mã hóa Golomb theo số mũ có dấu, và sau đó mã hóa nhóm hệ số chênh lệch nhờ sử dụng mã hóa Golomb theo số mũ có dấu; và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu được mã hóa và nhóm hệ số chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó phương pháp này còn bao gồm bước biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi, và

bước lượng tử hóa lượng tử hóa dữ liệu hệ số biến đổi đã tạo.

Khía cạnh khác nữa của sáng chế đề xuất phương pháp xử lý ảnh, trong đó bước biến đổi trực giao biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi nhờ sử dụng bộ chuyển đổi 16×16 , và

bước lượng tử hóa lượng tử hóa dữ liệu hệ số biến đổi đã tạo nhờ sử dụng ma trận lượng tử hóa 16×16 .

Hiệu quả của sáng chế

Theo sáng chế, có thể xử lý ảnh. Cụ thể, có thể ngăn ngừa việc tăng lượng mã hóa của ma trận lượng tử hóa.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ minh họa ví dụ của danh sách chia tỷ lệ.

Fig.2 là sơ đồ minh họa ví dụ của việc chuyển đổi lên.

Fig.3 là sơ đồ minh họa ví dụ của cách thức danh sách chia tỷ lệ được sử dụng trong bộ giải mã.

Fig.4 là sơ đồ minh họa ví dụ của việc mã hóa của danh sách chia tỷ lệ.

Fig.5 là sơ đồ minh họa ví dụ của việc mã hóa của danh sách chia tỷ lệ nhờ sử dụng sáng chế.

Fig.6 là sơ đồ minh họa ví dụ của các mã Golomb theo số mũ.

Fig.7 bao gồm các sơ đồ minh họa ví dụ của cú pháp đối với danh sách chia tỷ lệ.

Fig.8 là sơ đồ minh họa ví dụ của cú pháp dùng cho ma trận mặc định.

Fig.9 bao gồm các sơ đồ minh họa các ví dụ của các ngữ nghĩa của ma trận mặc định.

Fig.10 là sơ đồ minh họa ví dụ của cú pháp đối với danh sách chia tỷ lệ.

Fig.11 là sơ đồ minh họa ví dụ của cú pháp đối với danh sách chia tỷ lệ nhờ sử dụng sáng chế.

Fig.12 bao gồm các sơ đồ minh họa ví dụ về cú pháp của danh sách chia tỷ lệ theo giải pháp kỹ thuật đã biết.

Fig.13 là sơ đồ minh họa ví dụ về cú pháp của danh sách chia tỷ lệ.

Fig.14 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị mã hóa ảnh.

Fig.15 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ lượng tử hóa/biến đổi trực giao.

Fig.16 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ xử lý ma trận.

Fig.17 là sơ đồ minh họa ví dụ của việc lấy mẫu xuống.

Fig.18 là sơ đồ minh họa ví dụ của việc loại bỏ phần xếp chồng.

Fig.19 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ DPCM.

Fig.20 là lưu đồ minh họa ví dụ của dòng xử lý của xử lý mã hóa ma trận lượng tử hóa.

Fig.21 là lưu đồ minh họa ví dụ của dòng xử lý của xử lý DPCM.

Fig.22 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị giải mã ảnh.

Fig.23 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ biến đổi trực

giao ngược/giải lượng tử hóa.

Fig.24 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ tạo ma trận.

Fig.25 là sơ đồ minh họa ví dụ của xử lý nội suy lân cận gần nhất.

Fig.26 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ DPCM ngược.

Fig.27 là lưu đồ minh họa ví dụ của dòng xử lý của xử lý tạo ma trận.

Fig.28 là lưu đồ minh họa ví dụ của dòng xử lý của xử lý giải mã tín hiệu dư.

Fig.29 là lưu đồ minh họa ví dụ của dòng xử lý của xử lý DPCM ngược.

Fig.30 là sơ đồ minh họa ví dụ khác của cú pháp của danh sách chia tỷ lệ.

Fig.31 là sơ đồ khối minh họa cấu trúc ví dụ khác của bộ DPCM.

Fig.32 là lưu đồ minh họa ví dụ khác của dòng xử lý của xử lý DPCM.

Fig.33 là sơ đồ khối minh họa cấu trúc ví dụ khác của bộ DPCM ngược.

Fig.34 là lưu đồ minh họa ví dụ khác của dòng xử lý của xử lý DPCM ngược.

Fig.35 là sơ đồ minh họa ví dụ khác của cú pháp của danh sách chia tỷ lệ.

Fig.36 là lưu đồ minh họa ví dụ khác của dòng xử lý của xử lý DPCM ngược.

Fig.37 là sơ đồ minh họa ví dụ khác của cú pháp của danh sách chia tỷ lệ.

Fig.38 là sơ đồ khối minh họa cấu trúc ví dụ khác của bộ DPCM.

Fig.39 là lưu đồ minh họa ví dụ khác của xử lý DPCM.

Fig.40 là sơ đồ khối minh họa cấu trúc ví dụ khác của bộ DPCM ngược.

Fig.41 là lưu đồ minh họa ví dụ khác của dòng xử lý của xử lý DPCM ngược.

Fig.42 là lưu đồ được tiếp tục từ Fig.41, minh họa ví dụ khác của dòng xử lý của xử lý DPCM ngược.

Fig.43 bao gồm các sơ đồ minh họa ví dụ khác của cú pháp của danh sách chia tỷ lệ.

Fig.44 bao gồm các sơ đồ minh họa ví dụ khác của cú pháp của danh sách chia tỷ lệ.

Fig.45 bao gồm các sơ đồ minh họa ví dụ khác của cú pháp của danh sách

chia tỷ lệ.

Fig.46 là sơ đồ minh họa ví dụ của phương pháp mã hóa ảnh đa cảnh nhìn.

Fig.47 là sơ đồ minh họa ví dụ về cấu trúc chính của thiết bị mã hóa ảnh đa cảnh nhìn mà sáng chế được áp dụng.

Fig.48 là sơ đồ minh họa ví dụ về cấu trúc chính của thiết bị giải mã ảnh đa cảnh nhìn mà sáng chế được áp dụng.

Fig.49 là sơ đồ minh họa ví dụ của phương pháp mã hóa ảnh được phân lớp.

Fig.50 là sơ đồ minh họa ví dụ về cấu trúc chính của thiết bị mã hóa ảnh được phân lớp mà sáng chế được áp dụng.

Fig.51 là sơ đồ minh họa ví dụ về cấu trúc chính của thiết bị giải mã ảnh được phân lớp mà sáng chế được áp dụng.

Fig.52 là sơ đồ khối minh họa ví dụ về cấu trúc chính của máy tính.

Fig.53 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị tivi.

Fig.54 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị đầu cuối di động.

Fig.55 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị ghi/tái tạo.

Fig.56 là sơ đồ khối minh họa ví dụ về cấu trúc chính của thiết bị tạo ảnh.

Fig.57 là sơ đồ khối minh họa ví dụ của việc sử dụng mã hóa có thể chia tỷ lệ.

Fig.58 là sơ đồ khối minh họa ví dụ khác của việc sử dụng mã hóa có thể chia tỷ lệ.

Fig.59 là sơ đồ khối minh họa ví dụ khác của việc sử dụng mã hóa có thể chia tỷ lệ.

Mô tả chi tiết sáng chế

Các phương án thực hiện sáng chế (sau đây gọi là các phương án) sẽ được mô tả sau đây. Về mặt này, phần mô tả sẽ được thực hiện theo thứ tự sau đây.

1. Phương án thứ nhất (ứng dụng ví dụ của sáng chế)
2. Phương án thứ hai (thiết bị mã hóa ảnh, thiết bị giải mã ảnh: phương

pháp thứ nhất)

3. Phương án thứ ba (thiết bị mã hóa ảnh, thiết bị giải mã ảnh: phương pháp thứ hai)

4. Phương án thứ tư (thiết bị mã hóa ảnh, thiết bị giải mã ảnh: phương pháp thứ ba)

5. Phương án thứ năm (thiết bị mã hóa ảnh, thiết bị giải mã ảnh: phương pháp thứ tư)

6. Phương án thứ sáu (thiết bị mã hóa ảnh, thiết bị giải mã ảnh: các phương pháp khác)

7. Phương án thứ bảy (thiết bị mã hóa ảnh đa cảnh nhìn, thiết bị giải mã ảnh đa cảnh nhìn)

8. Phương án thứ tám (thiết bị mã hóa ảnh được phân lớp, thiết bị giải mã ảnh được phân lớp)

9. Phương án thứ chín (máy tính)

10. Các ứng dụng ví dụ

11. Các ứng dụng ví dụ của mã hóa có thể chia tỷ lệ

1. Phương án thứ nhất

Trong phương án này, phần mô tả sẽ được đưa ra về ứng dụng ví dụ của sáng chế, mà sẽ được mô tả chi tiết trong các phương án thứ hai và tiếp theo của nó.

1-1. Ứng dụng ví dụ của sáng chế

Đầu tiên, ví dụ trong đó sáng chế có thể được áp dụng sẽ được mô tả. Sáng chế là kỹ thuật liên quan đến việc mã hóa và giải mã của danh sách chia tỷ lệ được sử dụng trong các xử lý lượng tử hóa và giải lượng tử hóa được thực hiện khi dữ liệu ảnh được mã hóa và giải mã.

Việc mã hóa và giải mã của dữ liệu ảnh có thể bao gồm việc lượng tử hóa và giải lượng tử hóa của dữ liệu hệ số. Việc lượng tử hóa và giải lượng tử hóa này được thực hiện trong các đơn vị của khối có kích cỡ định trước, và danh sách chia tỷ lệ (hoặc ma trận lượng tử hóa) có kích cỡ tương ứng với kích cỡ khối được sử dụng. Ví dụ, trong HEVC (High Efficiency Video Coding, mã hóa video hiệu quả cao), việc lượng tử hóa (hoặc giải lượng tử hóa) được thực hiện

với các kích cỡ như 4x4, 8x8, 16x16, và 32x32. Trong HEVC, các ma trận lượng tử hóa có các kích cỡ 4x4 và 8x8 có thể được bố trí.

Fig.1 minh họa ví dụ của danh sách chia tỷ lệ 8x8. Như được minh họa trên Fig.1, danh sách chia tỷ lệ bao gồm hệ số DC và các hệ số AC. Hệ số DC bao gồm một giá trị là hệ số (0, 0) của ma trận lượng tử hóa, và tương ứng với hệ số DC của biến đổi cosin rời rạc (DCT). Các hệ số AC là các hệ số của ma trận lượng tử hóa ngoài hệ số (0, 0), và tương ứng với các hệ số của DCT ngoài hệ số DC. Lưu ý rằng, như được minh họa trên Fig.1, các hệ số AC được biểu diễn bởi ma trận. Tức là, các hệ số AC cũng bao gồm hệ số (0, 0) (sau đây cũng được gọi là hệ số AC (0, 0)), và hệ số (0, 0), mà có vị trí tại phần đầu của ma trận lượng tử hóa, được thay thế bằng hệ số DC khi được sử dụng cho việc lượng tử hóa/giải lượng tử hóa. Do đó, hệ số DC cũng được gọi là hệ số thay thế. Trong ví dụ được minh họa trên Fig.1, các hệ số AC tạo thành ma trận 8x8.

Trong HEVC, ngoài ra, phiên bản được chuyển đổi lên (chuyển đổi hướng lên) của ma trận lượng tử hóa 8x8 được sử dụng đối với lượng tử hóa (hoặc giải lượng tử hóa) 16x16 hoặc 32x32.

Fig.2 minh họa ví dụ của việc chuyển đổi lên của danh sách chia tỷ lệ 8x8 thành danh sách chia tỷ lệ 16x16. Như được minh họa trên Fig.2, danh sách chia tỷ lệ được chuyển đổi lên nhờ sử dụng, ví dụ, xử lý nội suy lân cận gần nhất. Các chi tiết của xử lý nội suy lân cận gần nhất sẽ được mô tả dưới đây có viện dẫn tới, ví dụ, Fig.25. Như được minh họa trên Fig.2, việc chuyển đổi lên được thực hiện trên các hệ số AC của danh sách chia tỷ lệ. Sau đó, hệ số (0, 0) trong số các hệ số AC được chuyển đổi lên được thay thế bằng hệ số DC.

Hai loại của danh sách chia tỷ lệ 8x8 được bố trí, cụ thể là, được sử dụng cho việc chuyển đổi lên thành 16x16 (“8x8 đối với 16x16”) và được sử dụng cho việc chuyển đổi lên tới 32x32 (“8x8 đối với 32x32”).

Danh sách chia tỷ lệ được sử dụng cho việc lượng tử hóa trong khi mã hóa (sử dụng bộ mã hóa) cũng được sử dụng cho việc giải lượng tử hóa trong khi giải mã (sử dụng bộ giải mã). Tức là, danh sách chia tỷ lệ được truyền từ phía mã hóa (bộ mã hóa) tới phía giải mã (bộ giải mã). Fig.3 minh họa ví dụ về việc truyền của các danh sách chia tỷ lệ.

Như trong ví dụ được minh họa trên Fig.3, hai loại của danh sách chia tỷ lệ của 8×8 , cụ thể là, được sử dụng cho việc chuyển đổi lên thành kích cỡ 16×16 và mà được sử dụng cho việc chuyển đổi lên thành kích cỡ 32×32 , như được mô tả trên đây, được truyền. Mặc dù không được minh họa trên các hình vẽ, danh sách chia tỷ lệ 4×4 cũng được truyền.

Các hệ số AC của danh sách chia tỷ lệ 8×8 được sử dụng cho việc chuyển đổi lên thành kích cỡ 16×16 , mà được truyền theo cách được mô tả trên đây, được chuyển đổi lên thành kích cỡ 16×16 tại phía giải mã (bộ giải mã) nhờ sử dụng xử lý nội suy lân cận gần nhất được mô tả trên đây, và được sử dụng cho việc giải lượng tử hóa của khối có kích cỡ 16×16 sau hệ số $(0, 0)$ được thay thế bằng hệ số DC.

Tương tự, các hệ số AC của danh sách chia tỷ lệ 8×8 được sử dụng cho việc chuyển đổi lên thành kích cỡ 32×32 , mà được truyền theo cách được mô tả trên đây, cũng được chuyển đổi lên thành kích cỡ 32×32 tại phía giải mã (bộ giải mã) nhờ sử dụng xử lý nội suy lân cận gần nhất được mô tả trên đây, và được sử dụng cho việc giải lượng tử hóa của khối có kích cỡ 32×32 sau khi hệ số $(0, 0)$ được thay thế bằng hệ số DC.

1-2. Mã hóa danh sách chia tỷ lệ

Việc truyền của các danh sách chia tỷ lệ theo cách được mô tả trên đây sẽ làm tăng lượng mã hóa tương ứng. Do đó, để ngăn ngừa việc giảm của hiệu quả mã hóa, các danh sách chia tỷ lệ được mã hóa sử dụng phương pháp đã biết để làm giảm lượng mã hóa của các danh sách chia tỷ lệ. Fig.4 minh họa ví dụ của việc mã hóa của danh sách chia tỷ lệ. Cụ thể, danh sách chia tỷ lệ 8×8 được truyền như sau.

Trong trường hợp của chuyển đổi lên của ma trận 8×8 thành ma trận 16×16 :

(1) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số AC $(0, 0)$) của ma trận 8×8 và giá trị khởi tạo định trước “8” được thu.

(2) Các chênh lệch giữa các hệ số (tức là, các hệ số AC) (các hệ số lân cận trong chuỗi của các hệ số được bố trí một chiều trong thứ tự quét) của ma trận 8×8 được thu.

(3) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số DC) của ma trận 16×16 và giá trị khởi tạo định trước “8” được thu.

(4) Các độ chênh lệch thu được trong mục (1) và (2) và độ chênh lệch thu được trong mục (3) được truyền riêng biệt.

Trong trường hợp của chuyển đổi lên của ma trận 8×8 thành ma trận 32×32 :

(1) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số AC $(0, 0)$) của ma trận 8×8 và giá trị khởi tạo định trước “8” được thu.

(2) Các độ chênh lệch giữa các hệ số (tức là, các hệ số AC) (các hệ số lân cận trong chuỗi của các hệ số được bố trí một chiều trong thứ tự quét) của ma trận 8×8 được thu.

(3) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số DC) của ma trận 32×32 và giá trị khởi tạo định trước “8” được thu.

(4) Các độ chênh lệch thu được trong mục (1) và (2) và độ chênh lệch thu được trong mục (3) được truyền riêng biệt.

Trong phương pháp được mô tả trên đây, tuy nhiên, các độ chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu và được truyền trong mục (4). Như được mô tả trên đây, độ chênh lệch thu được trong mục (1) là độ chênh lệch giữa hệ số AC $(0, 0)$ và giá trị khởi tạo “8”. Do đó, có vấn đề rằng lượng mã hóa có thể được tăng lên nếu giá trị của hệ số AC $(0, 0)$ không phải là giá trị gần với giá trị khởi tạo “8”.

Ví dụ, trên Fig.4, giá trị của hệ số AC $(0, 0)$ là “12”, và giá trị “4” được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu và được truyền như là độ chênh lệch thu được trong mục (1). Tức là, 7 bit được yêu cầu cho việc truyền của độ chênh lệch thu được trong mục (1) và hiệu quả mã hóa có thể được làm giảm tương ứng. Nếu giá trị của độ chênh lệch thu được trong mục (1) tăng lên, hiệu quả mã hóa có thể bị giảm hơn nữa. Điều tương tự là đúng đối với trường hợp của danh sách chia tỷ lệ 8×8 được sử dụng cho việc chuyển đổi lên thành kích cỡ 16×16 và danh sách chia tỷ lệ 8×8 được sử dụng cho việc chuyển đổi lên thành kích cỡ 32×32 .

Trong khi đó, năng lượng của các hệ số DCT thường được tập trung trong

hệ số DC và các hệ số bậc thấp lân cận. Do đó, nói chung, ma trận lượng tử hóa cũng có các giá trị nhỏ đối với hệ số DC và các hệ số lân cận. Ngoài ra, nếu các giá trị mà khác nhau đáng kể được sử dụng đối với các tần số riêng biệt, sai số lượng tử hóa có thể là đáng chú ý. Để ngăn ngừa suy giảm trực quan trong chất lượng ảnh, các giá trị liên tục được sử dụng đối với hệ số DC và các hệ số lân cận.

Hệ số (0, 1), hệ số (1, 0), và hệ số (1, 1) thu được sau khi chuyển đổi lên tương ứng với hệ số AC (0, 0) trước khi chuyển đổi lên. Ngoài ra, hệ số (0, 0) thu được sau khi chuyển đổi lên tương ứng với hệ số DC.

Do đó, trong các danh sách chia tỷ lệ, giá trị của hệ số AC (0, 0) và giá trị của hệ số DC nói chung là gần nhau. Ví dụ, MPEG2, AVC, và các ma trận mặc định HEVC thu các giá trị có quan hệ này. Ngoài ra trong ví dụ được minh họa trên Fig.4, giá trị của hệ số DC là giống như giá trị của hệ số AC (0, 0), tức là, “12”. Do đó, giá trị của độ chênh lệch thu được trong mục (3), tức là, độ chênh lệch giữa hệ số DC và giá trị khởi tạo “8”, cũng là “4”.

Tức là, việc thu độ chênh lệch giữa mỗi hệ số DC và hệ số AC (0, 0), mà các giá trị của chúng là gần nhau, và giá trị khởi tạo có thể làm tăng giá trị chênh lệch giữa chúng, và cũng có thể gây ra sự dư thừa. Có thể nói rằng sẽ có nguy cơ của việc giảm hơn nữa hiệu quả mã hóa.

Để giải quyết vấn đề này, danh sách chia tỷ lệ được truyền nhờ sử dụng phương pháp sau đây thay vì sử dụng phương pháp được minh họa trên Fig.4. Fig.5 minh họa ví dụ của phương pháp này.

Trong trường hợp của chuyển đổi lên của ma trận 8x8 thành ma trận 16x16:

(1) Độ chênh lệch giữa hệ số (0, 0) (tức là, hệ số AC (0, 0)) của ma trận 8x8 và hệ số (0, 0) (tức là, hệ số DC) của ma trận 16x16 được thu.

(2) Các độ chênh lệch giữa các hệ số (tức là, các hệ số AC) (các hệ số lân cận trong chuỗi của các hệ số được bố trí một chiều trong thứ tự quét) của ma trận 8x8 được thu.

(3) Độ chênh lệch giữa hệ số (0, 0) (tức là, hệ số DC) của ma trận 16x16 và giá trị khởi tạo định trước “8” được thu.

(4) Các độ chênh lệch thu được trong mục (1) tới (3) được truyền tổng hợp.

Trong trường hợp của chuyển đổi lên của ma trận 8×8 thành ma trận 32×32 :

(1) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số AC $(0, 0)$) của ma trận 8×8 và hệ số $(0, 0)$ (tức là, hệ số DC) của ma trận 32×32 được thu.

(2) Các độ chênh lệch giữa các hệ số (tức là, các hệ số AC) (các hệ số lân cận trong chuỗi của các hệ số được bố trí một chiều trong thứ tự quét) của ma trận 8×8 được thu.

(3) Độ chênh lệch giữa hệ số $(0, 0)$ (tức là, hệ số DC) của ma trận 32×32 và giá trị khởi tạo định trước “8” được thu.

(4) Các độ chênh lệch thu được trong mục (1) đến (3) được truyền tổng hợp.

Tương tự phương pháp được minh họa trên Fig.4, trong mục (4), các độ chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ và được truyền như là các mã Golomb theo số mũ.

Tại đích mà các độ chênh lệch được truyền tới như là các mã Golomb theo số mũ, khi các mã Golomb theo số mũ được thu, các mã Golomb theo số mũ được thu được giải mã để thu được các độ chênh lệch riêng biệt, và các xử lý ngược với các xử lý trong mục (1) đến (3) được mô tả trên đây được thực hiện trên các độ chênh lệch thu được để xác định các hệ số riêng biệt (hệ số DC và các hệ số AC).

1-3. Các đặc điểm ví dụ của sáng chế

Các đặc điểm ví dụ của sáng chế liên quan đến phương pháp truyền được mô tả trên đây sẽ được mô tả.

1-3-1. DPCM giữa hệ số AC $(0, 0)$ và hệ số DC

Các danh sách chia tỷ lệ được mã hóa nhờ sử dụng điều chế mã xung vi sai (differential pulse-code modulation, DPCM) và được truyền. Trong ví dụ được minh họa trên Fig.4, các hệ số AC và hệ số DC được mã hóa DPCM riêng biệt, trong khi đó, theo một trong các đặc điểm của sáng chế, như trong ví dụ được minh họa trên Fig.5, độ chênh lệch (cũng được gọi là hệ số chênh lệch thay thế) giữa hệ số AC $(0, 0)$ và hệ số DC được xác định và được truyền.

Như được mô tả trên đây, hệ số AC (0, 0) và hệ số DC nói chung thu các giá trị mà là gần nhau. Do đó, độ chênh lệch giữa hệ số AC (0, 0) và hệ số DC có thể nhỏ hơn so với độ chênh lệch giữa hệ số AC (0, 0) và giá trị khởi tạo “8”. Tức là, việc truyền của hệ số chênh lệch thay thế mà là độ chênh lệch giữa hệ số AC (0, 0) và hệ số DC nhờ sử dụng sáng chế có thể làm giảm hơn nữa lượng mã hóa.

Ví dụ, trong ví dụ được minh họa trên Fig.5, giá trị của độ chênh lệch thu được trong mục (1) là “0”.

Fig.6 là bảng minh họa ví dụ của mã hóa Golomb theo số mũ có dấu. Như được thể hiện trong bảng được minh họa trên Fig.6, mã Golomb theo số mũ đối với giá trị “4” có độ dài mã của 7 bit trong khi mã Golomb theo số mũ đối với giá trị “0” có độ dài mã của 1 bit. Tức là, phương pháp được minh họa trên Fig.5 có thể làm giảm lượng mã hóa bởi 6 bit so với phương pháp được minh họa trên Fig.4.

Nói chung, lượng mã hóa của xấp xỉ 100 bit đến 200 bit được yêu cầu cho việc truyền của ma trận lượng tử hóa có kích cỡ 8x8. Do đó, 6 bit chiếm giữa xấp xỉ 6% của tổng lượng mã. Việc làm giảm lượng mã hóa bởi 6% trong cú pháp mức cao có được xem là hiệu quả rất lớn.

1-3-2. Sự truyền tổng hợp của hệ số DC và các hệ số AC

Fig.7 minh họa ví dụ về cú pháp của danh sách chia tỷ lệ. Cú pháp đối với ví dụ được minh họa trên Fig.4 được minh họa trên ví dụ được minh họa trên phần A của Fig.7. Cụ thể, sau khi độ chênh lệch giữa hệ số AC (0, 0) và giá trị khởi tạo “8” và các độ chênh lệch giữa các hệ số AC (scaling_list_delta_coef) được truyền, độ chênh lệch giữa hệ số DC và giá trị khởi tạo “8” (scaling_list_dc_coef_minus8) được truyền riêng biệt.

Ngược lại, một trong các đặc điểm của sáng chế đó là độ chênh lệch giữa hệ số DC và hệ số AC (0, 0) và các độ chênh lệch giữa các hệ số AC được bố trí theo thứ tự này và được truyền tổng hợp. Cụ thể, như được minh họa trên Fig.5, sau khi hệ số DC và các hệ số AC được bố trí theo thứ tự quét định trước được bố trí theo một chiều và độ chênh lệch giữa hệ số DC và giá trị khởi tạo “8” được xác định, các độ chênh lệch giữa các hệ số lân cận trong chuỗi của các hệ

số được xác định. Ngoài ra, các độ chênh lệch cuối cùng (các độ chênh lệch giữa các hệ số) được bố trí theo một chiều theo thứ tự thu được và được truyền tổng hợp.

Cú pháp trong trường hợp này được minh họa trong ví dụ trong phần B của Fig.7. Cụ thể, đầu tiên, độ chênh lệch giữa hệ số DC và giá trị khởi tạo “8” (`scaling_list_dc_coef_minus8`) được truyền, và sau đó độ chênh lệch giữa hệ số DC và hệ số AC (0, 0) và các độ chênh lệch giữa các hệ số AC (`scaling_list_delta_coef`) được truyền. Tức là, hệ số DC và các hệ số AC được mã hóa và được truyền tổng hợp.

Theo cách này, việc truyền tổng hợp của các độ chênh lệch được bố trí theo thứ tự thu được cho phép phía giải mã (bộ giải mã) mà các độ chênh lệch được truyền tới đó giải mã các độ chênh lệch theo thứ tự được truyền và thu được các hệ số riêng biệt. Tức là, danh sách chia tỷ lệ được mã hóa DPCM có thể được giải mã dễ dàng. Cụ thể hơn, tải xử lý có thể được làm giảm. Ngoài ra, việc bố trí lại của các độ chênh lệch không còn cần thiết, dẫn đến việc làm giảm dung lượng bộ đệm. Ngoài ra, các độ chênh lệch tương ứng có thể được giải mã theo thứ tự được cấp, thu được việc ngăn ngừa của tăng thời gian xử lý.

1-3-3. Truyền ma trận mặc định

Fig.8 là sơ đồ minh họa ví dụ của cú pháp đối với việc truyền của ma trận mặc định. Trong kỹ thuật đã biết, như được minh họa trên Fig.8, hệ số khởi tạo (tức là, hệ số DC) được truyền như là “0” để truyền thông tin chỉ báo việc sử dụng của ma trận mặc định. Tức là, giá trị của độ chênh lệch giữa hệ số DC và giá trị khởi tạo “8” (`scaling_list_dc_coef_minus8`) là “-8”. Tuy nhiên, như được minh họa trên Fig.6, mã Golomb theo số mũ đối với giá trị “-8” có độ dài mã của 9 bit. Tức là, có vấn đề rằng hiệu quả mã hóa có thể bị làm giảm đáng kể. Nói chung, có mong muốn rằng số lượng bit của cú pháp mức cao là nhỏ nhất có thể. Ngoài ra, như được minh họa trên Fig.8, do độ phức tạp tăng lên của cú pháp, tải xử lý có thể bị tăng lên.

Để giải quyết vấn đề này, hệ số khởi tạo không được thiết lập là “0” nhưng ngữ nghĩa của `scaling_list_pred_matrix_id_delta` được cải biến. Cụ thể hơn, ngữ nghĩa của `scaling_list_pred_matrix_id_delta` được cải biến từ ngữ nghĩa được

minh họa trên phần A của Fig.9 thành ngữ nghĩa được minh họa trên phần B của Fig.9. Tức là, theo giải pháp kỹ thuật đã biết, như được minh họa trên phần A của Fig.9, giá trị bằng “0” chỉ báo rằng ma trận ngay trước đó (MatrixID - 1) được viện dẫn tới. Thay vì phần mô tả này, như được minh họa trên phần B của Fig.9, giá trị của `scaling_list_pred_matrix_id_delta` bằng “0” nghĩa là ma trận mặc định được viện dẫn tới.

Do đó, độ dài mã của mã Golomb theo số mũ cho việc truyền của thông tin chỉ báo việc sử dụng của ma trận mặc định có thể bằng 1 bit, và việc giảm hiệu quả mã hóa có thể được ngăn ngừa. Ngoài ra, theo giải pháp kỹ thuật đã biết, cú pháp như được minh họa trên các phần A và B của Fig.10 là cần thiết đối với danh sách chia tỷ lệ. Cú pháp này có thể được đơn giản hóa như trong ví dụ được minh họa trên Fig.11. Tức là, tải xử lý được chứa trong việc mã hóa và giải mã của danh sách chia tỷ lệ có thể được làm giảm.

1-4. Các đặc điểm của cú pháp với sử dụng sáng chế

Cú pháp sẽ được mô tả cụ thể hơn.

Trong ví dụ của kỹ thuật đã biết được minh họa trên các phần A và B của Fig.10, việc xác định của mặc định cần được thực hiện hai lần, cụ thể là, `scaling_list_dc_coef_minus8` và `scaling_list_delta_coef`. Ngoài ra, đối với `scaling_list_delta_coef`, việc xác định được thực hiện trong phần giữa của vòng lặp “for”, và vòng lặp này tồn tại khi `useDefaultScalingMatrixFlag = 1`. Ngoài ra, cờ trung gian gọi là “stopNow” là cần thiết, và, do điều kiện này, nhánh chẳng hạn như thay thế `nextCoef` thành giá trị của `scalingList` cũng tồn tại. Theo cách này, cú pháp của kỹ thuật đã biết kéo theo xử lý phức tạp.

Theo sáng chế, do đó, như trong ví dụ được minh họa trên Fig.11, hệ số DC được tính toán từ `scaling_list_dc_coef_minus8` được thay thế thành `nextCoef` để thiết lập giá trị khởi tạo của `scaling_list_delta_coef` thành hệ số DC.

Ngoài ra, trong các ngữ nghĩa, giá trị của `scaling_list_pred_matrix_id_delta`, mà được biểu diễn bởi “+1” theo giải pháp kỹ thuật đã biết, vẫn không thay đổi, và giá trị “0” được sử dụng như là giá trị đặc biệt.

Có thể nói, theo giải pháp kỹ thuật đã biết, khi `ScalingList[0][2]` cần được

giải mã ($\text{matrixId} = 2$), nếu $\text{scaling_list_pred_matrix_id_delta} = 0$, sau đó $\text{matrixId} = 2$ thu được từ $\text{refMatrixId} = \text{matrixId} - (1 + \text{scaling_list_pred_matrix_id_delta})$. Do đó, $\text{refMatrixId} = 1$ thu được, và giá trị của $\text{ScalingList}[0][1]$ được sao chép.

Ngược lại, theo sáng chế, $\text{refMatrixId} = \text{matrixId} - \text{scaling_list_pred_matrix_id_delta}$ được thiết lập. Khi $\text{ScalingList}[0][2]$ cần được giải mã ($\text{matrixId} = 2$), $\text{scaling_list_pred_matrix_id_delta} = 1$ có thể được thiết lập nếu $\text{ScalingList}[0][1]$ cần được sao chép (hoặc nếu $\text{refMatrixId} = 1$ cần thu được).

Do đó, như được minh họa trên Fig.11, số lượng các hàng của cú pháp đối với danh sách chia tỷ lệ có thể được làm giảm đáng kể. Ngoài ra, hai biến được chứa làm dữ liệu trung gian, cụ thể là, $\text{UseDefaultScalingMatrix}$ và stopNow , có thể được bỏ qua. Ngoài ra, nhánh được thực hiện trong vòng lặp “for” như được minh họa trên Fig.10 có thể không còn được yêu cầu. Do đó, tải xử lý được chứa trong việc mã hóa và giải mã của danh sách chia tỷ lệ có thể được làm giảm.

1-5. Các bộ xử lý thực hiện sáng chế

Trong trường hợp mà sáng chế được áp dụng cho việc truyền danh sách chia tỷ lệ, danh sách chia tỷ lệ được mã hóa và giải mã theo cách được mô tả trên đây. Cụ thể, thiết bị mã hóa ảnh 10 được mô tả dưới đây có viện dẫn tới Fig.14 mã hóa danh sách chia tỷ lệ và truyền danh sách chia tỷ lệ được mã hóa, và thiết bị giải mã ảnh 300 được mô tả dưới đây có viện dẫn tới Fig.22 thu và giải mã danh sách chia tỷ lệ được mã hóa.

Danh sách chia tỷ lệ được mã hóa bởi bộ xử lý ma trận 150 (Fig.15) trong bộ lượng tử hóa/biến đổi trực giao 14 (Fig.14) của thiết bị mã hóa ảnh 10. Cụ thể hơn, danh sách chia tỷ lệ được mã hóa bởi bộ DPCM 192 và bộ exp-G 193 (đều được minh họa trên Fig.16) trong bộ mã hóa entropy 164 (Fig.16) trong bộ xử lý ma trận 150. Tức là, bộ DPCM 192 xác định độ chênh lệch giữa các hệ số (hệ số DC và các hệ số AC) của danh sách chia tỷ lệ, và bộ exp-G 193 mã hóa các độ chênh lệch riêng biệt nhờ sử dụng mã hóa Golomb theo số mũ.

Để mã hóa danh sách chia tỷ lệ nhờ sử dụng sáng chế như được mô tả trên đây, bộ DPCM 192 có thể có cấu trúc ví dụ như được minh họa trên, ví dụ,

Fig.19, và có thể thực hiện xử lý DPCM như trong ví dụ được minh họa trên Fig.21. Ngoài ra, các ngữ nghĩa như trong ví dụ được minh họa trên phần C của Fig.44 hoặc phần C của Fig.45 có thể được sử dụng.

Nói cách khác, chỉ bộ DPCM 192 và bộ exp-G 193 có thể được yêu cầu để đạt được việc mã hóa của danh sách chia tỷ lệ nhờ sử dụng sáng chế, và các thành phần khác có cấu trúc bất kỳ có thể được sử dụng nếu cần. Cấu trúc cần thiết, như bộ xử lý để chuyển đổi lên danh sách chia tỷ lệ và bộ xử lý để thực hiện việc lượng tử hóa nhờ sử dụng danh sách chia tỷ lệ, có thể được bố trí theo các phương án.

Ngoài ra, danh sách chia tỷ lệ được giải mã bởi bộ tạo ma trận 410 (Fig.23) trong bộ biến đổi trực giao ngược/giải lượng tử hóa 313 (Fig.22) của thiết bị giải mã ảnh 300. Cụ thể hơn, danh sách chia tỷ lệ được giải mã bởi bộ exp-G 551 và bộ DPCM ngược 552 (Fig.24) trong bộ giải mã entropi 533 (Fig.24) trong bộ tạo ma trận 410. Tức là, bộ exp-G 551 giải mã các mã Golomb để thu được các độ chênh lệch, và bộ DPCM ngược 552 xác định các hệ số riêng biệt (hệ số DC và các hệ số AC) của danh sách chia tỷ lệ từ các độ chênh lệch tương ứng.

Để giải mã danh sách chia tỷ lệ được mã hóa nhờ sử dụng sáng chế như được mô tả trên đây, bộ DPCM ngược 552 có thể có cấu trúc ví dụ như được minh họa trên, ví dụ, Fig.26, và có thể thực hiện xử lý DPCM ngược như trong ví dụ được minh họa trên Fig.29. Ngoài ra, các ngữ nghĩa như trong ví dụ được minh họa trên phần C của Fig.44 hoặc phần C của Fig.45 có thể được sử dụng.

Nói cách khác, chỉ bộ exp-G 551 và bộ DPCM ngược 552 có thể được yêu cầu để đạt được việc giải mã của danh sách chia tỷ lệ nhờ sử dụng sáng chế, và các thành phần khác có cấu trúc bất kỳ có thể được sử dụng nếu cần. Cấu trúc cần thiết, như bộ xử lý để chuyển đổi lên danh sách chia tỷ lệ và bộ xử lý để thực hiện giải lượng tử hóa nhờ sử dụng danh sách chia tỷ lệ, có thể được bố trí theo các phương án.

Các phương án riêng biệt mà sáng chế được áp dụng cho sẽ được mô tả sau đây để mô tả chi tiết hơn sáng chế.

2. Phương án thứ hai

2-1. Cú pháp: phương pháp thứ nhất

(1) Cú pháp của kỹ thuật đã biết

Đầu tiên, Fig.12 minh họa ví dụ về cú pháp của ma trận lượng tử hóa (hoặc danh sách chia tỷ lệ) theo giải pháp kỹ thuật đã biết. Trong sử dụng thực tế, ma trận chệch lệch giữa danh sách chia tỷ lệ và ma trận dự đoán của nó, ngoài danh sách chia tỷ lệ, thường được truyền. Do đó, trong phần mô tả sau đây của cú pháp và v.v, giả thiết rằng phần mô tả của danh sách chia tỷ lệ cũng có thể áp dụng cho ma trận chệch lệch.

Phần A của Fig.12 minh họa cú pháp đối với dữ liệu danh sách chia tỷ lệ (cú pháp dữ liệu danh sách chia tỷ lệ), và phần B của Fig.12 minh họa cú pháp của danh sách chia tỷ lệ (cú pháp danh sách chia tỷ lệ).

(1-1) Cú pháp dữ liệu danh sách chia tỷ lệ

Như được minh họa trên phần A của Fig.12, cú pháp đối với dữ liệu danh sách chia tỷ lệ chỉ rõ rằng cờ (`scaling_list_present_flag`) chỉ báo danh sách chia tỷ lệ có được bố trí hay không, cờ (`scaling_list_pred_mode_flag`) chỉ báo chế độ hiện tại có phải là chế độ sao chép hay không, thông tin (`scaling_list_pred_matrix_id_delta`) chỉ báo danh sách chia tỷ lệ nào được viện dẫn tới trong chế độ sao chép, và v.v được đọc.

(1-2) Cú pháp danh sách chia tỷ lệ

Như được minh họa trên phần B của Fig.12, cú pháp của danh sách chia tỷ lệ chỉ rõ rằng hệ số DC mà từ đó hàng số (ví dụ, 8) được trừ (`scaling_list_dc_coef_minus8`), giá trị chệch lệch (`scaling_list_delta_coef`) giữa các hệ số AC, và v.v được đọc và hệ số DC và các hệ số AC được khôi phục.

Tuy nhiên, có vấn đề rằng các đoạn của cú pháp được mô tả trên đây sẽ không cung cấp hiệu quả nén đầy đủ của hệ số DC mặc dù nó hỗ trợ các xử lý.

Do đó, để thu được hiệu quả nén đầy đủ của hệ số DC (cũng được gọi là thành phần dòng một chiều), mà là hệ số của thành phần DC (thành phần dòng một chiều), độ chệch lệch giữa hệ số DC và hệ số khác được xác định, và giá trị chệch lệch được truyền thay vì hệ số DC. Tức là, giá trị chệch lệch là thông tin để tính toán hệ số DC, và, nói cách khác, về cơ bản tương đương với hệ số DC. Tuy nhiên, giá trị chệch lệch thường nhỏ hơn hệ số DC. Do đó, việc truyền của

giá trị chênh lệch thay vì hệ số DC có thể thu được việc làm giảm của lượng mã hóa.

Trong phần mô tả sau đây, nhằm thuận tiện việc mô tả, danh sách chia tỷ lệ (ma trận lượng tử hóa) có kích cỡ 8×8 . Ví dụ cụ thể của phương pháp truyền độ chênh lệch giữa hệ số DC và hệ số khác, thay vì hệ số DC, được mô tả trên đây sẽ được mô tả sau đây.

(2) Cú pháp đối với phương pháp thứ nhất

Ví dụ, 65 hệ số có thể được truyền nhờ sử dụng DPCM (Differential Pulse Code Modulation, điều chế mã xung vi sai), trong đó hệ số DC được xem là thành phần có vị trí tại phần đầu của ma trận 8×8 (các hệ số AC) (phương pháp thứ nhất).

Tức là, đầu tiên, độ chênh lệch giữa hằng số định trước và hệ số DC được tính toán, và được sử dụng như là hệ số khởi tạo của dữ liệu DPCM. Sau đó, độ chênh lệch giữa hệ số DC và hệ số AC khởi tạo được tính toán, và được sử dụng như là hệ số thứ hai của dữ liệu DPCM. Sau đó, độ chênh lệch giữa hệ số AC khởi tạo và hệ số AC thứ hai được tính toán, và được sử dụng như là hệ số thứ ba của dữ liệu DPCM. Sau đó, độ chênh lệch từ hệ số AC ngay trước đó được tính toán, và được sử dụng như là hệ số thứ tư của dữ liệu DPCM, và các hệ số phía sau của dữ liệu DPCM được xác định theo cách tương tự như được mô tả trên đây. Các hệ số của dữ liệu DPCM được tạo ra theo cách được mô tả trên đây được truyền tuần tự, bắt đầu từ hệ số khởi tạo.

Do đó, tỷ số nén có thể được cải thiện khi các giá trị của hệ số (0, 0) (hệ số AC) của ma trận 8×8 và hệ số DC là gần nhau. Bằng cách thực hiện phương pháp thứ nhất được mô tả trên đây, thiết bị mã hóa ảnh có thể xử lý hệ số DC theo cách tương tự như của các hệ số AC (các hệ số dòng xoay chiều), mà là các hệ số của các thành phần AC (cũng được gọi là các thành phần dòng xoay chiều). Lưu ý rằng, để thực hiện phương pháp thứ nhất được mô tả trên đây, thiết bị giải mã ảnh mà các hệ số được mô tả trên đây được truyền tới đó cần điều khiển đặc biệt chỉ hệ số khởi tạo. Cụ thể, thiết bị giải mã ảnh cần tách hệ số DC từ trong số các hệ số AC.

Fig.13 minh họa cú pháp của danh sách chia tỷ lệ trong trường hợp được

mô tả trên đây. Trong ví dụ được minh họa trên Fig.13, 65 giá trị chênh lệch (scaling_list_delta_coef) giữa các hệ số được đọc, và trong số các hệ số (nextcoef) được xác định từ các giá trị chênh lệch, hệ số (nextcoef) nằm tại phần đầu được sử dụng là hệ số DC (scaling_list_dc_coef) trong khi các hệ số khác được sử dụng là các hệ số AC (ScalingList[i]).

Thiết bị mã hóa ảnh mà thực hiện cú pháp đối với phương pháp thứ nhất được mô tả trên đây sẽ được mô tả sau đây.

2-2. Thiết bị mã hóa ảnh

Fig.14 là sơ đồ khối minh họa cấu trúc ví dụ của thiết bị mã hóa ảnh 10 theo phương án của sáng chế. Thiết bị mã hóa ảnh 10 được minh họa trên Fig.14 là thiết bị xử lý ảnh mà sáng chế được áp dụng cho và có cấu trúc để mã hóa dữ liệu ảnh đầu vào và xuất ra dữ liệu ảnh được mã hóa. Việc dẫn tới Fig.14, thiết bị mã hóa ảnh 10 bao gồm bộ chuyển đổi A/D (tương tự thành số) 11 (A/D), bộ đệm sắp xếp lại 12, bộ trừ 13, bộ lượng tử hóa/biến đổi trực giao 14, bộ mã hóa không tổn hao 16, bộ đệm tích lũy 17, bộ điều khiển tốc độ 18, bộ giải lượng tử hóa 21, bộ biến đổi trực giao ngược 22, bộ cộng 23, bộ lọc giải khối 24, bộ nhớ khung 25, bộ lựa chọn 26, bộ dự đoán trong 30, bộ tìm kiếm chuyển động 40, và bộ lựa chọn chế độ 50.

Bộ chuyển đổi A/D 11 chuyển đổi tín hiệu ảnh được đưa vào trong dạng tương tự thành dữ liệu ảnh trong dạng số, và xuất ra chuỗi dữ liệu ảnh số tới bộ đệm sắp xếp lại 12.

Bộ đệm sắp xếp lại 12 sắp xếp lại các ảnh được chứa trong chuỗi dữ liệu ảnh được đưa vào từ bộ chuyển đổi A/D 11. Sau khi sắp xếp lại các ảnh theo cấu trúc GOP (Group of Pictures, nhóm của các ảnh) để sử dụng trong xử lý mã hóa, bộ đệm sắp xếp lại 12 xuất ra dữ liệu ảnh trong đó các ảnh được sắp xếp lại tới bộ trừ 13, bộ dự đoán trong 30, và bộ tìm kiếm chuyển động 40.

Bộ trừ 13 được cấp dữ liệu ảnh được đưa vào từ bộ đệm sắp xếp lại 12 và dữ liệu ảnh dự đoán được lựa chọn bởi bộ lựa chọn chế độ 50, mà sẽ được mô tả dưới đây. Bộ trừ 13 tính toán dữ liệu sai số dự đoán mà biểu diễn độ chênh lệch giữa dữ liệu ảnh được đưa vào từ bộ đệm sắp xếp lại 12 và dữ liệu ảnh dự đoán được đưa vào từ bộ lựa chọn chế độ 50, và xuất ra dữ liệu sai số dự đoán được

tính toán tới bộ lượng tử hóa/biến đổi trực giao 14.

Bộ lượng tử hóa/biến đổi trực giao 14 thực hiện việc biến đổi trực giao và lượng tử hóa trên dữ liệu sai số dự đoán được đưa vào từ bộ trừ 13, và xuất ra dữ liệu hệ số biến đổi được lượng tử hóa (sau đây gọi là dữ liệu được lượng tử hóa) tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21. Tốc độ bit của dữ liệu được lượng tử hóa được xuất ra từ bộ lượng tử hóa/biến đổi trực giao 14 được điều khiển theo tín hiệu điều khiển tốc độ được cấp từ bộ điều khiển tốc độ 18. Cấu trúc chi tiết của bộ lượng tử hóa/biến đổi trực giao 14 sẽ được mô tả tiếp sau đây.

Bộ mã hóa không tổn hao 16 được cấp dữ liệu được lượng tử hóa được đưa vào từ bộ lượng tử hóa/biến đổi trực giao 14, thông tin để tạo ra danh sách chia tỷ lệ (hoặc ma trận lượng tử hóa) trên phía giải mã, và thông tin liên quan đến dự đoán trong hoặc dự đoán liên đới mà được lựa chọn bởi bộ lựa chọn chế độ 50. Thông tin liên quan đến dự đoán trong có thể bao gồm, ví dụ, thông tin chế độ dự đoán chỉ báo chế độ dự đoán trong tối ưu đối với mỗi khối. Ngoài ra, thông tin liên quan đến dự đoán liên đới có thể bao gồm, ví dụ, thông tin chế độ dự đoán để dự đoán theo từng khối của các vector chuyển động, thông tin vector chuyển động vi sai, thông tin ảnh tham chiếu, và v.v. Ngoài ra, thông tin để tạo ra danh sách chia tỷ lệ trên phía giải mã có thể bao gồm thông tin nhận dạng chỉ báo kích cỡ tối đa của danh sách chia tỷ lệ cần được truyền (hoặc ma trận chênh lệch giữa danh sách chia tỷ lệ (ma trận lượng tử hóa) và ma trận dự đoán của nó).

Bộ mã hóa không tổn hao 16 thực hiện xử lý mã hóa không tổn hao trên dữ liệu được lượng tử hóa để tạo ra dòng được mã hóa. Việc mã hóa không tổn hao được thực hiện bởi bộ mã hóa không tổn hao 16 có thể là, ví dụ, mã hóa độ dài biến thiên, mã hóa số học, hoặc loại tương tự. Ngoài ra, bộ mã hóa không tổn hao 16 ghép kênh thông tin để tạo ra danh sách chia tỷ lệ vào đoạn đầu (ví dụ, tập tham số chuỗi và tập tham số ảnh) của dòng được mã hóa. Bộ mã hóa không tổn hao 16 còn ghép kênh thông tin liên quan đến dự đoán trong hoặc dự đoán liên đới được mô tả trên đây vào đoạn đầu của dòng được mã hóa. Sau đó, bộ mã hóa không tổn hao 16 xuất ra dòng được mã hóa được tạo ra tới bộ đệm tích lũy 17.

Bộ đệm tích lũy 17 tích lũy tạm thời dòng được mã hóa được đưa vào từ bộ mã hóa không tổn hao 16, nhờ sử dụng vật lưu trữ như bộ nhớ bán dẫn. Sau đó, bộ đệm tích lũy 17 xuất ra dòng được mã hóa được tích lũy tại tốc độ tương ứng với băng thông của đường truyền (hoặc đường xuất ra từ thiết bị mã hóa ảnh 10).

Bộ điều khiển tốc độ 18 giám sát bộ đệm tích lũy 17 để kiểm tra khả năng khả dụng của dung lượng. Bộ điều khiển tốc độ 18 tạo ra tín hiệu điều khiển tốc độ theo dung lượng khả dụng của bộ đệm tích lũy 17, và xuất ra tín hiệu điều khiển tốc độ được tạo ra tới bộ lượng tử hóa/biến đổi trực giao 14. Ví dụ, khi dung lượng khả dụng của bộ đệm tích lũy 17 là thấp, bộ điều khiển tốc độ 18 tạo ra tín hiệu điều khiển tốc độ để làm giảm tốc độ bit của dữ liệu được lượng tử hóa. Ngoài ra, ví dụ, khi lượng khả dụng của bộ đệm tích lũy 17 là đủ cao, bộ điều khiển tốc độ 18 tạo ra tín hiệu điều khiển tốc độ để tăng tốc độ bit của dữ liệu được lượng tử hóa.

Bộ giải lượng tử hóa 21 thực hiện xử lý giải lượng tử hóa trên dữ liệu được lượng tử hóa được đưa vào từ bộ lượng tử hóa/biến đổi trực giao 14. Sau đó, bộ giải lượng tử hóa 21 xuất ra dữ liệu hệ số biến đổi thu được thông qua xử lý giải lượng tử hóa tới bộ biến đổi trực giao ngược 22.

Bộ biến đổi trực giao ngược 22 thực hiện xử lý biến đổi trực giao ngược trên dữ liệu hệ số biến đổi được đưa vào từ bộ giải lượng tử hóa 21 để khôi phục dữ liệu sai số dự đoán. Sau đó, bộ biến đổi trực giao ngược 22 xuất ra dữ liệu sai số dự đoán được khôi phục tới bộ cộng 23.

Bộ cộng 23 cộng với nhau dữ liệu sai số dự đoán được khôi phục được đưa vào từ bộ biến đổi trực giao ngược 22 và dữ liệu ảnh dự đoán được đưa vào từ bộ lựa chọn chế độ 50 để tạo ra dữ liệu ảnh được giải mã. Sau đó, bộ cộng 23 xuất ra dữ liệu ảnh được giải mã được tạo ra tới bộ lọc giải khối 24 và bộ nhớ khung 25.

Bộ lọc giải khối 24 thực hiện xử lý lọc để làm giảm nhiễu khối gây bởi việc mã hóa của ảnh. Bộ lọc giải khối 24 lọc dữ liệu ảnh được giải mã được đưa vào từ bộ cộng 23 để loại bỏ (hoặc ít nhất làm giảm) các nhiễu khối, và xuất ra dữ liệu ảnh được giải mã được lọc tới bộ nhớ khung 25.

Bộ nhớ khung 25 lưu trữ dữ liệu ảnh được giải mã được đưa vào từ bộ cộng 23 và dữ liệu ảnh được giải mã được lọc được đưa vào từ bộ lọc giải khối 24, nhờ sử dụng vật lưu trữ.

Bộ lựa chọn 26 đọc dữ liệu ảnh được giải mã cần được lọc, mà được sử dụng cho việc dự đoán trong, từ bộ nhớ khung 25, và cấp dữ liệu ảnh được giải mã được đọc tới bộ dự đoán trong 30 làm dữ liệu ảnh tham chiếu. Bộ lựa chọn 26 còn đọc dữ liệu ảnh được giải mã được lọc, mà được sử dụng cho dự đoán liên đới, từ bộ nhớ khung 25, và cấp dữ liệu ảnh được giải mã được đọc tới bộ tìm kiếm chuyển động 40 làm dữ liệu ảnh tham chiếu.

Bộ dự đoán trong 30 thực hiện xử lý dự đoán trong trong mỗi chế độ dự đoán trong trên cơ sở của dữ liệu ảnh cần được mã hóa, mà được đưa vào từ bộ đệm sắp xếp lại 12, và dữ liệu ảnh được giải mã được cấp thông qua bộ lựa chọn 26. Ví dụ, bộ dự đoán trong 30 đánh giá kết quả dự đoán thu được trong mỗi chế độ dự đoán trong nhờ sử dụng hàm giá trị định trước. Sau đó, bộ dự đoán trong 30 lựa chọn chế độ dự đoán trong mà tối thiểu hóa giá trị hàm giá trị, tức là, chế độ dự đoán trong mà cung cấp tỷ số nén cao nhất, làm chế độ dự đoán trong tối ưu. Ngoài ra, bộ dự đoán trong 30 xuất ra thông tin chế độ dự đoán chỉ báo chế độ dự đoán trong tối ưu, dữ liệu ảnh dự đoán, và thông tin liên quan đến dự đoán trong, như giá trị hàm giá trị, tới bộ lựa chọn chế độ 50.

Bộ tìm kiếm chuyển động 40 thực hiện xử lý dự đoán liên đới (hoặc xử lý dự đoán khung liên đới) trên cơ sở của dữ liệu ảnh cần được mã hóa, mà được đưa vào từ bộ đệm sắp xếp lại 12, và dữ liệu ảnh được giải mã được cấp thông qua bộ lựa chọn 26. Ví dụ, bộ tìm kiếm chuyển động 40 đánh giá kết quả dự đoán thu được trong mỗi chế độ dự đoán nhờ sử dụng hàm giá trị định trước. Sau đó, bộ tìm kiếm chuyển động 40 lựa chọn chế độ dự đoán mà tối thiểu hóa giá trị hàm giá trị, tức là, chế độ dự đoán mà cung cấp tỷ số nén cao nhất, như là chế độ dự đoán tối ưu. Ngoài ra, bộ tìm kiếm chuyển động 40 tạo ra dữ liệu ảnh dự đoán theo chế độ dự đoán tối ưu. Bộ tìm kiếm chuyển động 40 xuất ra thông tin liên quan đến việc dự đoán liên đới mà bao gồm thông tin chế độ dự đoán chỉ báo chế độ dự đoán tối ưu được lựa chọn, dữ liệu ảnh dự đoán, và thông tin liên quan đến việc dự đoán liên đới, như giá trị hàm giá trị, tới bộ lựa chọn chế độ 50.

Bộ lựa chọn chế độ 50 so sánh giá trị hàm giá trị đối với dự đoán trong, mà được đưa vào từ bộ dự đoán trong 30, với giá trị hàm giá trị đối với dự đoán liên đới, mà được đưa vào từ bộ tìm kiếm chuyển động 40. Sau đó, bộ lựa chọn chế độ 50 lựa chọn kỹ thuật dự đoán có giá trị nhỏ hơn của các giá trị hàm giá trị đối với dự đoán trong và dự đoán liên đới. Nếu dự đoán trong được lựa chọn, bộ lựa chọn chế độ 50 xuất ra thông tin liên quan đến việc dự đoán trong tới bộ mã hóa không tổn hao 16, và cũng xuất ra dữ liệu ảnh dự đoán tới bộ trừ 13 và bộ cộng 23. Ngoài ra, nếu dự đoán liên đới được lựa chọn, bộ lựa chọn chế độ 50 xuất ra thông tin liên quan đến việc dự đoán liên đới được mô tả trên đây tới bộ mã hóa không tổn hao 16, và cũng xuất ra dữ liệu ảnh dự đoán tới bộ trừ 13 và bộ cộng 23.

2-3. Cấu trúc ví dụ của bộ lượng tử hóa/biến đổi trực giao

Fig.15 là sơ đồ khối minh họa ví dụ của cấu trúc chi tiết của bộ lượng tử hóa/biến đổi trực giao 14 của thiết bị mã hóa ảnh 10 được minh họa trên Fig.14. Viện dẫn tới Fig.15, bộ lượng tử hóa/biến đổi trực giao 14 bao gồm bộ lựa chọn 110, bộ biến đổi trực giao 120, bộ lượng tử hóa 130, bộ đệm danh sách chia tỷ lệ 140, và bộ xử lý ma trận 150.

(1) Bộ lựa chọn

Bộ lựa chọn 110 lựa chọn đơn vị biến đổi (TU) cần được sử dụng cho biến đổi trực giao của dữ liệu ảnh cần được mã hóa từ trong số các đơn vị biến đổi có các kích cỡ khác nhau. Các ví dụ của các kích cỡ có thể của các đơn vị biến đổi có thể được lựa chọn bởi bộ lựa chọn 110 bao gồm 4x4 và 8x8 đối với H.264/AVC (Advanced Video Coding, mã hóa video cải tiến), và bao gồm 4x4, 8x8, 16x16, và 32x32 đối với HEVC (High Efficiency Video Coding, mã hóa video hiệu quả cao). Bộ lựa chọn 110 có thể lựa chọn đơn vị biến đổi theo, ví dụ, kích cỡ hoặc chất lượng của ảnh cần được mã hóa, đặc tính của thiết bị mã hóa ảnh 10, hoặc loại tương tự. Việc lựa chọn của đơn vị biến đổi bởi bộ lựa chọn 110 có thể được tinh chỉnh thủ công bởi người dùng mà phát triển thiết bị mã hóa ảnh 10. Sau đó, bộ lựa chọn 110 xuất ra thông tin mà chỉ rõ kích cỡ của đơn vị biến đổi được lựa chọn tới bộ biến đổi trực giao 120, bộ lượng tử hóa 130, bộ mã hóa không tổn hao 16, và bộ giải lượng tử hóa 21.

(2) Bộ biến đổi trực giao

Bộ biến đổi trực giao 120 thực hiện việc biến đổi trực giao trên dữ liệu ảnh (tức là, dữ liệu sai số dự đoán) được cấp từ bộ trừ 13, trong các đơn vị của đơn vị biến đổi được lựa chọn bởi bộ lựa chọn 110. Biến đổi trực giao được thực hiện bởi bộ biến đổi trực giao 120 có thể là, ví dụ, biến đổi cosin rời rạc (discrete cosine transform, DCT), biến đổi Karhunen-Loève, hoặc loại tương tự. Sau đó, bộ biến đổi trực giao 120 xuất ra dữ liệu hệ số biến đổi thu được thông qua xử lý biến đổi trực giao tới bộ lượng tử hóa 130.

(3) Bộ lượng tử hóa

Bộ lượng tử hóa 130 lượng tử hóa dữ liệu hệ số biến đổi được tạo ra bởi bộ biến đổi trực giao 120, bằng cách sử dụng danh sách chia tỷ lệ tương ứng với đơn vị biến đổi được lựa chọn bởi bộ lựa chọn 110. Ngoài ra, bộ lượng tử hóa 130 chuyển đổi kích cỡ bước lượng tử hóa theo tín hiệu điều khiển tốc độ được cấp từ bộ điều khiển tốc độ 18 để thay đổi tốc độ bit của dữ liệu được lượng tử hóa cần được xuất ra.

Ngoài ra, bộ lượng tử hóa 130 làm cho các tập của các danh sách chia tỷ lệ một cách tương ứng mà tương ứng với các đơn vị biến đổi có thể được lựa chọn bởi bộ lựa chọn 110 được lưu trữ trong bộ đệm danh sách chia tỷ lệ 140. Ví dụ, như trong HEVC, nếu có bốn kích cỡ có thể của các đơn vị biến đổi, cụ thể là, 4x4, 8x8, 16x16, và 32x32, bốn tập của các danh sách chia tỷ lệ một cách tương ứng mà tương ứng với bốn kích cỡ có thể được lưu trữ trong bộ đệm danh sách chia tỷ lệ 140. Lưu ý rằng nếu danh sách chia tỷ lệ được chỉ rõ được sử dụng đối với kích cỡ định trước, chỉ cần chỉ báo rằng danh sách chia tỷ lệ được chỉ rõ được sử dụng (danh sách chia tỷ lệ được xác định bởi người dùng không được sử dụng) có thể được lưu trữ trong bộ đệm danh sách chia tỷ lệ 140 kết hợp với kích cỡ định trước.

Tập của các danh sách chia tỷ lệ mà có thể được sử dụng bởi bộ lượng tử hóa 130 có thể được thiết lập một cách điển hình đối với mỗi chuỗi của dòng được mã hóa. Ngoài ra, bộ lượng tử hóa 130 có thể cập nhật tập của các danh sách chia tỷ lệ mà được thiết lập đối với mỗi chuỗi trên cơ sở từng ảnh. Thông tin để điều khiển việc thiết lập và cập nhật của tập của các danh sách chia tỷ lệ

có thể được chèn vào trong, ví dụ, tập tham số chuỗi và tập tham số ảnh.

(4) Bộ đệm danh sách chia tỷ lệ

Bộ đệm danh sách chia tỷ lệ 140 lưu trữ tạm thời tập của các danh sách chia tỷ lệ một cách tương ứng mà tương ứng với các đơn vị biến đổi có thể được lựa chọn bởi bộ lựa chọn 110, nhờ sử dụng vật lưu trữ như bộ nhớ bán dẫn. Tập của các danh sách chia tỷ lệ được lưu trữ trong bộ đệm danh sách chia tỷ lệ 140 được viện dẫn tới khi bộ xử lý ma trận 150 thực hiện xử lý được mô tả dưới đây.

(5) Bộ xử lý ma trận

Bộ xử lý ma trận 150 mã hóa danh sách chia tỷ lệ cần được sử dụng để mã hóa (lượng tử hóa). Sau đó, dữ liệu được mã hóa của danh sách chia tỷ lệ (sau đây được gọi là dữ liệu danh sách chia tỷ lệ được mã hóa) được tạo ra bởi bộ xử lý ma trận 150 được xuất ra tới bộ mã hóa không tổn hao 16, và có thể được chèn vào đoạn đầu của dòng được mã hóa.

2-4. Cấu trúc ví dụ chi tiết của bộ xử lý ma trận

Fig.16 là sơ đồ khối minh họa ví dụ về cấu trúc chi tiết hơn của bộ xử lý ma trận 150. Viện dẫn tới Fig.16, bộ xử lý ma trận 150 bao gồm bộ dự đoán 161, bộ tạo ma trận chệnh lệch 162, bộ biến đổi kích cỡ ma trận chệnh lệch 163, bộ mã hóa entropy 164, bộ giải mã 165, và bộ đầu ra 166.

(1) Bộ dự đoán

Bộ dự đoán 161 tạo ra ma trận dự đoán. Như được minh họa trên Fig.16, bộ dự đoán 161 bao gồm bộ sao chép 171 và bộ tạo ma trận dự đoán 172.

Trong chế độ sao chép, bộ sao chép 171 sao chép danh sách chia tỷ lệ được truyền trước đó, và sử dụng ma trận lượng tử hóa được sao chép làm ma trận dự đoán (hoặc dự đoán danh sách chia tỷ lệ của đơn vị biến đổi trực giao cần được xử lý). Cụ thể hơn, bộ sao chép 171 thu được kích cỡ và ID danh sách (ListID) của danh sách chia tỷ lệ được truyền trước đó từ bộ lưu trữ 202 trong bộ giải mã 165. Kích cỡ này là thông tin chỉ báo kích cỡ của danh sách chia tỷ lệ (phạm vi từ, ví dụ, 4x4 tới 32x32). ID danh sách là thông tin chỉ báo loại của dữ liệu sai số dự đoán cần được lượng tử hóa.

Ví dụ, ID danh sách bao gồm thông tin nhận dạng chỉ báo rằng dữ liệu sai số dự đoán cần được lượng tử hóa là dữ liệu sai số dự đoán (Intra Luma) của

thành phần độ chói mà được tạo ra nhờ sử dụng ảnh dự đoán được đưa vào dự đoán trong, dữ liệu sai số dự đoán (Intra Cr) của thành phần chênh lệch màu (Cr) mà được tạo ra nhờ sử dụng ảnh dự đoán được đưa vào dự đoán trong, dữ liệu sai số dự đoán (Intra Cb) của thành phần chênh lệch màu (Cb) mà được tạo ra nhờ sử dụng ảnh dự đoán được đưa vào dự đoán trong, hoặc dữ liệu sai số dự đoán (Inter Luma) của thành phần độ chói mà được tạo ra nhờ sử dụng ảnh dự đoán được đưa vào dự đoán liên đới.

Bộ sao chép 171 lựa chọn, như là danh sách chia tỷ lệ cần được sao chép, danh sách chia tỷ lệ được truyền trước đó có kích cỡ tương tự như danh sách chia tỷ lệ (danh sách chia tỷ lệ của đơn vị biến đổi trực giao cần được xử lý) được đưa vào bộ xử lý ma trận 150, và cấp ID danh sách của danh sách chia tỷ lệ cần được sao chép tới bộ đầu ra 166 để xuất ra ID danh sách tới các thiết bị phía ngoài bộ xử lý ma trận 150 (bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21). Tức là, trong trường hợp này, chỉ ID danh sách được truyền tới phía giải mã (hoặc được chứa trong dữ liệu được mã hóa) như là thông tin chỉ báo ma trận dự đoán được tạo ra bằng cách sao chép danh sách chia tỷ lệ được truyền trước đó. Do đó, thiết bị mã hóa ảnh 10 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Ngoài ra, trong chế độ thông thường, bộ tạo ma trận dự đoán 172 thu được danh sách chia tỷ lệ được truyền trước đó từ bộ lưu trữ 202 trong bộ giải mã 165, và tạo ra ma trận dự đoán nhờ sử dụng danh sách chia tỷ lệ (hoặc dự đoán danh sách chia tỷ lệ của đơn vị biến đổi trực giao cần được xử lý). Bộ tạo ma trận dự đoán 172 cấp ma trận dự đoán được tạo ra tới bộ tạo ma trận chênh lệch 162.

(2) Bộ tạo ma trận chênh lệch

Bộ tạo ma trận chênh lệch 162 tạo ra ma trận chênh lệch (ma trận dư) mà là độ chênh lệch giữa ma trận dự đoán được cấp từ bộ dự đoán 161 (bộ tạo ma trận dự đoán 172) và danh sách chia tỷ lệ được đưa vào bộ xử lý ma trận 150. Như được minh họa trên Fig.16, bộ tạo ma trận chênh lệch 162 bao gồm bộ biến đổi kích cỡ ma trận dự đoán 181, bộ tính toán 182, và bộ lượng tử hóa 183.

Bộ biến đổi kích cỡ ma trận dự đoán 181 biến đổi (sau đây cũng được gọi là chuyển đổi) kích cỡ của ma trận dự đoán được cấp từ bộ tạo ma trận dự đoán

172 sao cho kích cỡ của ma trận dự đoán khớp với kích cỡ của danh sách chia tỷ lệ được đưa vào từ bộ xử lý ma trận 150.

Ví dụ, nếu kích cỡ của ma trận dự đoán lớn hơn kích cỡ của danh sách chia tỷ lệ, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển đổi giảm dần (sau đây cũng được gọi là chuyển đổi xuống) ma trận dự đoán. Cụ thể hơn, ví dụ, khi ma trận dự đoán có kích cỡ 16×16 và danh sách chia tỷ lệ có kích cỡ 8×8 , bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển đổi xuống ma trận dự đoán thành ma trận dự đoán 8×8 . Lưu ý rằng phương pháp chuyển đổi xuống bất kỳ có thể được sử dụng. Ví dụ, bộ biến đổi kích cỡ ma trận dự đoán 181 có thể làm giảm số lượng thành phần trong ma trận dự đoán (sau đây cũng được gọi là lấy mẫu xuống) bằng cách sử dụng bộ lọc (thông qua tính toán). Ngoài ra, bộ biến đổi kích cỡ ma trận dự đoán 181 cũng có thể làm giảm số lượng thành phần trong ma trận dự đoán bằng cách, ví dụ, như được minh họa trên Fig.17, làm mỏng một vài thành phần (ví dụ, chỉ các thành phần được đánh số chẵn (trên Fig.17, các thành phần trong khối màu đen) trong số các thành phần hai chiều) mà không sử dụng bộ lọc (sau đây cũng được gọi là lấy mẫu con).

Ngoài ra, ví dụ, nếu kích cỡ của ma trận dự đoán nhỏ hơn kích cỡ của danh sách chia tỷ lệ, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển đổi tăng dần (sau đây cũng được gọi là chuyển đổi lên) ma trận dự đoán. Cụ thể hơn, ví dụ, khi ma trận dự đoán có kích cỡ 8×8 và danh sách chia tỷ lệ có kích cỡ 16×16 , bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển đổi lên ma trận dự đoán thành ma trận dự đoán 16×16 . Lưu ý rằng phương pháp bất kỳ để chuyển đổi lên có thể được sử dụng. Ví dụ, bộ biến đổi kích cỡ ma trận dự đoán 181 có thể tăng số lượng thành phần trong ma trận dự đoán (sau đây cũng được gọi là lấy mẫu lên) bằng cách sử dụng bộ lọc (thông qua tính toán). Ngoài ra, bộ biến đổi kích cỡ ma trận dự đoán 181 cũng có thể tăng số lượng thành phần trong ma trận dự đoán bằng cách, ví dụ, sao chép các thành phần riêng biệt trong ma trận dự đoán mà không sử dụng bộ lọc (sau đây cũng được gọi là lấy mẫu con ngược).

Bộ biến đổi kích cỡ ma trận dự đoán 181 cấp ma trận dự đoán ma kích cỡ của nó được làm cho khớp với kích cỡ của danh sách chia tỷ lệ tới bộ tính toán 182.

Bộ tính toán 182 trừ danh sách chia tỷ lệ được đưa vào bộ xử lý ma trận 150 từ ma trận dự đoán được cấp từ bộ biến đổi kích cỡ ma trận dự đoán 181, và tạo ra ma trận chênh lệch (ma trận dư). Bộ tính toán 182 cấp ma trận chênh lệch được tính toán tới bộ lượng tử hóa 183.

Bộ lượng tử hóa 183 lượng tử hóa ma trận chênh lệch được cấp từ bộ tính toán 182. Bộ lượng tử hóa 183 cấp ma trận chênh lệch được lượng tử hóa tới bộ biến đổi kích cỡ ma trận chênh lệch 163. Bộ lượng tử hóa 183 còn cấp thông tin được sử dụng cho việc lượng tử hóa, như các tham số lượng tử hóa, tới bộ đầu ra 166 để xuất ra thông tin này tới các thiết bị phía ngoài bộ xử lý ma trận 150 (bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21). Lưu ý rằng bộ lượng tử hóa 183 có thể được bỏ qua (tức là, việc lượng tử hóa của ma trận chênh lệch có thể không cần thiết được thực hiện).

(3) Bộ biến đổi kích cỡ ma trận chênh lệch

Bộ biến đổi kích cỡ ma trận chênh lệch 163 chuyển đổi kích cỡ của ma trận chênh lệch (dữ liệu được lượng tử hóa) được cấp từ bộ tạo ma trận chênh lệch 162 (bộ lượng tử hóa 183) thành kích cỡ nhỏ hơn hoặc bằng kích cỡ tối đa được phép trong việc truyền (sau đây cũng được gọi là kích cỡ truyền), nếu cần. Kích cỡ tối đa có thể có giá trị tùy chọn bất kỳ, và là, ví dụ, 8x8.

Dữ liệu được mã hóa được xuất ra từ thiết bị mã hóa ảnh 10 được truyền tới thiết bị giải mã ảnh tương ứng với thiết bị mã hóa ảnh 10 thông qua, ví dụ, đường truyền hoặc vật lưu trữ, và được giải mã bởi thiết bị giải mã ảnh. Giới hạn trên của kích cỡ (kích cỡ tối đa) của ma trận chênh lệch (dữ liệu được lượng tử hóa) trong việc truyền này, hoặc trong dữ liệu được mã hóa được xuất ra từ thiết bị mã hóa ảnh 10, được thiết lập trong thiết bị mã hóa ảnh 10.

Nếu kích cỡ của ma trận chênh lệch lớn hơn kích cỡ tối đa, bộ biến đổi kích cỡ ma trận chênh lệch 163 chuyển đổi xuống ma trận chênh lệch sao cho kích cỡ của ma trận chênh lệch trở nên nhỏ hơn hoặc bằng kích cỡ tối đa.

Lưu ý rằng, tương tự việc chuyển đổi xuống của ma trận dự đoán được mô tả trên đây, ma trận chênh lệch có thể được chuyển đổi xuống nhờ sử dụng phương pháp bất kỳ. Ví dụ, việc lấy mẫu xuống có thể được thực hiện nhờ sử dụng bộ lọc hoặc loại tương tự, hoặc việc lấy mẫu con mà bao gồm làm mỏng

các thành phần có thể được thực hiện.

Ngoài ra, ma trận chênh lệch được chuyển đổi xuống có thể có kích cỡ bất kỳ nhỏ hơn so với kích cỡ tối đa. Tuy nhiên, nói chung, độ chênh lệch về kích cỡ giữa trước và sau khi biến đổi càng lớn, sai số trở nên càng lớn. Do đó có mong muốn rằng ma trận chênh lệch được chuyển đổi xuống thành kích cỡ tối đa.

Bộ biến đổi kích cỡ ma trận chênh lệch 163 cấp ma trận chênh lệch được chuyển đổi xuống tới bộ mã hóa entropy 164. Lưu ý rằng nếu kích cỡ của ma trận chênh lệch nhỏ hơn kích cỡ tối đa, việc chuyển đổi xuống được mô tả trên đây là không cần thiết, và do đó bộ biến đổi kích cỡ ma trận chênh lệch 163 cấp ma trận chênh lệch được đưa vào đó tới bộ mã hóa entropy 164 như ban đầu (tức là, việc chuyển đổi xuống của ma trận chênh lệch được bỏ qua).

(4) Bộ mã hóa entropy

Bộ mã hóa entropy 164 mã hóa ma trận chênh lệch (dữ liệu được lượng tử hóa) được cấp từ bộ biến đổi kích cỡ ma trận chênh lệch 163 nhờ sử dụng phương pháp định trước. Như được minh họa trên Fig.16, bộ mã hóa entropy 164 bao gồm bộ xác định xếp chồng (bộ 135 độ) 191, bộ DPCM (Differential Pulse Code Modulation, điều chế mã xung vi sai) 192, và bộ exp-G 193.

Bộ xác định xếp chồng 191 xác định tính đối xứng của ma trận chênh lệch được cấp từ bộ biến đổi kích cỡ ma trận chênh lệch 163. Nếu phần dư (ma trận chênh lệch) biểu diễn ma trận đối xứng 135 độ, ví dụ, như được minh họa trên Fig.18, bộ xác định xếp chồng 191 loại bỏ dữ liệu (các thành phần ma trận) của phần đối xứng mà là dữ liệu xếp chồng. Nếu phần dư không biểu diễn ma trận đối xứng 135 độ, bộ xác định xếp chồng 191 bỏ qua việc loại bỏ của dữ liệu (các thành phần ma trận). Bộ xác định xếp chồng 191 cấp dữ liệu của ma trận chênh lệch mà từ đó phần đối xứng được loại bỏ, nếu cần, tới bộ DPCM 192.

Bộ DPCM 192 thực hiện mã hóa DPCM của dữ liệu của ma trận chênh lệch mà từ đó phần đối xứng được loại bỏ, nếu cần, mà được cấp từ bộ xác định xếp chồng 191, và tạo ra dữ liệu DPCM. Bộ DPCM 192 cấp dữ liệu DPCM được tạo ra tới bộ exp-G 193.

Bộ exp-G 193 mã hóa dữ liệu DPCM được cấp từ bộ DPCM 192 nhờ sử dụng các mã Golomb theo số mũ có dấu hoặc không dấu (sau đây cũng được gọi

là các mã Golomb theo số mũ). Bộ exp-G 193 cấp kết quả mã hóa tới bộ giải mã 165 và bộ đầu ra 166.

(5) Bộ giải mã

Bộ giải mã 165 khôi phục danh sách chia tỷ lệ từ dữ liệu được cấp từ bộ exp-G 193. Bộ giải mã 165 cấp thông tin liên quan đến danh sách chia tỷ lệ được khôi phục tới bộ dự đoán 161 như là danh sách chia tỷ lệ được truyền trước đó.

Như được minh họa trên Fig.16, bộ giải mã 165 bao gồm bộ khôi phục danh sách chia tỷ lệ 201 và bộ lưu trữ 202.

Bộ khôi phục danh sách chia tỷ lệ 201 giải mã các mã Golomb theo số mũ được cấp từ bộ mã hóa entropy 164 (bộ exp-G 193) để khôi phục danh sách chia tỷ lệ cần được đưa vào bộ xử lý ma trận 150. Ví dụ, bộ khôi phục danh sách chia tỷ lệ 201 giải mã các mã Golomb theo số mũ nhờ sử dụng phương pháp tương ứng với phương pháp mã hóa đối với bộ mã hóa entropy 164, và thu được ma trận chệnh lệch bằng cách thực hiện việc biến đổi ngược với việc biến đổi kích cỡ được thực hiện bởi bộ biến đổi kích cỡ ma trận chệnh lệch 163 và thực hiện việc giải lượng tử hóa tương ứng với việc lượng tử hóa được thực hiện bởi bộ lượng tử hóa 183. Bộ khôi phục danh sách chia tỷ lệ 201 còn trừ ma trận chệnh lệch thu được từ ma trận dự đoán để khôi phục danh sách chia tỷ lệ.

Bộ khôi phục danh sách chia tỷ lệ 201 cấp danh sách chia tỷ lệ được khôi phục tới bộ lưu trữ 202 để lưu trữ kết hợp với kích cỡ và ID danh sách của danh sách chia tỷ lệ.

Bộ lưu trữ 202 lưu trữ thông tin liên quan đến danh sách chia tỷ lệ được cấp từ bộ khôi phục danh sách chia tỷ lệ 201. Thông tin liên quan đến danh sách chia tỷ lệ được lưu trữ trong bộ lưu trữ 202 được sử dụng để tạo ra các ma trận dự đoán của các đơn vị biến đổi trực giao khác mà được xử lý sau đó đúng lúc. Tức là, bộ lưu trữ 202 cấp thông tin được lưu trữ liên quan đến danh sách chia tỷ lệ tới bộ dự đoán 161 như là thông tin liên quan đến danh sách chia tỷ lệ được truyền trước đó.

Lưu ý rằng, thay vì lưu trữ thông tin liên quan đến danh sách chia tỷ lệ được khôi phục theo cách được mô tả trên đây, bộ lưu trữ 202 có thể lưu trữ danh sách chia tỷ lệ được đưa vào bộ xử lý ma trận 150 kết hợp với kích cỡ và

ID danh sách của danh sách chia tỷ lệ được đưa vào. Trong trường hợp này, bộ khôi phục danh sách chia tỷ lệ 201 có thể được bỏ qua.

(6) Bộ đầu ra

Bộ đầu ra 166 xuất ra các loại thông tin khác nhau được cấp tới các thiết bị phía ngoài bộ xử lý ma trận 150. Ví dụ, trong chế độ sao chép, bộ đầu ra 166 cấp ID danh sách của ma trận dự đoán được cấp từ bộ sao chép 171 tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21. Ngoài ra, ví dụ, trong chế độ thông thường, bộ đầu ra 166 cấp các mã Golomb theo số mũ được cấp từ bộ exp-G 193 và các tham số lượng tử hóa được cấp từ bộ lượng tử hóa 183 tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21.

Bộ đầu ra 166 còn cấp thông tin nhận dạng chỉ báo kích cỡ tối đa (kích cỡ truyền) được cho phép trong việc truyền danh sách chia tỷ lệ (hoặc ma trận chênh lệch giữa danh sách chia tỷ lệ và ma trận dự đoán của nó) tới bộ mã hóa không tổn hao 16 như là thông tin để tạo ra danh sách chia tỷ lệ trên phía giải mã. Như được mô tả trên đây, bộ mã hóa không tổn hao 16 tạo ra dòng được mã hóa bao gồm thông tin để tạo ra danh sách chia tỷ lệ, và cấp dòng được mã hóa tới phía giải mã. Thông tin nhận dạng chỉ báo kích cỡ truyền có thể được chỉ rõ trước bởi mức, lược sử, và loại tương tự. Trong trường hợp này, thông tin liên quan đến kích cỡ truyền được chia sẻ trước giữa thiết bị trên phía mã hóa và thiết bị trên phía giải mã. Do đó, việc truyền của thông tin nhận dạng được mô tả trên đây có thể được bỏ qua.

2-5. Cấu trúc ví dụ chi tiết của bộ DPCM

Fig.19 là sơ đồ khối minh họa ví dụ về cấu trúc chi tiết hơn của bộ DPCM 192. Viền dẫn tới Fig.19, bộ DPCM 192 bao gồm bộ mã hóa hệ số DC 211 và bộ DPCM hệ số AC 212.

Bộ mã hóa hệ số DC 211 thu được hệ số DC từ trong số các hệ số được cấp từ bộ xác định xếp chồng 191, trừ giá trị của hệ số DC từ giá trị khởi tạo định trước (ví dụ, 8) để xác định giá trị chênh lệch, và sử dụng giá trị chênh lệch làm giá trị chênh lệch khởi tạo ($i = 0$) (`scaling_list_delta_coef`). Bộ mã hóa hệ số DC 211 cấp giá trị chênh lệch được tính toán (`scaling_list_delta_coef` ($i = 0$)) tới bộ exp-G 193 làm hệ số khởi tạo của danh sách chia tỷ lệ tương ứng với vùng

quan tâm được xử lý.

Bộ DPCM hệ số AC 212 thu được hệ số AC từ trong số các hệ số được cấp từ bộ xác định xếp chồng 191, và trừ giá trị của hệ số AC từ hệ số được xử lý ngay trước đó để xác định giá trị chênh lệch ($\text{scaling_list_delta_coef}(i > 0)$). Bộ DPCM hệ số AC 212 cấp các giá trị chênh lệch được xác định ($\text{scaling_list_delta_coef}(i > 0)$) tới bộ exp-G 193 như là hệ số của danh sách chia tỷ lệ tương ứng với vùng quan tâm được xử lý. Lưu ý rằng khi $i = 1$, hệ số ngay trước đó được biểu diễn bởi $i = 0$. Do đó, “hệ số DC” là hệ số được xử lý ngay trước đó.

Theo cách này, bộ DPCM 192 có thể truyền hệ số DC làm thành phần nằm tại phần đầu của danh sách chia tỷ lệ (các hệ số AC). Do đó, hiệu quả mã hóa của danh sách chia tỷ lệ có thể được cải thiện.

2-6. Dòng xử lý của xử lý mã hóa ma trận lượng tử hóa

Tiếp theo, ví dụ của dòng xử lý của xử lý mã hóa ma trận lượng tử hóa được thực hiện bởi bộ xử lý ma trận 150 được minh họa trên Fig.16 sẽ được mô tả có viển dẫn tới lưu đồ được minh họa trên Fig.20.

Khi xử lý mã hóa ma trận lượng tử hóa được bắt đầu, trong bước S101, bộ dự đoán 161 thu được danh sách chia tỷ lệ (hoặc ma trận lượng tử hóa) đối với vùng hiện tại (cũng được gọi là vùng quan tâm) mà là đơn vị biến đổi trực giao cần được xử lý.

Trong bước S102, bộ dự đoán 161 xác định rằng chế độ hiện tại có phải là chế độ sao chép hay không. Nếu được xác định rằng chế độ hiện tại không phải chế độ sao chép, bộ dự đoán 161 chuyển xử lý sang bước S103.

Trong bước S103, bộ tạo ma trận dự đoán 172 thu được danh sách chia tỷ lệ được truyền trước đó từ bộ lưu trữ 202, và tạo ra ma trận dự đoán nhờ sử dụng danh sách chia tỷ lệ.

Trong bước S104, bộ biến đổi kích cỡ ma trận dự đoán 181 xác định rằng kích cỡ của ma trận dự đoán được tạo ra trong bước S103 có khác với kích cỡ của danh sách chia tỷ lệ đối với vùng hiện tại (vùng quan tâm) thu được trong bước S101 hay không. Nếu được xác định rằng các kích cỡ là khác nhau, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển xử lý sang bước S105.

Trong bước S105, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển đổi kích cỡ của ma trận dự đoán được tạo ra trong bước S103 sang kích cỡ của danh sách chia tỷ lệ đối với vùng hiện tại thu được trong bước S101.

Khi xử lý của bước S105 được hoàn thành, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển xử lý sang bước S106. Nếu được xác định trong bước S104 rằng kích cỡ của ma trận dự đoán là giống như kích cỡ của danh sách chia tỷ lệ, bộ biến đổi kích cỡ ma trận dự đoán 181 chuyển xử lý sang bước S106 trong khi bỏ qua xử lý của bước S105 (hoặc không thực hiện xử lý của bước S105).

Trong bước S106, bộ tính toán 182 trừ danh sách chia tỷ lệ từ ma trận dự đoán để tính toán ma trận chênh lệch giữa ma trận dự đoán và danh sách chia tỷ lệ.

Trong bước S107, bộ lượng tử hóa 183 lượng tử hóa ma trận chênh lệch được tạo ra trong bước S106. Lưu ý rằng xử lý này có thể được bỏ qua.

Trong bước S108, bộ biến đổi kích cỡ ma trận chênh lệch 163 xác định rằng kích cỡ của ma trận chênh lệch được lượng tử hóa có lớn hơn kích cỡ truyền (kích cỡ tối đa được cho phép trong việc truyền) hay không. Nếu được xác định rằng kích cỡ của ma trận chênh lệch được lượng tử hóa lớn hơn kích cỡ truyền, bộ biến đổi kích cỡ ma trận chênh lệch 163 chuyển xử lý sang bước S109, và chuyển đổi xuống ma trận chênh lệch tới kích cỡ truyền hoặc nhỏ hơn.

Khi xử lý của bước S109 được hoàn thành, bộ biến đổi kích cỡ ma trận chênh lệch 163 chuyển xử lý sang bước S110. Ngoài ra, nếu được xác định trong bước S108 rằng kích cỡ của ma trận chênh lệch được lượng tử hóa nhỏ hơn hoặc bằng kích cỡ truyền, bộ biến đổi kích cỡ ma trận chênh lệch 163 chuyển xử lý sang bước S110 trong khi bỏ qua xử lý của bước S109 (hoặc không thực hiện xử lý của bước S109).

Trong bước S110, bộ xác định xếp chồng 191 xác định rằng ma trận chênh lệch được lượng tử hóa có đối xứng 135 độ hay không. Nếu được xác định rằng ma trận chênh lệch được lượng tử hóa có đối xứng 135 độ, bộ xác định xếp chồng 191 chuyển xử lý sang bước S111.

Trong bước S111, bộ xác định xếp chồng 191 loại bỏ phần xếp chồng (dữ liệu xếp chồng) trong ma trận chênh lệch được lượng tử hóa. Sau khi dữ liệu xếp

chồng được loại bỏ, bộ xác định xếp chồng 191 chuyển xử lý sang bước S112.

Ngoài ra, nếu được xác định trong bước S110 rằng ma trận chênh lệch được lượng tử hóa không có đối xứng 135 độ, bộ xác định xếp chồng 191 chuyển xử lý sang bước S112 trong khi bỏ qua xử lý của bước S111 (hoặc không thực hiện xử lý của bước S111).

Trong bước S112, bộ DPCM 192 thực hiện mã hóa DPCM của ma trận chênh lệch mà từ đó phần xếp chồng được loại bỏ, nếu cần.

Trong bước S113, bộ exp-G 193 xác định rằng dữ liệu DPCM được tạo ra trong bước S112 có dấu dương hay âm. Nếu được xác định rằng dấu được bao gồm, bộ exp-G 193 chuyển xử lý sang bước S114.

Trong bước S114, bộ exp-G 193 mã hóa dữ liệu DPCM nhờ sử dụng mã hóa Golomb theo số mũ có dấu. Bộ đầu ra 166 xuất ra các mã Golomb theo số mũ được tạo ra tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21. Khi xử lý của bước S114 được hoàn thành, bộ exp-G 193 chuyển xử lý sang bước S116.

Ngoài ra, nếu được xác định trong bước S113 rằng không có dấu được bao gồm, bộ exp-G 193 chuyển xử lý sang bước S115.

Trong bước S115, bộ exp-G 193 mã hóa dữ liệu DPCM nhờ sử dụng mã hóa Golomb theo số mũ không dấu. Bộ đầu ra 166 xuất ra các mã Golomb theo số mũ được tạo ra tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21. Khi xử lý của bước S115 được hoàn thành, bộ exp-G 193 chuyển xử lý sang bước S116.

Ngoài ra, nếu được xác định trong bước S102 rằng chế độ hiện tại là chế độ sao chép, bộ sao chép 171 sao chép danh sách chia tỷ lệ được truyền trước đó, và sử dụng danh sách chia tỷ lệ được sao chép làm ma trận dự đoán. Bộ đầu ra 166 xuất ra ID danh sách tương ứng với ma trận dự đoán tới bộ mã hóa không tổn hao 16 và bộ giải lượng tử hóa 21 làm thông tin chỉ báo ma trận dự đoán. Sau đó, bộ sao chép 171 chuyển xử lý sang bước S116.

Trong bước S116, bộ khôi phục danh sách chia tỷ lệ 201 khôi phục danh sách chia tỷ lệ. Trong bước S117, bộ lưu trữ 202 lưu trữ danh sách chia tỷ lệ được khôi phục trong bước S116.

Khi xử lý của bước S117 được hoàn thành, bộ xử lý ma trận 150 kết thúc xử lý mã hóa ma trận lượng tử hóa.

2-7. Dòng xử lý của xử lý DPCM

Tiếp theo, ví dụ của dòng xử lý của xử lý DPCM được thực hiện trong bước S112 trên Fig.20 sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.21.

Khi xử lý DPCM được bắt đầu, trong bước S131, bộ mã hóa hệ số DC 211 xác định độ chênh lệch giữa hệ số DC và hằng số. Trong bước S132, bộ DPCM hệ số AC 212 xác định độ chênh lệch giữa hệ số DC và hệ số AC khởi tạo.

Trong bước S133, bộ DPCM hệ số AC 212 xác định rằng tất cả các hệ số AC có được xử lý hay không. Nếu được xác định rằng có hệ số AC không được xử lý, bộ DPCM hệ số AC 212 chuyển xử lý sang bước S134.

Trong bước S134, bộ DPCM hệ số AC 212 dịch chuyển đích xử lý tới hệ số AC tiếp theo. Trong bước S135, bộ DPCM hệ số AC 212 xác định độ chênh lệch giữa hệ số AC được xử lý trước đó và hệ số AC hiện tại đang được xử lý. Khi xử lý của bước S135 được hoàn thành, bộ DPCM hệ số AC 212 đưa xử lý quay lại bước S133.

Theo cách này, miễn là được xác định trong bước S133 rằng có hệ số AC không được xử lý, bộ DPCM hệ số AC 212 thực hiện lặp lại xử lý của các bước S133 tới S135. Nếu được xác định trong bước S133 rằng không có hệ số AC không được xử lý, bộ DPCM hệ số AC 212 kết thúc xử lý DPCM, và đưa xử lý quay lại Fig.20.

Như được mô tả trên đây, độ chênh lệch giữa hệ số DC và hệ số AC nằm tại phần đầu trong số các hệ số AC được xác định, và độ chênh lệch này thay vì hệ số DC được truyền tới thiết bị giải mã ảnh. Do đó, thiết bị mã hóa ảnh 10 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Tiếp theo, cấu trúc ví dụ của thiết bị giải mã ảnh theo phương án của sáng chế sẽ được mô tả.

2-8. Thiết bị giải mã ảnh

Fig.22 là sơ đồ khối minh họa cấu trúc ví dụ của thiết bị giải mã ảnh 300 theo phương án của sáng chế. Thiết bị giải mã ảnh 300 được minh họa trên

Fig.22 là thiết bị xử lý ảnh mà sáng chế được áp dụng cho và có cấu trúc để giải mã dữ liệu được mã hóa được tạo ra bởi thiết bị mã hóa ảnh 10. Việc dẫn tới Fig.22, thiết bị giải mã ảnh 300 bao gồm bộ đệm tích lũy 311, bộ giải mã không tổn hao 312, bộ biến đổi trực giao ngược/giải lượng tử hóa 313, bộ cộng 315, bộ lọc giải khối 316, bộ đệm sắp xếp lại 317, bộ chuyển đổi D/A (dạng số thành dạng tương tự) 318, bộ nhớ khung 319, các bộ lựa chọn 320 và 321, bộ dự đoán trong 330, và bộ bù chuyển động 340.

Bộ đệm tích lũy 311 tích lũy tạm thời dòng được mã hóa được đưa vào thông qua đường truyền, nhờ sử dụng vật lưu trữ.

Bộ giải mã không tổn hao 312 giải mã dòng được mã hóa được đưa vào từ bộ đệm tích lũy 311 theo phương pháp mã hóa được sử dụng để mã hóa. Bộ giải mã không tổn hao 312 còn giải mã thông tin được ghép kênh trong vùng đoạn đầu của dòng được mã hóa. Thông tin được ghép kênh trong vùng đoạn đầu của dòng được mã hóa có thể bao gồm, ví dụ, thông tin để tạo ra danh sách chia tỷ lệ được mô tả trên đây, và thông tin liên quan đến dự đoán trong và thông tin liên quan đến dự đoán liên đới, mà được chứa trong đoạn đầu khối. Bộ giải mã không tổn hao 312 xuất ra dữ liệu được lượng tử hóa được giải mã và thông tin để tạo ra danh sách chia tỷ lệ tới bộ biến đổi trực giao ngược/giải lượng tử hóa 313. Bộ giải mã không tổn hao 312 còn xuất ra thông tin liên quan đến dự đoán trong tới bộ dự đoán trong 330. Bộ giải mã không tổn hao 312 còn xuất ra thông tin liên quan đến dự đoán liên đới tới bộ bù chuyển động 340.

Bộ biến đổi trực giao ngược/giải lượng tử hóa 313 thực hiện giải lượng tử hóa và biến đổi trực giao ngược trên dữ liệu được lượng tử hóa được đưa vào từ bộ giải mã không tổn hao 312 để tạo ra dữ liệu sai số dự đoán. Sau đó, bộ biến đổi trực giao ngược/giải lượng tử hóa 313 xuất ra dữ liệu sai số dự đoán được tạo ra tới bộ cộng 315.

Bộ cộng 315 cộng với nhau dữ liệu sai số dự đoán được đưa vào từ bộ biến đổi trực giao ngược/giải lượng tử hóa 313 và dữ liệu ảnh dự đoán được đưa vào từ bộ lựa chọn 321 để tạo ra dữ liệu ảnh được giải mã. Sau đó, bộ cộng 315 xuất ra dữ liệu ảnh được giải mã được tạo ra tới bộ lọc giải khối 316 và bộ nhớ khung 319.

Bộ lọc giải khối 316 lọc dữ liệu ảnh được giải mã được đưa vào từ bộ cộng 315 để loại bỏ các nhiễu khối, và xuất ra dữ liệu ảnh được giải mã được lọc tới bộ đệm sắp xếp lại 317 và bộ nhớ khung 319.

Bộ đệm sắp xếp lại 317 sắp xếp lại các ảnh được đưa vào từ bộ lọc giải khối 316 để tạo ra chuỗi dữ liệu ảnh theo thời gian. Sau đó, bộ đệm sắp xếp lại 317 xuất ra dữ liệu ảnh được tạo ra tới bộ chuyển đổi D/A 318.

Bộ chuyển đổi D/A 318 chuyển đổi dữ liệu ảnh trong dạng số mà được đưa vào từ bộ đệm sắp xếp lại 317 thành tín hiệu ảnh trong dạng tương tự. Sau đó, bộ chuyển đổi D/A 318 xuất ra tín hiệu ảnh tương tự tới, ví dụ, màn hình (không được thể hiện) được kết nối tới thiết bị giải mã ảnh 300 để hiển thị ảnh.

Bộ nhớ khung 319 lưu trữ dữ liệu ảnh được giải mã cần được lọc, mà được đưa vào từ bộ cộng 315, và dữ liệu ảnh được giải mã được lọc được đưa vào từ bộ lọc giải khối 316, nhờ sử dụng vật lưu trữ.

Bộ lựa chọn 320 chuyển đổi đích mà dữ liệu ảnh được cấp từ bộ nhớ khung 319 cần được xuất ra tới đó giữa bộ dự đoán trong 330 và bộ bù chuyển động 340, đối với mỗi khối trong ảnh, theo thông tin chế độ thu được bởi bộ giải mã không tổn hao 312. Ví dụ, nếu chế độ dự đoán trong được chỉ rõ, bộ lựa chọn 320 xuất ra dữ liệu ảnh được giải mã cần được lọc, mà được cấp từ bộ nhớ khung 319, tới bộ dự đoán trong 330 làm dữ liệu ảnh tham chiếu. Ngoài ra, nếu chế độ dự đoán liên đới được chỉ rõ, bộ lựa chọn 320 xuất ra dữ liệu ảnh được giải mã cần được lọc được cấp từ bộ nhớ khung 319 tới bộ bù chuyển động 340 như là dữ liệu ảnh tham chiếu.

Bộ lựa chọn 321 chuyển đổi nguồn mà từ đó dữ liệu ảnh dự đoán cần được cấp tới bộ cộng 315 được xuất ra từ đó giữa bộ dự đoán trong 330 và bộ bù chuyển động 340, đối với mỗi khối trong ảnh, theo thông tin chế độ thu được bởi bộ giải mã không tổn hao 312. Ví dụ, nếu chế độ dự đoán trong được chỉ rõ, bộ lựa chọn 321 cấp dữ liệu ảnh dự đoán được xuất ra từ bộ dự đoán trong 330 tới bộ cộng 315. Nếu chế độ dự đoán liên đới được chỉ rõ, bộ lựa chọn 321 cấp dữ liệu ảnh dự đoán được xuất ra từ bộ bù chuyển động 340 tới bộ cộng 315.

Bộ dự đoán trong 330 thực hiện dự đoán trong màn của giá trị điểm ảnh dựa trên thông tin liên quan đến dự đoán trong, mà được đưa vào từ bộ giải mã

không tổn hao 312, và dữ liệu ảnh tham chiếu được cấp từ bộ nhớ khung 319, và tạo ra dữ liệu ảnh dự đoán. Sau đó, bộ dự đoán trong 330 xuất ra dữ liệu ảnh dự đoán được tạo ra tới bộ lựa chọn 321.

Bộ bù chuyển động 340 thực hiện xử lý bù chuyển động dựa trên thông tin liên quan đến dự đoán liên đới, mà được đưa vào từ bộ giải mã không tổn hao 312, và dữ liệu ảnh tham chiếu được cấp từ bộ nhớ khung 319, và tạo ra dữ liệu ảnh dự đoán. Sau đó, bộ bù chuyển động 340 xuất ra dữ liệu ảnh dự đoán được tạo ra tới bộ lựa chọn 321.

2-9. Cấu trúc ví dụ của của bộ biến đổi trực giao ngược/giải lượng tử hóa

Fig.23 là sơ đồ khối minh họa ví dụ về cấu trúc chính của bộ biến đổi trực giao ngược/giải lượng tử hóa 313 của thiết bị giải mã ảnh 300 được minh họa trên Fig.22. Viền dẫn tới Fig.23, bộ biến đổi trực giao ngược/giải lượng tử hóa 313 bao gồm bộ tạo ma trận 410, bộ lựa chọn 430, bộ giải lượng tử hóa 440, và bộ biến đổi trực giao ngược 450.

(1) Bộ tạo ma trận

Bộ tạo ma trận 410 giải mã dữ liệu danh sách chia tỷ lệ được mã hóa mà được tách từ dòng bit và được cấp bởi bộ giải mã không tổn hao 312, và tạo ra danh sách chia tỷ lệ. Bộ tạo ma trận 410 cấp danh sách chia tỷ lệ được tạo ra tới bộ giải lượng tử hóa 440.

(2) Bộ lựa chọn

Bộ lựa chọn 430 lựa chọn đơn vị biến đổi (TU) cần được sử dụng cho biến đổi trực giao ngược của dữ liệu ảnh cần được giải mã từ trong số các đơn vị biến đổi có các kích cỡ khác nhau. Các ví dụ của các kích cỡ có thể của các đơn vị biến đổi có thể được lựa chọn bởi bộ lựa chọn 430 bao gồm 4x4 và 8x8 đối với H.264/AVC, và bao gồm 4x4, 8x8, 16x16, và 32x32 đối với HEVC. Bộ lựa chọn 430 có thể lựa chọn đơn vị biến đổi theo, ví dụ, LCU, SCU, và split_flag được chứa trong đoạn đầu của dòng được mã hóa. Sau đó, bộ lựa chọn 430 xuất ra thông tin chỉ rõ kích cỡ của đơn vị biến đổi được lựa chọn tới bộ giải lượng tử hóa 440 và bộ biến đổi trực giao ngược 450.

(3) Bộ giải lượng tử hóa

Bộ giải lượng tử hóa 440 giải lượng tử hóa dữ liệu hệ số biến đổi được

lượng tử hóa khi các ảnh được mã hóa, bằng cách sử dụng danh sách chia tỷ lệ của đơn vị biến đổi được lựa chọn bởi bộ lựa chọn 430. Sau đó, bộ giải lượng tử hóa 440 xuất ra dữ liệu hệ số biến đổi được giải lượng tử hóa tới bộ biến đổi trực giao ngược 450.

(4) Bộ biến đổi trực giao ngược

Bộ biến đổi trực giao ngược 450 thực hiện biến đổi trực giao ngược trên dữ liệu hệ số biến đổi được giải lượng tử hóa bởi bộ giải lượng tử hóa 440 trong các đơn vị của đơn vị biến đổi được lựa chọn theo phương pháp biến đổi trực giao được sử dụng cho việc mã hóa để tạo ra dữ liệu sai số dự đoán. Sau đó, bộ biến đổi trực giao ngược 450 xuất ra dữ liệu sai số dự đoán được tạo ra tới bộ cộng 315.

2-10. Cấu trúc ví dụ chi tiết của bộ tạo ma trận

Fig.24 là sơ đồ khối minh họa ví dụ của cấu trúc chi tiết của bộ tạo ma trận 410 được minh họa trên Fig.23. Viền dẫn tới Fig.24, bộ tạo ma trận 410 bao gồm bộ phân tích tham số 531, bộ dự đoán 532, bộ giải mã entropy 533, bộ khôi phục danh sách chia tỷ lệ 534, bộ đầu ra 535, và bộ lưu trữ 536.

(1) Bộ phân tích tham số

Bộ phân tích tham số 531 phân tích các cờ và các tham số khác nhau liên quan đến danh sách chia tỷ lệ, mà được cấp từ bộ giải mã không tổn hao 312. Ngoài ra, theo các kết quả phân tích, bộ phân tích tham số 531 cấp các dạng thông tin khác nhau được cấp từ bộ giải mã không tổn hao 312, như dữ liệu được mã hóa của ma trận chênh lệch, tới bộ dự đoán 532 hoặc bộ giải mã entropy 533.

Ví dụ, nếu `pred_mode` bằng 0, bộ phân tích tham số 531 xác định rằng chế độ hiện tại là chế độ sao chép, và cấp `pred_matrix_id_delta` tới bộ sao chép 541. Ngoài ra, ví dụ, nếu `pred_mode` bằng 1, bộ phân tích tham số 531 xác định rằng chế độ hiện tại là chế độ quét hoàn toàn (chế độ thông thường), và cấp `pred_matrix_id_delta` và `pred_size_id_delta` tới bộ tạo ma trận dự đoán 542.

Ngoài ra, ví dụ, nếu `residual_flag` là đúng, bộ phân tích tham số 531 cấp dữ liệu được mã hóa (các mã Golomb theo số mũ) của danh sách chia tỷ lệ được cấp từ bộ giải mã không tổn hao 312 tới bộ exp-G 551 của bộ giải mã entropy

533. Bộ phân tích tham số 531 còn cấp `residual_symmetry_flag` tới bộ exp-G 551.

Ngoài ra, bộ phân tích tham số 531 cấp `residual_down_sampling_flag` tới bộ biến đổi kích cỡ ma trận chèn lệch 562 của bộ khôi phục danh sách chia tỷ lệ 534.

(2) Bộ dự đoán

Bộ dự đoán 532 tạo ra ma trận dự đoán theo sự điều khiển của bộ phân tích tham số 531. Như được minh họa trên Fig.24, bộ dự đoán 532 bao gồm bộ sao chép 541 và bộ tạo ma trận dự đoán 542.

Trong chế độ sao chép, bộ sao chép 541 sao chép danh sách chia tỷ lệ được truyền trước đó, và sử dụng danh sách chia tỷ lệ được sao chép làm ma trận dự đoán. Cụ thể hơn, bộ sao chép 541 đọc danh sách chia tỷ lệ được truyền trước đó tương ứng với `pred_matrix_id_delta` và có cùng kích cỡ như danh sách chia tỷ lệ đối với vùng hiện tại từ bộ lưu trữ 536, sử dụng danh sách chia tỷ lệ được đọc như là ảnh dự đoán, và cấp ảnh dự đoán tới bộ đầu ra 535.

Trong chế độ thông thường, bộ tạo ma trận dự đoán 542 tạo ra (hoặc dự đoán) ma trận dự đoán nhờ sử dụng danh sách chia tỷ lệ được truyền trước đó. Cụ thể hơn, bộ tạo ma trận dự đoán 542 đọc danh sách chia tỷ lệ được truyền trước đó tương ứng với `pred_matrix_id_delta` và `pred_size_id_delta` từ bộ lưu trữ 536, và tạo ra ma trận dự đoán nhờ sử dụng danh sách chia tỷ lệ được đọc. Nói cách khác, bộ tạo ma trận dự đoán 542 tạo ra ma trận dự đoán tương tự ma trận dự đoán được tạo ra bởi bộ tạo ma trận dự đoán 172 (Fig.16) của thiết bị mã hóa ảnh 10. Bộ tạo ma trận dự đoán 542 cấp ma trận dự đoán được tạo ra tới bộ biến đổi kích cỡ ma trận dự đoán 561 của bộ khôi phục danh sách chia tỷ lệ 534.

(3) Bộ giải mã entropy

Bộ giải mã entropy 533 khôi phục ma trận chèn lệch từ các mã Golomb theo số mũ được cấp từ bộ phân tích tham số 531. Như được minh họa trên Fig.24, bộ giải mã entropy 533 bao gồm bộ exp-G 551, bộ DPCM ngược 552, và bộ xác định xếp chồng ngược 553.

Bộ exp-G 551 giải mã các mã Golomb theo số mũ có dấu hoặc không dấu (sau đây cũng được gọi là giải mã Golomb theo số mũ) để khôi phục dữ liệu

DPCM. Bộ exp-G 551 cấp dữ liệu DPCM được khôi phục với nhau với residual_symmetry_flag tới bộ DPCM ngược 552.

Bộ DPCM ngược 552 thực hiện giải mã DPCM của dữ liệu mà từ đó phần xếp chồng được loại bỏ để tạo ra dữ liệu dư từ dữ liệu DPCM. Bộ DPCM ngược 552 cấp dữ liệu dư được tạo ra cùng với residual_symmetry_flag tới bộ xác định xếp chồng ngược 553.

Nếu residual_symmetry_flag là đúng, tức là, nếu dữ liệu dư là phần còn lại của ma trận đối xứng 135 độ mà từ đó dữ liệu (các thành phần ma trận) của phần đối xứng xếp chồng được loại bỏ, bộ xác định xếp chồng ngược 553 khôi phục dữ liệu của phần đối xứng. Nói cách khác, ma trận chệch lệch của ma trận đối xứng 135 độ được khôi phục. Lưu ý rằng nếu residual_symmetry_flag không phải là đúng, tức là, nếu dữ liệu dư biểu diễn ma trận mà không phải là ma trận đối xứng 135 độ, bộ xác định xếp chồng ngược 553 sử dụng dữ liệu dư như là ma trận chệch lệch mà không khôi phục dữ liệu của phần đối xứng. Bộ xác định xếp chồng ngược 553 cấp ma trận chệch lệch được khôi phục theo cách được mô tả trên đây tới bộ khôi phục danh sách chia tỷ lệ 534 (bộ biến đổi kích cỡ ma trận chệch lệch 562).

(4) Bộ khôi phục danh sách chia tỷ lệ

Bộ khôi phục danh sách chia tỷ lệ 534 khôi phục danh sách chia tỷ lệ. Như được minh họa trên Fig.24, bộ khôi phục danh sách chia tỷ lệ 534 bao gồm bộ biến đổi kích cỡ ma trận dự đoán 561, bộ biến đổi kích cỡ ma trận chệch lệch 562, bộ giải lượng tử hóa 563, và bộ tính toán 564.

Nếu kích cỡ của ma trận dự đoán được cấp từ bộ dự đoán 532 (bộ tạo ma trận dự đoán 542) là khác với kích cỡ của danh sách chia tỷ lệ đối với vùng hiện tại cần được khôi phục, bộ biến đổi kích cỡ ma trận dự đoán 561 chuyển đổi kích cỡ của ma trận dự đoán.

Ví dụ, nếu kích cỡ của ma trận dự đoán lớn hơn kích cỡ của danh sách chia tỷ lệ, bộ biến đổi kích cỡ ma trận dự đoán 561 chuyển đổi xuống ma trận dự đoán. Ngoài ra, ví dụ, nếu kích cỡ của ma trận dự đoán nhỏ hơn so với kích cỡ của danh sách chia tỷ lệ, bộ biến đổi kích cỡ ma trận dự đoán 561 chuyển đổi lên ma trận dự đoán. Phương pháp tương tự như phương pháp dùng cho bộ biến đổi

kích cỡ ma trận dự đoán 181 (Fig.16) của thiết bị mã hóa ảnh 10 được lựa chọn như là phương pháp chuyển đổi.

Bộ biến đổi kích cỡ ma trận dự đoán 561 cấp ma trận dự đoán ma kích cỡ của nó được làm cho khớp với kích cỡ của danh sách chia tỷ lệ tới bộ tính toán 564.

Nếu `residual_down_sampling_flag` là đúng, tức là, nếu kích cỡ của ma trận chênh lệch được truyền nhỏ hơn so với kích cỡ của vùng hiện tại cần được giải lượng tử hóa, bộ biến đổi kích cỡ ma trận chênh lệch 562 chuyển đổi lên ma trận chênh lệch để tăng kích cỡ của ma trận chênh lệch thành kích cỡ tương ứng với vùng hiện tại cần được giải lượng tử hóa. Phương pháp chuyển đổi lên bất kỳ có thể được sử dụng. Ví dụ, phương pháp tương ứng với phương pháp chuyển đổi xuống được thực hiện bởi bộ biến đổi kích cỡ ma trận chênh lệch 163 (Fig.16) của thiết bị mã hóa ảnh 10 có thể được sử dụng.

Ví dụ, nếu bộ biến đổi kích cỡ ma trận chênh lệch 163 lấy mẫu xuống ma trận chênh lệch, bộ biến đổi kích cỡ ma trận chênh lệch 562 có thể lấy mẫu lên ma trận chênh lệch. Ngoài ra, nếu bộ biến đổi kích cỡ ma trận chênh lệch 163 lấy mẫu con ma trận chênh lệch, bộ biến đổi kích cỡ ma trận chênh lệch 562 có thể thực hiện lấy mẫu con ngược của ma trận chênh lệch.

Ví dụ, bộ biến đổi kích cỡ ma trận chênh lệch 562 có thể thực hiện xử lý nội suy lân cận gần nhất (lân cận gần nhất) như được minh họa trên Fig.25 ngoài nội suy tuyến tính chung. Xử lý nội suy lân cận gần nhất có thể làm giảm dung lượng bộ nhớ.

Do đó, ngay cả nếu danh sách chia tỷ lệ có kích cỡ lớn không được truyền, dữ liệu thu được sau khi lấy mẫu con không cần phải được lưu trữ để lấy mẫu lên từ danh sách chia tỷ lệ có kích cỡ nhỏ. Ngoài ra, bộ đệm trung gian hoặc loại tương tự là không cần thiết khi dữ liệu được bao gồm trong tính toán trong khi lấy mẫu lên được lưu trữ.

Lưu ý rằng nếu `residual_down_sampling_flag` không phải là đúng, tức là, nếu ma trận chênh lệch được truyền có cùng kích cỡ như kích cỡ khi được sử dụng cho xử lý lượng tử hóa, bộ biến đổi kích cỡ ma trận chênh lệch 562 bỏ qua việc chuyển đổi lên của ma trận chênh lệch (hoặc có thể chuyển đổi lên ma trận

chênh lệch bởi hệ số là 1).

Bộ biến đổi kích cỡ ma trận chênh lệch 562 cấp ma trận chênh lệch được chuyển đổi lên theo cách được mô tả trên đây, nếu cần, tới bộ giải lượng tử hóa 563.

Bộ giải lượng tử hóa 563 giải lượng tử hóa ma trận chênh lệch được cấp (dữ liệu được lượng tử hóa) nhờ sử dụng phương pháp tương ứng với phương pháp để lượng tử hóa được thực hiện bởi bộ lượng tử hóa 183 (Fig.16) của thiết bị mã hóa ảnh 10, và cấp ma trận chênh lệch được giải lượng tử hóa tới bộ tính toán 564. Lưu ý rằng nếu bộ lượng tử hóa 183 được bỏ qua, tức là, nếu ma trận chênh lệch được cấp từ bộ biến đổi kích cỡ ma trận chênh lệch 562 không phải dữ liệu được lượng tử hóa, bộ giải lượng tử hóa 563 có thể được bỏ qua.

Bộ tính toán 564 cộng với nhau ma trận dự đoán được cấp từ bộ biến đổi kích cỡ ma trận dự đoán 561 và ma trận chênh lệch được cấp từ bộ giải lượng tử hóa 563, và khôi phục danh sách chia tỷ lệ đối với vùng hiện tại. Bộ tính toán 564 cấp danh sách chia tỷ lệ được khôi phục tới bộ đầu ra 535 và bộ lưu trữ 536.

(5) Bộ đầu ra

Bộ đầu ra 535 xuất ra thông tin được cấp tới thiết bị phía ngoài bộ tạo ma trận 410. Ví dụ, trong chế độ sao chép, bộ đầu ra 535 cấp ma trận dự đoán được cấp từ bộ sao chép 541 tới bộ giải lượng tử hóa 440 như là danh sách chia tỷ lệ đối với vùng hiện tại. Ngoài ra, ví dụ, trong chế độ thông thường, bộ đầu ra 535 cấp danh sách chia tỷ lệ đối với vùng hiện tại được cấp từ bộ khôi phục danh sách chia tỷ lệ 534 (bộ tính toán 564) tới bộ giải lượng tử hóa 440.

(6) Bộ lưu trữ

Bộ lưu trữ 536 lưu trữ danh sách chia tỷ lệ được cấp từ bộ khôi phục danh sách chia tỷ lệ 534 (bộ tính toán 564) cùng với kích cỡ và ID danh sách của danh sách chia tỷ lệ. Thông tin liên quan đến danh sách chia tỷ lệ được lưu trữ trong bộ lưu trữ 536 được sử dụng để tạo ra các ma trận dự đoán của các đơn vị biến đổi trực giao khác mà được xử lý sau đó. Nói cách khác, bộ lưu trữ 536 cấp thông tin liên quan đến danh sách chia tỷ lệ được lưu trữ tới bộ dự đoán 532 như là thông tin liên quan đến danh sách chia tỷ lệ được truyền trước đó.

2-11. Cấu trúc ví dụ chi tiết của bộ DPCM ngược

Fig.26 là sơ đồ khối minh họa ví dụ của cấu trúc chi tiết của bộ DPCM ngược 552 được minh họa trên Fig.24. Viện dẫn tới Fig.26, bộ DPCM ngược 552 bao gồm bộ thiết lập khởi tạo 571, bộ giải mã DPCM 572, và bộ tách hệ số DC 573.

Bộ thiết lập khởi tạo 571 thu được sizeID và MatrixID, và thiết lập các biến khác nhau thành các giá trị khởi tạo. Bộ thiết lập khởi tạo 571 cấp thông tin được thiết lập và thu được tới bộ giải mã DPCM 572.

Bộ giải mã DPCM 572 xác định các hệ số riêng biệt (hệ số DC và các hệ số AC) từ các giá trị chênh lệch (scaling_list_delta_coef) của hệ số DC và các hệ số AC sử dụng các thiết lập khởi tạo và loại tương tự được cấp từ bộ thiết lập khởi tạo 571. Bộ giải mã DPCM 572 cấp các hệ số được xác định tới bộ tách hệ số DC 573 (ScalingList[i]).

Bộ tách hệ số DC 573 tách hệ số DC từ trong số các hệ số (ScalingList[i]) được cấp từ bộ giải mã DPCM 572. Hệ số DC nằm tại phần đầu của các hệ số AC. Tức là, hệ số khởi tạo (ScalingList[0]) trong số các hệ số được cấp từ bộ giải mã DPCM 572 là hệ số DC. Bộ tách hệ số DC 573 tách hệ số nằm tại phần đầu như là hệ số DC, và xuất ra hệ số được tách tới bộ xác định xếp chồng ngược 553 (DC_coef). Bộ tách hệ số DC 573 xuất ra các hệ số khác (ScalingList[i] ($i > 0$)) tới bộ xác định xếp chồng ngược 553 làm các hệ số AC.

Do đó, bộ DPCM ngược 552 có thể thực hiện việc giải mã DPCM chính xác, và có thể thu được hệ số DC và các hệ số AC. Tức là, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

2-12. Dòng xử lý của xử lý giải mã ma trận lượng tử hóa

Ví dụ của dòng xử lý của xử lý giải mã ma trận lượng tử hóa được thực hiện bởi bộ tạo ma trận 410 có cấu trúc nêu trên sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.27.

Khi xử lý giải mã ma trận lượng tử hóa được bắt đầu, trong bước S301, bộ phân tích tham số 531 đọc các giá trị được lượng tử hóa (Qscale0 tới Qscale3) của các vùng 0 tới 3.

Trong bước S302, bộ phân tích tham số 531 đọc pred_mode. Trong bước S303, bộ phân tích tham số 531 xác định rằng pred_mode có bằng 0 hay không.

Nếu được xác định rằng `pred_mode` bằng 0, bộ phân tích tham số 531 xác định rằng chế độ hiện tại là chế độ sao chép, và chuyển xử lý sang bước S304.

Trong bước S304, bộ phân tích tham số 531 đọc `pred_matrix_id_delta`. Trong bước S305, bộ sao chép 541 sao chép danh sách chia tỷ lệ mà đã được truyền, và sử dụng danh sách chia tỷ lệ được sao chép làm ma trận dự đoán. Trong chế độ sao chép, ma trận dự đoán được xuất ra như là danh sách chia tỷ lệ đối với vùng hiện tại. Khi xử lý của bước S305 được hoàn thành, bộ sao chép 541 kết thúc xử lý giải mã ma trận lượng tử hóa.

Ngoài ra, nếu được xác định trong bước S303 rằng `pred_mode` không bằng 0, bộ phân tích tham số 531 xác định rằng chế độ hiện tại là chế độ quét toàn phần (chế độ thông thường), và chuyển xử lý sang bước S306.

Trong bước S306, bộ phân tích tham số 531 đọc `pred_matrix_id_delta`, `pred_size_id_delta`, và `residual_flag`. Trong bước S307, bộ tạo ma trận dự đoán 542 tạo ra ma trận dự đoán từ danh sách chia tỷ lệ mà đã được truyền.

Trong bước S308, bộ phân tích tham số 531 xác định rằng `residual_flag` có phải là đúng hay không. Nếu được xác định rằng `residual_flag` không phải là đúng, không có ma trận dư tồn tại, và ma trận dự đoán được tạo ra trong bước S307 được xuất ra như là danh sách chia tỷ lệ đối với vùng hiện tại. Trong trường hợp này, do đó, bộ phân tích tham số 531 kết thúc xử lý giải mã ma trận lượng tử hóa.

Ngoài ra, nếu được xác định trong bước S308 rằng `residual_flag` là đúng, bộ phân tích tham số 531 chuyển xử lý sang bước S309.

Trong bước S309, bộ phân tích tham số 531 đọc `residual_down_sampling_flag` và `residual_symmetry_flag`.

Trong bước S310, bộ exp-G 551 và bộ DPCM ngược 552 giải mã các mã Golomb theo số mũ của ma trận dư, và tạo ra dữ liệu dư.

Trong bước S311, bộ xác định xếp chồng ngược 553 xác định rằng `residual_symmetry_flag` có phải là đúng hay không. Nếu được xác định rằng `residual_symmetry_flag` là đúng, bộ xác định xếp chồng ngược 553 chuyển xử lý sang bước S312, và khôi phục phần xếp chồng bị loại bỏ của dữ liệu dư (hoặc thực hiện xử lý đối xứng ngược). Khi ma trận chênh lệch mà là ma trận đối xứng

135 độ được tạo ra theo cách được mô tả trên đây, bộ xác định xếp chồng ngược 553 chuyển xử lý sang bước S313.

Ngoài ra, nếu được xác định trong bước S311 rằng `residual_symmetry_flag` không phải là đúng (hoặc nếu dữ liệu dư là ma trận chênh lệch mà không phải là ma trận đối xứng 135 độ), bộ xác định xếp chồng ngược 553 chuyển xử lý sang bước S313 trong khi bỏ qua xử lý của bước S312 (hoặc không thực hiện xử lý đối xứng ngược).

Trong bước S313, bộ biến đổi kích cỡ ma trận chênh lệch 562 xác định rằng `residual_down_sampling_flag` có phải là đúng hay không. Nếu được xác định rằng `residual_down_sampling_flag` là đúng, bộ biến đổi kích cỡ ma trận chênh lệch 562 chuyển xử lý sang bước S314, và chuyển đổi lên ma trận chênh lệch thành kích cỡ tương ứng với vùng hiện tại cần được giải lượng tử hóa. Sau khi ma trận chênh lệch được chuyển đổi lên, bộ biến đổi kích cỡ ma trận chênh lệch 562 chuyển xử lý sang bước S315.

Ngoài ra, nếu được xác định trong bước S313 rằng `residual_down_sampling_flag` không phải là đúng, bộ biến đổi kích cỡ ma trận chênh lệch 562 chuyển xử lý sang bước S315 trong khi bỏ qua xử lý của bước S314 (hoặc không chuyển đổi lên ma trận chênh lệch).

Trong bước S315, bộ tính toán 564 cộng ma trận chênh lệch vào ma trận dự đoán để tạo ra danh sách chia tỷ lệ đối với vùng hiện tại. Khi xử lý của bước S315 được hoàn thành, xử lý giải mã ma trận lượng tử hóa kết thúc.

2-13. Dòng xử lý của xử lý giải mã tín hiệu dư

Tiếp theo, ví dụ của dòng xử lý của xử lý giải mã tín hiệu dư được thực hiện trong bước S310 trên Fig.27 sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.28.

Khi xử lý giải mã tín hiệu dư được bắt đầu, trong bước S331, bộ exp-G 551 giải mã các mã Golomb theo số mũ được cấp.

Trong bước S332, bộ DPCM ngược 552 thực hiện xử lý DPCM ngược trên dữ liệu DPCM thu được bởi bộ exp-G 551 thông qua giải mã.

Khi xử lý DPCM ngược được hoàn thành, bộ DPCM ngược 552 kết thúc xử lý giải mã tín hiệu dư, và đưa xử lý quay lại Fig.27.

2-14. Dòng xử lý của xử lý DPCM ngược

Tiếp theo, ví dụ của dòng xử lý của xử lý DPCM ngược được thực hiện trong bước S332 trên Fig.28 sẽ được mô tả có việ dẫn tới lưu đồ được minh họa trên Fig.29.

Khi xử lý DPCM ngược được bắt đầu, trong bước S351, bộ thiết lập khởi tạo 571 thu được sizeID và MatrixID.

Trong bước S352, bộ thiết lập khởi tạo 571 thiết lập coefNum như sau.

$$\text{coefNum} = \min((1 \ll (4 + (\text{sizeID} \ll 1))), 65)$$

Trong bước S353, bộ thiết lập khởi tạo 571 thiết lập biến i và biến nextcoef như sau.

$$i = 0$$

$$\text{nextcoef} = 8$$

Trong bước S354, bộ giải mã DPCM 572 xác định rằng biến i < coefNum hay không. Nếu biến i nhỏ hơn so với coefNum, bộ thiết lập khởi tạo 571 chuyển xử lý sang bước S355.

Trong bước S355, bộ giải mã DPCM 572 đọc dữ liệu DPCM của hệ số (scaling_list_delta_coef).

Trong bước S356, bộ giải mã DPCM 572 xác định nextcoef như dưới đây nhờ sử dụng dữ liệu DPCM được đọc, và còn xác định scalingList[i].

$$\text{nextcoef} = (\text{nextcoef} + \text{scaling_list_delta_coef} + 256) \% 256$$

$$\text{scalingList}[i] = \text{nextcoef}$$

Trong bước S357, bộ tách hệ số DC 573 xác định rằng sizeID có lớn hơn 1 hay không và xác định rằng biến i có bằng 0 hay không (tức là, hệ số nằm tại phần đầu). Nếu được xác định rằng sizeID lớn hơn 1 và biến i biểu diễn hệ số nằm tại phần đầu, bộ tách hệ số DC 573 chuyển xử lý sang bước S358, và sử dụng hệ số này làm hệ số DC (DC_coef = nextcoef). Khi xử lý của bước S358 được hoàn thành, bộ tách hệ số DC 573 chuyển xử lý sang bước S360.

Ngoài ra, nếu được xác định trong bước S357 rằng sizeID nhỏ hơn hoặc bằng 1 hoặc biến i không biểu diễn hệ số nằm tại phần đầu, bộ tách hệ số DC 573 chuyển xử lý sang bước S359, và dịch chuyển biến i đối với mỗi hệ số bởi một do hệ số DC đã được tách. (ScalingList[(i-(sizeID)>1)?1;0] = nextcoef) Nếu

xử lý của bước S359 được hoàn thành, bộ tách hệ số DC 573 chuyển xử lý sang bước S360.

Trong bước S360, bộ giải mã DPCM 572 gia tăng biến i để thay đổi đích xử lý thành hệ số tiếp theo, và sau đó đưa xử lý quay lại bước S354.

Trong bước S354, xử lý của các bước S354 tới S360 được thực hiện lặp lại cho đến khi được xác định rằng biến i là lớn hơn hoặc bằng `coefNum`. Nếu được xác định trong bước S354 rằng biến i là lớn hơn hoặc bằng `coefNum`, bộ giải mã DPCM 572 kết thúc xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Do đó, độ chênh lệch giữa hệ số DC và hệ số AC nằm tại phần đầu của các hệ số AC có thể được giải mã chính xác. Do đó, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

3. Phương án thứ ba

3-1. Cú pháp: phương pháp thứ hai

Phương pháp truyền khác độ chênh lệch giữa hệ số DC và hệ số khác, thay vì hệ số DC, có thể, ví dụ, truyền độ chênh lệch giữa hệ số DC và thành phần (0, 0) của ma trận 8×8 như là dữ liệu DPCM khác với dữ liệu DPCM của ma trận 8×8 (phương pháp thứ hai). Ví dụ, sau khi truyền DPCM của ma trận 8×8 , độ chênh lệch giữa hệ số DC và thành phần (0, 0) của ma trận 8×8 có thể được truyền.

Do đó, tương tự phương pháp thứ nhất, tỷ số nén có thể được cải thiện hơn nữa khi giá trị của hệ số (0, 0) (hệ số AC) của ma trận 8×8 và giá trị của hệ số DC là gần nhau.

Fig.30 minh họa cú pháp của danh sách chia tỷ lệ trong phương pháp thứ hai. Trong ví dụ được minh họa trên Fig.30, 64 giá trị chênh lệch (`scaling_list_delta_coef`) giữa các hệ số được đọc. Cuối cùng, độ chênh lệch (`scaling_list_dc_coef_delta`) giữa hệ số DC và hệ số (0, 0) (hệ số AC) được đọc, và hệ số DC được xác định từ độ chênh lệch.

Trong phương pháp thứ hai, do đó, cú pháp để giải mã các hệ số AC có thể giống như cú pháp của kỹ thuật đã biết được minh họa trên Fig.12. Tức là, cú pháp đối với phương pháp thứ hai có thể thu được bằng cách cải biến ví dụ của kỹ thuật đã biết bởi lượng nhỏ, và có thể khả thi hơn so với cú pháp dùng cho

phương pháp thứ nhất.

Tuy nhiên, trong khi phương pháp thứ hai không cho phép thiết bị giải mã ảnh thu được hệ số DC cho đến khi thiết bị giải mã ảnh thu được tất cả các hệ số và giải nén tất cả dữ liệu DPCM, phương pháp thứ nhất cho phép thiết bị giải mã ảnh khôi phục hệ số DC tại thời điểm khi thiết bị giải mã ảnh thu hệ số khởi tạo.

Thiết bị mã hóa ảnh mà thực hiện cú pháp đối với phương pháp thứ hai được mô tả trên đây sẽ được mô tả sau đây.

3-2. Cấu trúc ví dụ chi tiết của bộ DPCM

Trong phương pháp thứ hai, thiết bị mã hóa ảnh 10 có cấu trúc về cơ bản tương tự như cấu trúc trong phương pháp thứ nhất được mô tả trên đây. Cụ thể, thiết bị mã hóa ảnh 10 có cấu trúc như trong ví dụ được minh họa trên Fig.14. Ngoài ra, bộ lượng tử hóa/biến đổi trực giao 14 có cấu trúc như trong ví dụ được minh họa trên Fig.15. Ngoài ra, bộ xử lý ma trận 150 có cấu trúc như trong ví dụ được minh họa trên Fig.16.

Cấu trúc ví dụ của bộ DPCM 192 trong ví dụ thứ hai được minh họa trên Fig.31. Như được minh họa trên Fig.31, trong ví dụ thứ hai, bộ DPCM 192 bao gồm bộ đệm hệ số AC 611, bộ mã hóa hệ số AC 612, bộ DPCM hệ số AC 613, và bộ DPCM hệ số DC 614.

Bộ đệm hệ số AC 611 lưu trữ hệ số AC khởi tạo (tức là, hệ số $(0, 0)$) được cấp từ bộ xác định xếp chồng 191. Bộ đệm hệ số AC 611 cấp hệ số AC khởi tạo được lưu trữ (hệ số AC $(0, 0)$) tới bộ DPCM hệ số DC 614 tại thời điểm định trước sau khi tất cả các hệ số AC được đưa vào xử lý DPCM, hoặc phản hồi lại yêu cầu.

Bộ mã hóa hệ số AC 612 thu được hệ số AC khởi tạo (hệ số AC $(0, 0)$) được cấp từ bộ xác định xếp chồng 191, và trừ giá trị của hệ số AC khởi tạo từ hằng số (ví dụ, 8). Bộ mã hóa hệ số AC 612 cấp kết quả trừ (độ chênh lệch) tới bộ exp-G 193 như là hệ số khởi tạo ($\text{scaling_list_delta_coef}(i = 0)$) của dữ liệu DPCM của các hệ số AC.

Bộ DPCM hệ số AC 613 thu được các hệ số AC được cấp từ bộ xác định xếp chồng 191, xác định, đối với mỗi hệ số AC thứ hai và tiếp theo, độ chênh lệch (DPCM) từ hệ số AC ngay trước đó, và cấp các độ chênh lệch được xác

định tới bộ exp-G 193 làm dữ liệu DPCM ($\text{scaling_list_delta_coef}(i = 1 \text{ tới } 63)$).

Bộ DPCM hệ số DC 614 thu được hệ số DC được cấp từ bộ xác định xếp chồng 191. Bộ DPCM hệ số DC 614 còn thu được hệ số AC khởi tạo (hệ số AC (0, 0)) được lưu trữ trong bộ đệm hệ số AC 611. Bộ DPCM hệ số DC 614 trừ hệ số AC khởi tạo (hệ số AC (0, 0)) từ hệ số DC để xác định độ chênh lệch giữa chúng, và cấp độ chênh lệch được xác định tới bộ exp-G 193 làm dữ liệu DPCM của hệ số DC ($\text{scaling_list_dc_coef_delta}$).

Như được mô tả trên đây, trong phương pháp thứ hai, độ chênh lệch giữa hệ số DC và hệ số khác (hệ số AC khởi tạo) được xác định. Sau đó, độ chênh lệch này được truyền, làm dữ liệu DPCM của hệ số DC ($\text{scaling_list_dc_coef_delta}$) khác với dữ liệu DPCM của các hệ số AC, sau khi truyền dữ liệu DPCM của các hệ số AC ($\text{scaling_list_delta_coef}$) mà là độ chênh lệch giữa các hệ số AC. Do đó, tương tự như phương pháp thứ nhất, thiết bị mã hóa ảnh 10 có thể cải thiện hiệu quả mã hóa của danh sách chia tỷ lệ.

3-3. Dòng xử lý của xử lý DPCM

Cũng trong phương pháp thứ hai, thiết bị mã hóa ảnh 10 thực hiện xử lý mã hóa ma trận lượng tử hóa theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.20.

Ví dụ của dòng xử lý của xử lý DPCM trong phương pháp thứ hai, mà được thực hiện trong bước S112 trên Fig.20, sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.32.

Khi xử lý DPCM được bắt đầu, trong bước S401, bộ đệm hệ số AC 611 lưu trữ hệ số AC khởi tạo.

Trong bước S402, bộ mã hóa hệ số AC 612 trừ hệ số AC khởi tạo từ hằng số định trước (ví dụ, 8) để xác định độ chênh lệch giữa chúng (dữ liệu DPCM khởi tạo).

Xử lý của các bước S403 tới S405 được thực hiện bởi bộ DPCM hệ số AC 613 theo cách tương tự như xử lý của các bước S133 tới S135 trên Fig.21. Tức là, xử lý của các bước S403 tới S405 được thực hiện lặp lại để tạo ra dữ liệu DPCM của tất cả các hệ số AC (các độ chênh lệch từ các hệ số AC ngay trước đó).

Nếu được xác định trong bước S403 rằng tất cả các hệ số AC được xử lý (tức là, nếu không có hệ số AC không được xử lý), bộ DPCM hệ số AC 613 chuyển xử lý sang bước S406.

Trong bước S406, bộ DPCM hệ số DC 614 trừ hệ số AC khởi tạo được lưu trữ trong bước S401 từ hệ số DC để xác định độ chênh lệch giữa chúng (dữ liệu DPCM đối với hệ số DC).

Khi xử lý của bước S406 được hoàn thành, bộ DPCM hệ số DC 614 kết thúc xử lý DPCM, và đưa xử lý quay lại Fig.20.

Do đó, độ chênh lệch giữa hệ số DC và hệ số khác cũng được xác định và được truyền tới thiết bị giải mã ảnh như là dữ liệu DPCM. Do đó, thiết bị mã hóa ảnh 10 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

3-4. Cấu trúc ví dụ chi tiết của bộ DPCM ngược

Trong phương pháp thứ hai, thiết bị giải mã ảnh 300 có cấu trúc về cơ bản tương tự như cấu trúc trong phương pháp thứ nhất. Cụ thể, cũng trong phương pháp thứ hai, thiết bị giải mã ảnh 300 có cấu trúc như trong ví dụ được minh họa trên Fig.22. Ngoài ra, bộ biến đổi trực giao ngược/giải lượng tử hóa 313 có cấu trúc như trong ví dụ được minh họa trên Fig.23. Ngoài ra, bộ tạo ma trận 410 có cấu trúc như trong ví dụ được minh họa trên Fig.24.

Fig.33 là sơ đồ khối minh họa ví dụ của cấu trúc chi tiết của bộ DPCM ngược 552 được minh họa trên Fig.24 trong phương pháp thứ hai. Viện dẫn tới Fig.33, bộ DPCM ngược 552 bao gồm bộ thiết lập khởi tạo 621, bộ giải mã DPCM hệ số AC 622, bộ đệm hệ số AC 623, và bộ giải mã DPCM hệ số DC 624.

Bộ thiết lập khởi tạo 621 thu được sizeID và MatrixID, và thiết lập các biến khác nhau thành các giá trị khởi tạo. Bộ thiết lập khởi tạo 621 cấp thông tin được thiết lập và thu được tới bộ giải mã DPCM hệ số AC 622.

Bộ giải mã DPCM hệ số AC 622 thu được dữ liệu DPCM của các hệ số AC (scaling_list_delta_coef) được cấp từ bộ exp-G 551. Bộ giải mã DPCM hệ số AC 622 giải mã dữ liệu DPCM thu được của các hệ số AC nhờ sử dụng các thiết lập khởi tạo và loại tương tự được cấp từ bộ thiết lập khởi tạo 621 để xác định các hệ số AC. Bộ giải mã DPCM hệ số AC 622 cấp các hệ số AC được xác định (ScalingList[i]) tới bộ xác định xếp chồng ngược 553. Bộ giải mã DPCM

hệ số AC 622 còn cấp hệ số AC khởi tạo (ScalingList[0], tức là, hệ số AC (0, 0)) trong số các hệ số AC được xác định tới bộ đệm hệ số AC 623 để lưu trữ.

Bộ đệm hệ số AC 623 lưu trữ hệ số AC khởi tạo (ScalingList[0], tức là, hệ số AC (0, 0)) được cấp từ bộ giải mã DPCM hệ số AC 622. Bộ đệm hệ số AC 623 cấp hệ số AC khởi tạo (ScalingList[0], tức là, hệ số AC (0, 0)) tới bộ giải mã DPCM hệ số DC 624 tại thời điểm định trước hoặc phản hồi lại yêu cầu.

Bộ giải mã DPCM hệ số DC 624 thu được dữ liệu DPCM của hệ số DC (scaling_list_dc_coef_delta) được cấp từ bộ exp-G 551. Bộ giải mã DPCM hệ số DC 624 còn thu được hệ số AC khởi tạo (ScalingList[0], tức là, hệ số AC (0, 0)) được lưu trữ trong bộ đệm hệ số AC 623. Bộ giải mã DPCM hệ số DC 624 giải mã dữ liệu DPCM của hệ số DC nhờ sử dụng hệ số AC khởi tạo để xác định hệ số DC. Bộ giải mã DPCM hệ số DC 624 cấp hệ số DC được xác định (DC_coef) tới bộ xác định xếp chồng ngược 553.

Do đó, bộ DPCM ngược 552 có thể thực hiện việc giải mã DPCM chính xác, và có thể thu được hệ số DC và các hệ số AC. Tức là, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

3-5. Dòng xử lý của xử lý DPCM ngược

Cũng trong phương pháp thứ hai, thiết bị giải mã ảnh 300 thực hiện xử lý giải mã ma trận lượng tử hóa theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả trên đây có viện dẫn đến lưu đồ được minh họa trên Fig.27. Tương tự, thiết bị giải mã ảnh 300 thực hiện xử lý giải mã tín hiệu dư theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả trên đây có viện dẫn đến lưu đồ được minh họa trên Fig.28.

Ví dụ của dòng xử lý của xử lý DPCM ngược được thực hiện bởi bộ DPCM ngược 552 sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.34.

Khi xử lý DPCM ngược được bắt đầu, trong bước S421, bộ thiết lập khởi tạo 621 thu được sizeID và MatrixID.

Trong bước S422, bộ thiết lập khởi tạo 621 thiết lập coefNum như sau.

$$\text{coefNum} = \min((1 \ll (4 + (\text{sizeID} \ll 1))), 64)$$

Trong bước S423, bộ thiết lập khởi tạo 621 thiết lập biến i và biến

nextcoef như sau.

$i = 0$

nextcoef = 8

Trong bước S424, bộ giải mã DPCM hệ số AC 622 xác định rằng biến $i < \text{coefNum}$ hay không. Nếu biến i nhỏ hơn so với coefNum , bộ thiết lập khởi tạo 621 chuyển xử lý sang bước S425.

Trong bước S425, bộ giải mã DPCM hệ số AC 622 đọc dữ liệu DPCM của các hệ số AC ($\text{scaling_list_delta_coef}$).

Trong bước S426, bộ giải mã DPCM hệ số AC 622 xác định nextcoef như dưới đây nhờ sử dụng dữ liệu DPCM được đọc, và còn xác định $\text{scalingList}[i]$.

$\text{nextcoef} = (\text{nextcoef} + \text{scaling_list_delta_coef} + 256) \% 256$

$\text{scalingList}[i] = \text{nextcoef}$

Lưu ý rằng hệ số AC khởi tạo được tính toán ($\text{ScalingList}[0]$, tức là, hệ số AC (0, 0)) được lưu trữ trong bộ đệm hệ số AC 623.

Trong bước S427, bộ giải mã DPCM hệ số AC 622 gia tăng biến i để thay đổi đích cần được xử lý thành hệ số tiếp theo, và sau đó đưa xử lý quay lại bước S424.

Trong bước S424, xử lý của các bước S424 tới S427 được thực hiện lặp lại cho đến khi được xác định rằng biến i lớn hơn hoặc bằng coefNum . Nếu được xác định trong bước S424 rằng biến i lớn hơn hoặc bằng coefNum , bộ giải mã DPCM hệ số AC 622 chuyển xử lý sang bước S428.

Trong bước S428, bộ giải mã DPCM hệ số DC 624 xác định rằng sizeID có lớn hơn 1 hay không. Nếu được xác định rằng sizeID lớn hơn 1, bộ giải mã DPCM hệ số DC 624 chuyển xử lý sang bước S429, và đọc dữ liệu DPCM của hệ số DC ($\text{scaling_list_dc_coef_delta}$).

Trong bước S430, bộ giải mã DPCM hệ số DC 624 thu được hệ số AC khởi tạo ($\text{ScalingList}[0]$, tức là, hệ số AC (0, 0)) được lưu trữ trong bộ đệm hệ số AC 623, và giải mã dữ liệu DPCM của hệ số DC (DC_coef) nhờ sử dụng hệ số AC khởi tạo như sau.

$\text{DC_coef} = \text{scaling_list_dc_coef_delta} + \text{ScalingList}[0]$

Khi hệ số DC (DC_coef) thu được, bộ giải mã DPCM hệ số DC 624 kết

thức xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Ngoài ra, nếu được xác định trong bước S428 rằng sizeID nhỏ hơn hoặc bằng 1, bộ giải mã DPCM hệ số DC 624 kết thúc xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Do đó, độ chênh lệch giữa hệ số DC và hệ số AC nằm tại phần đầu của các hệ số AC có thể được giải mã chính xác. Do đó, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

4. Phương án thứ tư

4-1. Cú pháp: phương pháp thứ ba

Trong phương pháp thứ hai được mô tả trên đây, hệ số DC cũng có thể được giới hạn ở giá trị nhỏ hơn so với hệ số AC khởi tạo (hệ số AC (0, 0)) (phương pháp thứ ba).

Điều này đảm bảo rằng dữ liệu DPCM của hệ số DC, tức là, giá trị chênh lệch thu được bằng cách trừ hệ số AC khởi tạo từ hệ số DC, có thể là giá trị dương. Dữ liệu DPCM này do đó có thể được mã hóa nhờ sử dụng các mã Golomb theo số mũ không dấu. Do đó, phương pháp thứ ba có thể ngăn ngừa hệ số DC lớn hơn so với hệ số AC khởi tạo, nhưng có thể làm giảm lượng mã hóa so với phương pháp thứ nhất và phương pháp thứ hai.

Fig.35 minh họa cú pháp của danh sách chia tỷ lệ trong phương pháp thứ ba. Như được minh họa trên Fig.35, trong trường hợp này, dữ liệu DPCM của hệ số DC (scaling_list_dc_coef_delta) được giới hạn ở giá trị dương.

Cú pháp đối với phương pháp thứ ba được mô tả trên đây có thể được thực hiện bởi thiết bị mã hóa ảnh 10 tương tự như trong phương pháp thứ hai. Trong phương pháp thứ ba, tuy nhiên, bộ exp-G 193 có thể mã hóa dữ liệu DPCM của hệ số DC nhờ sử dụng các mã Golomb theo số mũ không dấu. Lưu ý rằng thiết bị mã hóa ảnh 10 có thể thực hiện các xử lý như xử lý mã hóa ma trận lượng tử hóa và xử lý DPCM theo cách tương tự như trong phương pháp thứ hai.

Ngoài ra, cú pháp đối với phương pháp thứ ba có thể được thực hiện bởi thiết bị giải mã ảnh 300 theo cách tương tự như trong phương pháp thứ hai. Ngoài ra, thiết bị giải mã ảnh 300 có thể thực hiện xử lý giải mã ma trận lượng tử hóa theo cách tương tự như trong phương pháp thứ hai.

4-2. Dòng xử lý của xử lý DPCM ngược

Ví dụ của dòng xử lý của xử lý DPCM ngược được thực hiện bởi bộ DPCM ngược 552 sẽ được mô tả có việ dẫn tới lưu đồ được minh họa trên Fig.36.

Xử lý của các bước S451 đến S459 được thực hiện theo cách tương tự như xử lý của các bước S421 đến S429 trên Fig.34.

Trong bước S460, bộ giải mã DPCM hệ số DC 624 thu được hệ số AC khởi tạo (ScalingList[0], tức là, hệ số AC (0, 0)) được lưu trữ trong bộ đệm hệ số AC 623, và giải mã dữ liệu DPCM của hệ số DC (DC_coef) như dưới đây nhờ sử dụng hệ số AC khởi tạo.

$$\text{DC_coef} = \text{ScalingList}[0] - \text{scaling_list_dc_coef_delta}$$

Khi hệ số DC (DC_coef) thu được, bộ giải mã DPCM hệ số DC 624 kết thúc xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Ngoài ra, nếu được xác định trong bước S458 rằng sizeID nhỏ hơn hoặc bằng 1, bộ giải mã DPCM hệ số DC 624 kết thúc xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Do đó, độ chênh lệch giữa hệ số DC và hệ số AC nằm tại phần đầu của các hệ số AC có thể được giải mã chính xác. Do đó, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

5. Phương án thứ năm

5-1. Cú pháp: phương pháp thứ tư

Phương pháp khác để truyền độ chênh lệch giữa hệ số DC và hệ số khác, thay vì hệ số DC, có thể là để, ví dụ, thu thập chỉ các hệ số DC của các danh sách chia tỷ lệ và để thực hiện DPCM bằng cách lấy các độ chênh lệch giữa các hệ số DC một cách riêng biệt từ các hệ số AC của các danh sách chia tỷ lệ riêng biệt (phương pháp thứ tư). Trong trường hợp này, dữ liệu DPCM của các hệ số DC là tập hợp của các đoạn dữ liệu đối với các danh sách chia tỷ lệ, và được truyền như là dữ liệu khác với dữ liệu DPCM của các hệ số AC của các danh sách chia tỷ lệ riêng biệt.

Do đó, tỷ lệ nén có thể được cải thiện hơn nữa khi, ví dụ, có các tương quan giữa các hệ số DC của các danh sách chia tỷ lệ (MatrixID).

Fig.37 minh họa cú pháp đối với hệ số DC của danh sách chia tỷ lệ trong phương pháp thứ tư. Trong trường hợp này, do các hệ số DC được xử lý trong các chu kỳ khác với các chu kỳ đối với các hệ số AC của các danh sách chia tỷ lệ riêng biệt, như được minh họa trên ví dụ được minh họa trên Fig.37, các xử lý đối với các hệ số AC và các xử lý đối với các hệ số DC cần độc lập với nhau.

Điều này đảm bảo rằng nhiều hơn các phương pháp khác nhau đối với các xử lý mã hóa và giải mã danh sách chia tỷ lệ có thể đạt được mặc dù độ phức tạp của xử lý DPCM và xử lý DPCM ngược có thể được tăng lên. Ví dụ, xử lý sao chép chỉ các hệ số AC và làm cho các giá trị của các hệ số DC khác nhau trong chế độ sao chép có thể được thực hiện dễ dàng.

Số lượng các danh sách chia tỷ lệ trong đó các hệ số DC được xử lý tổng hợp là tùy ý.

5-2. Cấu trúc ví dụ chi tiết của bộ DPCM

Trong phương pháp thứ tư, thiết bị mã hóa ảnh 10 có cấu trúc về cơ bản tương tự như cấu trúc trong phương pháp thứ nhất được mô tả trên đây. Cụ thể, thiết bị mã hóa ảnh 10 có cấu trúc như trong ví dụ được minh họa trên Fig.14. Ngoài ra, bộ lượng tử hóa/biến đổi trực giao 14 có cấu trúc như trong ví dụ được minh họa trên Fig.15. Ngoài ra, bộ xử lý ma trận 150 có cấu trúc như trong ví dụ được minh họa trên Fig.16.

Cấu trúc ví dụ của bộ DPCM 192 trong phương pháp thứ tư được minh họa trên Fig.38. Như được minh họa trên Fig.38, trong trường hợp này, bộ DPCM 192 bao gồm bộ DPCM hệ số AC 631, bộ đệm hệ số DC 632, và bộ DPCM hệ số DC 633.

Bộ DPCM hệ số AC 631 thực hiện xử lý DPCM của các hệ số AC riêng biệt của mỗi danh sách chia tỷ lệ mà được cấp từ bộ xác định xếp chồng 191. Cụ thể, bộ DPCM hệ số AC 631 trừ, đối với mỗi danh sách chia tỷ lệ, hệ số AC khởi tạo từ hằng số định trước (ví dụ, 8), và trừ hệ số AC được xử lý (hệ số AC hiện tại) từ hệ số AC ngay trước đó. Bộ DPCM hệ số AC 631 cấp dữ liệu DPCM (scaling_list_delta_coef) được tạo ra đối với mỗi danh sách chia tỷ lệ tới bộ exp-G 193.

Bộ đệm hệ số DC 632 lưu trữ các hệ số DC của các danh sách chia tỷ lệ

riêng biệt được cấp từ bộ xác định xếp chồng 191. Bộ đệm hệ số DC 632 cấp các hệ số DC được lưu trữ tới bộ DPCM hệ số DC 633 tại thời điểm định trước hoặc phản hồi lại yêu cầu.

Bộ DPCM hệ số DC 633 thu được các hệ số DC được tích lũy trong bộ đệm hệ số DC 632. Bộ DPCM hệ số DC 633 xác định dữ liệu DPCM của các hệ số DC thu được. Cụ thể, bộ DPCM hệ số DC 633 trừ hệ số DC khởi tạo từ hằng số định trước (ví dụ, 8), và trừ hệ số DC đang được xử lý (hệ số DC hiện tại) từ hệ số DC ngay trước đó. Bộ DPCM hệ số DC 633 cấp dữ liệu DPCM được tạo ra (`scaling_list_delta_coef`) tới bộ exp-G 193.

Do đó, thiết bị mã hóa ảnh 10 có thể cải thiện hiệu quả mã hóa của danh sách chia tỷ lệ.

5-3. Dòng xử lý của xử lý DPCM

Cũng trong phương pháp thứ tư, thiết bị mã hóa ảnh 10 thực hiện xử lý mã hóa ma trận lượng tử hóa theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả trên đây có viện dẫn đến lưu đồ được minh họa trên Fig.20.

Ví dụ của dòng xử lý của xử lý DPCM trong phương pháp thứ tư, mà được thực hiện trong bước S112 trên Fig.20, sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên Fig.39.

Xử lý của các bước S481 tới S485 được thực hiện bởi bộ DPCM hệ số AC 631 theo cách tương tự như xử lý của các bước S401 tới S405 (xử lý trong phương pháp thứ hai) trên Fig.32.

Nếu được xác định trong bước S483 rằng tất cả các hệ số AC được xử lý, bộ DPCM hệ số AC 631 chuyển xử lý sang bước S486.

Trong bước S486, bộ DPCM hệ số AC 631 xác định rằng tất cả các danh sách chia tỷ lệ (hoặc các ma trận chênh lệch) trong đó các hệ số DC được mã hóa DPCM tổng hợp có được xử lý hay không. Nếu được xác định rằng có danh sách chia tỷ lệ (hoặc ma trận chênh lệch) không được xử lý, bộ DPCM hệ số AC 631 đưa xử lý quay lại bước S481.

Nếu được xác định trong bước S486 rằng tất cả các danh sách chia tỷ lệ (hoặc các ma trận chênh lệch) được xử lý, bộ DPCM hệ số AC 631 chuyển xử lý sang bước S487.

Bộ DPCM hệ số DC 633 thực hiện xử lý của các bước S487 tới S491 trên các hệ số DC được lưu trữ trong bộ đệm hệ số DC 632 theo cách tương tự như xử lý của các bước S481 tới S485.

Nếu được xác định trong bước S489 rằng tất cả các hệ số DC được lưu trữ trong bộ đệm hệ số DC 632 được xử lý, bộ DPCM hệ số DC 633 kết thúc xử lý DPCM, và đưa xử lý quay lại Fig.20.

Bằng cách thực hiện xử lý DPCM theo cách được mô tả trên đây, thiết bị mã hóa ảnh 10 có thể cải thiện hiệu quả mã hóa của danh sách chia tỷ lệ.

5-4. Cấu trúc ví dụ chi tiết của bộ DPCM ngược

Thiết bị giải mã ảnh 300 trong phương pháp thứ tư có cấu trúc về cơ bản tương tự như cấu trúc trong phương pháp thứ nhất. Cụ thể, cũng trong phương pháp thứ tư, thiết bị giải mã ảnh 300 có cấu trúc như trong ví dụ được minh họa trên Fig.22. Ngoài ra, bộ biến đổi trực giao ngược/giải lượng tử hóa 313 có cấu trúc như trong ví dụ được minh họa trên Fig.23. Ngoài ra, bộ tạo ma trận 410 có cấu trúc như trong ví dụ được minh họa trên Fig.24.

Fig.40 là sơ đồ khối minh họa ví dụ của cấu trúc chi tiết của bộ DPCM ngược 552 được minh họa trên Fig.24 trong phương pháp thứ tư. Viện dẫn tới Fig.40, bộ DPCM ngược 552 bao gồm bộ thiết lập khởi tạo 641, bộ giải mã DPCM hệ số AC 642, và bộ giải mã DPCM hệ số DC 643.

Bộ thiết lập khởi tạo 641 thu được sizeID và MatrixID, và thiết lập các biến khác nhau thành các giá trị khởi tạo. Bộ thiết lập khởi tạo 641 cấp thông tin được thiết lập và thu được tới bộ giải mã DPCM hệ số AC 642 và bộ giải mã DPCM hệ số DC 643.

Bộ giải mã DPCM hệ số AC 642 thu được dữ liệu DPCM của các hệ số AC (scaling_list_delta_coef(ac)) được cấp từ bộ exp-G 551. Bộ giải mã DPCM hệ số AC 642 giải mã dữ liệu DPCM thu được của các hệ số AC nhờ sử dụng các thiết lập khởi tạo và loại tương tự được cấp từ bộ thiết lập khởi tạo 641, và xác định các hệ số AC. Bộ giải mã DPCM hệ số AC 642 cấp các hệ số AC được xác định (ScalingList[i]) tới bộ xác định xếp chồng ngược 553. Bộ giải mã DPCM hệ số AC 642 thực hiện xử lý được mô tả trên đây trên các danh sách chia tỷ lệ.

Bộ giải mã DPCM hệ số DC 643 thu được dữ liệu DPCM của hệ số DC (scaling_list_delta_coef(dc)) được cấp từ bộ exp-G 551. Bộ giải mã DPCM hệ số DC 643 giải mã dữ liệu DPCM thu được của hệ số DC nhờ sử dụng các thiết lập khởi tạo và loại tương tự được cấp từ bộ thiết lập khởi tạo 641, và xác định các hệ số DC của các danh sách chia tỷ lệ riêng biệt. Bộ giải mã DPCM hệ số DC 643 cấp các hệ số DC định trước (scaling_list_dc_coef) tới bộ xác định xếp chồng ngược 553.

Do đó, bộ DPCM ngược 552 có thể thực hiện việc giải mã DPCM chính xác, và có thể thu được các hệ số DC và các hệ số AC. Tức là, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của các danh sách chia tỷ lệ.

5-5. Dòng xử lý của xử lý DPCM ngược

Cũng trong phương pháp thứ tư, thiết bị giải mã ảnh 300 thực hiện xử lý giải mã ma trận lượng tử hóa theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả trên đây có viện dẫn đến lưu đồ được minh họa trên Fig.27. Tương tự, thiết bị giải mã ảnh 300 thực hiện xử lý giải mã tín hiệu dư theo cách tương tự như xử lý trong phương pháp thứ nhất được mô tả trên đây có viện dẫn đến lưu đồ được minh họa trên Fig.28.

Ví dụ của dòng xử lý của xử lý DPCM ngược được thực hiện bởi bộ DPCM ngược 552 sẽ được mô tả có viện dẫn tới lưu đồ được minh họa trên các Fig.41 và 42.

Khi xử lý DPCM ngược được bắt đầu, bộ thiết lập khởi tạo 641 và bộ giải mã DPCM hệ số AC 642 thực hiện xử lý của các bước S511 tới S517 theo cách tương tự như trong xử lý của các bước S421 tới S427 trên Fig.34.

Nếu được xác định trong bước S514 rằng biến i lớn hơn hoặc bằng coefNum, bộ giải mã DPCM hệ số AC 642 chuyển xử lý sang bước S518.

Trong bước S518, bộ giải mã DPCM hệ số AC 642 xác định rằng tất cả các danh sách chia tỷ lệ (các ma trận chênh lệch) trong đó các hệ số DC được đưa vào xử lý DPCM chung có được xử lý hay không. Nếu được xác định rằng có danh sách chia tỷ lệ (ma trận chênh lệch) không được xử lý, bộ giải mã DPCM hệ số AC 642 đưa xử lý quay lại bước S511, và lặp lại thực hiện xử lý tiếp theo.

Ngoài ra, nếu được xác định rằng không có danh sách chia tỷ lệ (ma trận chênh lệch) không được xử lý, bộ giải mã DPCM hệ số AC 642 đưa xử lý tới Fig.42.

Trong bước S521 trên Fig.42, bộ thiết lập khởi tạo 641 thiết lập sizeID và biến nextcoef như sau.

sizeID = 2

nextcoef = 8

Ngoài ra, trong bước S522, bộ thiết lập khởi tạo 641 thiết lập MatrixID như sau.

MatrixID = 0

Trong bước S523, bộ giải mã DPCM hệ số DC 643 xác định rằng sizeID < 4 hay không. Nếu được xác định rằng sizeID nhỏ hơn 4, bộ giải mã DPCM hệ số DC 643 chuyển xử lý sang bước S524.

Trong bước S524, bộ giải mã DPCM hệ số DC 643 xác định rằng $\text{MatrixID} < (\text{sizeID} == 3)?2:6$ có được thỏa mãn hay không. Nếu được xác định rằng $\text{MatrixID} < (\text{sizeID} == 3)?2:6$ được thỏa mãn, bộ giải mã DPCM hệ số DC 643 chuyển xử lý sang bước S525.

Trong bước S525, bộ giải mã DPCM hệ số DC 643 đọc dữ liệu DPCM của hệ số DC (scaling_list_delta_coef).

Trong bước S526, bộ giải mã DPCM hệ số DC 643 xác định nextcoef như dưới đây nhờ sử dụng dữ liệu DPCM được đọc, và còn xác định scaling_dc_coef.

$\text{nextcoef} = (\text{nextcoef} + \text{scaling_list_delta_coef} + 256) \% 256$

$\text{scaling_dc_coef}[\text{sizeID} - 2][\text{MatrixID}] = \text{nextcoef}$

Trong bước S527, bộ giải mã DPCM hệ số DC 643 gia tăng MatrixID để thay đổi đích xử lý thành hệ số DC tiếp theo (danh sách chia tỷ lệ hoặc ma trận dư tiếp theo), và sau đó đưa xử lý quay lại bước S524.

Nếu được xác định trong bước S524 rằng $\text{MatrixID} < (\text{sizeID} == 3)?2:6$ không được thỏa mãn, bộ giải mã DPCM hệ số DC 643 chuyển xử lý sang bước S528.

Trong bước S528, bộ giải mã DPCM hệ số DC 643 gia tăng sizeID để thay đổi đích xử lý thành hệ số DC tiếp theo (danh sách chia tỷ lệ hoặc ma trận

đư tiếp theo), và sau đó đưa xử lý quay lại bước S523.

Nếu được xác định trong bước S523 rằng sizeID lớn hơn hoặc bằng 4, bộ giải mã DPCM hệ số DC 643 kết thúc xử lý DPCM ngược, và đưa xử lý quay lại Fig.28.

Do đó, các độ chênh lệch giữa các hệ số DC có thể được giải mã chính xác. Do đó, thiết bị giải mã ảnh 300 có thể ngăn ngừa việc tăng lượng mã hóa của các danh sách chia tỷ lệ.

6. Phương án thứ sáu

6-1. Cú pháp khác: ví dụ thứ nhất

Fig.43 minh họa ví dụ khác của cú pháp đối với danh sách chia tỷ lệ. Hình vẽ này tương ứng với Fig.12. Trong ví dụ được minh họa trên Fig.12, giá trị khởi tạo của nextcoef được thiết lập là hằng số định trước (ví dụ, 8). Ngoài ra, như được minh họa trên Fig.43, giá trị khởi tạo của nextcoef có thể được ghi đè với dữ liệu DPCM của hệ số DC (`scaling_list_dc_coef_minus8`).

Do đó, lượng mã hóa của các hệ số AC khởi tạo (các hệ số AC (0, 0)) trong danh sách chia tỷ lệ 16x16 và danh sách chia tỷ lệ 32x32 có thể được làm giảm.

6-2. Cú pháp khác: ví dụ thứ hai

Fig.44 minh họa ví dụ khác của cú pháp đối với danh sách chia tỷ lệ. Hình vẽ này tương ứng với Fig.12.

Trong ví dụ được minh họa trên Fig.12, khi giá trị của `scaling_list_pred_matrix_id_delta`, mà là thông tin mà chỉ rõ đích tham chiếu trong chế độ sao chép, là “0”, danh sách chia tỷ lệ mà đặt trước danh sách chia tỷ lệ hiện tại được xử lý bởi một danh sách chia tỷ lệ được viện dẫn tới, và khi giá trị của `scaling_list_pred_matrix_id_delta` là “1”, danh sách chia tỷ lệ mà đặt trước danh sách chia tỷ lệ hiện tại được xử lý bởi hai danh sách chia tỷ lệ được viện dẫn tới.

Ngược lại, trong ví dụ được minh họa trên Fig.44, như được minh họa trên phần C của Fig.44, khi giá trị của `scaling_list_pred_matrix_id_delta`, mà là thông tin mà chỉ rõ đích tham chiếu trong chế độ sao chép, là “0”, danh sách chia tỷ lệ mặc định được viện dẫn tới, và khi giá trị của

scaling_list_pred_matrix_id_delta là “1”, danh sách chia tỷ lệ ngay trước đó được viện dẫn tới.

Theo cách này, việc cải biến các ngữ nghĩa của scaling_list_pred_matrix_id_delta có thể đơn giản hóa cú pháp theo cách được minh họa trên phần B của Fig.44 và có thể làm giảm tải của xử lý DPCM và xử lý DPCM ngược.

6-3. Cú pháp khác: ví dụ thứ ba

Fig.45 minh họa ví dụ khác của cú pháp đối với danh sách chia tỷ lệ. Hình vẽ này tương ứng với Fig.12.

Trong ví dụ được minh họa trên Fig.45, cả hai ví dụ được minh họa trên Fig.43 và ví dụ được minh họa trên Fig.44 được mô tả trên đây được sử dụng.

Trong ví dụ được minh họa trên Fig.45, do đó, lượng mã hóa của các hệ số AC khởi tạo (các hệ số AC (0, 0)) trong danh sách chia tỷ lệ 16x16 và danh sách chia tỷ lệ 32x32 có thể được làm giảm. Ngoài ra, cú pháp có thể được đơn giản hóa và tải của xử lý DPCM và xử lý DPCM ngược có thể được làm giảm.

Trong các phương án nêu trên, các giá trị của các hằng số định trước là tùy ý. Ngoài ra, các kích cỡ của các danh sách chia tỷ lệ cũng là tùy ý.

Ngoài ra, trong khi phần mô tả nêu trên được đưa ra về xử lý biến đổi kích cỡ đối với danh sách chia tỷ lệ, ma trận dự đoán, hoặc ma trận chênh lệch giữa chúng, xử lý biến đổi kích cỡ có thể là xử lý để thực tế tạo ra ma trận mà kích cỡ của nó được biến đổi, hoặc có thể là xử lý để thiết lập cách thức để đọc mỗi thành phần trong ma trận từ bộ nhớ (đọc điều khiển của ma trận dữ liệu) mà không thực tế tạo ra dữ liệu của ma trận.

Trong xử lý biến đổi kích cỡ được mô tả trên đây, mỗi thành phần trong ma trận mà kích cỡ của nó được biến đổi được cấu thành bởi bất kỳ thành phần trong ma trận mà kích cỡ của nó chưa được biến đổi. Tức là, ma trận mà kích cỡ của nó được biến đổi có thể được tạo ra bằng cách đọc các thành phần trong ma trận mà kích cỡ của nó chưa được biến đổi, mà được lưu trữ trong bộ nhớ, nhờ sử dụng phương pháp đã biết như đọc một vài thành phần trong ma trận hoặc đọc một thành phần nhiều lần. Nói cách khác, phương pháp đọc mỗi thành phần được xác định (hoặc đọc điều khiển của dữ liệu ma trận được thực hiện) để về

cơ bản thực hiện việc biến đổi kích cỡ được mô tả trên đây. Phương pháp này có thể loại bỏ xử lý như ghi dữ liệu ma trận mà kích cỡ của nó được biến đổi vào bộ nhớ. Ngoài ra, việc đọc của dữ liệu ma trận mà kích cỡ của nó được biến đổi về cơ bản phụ thuộc vào cách thức thực hiện phép nội suy lân cận gần nhất và loại tương tự, và do đó việc biến đổi kích cỡ có thể được thực hiện bởi xử lý tải tương đối thấp như lựa chọn một trong số nhiều tùy chọn thích hợp được bố trí trước. Do đó, phương pháp được mô tả trên đây có thể làm giảm tải của việc biến đổi kích cỡ.

Tức là, xử lý biến đổi kích cỡ được mô tả trên đây bao gồm xử lý để thực tế tạo ra dữ liệu ma trận mà kích cỡ của nó được biến đổi và cũng bao gồm đọc điều khiển của dữ liệu ma trận.

Lưu ý rằng trong khi phần mô tả nêu trên được thực hiện trong ngữ cảnh của ma trận chên lệch được mã hóa và được truyền, đây chỉ là minh họa và danh sách chia tỷ lệ có thể được mã hóa và được truyền. Nói cách khác, các hệ số AC và hệ số DC của danh sách chia tỷ lệ mà được mô tả trên đây như là các hệ số cần được xử lý có thể là các hệ số AC và hệ số DC của ma trận chên lệch giữa danh sách chia tỷ lệ và ma trận dự đoán.

Ngoài ra, lượng mã hóa đối với thông tin trên các tham số, các cờ, và v.v của danh sách chia tỷ lệ, như kích cỡ và ID danh sách của danh sách chia tỷ lệ, có thể được làm giảm bằng cách, ví dụ, lấy độ chên lệch giữa thông tin này và thông tin được truyền trước đó và truyền độ chên lệch này.

Ngoài ra, trong khi phần mô tả nêu trên được thực hiện trong ngữ cảnh của ma trận lượng tử hóa hoặc ma trận chên lệch có kích cỡ lớn được chuyển đổi xuống và được truyền, đây chỉ là minh họa và ma trận lượng tử hóa hoặc ma trận chên lệch có thể được truyền mà không được chuyển đổi xuống, trong khi kích cỡ của ma trận lượng tử hóa được sử dụng cho việc lượng tử hóa được giữ không thay đổi.

Sáng chế có thể được áp dụng cho loại bất kỳ của việc mã hóa và giải mã ảnh bao gồm việc lượng tử hóa và giải lượng tử.

Ngoài ra, sáng chế cũng có thể được áp dụng tới, ví dụ, thiết bị mã hóa ảnh và thiết bị giải mã ảnh được sử dụng để thu thông tin ảnh (dòng bit) được

nén nhờ sử dụng biến đổi trực giao như biến đổi cosin rời rạc và bù chuyển động, như MPEG hoặc H.26x, thông qua môi trường mạng như quảng bá vệ tinh, truyền hình cáp, Internet, hoặc điện thoại di động. Sáng chế cũng có thể được áp dụng cho thiết bị mã hóa ảnh và thiết bị giải mã ảnh được sử dụng để xử lý trên phương tiện lưu trữ như đĩa quang, đĩa từ, và bộ nhớ tốc độ nhanh. Ngoài ra, sáng chế cũng có thể được áp dụng cho thiết bị lượng tử hóa và thiết bị giải lượng tử hóa được chứa trong thiết bị mã hóa ảnh và thiết bị giải mã ảnh được mô tả trên đây, và loại tương tự.

7. Phương án thứ bảy

Áp dụng cho mã hóa ảnh đa cảnh nhìn và giải mã ảnh đa cảnh nhìn

Chuỗi xử lý được mô tả trên đây có thể được áp dụng cho việc mã hóa ảnh đa cảnh nhìn và việc giải mã ảnh đa cảnh nhìn. Fig.46 minh họa ví dụ của phương pháp mã hóa ảnh đa cảnh nhìn.

Như được minh họa trên Fig.46, các ảnh đa cảnh nhìn bao gồm các ảnh tại nhiều điểm nhìn (hoặc cảnh nhìn). Các cảnh nhìn trong các ảnh đa cảnh nhìn bao gồm các cảnh nhìn gốc, mỗi trong số chúng được mã hóa và giải mã sử dụng ảnh của nó mà không sử dụng ảnh của cảnh nhìn khác, và các cảnh nhìn không phải gốc, mỗi trong số chúng được mã hóa và giải mã sử dụng ảnh của cảnh nhìn khác. Mỗi cảnh nhìn không phải gốc có thể được mã hóa và giải mã nhờ sử dụng ảnh của cảnh nhìn gốc hoặc sử dụng ảnh của bất kỳ cảnh nhìn không phải gốc khác.

Khi các ảnh đa cảnh nhìn được minh họa trên Fig.46 cần được mã hóa và giải mã, ảnh của mỗi cảnh nhìn được mã hóa và giải mã. Phương pháp được mô tả trên đây trong các phương án nêu trên có thể được áp dụng cho việc mã hóa và giải mã của mỗi cảnh nhìn. Điều này có thể ngăn ngừa việc làm giảm chất lượng ảnh của mỗi cảnh nhìn riêng biệt.

Ngoài ra, các cờ và các tham số được sử dụng trong phương pháp được mô tả trên đây trong các phương án nêu trên có thể được chia sẻ trong việc mã hóa và giải mã của mỗi cảnh nhìn. Điều này có thể ngăn ngừa việc làm giảm hiệu quả mã hóa.

Cụ thể hơn, ví dụ, thông tin liên quan đến danh sách chia tỷ lệ (ví dụ, các

tham số, các cờ, và v.v) có thể được chia sẻ trong việc mã hóa và giải mã của mỗi cảnh nhìn.

Rõ ràng, bất kỳ thông tin cần thiết khác có thể được chia sẻ trong việc mã hóa và giải mã của mỗi cảnh nhìn.

Ví dụ, khi danh sách chia tỷ lệ hoặc thông tin liên quan đến danh sách chia tỷ lệ mà được chứa trong tập tham số chuỗi (sequence parameter set, SPS) hoặc tập tham số ảnh (picture parameter set, PPS) cần được truyền, nếu chúng (SPS và PPS) được chia sẻ trong số các cảnh nhìn, danh sách chia tỷ lệ hoặc thông tin liên quan đến danh sách chia tỷ lệ cũng được chia sẻ tương ứng. Điều này có thể ngăn ngừa việc làm giảm hiệu quả mã hóa.

Ngoài ra, các thành phần ma trận trong danh sách chia tỷ lệ (hoặc ma trận lượng tử hóa) đối với cảnh nhìn gốc có thể được thay đổi theo các giá trị chênh lệch giữa các cảnh nhìn. Ngoài ra, giá trị bù để điều chỉnh thành phần ma trận cảnh nhìn không phải gốc đối với thành phần ma trận trong danh sách chia tỷ lệ (ma trận lượng tử hóa) đối với cảnh nhìn gốc có thể được truyền. Do đó, việc tăng lượng mã hóa có thể được ngăn ngừa.

Ví dụ, danh sách chia tỷ lệ đối với mỗi cảnh nhìn có thể được truyền riêng biệt trước. Khi danh sách chia tỷ lệ cần được thay đổi đối với mỗi cảnh nhìn, chỉ thông tin chỉ báo độ chênh lệch từ danh sách chia tỷ lệ tương ứng trong số các danh sách chia tỷ lệ được truyền trước có thể được truyền. Thông tin chỉ báo độ chênh lệch là tùy ý, và có thể là, ví dụ, thông tin trong các đơn vị của 4x4 hoặc 8x8 hoặc độ chênh lệch giữa các ma trận.

Lưu ý rằng nếu danh sách chia tỷ lệ hoặc thông tin liên quan đến danh sách chia tỷ lệ được chia sẻ trong số các cảnh nhìn mặc dù SPS hoặc PPS không được chia sẻ, các SPS hoặc PPS đối với các cảnh nhìn khác có thể được viện dẫn tới (tức là, các danh sách chia tỷ lệ hoặc thông tin liên quan đến các danh sách chia tỷ lệ đối với các cảnh nhìn khác có thể được sử dụng).

Ngoài ra, nếu các ảnh đa cảnh nhìn này được biểu diễn như các ảnh có, làm các thành phần, các ảnh YUV và các ảnh độ sâu (độ sâu) tương ứng với lượng chênh lệch giữa các cảnh nhìn, danh sách chia tỷ lệ độc lập hoặc thông tin liên quan đến danh sách chia tỷ lệ đối với ảnh của mỗi thành phần (Y, U, V, và

độ sâu) có thể được sử dụng.

Ví dụ, do ảnh độ sâu (độ sâu) là ảnh của biên, các danh sách chia tỷ lệ là không cần thiết. Do đó, ngay cả mặc dù SPS hoặc PPS chỉ rõ việc sử dụng của danh sách chia tỷ lệ, danh sách chia tỷ lệ có thể không được áp dụng (hoặc danh sách chia tỷ lệ trong đó tất cả các thành phần ma trận là giống nhau (hoặc phẳng) có thể được áp dụng) tới ảnh độ sâu (độ sâu).

Thiết bị mã hóa ảnh đa cảnh nhìn

Fig.47 là sơ đồ minh họa thiết bị mã hóa ảnh đa cảnh nhìn để thực hiện thao tác mã hóa ảnh đa cảnh nhìn được mô tả trên đây. Như được minh họa trên Fig.47, thiết bị mã hóa ảnh đa cảnh nhìn 700 bao gồm bộ mã hóa 701, bộ mã hóa 702, và bộ ghép kênh 703.

Bộ mã hóa 701 mã hóa ảnh của cảnh nhìn gốc, và tạo ra dòng ảnh cảnh nhìn gốc được mã hóa. Bộ mã hóa 702 mã hóa ảnh của cảnh nhìn không phải gốc, và tạo ra dòng ảnh cảnh nhìn không phải gốc được mã hóa. Bộ ghép kênh 703 ghép kênh dòng ảnh cảnh nhìn gốc được mã hóa được tạo ra bởi bộ mã hóa 701 và dòng ảnh cảnh nhìn không phải gốc được mã hóa được tạo ra bởi bộ mã hóa 702, và tạo ra dòng ảnh đa cảnh nhìn được mã hóa.

Thiết bị mã hóa ảnh 10 (Fig.14) có thể được sử dụng đối với mỗi trong số của bộ mã hóa 701 và bộ mã hóa 702 của thiết bị mã hóa ảnh đa cảnh nhìn 700. Tức là, việc tăng lượng mã hóa của danh sách chia tỷ lệ trong việc mã hóa của mỗi cảnh nhìn có thể được ngăn ngừa, và việc giảm chất lượng ảnh của mỗi cảnh nhìn có thể được ngăn ngừa. Ngoài ra, bộ mã hóa 701 và bộ mã hóa 702 có thể thực hiện các xử lý như lượng tử hóa và giải lượng tử hóa nhờ sử dụng cùng các cờ hoặc các tham số (tức là, các cờ và các tham số có thể được chia sẻ). Do đó, việc giảm hiệu quả mã hóa có thể được ngăn ngừa.

Thiết bị giải mã ảnh đa cảnh nhìn

Fig.48 là sơ đồ minh họa thiết bị giải mã ảnh đa cảnh nhìn để thực hiện thao tác giải mã ảnh đa cảnh nhìn được mô tả trên đây. Như được minh họa trên Fig.48, thiết bị giải mã ảnh đa cảnh nhìn 710 bao gồm bộ giải ghép kênh 711, bộ giải mã 712, và bộ giải mã 713.

Bộ giải ghép kênh 711 giải ghép kênh dòng ảnh đa cảnh nhìn được mã hóa

trong đó dòng ảnh cảnh nhìn gốc được mã hóa và dòng ảnh cảnh nhìn không phải gốc được mã hóa được ghép kênh, và tách dòng ảnh cảnh nhìn gốc được mã hóa và dòng ảnh cảnh nhìn không phải gốc được mã hóa. Bộ giải mã 712 giải mã dòng ảnh cảnh nhìn gốc được mã hóa được tách bởi bộ giải ghép kênh 711, và thu được ảnh của cảnh nhìn gốc. Bộ giải mã 713 giải mã dòng ảnh cảnh nhìn không phải gốc được mã hóa được tách bởi bộ giải ghép kênh 711, và thu được ảnh của cảnh nhìn không phải gốc.

Thiết bị giải mã ảnh 300 (Fig.22) có thể được sử dụng đối với mỗi bộ giải mã 712 và bộ giải mã 713 của thiết bị giải mã ảnh đa cảnh nhìn 710. Tức là, việc tăng lượng mã hóa của danh sách chia tỷ lệ trong việc giải mã của mỗi cảnh nhìn có thể được ngăn ngừa, và việc giảm chất lượng ảnh của mỗi cảnh nhìn có thể được ngăn ngừa. Ngoài ra, bộ giải mã 712 và bộ giải mã 713 có thể thực hiện các xử lý như lượng tử hóa và giải lượng tử hóa nhờ sử dụng cùng các cờ và các tham số (tức là, các cờ và các tham số có thể được chia sẻ). Do đó, việc giảm hiệu quả mã hóa có thể được ngăn ngừa.

8. Phương án thứ tám

Áp dụng cho mã hóa ảnh được phân lớp và giải mã ảnh được phân lớp

Chuỗi xử lý được mô tả trên đây có thể được áp dụng cho mã hóa ảnh được phân lớp và giải mã ảnh được phân lớp (mã hóa có thể chia tỷ lệ và giải mã có thể chia tỷ lệ). Fig.49 minh họa ví dụ của phương pháp mã hóa ảnh được phân lớp.

Mã hóa ảnh được phân lớp (mã hóa có thể chia tỷ lệ) là xử lý để chia ảnh thành nhiều lớp (phân lớp) để cung cấp cho dữ liệu ảnh với chức năng chia tỷ lệ đối với tham số định trước và để mã hóa các lớp riêng biệt. Giải mã ảnh được phân lớp (giải mã có thể chia tỷ lệ) là xử lý giải mã tương ứng với việc mã hóa được phân lớp.

Như được minh họa trên Fig.49, trong việc tạo lớp ảnh, một ảnh được chia thành nhiều ảnh con (hoặc các lớp) nhờ sử dụng như là tham chiếu tham số định trước với chức năng chia tỷ lệ. Tức là, các ảnh được phân tách thành các lớp (hoặc các ảnh được phân lớp) bao gồm nhiều ảnh (hoặc lớp) được phân lớp có các giá trị khác nhau của tham số định trước. Các lớp trong các ảnh được phân

lớp bao gồm các lớp gốc, mỗi trong số chúng được mã hóa và giải mã nhờ sử dụng ảnh của nó mà không sử dụng ảnh của lớp khác, và các lớp không phải gốc (cũng được gọi là các lớp tăng cường), mỗi trong số chúng được mã hóa và giải mã sử dụng ảnh của lớp khác. Mỗi lớp không phải gốc có thể được mã hóa và giải mã nhờ sử dụng ảnh của lớp gốc hoặc sử dụng ảnh của bất kỳ lớp không phải gốc khác.

Nói chung, mỗi lớp không phải gốc bao gồm dữ liệu của ảnh chênh lệch (dữ liệu chênh lệch) giữa ảnh của nó và ảnh của lớp khác để làm giảm độ dư thừa. Ví dụ, trong trường hợp mà một ảnh được phân tách thành hai lớp, cụ thể là, lớp gốc và lớp không phải gốc (cũng được gọi là lớp tăng cường), ảnh có chất lượng thấp hơn so với ảnh gốc có thể thu được nhờ sử dụng chỉ dữ liệu của lớp gốc, và ảnh gốc (tức là, ảnh có chất lượng cao) có thể thu được bằng cách kết hợp dữ liệu của lớp gốc và dữ liệu của lớp không phải gốc.

Việc phân lớp của ảnh theo cách được mô tả trên đây có thể hỗ trợ thu được các ảnh có sự thay đổi rộng các chất lượng theo các trường hợp. Điều này đảm bảo rằng thông tin nén ảnh có thể được truyền từ máy chủ theo các khả năng của các thiết bị đầu cuối và các mạng mà không cần thực hiện việc chuyển mã sao cho, ví dụ, thông tin nén ảnh trên chỉ các lớp gốc được truyền tới các thiết bị đầu cuối có các khả năng xử lý thấp, như các điện thoại di động, để tái tạo các ảnh động có độ phân giải theo không gian-thời gian thấp hoặc chất lượng thấp, và thông tin nén ảnh trên các lớp tăng cường ngoài các lớp gốc được truyền tới các thiết bị đầu cuối có các khả năng xử lý cao, như các máy thu hình và máy tính cá nhân, để tái tạo các ảnh động có độ phân giải theo không gian-thời gian cao hoặc chất lượng cao.

Khi các ảnh được phân lớp như trong ví dụ được minh họa trên Fig.49 cần được mã hóa và giải mã, ảnh của mỗi lớp được mã hóa và giải mã. Phương pháp được mô tả trên đây trong mỗi trong số các phương án nêu trên có thể được áp dụng cho việc mã hóa và giải mã của mỗi lớp. Điều này có thể ngăn ngừa việc làm giảm chất lượng ảnh của các lớp riêng biệt.

Ngoài ra, các cờ và các tham số được sử dụng trong phương pháp được mô tả trên đây trong mỗi trong số các phương án nêu trên có thể được chia sẻ

trong việc mã hóa và giải mã của mỗi lớp. Điều này có thể ngăn ngừa việc làm giảm hiệu quả mã hóa.

Cụ thể hơn, ví dụ, thông tin liên quan đến danh sách chia tỷ lệ (ví dụ, các tham số, các cờ, và v.v) có thể được chia sẻ trong việc mã hóa và giải mã của mỗi lớp.

Rõ ràng, bất kỳ thông tin cần thiết khác có thể được chia sẻ trong việc mã hóa và giải mã của mỗi lớp.

Các ví dụ của các ảnh được phân lớp bao gồm các ảnh được phân lớp trong độ phân giải theo không gian (cũng được gọi là khả năng chia tỷ lệ độ phân giải theo không gian) (khả năng chia tỷ lệ không gian). Trong các ảnh được phân lớp với khả năng chia tỷ lệ độ phân giải theo không gian, các độ phân giải của các ảnh là khác nhau giữa từng lớp. Ví dụ, lớp của ảnh có độ phân giải theo không gian thấp nhất được chỉ định là lớp gốc, và lớp của ảnh có độ phân giải cao hơn so với lớp gốc được chỉ định là lớp không phải gốc (lớp tăng cường).

Dữ liệu ảnh của lớp không phải gốc (lớp tăng cường) có thể là dữ liệu độc lập với các lớp khác, và, tương tự như các lớp gốc, ảnh có độ phân giải tương đương độ phân giải của lớp này có thể thu được nhờ sử dụng chỉ dữ liệu ảnh. Nói chung, tuy nhiên, dữ liệu ảnh của lớp không phải gốc (lớp tăng cường) là dữ liệu tương ứng với ảnh chênh lệch giữa ảnh của lớp này và ảnh của lớp khác (ví dụ, từng lớp một dưới lớp này). Trong trường hợp này, ảnh có độ phân giải tương đương với độ phân giải của lớp gốc thu được nhờ chỉ sử dụng dữ liệu ảnh của lớp gốc trong khi ảnh có độ phân giải tương đương với độ phân giải của lớp không phải gốc (lớp tăng cường) thu được bằng kết hợp của dữ liệu ảnh của lớp này và dữ liệu ảnh của lớp khác (ví dụ, từng lớp một dưới lớp này). Điều này có thể ngăn ngừa độ dư thừa của dữ liệu ảnh giữa các lớp.

Trong các ảnh được phân lớp có khả năng chia tỷ lệ độ phân giải theo không gian được mô tả trên đây, các độ phân giải của các ảnh là khác nhau giữa từng lớp. Do đó, các độ phân giải của các đơn vị xử lý mà nhờ đó các lớp riêng biệt được mã hóa và giải mã cũng khác nhau. Do đó, nếu danh sách chia tỷ lệ (ma trận lượng tử hóa) được chia sẻ trong việc mã hóa và giải mã của các lớp riêng biệt, danh sách chia tỷ lệ (ma trận lượng tử hóa) có thể được chuyển đổi

lên theo các tỷ lệ độ phân giải của các lớp riêng biệt.

Ví dụ, giả thiết rằng ảnh của lớp gốc có độ phân giải của 2K (ví dụ, 1920x1080), và ảnh của lớp không phải gốc (lớp tăng cường) có độ phân giải của 4K (ví dụ, 3840x2160). Trong trường hợp này, ví dụ, kích cỡ 16x16 của ảnh của lớp gốc (ảnh 2K) tương ứng với kích cỡ 32x32 của ảnh của lớp không phải gốc (ảnh 4K). Danh sách chia tỷ lệ (ma trận lượng tử hóa) được chuyển đổi lên nếu thích hợp theo tỷ lệ độ phân giải.

Ví dụ, ma trận lượng tử hóa 4x4 được sử dụng cho việc lượng tử hóa và giải lượng tử hóa của lớp gốc được chuyển đổi lên thành 8x8 trong việc lượng tử hóa và giải lượng tử hóa của lớp không phải gốc và được sử dụng. Tương tự, danh sách chia tỷ lệ 8x8 của lớp gốc được chuyển đổi lên thành 16x16 trong lớp không phải gốc. Tương tự, ma trận lượng tử hóa được chuyển đổi lên thành 16x16 trong lớp gốc và được sử dụng được chuyển đổi lên thành 32x32 trong lớp không phải gốc.

Lưu ý rằng tham số để cho khả năng chia tỷ lệ được đưa ra không bị giới hạn ở độ phân giải theo không gian, và các ví dụ của tham số này có thể bao gồm độ phân giải theo thời gian (khả năng chia tỷ lệ theo thời gian). Trong các ảnh được phân lớp có khả năng chia tỷ lệ độ phân giải theo thời gian, các tốc độ khung của các ảnh là khác nhau giữa từng lớp. Các ví dụ khác bao gồm khả năng chia tỷ lệ độ sâu bit trong đó độ sâu bit của dữ liệu ảnh là khác nhau giữa từng lớp, và khả năng chia tỷ lệ thành phần màu trong đó định dạng của các thành phần là khác nhau giữa từng lớp.

Các ví dụ khác bao gồm khả năng chia tỷ lệ SNR trong đó các tỷ lệ tín hiệu trên nhiễu (SNR) của các ảnh là khác nhau giữa từng lớp.

Về mặt cải thiện chất lượng ảnh, có mong muốn rằng, ảnh có tỷ lệ tín hiệu trên nhiễu càng thấp, sai số lượng tử hóa được tạo ra càng thấp. Đối với việc này, trong khả năng chia tỷ lệ SNR, có mong muốn rằng, các danh sách chia tỷ lệ khác nhau (các danh sách chia tỷ lệ không phải chung) được sử dụng cho việc lượng tử hóa và giải lượng tử hóa của các lớp riêng biệt theo tỷ lệ tín hiệu trên nhiễu. Vì lý do này, như được mô tả trên đây, nếu danh sách chia tỷ lệ được chia sẻ trong số các lớp, giá trị bù để điều chỉnh các thành phần ma trận đối với lớp

tăng cường đối với các thành phần ma trận trong danh sách chia tỷ lệ đối với lớp gốc có thể được truyền. Cụ thể hơn, thông tin chỉ báo độ chênh lệch giữa danh sách chia tỷ lệ chung và danh sách chia tỷ lệ được sử dụng thực tế có thể được truyền trên cơ sở từng lớp. Ví dụ, thông tin chỉ báo độ chênh lệch có thể được truyền trong tập tham số chuỗi (SPS) hoặc tập tham số ảnh (PPS) đối với mỗi lớp. Thông tin chỉ báo độ chênh lệch là tùy ý. Ví dụ, thông tin này có thể là ma trận có các thành phần biểu diễn các giá trị chênh lệch giữa các thành phần tương ứng trong cả hai danh sách chia tỷ lệ, hoặc có thể là hàm chỉ báo độ chênh lệch này.

Thiết bị mã hóa ảnh được phân lớp

Fig.50 là sơ đồ minh họa thiết bị mã hóa ảnh được phân lớp để thực hiện thao tác mã hóa ảnh được phân lớp được mô tả trên đây. Như được minh họa trên Fig.50, thiết bị mã hóa ảnh được phân lớp 720 bao gồm bộ mã hóa 721, bộ mã hóa 722, và bộ ghép kênh 723.

Bộ mã hóa 721 mã hóa ảnh của lớp gốc, và tạo ra dòng ảnh lớp gốc được mã hóa. Bộ mã hóa 722 mã hóa ảnh của lớp không phải gốc, và tạo ra dòng ảnh lớp không phải gốc được mã hóa. Bộ ghép kênh 723 ghép kênh dòng ảnh lớp gốc được mã hóa được tạo ra bởi bộ mã hóa 721 và dòng ảnh lớp không phải gốc được mã hóa được tạo ra bởi bộ mã hóa 722, và tạo ra dòng ảnh được phân lớp được mã hóa.

Thiết bị mã hóa ảnh 10 (Fig.14) có thể được sử dụng đối với mỗi bộ mã hóa 721 và bộ mã hóa 722 của thiết bị mã hóa ảnh được phân lớp 720. Tức là, việc tăng lượng mã hóa của danh sách chia tỷ lệ trong việc mã hóa của mỗi lớp có thể được ngăn ngừa, và việc làm giảm chất lượng ảnh của mỗi lớp có thể được ngăn ngừa. Ngoài ra, bộ mã hóa 721 và bộ mã hóa 722 có thể thực hiện các xử lý như lượng tử hóa và giải lượng tử hóa nhờ sử dụng cùng các cờ hoặc các tham số (tức là, các cờ và các tham số có thể được chia sẻ). Do đó, việc giảm hiệu quả mã hóa có thể được ngăn ngừa.

Thiết bị giải mã ảnh được phân lớp

Fig.51 là sơ đồ minh họa thiết bị giải mã ảnh được phân lớp để thực hiện thao tác giải mã ảnh được phân lớp được mô tả trên đây. Như được minh họa

trên Fig.51, thiết bị giải mã ảnh được phân lớp 730 bao gồm bộ giải ghép kênh 731, bộ giải mã 732, và bộ giải mã 733.

Bộ giải ghép kênh 731 giải ghép kênh dòng ảnh được phân lớp được mã hóa trong đó dòng ảnh lớp gốc được mã hóa và dòng ảnh lớp không phải gốc được mã hóa được ghép kênh, và tách dòng ảnh lớp gốc được mã hóa và dòng ảnh lớp không phải gốc được mã hóa. Bộ giải mã 732 giải mã dòng ảnh lớp gốc được mã hóa được tách bởi bộ giải ghép kênh 731, và thu được ảnh của lớp gốc. Bộ giải mã 733 giải mã dòng ảnh lớp không phải gốc được mã hóa được tách bởi bộ giải ghép kênh 731, và thu được ảnh của lớp không phải gốc.

Thiết bị giải mã ảnh 300 (Fig.22) có thể được sử dụng đối với mỗi bộ giải mã 732 và bộ giải mã 733 của thiết bị giải mã ảnh được phân lớp 730. Tức là, việc tăng lượng mã hóa của danh sách chia tỷ lệ trong việc giải mã của mỗi lớp có thể được ngăn ngừa, và việc giảm chất lượng ảnh của mỗi lớp có thể được ngăn ngừa. Ngoài ra, bộ giải mã 712 và bộ giải mã 713 có thể thực hiện các xử lý như lượng tử hóa và giải lượng tử hóa nhờ sử dụng cùng các cờ hoặc các tham số (tức là, các cờ và các tham số có thể được chia sẻ). Do đó, việc giảm hiệu quả mã hóa có thể được ngăn ngừa.

9. Phương án thứ chín

Máy tính

Chuỗi xử lý được mô tả trên đây có thể được thực hiện bởi phần cứng hoặc cũng có thể được thực hiện bởi phần mềm. Trong trường hợp này, chuỗi xử lý có thể được thực hiện bởi, ví dụ, máy tính được minh họa trên Fig.52.

Trên Fig.52, CPU (Central Processing Unit, bộ xử lý trung tâm) 801 trong máy tính 800 thực hiện các thao tác xử lý khác nhau theo chương trình được lưu trữ trong ROM (Read Only Memory, bộ nhớ chỉ đọc) 802 hoặc chương trình được tải vào RAM (Random Access Memory, bộ nhớ truy nhập ngẫu nhiên) 803 từ bộ lưu trữ 813. RAM 803 cũng lưu trữ, nếu cần, dữ liệu và loại tương tự cần thiết cho CPU 801 để thực hiện các thao tác xử lý khác nhau.

CPU 801, ROM 802, và RAM 803 được kết nối với nhau thông qua kênh truyền 804. Giao diện vào/ra 810 cũng được kết nối tới kênh truyền 804.

Giao diện vào/ra 810 được kết nối tới bộ đầu vào 811, bộ đầu ra 812, bộ

lưu trữ 813, và bộ truyền thông 814. Bộ đầu vào 811 bao gồm bàn phím, chuột, panen cảm ứng chạm, đầu nối vào, và v.v.. Bộ đầu ra 812 bao gồm các thiết bị đầu ra mong muốn, như loa và màn hình bao gồm CRT (Cathode Ray Tube, đèn tia âm cực), LCD (Liquid Crystal Display, màn hình tinh thể lỏng), và OLED (Organic ElectroLuminescence Display, màn hình điện phát quang hữu cơ), đầu nối ra, và v.v.. Bộ lưu trữ 813 bao gồm phương tiện lưu trữ mong muốn như ổ cứng hoặc bộ nhớ tác động nhanh, và bộ điều khiển mà điều khiển đầu ra và đầu vào của phương tiện lưu trữ. Bộ truyền thông 814 bao gồm các thiết bị truyền thông không dây hoặc có dây mong muốn như modem, giao diện LAN, thiết bị USB (Universal Serial Bus, kênh truyền nối tiếp đa năng), và thiết bị Bluetooth (nhãn hiệu được đăng ký). Bộ truyền thông 814 thực hiện xử lý truyền thông với các thiết bị truyền thông khác thông qua các mạng bao gồm, ví dụ, Internet.

Ổ 815 còn được kết nối tới giao diện vào/ra 810, nếu cần. Vật ghi có thể tháo rời 821 như đĩa từ, đĩa quang, đĩa quang từ, hoặc bộ nhớ bán dẫn được đặt trong ổ 815, nếu cần. Ổ 815 đọc chương trình máy tính, dữ liệu, và loại tương tự từ vật ghi có thể tháo rời 821 được đặt trong đó theo sự điều khiển của, ví dụ, CPU 801. Dữ liệu và chương trình máy tính được đọc được cấp tới, ví dụ, RAM 803. Chương trình máy tính được đọc từ vật ghi có thể tháo rời 821 còn được cài đặt vào bộ lưu trữ 813, nếu cần.

Khi chuỗi xử lý được mô tả trên đây được thực hiện bởi phần mềm, chương trình cấu thành phần mềm được cài đặt từ mạng hoặc vật ghi.

Các ví dụ của vật ghi bao gồm, như được minh họa trên Fig.52, vật ghi có thể tháo rời 821, mà được phân phối riêng biệt từ thân thiết bị để cấp phát chương trình tới người dùng, như đĩa từ (bao gồm đĩa mềm), đĩa quang (bao gồm CD-ROM (Compact Disc - Read Only Memory, bộ nhớ chỉ đọc dạng đĩa compắc) và DVD (Digital Versatile Disc, đĩa đa năng số)), đĩa quang từ (bao gồm MD (Mini Disc, đĩa mini)), hoặc bộ nhớ bán dẫn mà trên đó chương trình được ghi. Các ví dụ khác của vật ghi bao gồm các cơ cấu được phân phối tới người dùng theo cách được tích hợp trước trong thân thiết bị, như ROM 802 và đĩa cứng được chứa trong bộ lưu trữ 813 mà trên đó chương trình được ghi.

Lưu ý rằng chương trình mà máy tính 800 thực hiện có thể là chương trình

trong đó các thao tác xử lý được thực hiện trong trình tự thời gian theo thứ tự được thể hiện ở đây, hoặc có thể là chương trình trong đó các thao tác xử lý được thực hiện song song hoặc tại các thời điểm cần thiết như khi được gọi.

Ngoài ra, các bước mô tả chương trình được lưu trữ trong vật ghi, như được sử dụng ở đây, bao gồm, một cách rõ ràng, các thao tác xử lý được thực hiện trong trình tự thời gian theo thứ tự được thể hiện, và các thao tác xử lý được thực hiện song song hoặc riêng biệt mà không cần thiết được thực hiện trong trình tự thời gian.

Ngoài ra, thuật ngữ “hệ thống”, như được sử dụng ở đây, liên quan đến tập hợp của các thành phần cấu thành (các cơ cấu, môđun (thành phần), ...) mà không quan tâm đến tất cả các thành phần cấu thành này có được bố trí trong cùng vỏ hay không. Do đó, các cơ cấu nằm trong các vỏ riêng biệt và được kết nối thông qua mạng, và cơ cấu bao gồm nhiều môđun nằm trong một vỏ được xác định như hệ thống.

Ngoài ra, cấu trúc được mô tả trên đây như là một cơ cấu (hoặc các bộ xử lý) có thể được chia thành nhiều cơ cấu (hoặc các bộ xử lý). Ngược lại, các cấu trúc được mô tả trên đây như nhiều cơ cấu (hoặc các bộ xử lý) có thể được kết hợp thành một cơ cấu (hoặc bộ xử lý). Ngoài ra, một cách rõ ràng, cấu trúc ngoài cấu trúc được mô tả trên đây có thể được bổ sung cho cấu trúc của mỗi cơ cấu (hoặc mỗi bộ xử lý). Ngoài ra, một phần của cấu trúc của cấu trúc (hoặc bộ xử lý) đã biết có thể được chứa trong cấu trúc của cơ cấu khác (hoặc bộ xử lý khác) nếu các cơ cấu (hoặc các bộ xử lý) về cơ bản có cùng cấu trúc và/hoặc thao tác về mặt toàn bộ hệ thống. Nói cách khác, các phương án của sáng chế không bị giới hạn ở các phương án nêu trên, và các cải biến khác nhau có thể được thực hiện mà không đi chệch khỏi phạm vi của sáng chế.

Trong khi các phương án ưu tiên của sáng chế đã được mô tả chi tiết có viện dẫn tới các hình vẽ kèm theo, phạm vi kỹ thuật của sáng chế không bị giới hạn ở các ví dụ được bộc lộ ở đây. Rõ ràng rằng chuyên gia trong lĩnh vực kỹ thuật của sáng chế có thể thực hiện các thay đổi hoặc cải biến khác nhau mà không đi chệch khỏi phạm vi của khái niệm kỹ thuật như được xác định trong phần yêu cầu bảo hộ, và có thể được hiểu rằng các thay đổi hoặc các cải biến

này rõ ràng cũng nằm trong phạm vi kỹ thuật của sáng chế.

Ví dụ, sáng chế có thể được thực hiện bằng cấu trúc điện toán đám mây trong đó các thiết bị chia sẻ và kết hợp để xử lý một chức năng thông qua mạng.

Ngoài ra, mỗi bước được minh họa trên các lưu đồ được mô tả trên đây có thể được thực hiện bởi một cơ cấu hoặc bởi nhiều cơ cấu theo cách được chia sẻ.

Ngoài ra, nếu một bước bao gồm nhiều xử lý, các xử lý được chứa trong một bước có thể được thực hiện bởi một cơ cấu hoặc bởi nhiều cơ cấu theo cách được chia sẻ.

Thiết bị mã hóa ảnh 10 (Fig.14) và thiết bị giải mã ảnh 300 (Fig.22) theo các phương án nêu trên có thể được áp dụng cho các phần khác nhau của thiết bị điện tử như bộ truyền hoặc bộ thu được sử dụng để cấp phát dữ liệu thông qua quảng bá vệ tinh, quảng bá có dây như TV cáp, hoặc Internet hoặc được sử dụng để cấp phát dữ liệu tới hoặc từ các thiết bị đầu cuối thông qua truyền thông mạng tế bào, thiết bị ghi mà ghi các ảnh trên các phương tiện lưu trữ như đĩa quang, đĩa từ, và bộ nhớ tác động nhanh, và thiết bị tái tạo mà tái tạo các ảnh từ các phương tiện lưu trữ này. Bốn ứng dụng ví dụ sẽ được mô tả sau đây.

10. Các ứng dụng ví dụ

Ứng dụng ví dụ thứ nhất: máy thu tivi

Fig.53 minh họa ví dụ của cấu trúc của thiết bị tivi mà các phương án nêu trên được áp dụng. Thiết bị tivi 900 bao gồm anten 901, bộ điều hướng 902, bộ giải ghép kênh 903, bộ giải mã 904, bộ xử lý tín hiệu video 905, bộ hiển thị 906, bộ xử lý tín hiệu audio 907, loa 908, giao diện bên ngoài 909, bộ điều khiển 910, giao diện người dùng 911, và kênh truyền 912.

Bộ điều hướng 902 tách tín hiệu trong kênh mong muốn từ tín hiệu quảng bá thu được thông qua anten 901, và giải điều chế tín hiệu được tách. Sau đó, bộ điều hướng 902 xuất ra dòng bit được mã hóa thu được bằng việc giải điều chế tới bộ giải ghép kênh 903. Nói cách khác, bộ điều hướng 902 thực hiện chức năng như bộ truyền trong thiết bị tivi 900 để thu dòng được mã hóa bao gồm các ảnh được mã hóa.

Bộ giải ghép kênh 903 giải ghép kênh dòng bit được mã hóa thành dòng video và dòng audio của chương trình cần được xem, và xuất ra các dòng được

giải ghép kênh tới bộ giải mã 904. Bộ giải ghép kênh 903 còn tách dữ liệu bổ sung như EPG (Electronic Program Guide, chỉ dẫn chương trình điện tử) từ dòng bit được mã hóa, và cấp dữ liệu được tách tới bộ điều khiển 910. Lưu ý rằng bộ giải ghép kênh 903 cũng có thể giải xáo trộn dòng bit được mã hóa nếu dòng bit được mã hóa đã được xáo trộn.

Bộ giải mã 904 giải mã dòng video và dòng audio được đưa vào từ bộ giải ghép kênh 903. Sau đó, bộ giải mã 904 xuất ra dữ liệu video được tạo ra thông qua xử lý giải mã tới bộ xử lý tín hiệu video 905. Bộ giải mã 904 còn xuất ra dữ liệu audio được tạo ra thông qua xử lý giải mã tới bộ xử lý tín hiệu audio 907.

Bộ xử lý tín hiệu video 905 tái tạo dữ liệu video được đưa vào từ bộ giải mã 904, và làm cho video được hiển thị trên bộ hiển thị 906. Bộ xử lý tín hiệu video 905 cũng có thể làm cho màn ứng dụng được cấp thông qua mạng được hiển thị trên bộ hiển thị 906. Bộ xử lý tín hiệu video 905 còn có thể thực hiện xử lý bổ sung, như khử nhiễu, trên dữ liệu video theo các thiết lập. Ngoài ra, bộ xử lý tín hiệu video 905 cũng có thể tạo ra ảnh GUI (Graphical User Interface, giao diện người dùng đồ họa) như trình đơn, phím, hoặc con trỏ, và xếp chồng ảnh được tạo ra trên ảnh đầu ra.

Bộ hiển thị 906 được điều hướng bởi tín hiệu điều hướng được cấp từ bộ xử lý tín hiệu video 905, và hiển thị video hoặc ảnh trên bề mặt video của thiết bị hiển thị (như màn hình tinh thể lỏng, màn hình plasma, hoặc OLED (Organic Electro Luminescence Display, màn hình phát quang điện hữu cơ) (màn hình EL hữu cơ)).

Bộ xử lý tín hiệu audio 907 thực hiện các xử lý tái tạo, như chuyển đổi D/A và khuếch đại, trên dữ liệu audio được đưa vào từ bộ giải mã 904, và làm cho audio được xuất ra từ loa 908. Bộ xử lý tín hiệu audio 907 có thể còn thực hiện xử lý bổ sung, như khử nhiễu, trên dữ liệu audio.

Giao diện bên ngoài 909 là giao diện để kết nối thiết bị tivi 900 tới thiết bị phía ngoài hoặc mạng. Ví dụ, dòng video hoặc dòng audio thu được thông qua giao diện bên ngoài 909 có thể được giải mã bởi bộ giải mã 904. Nói cách khác, giao diện bên ngoài 909 cũng thực hiện chức năng như bộ truyền trong thiết bị tivi 900 để thu dòng được mã hóa bao gồm các ảnh được mã hóa.

Bộ điều khiển 910 bao gồm bộ xử lý như CPU, và các bộ nhớ như RAM và ROM. Các bộ nhớ lưu trữ chương trình cần được thực hiện bởi CPU, dữ liệu chương trình, dữ liệu EPG, dữ liệu thu được thông qua mạng, và v.v.. Chương trình được lưu trữ trong các bộ nhớ được đọc và được thực hiện bởi CPU khi, ví dụ, thiết bị tivi 900 được khởi động. CPU thực hiện chương trình để điều khiển thao tác của thiết bị tivi 900 theo, ví dụ, tín hiệu thao tác được đưa vào từ giao diện người dùng 911.

Giao diện người dùng 911 được kết nối tới bộ điều khiển 910. Giao diện người dùng 911 bao gồm, ví dụ, các phím và các nút chuyển đổi để cho phép người dùng thao tác thiết bị tivi 900, bộ thu dùng cho tín hiệu điều khiển từ xa, và v.v. Giao diện người dùng 911 phát hiện thao tác của người dùng thông qua các thành phần nêu trên để tạo ra tín hiệu thao tác, và xuất ra tín hiệu thao tác được tạo ra tới bộ điều khiển 910.

Kênh truyền 912 đóng vai trò để kết nối bộ điều hưởng 902, bộ giải ghép kênh 903, bộ giải mã 904, bộ xử lý tín hiệu video 905, bộ xử lý tín hiệu audio 907, giao diện bên ngoài 909, và bộ điều khiển 910 với nhau.

Trong thiết bị tivi 900 có cấu trúc nêu trên, bộ giải mã 904 có chức năng của thiết bị giải mã ảnh 300 (Fig.22) theo các phương án nêu trên. Do đó, thiết bị tivi 900 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Ứng dụng ví dụ thứ hai: điện thoại di động

Fig.54 minh họa ví dụ của cấu trúc của điện thoại di động mà các phương án nêu trên được áp dụng. Điện thoại di động 920 bao gồm anten 921, bộ truyền thông 922, bộ mã hóa-giải mã audio 923, loa 924, micrô 925, bộ camera 926, bộ xử lý ảnh 927, bộ ghép kênh/giải ghép kênh 928, bộ ghi/tái tạo 929, bộ hiển thị 930, bộ điều khiển 931, bộ thao tác 932, và kênh truyền 933.

Anten 921 được kết nối tới bộ truyền thông 922. Loa 924 và micrô 925 được kết nối tới bộ mã hóa-giải mã audio 923. Bộ thao tác 932 được kết nối tới bộ điều khiển 931. Kênh truyền 933 đóng vai trò để kết nối bộ truyền thông 922, bộ mã hóa-giải mã audio 923, bộ camera 926, bộ xử lý ảnh 927, bộ ghép kênh/giải ghép kênh 928, bộ ghi/tái tạo 929, bộ hiển thị 930, và bộ điều khiển 931 với nhau.

Điện thoại di động 920 thực hiện các thao tác, như truyền và thu tín hiệu audio, truyền và thu thư điện tử hoặc dữ liệu ảnh, chụp ảnh, và ghi dữ liệu, trong các chế độ thao tác khác nhau bao gồm chế độ cuộc gọi thoại, chế độ truyền thông dữ liệu, chế độ chụp ảnh, và chế độ gọi điện thoại video.

Trong chế độ cuộc gọi thoại, tín hiệu audio tương tự được tạo ra bởi micro 925 được cấp tới bộ mã hóa-giải mã audio 923. Bộ mã hóa-giải mã audio 923 chuyển đổi tín hiệu audio tương tự thành dữ liệu audio, và thực hiện chuyển đổi A/D và nén trên dữ liệu audio được chuyển đổi. Bộ mã hóa-giải mã audio 923 sau đó xuất ra dữ liệu audio được nén tới bộ truyền thông 922. Bộ truyền thông 922 mã hóa và điều chế dữ liệu audio, và tạo ra tín hiệu truyền. Bộ truyền thông 922 sau đó truyền tín hiệu truyền được tạo ra tới trạm gốc (không được thể hiện) thông qua anten 921. Ngoài ra, bộ truyền thông 922 khuếch đại tín hiệu vô tuyến thu được thông qua anten 921, và thực hiện chuyển đổi tần số trên tín hiệu được khuếch đại để thu được tín hiệu thu. Sau đó, bộ truyền thông 922 giải điều chế và giải mã tín hiệu thu để tạo ra dữ liệu audio, và xuất ra dữ liệu audio được tạo ra tới bộ mã hóa-giải mã audio 923. Bộ mã hóa-giải mã audio 923 mở rộng dữ liệu audio, và thực hiện chuyển đổi D/A để tạo ra tín hiệu audio tương tự. Bộ mã hóa-giải mã audio 923 sau đó cấp tín hiệu audio được tạo ra tới loa 924 để đưa ra âm thanh.

Ngoài ra, trong chế độ truyền thông dữ liệu, ví dụ, bộ điều khiển 931 tạo ra dữ liệu văn bản mà tạo thành thư điện tử theo thao tác của người dùng thông qua bộ thao tác 932. Ngoài ra, bộ điều khiển 931 làm cho văn bản được hiển thị trên bộ hiển thị 930. Bộ điều khiển 931 còn tạo ra dữ liệu thư điện tử theo chỉ dẫn truyền được đưa ra từ người dùng thông qua bộ thao tác 932, và xuất ra dữ liệu thư điện tử được tạo ra tới bộ truyền thông 922. Bộ truyền thông 922 mã hóa và điều chế dữ liệu thư điện tử để tạo ra tín hiệu truyền. Sau đó, bộ truyền thông 922 truyền tín hiệu truyền được tạo ra tới trạm gốc (không được thể hiện) thông qua anten 921. Ngoài ra, bộ truyền thông 922 khuếch đại tín hiệu vô tuyến thu được thông qua anten 921, và thực hiện chuyển đổi tần số trên tín hiệu được khuếch đại để thu được tín hiệu thu. Sau đó, bộ truyền thông 922 giải điều chế và giải mã tín hiệu thu để khôi phục dữ liệu thư điện tử, và xuất ra dữ liệu thư

điện tử được khôi phục tới bộ điều khiển 931. Bộ điều khiển 931 làm cho nội dung của thư điện tử được hiển thị trên bộ hiển thị 930, và cũng làm cho dữ liệu thư điện tử được lưu trữ trong vật lưu trữ của bộ ghi/tái tạo 929.

Bộ ghi/tái tạo 929 bao gồm phương tiện lưu trữ có thể ghi/đọc mong muốn. Phương tiện lưu trữ có thể là, ví dụ, phương tiện lưu trữ gắn trong như RAM hoặc bộ nhớ tốc độ nhanh, hoặc phương tiện lưu trữ ngoài như đĩa cứng, đĩa từ, đĩa quang từ, đĩa quang, bộ nhớ USB, hoặc thẻ nhớ.

Ngoài ra, trong chế độ chụp ảnh, ví dụ, bộ camera 926 chụp ảnh của đối tượng để tạo ra dữ liệu ảnh, và xuất ra dữ liệu ảnh được tạo ra tới bộ xử lý ảnh 927. Bộ xử lý ảnh 927 mã hóa dữ liệu ảnh được đưa vào từ bộ camera 926, và làm cho dòng được mã hóa được lưu trữ trong phương tiện lưu trữ của bộ ghi/tái tạo 929.

Ngoài ra, trong chế độ gọi điện thoại video, ví dụ, bộ ghép kênh/giải ghép kênh 928 ghép kênh dòng video được mã hóa bởi bộ xử lý ảnh 927 và dòng audio được đưa vào từ bộ mã hóa-giải mã audio 923, và xuất ra dòng được ghép kênh tới bộ truyền thông 922. Bộ truyền thông 922 mã hóa và điều chế dòng này để tạo ra tín hiệu truyền. Sau đó, bộ truyền thông 922 truyền tín hiệu truyền được tạo ra tới trạm gốc (không được thể hiện) thông qua anten 921. Bộ truyền thông 922 còn khuếch đại tín hiệu vô tuyến thu được thông qua anten 921, và thực hiện chuyển đổi tần số trên tín hiệu được khuếch đại để thu được tín hiệu thu. Tín hiệu truyền và tín hiệu thu có thể bao gồm dòng bit được mã hóa. Bộ truyền thông 922 giải điều chế và giải mã tín hiệu thu để khôi phục dòng, và xuất ra dòng được khôi phục tới bộ ghép kênh/giải ghép kênh 928. Sau đó, bộ ghép kênh/giải ghép kênh 928 giải ghép kênh dòng đầu vào thành dòng video và dòng audio, và xuất ra dòng video và dòng audio tới bộ xử lý ảnh 927 và bộ mã hóa-giải mã audio 923, một cách tương ứng. Bộ xử lý ảnh 927 giải mã dòng video để tạo ra dữ liệu video. Dữ liệu video được cấp tới bộ hiển thị 930, và một loạt ảnh được hiển thị bởi bộ hiển thị 930. Bộ mã hóa-giải mã audio 923 mở rộng dòng audio, và thực hiện chuyển đổi D/A để tạo ra tín hiệu audio tương tự. Bộ mã hóa-giải mã audio 923 sau đó cấp tín hiệu audio được tạo ra tới loa 924 để đưa ra âm thanh.

Trong điện thoại di động 920 có cấu trúc nêu trên, bộ xử lý ảnh 927 có chức năng của thiết bị mã hóa ảnh 10 (Fig.14) và chức năng của thiết bị giải mã ảnh 300 (Fig.22) theo các phương án nêu trên. Do đó, điện thoại di động 920 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Ngoài ra, trong khi phần mô tả được đưa ra về điện thoại di động 920, ví dụ, thiết bị mã hóa ảnh và thiết bị giải mã ảnh mà sáng chế được áp dụng có thể được sử dụng trong, tương tự như điện thoại di động 920, thiết bị bất kỳ có chức năng tạo ảnh và chức năng truyền thông tương tự như của điện thoại di động 920, như PDA (Personal Digital Assistants, thiết bị số hỗ trợ cá nhân), điện thoại thông minh, UMPC (Ultra Mobile Personal Computer, máy tính cá nhân siêu di động), máy tính xách tay dạng nhỏ, hoặc máy tính cá nhân xách tay.

Ứng dụng ví dụ thứ ba: thiết bị ghi/tái tạo

Fig.55 minh họa ví dụ của cấu trúc của thiết bị ghi/tái tạo mà các phương án nêu trên được áp dụng. Thiết bị ghi/tái tạo 940 mã hóa, ví dụ, dữ liệu audio và dữ liệu video của chương trình quảng bá thu được, và ghi dữ liệu audio và dữ liệu video được mã hóa trên vật ghi. Ngoài ra, thiết bị ghi/tái tạo 940 cũng có thể mã hóa dữ liệu audio và dữ liệu video thu được từ, ví dụ, thiết bị khác, và ghi dữ liệu audio và dữ liệu video được mã hóa trên vật ghi. Ngoài ra, thiết bị ghi/tái tạo 940 tái tạo, ví dụ, dữ liệu được ghi trên vật ghi sử dụng màn hình và loa theo chỉ dẫn được đưa ra từ người dùng. Trong trường hợp này, thiết bị ghi/tái tạo 940 giải mã dữ liệu audio và dữ liệu video.

Thiết bị ghi/tái tạo 940 bao gồm bộ điều hưởng 941, giao diện bên ngoài 942, bộ mã hóa 943, HDD (Hard Disk Drive, ổ đĩa cứng) 944, ổ đĩa 945, bộ lựa chọn 946, bộ giải mã 947, OSD (On-Screen Display, hiển thị trên màn hình) 948, bộ điều khiển 949, và giao diện người dùng 950.

Bộ điều hưởng 941 tách tín hiệu trong kênh mong muốn từ tín hiệu quảng bá thu được thông qua anten (không được thể hiện), và giải điều chế tín hiệu được tách. Bộ điều hưởng 941 sau đó xuất ra dòng bit được mã hóa thu được bằng việc giải điều chế tới bộ lựa chọn 946. Nói cách khác, bộ điều hưởng 941 thực hiện chức năng như bộ truyền trong thiết bị ghi/tái tạo 940.

Giao diện bên ngoài 942 là giao diện để kết nối thiết bị ghi/tái tạo 940 tới

thiết bị phía ngoài hoặc mạng. Giao diện bên ngoài 942 có thể là, ví dụ, giao diện IEEE 1394, giao diện mạng, giao diện USB, giao diện bộ nhớ tốc độ nhanh, hoặc loại tương tự. Ví dụ, dữ liệu video và dữ liệu audio thu được thông qua giao diện bên ngoài 942 được đưa vào bộ mã hóa 943. Nói cách khác, giao diện bên ngoài 942 thực hiện chức năng như bộ truyền trong thiết bị ghi/tái tạo 940.

Bộ mã hóa 943 mã hóa dữ liệu video và dữ liệu audio được đưa vào từ giao diện bên ngoài 942 nếu dữ liệu video và dữ liệu audio chưa được mã hóa. Bộ mã hóa 943 sau đó xuất ra dòng bit được mã hóa tới bộ lựa chọn 946.

HDD 944 ghi dòng bit được mã hóa bao gồm dữ liệu nội dung được nén như video và audio, các chương trình khác nhau, và dữ liệu khác trên đĩa cứng trong. Ngoài ra, HDD 944 đọc dữ liệu nêu trên từ đĩa cứng khi tái tạo video và audio.

Ổ đĩa 945 ghi và đọc dữ liệu trên và từ vật ghi được đặt trong đó. Vật ghi được đặt trong ổ đĩa 945 có thể là, ví dụ, đĩa DVD (như DVD-Video, DVD-RAM, DVD-R, DVD-RW, DVD+R, hoặc DVD+RW) hoặc đĩa Blu-ray (nhãn hiệu được đăng ký).

Bộ lựa chọn 946 lựa chọn dòng bit được mã hóa được đưa vào từ bộ điều hướng 941 hoặc bộ mã hóa 943 khi ghi video và audio, và xuất ra dòng bit được mã hóa được lựa chọn tới HDD 944 hoặc ổ đĩa 945. Khi tái tạo video và audio, bộ lựa chọn 946 xuất ra dòng bit được mã hóa được đưa vào từ HDD 944 hoặc ổ đĩa 945 tới bộ giải mã 947.

Bộ giải mã 947 giải mã dòng bit được mã hóa để tạo ra dữ liệu video và dữ liệu audio. Bộ giải mã 947 sau đó xuất ra dữ liệu video được tạo ra tới OSD 948. Bộ giải mã 947 còn xuất ra dữ liệu audio được tạo ra tới loa ngoài.

OSD 948 tái tạo dữ liệu video được đưa vào từ bộ giải mã 947, và hiển thị video. Ngoài ra, OSD 948 cũng có thể xếp chồng ảnh GUI như trình đơn, phím, hoặc con trỏ trên video cần được hiển thị.

Bộ điều khiển 949 bao gồm bộ xử lý như CPU, và các bộ nhớ như RAM và ROM. Các bộ nhớ lưu trữ chương trình cần được thực hiện bởi CPU, dữ liệu chương trình, và v.v. Chương trình được lưu trữ trong các bộ nhớ được đọc và

được thực hiện bởi CPU khi, ví dụ, thiết bị ghi/tái tạo 940 được bắt đầu. CPU thực hiện chương trình để điều khiển thao tác của thiết bị ghi/tái tạo 940 theo, ví dụ, tín hiệu thao tác được đưa vào từ giao diện người dùng 950.

Giao diện người dùng 950 được kết nối tới bộ điều khiển 949. Giao diện người dùng 950 bao gồm, ví dụ, các phím và các nút chuyển đổi để cho phép người dùng thao tác thiết bị ghi/tái tạo 940, bộ thu dùng cho tín hiệu điều khiển từ xa, và v.v.. Giao diện người dùng 950 phát hiện thao tác của người dùng thông qua các thành phần nêu trên để tạo ra tín hiệu thao tác, và xuất ra tín hiệu thao tác được tạo ra tới bộ điều khiển 949.

Trong thiết bị ghi/tái tạo 940 có cấu trúc nêu trên, bộ mã hóa 943 có chức năng của thiết bị mã hóa ảnh 10 (Fig.14) theo các phương án nêu trên. Ngoài ra, bộ giải mã 947 có chức năng của thiết bị giải mã ảnh 300 (Fig.22) theo các phương án nêu trên. Do đó, thiết bị ghi/tái tạo 940 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

Ứng dụng ví dụ thứ tư: thiết bị tạo ảnh

Fig.56 minh họa ví dụ của cấu trúc của thiết bị tạo ảnh mà các phương án nêu trên được áp dụng. Thiết bị tạo ảnh 960 chụp ảnh của đối tượng để tạo ra dữ liệu ảnh, mã hóa dữ liệu ảnh, và ghi dữ liệu ảnh được mã hóa trên vật ghi.

Thiết bị tạo ảnh 960 bao gồm khối quang 961, bộ tạo ảnh 962, bộ xử lý tín hiệu 963, bộ xử lý ảnh 964, bộ hiển thị 965, giao diện bên ngoài 966, bộ nhớ 967, ổ đĩa 968, OSD 969, bộ điều khiển 970, giao diện người dùng 971, và kênh truyền 972.

Khối quang 961 được kết nối tới bộ tạo ảnh 962. Bộ tạo ảnh 962 được kết nối tới bộ xử lý tín hiệu 963. Bộ hiển thị 965 được kết nối tới bộ xử lý ảnh 964. Giao diện người dùng 971 được kết nối tới bộ điều khiển 970. Kênh truyền 972 đóng vai trò để kết nối bộ xử lý ảnh 964, giao diện bên ngoài 966, bộ nhớ 967, ổ đĩa 968, OSD 969, và bộ điều khiển 970 với nhau.

Khối quang 961 bao gồm thấu kính hội tụ, cơ cấu khẩu độ, và v.v.. Khối quang 961 tạo ra ảnh quang của đối tượng trên bề mặt tạo ảnh của bộ tạo ảnh 962. Bộ tạo ảnh 962 bao gồm cảm biến ảnh như cảm biến ảnh CCD hoặc CMOS, và chuyển đổi ảnh quang được tạo ra trên bề mặt tạo ảnh thành tín hiệu ảnh đóng

vai trò như tín hiệu điện bằng cách thực hiện biến đổi quang điện. Bộ tạo ảnh 962 sau đó xuất ra tín hiệu ảnh tới bộ xử lý tín hiệu 963.

Bộ xử lý tín hiệu 963 thực hiện các thao tác xử lý tín hiệu camera khác nhau, như hiệu chỉnh độ cong, hiệu chỉnh gamma, và hiệu chỉnh màu, trên tín hiệu ảnh được đưa vào từ bộ tạo ảnh 962. Bộ xử lý tín hiệu 963 xuất ra dữ liệu ảnh được đưa vào các thao tác xử lý tín hiệu camera tới bộ xử lý ảnh 964.

Bộ xử lý ảnh 964 mã hóa dữ liệu ảnh được đưa vào từ bộ xử lý tín hiệu 963 để tạo ra dữ liệu được mã hóa. Bộ xử lý ảnh 964 sau đó xuất ra dữ liệu được mã hóa được tạo ra tới giao diện bên ngoài 966 hoặc ổ đĩa 968. Ngoài ra, bộ xử lý ảnh 964 giải mã dữ liệu được mã hóa được đưa vào từ giao diện bên ngoài 966 hoặc ổ đĩa 968 để tạo ra dữ liệu ảnh. Bộ xử lý ảnh 964 sau đó xuất ra dữ liệu ảnh được tạo ra tới bộ hiển thị 965. Ngoài ra, bộ xử lý ảnh 964 cũng có thể xuất ra dữ liệu ảnh được đưa vào từ bộ xử lý tín hiệu 963 tới bộ hiển thị 965 để làm cho ảnh được hiển thị. Ngoài ra, bộ xử lý ảnh 964 cũng có thể xếp chồng dữ liệu hiển thị thu được từ OSD 969 trên ảnh cần được xuất ra tới bộ hiển thị 965.

OSD 969 tạo ra ảnh GUI như trình đơn, phím, hoặc con trỏ, và xuất ra ảnh được tạo ra tới bộ xử lý ảnh 964.

Giao diện bên ngoài 966 được tạo ra như là, ví dụ, đầu nối vào/ra USB. Giao diện bên ngoài 966 kết nối, ví dụ, thiết bị tạo ảnh 960 tới máy in khi in ảnh. Ổ đĩa còn được kết nối tới giao diện bên ngoài 966, nếu cần. Vật ghi có thể tháo rời như đĩa từ hoặc đĩa quang được đặt trong ổ đĩa, và chương trình được đọc từ vật ghi có thể tháo rời có thể được cài đặt vào thiết bị tạo ảnh 960. Ngoài ra, giao diện bên ngoài 966 cũng có thể được tạo thành như là giao diện mạng để được kết nối tới mạng như LAN hoặc Internet. Nói cách khác, giao diện bên ngoài 966 thực hiện chức năng như bộ truyền trong thiết bị tạo ảnh 960.

Vật ghi được đặt trong ổ đĩa 968 có thể là, ví dụ, vật ghi tháo rời bất kỳ có thể ghi/đọc như đĩa từ, đĩa quang từ, đĩa quang, hoặc bộ nhớ bán dẫn. Ngoài ra, vật ghi có thể được gắn cố định vào ổ đĩa 968, và có thể tạo thành ổ đĩa cứng gắn trong hoặc bộ lưu trữ không tháo rời như SSD (Solid State Drive, ổ bán dẫn).

Bộ điều khiển 970 bao gồm bộ xử lý như CPU, và các bộ nhớ như RAM và ROM. Các bộ nhớ lưu trữ chương trình cần được thực hiện bởi CPU, dữ liệu

chương trình, và v.v. Chương trình được lưu trữ trong các bộ nhớ được đọc và được thực hiện bởi CPU khi, ví dụ, thiết bị tạo ảnh 960 được khởi động. CPU thực hiện chương trình để điều khiển thao tác của thiết bị tạo ảnh 960 theo, ví dụ, tín hiệu thao tác được đưa vào từ giao diện người dùng 971.

Giao diện người dùng 971 được kết nối tới bộ điều khiển 970. Giao diện người dùng 971 bao gồm, ví dụ, các phím, các nút chuyển đổi, và v.v để cho phép người dùng thao tác thiết bị tạo ảnh 960. Giao diện người dùng 971 phát hiện thao tác của người dùng thông qua các thành phần nêu trên để tạo ra tín hiệu thao tác, và xuất ra tín hiệu thao tác được tạo ra tới bộ điều khiển 970.

Trong thiết bị tạo ảnh 960 có cấu trúc nêu trên, bộ xử lý ảnh 964 có chức năng của thiết bị mã hóa ảnh 10 (Fig.14) và chức năng của thiết bị giải mã ảnh 300 (Fig.22) theo các phương án nêu trên. Do đó, thiết bị tạo ảnh 960 có thể ngăn ngừa việc tăng lượng mã hóa của danh sách chia tỷ lệ.

7. Các ứng dụng ví dụ của mã hóa có thể chia tỷ lệ

Hệ thống thứ nhất

Tiếp theo, ví dụ cụ thể của việc sử dụng của dữ liệu được mã hóa có thể chia tỷ lệ mà được mã hóa nhờ sử dụng mã hóa có thể chia tỷ lệ (mã hóa (ảnh) được phân lớp) sẽ được mô tả. Mã hóa có thể chia tỷ lệ có thể được sử dụng để, ví dụ, lựa chọn dữ liệu cần được truyền, như trong ví dụ được minh họa trên Fig.57.

Trong hệ thống truyền dữ liệu 1000 được minh họa trên Fig.57, máy chủ phân phối 1002 đọc dữ liệu được mã hóa có thể chia tỷ lệ được lưu trữ trong bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1001, và phân phối dữ liệu được mã hóa có thể chia tỷ lệ tới các thiết bị đầu cuối, như máy tính cá nhân 1004, thiết bị AV1005, thiết bị dạng bảng 1006, và điện thoại di động 1007, thông qua mạng 1003.

Trong trường hợp này, máy chủ phân phối 1002 lựa chọn dữ liệu được mã hóa có chất lượng mong muốn theo đặc tính của thiết bị đầu cuối, môi trường truyền thông, và loại tương tự, và truyền dữ liệu được mã hóa được lựa chọn. Ngay cả nếu máy chủ phân phối 1002 truyền dữ liệu có chất lượng tốt hơn cần thiết, thiết bị đầu cuối có thể không luôn thu được ảnh chất lượng cao, và độ trễ

hoặc tràn bộ nhớ có thể diễn ra. Ngoài ra, dữ liệu này có thể chiếm băng thông truyền thông nhiều hơn cần thiết, hoặc có thể tăng tải trên thiết bị đầu cuối nhiều hơn cần thiết. Ngược lại, ngay cả nếu máy chủ phân phối 1002 truyền dữ liệu có chất lượng thấp hơn cần thiết, thiết bị đầu cuối có thể không cần thiết thu được ảnh với đủ chất lượng. Do đó, máy chủ phân phối 1002 đọc dữ liệu được mã hóa có thể chia tỷ lệ được lưu trữ trong bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1001, nếu cần, như dữ liệu được mã hóa có chất lượng thích hợp cho đặc tính của thiết bị đầu cuối, môi trường truyền thông, và loại tương tự, và truyền dữ liệu được mã hóa được đọc.

Ví dụ, giả thiết rằng bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1001 lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011 mà được mã hóa chia tỷ lệ. Dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011 là dữ liệu được mã hóa bao gồm lớp gốc và lớp tăng cường, và là dữ liệu mà được giải mã để thu được cả ảnh của lớp gốc và ảnh của lớp tăng cường.

Máy chủ phân phối 1002 lựa chọn lớp thích hợp theo đặc tính của thiết bị đầu cuối mà truyền dữ liệu, môi trường truyền thông, và loại tương tự, và đọc dữ liệu của lớp này. Ví dụ, máy chủ phân phối 1002 đọc dữ liệu được mã hóa có thể chia tỷ lệ chất lượng cao (BL+EL) 1011 từ bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1001, và truyền dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011 được đọc tới máy tính cá nhân 1004 hoặc thiết bị dạng bảng 1006, mà có các khả năng xử lý cao, như ban đầu. Ngược lại, ví dụ, máy chủ phân phối 1002 tách dữ liệu của lớp gốc từ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011, và truyền dữ liệu được tách của lớp gốc tới thiết bị AV 1005 và điện thoại di động 1007, mà có các khả năng xử lý thấp, như là dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1012 có cùng nội dung như dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011 nhưng có chất lượng thấp hơn so với dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1011.

Việc sử dụng của dữ liệu được mã hóa có thể chia tỷ lệ theo cách này hỗ trợ việc điều chỉnh lượng dữ liệu, nhờ đó ngăn ngừa sự xuất hiện của trễ hoặc việc tràn bộ nhớ và ngăn ngừa việc tăng không cần thiết trong tải trên thiết bị đầu cuối hoặc môi trường truyền thông. Ngoài ra, dữ liệu được mã hóa có thể

chia tỷ lệ (BL+EL) 1011 làm giảm độ dư thừa giữa các lớp, và do đó có lượng dữ liệu nhỏ hơn so với dữ liệu có dữ liệu được mã hóa riêng biệt của các lớp tương ứng. Do đó, vùng lưu trữ của bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1001 có thể được sử dụng hiệu quả hơn.

Lưu ý rằng do các thiết bị khác nhau như máy tính cá nhân 1004, thiết bị AV 1005, thiết bị dạng bảng 1006, và điện thoại di động 1007 có thể được sử dụng như là các thiết bị đầu cuối, đặc tính phần cứng của các thiết bị đầu cuối là khác nhau giữa từng thiết bị. Ngoài ra, do các ứng dụng khác nhau có thể được thực hiện bởi các thiết bị đầu cuối, các khả năng phần mềm của các ứng dụng có thể thay đổi. Ngoài ra, mạng 1003 đóng vai trò như là môi trường truyền thông có thể được thực hiện như là mạng truyền thông bất kỳ mà có thể là mạng có dây, không dây, hoặc cả hai, như Internet và LAN (Local Area Network, mạng vùng cục bộ), và có các khả năng truyền dữ liệu khác nhau. Đặc tính và các khả năng này có thể thay đổi phụ thuộc vào việc truyền thông khác và loại tương tự.

Do đó, trước khi bắt đầu truyền dữ liệu, máy chủ phân phối 1002 có thể truyền thông với thiết bị đầu cuối mà dữ liệu cần được truyền tới đó, và có thể thu được thông tin liên quan đến các khả năng của thiết bị đầu cuối, như đặc tính phần cứng của thiết bị đầu cuối hoặc đặc tính của ứng dụng (phần mềm) được thực hiện bởi thiết bị đầu cuối, và cũng như thông tin liên quan đến môi trường truyền thông, như băng thông khả dụng của mạng 1003. Ngoài ra, máy chủ phân phối 1002 có thể lựa chọn lớp thích hợp trên cơ sở của thông tin thu được.

Lưu ý rằng lớp có thể được tách bởi thiết bị đầu cuối. Ví dụ, máy tính cá nhân 1004 có thể giải mã dữ liệu được mã hóa có thể chia tỷ lệ được truyền (BL+EL) 1011, và hiển thị ảnh của lớp gốc hoặc ảnh của lớp tăng cường. Ngoài ra, ví dụ, máy tính cá nhân 1004 có thể tách dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1012 của lớp gốc từ dữ liệu được mã hóa có thể chia tỷ lệ được truyền (BL+EL) 1011, lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ được tách (BL) 1012, truyền dữ liệu được mã hóa có thể chia tỷ lệ được tách (BL) 1012 tới thiết bị khác, hoặc giải mã dữ liệu được mã hóa có thể chia tỷ lệ được tách (BL) 1012 để hiển thị ảnh của lớp gốc.

Rõ ràng, số lượng của bộ lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ

1001, số lượng của máy chủ phân phối 1002, số lượng của mạng 1003, và số lượng của thiết bị đầu cuối là tùy ý. Ngoài ra, trong khi phân mô tả được đưa ra về ví dụ trong đó máy chủ phân phối 1002 truyền dữ liệu tới thiết bị đầu cuối, các ví dụ của việc sử dụng không bị giới hạn ở ví dụ này. Hệ thống truyền dữ liệu 1000 có thể được sử dụng trong hệ thống bất kỳ mà lựa chọn lớp thích hợp, khi truyền dữ liệu được mã hóa mà được mã hóa nhờ sử dụng mã hóa có thể chia tỷ lệ tới thiết bị đầu cuối, theo các khả năng của thiết bị đầu cuối, môi trường truyền thông, và loại tương tự.

Ngoài ra, sáng chế cũng có thể được áp dụng cho hệ thống truyền dữ liệu 1000 như được minh họa trên Fig.57 được mô tả trên đây theo cách tương tự như áp dụng cho việc mã hóa phân cấp và giải mã phân cấp được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51, nhờ đó đạt được các hiệu quả tương tự các hiệu quả được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51.

Hệ thống thứ hai

Mã hóa có thể chia tỷ lệ cũng có thể được sử dụng cho, ví dụ, như trong ví dụ được minh họa trên Fig.58, việc truyền thông qua nhiều môi trường truyền thông.

Trong hệ thống truyền dữ liệu 1100 được minh họa trên Fig.58, trạm quảng bá 1101 truyền dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121 của lớp gốc thông qua quảng bá mặt đất 1111. Trạm quảng bá 1101 còn truyền (ví dụ, đóng gói và truyền) dữ liệu được mã hóa có thể chia tỷ lệ (EL) 1122 của lớp tăng cường thông qua mạng 1112 mong muốn được tạo thành từ mạng truyền thông mà có thể là có dây, không dây, hoặc cả hai.

Thiết bị đầu cuối 1102 có chức năng để thu quảng bá mặt đất 1111 từ trạm quảng bá 1101, và thu dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121 của lớp gốc được truyền thông qua quảng bá mặt đất 1111. Thiết bị đầu cuối 1102 còn có chức năng truyền thông để thực hiện truyền thông thông qua mạng 1112, và thu dữ liệu được mã hóa có thể chia tỷ lệ (EL) 1122 của lớp tăng cường được truyền thông qua mạng 1112.

Thiết bị đầu cuối 1102 giải mã dữ liệu được mã hóa có thể chia tỷ lệ (BL)

1121 của lớp gốc thu được thông qua quảng bá mặt đất 1111 theo, ví dụ, chỉ dẫn người dùng hoặc loại tương tự để thu được ảnh của lớp gốc, lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121, hoặc truyền dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121 tới thiết bị khác.

Ngoài ra, thiết bị đầu cuối 1102 kết hợp dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121 của lớp gốc thu được thông qua quảng bá mặt đất 1111 với dữ liệu được mã hóa có thể chia tỷ lệ (EL) 1122 của lớp tăng cường thu được thông qua mạng 1112 theo, ví dụ, chỉ dẫn người dùng hoặc loại tương tự để thu được dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL), và giải mã dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) để thu được ảnh của lớp tăng cường, lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL), hoặc truyền dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) tới thiết bị khác.

Như được mô tả trên đây, dữ liệu được mã hóa có thể chia tỷ lệ có thể được truyền thông qua, ví dụ, môi trường truyền thông khác nhau từ lớp này tới lớp khác. Do đó, tải có thể được phân phối, và trễ hoặc tràn bộ nhớ có thể được ngăn ngừa xảy ra.

Ngoài ra, môi trường truyền thông được sử dụng cho việc truyền có thể được lựa chọn đối với mỗi lớp theo mỗi trường hợp. Ví dụ, dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1121 của lớp gốc có lượng dữ liệu tương đối lớn có thể được truyền thông qua môi trường truyền thông có băng thông lớn, và dữ liệu được mã hóa có thể chia tỷ lệ (EL) 1122 của lớp tăng cường có lượng dữ liệu tương đối nhỏ có thể được truyền thông qua môi trường truyền thông có băng thông hẹp. Ngoài ra, ví dụ, môi trường truyền thông mà thông qua đó dữ liệu được mã hóa có thể chia tỷ lệ (EL) 1122 của lớp tăng cường được truyền có thể được chuyển đổi giữa mạng 1112 và quảng bá mặt đất 1111 theo băng thông khả dụng của mạng 1112. Rõ ràng, phần nêu trên áp dụng tương tự tới dữ liệu của lớp bất kỳ.

Việc điều khiển theo cách được mô tả trên đây có còn ngăn ngừa việc tăng tải của việc truyền dữ liệu.

Rõ ràng, số lượng lớp là tùy ý, và số lượng môi trường truyền thông được sử dụng cho việc truyền cũng là tùy ý. Ngoài ra, số lượng thiết bị đầu cuối 1102

mà dữ liệu được phân phối tới đó cũng là tùy ý. Ngoài ra, trong khi phân mô tả được đưa ra trong ngữ cảnh của quảng bá từ trạm quảng bá 1101 bằng ví dụ, các ví dụ sử dụng không bị giới hạn ở ví dụ này. Hệ thống truyền dữ liệu 1100 có thể được sử dụng trong hệ thống bất kỳ mà chia dữ liệu được mã hóa nhờ sử dụng mã hóa có thể chia tỷ lệ thành các phân đoạn trong các đơn vị của các lớp và truyền các phân đoạn dữ liệu thông qua nhiều dòng.

Ngoài ra, sáng chế cũng có thể được áp dụng cho hệ thống truyền dữ liệu 1100 như được minh họa trên Fig.58 được mô tả trên đây theo cách tương tự như áp dụng cho việc mã hóa phân cấp và giải mã phân cấp được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51, nhờ đó thu được các hiệu quả tương tự như các hiệu quả được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51.

Hệ thống thứ ba

Mã hóa có thể chia tỷ lệ cũng có thể được sử dụng cho việc, ví dụ, như trong ví dụ được minh họa trên Fig.59, lưu trữ của dữ liệu được mã hóa.

Trong hệ thống tạo ảnh 1200 được minh họa trên Fig.59, thiết bị tạo ảnh 1201 thực hiện mã hóa có thể chia tỷ lệ trên dữ liệu ảnh thu được bằng cách chụp ảnh của đối tượng 1211, và cấp dữ liệu kết quả tới thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 như là dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221.

Thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221 được cấp từ thiết bị tạo ảnh 1201 tại chất lượng tương ứng với mỗi trường hợp. Ví dụ, thông thường, thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 tách dữ liệu của lớp gốc từ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221, và lưu trữ dữ liệu được tách của lớp gốc như là dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1222 của lớp gốc có chất lượng thấp và lượng dữ liệu nhỏ. Ngược lại, ví dụ, trong thời gian đề phòng, thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221 có chất lượng cao và lượng dữ liệu lớn, như ban đầu.

Do đó, thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 có thể

bảo lưu ảnh chất lượng cao chỉ khi cần thiết. Điều này có thể ngăn ngừa việc tăng lượng dữ liệu trong khi ngăn ngừa việc giảm giá trị của ảnh do việc giảm chất lượng, và có thể cải thiện hiệu quả sử dụng của vùng lưu trữ.

Ví dụ, giả thiết rằng thiết bị tạo ảnh 1201 là camera bảo mật. Nếu đối tượng cần được giám sát (ví dụ, đối tượng xâm phạm) không xuất hiện trong ảnh được chụp (thời gian thông thường), có thể có khả năng rằng ảnh được chụp không có nội dung quan trọng. Do đó, việc làm giảm lượng dữ liệu được ưu tiên, và dữ liệu ảnh (dữ liệu được mã hóa có thể chia tỷ lệ) của ảnh được lưu trữ tại chất lượng thấp. Ngược lại, nếu đối tượng cần được giám sát xuất hiện như là đối tượng 1211 trong ảnh được chụp (thời gian đề phòng), có thể có khả năng rằng ảnh được chụp có nội dung quan trọng. Do đó, chất lượng ảnh được ưu tiên, và dữ liệu ảnh (dữ liệu được mã hóa có thể chia tỷ lệ) của ảnh được lưu trữ tại chất lượng cao.

Lưu ý rằng hoặc thời gian thông thường hoặc thời gian đề phòng có thể được xác định bởi, ví dụ, thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202 bằng cách phân tích ảnh. Ngoài ra, thiết bị tạo ảnh 1201 có thể xác định thời gian thông thường hoặc thời gian đề phòng, và có thể truyền kết quả xác định tới thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202.

Lưu ý rằng việc xác định của hoặc thời gian thông thường hoặc thời gian đề phòng có thể được dựa trên tiêu chuẩn bất kỳ, và ảnh mà việc xác định được dựa trên đó có thể có nội dung bất kỳ. Rõ ràng, các điều kiện ngoài nội dung của ảnh có thể được sử dụng như là tiêu chuẩn xác định. Trạng thái có thể được thay đổi theo, ví dụ, biên độ, dạng sóng, hoặc loại tương tự của audio được ghi, hoặc có thể được thay đổi tại các khoảng của khoảng thời gian định trước. Ngoài ra, trạng thái này có thể được thay đổi theo chỉ dẫn phía ngoài như chỉ dẫn người dùng.

Ngoài ra, trong khi phần mô tả được đưa ra của ví dụ của việc thay đổi giữa hai trạng thái, cụ thể là, thời gian thông thường và thời gian đề phòng, số lượng trạng thái là tùy ý, và việc thay đổi trạng thái có thể được thực hiện giữa nhiều hơn hai trạng thái, như thời gian thông thường, thời gian đề phòng, thời gian đề phòng cao hơn, và thời gian đề phòng cao hơn nữa. Lưu ý rằng giới hạn

trên của số lượng trạng thái được thay đổi phụ thuộc vào số lượng lớp của dữ liệu được mã hóa có thể chia tỷ lệ.

Ngoài ra, thiết bị tạo ảnh 1201 có thể có cấu trúc để xác định số lượng lớp của mã hóa có thể chia tỷ lệ theo trạng thái. Ví dụ, trong thời gian thông thường, thiết bị tạo ảnh 1201 có thể tạo ra dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1222 của lớp gốc có chất lượng thấp và lượng dữ liệu nhỏ, và cấp dữ liệu được mã hóa có thể chia tỷ lệ (BL) 1222 được tạo ra tới thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202. Ngoài ra, ví dụ, trong thời gian đề phòng, thiết bị tạo ảnh 1201 có thể tạo ra dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221 của lớp gốc có chất lượng cao và lượng dữ liệu lớn, và cấp dữ liệu được mã hóa có thể chia tỷ lệ (BL+EL) 1221 được tạo ra tới thiết bị lưu trữ dữ liệu được mã hóa có thể chia tỷ lệ 1202.

Trong khi camera bảo mật được mô tả như là ví dụ, hệ thống tạo ảnh 1200 có thể được sử dụng trong ứng dụng bất kỳ, và có thể được sử dụng trong các ứng dụng ngoài camera bảo mật.

Ngoài ra, sáng chế cũng có thể được áp dụng cho hệ thống tạo ảnh 1200 được minh họa trên Fig.59 được mô tả trên đây theo cách tương tự như áp dụng cho việc mã hóa phân cấp và giải mã phân cấp được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51, nhờ đó thu được các hiệu quả tương tự như các hiệu quả được mô tả trên đây có viện dẫn tới các hình vẽ từ Fig.49 tới Fig.51.

Lưu ý rằng sáng chế cũng có thể được áp dụng cho việc tạo dòng HTTP, như MPEG DASH, trong đó đoạn thích hợp của dữ liệu được mã hóa được lựa chọn và được sử dụng trong các đơn vị của phân đoạn từ trong số các đoạn của dữ liệu được mã hóa được bố trí trước và có các độ phân giải khác nhau. Nói cách khác, thông tin liên quan đến việc mã hóa và giải mã cũng có thể được chia sẻ trong số các đoạn của dữ liệu được mã hóa.

Rõ ràng thiết bị mã hóa ảnh và thiết bị giải mã ảnh mà sáng chế được áp dụng cũng có thể được áp dụng cho các thiết bị ngoài các thiết bị được mô tả trên đây hoặc tới các hệ thống.

Lưu ý rằng ví dụ đã được mô tả ở đây trong đó ma trận lượng tử hóa (hoặc hệ số được sử dụng để tạo thành ma trận lượng tử hóa) được truyền từ phía mã

hóa tới phía giải mã. Kỹ thuật truyền ma trận lượng tử hóa có thể dùng truyền hoặc ghi ma trận lượng tử hóa như là dữ liệu riêng biệt được kết hợp với dòng bit được mã hóa mà không ghép kênh tham số lượng tử hóa thành dòng bit được mã hóa. Thuật ngữ “kết hợp”, như được sử dụng ở đây, nghĩa là cho phép ảnh (mà có thể là một phần của ảnh, như lát hoặc khối) được chứa trong dòng bit được liên kết tới thông tin tương ứng với ảnh khi ảnh được giải mã. Tức là, thông tin này có thể được truyền trên đường truyền khác với đối với ảnh (hoặc dòng bit). Ngoài ra, thông tin này có thể được ghi trên vật ghi khác với đối với ảnh (hoặc dòng bit) (hoặc được ghi trong vùng ghi khác nhau của cùng vật ghi). Ngoài ra, thông tin và ảnh (hoặc dòng bit) có thể được kết hợp với nhau trong các đơn vị bất kỳ như nhiều khung, một khung, hoặc một phần trong khung.

Lưu ý rằng sáng chế cũng có thể đề xuất các cấu trúc sau đây.

(1) Thiết bị xử lý ảnh bao gồm:

mạch được tạo cấu hình để:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng được sao chép, làm cú pháp mà ngữ nghĩa của nó được thiết đặt sao cho ma trận lượng tử hóa mặc định được tham chiếu tới điều kiện là giá trị bằng 0, giá trị chênh lệch giữa giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa tham chiếu và giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa dòng;

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch thay thế, mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số (0, 0) của ma trận lượng tử hóa 16x16 và hệ số (0, 0) của ma trận lượng tử hóa 8x8, ma trận lượng tử hóa 16x16 thu được nhờ sử dụng xử lý lân cận gần nhất dựa trên ma trận lượng tử hóa 8x8 dùng để chuyển đổi lên;

lượng tử hóa dữ liệu hệ số biến đổi thu được bằng cách biến đổi trực giao ảnh, và tạo dữ liệu được lượng tử hóa; và

mã hóa dữ liệu được lượng tử hóa đã tạo và hệ số chênh lệch thay thế để tạo dữ liệu được mã hóa, dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế và giá trị chênh lệch.

(2) Thiết bị xử lý ảnh theo mục 1, trong đó mạch được tạo cấu hình để:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao

chép, hệ số chênh lệch, mà là độ chênh lệch giữa mỗi hệ số của ma trận lượng tử hóa 8×8 ; và

tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch.

(3) Thiết bị xử lý ảnh theo mục 2, trong đó mạch được tạo cấu hình để tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch mà tạo nhóm hệ số chênh lệch thay thế và hệ số chênh lệch.

(4) Thiết bị xử lý ảnh theo mục 3, trong đó mạch được tạo cấu hình để: mã hóa hệ số chênh lệch thay thế, và sau đó mã hóa hệ số chênh lệch; và tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch.

(5) Thiết bị xử lý ảnh theo mục 4, trong đó mạch được tạo cấu hình để: thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch ban đầu, mà là độ chênh lệch giữa hệ số thay thế và giá trị ban đầu của hệ số được thiết đặt cho ma trận lượng tử hóa; và

tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch ban đầu.

(6) Thiết bị xử lý ảnh theo mục 5, trong đó mạch được tạo cấu hình để: mã hóa hệ số chênh lệch ban đầu, và sau đó mã hóa nhóm hệ số chênh lệch; và

tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu và nhóm hệ số chênh lệch.

(7) Thiết bị xử lý ảnh theo mục 6, trong đó mạch được tạo cấu hình để: mã hóa hệ số chênh lệch ban đầu nhờ sử dụng mã hóa Golomb theo số mũ có dấu, và sau đó mã hóa nhóm hệ số chênh lệch nhờ sử dụng mã hóa Golomb theo số mũ có dấu; và

tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu được mã hóa và nhóm hệ số chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu.

(8) Thiết bị xử lý ảnh theo mục 7, trong đó mạch được tạo cấu hình để:

biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi; và

lượng tử hóa dữ liệu hệ số biến đổi đã tạo.

(9) Thiết bị xử lý ảnh theo mục 8, trong đó mạch được tạo cấu hình để:

biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi nhờ sử dụng bộ chuyển

đôi 16×16 ; và

lượng tử hóa dữ liệu hệ số biến đổi đã tạo nhờ sử dụng ma trận lượng tử hóa 16×16 .

(10) Phương pháp xử lý ảnh bao gồm các bước:

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng được sao chép, làm cú pháp mà ngữ nghĩa của nó được thiết đặt sao cho ma trận lượng tử hóa mặc định được tham chiếu tới điều kiện là giá trị bằng 0, giá trị chênh lệch giữa giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa tham chiếu và giá trị của thông tin nhận dạng mà nhận dạng ma trận lượng tử hóa dòng;

thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, hệ số chênh lệch thay thế, mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số (0, 0) của ma trận lượng tử hóa 16×16 và hệ số (0, 0) của ma trận lượng tử hóa 8×8 , ma trận lượng tử hóa 16×16 thu được nhờ sử dụng xử lý lân cận gần nhất dựa trên ma trận lượng tử hóa 8×8 dùng để chuyển đổi lên;

lượng tử hóa dữ liệu hệ số biến đổi thu được bằng cách biến đổi trực giao ảnh, và tạo dữ liệu được lượng tử hóa; và

mã hóa, nhờ sử dụng mạch, dữ liệu được lượng tử hóa đã tạo và hệ số chênh lệch thay thế để tạo dữ liệu được mã hóa, dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế và giá trị chênh lệch.

(11) Phương pháp xử lý ảnh theo mục 10, trong đó:

bước thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, thiết đặt hệ số chênh lệch, mà là độ chênh lệch giữa mỗi hệ số của ma trận lượng tử hóa 8×8 , và

bước mã hóa tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch.

(12) Phương pháp xử lý ảnh theo mục 11, trong đó bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số chênh lệch mà tạo nhóm hệ số chênh lệch thay thế và hệ số chênh lệch.

(13) Phương pháp xử lý ảnh theo mục 12, trong đó:

bước mã hóa mã hóa hệ số chênh lệch thay thế, và sau đó mã hóa hệ số chênh lệch, và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là nhóm hệ số

chênh lệch.

(14) Phương pháp xử lý ảnh theo mục 13, trong đó:

bước thiết đặt, trong trường hợp ma trận lượng tử hóa dòng không được sao chép, thiết đặt hệ số chênh lệch ban đầu, mà là độ chênh lệch giữa hệ số thay thế và giá trị ban đầu của hệ số được thiết đặt cho ma trận lượng tử hóa, và

bước mã hóa tạo dữ liệu được mã hóa bao gồm hệ số chênh lệch ban đầu.

(15) Phương pháp xử lý ảnh theo mục 14, trong đó:

bước mã hóa mã hóa hệ số chênh lệch ban đầu, và sau đó mã hóa nhóm hệ số chênh lệch, và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu và nhóm hệ số chênh lệch.

(16) Phương pháp xử lý ảnh theo mục 15, trong đó:

bước mã hóa mã hóa hệ số chênh lệch ban đầu nhờ sử dụng mã hóa Golomb theo số mũ có dấu, và sau đó mã hóa nhóm hệ số chênh lệch nhờ sử dụng mã hóa Golomb theo số mũ có dấu; và

bước mã hóa tạo dữ liệu được mã hóa mà cú pháp của nó là hệ số chênh lệch ban đầu được mã hóa và nhóm hệ số chênh lệch được mã hóa nhờ sử dụng mã hóa Golomb theo số mũ có dấu.

(17) Phương pháp xử lý ảnh theo mục 16, trong đó:

phương pháp này còn bao gồm bước biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi, và

bước lượng tử hóa lượng tử hóa dữ liệu hệ số biến đổi đã tạo.

(18) Phương pháp xử lý ảnh theo mục 17, trong đó:

bước biến đổi trực giao biến đổi trực giao ảnh để tạo dữ liệu hệ số biến đổi nhờ sử dụng bộ chuyển đổi 16×16 , và

bước lượng tử hóa lượng tử hóa dữ liệu hệ số biến đổi đã tạo nhờ sử dụng ma trận lượng tử hóa 16×16 .

YÊU CẦU BẢO HỘ

1. Thiết bị xử lý ảnh bao gồm:

mạch được tạo cấu hình để giải mã dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 mà thu được bằng cách chuyển đổi lên ma trận lượng tử hóa 8×8 theo xử lý nội suy lân cận gần nhất và hệ số $(0, 0)$ của ma trận lượng tử hóa 8×8 để tạo dữ liệu được lượng tử hóa;

thiết đặt ma trận lượng tử hóa 16×16 bằng cách thực hiện xử lý nội suy lân cận gần nhất đối với ma trận lượng tử hóa 8×8 trong đó hệ số thu được bằng cách cộng hệ số thay thế với hệ số chênh lệch thay thế được thiết đặt là số $(0, 0)$ của ma trận lượng tử hóa 8×8 ;

thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 bằng hệ số thay thế; và

giải lượng tử hóa dữ liệu được lượng tử hóa đã tạo sử dụng ma trận lượng tử hóa 16×16 trong đó hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 đã được thay thế bằng hệ số thay thế, và tạo dữ liệu hệ số biến đổi.

2. Thiết bị xử lý ảnh theo điểm 1, trong đó mạch được tạo cấu hình để giải mã dữ liệu được mã hóa bao gồm các hệ số chênh lệch mà là các độ chênh lệch giữa các hệ số của ma trận lượng tử hóa 8×8 , và mạch được tạo cấu hình để thiết đặt ma trận lượng tử hóa 8×8 bằng cách xử lý cộng đối với các hệ số chênh lệch.

3. Thiết bị xử lý ảnh theo điểm 2, trong đó mạch được tạo cấu hình để giải mã dữ liệu được mã hóa trong cú pháp trong đó hệ số chênh lệch thay thế và các hệ số chênh lệch được gọi chung là nhóm hệ số chênh lệch.

4. Thiết bị xử lý ảnh theo điểm 3, trong đó hệ số chênh lệch thay thế và các hệ số chênh lệch này trong cú pháp của dữ liệu được mã hóa theo thứ tự của hệ số chênh lệch thay thế và các hệ số chênh lệch, và mạch được tạo cấu hình để giải mã hệ số chênh lệch thay thế và các hệ số chênh lệch theo thứ tự của hệ số chênh lệch thay thế và các hệ số chênh lệch.

5. Thiết bị xử lý ảnh theo điểm 4, trong đó mạch được tạo cấu hình để giải mã dữ liệu được mã hóa bao gồm giá trị chênh lệch ban đầu giữa hệ số thay thế và giá trị ban đầu được thiết đặt đối với ma trận lượng tử hóa, và mạch được tạo

cấu hình để thiết đặt hệ số thay thế bằng cách cộng giá trị ban đầu với giá trị chênh lệch ban đầu.

6. Thiết bị xử lý ảnh theo điểm 5, trong đó giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch được chứa trong cú pháp của dữ liệu được mã hóa theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch, và mạch được tạo cấu hình để giải mã giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch.

7. Thiết bị xử lý ảnh theo điểm 6, trong đó giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch, bằng cách thực hiện xử lý mã hóa Golomb theo số mũ có dấu, có trong dữ liệu được mã hóa, và mạch được tạo cấu hình để thực hiện xử lý giải mã Golomb theo số mũ có dấu đối với giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch thu được bằng cách thực hiện mã hóa Golomb theo số mũ có dấu theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch.

8. Thiết bị xử lý ảnh theo điểm 7, trong đó mạch còn được tạo cấu hình để biến đổi ngược dữ liệu hệ số biến đổi đã tạo.

9. Thiết bị xử lý ảnh theo điểm 5, trong đó mạch được tạo cấu hình để biến đổi ngược theo bộ biến đổi 16×16 .

10. Phương pháp xử lý ảnh bao gồm các bước:

giải mã dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 mà thu được bằng cách chuyển đổi lên ma trận lượng tử hóa 8×8 theo xử lý nội suy lân cận gần nhất và hệ số $(0, 0)$ của ma trận lượng tử hóa 8×8 để tạo dữ liệu được lượng tử hóa;

thiết đặt ma trận lượng tử hóa 16×16 bằng cách thực hiện xử lý nội suy lân cận gần nhất đối với ma trận lượng tử hóa 8×8 trong đó hệ số thu được bằng cách cộng hệ số thay thế với hệ số chênh lệch thay thế được thiết đặt là số $(0, 0)$ của ma trận lượng tử hóa 8×8 ;

thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 với hệ số thay thế; và
giải lượng tử hóa dữ liệu được lượng tử hóa đã tạo sử dụng ma trận lượng tử hóa 16×16 trong đó hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 đã được thay thế bằng hệ số thay thế, và tạo dữ liệu hệ số biến đổi.

11. Phương pháp xử lý ảnh theo điểm 10, trong đó bước giải mã dữ liệu được mã hóa bao gồm các hệ số chênh lệch mà là các độ chênh lệch giữa các hệ số của ma trận lượng tử hóa 8×8 , và bước thiết đặt ma trận lượng tử hóa 8×8 bằng cách xử lý cộng đối với các hệ số chênh lệch.

12. Phương pháp xử lý ảnh theo điểm 11, trong đó bước giải mã dữ liệu được mã hóa trong cú pháp trong đó hệ số chênh lệch thay thế và các hệ số chênh lệch được gọi chung là nhóm hệ số chênh lệch.

13. Phương pháp xử lý ảnh theo điểm 12, trong đó hệ số chênh lệch thay thế và các hệ số chênh lệch được chứa trong cú pháp của dữ liệu được mã hóa theo thứ tự của hệ số chênh lệch thay thế và các hệ số chênh lệch, và bước giải mã bao gồm giải mã hệ số chênh lệch thay thế và các hệ số chênh lệch theo thứ tự của hệ số chênh lệch thay thế và các hệ số chênh lệch.

14. Phương pháp xử lý ảnh theo điểm 13, trong đó bước giải mã bao gồm giải mã dữ liệu được mã hóa bao gồm giá trị chênh lệch ban đầu giữa hệ số thay thế và giá trị ban đầu được thiết đặt đối với ma trận lượng tử hóa, và phương pháp này còn bao gồm bước thiết đặt hệ số thay thế bằng cách cộng giá trị ban đầu với giá trị chênh lệch ban đầu.

15. Phương pháp xử lý ảnh theo điểm 14, trong đó giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch được chứa trong cú pháp của dữ liệu được mã hóa theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch, và bước giải mã bao gồm giải mã giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch.

16. Phương pháp xử lý ảnh theo điểm 15, trong đó giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch, bằng cách thực hiện xử lý mã hóa Golomb theo số mũ có dấu, có trong dữ liệu được mã hóa, và bước giải mã bao gồm thực hiện xử lý giải mã Golomb theo số mũ có dấu đối với giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch thu được bằng cách thực hiện mã hóa Golomb theo số mũ có dấu theo thứ tự của giá trị chênh lệch ban đầu và nhóm hệ số chênh lệch.

17. Phương pháp xử lý ảnh theo điểm 16, phương pháp này còn bao gồm: bước biến đổi ngược dữ liệu hệ số biến đổi đã tạo.

18. Phương pháp xử lý ảnh theo điểm 17, trong đó bước biến đổi ngược bao

gồm biến đổi ngược theo bộ biến đổi 16×16 .

19. Vật ghi đọc được bằng máy tính chứa các lệnh đọc được bằng máy tính, mà khi được thực hiện bằng máy tính, khiến máy tính thực hiện phương pháp xử lý ảnh bao gồm các bước:

giải mã dữ liệu được mã hóa bao gồm hệ số chênh lệch thay thế mà là độ chênh lệch giữa hệ số thay thế được sử dụng để thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 mà thu được bằng cách chuyển đổi lên ma trận lượng tử hóa 8×8 theo xử lý nội suy lân cận gần nhất và hệ số $(0, 0)$ của ma trận lượng tử hóa 8×8 để tạo dữ liệu được lượng tử hóa;

thiết đặt ma trận lượng tử hóa 16×16 bằng cách thực hiện xử lý nội suy lân cận gần nhất đối với ma trận lượng tử hóa 8×8 trong đó hệ số thu được bằng cách cộng hệ số thay thế với hệ số chênh lệch thay thế được thiết đặt là số $(0, 0)$ của ma trận lượng tử hóa 8×8 ;

thay thế hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 với hệ số thay thế; và giải lượng tử hóa dữ liệu được lượng tử hóa đã tạo sử dụng ma trận lượng tử hóa 16×16 trong đó hệ số $(0, 0)$ của ma trận lượng tử hóa 16×16 đã được thay thế bằng hệ số thay thế, và tạo dữ liệu hệ số biến đổi.

1/58

FIG. 1

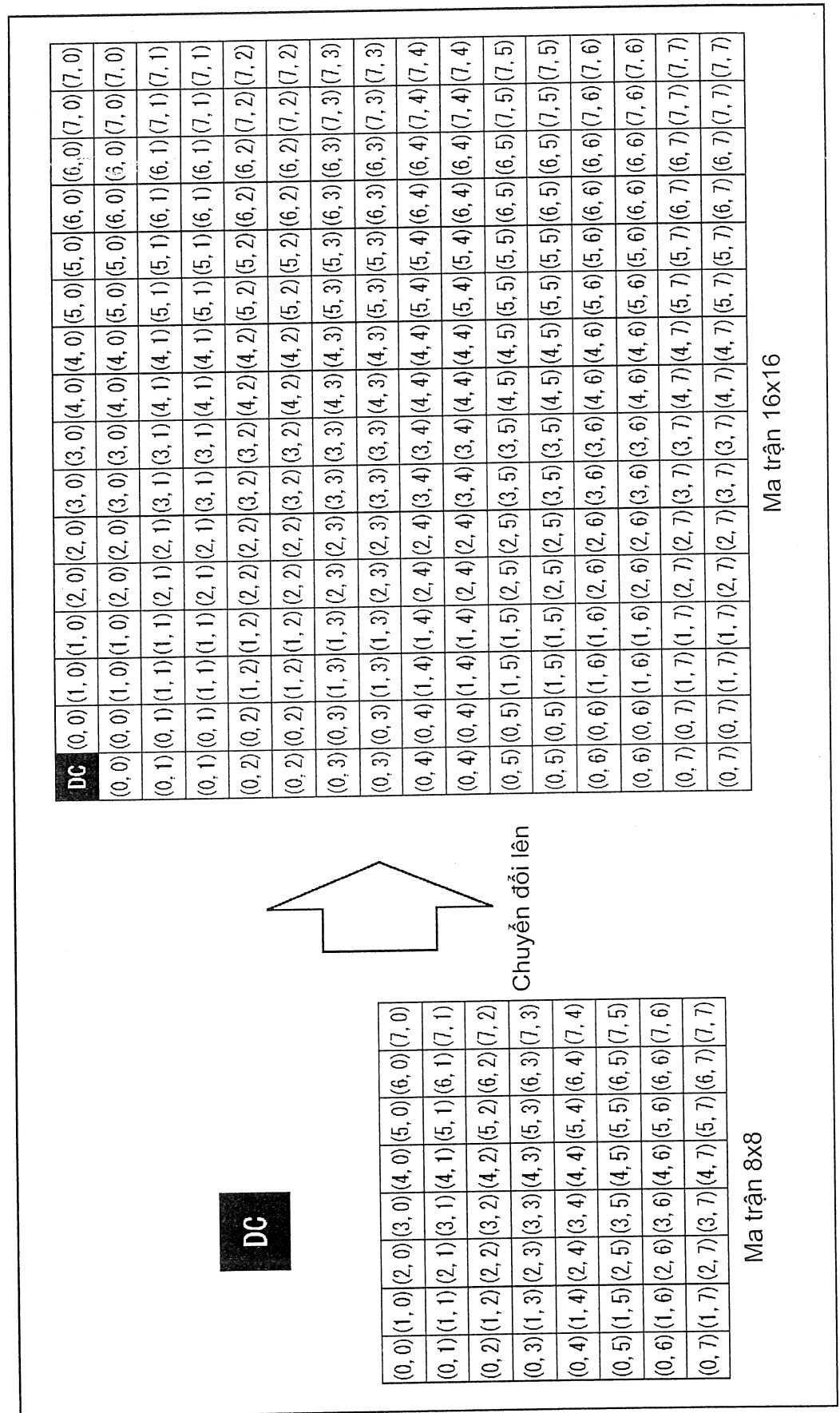
DC=12

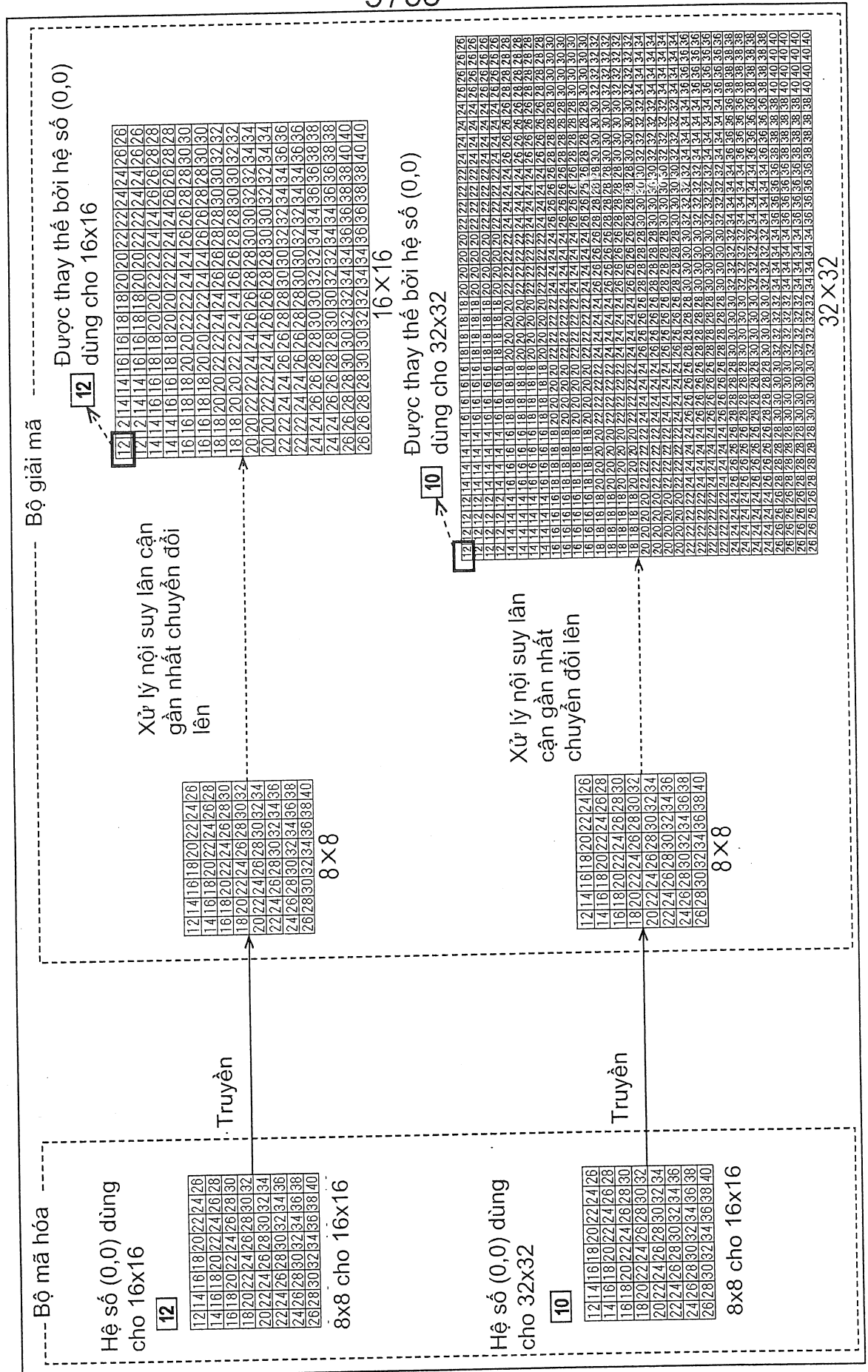
AC=

12	14	16	18	20	22	24	26
14	16	18	20	22	24	26	28
16	18	20	22	24	26	28	30
18	20	22	24	26	28	30	32
20	22	24	26	28	30	32	34
22	24	26	28	30	32	34	36
24	26	28	30	32	34	36	38
26	28	30	32	34	36	38	40

2/58

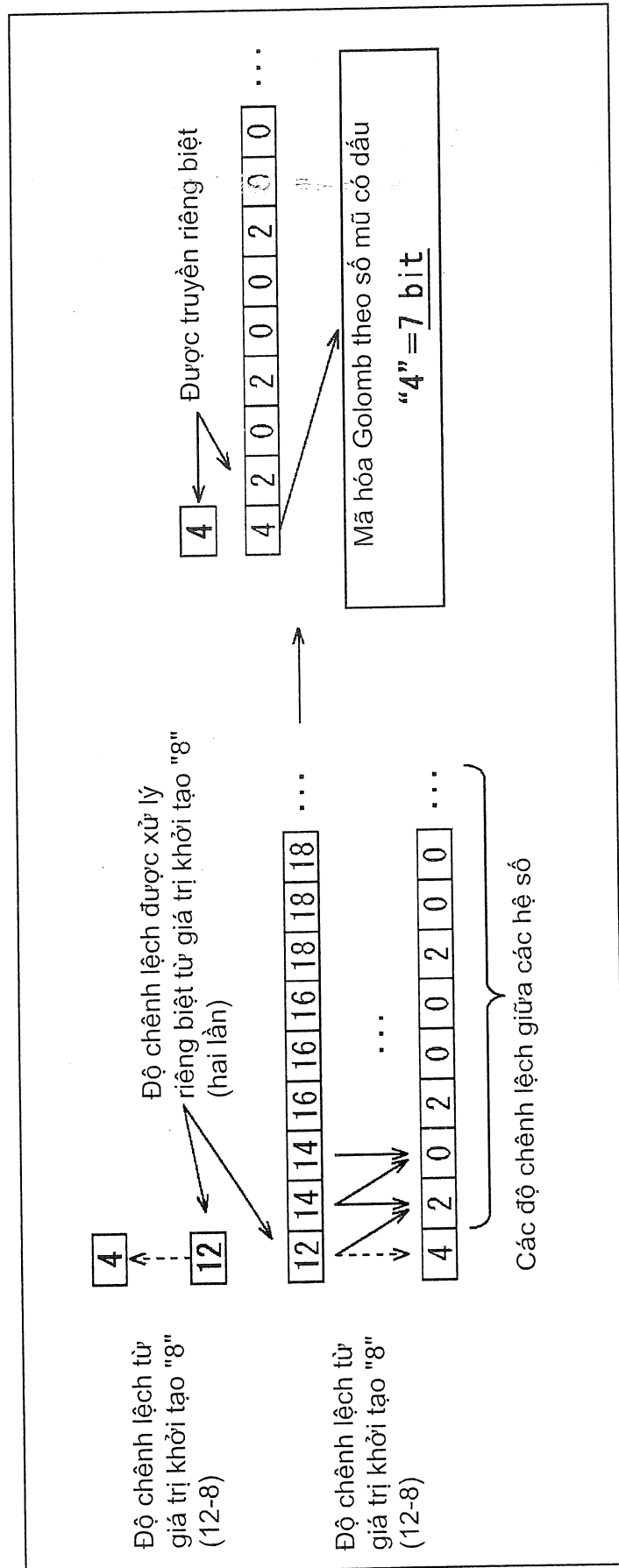
FIG. 2





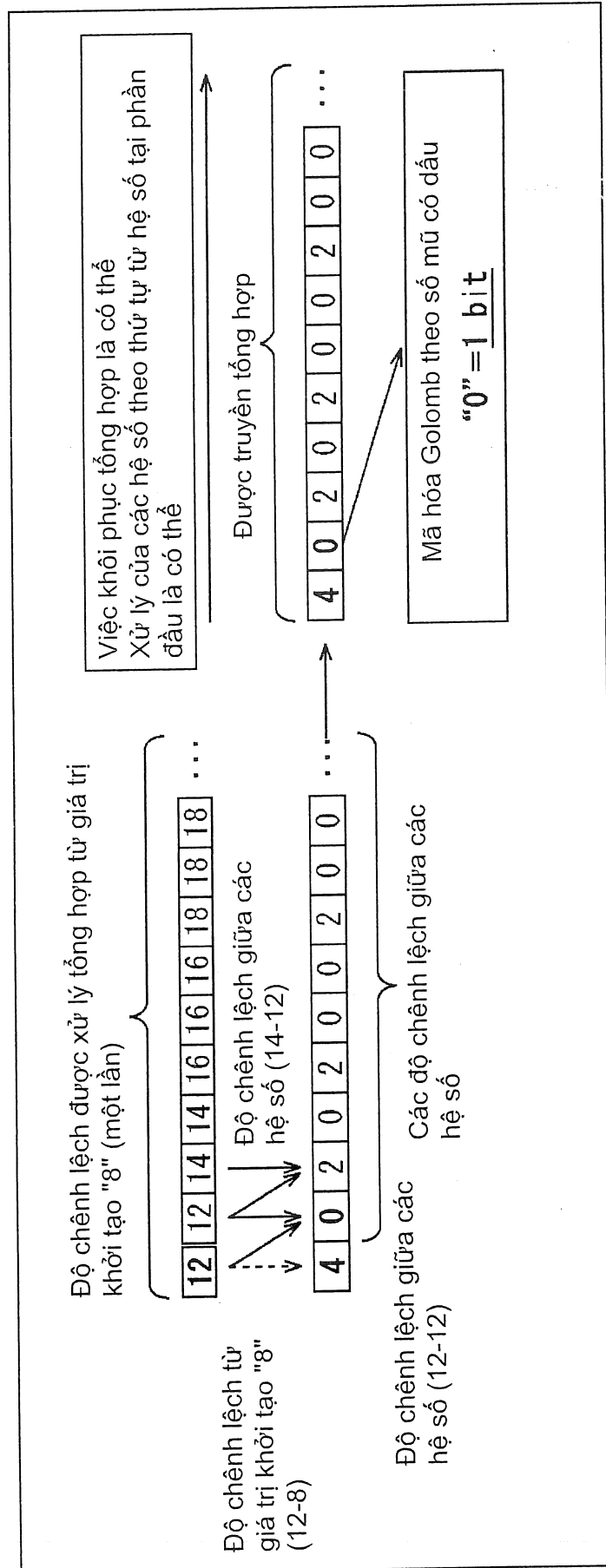
4/58

FIG. 4



5/58

FIG. 5



6/58

FIG. 6

Mã hóa Golomb theo số mũ có dấu

Mã	Ký tự	Độ dài
0	1	1
1	010	3
-1	011	3
2	00100	5
-2	00101	5
3	00110	5
-3	00111	5
4	0001000	7
-4	0001001	7
5	0001010	7
-5	0001011	7
6	0001100	7
-6	0001101	7
7	0001110	7
-7	0001111	7
8	000010000	9
-8	000010001	9
...	...	...

7/58

FIG. 7

	Phản mô tả
scaling_list(ScalingList, sizeID, MatrixID) {	
nextcoef = 8	
coefNum = Min(64, (1 << (4 + (sizeId << 1))))	
for (i=0; i < coefNum, i++) {	
scaling_list_delta_coef	se (v)
nextcoef = (nextcoef + scaling_list_delta_coef + 256) % 256	
ScalingList[sizeId][matrixId][i] = nextcoef	
}	
if (sizeId > 1)	
scaling_list_dc_coef_minus8 [sizeId-2][matrixId]	se (v)
}	

A

nextCoef = 8	
coefNum = Min(64, (1 << (4 + (sizeId << 1))))	
if (sizeId > 1) {	
scaling_list_dc_coef_minus8 [sizeId-2][matrixId]	se (v)
nextCoef =	
scaling_list_dc_coef_minus8[sizeId-2][matrixId] + 8	
}	
for (i = 0; i < coefNum; i++) {	se (v)
scaling_list_delta_coef	
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
ScalingList[sizeId][matrixId][i] = nextCoef	
}	

B

8/58

FIG. 8

	Phần mô tả
scaling_list(scalingList, sizeID, matrixID) {	u(1)
nextCoef = 8	
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
if(scaling_list_dc_coef_minus8[sizeID - 2][matrixID] == -8)	
UseDefaultScalingMatrixFlag = 1	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
stopNow = 0	
for(i=0; i < coefNum && !stopNow; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
if(sizeID < 2) {	
useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)	
if(useDefaultScalingMatrixFlag)	
stopNow = 1	
}	
if(!stopNow)	
scalingList[i] = nextCoef	
}	
}	
}	

9/58

FIG. 9

scaling_list_pred_matrix_id_delta chỉ rõ ma trận tham chiếu đích để sao chép giá trị của danh sách chia tỷ lệ. Khi scaling_list_pred_mode_flag bằng 0, scaling_list_pred_matrix_id_delta chỉ rõ ma trận nào được sử dụng trong ma trận hiện tại bằng phần sau đây:

$$\text{RefMatrixID} = \text{MatrixID} - (1 + \text{scaling_list_pred_matrix_id_delta})$$

A

Nếu scaling_list_pred_matrix_id_delta bằng 0, danh sách chia tỷ lệ được suy ra từ danh sách chia tỷ lệ mặc định ScalingList[sizeId][matrixId][i] như được chỉ rõ trong bảng 7-5 và bảng 7-6 đối với i=0 .. Min(64, (1 << (4 + (sizeId << 1)))).

- Ngoài ra, danh sách chia tỷ lệ được suy ra từ danh sách chia tỷ lệ tham chiếu như sau

$$\text{refMatrixId} = \text{matrixId} - \text{scaling_list_pred_matrix_id_delta}[\text{sizeId}][\text{matrixId}]$$

B

10/58

FIG. 10

scaling_list_param() {	Phần mô tả
scaling_list_present_flag	u(1)
if (scaling_list_present_flag)	
for (sizeID=0; sizeID<4; sizeID++)	
for (matrixID=0; matrixID<(sizeID==3)?2:6; matrixID++) {	
scaling_list_pred_mode_flag	u(1)
if (!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list (ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	

A

scaling_list(scalingList, sizeID, matrixID) {	Phần mô tả
nextCoef=8	u(1)
coefNum=Min(64, (1<<(4+(sizeID<<1))))	
UseDefaultScalingMatrix=0	
if (sizeID>1) {	
scaling_list_dc_coef_minus8 [sizeID-2][matrixID]	se(v)
if (scaling_list_dc_coef_minus8[sizeID-2][matrixID]==-8)	
UseDefaultScalingMatrixFlag=1	
}	
if (UseDefaultScalingMatrixFlag==0) {	
stopNow=0	
for (i=0; i<coefNum && !stopNow; i++) {	
scaling_list_delta_coef	se(v)
nextCoef=(nextCoef+scaling_list_delta_coef+256)%256	
if (sizeID<2) {	
useDefaultScalingMatrixFlag=(i==0 && nextCoef==0)	
if (useDefaultScalingMatrixFlag)	
stopNow=1	
}	
if (!stopNow)	
scalingList[i]=nextCoef	
}	
}	

B

11/58

FIG. 11

scaling_list_data() {	Phần mô tả
for (sizeId=0; sizeId<4; sizeId++)	
for (matrixId=0; matrixId<(sizeId==3)?2:6; matrixId++) {	
scaling_list_pred_mode_flag[sizeId][matrixId]	u(1)
if (!scaling_list_pred_mode_flag[sizeId][matrixId])	
scaling_list_pred_matrix_id_delta[sizeId][matrixId]	ue(1)
else {	
nextCoef=8	
coefNum=Min(64, (1<<(4+(sizeId<<1))))	
if (sizeId>1) {	
scaling_list_dc_coef_minus8[sizeId-2][matrixId]	se(v)
nextCoef=	
scaling_list_dc_coef_minus8[sizeId-2][matrixId]+8	
}	
for (i=0; i<coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef=(nextCoef+scaling_list_delta_coef+256)%256	
ScalingList[sizeId][matrixId][i]=nextCoef	
}	
}	
}	

FIG. 12

12 / 58

scaling_list_param() {	Phần mô tả
scaling_list_present_flag	u(l)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID ++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID ++) {	
scaling_list_pred_mode_flag	u(l)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	

A

scaling_list(scalingList, sizeID, matrixID) {	Phần mô tả
nextCoef = 8	u(l)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
if(scaling_list_dc_coef_minus8[sizeID - 2][matrixID] == -8)	
UseDefaultScalingMatrixFlag = 1	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
for(i = 0; i < coefNum, i ++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
if(sizeID < 2) {	
useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)	
if(useDefaultScalingMatrixFlag)	
break	
}	
scalingList[i] = nextCoef	
}	
}	
}	

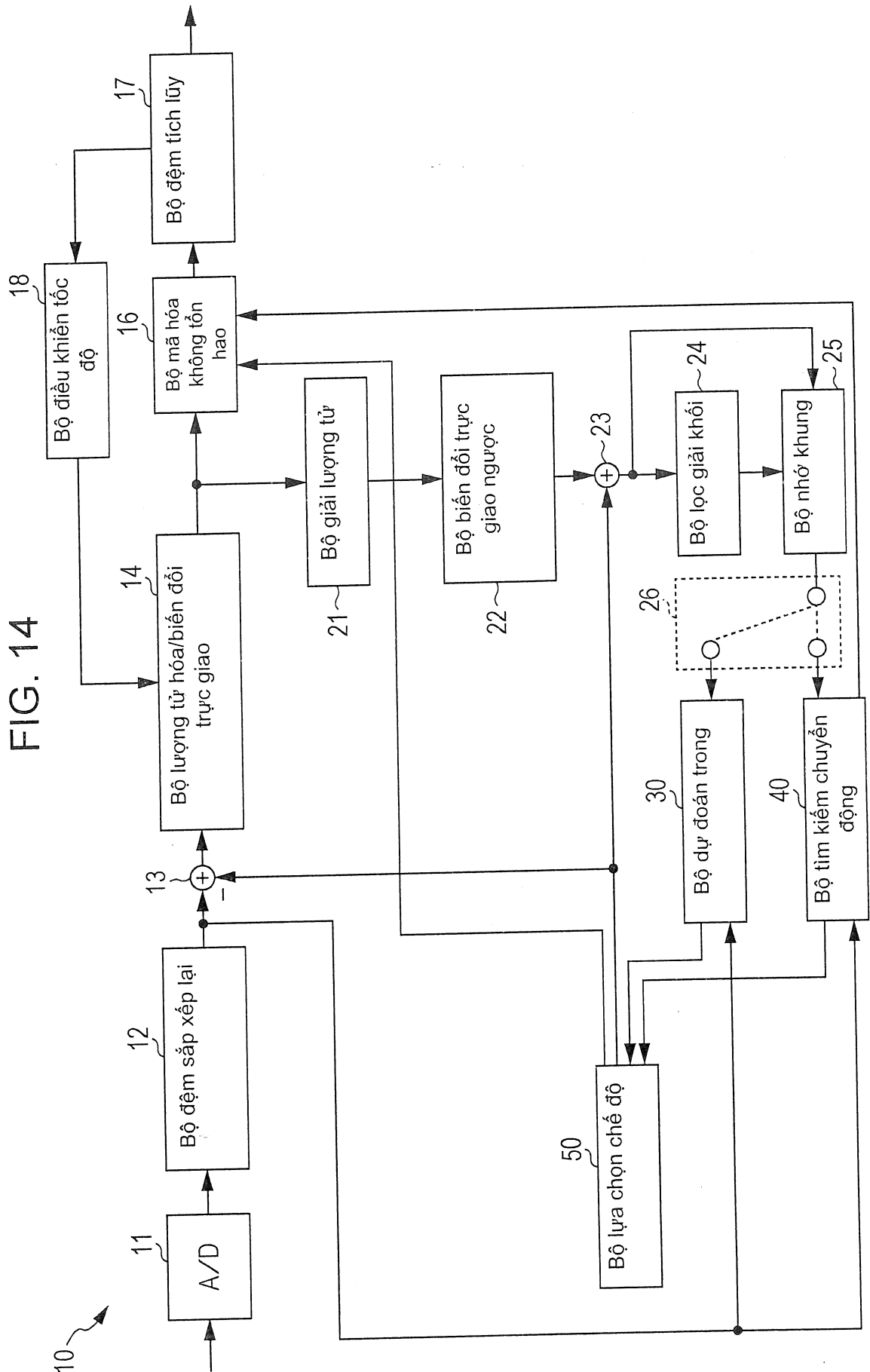
B

13/58

FIG. 13

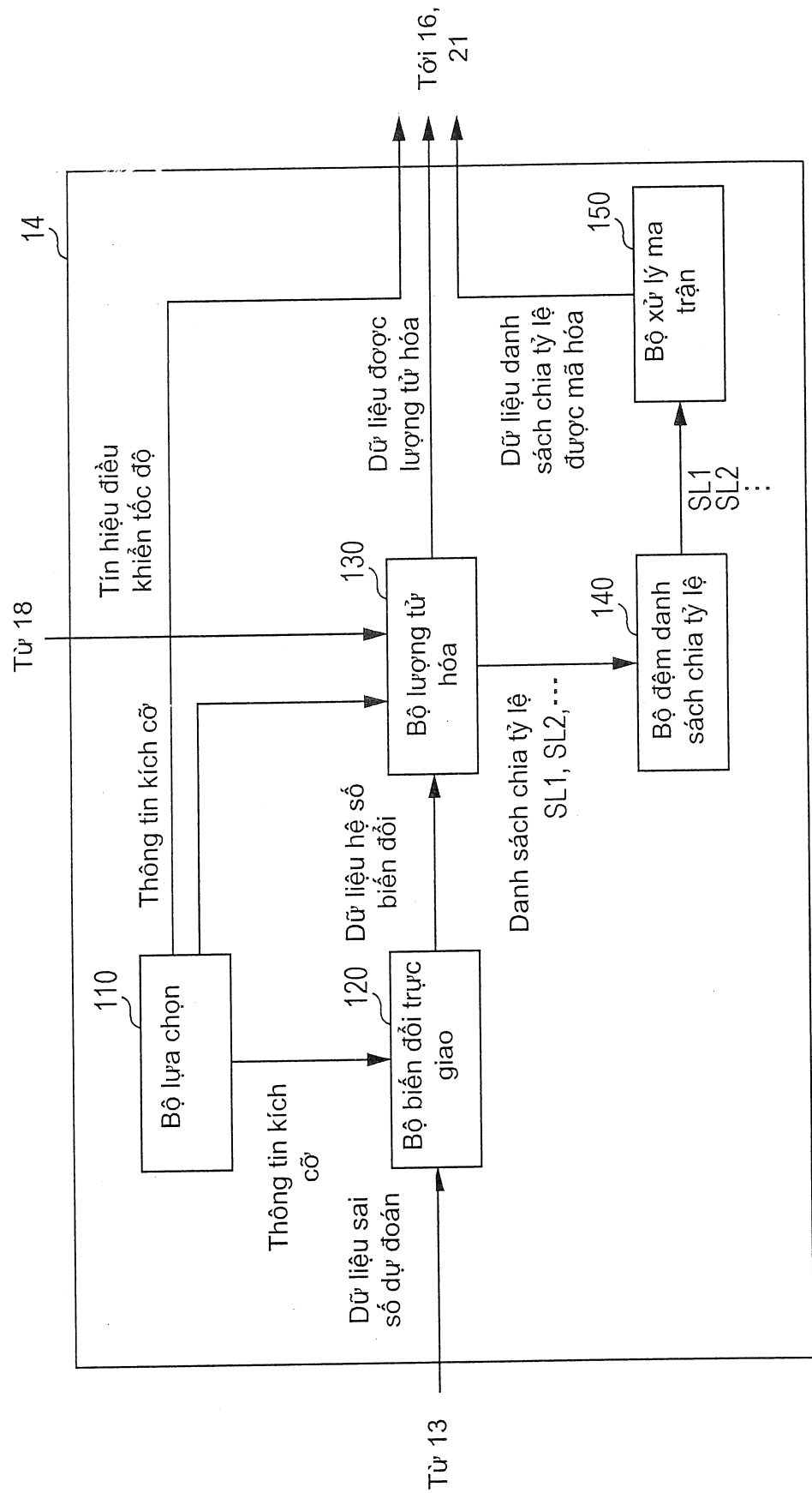
scaling_list(ScalingList, sizeID, MatrixID) {	Phần mô tả
nextcoef=8	
coefNum=min((1<<(4+(sizeID<<1))), 65)	
for(i=0; i<coefNum, i++) {	
scaling_list_delta_coef	se(v)
nextcoef=(nextcoef+scaling_list_delta_coef+256) % 256	
if(sizeID>1 && i==0)	
scaling_list_dc_coef[sizeID-2][MatrixID]=nextcoef	
else	
ScalingList[i-(sizeID>1)?1:0]=nextcoef	
}	
}	

14/58



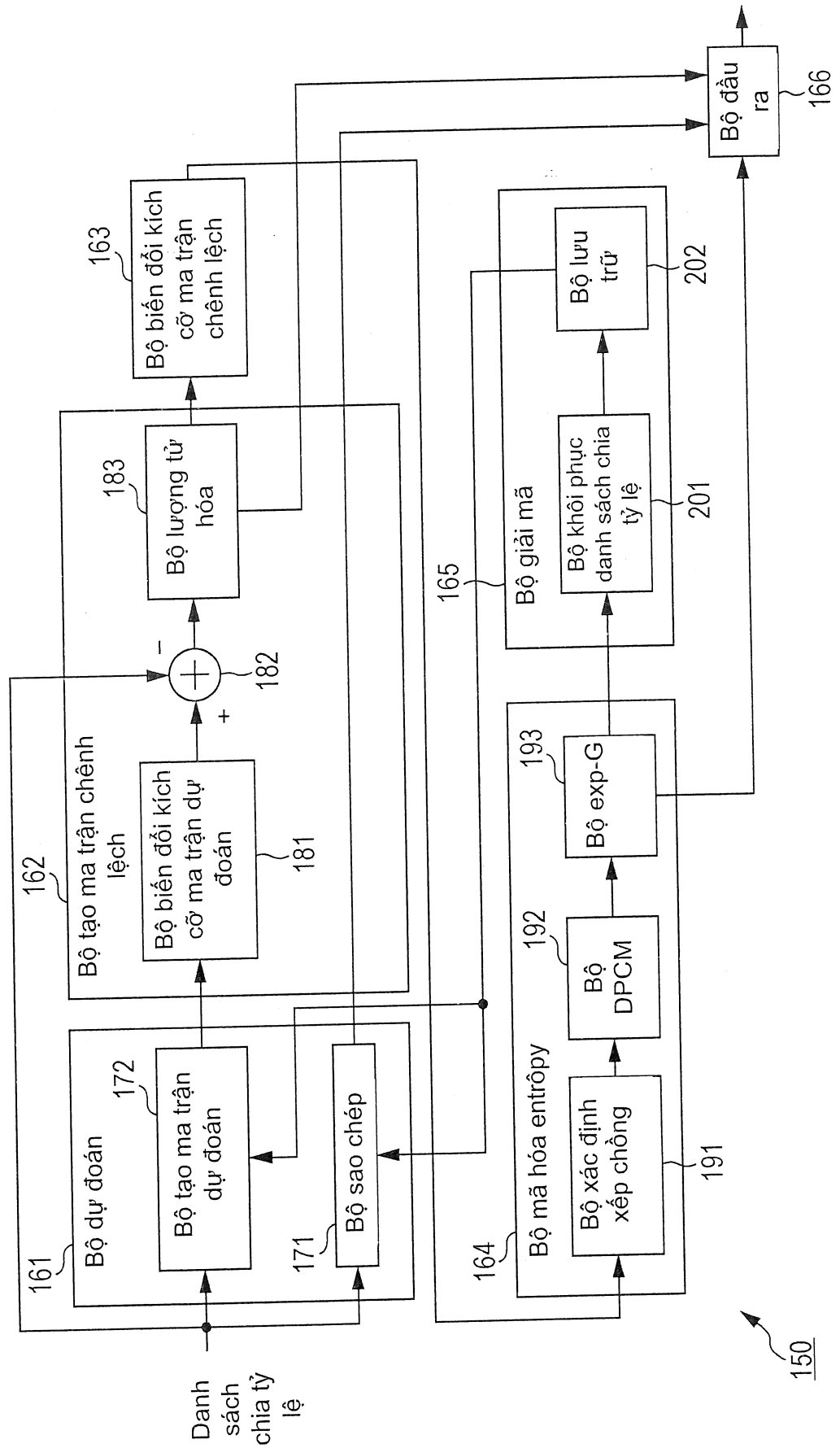
15/58

FIG. 15



16/58

FIG. 16



17/58

FIG. 17

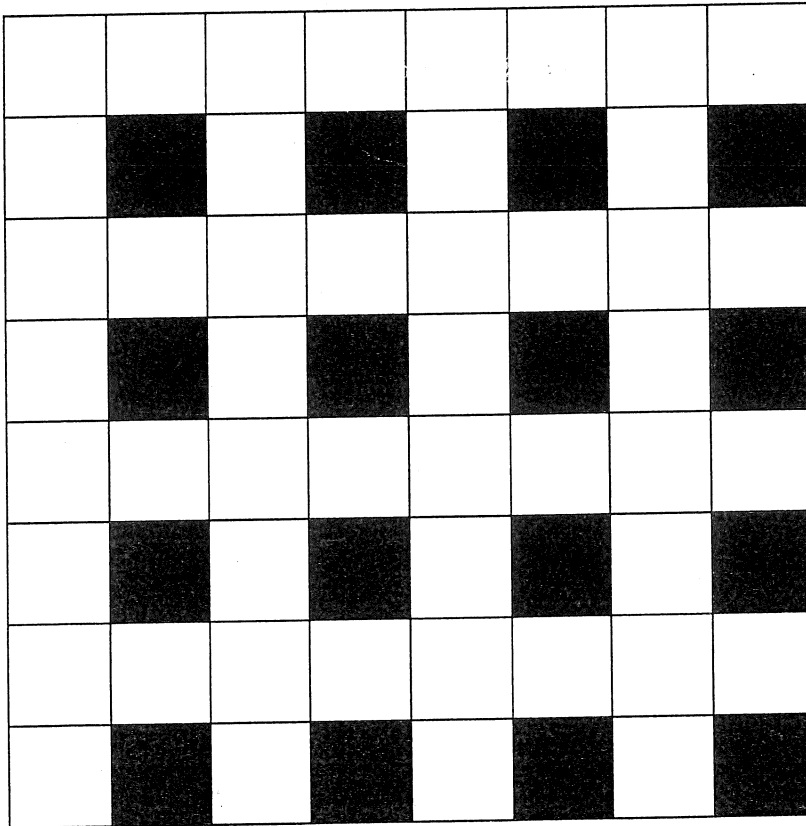


FIG. 18

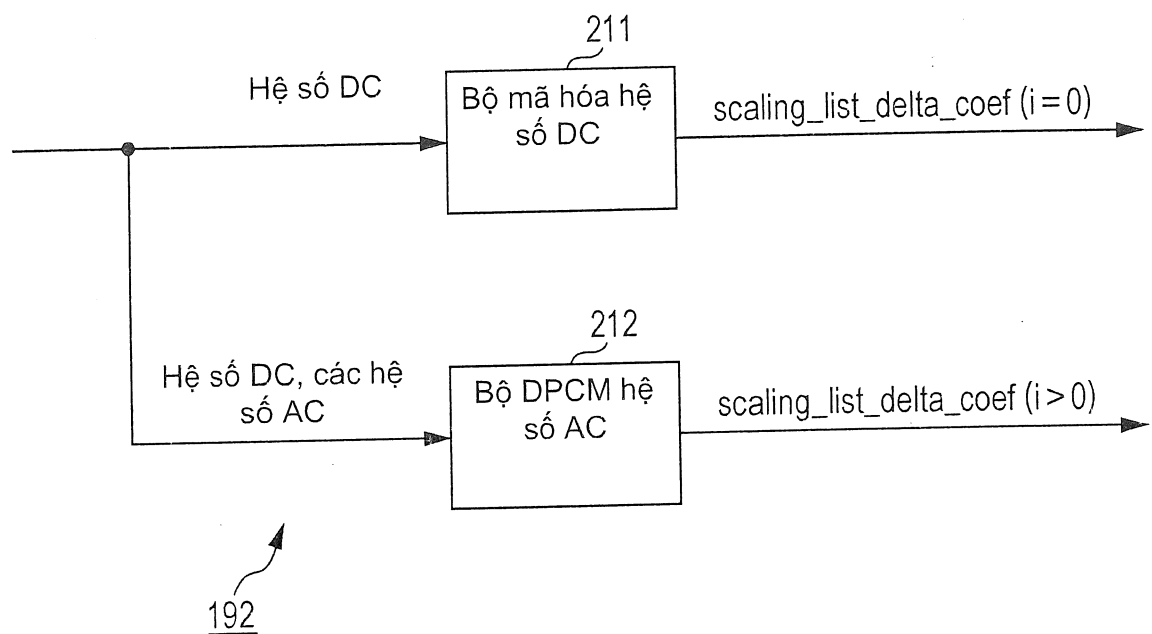
1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10
4	5	6	7	8	9	10	11
5	6	7	8	9	10	11	12
6	7	8	9	10	11	12	13
7	8	9	10	11	12	13	14
8	9	10	11	12	13	14	15



1	2	3	4	5	6	7	8
	3	4	5	6	7	8	9
		5	6	7	8	9	10
			7	8	9	10	11
				9	10	11	12
					11	12	13
						13	14
							15

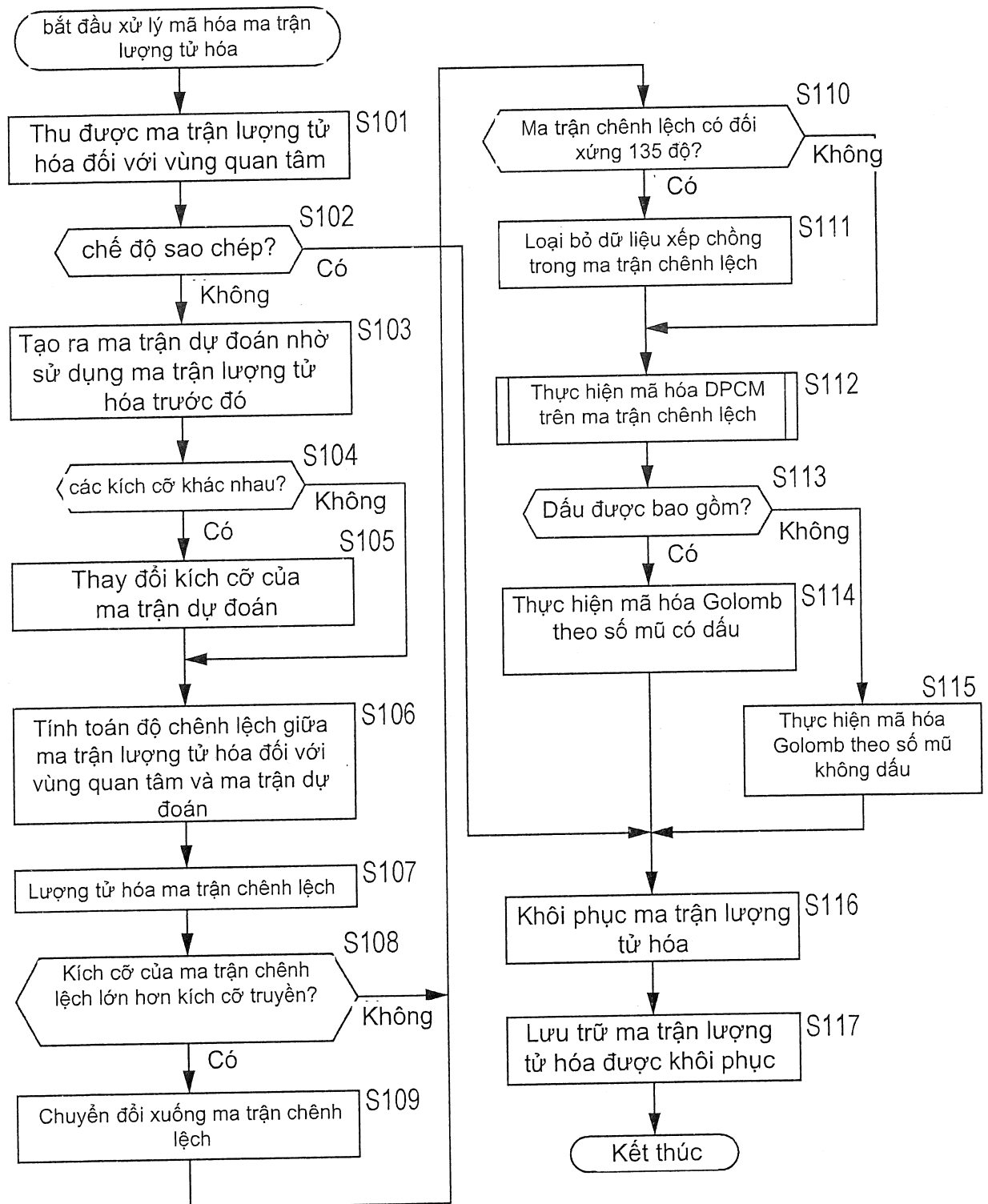
18 / 58

FIG. 19



19/58

FIG. 20



20 / 58

FIG. 21

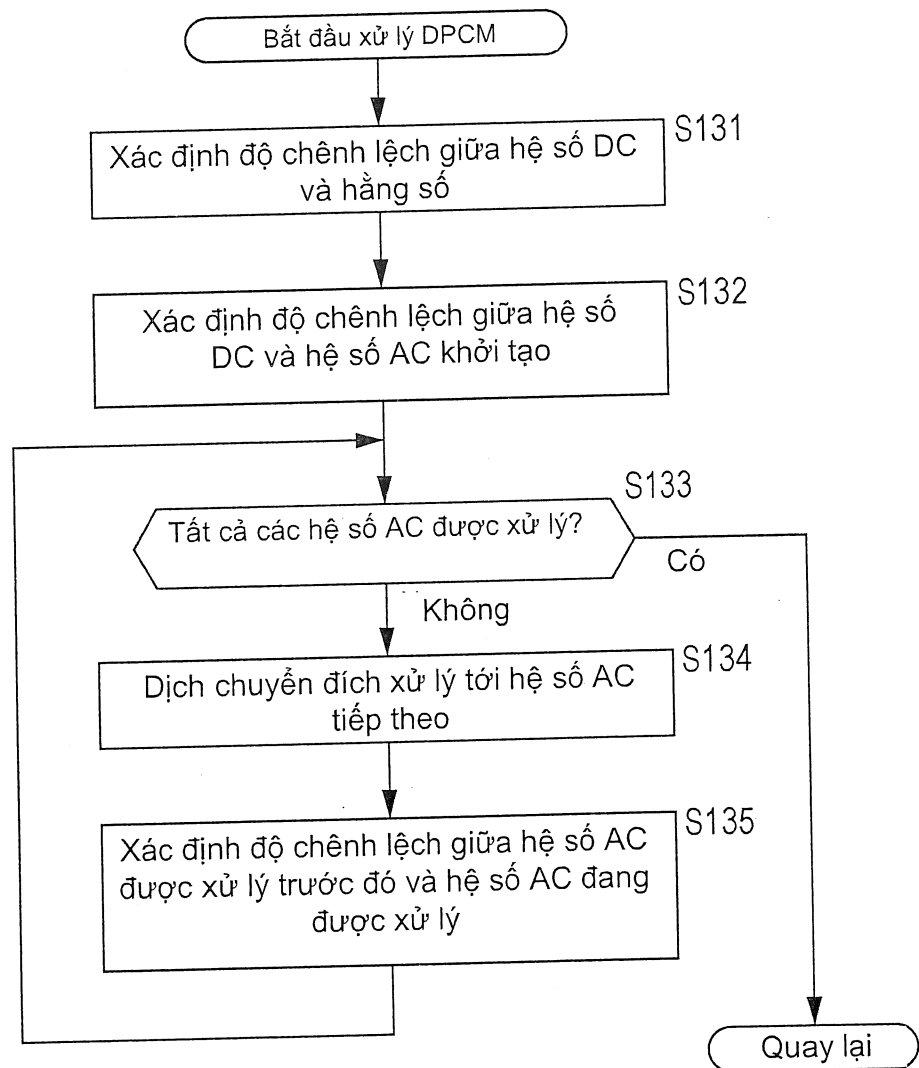
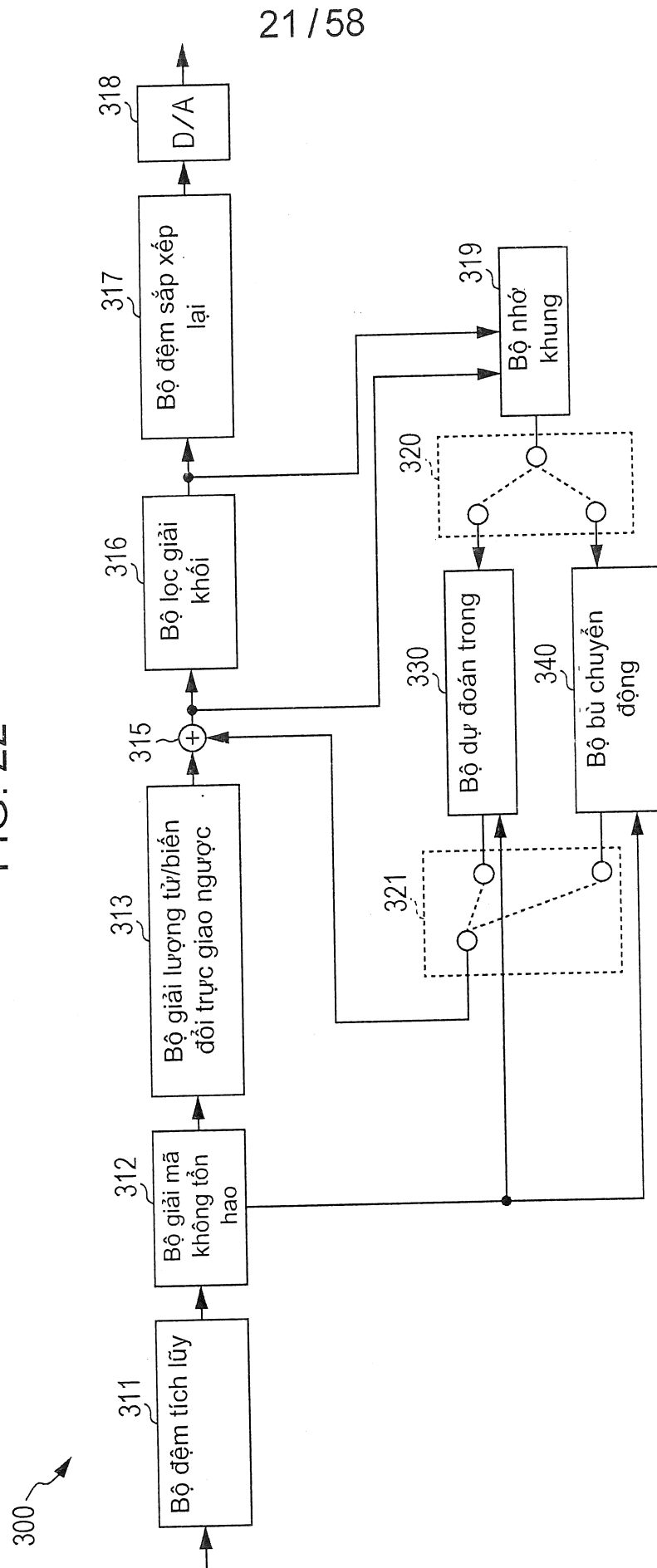
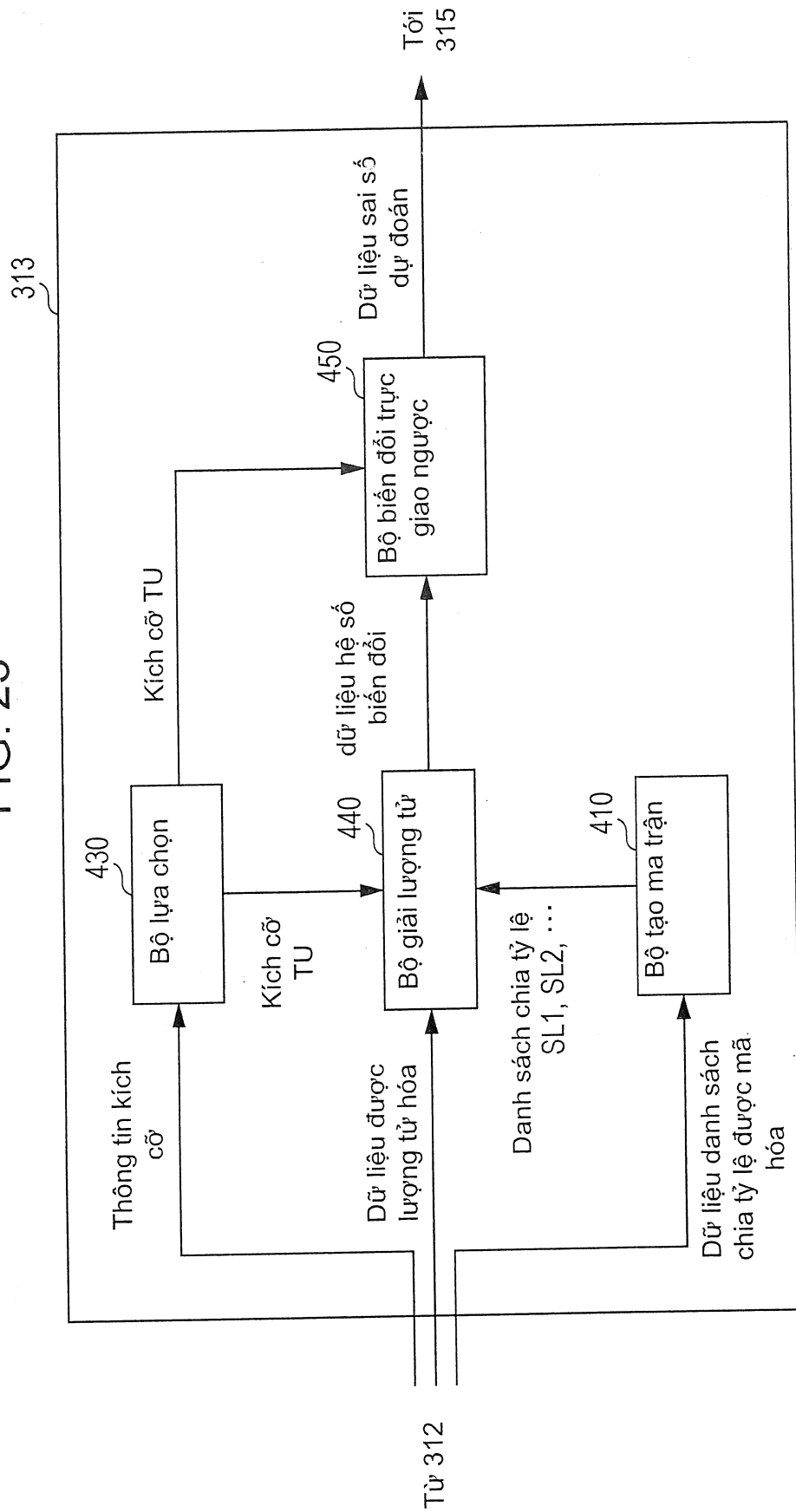


FIG. 22



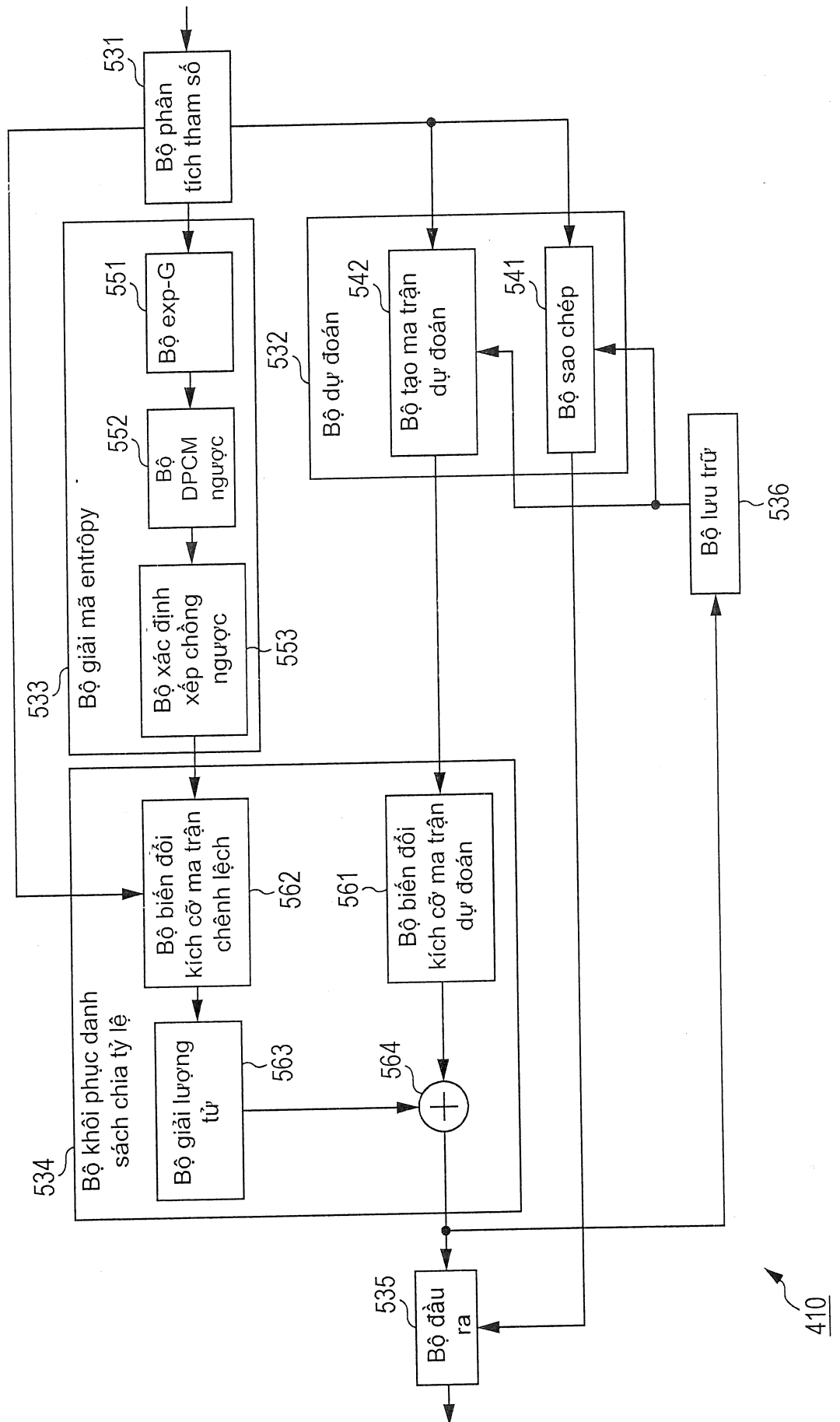
22/58

FIG. 23



23/58

FIG. 24



24/58

FIG. 25

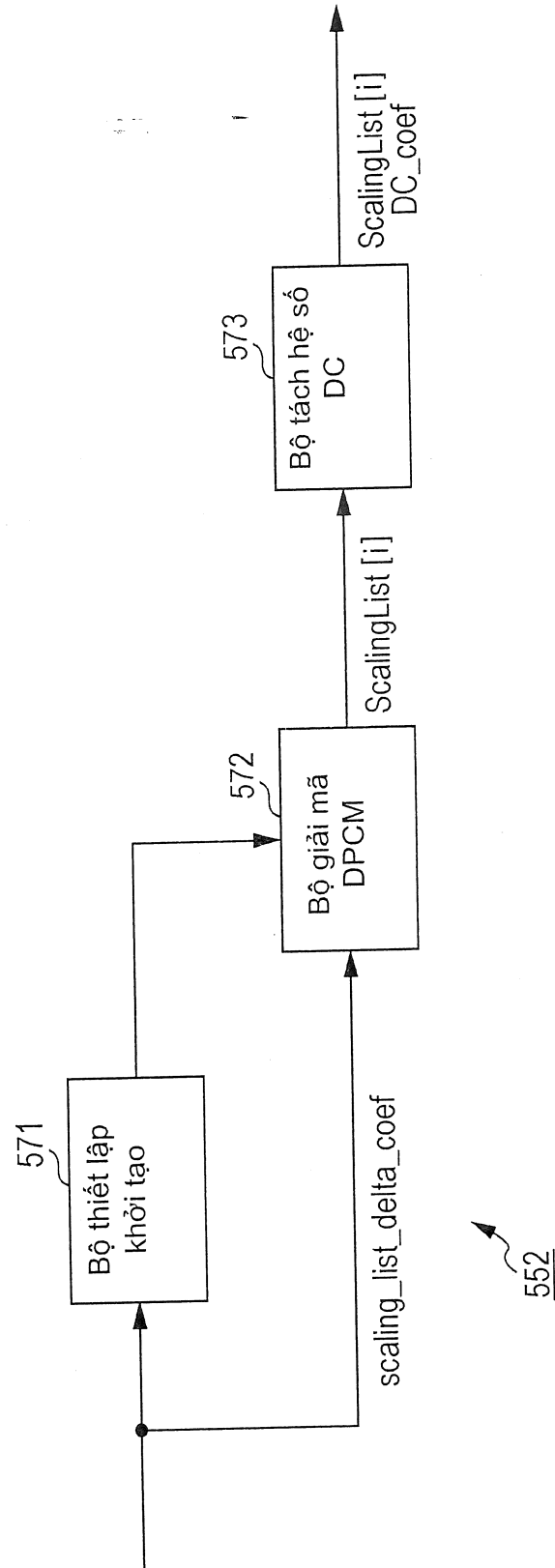
1		2		3		4
5		6		7		8
9		10		11		12
13		14		15		16



1	1	2	2	3	3	4	4
1	1	2	2	3	3	4	4
5	5	6	6	7	7	8	8
5	5	6	6	7	7	8	8
9	9	10	10	11	11	12	12
9	9	10	10	11	11	12	12
13	13	14	14	15	15	16	16
13	13	14	14	15	15	16	16

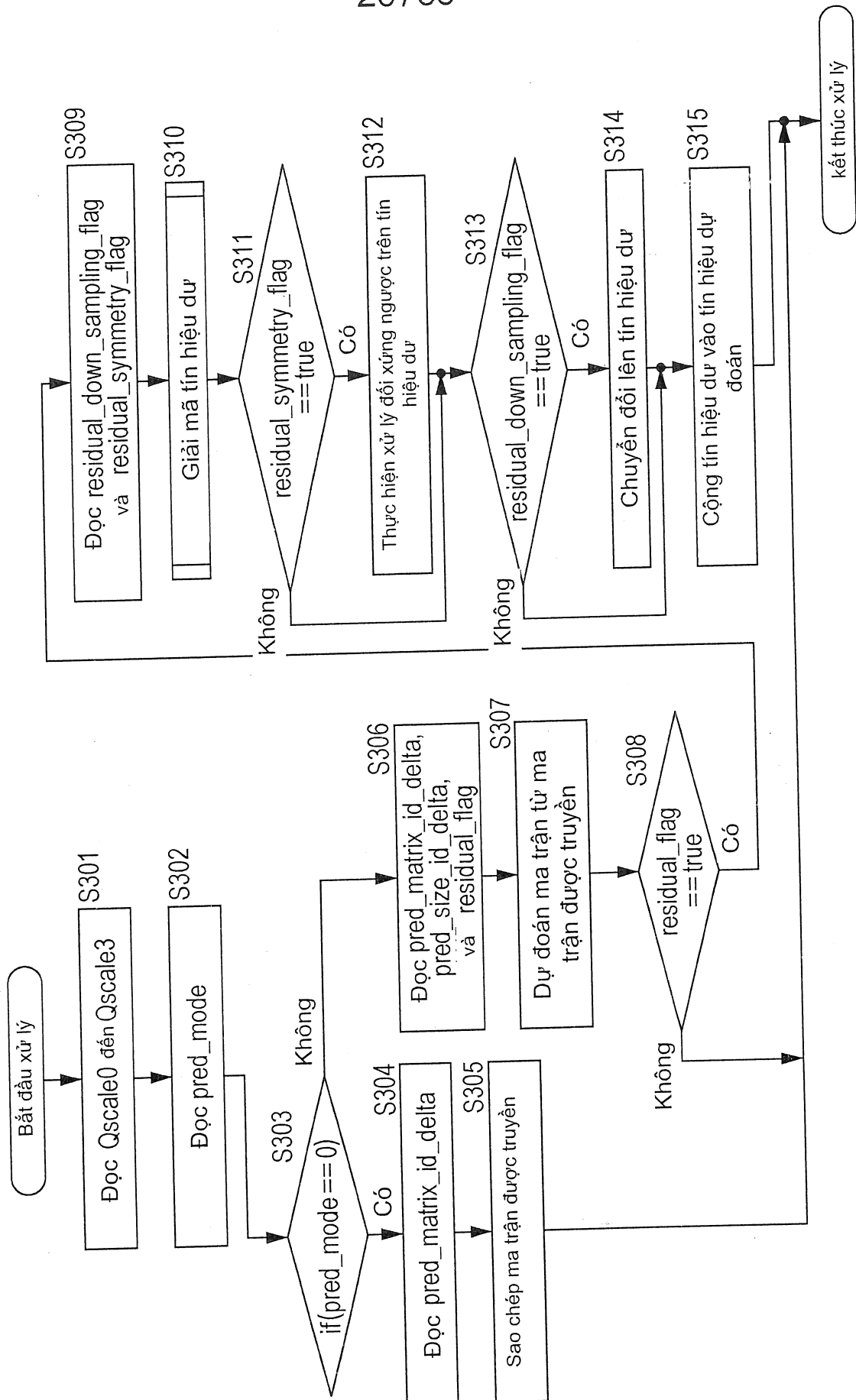
25/58

FIG. 26



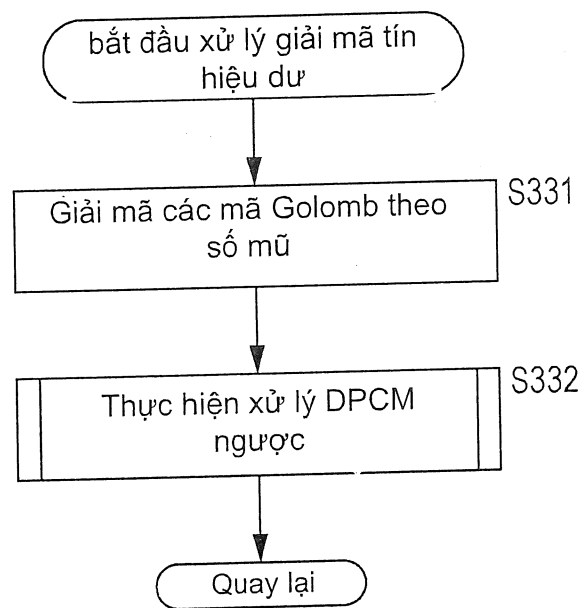
26/58

FIG. 27



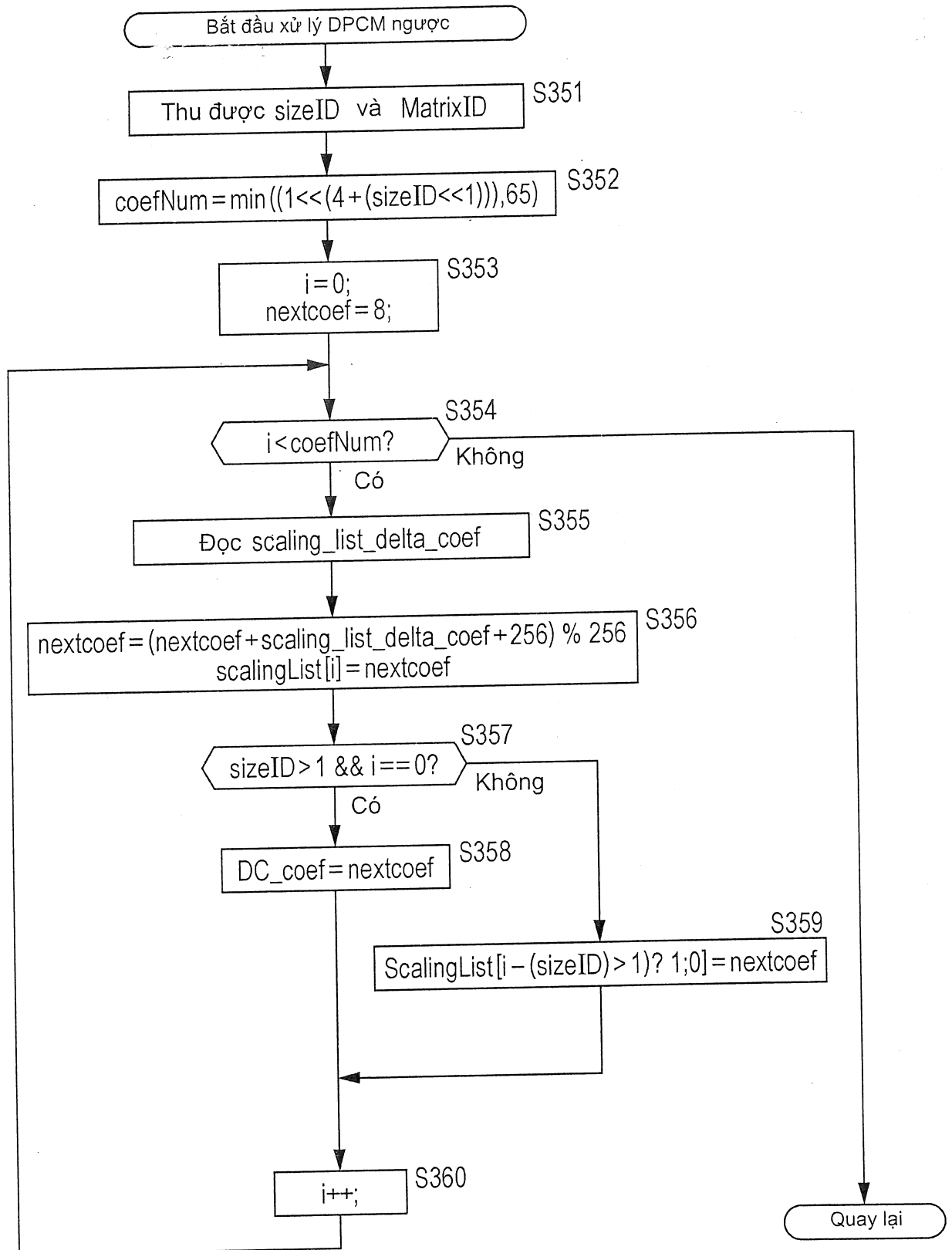
27/58

FIG. 28



28 / 58

FIG. 29



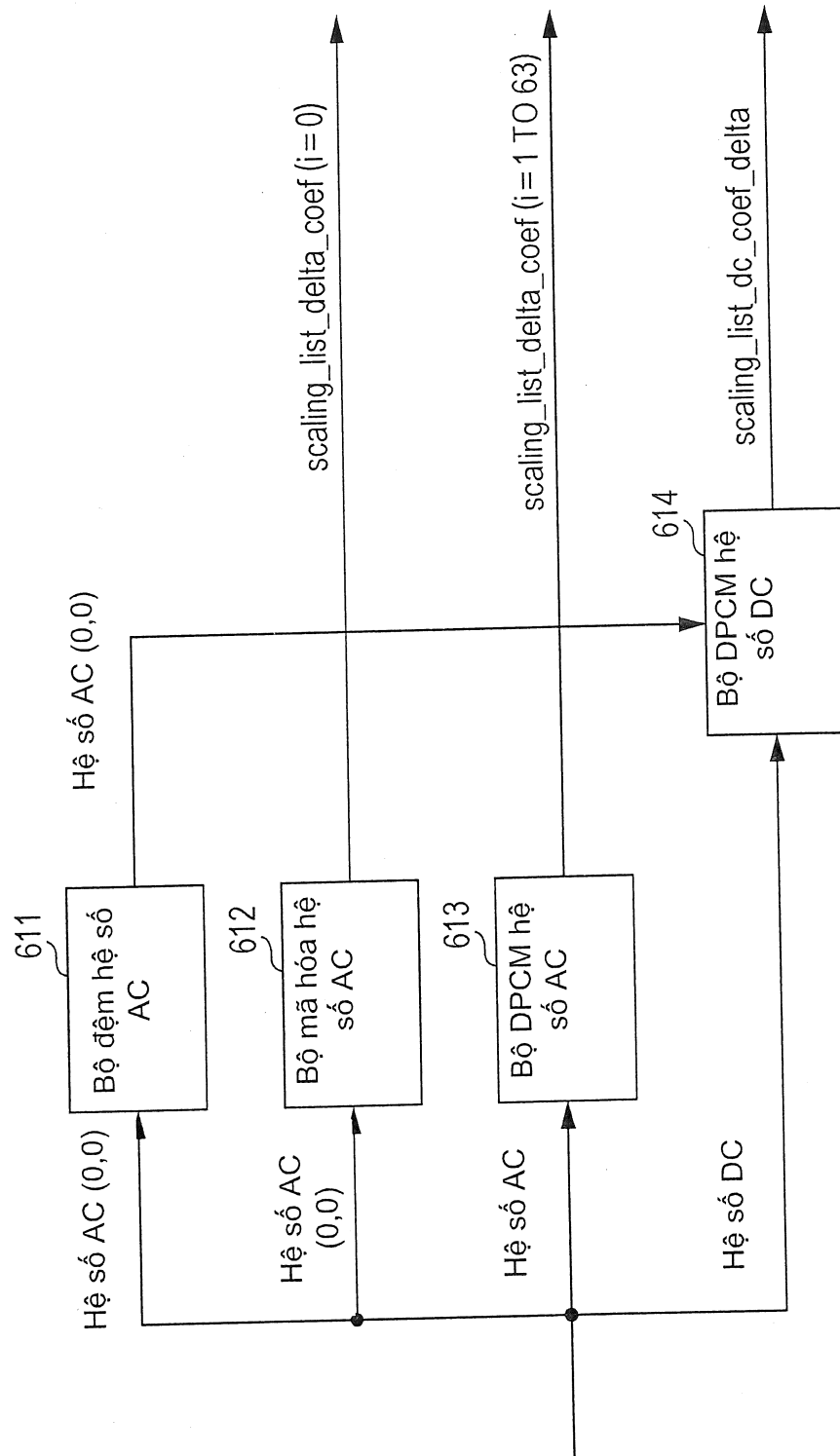
29/58

FIG. 30

	Phần mô tả
scaling_list(ScalingList, sizeID, MatrixID) {	
nextcoef = 8	
coefNum = min((1 << (4 + (sizeID << 1))), 64)	
for(i = 0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextcoef = (nextcoef + scaling_list_delta_coef + 256) % 256	
ScalingList[i] = nextcoef	
}	
if(sizeID > 1)	
scaling_list_dc_coef_delta[sizeID - 2][MatrixID]	se(v)
}	

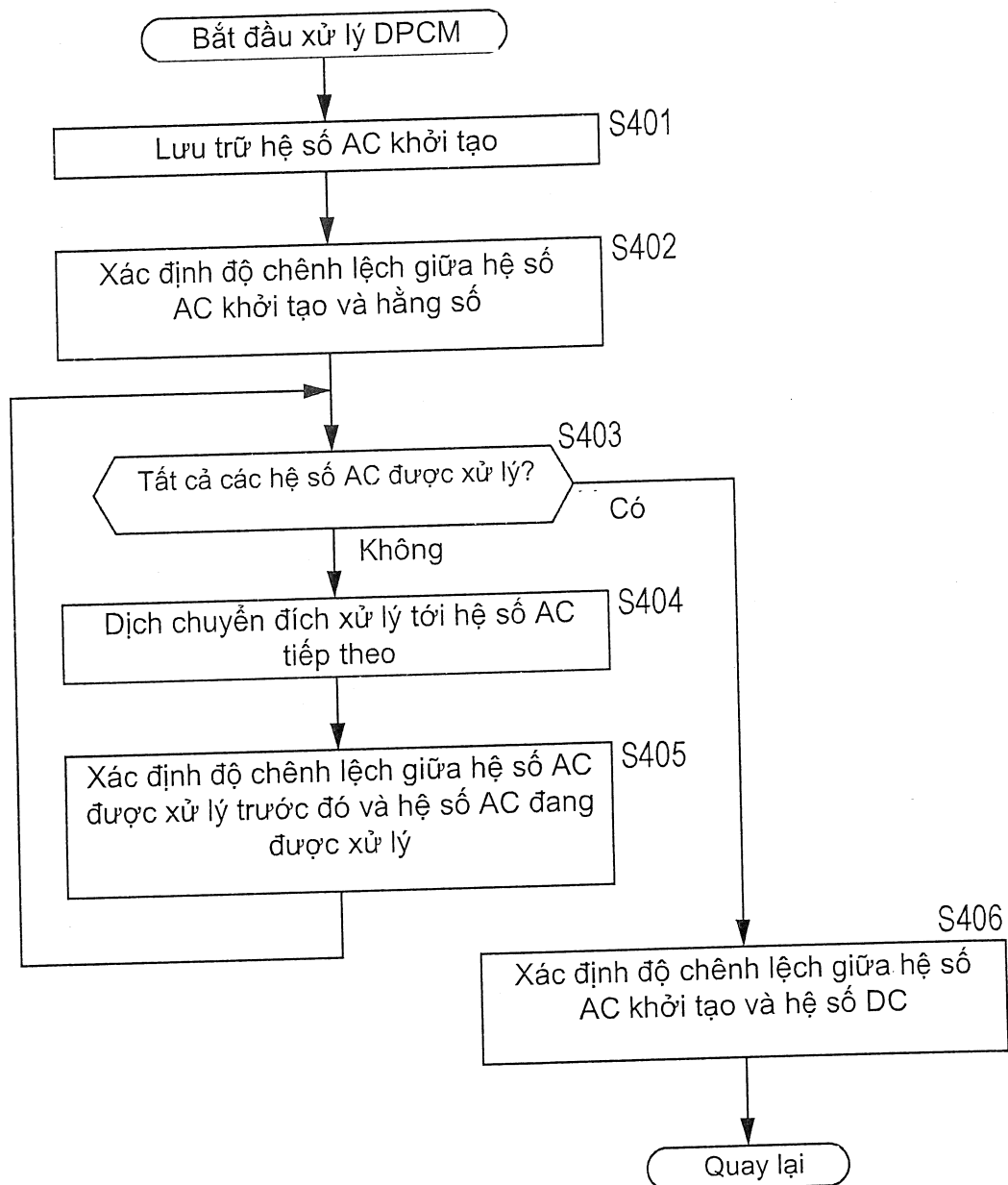
30/58

FIG. 31



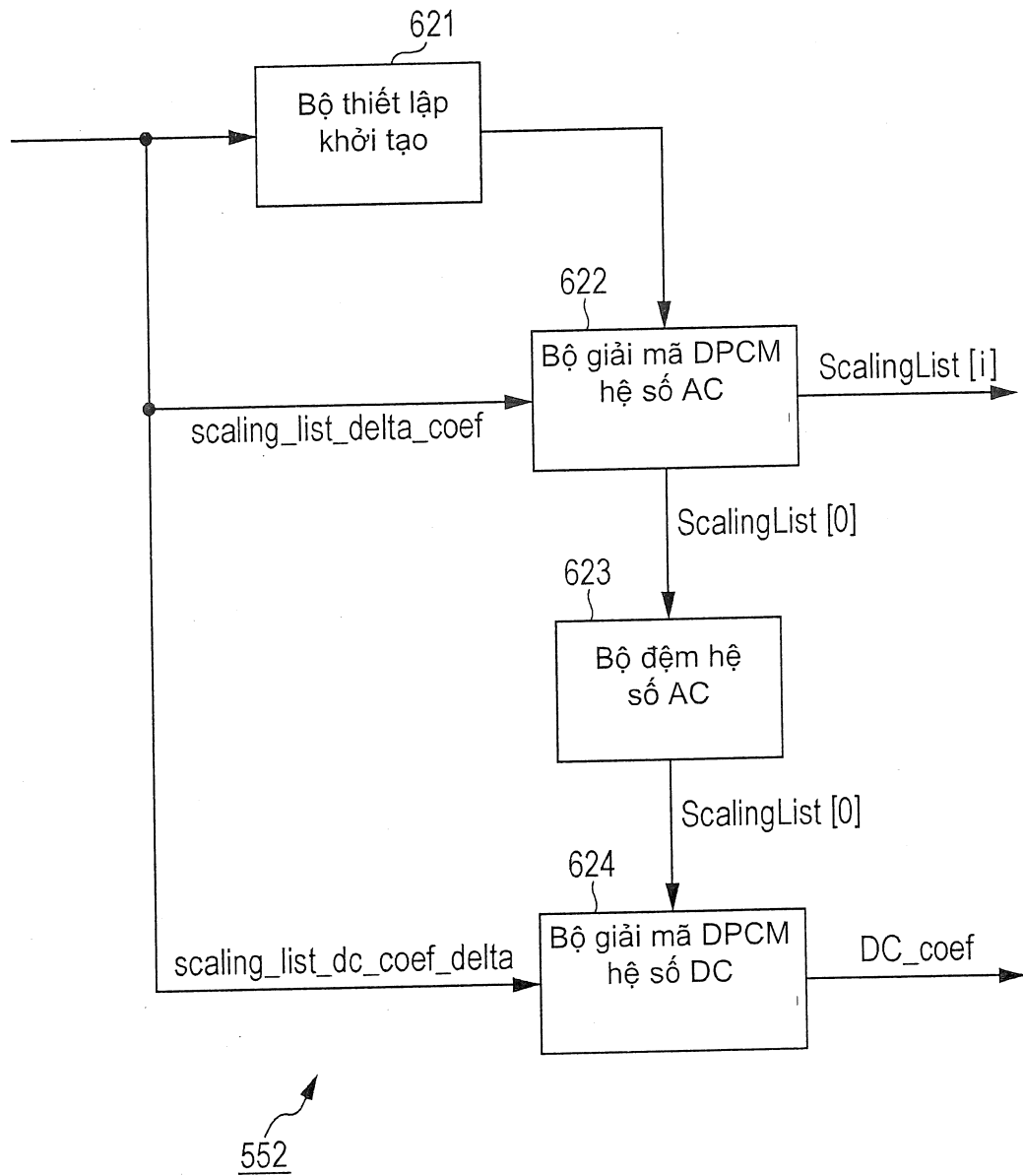
31 / 58

FIG. 32



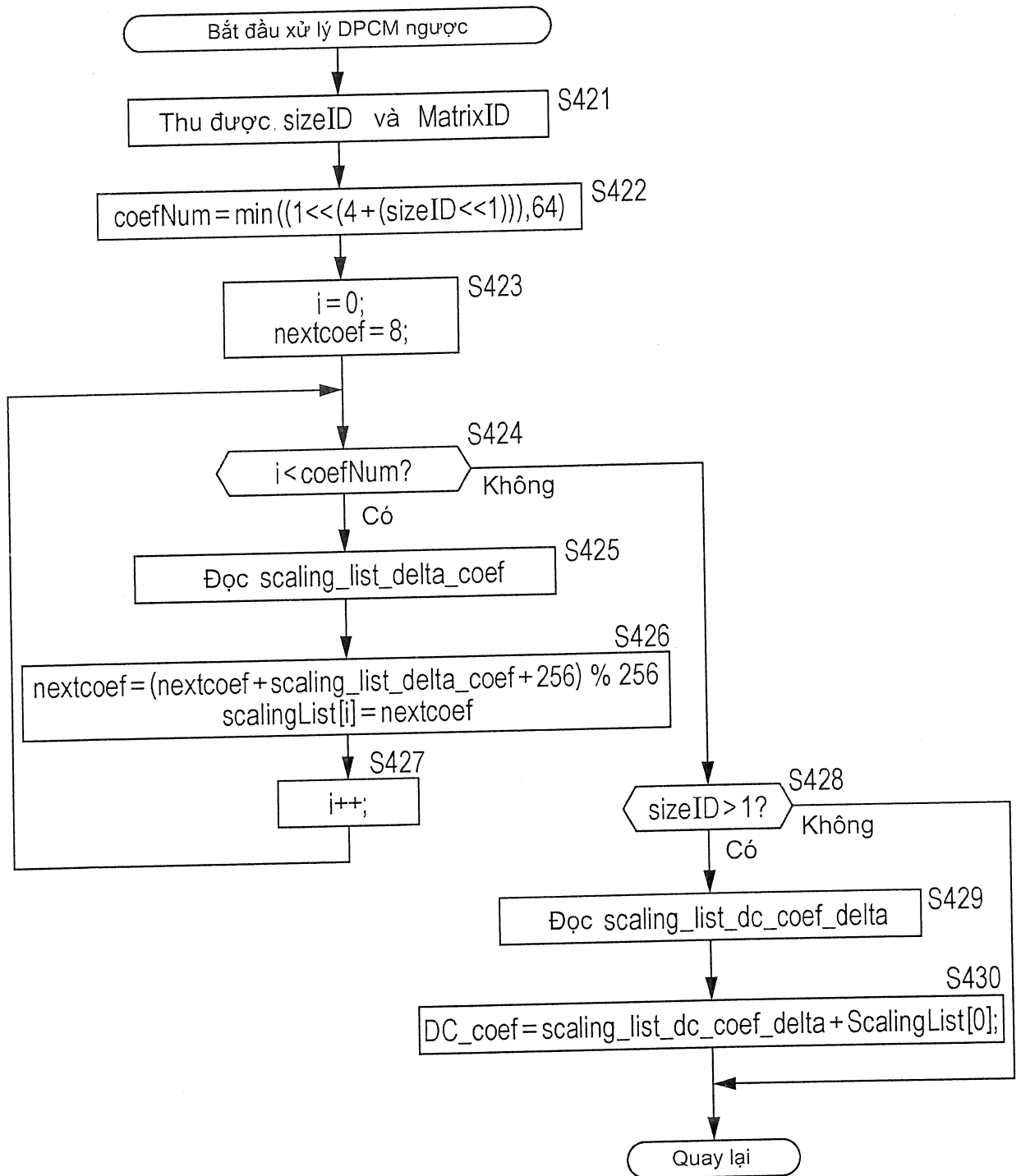
32 / 58

FIG. 33



33 / 58

FIG. 34



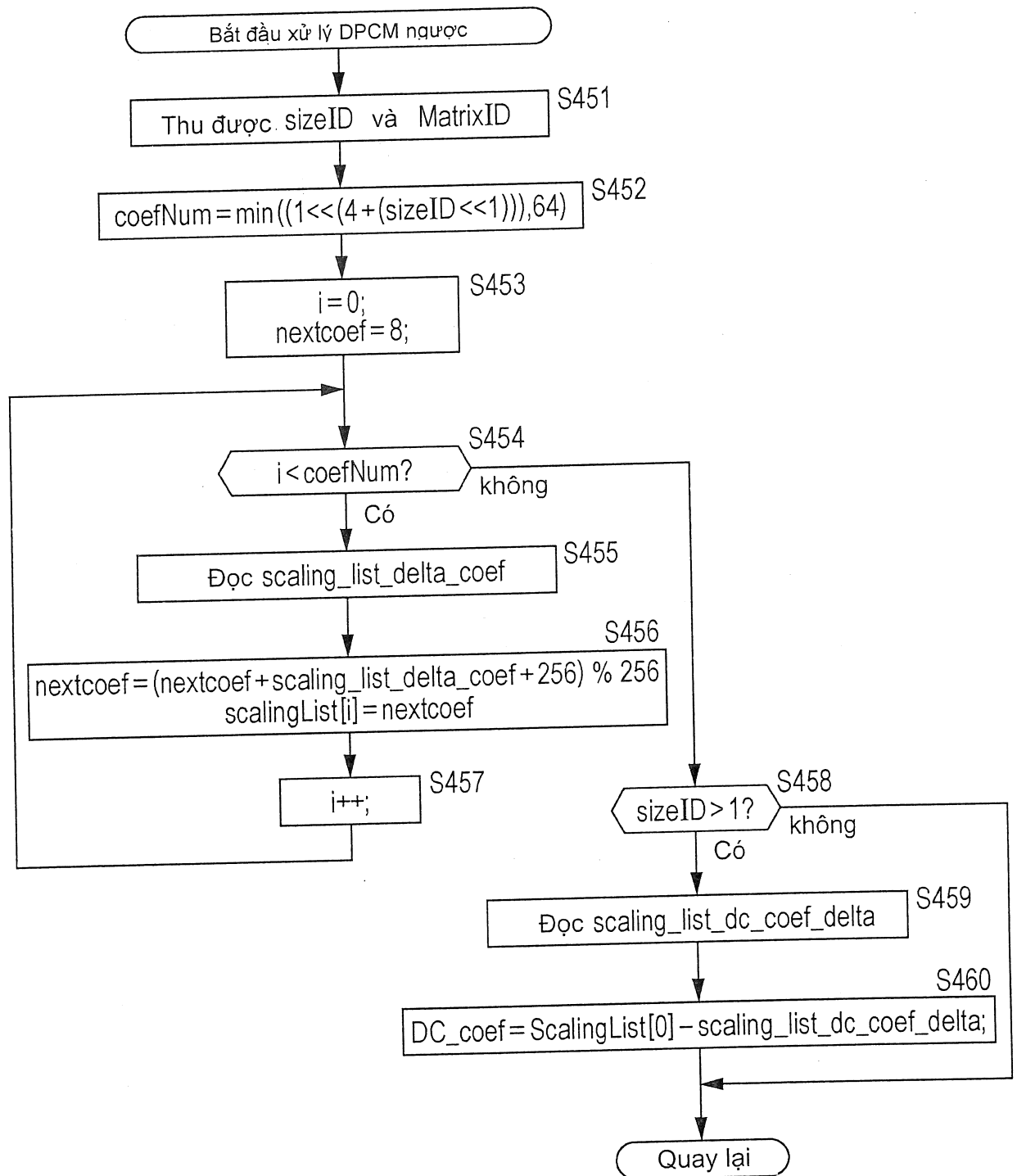
34/58

FIG. 35

	Phần mô tả
scaling_list(ScalingList, sizeID, MatrixID) {	
nextcoef=8	
coefNum = min((1 << (4 + (sizeID << 1))), 64)	
for(i=0; i < coefNum, i++) {	
scaling_list_delta_coef	se(v)
nextcoef = (nextcoef + scaling_list_delta_coef + 256) % 256	
ScalingList[i] = nextcoef	
}	
if(sizeID > 1)	
scaling_list_dc_coef_delta[sizeID-2][MatrixID]	ue(v)
}	

35 / 58

FIG. 36



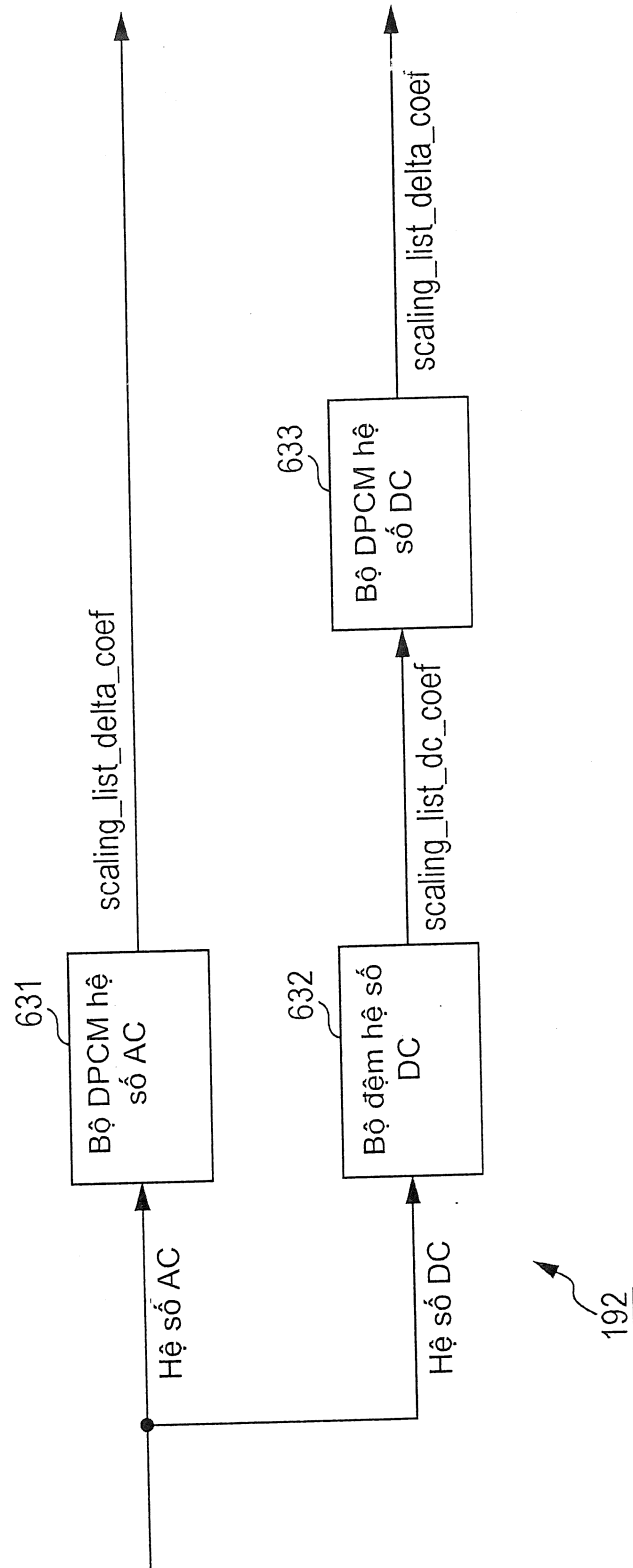
36/58

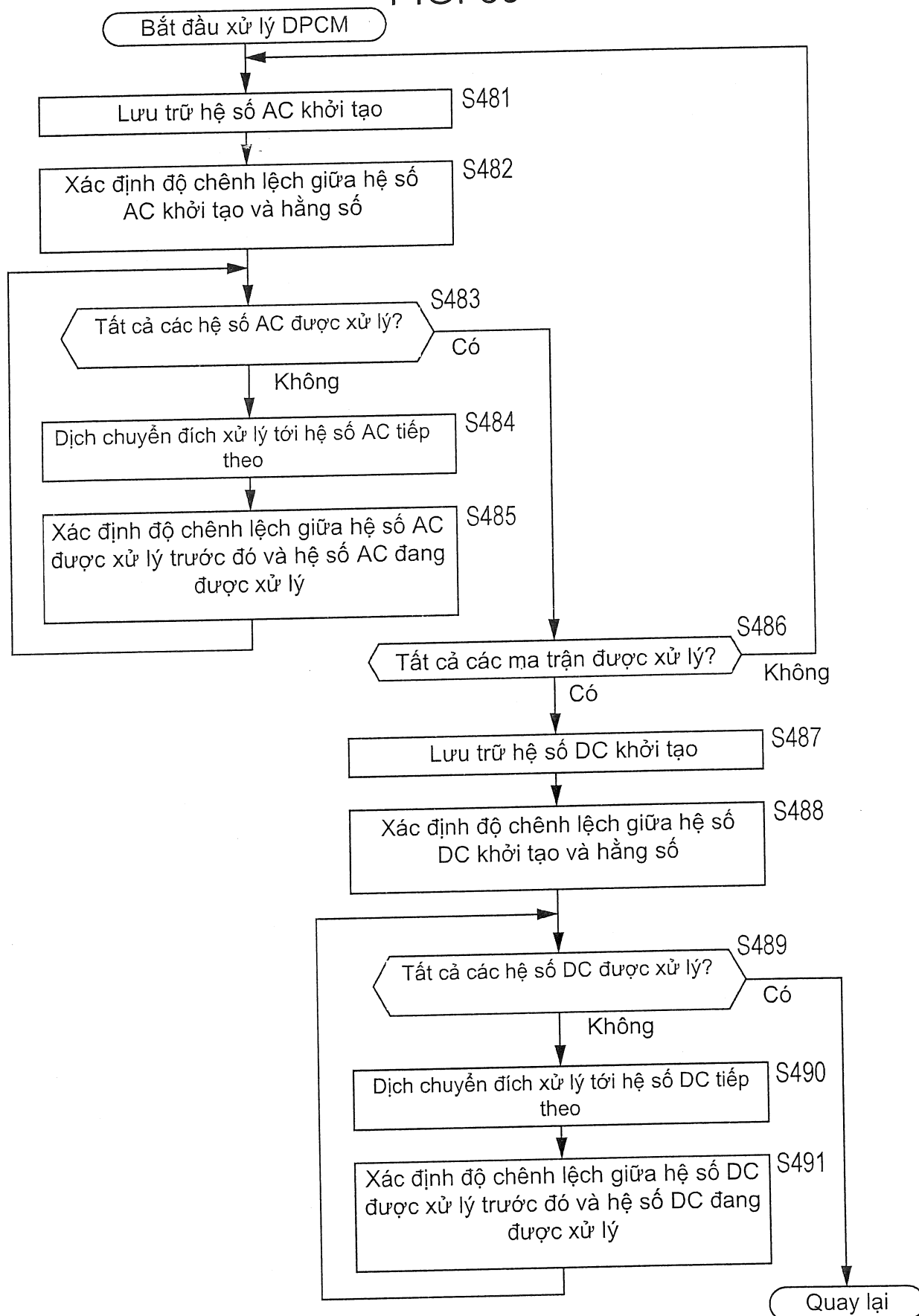
FIG. 37

	Phần mô tả
scaling_list_dc() {	
nextcoef = 8	
for (sizeID = 2; sizeID < 4; sizeID++)	
for (MatrixID = 0; MatrixID < (SizeID == 3) ? 2 : 6; MatrixID++) {	
scaling_list_delta_coef	se (v)
nextcoef = (nextcoef + scaling_list_delta_coef + 256) % 256	
scaling_list_dc_coef[sizeID-2][MatrixID] = nextcoef	
}	
}	

37/58

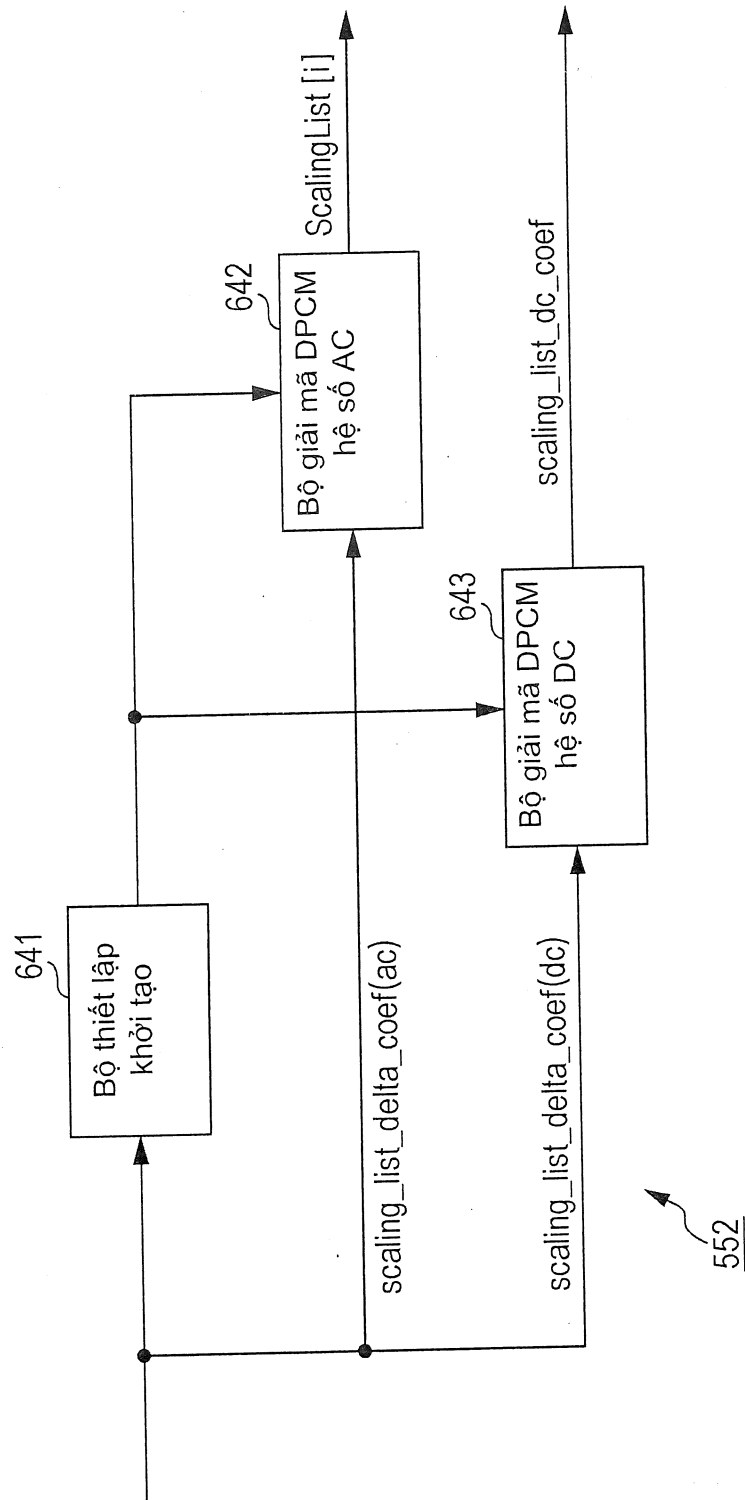
FIG. 38



38 / 58
FIG. 39

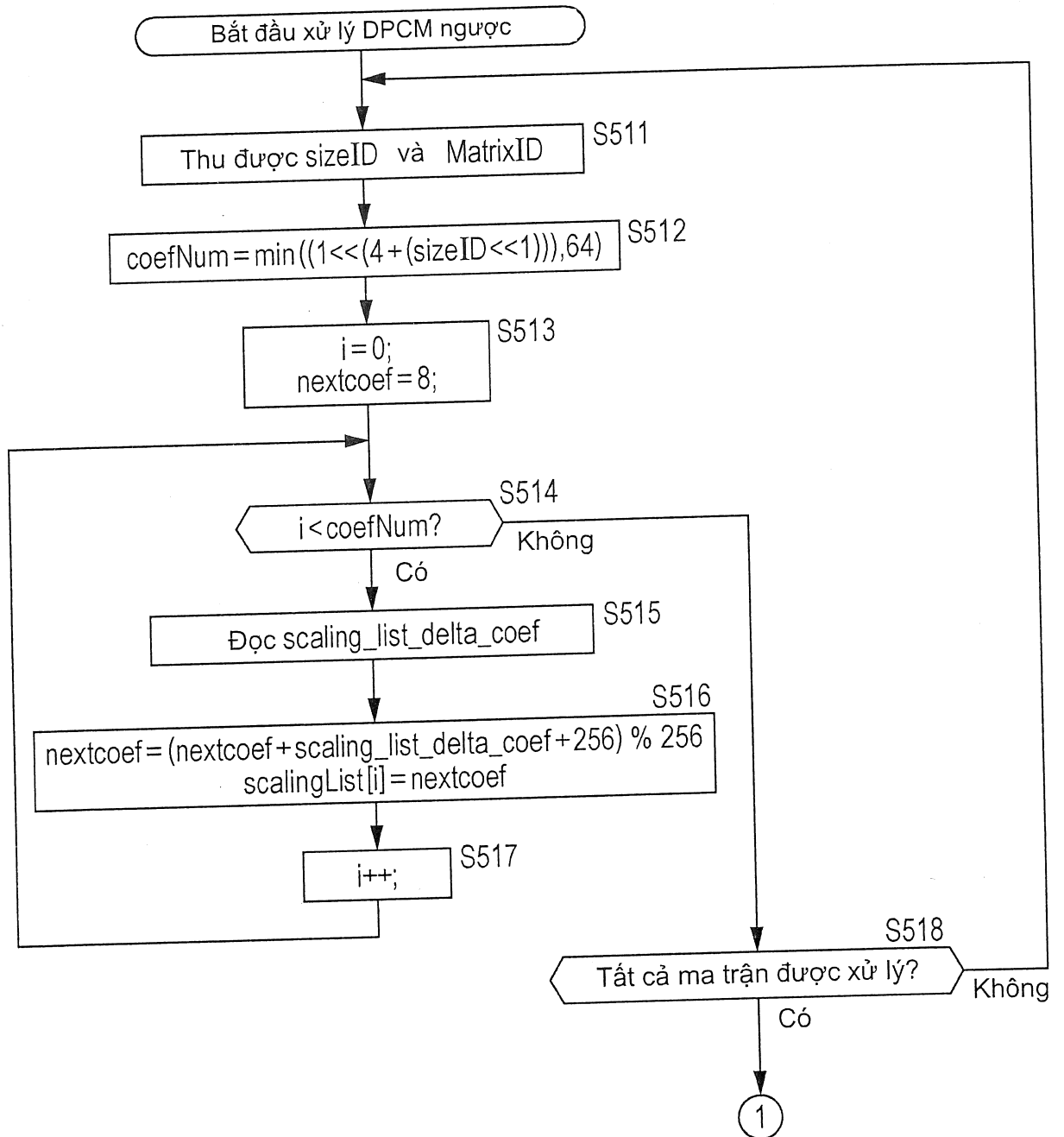
39/58

FIG. 40



40/58

FIG. 41



41 / 58

FIG. 42

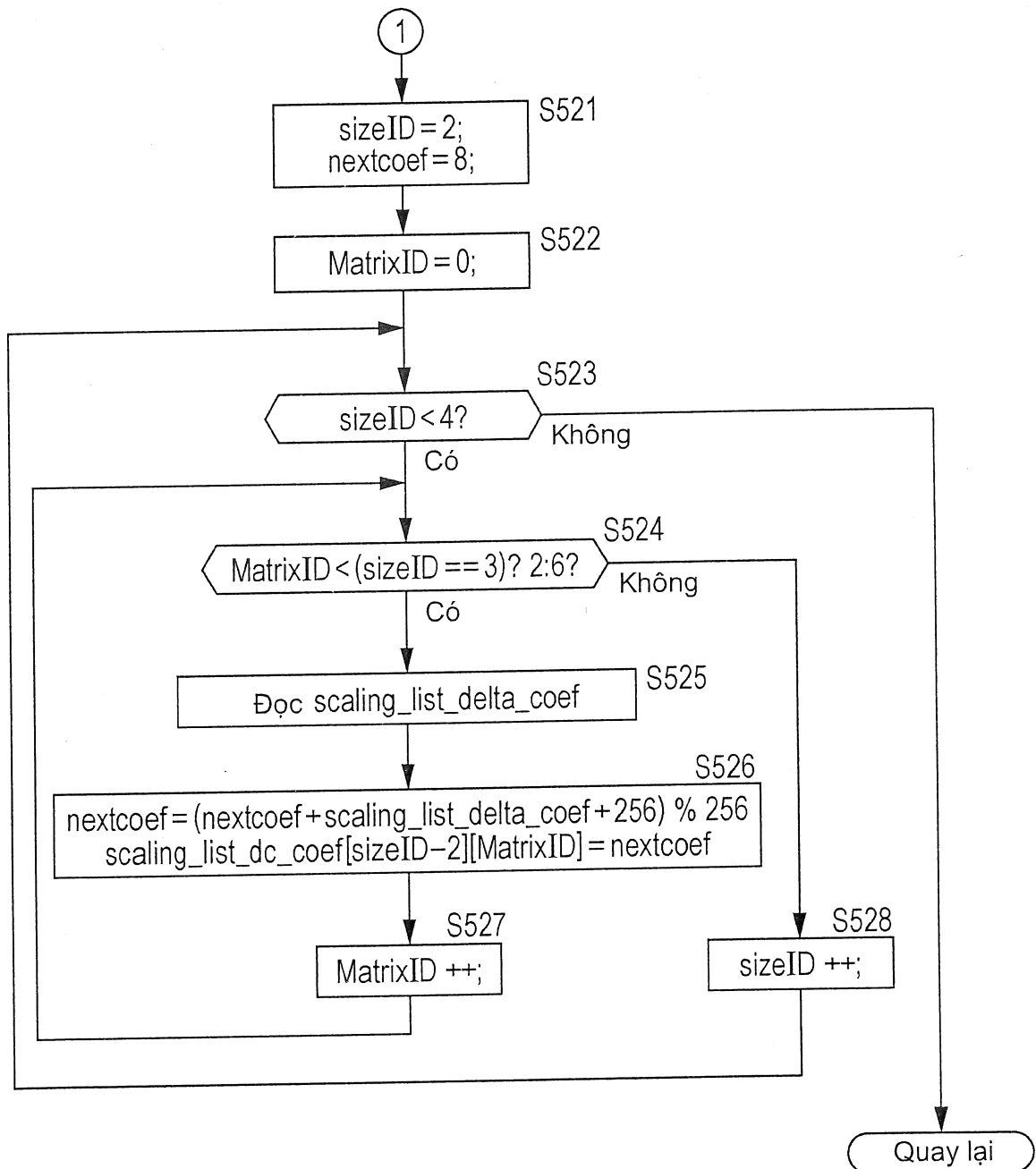


FIG. 43

42 / 58

scaling_list_param() {	Phần mô tả
scaling_list_present_flag	u(l)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID ++)	
for(matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID ++) {	
scaling_list_pred_mode_flag	u(l)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	
}	

A

scaling_list(scalingList, sizeID, matrixID) {	Phần mô tả
nextCoef = 8	u(l)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
if(scaling_list_dc_coef_minus8[sizeID - 2][matrixID] == -8)	
UseDefaultScalingMatrixFlag = 1	
nextCoef = scaling_list_dc_coef_minus8[sizeID - 2][matrixID] + 8	
}	
if(UseDefaultScalingMatrixFlag == 0) {	
for(i = 0; i < coefNum, i ++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
if(sizeID < 2) {	
useDefaultScalingMatrixFlag = (i == 0 && nextCoef == 0)	
if(useDefaultScalingMatrixFlag)	
break	
}	
scalingList[i] = nextCoef	
}	
}	
}	

B

43 / 58

FIG. 44

scaling_list_param() {	Phần mô tả
scaling_list_present_flag	u(l)
if (scaling_list_present_flag)	
for (sizeID = 0; sizeID < 4; sizeID++)	
for (matrixID = 0; matrixID < (sizeID == 3) ? 2 : 6; matrixID++) {	
scaling_list_pred_mode_flag	u(l)
if (!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	se(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	
}	

A

scaling_list(ScalingList, sizeID, matrixID) {	Phần mô tả
nextCoef = 8	u(l)
coefNum = Min(64, (1 << (4 + (sizeID < 1))))	
if (sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
}	
for (i = 0; i < coefNum; i++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
ScalingList[i] = nextCoef	
}	
}	

B

Nếu scaling_list_pred_matrix_id_delta bằng 0,
 UseDefaultScalingMatrixFlag = 1
 Ngoài ra thì
 RefMatrixID = MatrixID - (scaling_list_pred_matrix_id_delta)

C

44 / 58

FIG. 45

scaling_list_param() {	Phần mô tả
scaling_list_present_flag	u(l)
if(scaling_list_present_flag)	
for(sizeID = 0; sizeID < 4; sizeID ++)	
for(matrixID = 0; matrixID < (sizeID == 3)? 2:6; matrixID ++) {	
scaling_list_pred_mode_flag	u(l)
if(!scaling_list_pred_mode_flag)	
scaling_list_pred_matrix_id_delta	ue(v)
else	
scaling_list(ScalingList[sizeID][matrixID], sizeID, matrixID)	
}	
}	
}	

A

scaling_list(scalingList, sizeID, matrixID) {	Phần mô tả
nextCoef = 8	u(l)
coefNum = Min(64, (1 << (4 + (sizeID << 1))))	
UseDefaultScalingMatrix = 0	
if(sizeID > 1) {	
scaling_list_dc_coef_minus8[sizeID - 2][matrixID]	se(v)
nextCoef = scaling_list_dc_coef_minus8[sizeID - 2][matrixID] + 8	
}	
for(i = 0; i < coefNum; i ++) {	
scaling_list_delta_coef	se(v)
nextCoef = (nextCoef + scaling_list_delta_coef + 256) % 256	
scalingList[i] = nextCoef	
}	
}	

B

Nếu scaling_list_pred_matrix_id_delta bằng 0,
 UseDefaultScalingMatrixFlag = 1
 Ngoài ra thì
 RefMatrixID = MatrixID - (scaling_list_pred_matrix_id_delta)

C

45/58

FIG. 46

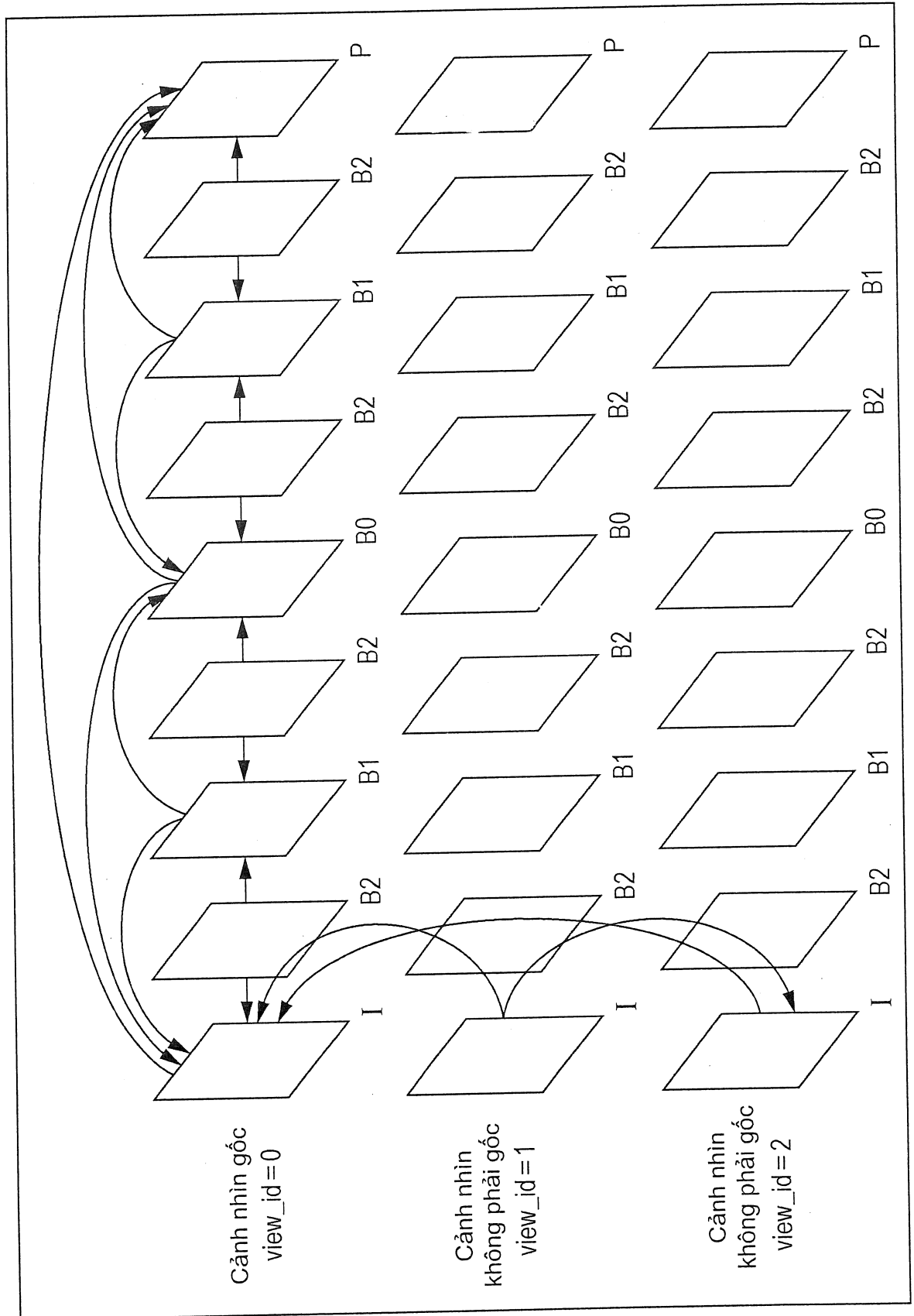
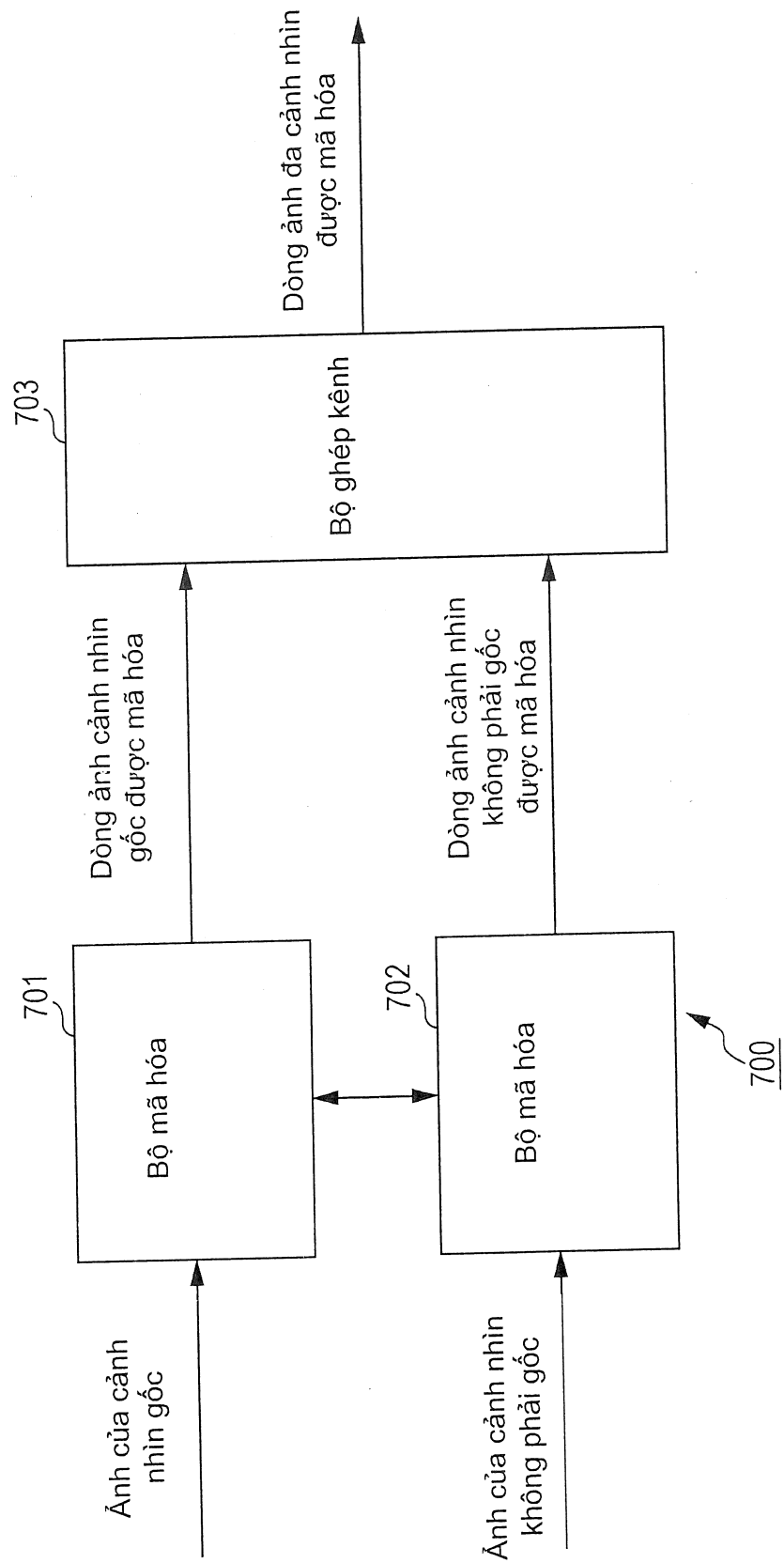
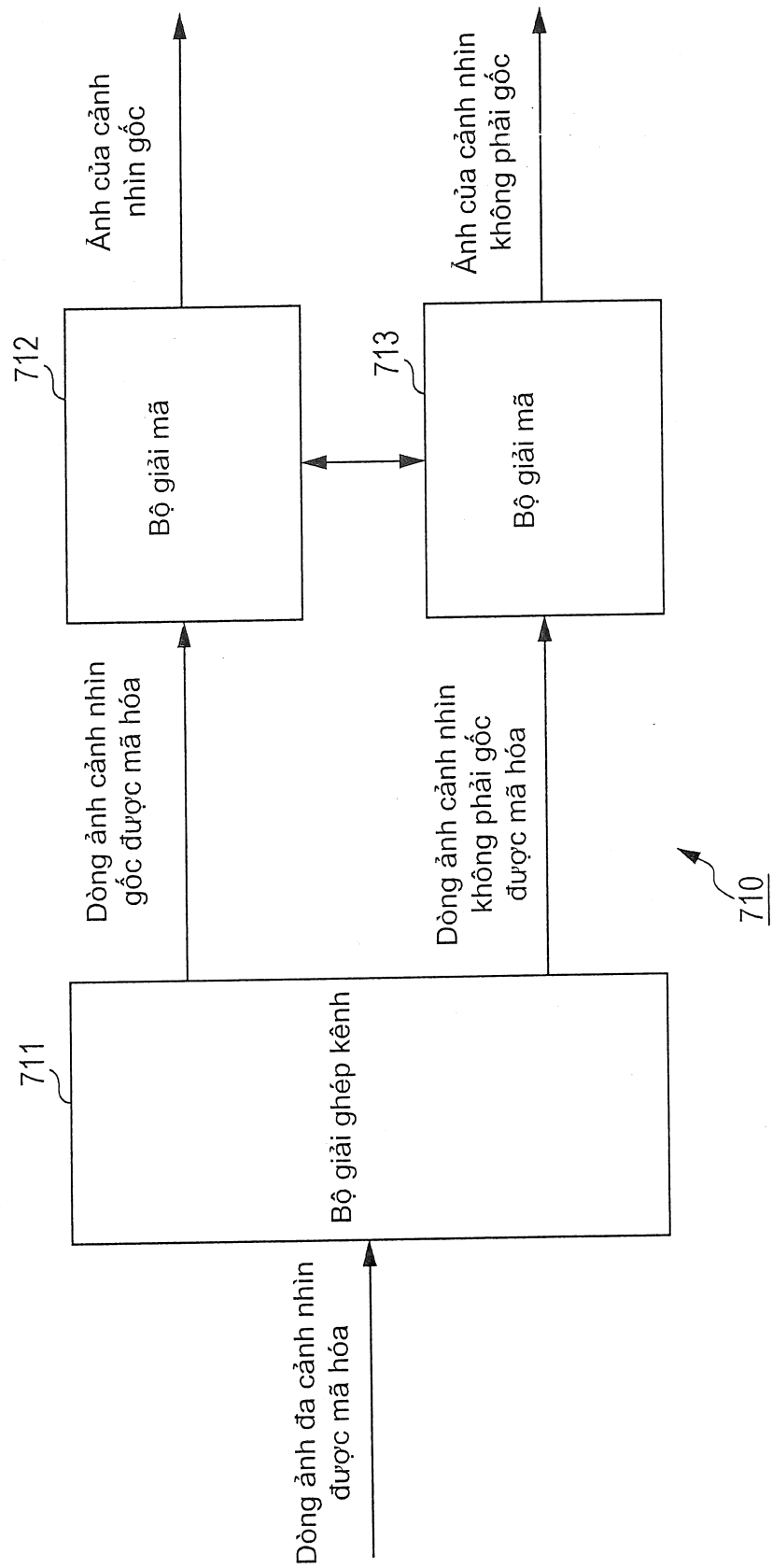


FIG. 47



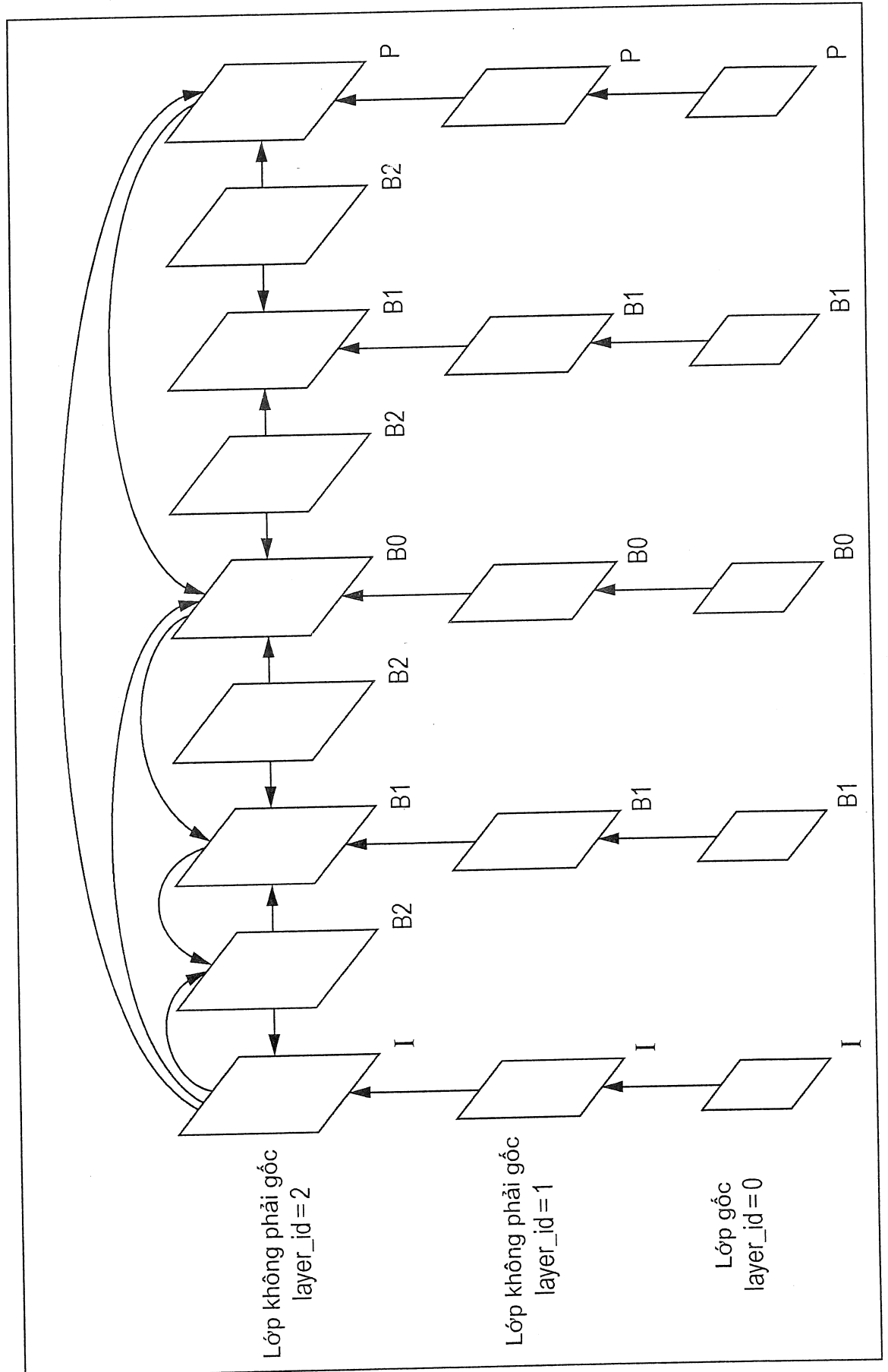
47/58

FIG. 48



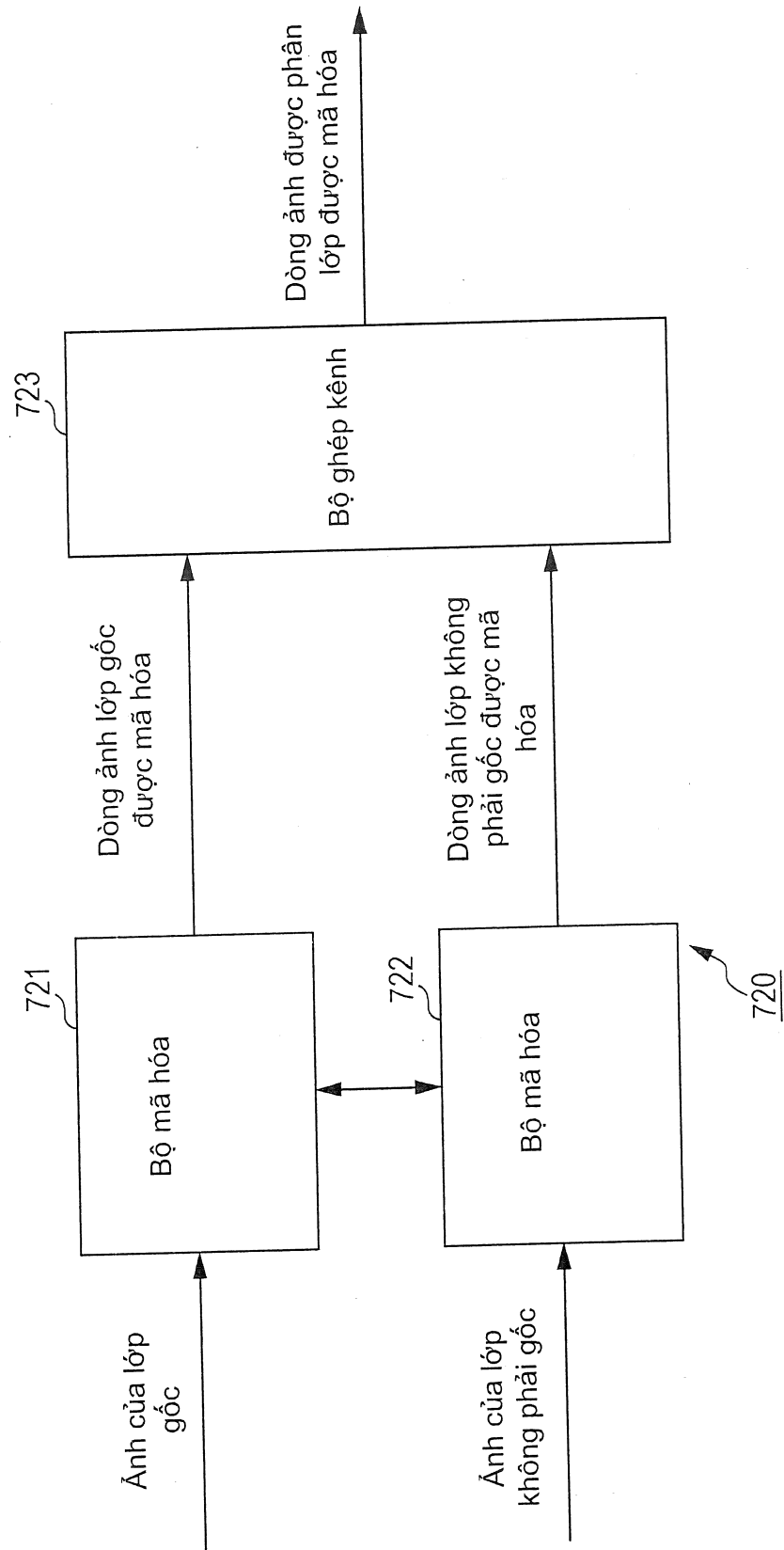
48/58

FIG. 49



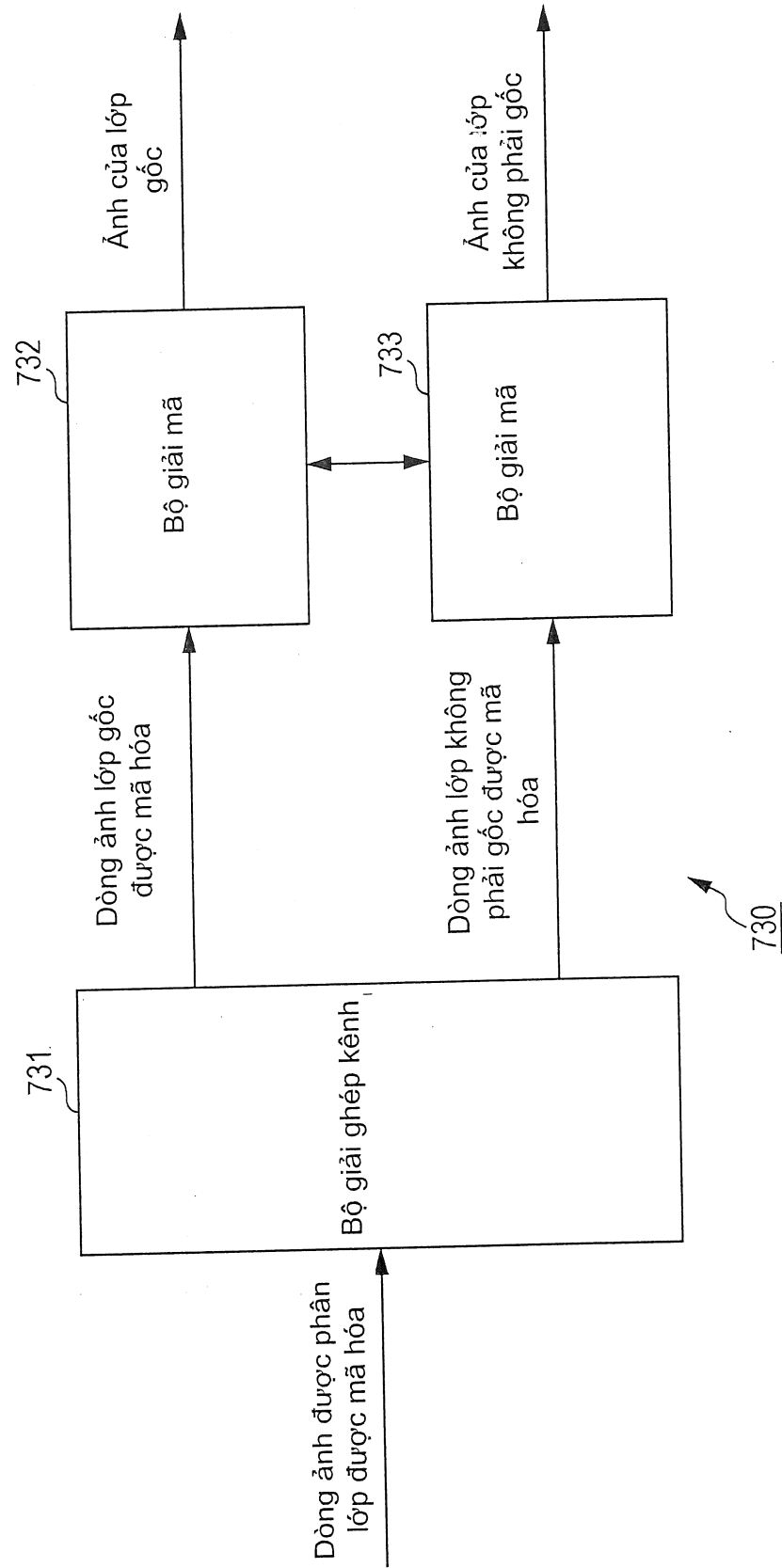
49/58

FIG. 50



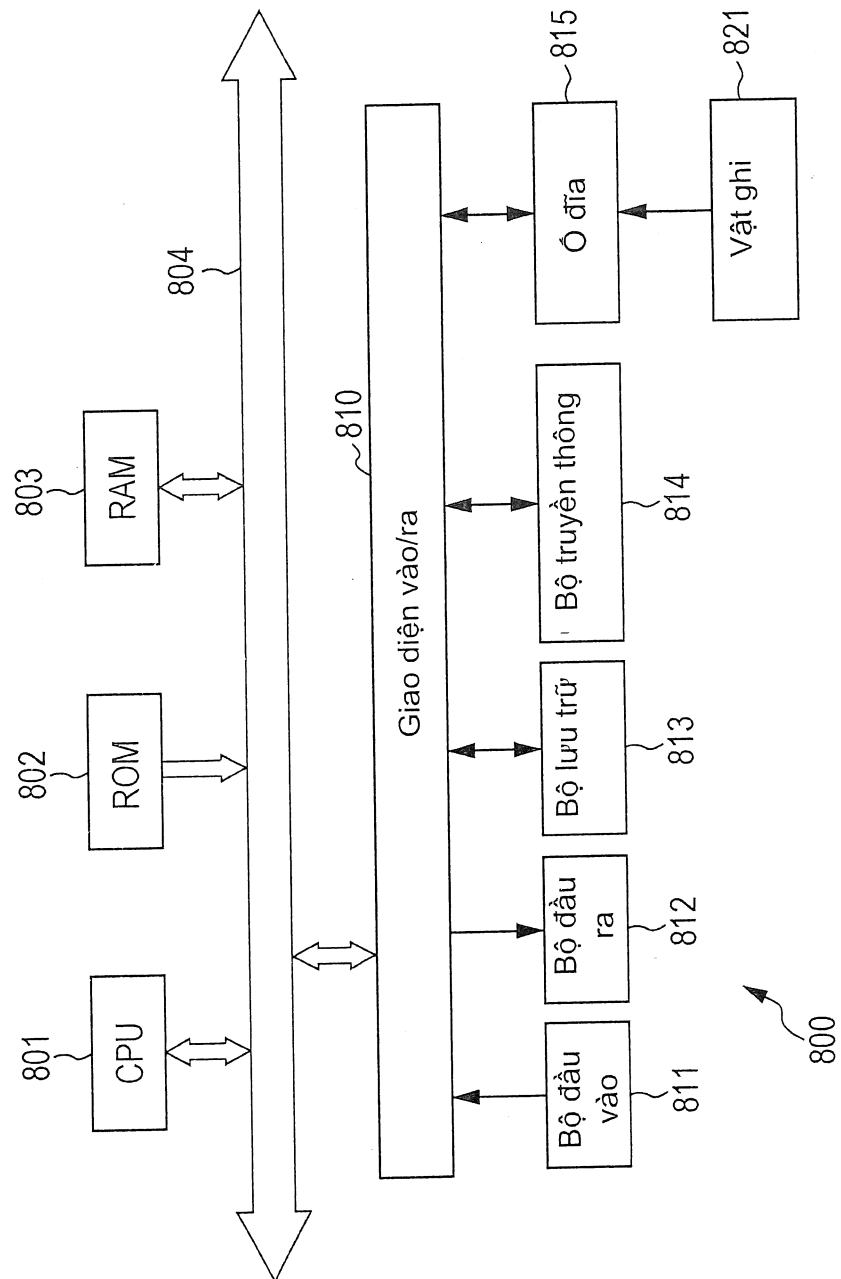
50/58

FIG. 51



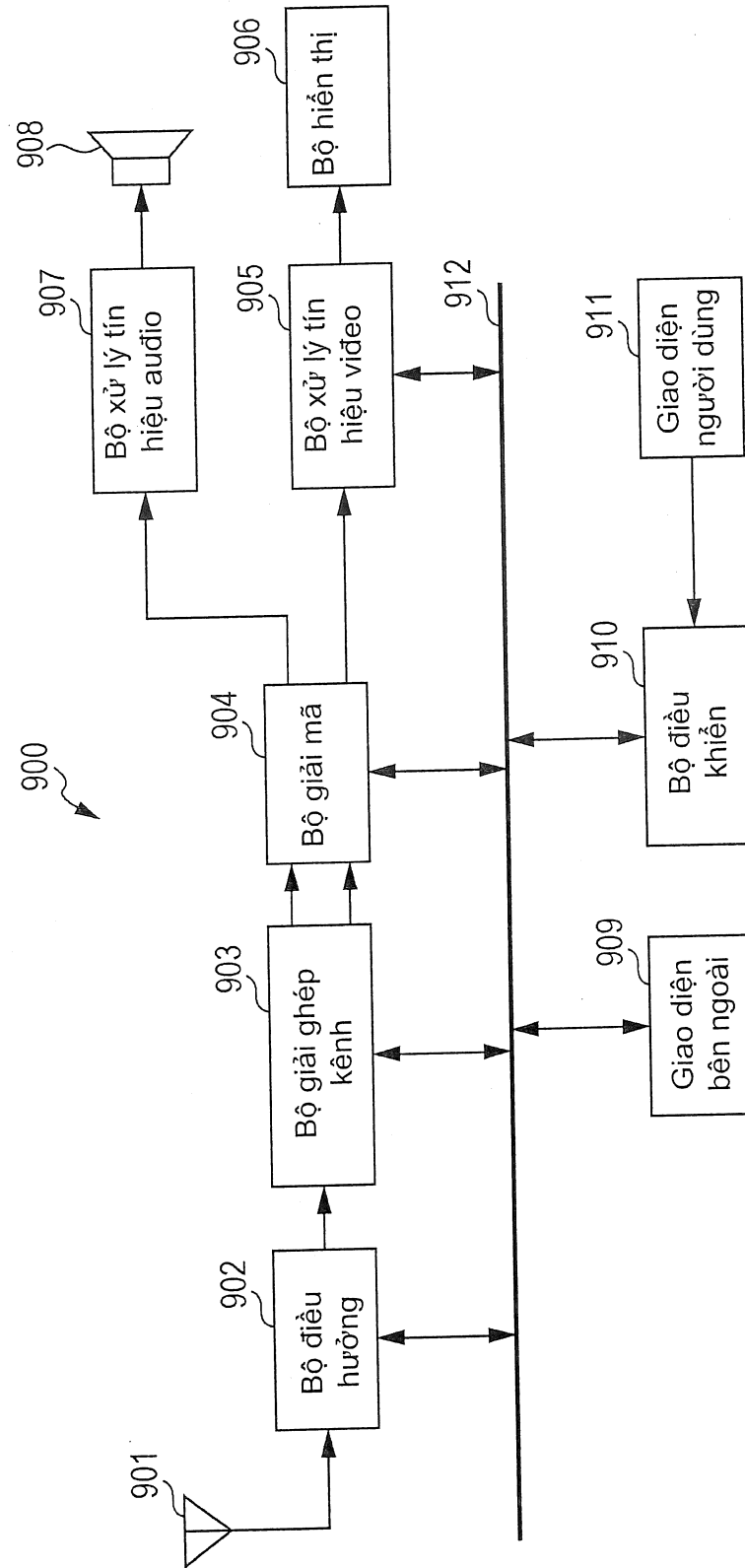
51/58

FIG. 52



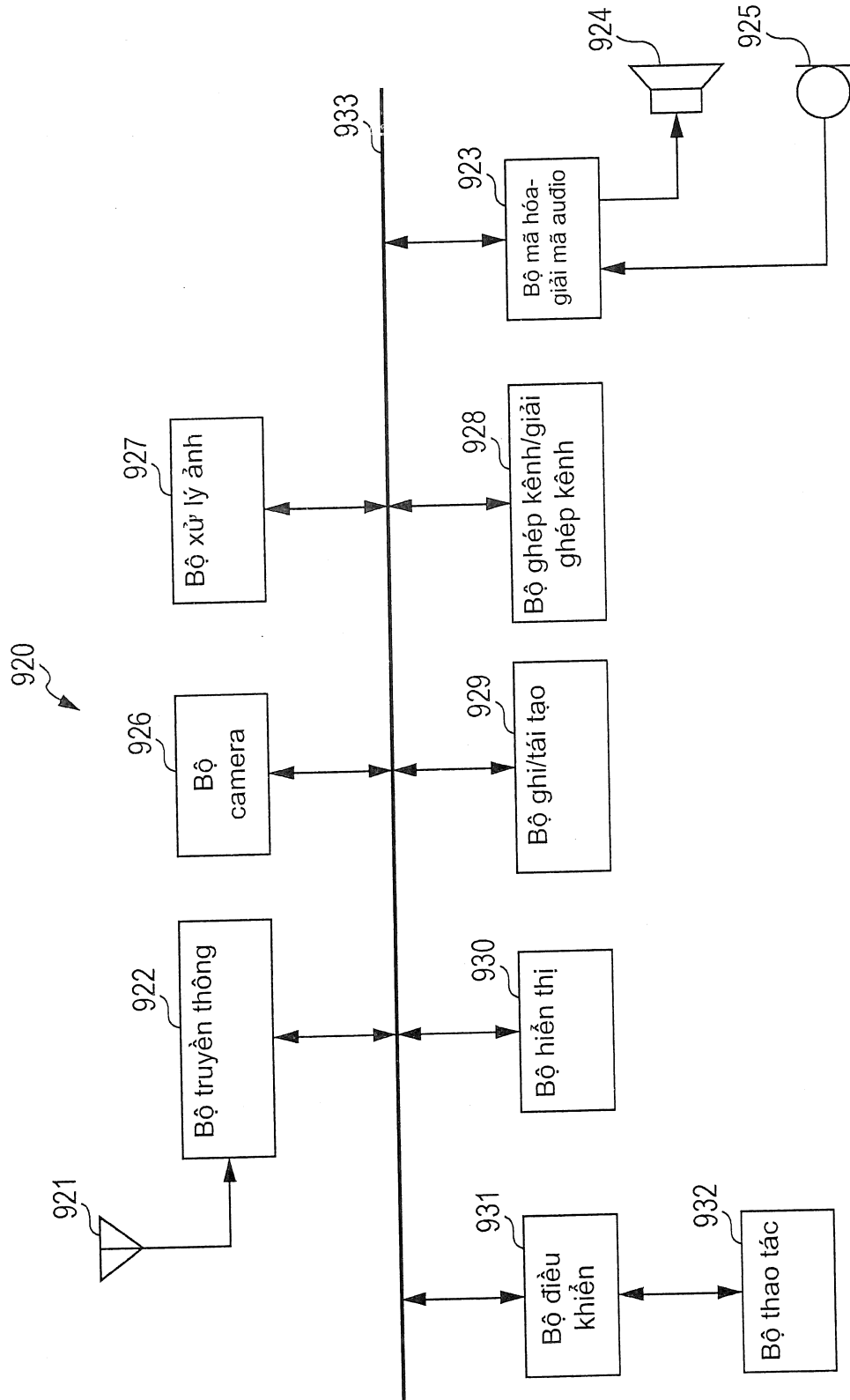
52/58

FIG. 53



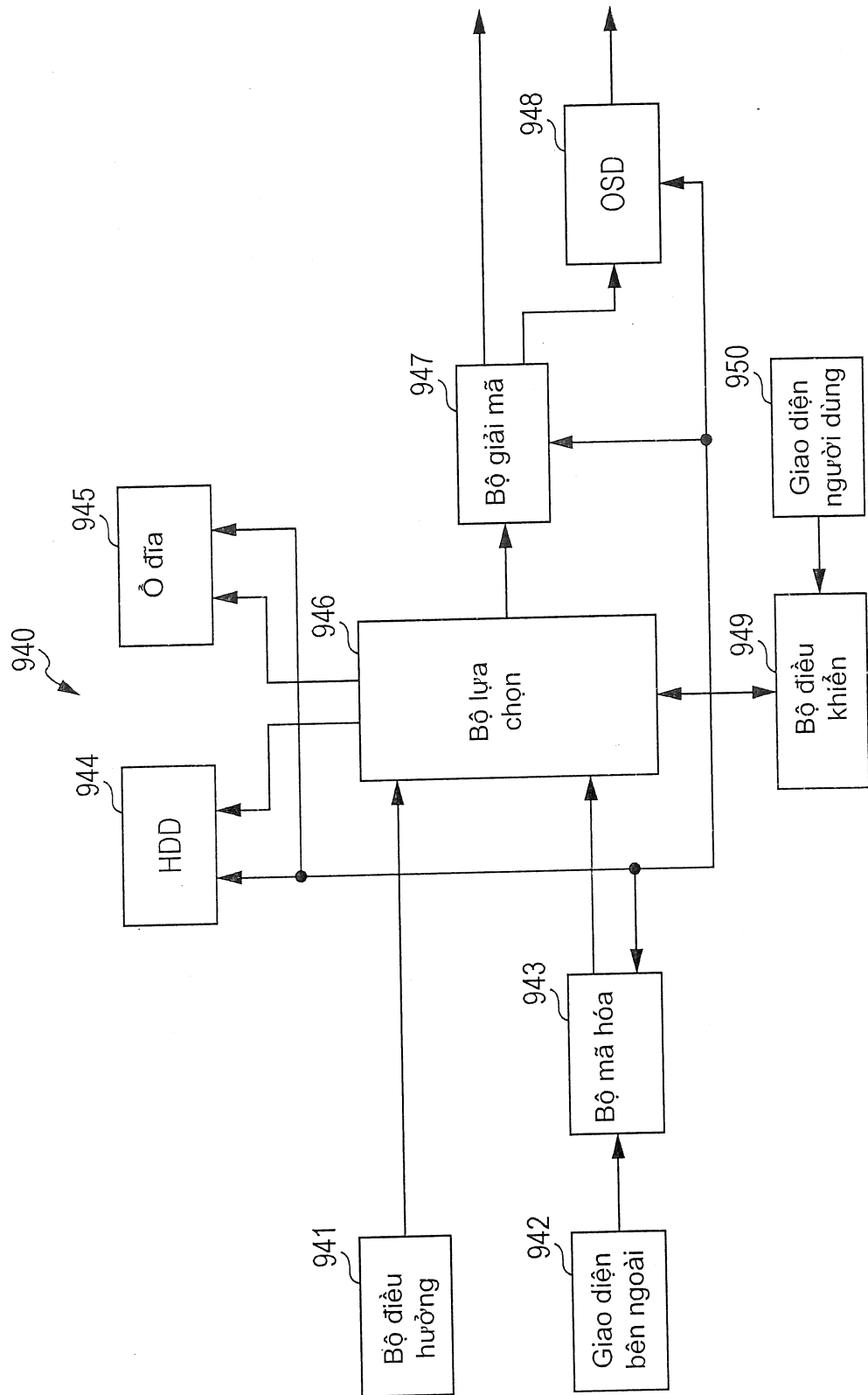
53/58

FIG. 54



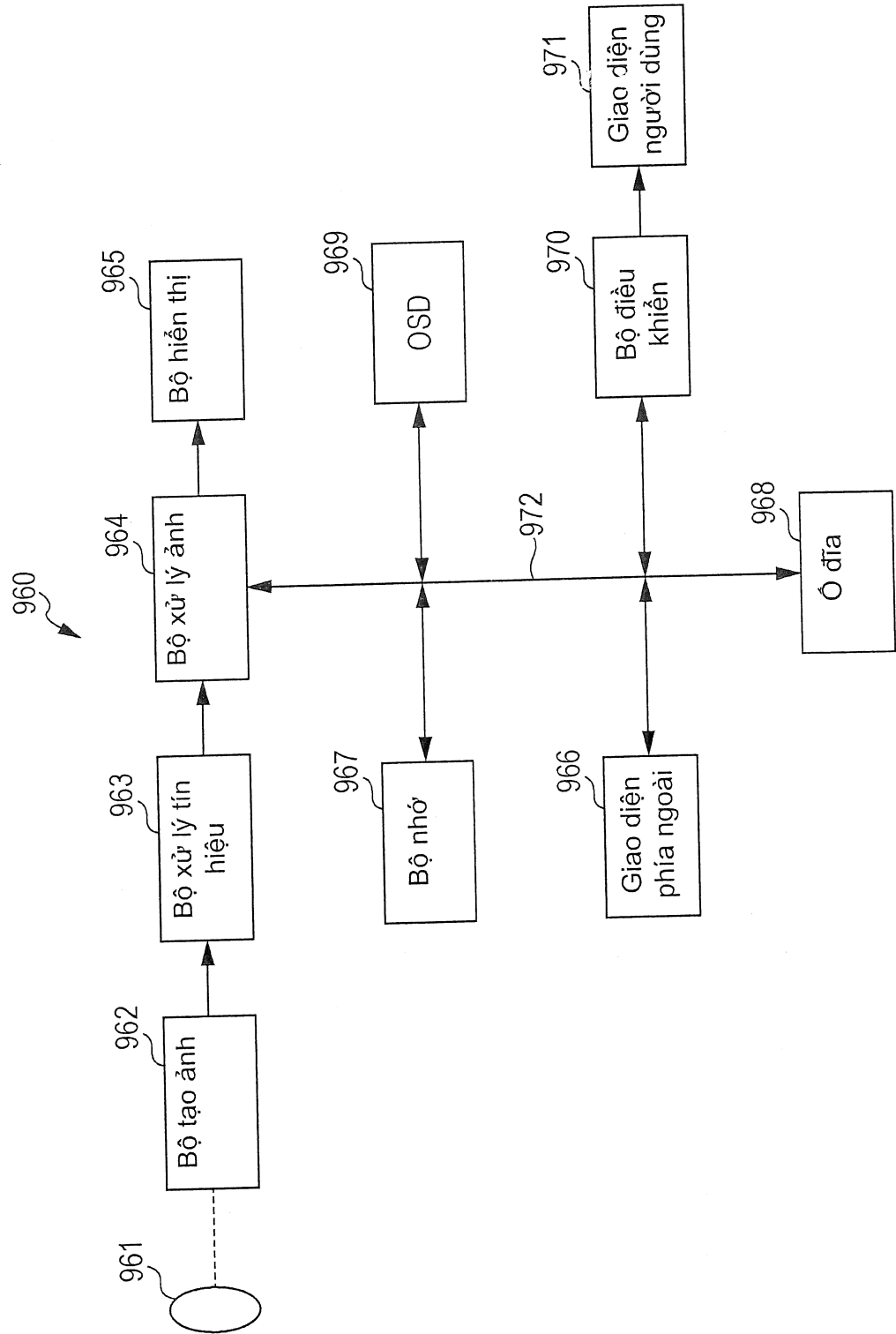
54/58

FIG. 55



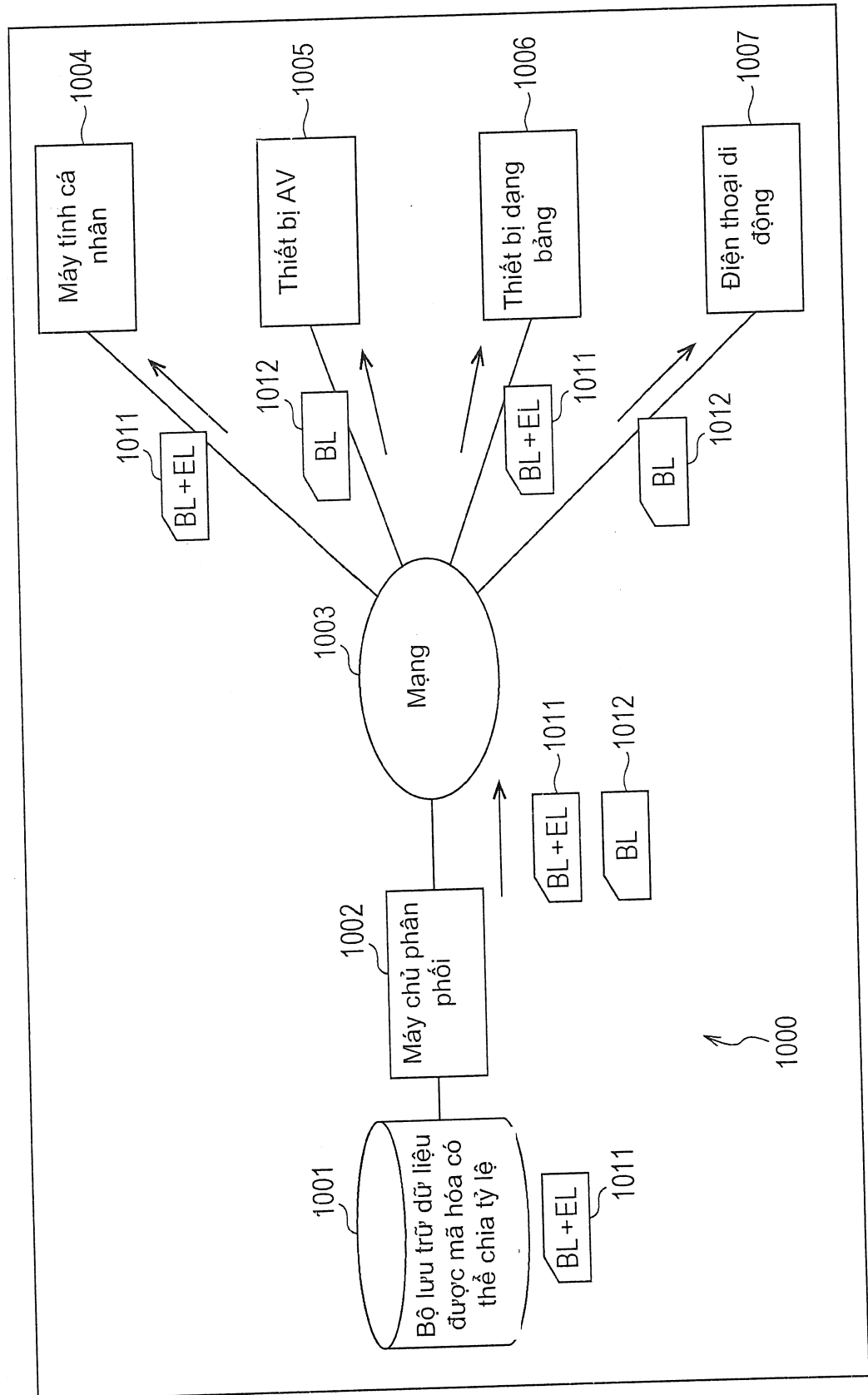
55/58

FIG. 56



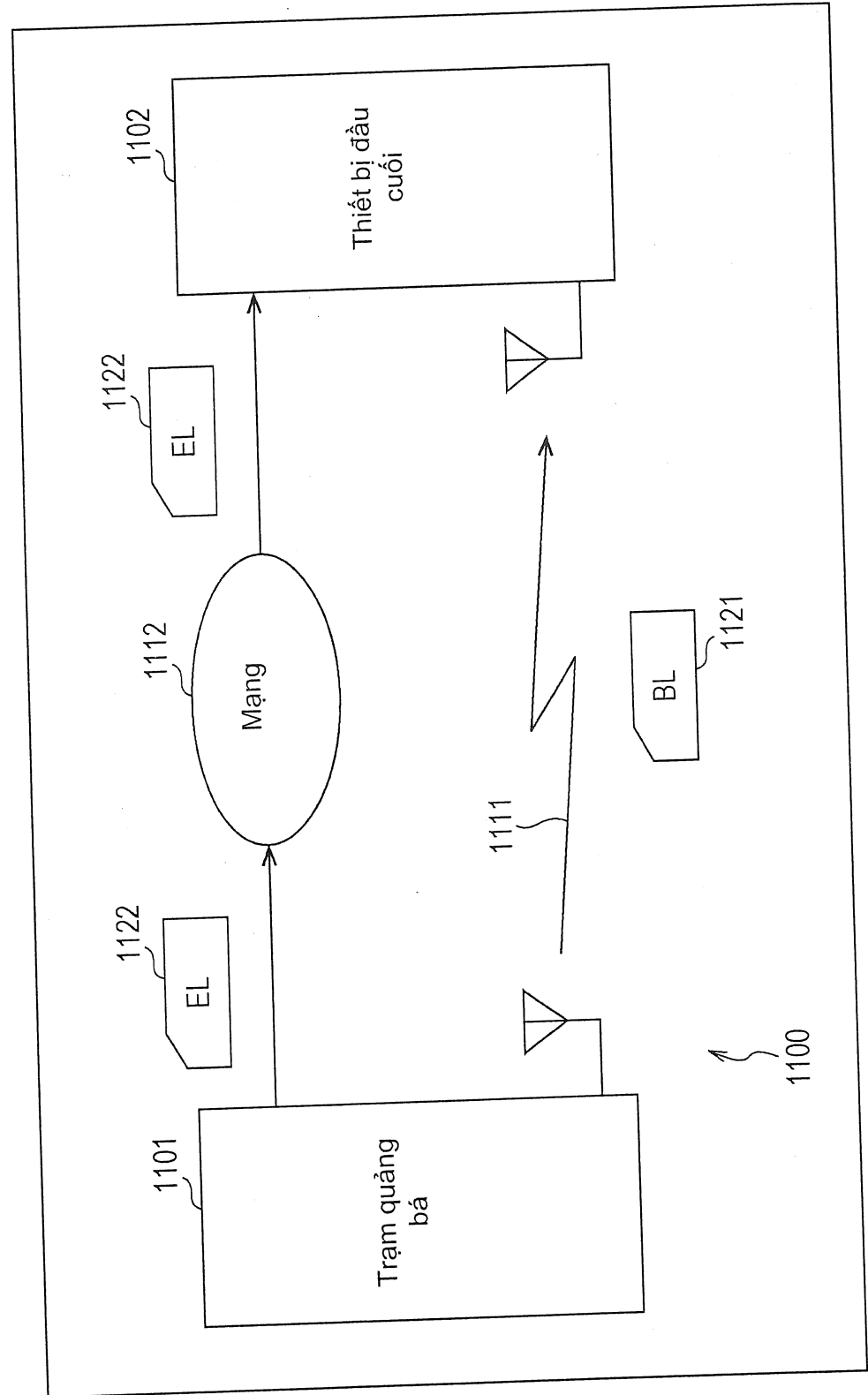
56/58

FIG. 57



57/58

FIG. 58



58/58

FIG. 59

