



- (12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ
1-0049699
(51)^{2020.01} H04N 19/46; H04N 19/176; H04N 19/119; H04N 19/136 (13) B

-
- (21) 1-2022-01793 (22) 28/09/2020
(86) PCT/CN2020/118389 28/09/2020 (87) WO 2021/063308 08/04/2021
(30) PCT/EP2019/076805 03/10/2019 EP
(45) 25/08/2025 449 (43) 25/07/2022 412A
(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)
Huawei Administration Building Bantian, Longgang District Shenzhen, Guangdong
518129, China
(72) GAO, Han (CN); ESENLIK, Semih (TR); ALSHINA, Elena Alexandrovna (RU);
KOTRA, Anand Meher (IN); WANG, Biao (CN).
(74) Công ty Luật TNHH Phạm và Liên danh (PHAM & ASSOCIATES)
-

- (54) PHƯƠNG PHÁP TẠO MÃ, BỘ GIẢI MÃ VIDEO VÀ VẬT GHI MÁY TÍNH ĐƯỢC ĐƯỢC

(21) 1-2022-01793

(57) Sáng chế đề cập đến phương pháp tạo mã được thực hiện bằng thiết bị giải mã, phương pháp bao gồm các bước: thu được giá trị chỉ số chế độ tách đối với khối mã hiện tại; thu được giá trị chỉ số góc angleIdx đối với khối mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc angleIdx dựa trên giá trị chỉ số chế độ tách; thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; và giải mã khối mã hiện tại theo giá trị chỉ số partIdx.

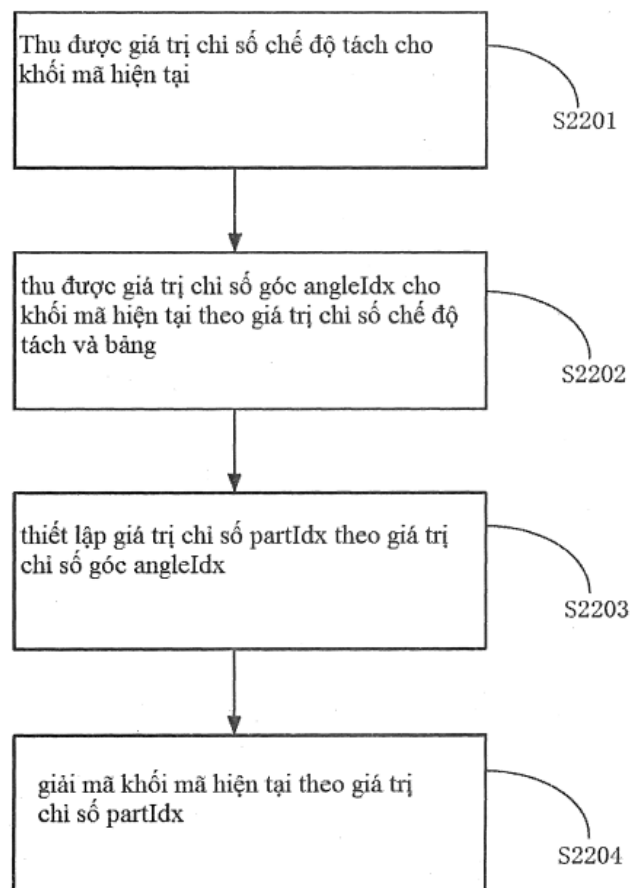


Fig.22

Lĩnh vực kỹ thuật được đề cập

Sáng chế nói chung đề cập đến lĩnh vực xử lý ảnh và cụ thể hơn đến phân vùng hình học.

Tình trạng kỹ thuật của sáng chế

Kỹ thuật tạo mã video (mã hóa và giải mã video) được sử dụng trong phạm vi rộng của các ứng dụng video số, chẳng hạn phát sóng TV số, truyền video trên mạng Internet và mạng di động, các ứng dụng hội thoại thời gian thực chẳng hạn trò chuyện video (video chat), hội nghị video, đĩa DVD và Blu-ray, hệ thống thu thập và chỉnh sửa video, và đầu ghi của các ứng dụng an ninh.

Lượng dữ liệu video cần mô tả đoạn video rất ngắn có thể là lớn, có thể dẫn đến các khó khăn khi dữ liệu cần được truyền trực tuyến hoặc ngược lại được truyền trên mạng truyền thông có dung lượng băng thông hữu hạn. Do vậy, dữ liệu video thường được nén trước khi được truyền trên các mạng viễn thông hiện đại. Kích thước của video cũng có thể là vấn đề khi video được lưu trữ trên thiết bị lưu trữ do các tài nguyên bộ nhớ có thể bị giới hạn. Thiết bị nén video thường sử dụng phần mềm và/hoặc phần cứng ở phía nguồn để mã hóa dữ liệu video trước khi truyền hoặc lưu trữ, nhờ đó giảm số lượng dữ liệu cần biểu diễn ảnh video số. Sau đó, dữ liệu được nén được nhận ở phía đích bởi thiết bị giải nén video mà giải mã dữ liệu video. Với tài nguyên mạng hữu hạn và các nhu cầu ngày càng tăng của chất lượng video cao hơn, các kỹ thuật nén và giải nén cải tiến mà cải thiện tỷ lệ nén không làm hi sinh chất lượng ảnh được mong muốn.

Bản chất kỹ thuật của sáng chế

Sáng chế đề cập đến các thiết bị và phương pháp mã hóa và giải mã theo các điểm yêu cầu bảo hộ độc lập.

Các mục tiêu nêu trên và các mục tiêu khác đạt được nhờ đối tượng của các yêu cầu bảo hộ độc lập. Các dạng thực hiện khác rõ ràng từ các điểm yêu cầu bảo hộ phụ thuộc, phần mô tả và các hình vẽ.

Khía cạnh thứ nhất của sáng chế đề cập đến phương pháp tạo mã được thực hiện bởi thiết bị giải mã, phương pháp bao gồm các bước: thu được giá trị chỉ số chế độ tách đối với khối mã hiện tại; thu được giá trị chỉ số góc angleIdx đối với khối mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc angleIdx dựa trên giá trị chỉ số chế độ tách; thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; giải mã khối mã hiện tại theo giá trị chỉ số partIdx.

Theo các phương án thực hiện sáng chế, khối mã được giải mã theo giá trị chỉ số. Quá trình giải mã có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vector chuyển động và v.v. Do vậy, hiệu suất tận dụng bộ đệm và giải mã đã được cải thiện.

Theo một phương án thực hiện, giá trị chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng đối với khối mã hiện tại. Chẳng hạn, geo_partition_idx or merge_gpm_partition_idx.

Trong ví dụ, merge_gpm_partition_idx[x0][y0] hoặc geo_partition_idx xác định hình dạng phân vùng của chế độ hợp nhất phân vùng hình học. Các chỉ số mảng x0, y0 xác định vị trí (x0, y0) của mẫu độ sáng trên bên trái của khối mã được xem xét so với mẫu độ sáng trên bên trái của ảnh.

Khi merge_gpm_partition_idx[x0][y0] hoặc geo_partition_idx[x0][y0] không hiện có, được suy ra bằng 0.

Trong ví dụ, giá trị chỉ số chế độ tách có thể thu được bằng cách phân tách

giá trị chỉ số được mã hóa trong dòng bit video, hoặc giá trị chỉ số chế độ tách có thể được xác định theo giá trị cú pháp được phân tách từ dòng bit video.

Dòng bit có thể thu được theo mạng không dây hoặc mạng hữu tuyến. Dòng bit có thể được truyền từ website, máy chủ, hoặc nguồn từ xa khác nhờ sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn, đường thuê bao số (digital subscriber line, DSL), hoặc các công nghệ không dây chẳng hạn hồng ngoại, vô tuyến, vi sóng, WiFi, Bluetooth, LTE hoặc 5G.

Theo phương án thực hiện, dòng bit là chuỗi bit, chẳng hạn, ở dạng của dòng đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) hoặc dòng byte, tạo biểu diễn chuỗi đơn vị truy nhập (access unit, AU) tạo thành một hoặc nhiều chuỗi video được mã hóa (coded video sequence, CVS).

Theo một số phương án thực hiện, đối với quá trình giải mã, phía bộ giải mã đọc dòng bit và dẫn xuất các ảnh được giải mã từ dòng bit; đối với quá trình mã hóa, phía bộ mã hóa tạo dòng bit.

Thông thường, dòng bit sẽ bao gồm các phần tử cú pháp được tạo bởi cấu trúc cú pháp. phần tử cú pháp: Phần tử của dữ liệu được biểu diễn trong dòng bit.

cấu trúc cú pháp: 0 hoặc nhiều phần tử cú pháp đồng thời có mặt trong dòng bit theo thứ tự cụ thể.

Trong ví dụ cụ thể, các định dạng dòng bit xác định mối quan hệ giữa dòng đơn vị NAL và dòng byte, trong số đó được gọi là dòng bit.

Dòng bit có thể, chẳng hạn, có một trong hai định dạng: định dạng dòng đơn vị NAL hoặc định dạng dòng byte. Định dạng dòng đơn vị NAL về mặt khái niệm là loại cao hơn “cơ bản”. Định dạng dòng đơn vị NAL bao gồm chuỗi cấu trúc cú pháp được gọi là các đơn vị NAL. Chuỗi này được sắp thứ tự theo thứ tự

giải mã. Có các ràng buộc đặt lên thứ tự giải mã (và các nội dung) của các đơn vị NAL trong dòng đơn vị NAL.

Định dạng dòng byte có thể được tạo từ định dạng dòng đơn vị NAL bằng cách sắp xếp thứ tự các đơn vị NAL theo thứ tự giải mã và tạo tiền tố mỗi đơn vị NAL có tiền tố mã bắt đầu và 0 hoặc nhiều byte có giá trị bằng 0 để tạo dòng byte. Định dạng dòng đơn vị NAL có thể được trích xuất từ định dạng dòng byte bằng cách tìm kiếm vị trí của mẫu hình tiền tố mã bắt đầu duy nhất trong dòng byte này.

Mệnh đề này xác định phương án thực hiện của mối quan hệ giữa các ảnh nguồn và được giải mã được tạo qua dòng bit.

Nguồn video được biểu diễn bằng dòng bit là chuỗi ảnh theo thứ tự giải mã.

Thông thường, giá trị của `merge_gpm_partition_idx[x0][y0]` được giải mã từ dòng bit. Trong ví dụ, khoảng giá trị đối với `merge_gpm_partition_idx[][]` là từ 0 đến 63, bao gồm 0 và 63. Trong ví dụ, quá trình giải mã đối với `merge_gpm_partition_idx[][]` là “bypass”.

Khi `merge_gpm_partition_idx[x0][y0]` không hiện có, được suy ra bằng 0.

Theo một phương án thực hiện, trong đó giá trị chỉ số góc `angleIdx` được sử dụng đối với phân vùng hình học của khối mã hiện tại.

Trong ví dụ, giá trị chỉ số góc `angleIdx` xác định chỉ số góc của phân vùng hình học. Dưới đây, giá trị chỉ số góc `angleIdx` đôi lúc cũng được gọi là biến góc phân vùng `angleIdx`.

Giá trị của tham số góc đối với khối hiện tại thu được theo giá trị của giá trị chỉ số chế độ tách và bảng tra cứu định trước.

Theo phương án thực hiện, biến góc phân vùng `angleIdx` (tham số góc) và biến khoảng cách `distanceIdx` của chế độ phân vùng hình học được thiết lập theo giá trị của `merge_gpm_partition_idx[xCb][yCb]` (bộ chỉ báo) như được xác định

trong bảng dưới đây. Theo cách thực hiện, mỗi quan hệ này có thể được thực hiện theo Bảng 1 hoặc theo hàm số.

Bảng 1: Mô tả angleIdx và distanceIdx dựa trên merge_gpm_partition_idx.

merge_gpm_partition_idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angleIdx	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
distanceIdx	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
merge_gpm_partition_idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
angleIdx	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
distanceIdx	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
merge_gpm_partition_idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
merge_gpm_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

Theo một phương án thực hiện, trong đó giá trị chỉ số partIdx thỏa mãn:

$$\text{partIdx} = (\text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2}) ? 1 : 0,$$

trong đó threshold1 và threshold2 là các giá trị nguyên, và threshold1 nhỏ hơn threshold2.

Tương ứng, khi giá trị chỉ số góc angleIdx nằm ở khoảng giữa threshold1 và threshold2, hoặc khi giá trị chỉ số góc angleIdx bằng threshold1 hoặc threshold2, giá trị chỉ số partIdx được gán bằng 1, ngược lại, khi giá trị chỉ số góc angleIdx nằm ngoài khoảng này, giá trị chỉ số partIdx được gán bằng 0. Khối mã hiện tại được giải mã theo giá trị chỉ số partIdx. Chẳng hạn, giá trị chỉ số partIdx định nghĩa một trong hai khối phụ là khối phụ thứ nhất (có partIdx = 0) và khối còn lại là khối phụ thứ hai được giải mã (có partIdx = 1).

Trong ví dụ, trong đó threshold1 bằng 13 và threshold2 bằng 27.

Trong ví dụ, giá trị chỉ số có thể là chỉ số tách biểu diễn dưới dạng isFlip, giá trị chỉ số isFlip cũng thỏa mãn

$$\text{isFlip} = (\text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27) ? 1 : 0.$$

Quá trình giải mã có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vector chuyển động và v.v.. Do vậy, hiệu suất sử dụng và giải mã bộ đệm đã được cải thiện.

Theo một phương án thực hiện, trong đó việc giải mã khối mã hiện tại bao gồm lưu trữ thông tin chuyển động đối với khối hiện tại theo giá trị chỉ số partIdx.

Khía cạnh thứ hai của sáng chế đề cập đến bộ giải mã video, trong đó bộ giải mã bao gồm: khối phân tách, được tạo cấu hình để thu được giá trị chỉ số chế độ tách đối với khối mã hiện tại; khối nhận giá trị chỉ số góc, được tạo cấu hình để thu được giá trị chỉ số góc angleIdx đối với khối mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc angleIdx dựa trên giá trị chỉ số chế độ tách; khối thiết lập, được tạo cấu hình để thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; khối xử lý, được tạo cấu hình để giải mã khối mã hiện tại theo giá trị chỉ số partIdx.

Theo một phương án thực hiện, trong đó giá trị chỉ số partIdx thỏa mãn:

$$\text{partIdx} = (\text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2}) ? 1 : 0,$$

trong đó threshold1 và threshold2 là các giá trị nguyên, và threshold1 nhỏ hơn threshold2.

Theo một phương án thực hiện, trong đó threshold1 bằng 13 và threshold2 bằng 27.

Theo một phương án thực hiện, trong đó khối xử lý được tạo cấu hình để lưu trữ thông tin chuyển động đối với khối hiện tại theo giá trị chỉ số partIdx.

Theo một phương án thực hiện, trong đó giá trị chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng đối với khối mã hiện tại.

Theo một phương án thực hiện, trong đó angleIdx được sử dụng đối với phân vùng hình học của khối mã hiện tại.

Phương pháp theo khía cạnh thứ nhất của sáng chế có thể được thực hiện bằng thiết bị theo khía cạnh thứ hai của sáng chế. Các dấu hiệu và các dạng triển khai khác của các phương pháp nêu trên tương ứng với các dấu hiệu và các dạng triển khai của thiết bị theo khía cạnh thứ hai của sáng chế.

Phương án thực hiện đề xuất bộ giải mã bao gồm hệ mạch xử lý để thực hiện phương pháp theo một trong các phương án thực hiện và các cách triển khai nêu trên.

Phương án thực hiện đề cập đến sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo một trong các phương án thực hiện và các cách triển khai nêu trên.

Phương án thực hiện đề cập đến bộ giải mã hoặc bộ mã hóa bao gồm:

một hoặc nhiều bộ xử lý; và

vật ghi máy tính đọc được bất biến được ghép nối với các bộ xử lý và lưu trữ chương trình để thực thi bằng các bộ xử lý, trong đó chương trình, khi được thực thi bằng các bộ xử lý, tạo cấu hình bộ giải mã hoặc bộ mã hóa để thực hiện phương pháp theo một trong các phương án thực hiện và cách triển khai nêu trên.

Phương án thực hiện đề cập đến phương tiện lưu trữ bất biến, bao gồm dòng bit được mã hóa được giải mã bằng thiết bị giải mã ảnh, dòng bit được tạo bằng cách phân chia khung của tín hiệu video hoặc tín hiệu ảnh thành các khối, và bao gồm các phần tử cú pháp, trong đó các phần tử cú pháp bao gồm bộ chỉ báo (cú pháp) theo một trong các phương án thực hiện và cách triển khai nêu trên được nêu.

Chi tiết về một hoặc nhiều phương án thực hiện được đề xuất trong các hình vẽ đi kèm và phần mô tả dưới đây. Các dấu hiệu, các đối tượng, và các ưu điểm khác sẽ trở nên rõ ràng trong phần mô tả, hình vẽ, và các điểm yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Dưới đây các phương án thực hiện sáng chế được mô tả chi tiết hơn dựa vào các hình vẽ đi kèm, trong đó:

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

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

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

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

Fig.4 là sơ đồ khối minh họa ví dụ của thiết bị mã hóa hoặc thiết bị giải mã;

Fig.5 là sơ đồ khối minh họa ví dụ khác của thiết bị mã hóa hoặc thiết bị giải mã;

Fig.6a là hình vẽ minh họa ví dụ của khối cùng vị trí;

Fig.6b là hình vẽ minh họa ví dụ của các khối lân cận không gian;

Fig.7 là hình vẽ minh họa một số ví dụ của chế độ dự báo tam giác;

Fig.8 là hình vẽ minh họa một số ví dụ của chế độ dự báo khối phụ;

Fig.9 đến Fig.12 là các hình vẽ thể hiện một số ví dụ về phân vùng của khối;

Fig.13 là hình vẽ minh họa cách triển khai ví dụ của bảng tra cứu định trước cho bướcD;

Fig.14 là hình vẽ minh họa cách triển khai ví dụ của bảng tra cứu định trước cho $f()$;

Fig.15 là hình vẽ minh họa ví dụ của khía cạnh lượng tử so với bảng tra cứu định trước cho bướcD;

Fig.16 là hình vẽ minh họa ví dụ của phương tiện lượng tử, trong đó khoảng cách lớn nhất $\rho_{\max}$ được định nghĩa đối với khối mã cụ thể;

Fig.17 là hình vẽ minh họa ví dụ của phương tiện lượng tử, trong đó khoảng cách lớn nhất tùy chọn $\rho_{\max}$ được định nghĩa đối với khối mã cụ thể;

Fig.18 là sơ đồ khối thể hiện cấu trúc của ví dụ của hệ thống cấp nội dung;

Fig.19 là sơ đồ thể hiện cấu trúc của ví dụ của thiết bị đầu cuối;

Fig.20 là hình vẽ minh họa phương án thực hiện sáng chế;

Fig.21 là hình vẽ minh họa phương án thực hiện khác của sáng chế;

Fig.22 là lưu đồ thể hiện phương pháp theo phương án thực hiện sáng chế;

Fig.23 là sơ đồ khối thể hiện thiết bị theo phương án thực hiện sáng chế; và

Fig.24 là sơ đồ khối thể hiện bộ giải mã theo phương án thực hiện sáng chế.

Các dấu hiệu tham chiếu giống hệt dưới đây đề cập đến các dấu hiệu giống hệt hoặc ít nhất tương đương về chức năng nếu không được xác định tường minh theo cách ngược lại.

Mô tả chi tiết sáng chế

Trong phần mô tả dưới đây, tham khảo các hình vẽ đi kèm, tạo thành một phần của sáng chế, và thể hiện, thông qua minh họa, các khía cạnh cụ thể của các phương án thực hiện sáng chế hoặc các khía cạnh cụ thể trong đó các phương án thực hiện sáng chế có thể được sử dụng. Cần hiểu rằng các phương án thực hiện sáng chế có thể được sử dụng trong các khía cạnh khác và bao gồm các thay đổi về cấu trúc hoặc logic không được mô tả trong các hình vẽ. Do vậy, phần mô tả chi tiết sau không được hiểu theo nghĩa giới hạn, và phạm vi của sáng chế được định nghĩa theo các điểm yêu cầu bảo hộ đi kèm.

Chẳng hạn, cần hiểu rằng sáng chế cùng với phương pháp được mô tả cũng có thể đúng đối với thiết bị hoặc hệ thống tương ứng được tạo cấu hình để thực

hiện phương pháp và ngược lại. Chẳng hạn, nếu một hoặc các bước của phương pháp cụ thể được mô tả, thiết bị tương ứng có thể bao gồm một hoặc các khối, chẳng hạn, các khối chức năng, để thực hiện một hoặc nhiều bước phương pháp được mô tả (chẳng hạn, một khối thực hiện một hoặc nhiều bước, hoặc các khối mà mỗi khối thực hiện một hoặc nhiều bước), thậm chí nếu một hoặc nhiều khối này được mô tả hoặc minh họa tường minh trên các hình vẽ. Ngược lại, chẳng hạn, nếu thiết bị cụ thể được mô tả dựa trên một hoặc nhiều khối, chẳng hạn, các khối chức năng, phương pháp tương ứng có thể bao gồm một bước để thực hiện chức năng của một hoặc các đơn vị (chẳng hạn, một bước thực hiện chức năng của một hoặc các đơn vị, hoặc các bước mà mỗi bước thực hiện chức năng của một hoặc nhiều khối), thậm chí nếu một hoặc các bước không được mô tả hoặc minh họa tường minh trong các hình vẽ. Ngoài ra, cần hiểu rằng các dấu hiệu của các phương án thực hiện và/hoặc các khía cạnh lấy làm ví dụ khác nhau được mô tả ở đây có thể được kết hợp với nhau, trừ khi được lưu ý cụ thể ngược lại.

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

Trong trường hợp tạo mã video không tổn hao, các ảnh video ban đầu có thể được tái tạo, các ảnh video được tái tạo có cùng chất lượng với các ảnh video ban đầu (giả sử không tổn hao truyền hoặc tổn hao dữ liệu khác trong khi lưu trữ hoặc truyền). Trong trường hợp tạo mã video có tổn hao, bước nén tiếp, chẳng hạn, bằng cách lượng tử, được thực hiện, để giảm lượng dữ liệu biểu diễn các ảnh video, mà không thể được tái tạo hoàn toàn ở bộ giải mã, chất lượng của các ảnh video đã được tái tạo thấp hơn hoặc tệ hơn so với chất lượng của các ảnh video ban đầu.

Vài chuẩn tạo mã video thuộc nhóm “các codec video lai có tổn hao” (chẳng hạn, kết hợp dự báo thời gian và không gian trong miền mẫu và mã hóa biến đổi 2D để áp dụng lượng tử trong miền biến đổi). Mỗi ảnh của chuỗi video thường được phân vùng thành tập hợp khối không chồng lấn và việc mã hóa thường được thực hiện ở mức khối. Nói cách khác, ở bộ mã hóa, video thường được xử lý, chẳng hạn, được mã hóa, ở mức khối (khối video), chẳng hạn, nhờ sử dụng dự báo không gian (trong ảnh) và/hoặc thời gian (ngoài ảnh) để tạo khối dự báo, lấy khối hiện tại (khối đang được xử lý/cần được xử lý) trừ đi khối dự báo để thu được khối dư, biến đổi khối dư và lượng tử khối dư trong miền biến đổi để giảm lượng dữ liệu cần được truyền (nén), cùng lúc ở bộ giải mã, bước xử lý ngược so với bộ mã hóa được áp dụng cho khối được mã hóa hoặc được nén để tái tạo khối hiện tại để biểu diễn. Ngoài ra, bộ mã hóa sao chép vòng xử lý giải mã sao cho cả hai sẽ tạo ra các dự báo giống hệt nhau (chẳng hạn, các dự báo trong và ngoài) và/hoặc các tái tạo để xử lý, chẳng hạn, mã hóa, các khối tiếp theo.

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

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

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

Thiết bị nguồn 12 bao gồm bộ mã hóa 20, và có thể còn, chẳng hạn, tùy chọn, bao gồm nguồn ảnh 16, bộ tiền xử lý (hoặc khối tiền xử lý) 18, chẳng hạn, bộ tiền xử lý ảnh 18, và giao diện truyền thông hoặc khối truyền thông 22.

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

Để phân biệt với bộ tiền xử lý 18 và quá trình xử lý được thực hiện bởi khối tiền xử lý 18, ảnh hoặc dữ liệu ảnh 17 cũng có thể được gọi là ảnh thô hoặc dữ liệu ảnh thô 17.

Bộ tiền xử lý 18 được tạo cấu hình để nhận dữ liệu ảnh (thô) 17 và để thực hiện tiền xử lý trên dữ liệu ảnh 17 để thu được ảnh được tiền xử lý 19 hoặc dữ liệu ảnh được tiền xử lý 19. Bước tiền xử lý được thực hiện bởi bộ tiền xử lý 18

có thể, chẳng hạn, bao gồm cắt tỉa, biến đổi định dạng màu (chẳng hạn, từ RGB sang YCbCr), hiệu chỉnh màu, hoặc khử nhiễu. Có thể hiểu rằng khối tiền xử lý 18 có thể là thành phần tùy chọn.

Bộ mã hóa video 20 được tạo cấu hình để nhận dữ liệu ảnh được tiền xử lý 19 và tạo dữ liệu ảnh được mã hóa 21 (các chi tiết khác sẽ được mô tả dưới đây, chẳng hạn, dựa trên Fig.2).

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

Thiết bị đích 14 bao gồm bộ giải mã 30 (chẳng hạn, bộ giải mã video 30), và có thể còn, chẳng hạn một cách tùy chọn, bao gồm giao diện truyền thông hoặc khối truyền thông 28, bộ hậu xử lý 32 (hoặc khối hậu xử lý 32) và bộ phận hiển thị 34.

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

Giao diện truyền thông 22 và giao diện truyền thông 28 có thể được tạo cấu hình để truyền hoặc nhận dữ liệu ảnh được mã hóa 21 hoặc dữ liệu được mã hóa 13 qua đường truyền thông trực tiếp giữa thiết bị nguồn 12 và thiết bị đích 14, chẳng hạn, kết nối không dây hoặc hữu tuyến trực tiếp, hoặc qua loại bất kỳ của mạng, chẳng hạn, mạng không dây hoặc hữu tuyến hoặc kết hợp bất kỳ của nó,

hoặc loại bất kỳ của mạng riêng và công cộng, hoặc loại bất kỳ của kết hợp của nó.

Giao diện truyền thông 22 có thể, chẳng hạn, được tạo cấu hình để đóng gói dữ liệu ảnh được mã hóa 21 thành định dạng thích hợp, chẳng hạn, các gói tin, và/hoặc xử lý dữ liệu ảnh được mã hóa nhờ sử dụng loại bất kỳ của mã hóa truyền hoặc xử lý để truyền trên đường truyền thông hoặc mạng truyền thông.

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

Cả giao diện truyền thông 22 lẫn giao diện truyền thông 28 có thể được tạo cấu hình dưới dạng các giao diện truyền thông đơn hướng như được chỉ báo bằng mũi tên cho kênh truyền thông 13 trên Fig.1A trở từ thiết bị nguồn 12 đến thiết bị đích 14, hoặc các giao diện truyền thông hai hướng, và có thể được tạo cấu hình, chẳng hạn, để truyền và nhận thông điệp, chẳng hạn, để thiết lập kết nối, để báo nhận và trao đổi bất kỳ thông tin khác liên quan đến đường truyền thông và/hoặc truyền dữ liệu, chẳng hạn, truyền dữ liệu ảnh được mã hóa.

Bộ giải mã 30 được tạo cấu hình để nhận dữ liệu ảnh được mã hóa 21 và tạo dữ liệu ảnh được giải mã 31 hoặc ảnh được giải mã 31 (các chi tiết khác sẽ được mô tả dưới đây, chẳng hạn, dựa trên Fig.3 hoặc Fig.5).

Bộ hậu xử lý 32 của thiết bị đích 14 được tạo cấu hình để hậu xử lý dữ liệu ảnh được giải mã 31 (cũng được gọi là dữ liệu ảnh được tái tạo), chẳng hạn, ảnh được giải mã 31, để thu được dữ liệu ảnh được hậu xử lý 33, chẳng hạn, ảnh được hậu xử lý 33. Việc hậu xử lý được thực hiện bởi khối hậu xử lý 32 có thể bao gồm, chẳng hạn, biến đổi định dạng màu (chẳng hạn, từ YCbCr đến RGB), hiệu chỉnh màu, cắt tia, hoặc lấy mẫu lại, hoặc bất kỳ xử lý khác, chẳng hạn, để

chuẩn bị dữ liệu ảnh được giải mã 31 để hiển thị, chẳng hạn, nhờ bộ phận hiển thị 34.

Bộ phận hiển thị 34 của thiết bị đích 14 được tạo cấu hình để nhận dữ liệu ảnh được hậu xử lý 33 để hiển thị ảnh, chẳng hạn, đến người dùng hoặc người xem. Bộ phận hiển thị 34 có thể là hoặc bao gồm loại bất kỳ của bộ phận hiển thị để biểu diễn ảnh được tái tạo, chẳng hạn, màn hình hoặc bộ phận hiển thị tích hợp hoặc bên ngoài. Các bộ phận hiển thị có thể, chẳng hạn, bao gồm màn hình tinh thể lỏng (liquid crystal display, LCD), màn hình điốt phát quang hữu cơ (organic light emitting diode, OLED), màn hình plasma, máy chiếu, màn hình microLED, màn hình tinh thể lỏng trên silic (liquid crystal on silicon, LCoS), bộ xử lý ánh sáng số (digital light processor, DLP) hoặc loại hiển thị khác bất kỳ.

Mặc dù Fig.1A mô tả thiết bị nguồn 12 và thiết bị đích 14 như là các thiết bị riêng rẽ, song các thiết bị theo các phương án thực hiện cũng có thể bao gồm cả hai chức năng, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng. Theo các phương án thực hiện, thiết bị nguồn 12 hoặc chức năng tương ứng và thiết bị đích 14 hoặc chức năng tương ứng có thể được thực hiện nhờ sử dụng phần cứng và/hoặc phần mềm tương tự hoặc bằng phần cứng và/hoặc phần mềm riêng rẽ hoặc kết hợp bất kỳ của nó.

Người có hiểu biết trung bình trong lĩnh vực sẽ hiểu dựa trên phần mô tả này, sự tồn tại và tách (chính xác) các chức năng của các đơn vị hoặc các chức năng khác nhau trong thiết bị nguồn 12 và/hoặc thiết bị đích 14 như được thể hiện trên Fig.1A có thể thay đổi tùy thuộc vào thiết bị và ứng dụng thực.

Bộ mã hóa 20 (chẳng hạn, bộ mã hóa video 20) hoặc bộ giải mã 30 (chẳng hạn, bộ giải mã video 30) hoặc cả hai bộ mã hóa 20 và bộ giải mã 30 có thể được thực hiện qua hệ mạch xử lý như được thể hiện trên Fig. 1B, chẳng hạn một hoặc nhiều bộ vi xử lý, bộ xử lý tín hiệu số (digital signal processor, DSP), mạch tích

hợp ứng dụng cụ thể (application-specific integrated circuit, ASIC), mảng cổng dạng trường lập trình được (field-programmable gate array, FPGA), logic rời rạc, phần cứng, tạo mã video dành riêng hoặc bất kỳ kết hợp nào của nó. Bộ mã hóa 20 có thể được thực hiện qua hệ mạch xử lý 46 để triển khai các môđun khác nhau như được nêu dựa vào bộ mã hóa 20 trên Fig.2 và/hoặc bất kỳ hệ thống hoặc hệ thống phụ mã hóa khác được mô tả ở đây. Bộ giải mã 30 có thể được thực hiện qua hệ mạch xử lý 46 để triển khai các môđun khác nhau như được nêu dựa vào bộ giải mã 30 trên Fig.3 và/hoặc bất kỳ hệ thống hoặc hệ thống phụ giải mã khác được mô tả ở đây. Hệ mạch xử lý có thể be được tạo cấu hình để thực hiện các hoạt động khác nhau như được nêu sau đây. Như được thể hiện trên Fig.5, nếu các kỹ thuật được thực hiện một phần trong phần mềm, thiết bị có thể lưu trữ các lệnh cho phần mềm trong vật ghi máy tính đọc được bất biến thích hợp và có thể thực thi các lệnh trong phần cứng nhờ sử dụng một hoặc nhiều bộ xử lý để thực hiện các kỹ thuật của sáng chế. Một trong bộ mã hóa video 20 và bộ giải mã video 30 có thể được tích hợp như là một phần của bộ mã hóa/bộ giải mã kết hợp (CODEC) trong một thiết bị, chẳng hạn, như được thể hiện trên Fig.1B.

Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm bất kỳ thiết bị trong các loại thiết bị, bao gồm loại thiết bị bất kỳ trong các thiết bị cầm tay hoặc tĩnh, chẳng hạn, notebook hoặc máy tính xách tay, điện thoại di động, điện thoại thông minh, tablet hoặc máy tính bảng, camera, máy tính để bàn, hộp tín hiệu truyền hình cáp (set-top box), TV, các bộ phận hiển thị, máy phát media số, bảng điều khiển trò chơi video, các thiết bị phát trực tuyến video (chẳng hạn, máy chủ dịch vụ nội dung hoặc máy chủ phân phối nội dung), bộ thu phát sóng phát thanh, bộ truyền phát thanh, hoặc tương tự và có thể không sử dụng hoặc loại hệ điều hành bất kỳ. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể

được trang bị cho truyền thông không dây. Do vậy, thiết bị nguồn 12 và thiết bị đích 14 có thể là các thiết bị truyền thông không dây.

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

Để tiện mô tả, các phương án thực hiện sáng chế được mô tả ở đây, chẳng hạn, bằng cách viện dẫn đến chuẩn tạo mã video hiệu suất cao (High-Efficiency video coding, HEVC) hoặc đến phần mềm tham chiếu của chuẩn tạo mã video đa dụng (Versatile video coding, VVC), chuẩn tạo mã video thế hệ tiếp theo được phát triển bởi nhóm cộng tác chung về tạo mã video (Joint Collaboration Team on video coding, JCT-VC) của nhóm chuyên gia tạo mã video (Video Coding Experts Group, VCEG) ITU-T và nhóm chuyên gia ảnh động (Motion Picture Experts Group, MPEG) ISO/IEC. Người có hiểu biết trung bình trong lĩnh vực sẽ hiểu rằng các phương án thực hiện sáng chế không bị giới hạn ở HEVC hoặc VVC.

Bộ mã hóa và phương pháp mã hóa

Fig. 2 thể hiện sơ đồ khối của bộ mã hóa video 20 lấy làm ví dụ được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Trong ví dụ trên Fig.2, bộ mã hóa video 20 bao gồm đầu vào 201 (hoặc giao diện nhập 201), khối tính toán phân

đư 204, khối xử lý biến đổi 206, khối lượng tử 208, khối lượng tử ngược 210, và khối xử lý biến đổi ngược 212, khối tái tạo 214, khối lọc vòng 220, bộ đệm ảnh được giải mã (decoded picture buffer, DPB) 230, khối chọn chế độ 260, khối mã hóa entropy 270 và đầu ra 272 (hoặc giao diện xuất 272). Khối chọn chế độ 260 có thể bao gồm khối dự báo ngoài 244, khối dự báo trong 254 và khối phân vùng 262. Khối dự báo ngoài 244 có thể bao gồm khối ước tính chuyển động và khối bù chuyển động (không được thể hiện). Bộ mã hóa video 20 như được thể hiện trên Fig.2 có thể cũng được gọi là bộ mã hóa video lai hoặc bộ mã hóa video theo codec video lai.

Khối tính toán phần dư 204, khối xử lý biến đổi 206, khối lượng tử 208, khối chọn chế độ 260 có thể được gọi là tạo đường tín hiệu thuận của bộ mã hóa 20, cùng lúc khối lượng tử ngược 210, khối xử lý biến đổi ngược 212, khối tái tạo 214, bộ đệm 216, bộ lọc vòng 220, DPB 230, khối dự báo ngoài 244 và khối dự báo trong 254 có thể được gọi là tạo đường tín hiệu ngược của bộ mã hóa video 20, trong đó đường tín hiệu ngược của bộ mã hóa video 20 tương ứng với đường tín hiệu của bộ giải mã (xem bộ giải mã video 30 trên Fig.3). Khối lượng tử ngược 210, khối xử lý biến đổi ngược 212, khối tái tạo 214, bộ lọc vòng 220, DPB 230, khối dự báo ngoài 244 và khối dự báo trong 254 cũng được gọi là tạo “bộ giải mã cài sẵn” của bộ mã hóa video 20.

Ảnh & Phân vùng ảnh (ảnh & khối)

Bộ mã hóa 20 có thể được tạo cấu hình để nhận, chẳng hạn, qua đầu vào 201, ảnh 17 (hoặc dữ liệu ảnh 17), chẳng hạn, ảnh trong chuỗi ảnh tạo video hoặc chuỗi video. Ảnh được nhận hoặc dữ liệu ảnh cũng có thể là ảnh được tiền xử lý 19 (hoặc dữ liệu ảnh được tiền xử lý 19). Để cho đơn giản, phần mô tả sau đề cập đến ảnh 17. Ảnh 17 có thể cũng được gọi là ảnh hiện tại hoặc ảnh cần được tạo mã (cụ thể là trong chuẩn tạo mã video để phân biệt ảnh hiện tại với các ảnh

khác, chẳng hạn các ảnh được mã hóa trước đó và/hoặc được giải mã của cùng chuỗi video, chẳng hạn, chuỗi video cũng bao gồm ảnh hiện tại).

Ảnh (số) là hoặc có thể được xem là mảng hai chiều hoặc ma trận mẫu có các giá trị tăng cường. Mẫu trong mảng có thể cũng được gọi là pixel (gọi tắt của phần tử ảnh) hoặc pel. Số lượng mẫu theo chiều ngang và dọc (hoặc trục) của mảng hoặc ảnh định nghĩa kích thước và/hoặc độ phân giải của ảnh. Để biểu diễn màu, ba thành phần màu được sử dụng, chẳng hạn, ảnh có thể được biểu diễn hoặc bao gồm ba mảng màu. Trong định dạng RGB hoặc không gian màu, ảnh bao gồm mảng mẫu đỏ, lục và lam tương ứng. Tuy nhiên, trong chuẩn tạo mã video, mỗi pixel được biểu diễn đặc trưng theo định dạng độ sáng và sắc độ hoặc không gian màu sắc, chẳng hạn, YCbCr, bao gồm thành phần độ sáng được chỉ báo bởi Y (đôi lúc L cũng được sử dụng thay thế) và hai thành phần sắc độ được chỉ báo bởi Cb và Cr. Thành phần độ sáng (hoặc gọi tắt là luma) Y biểu diễn độ sáng hoặc cường độ mức xám (chẳng hạn, giống như trong ảnh thang xám), trong khi hai thành phần sắc độ (hoặc gọi tắt là chroma) Cb và Cr biểu diễn sắc sai hoặc thành phần thông tin màu. Tương ứng, ảnh ở định dạng YCbCr bao gồm mảng mẫu độ sáng của các giá trị mẫu độ sáng (Y), và hai mảng mẫu sắc độ của các giá trị sắc độ (Cb và Cr). Các ảnh ở định dạng RGB có thể được biến đổi thành định dạng YCbCr và ngược lại, quá trình này cũng được biết đến như là biến đổi màu sắc. Nếu ảnh là đơn sắc, ảnh có thể bao gồm chỉ mảng mẫu độ sáng. Tương ứng, ảnh có thể là, chẳng hạn, mảng của các mẫu độ sáng theo định dạng đơn sắc hoặc mảng của các mẫu độ sáng và hai mảng tương ứng của các mẫu sắc độ theo định dạng màu 4:2:0, 4:2:2, và 4:4:4.

Theo các phương án thực hiện, bộ mã hóa video 20 có thể bao gồm khối phân vùng ảnh (không được mô tả trên Fig.2) được tạo cấu hình để phân vùng ảnh 17 thành các khối ảnh (thường là không chồng lấn) 203. Các khối này có thể cũng

được gọi là các khối gốc, các khối lớn (H.264/AVC) hoặc các khối cây mã (coding tree block, CTB) hoặc các đơn vị cây mã (coding tree unit, CTU) (H.265/HEVC và VVC). Khối phân vùng ảnh có thể được tạo cấu hình để sử dụng cùng kích thước khối cho tất cả các ảnh của chuỗi video và lưới tương ứng định nghĩa kích thước khối, hoặc thay đổi kích thước khối giữa các ảnh hoặc các tập con hoặc các nhóm ảnh, và phân vùng mỗi ảnh thành các khối tương ứng.

Theo các phương án thực hiện khác, bộ mã hóa video có thể được tạo cấu hình để nhận trực tiếp khối 203 của ảnh 17, chẳng hạn, một, vài hoặc tất cả các khối tạo ảnh 17. Khối ảnh 203 có thể cũng được gọi là khối ảnh hiện tại hoặc khối ảnh cần được tạo mã.

Giống như ảnh 17, khối ảnh 203 lại là hoặc có thể được xem là mảng hai chiều hoặc ma trận mẫu có các giá trị cường độ (các giá trị mẫu), mặc dù có kích thước nhỏ hơn ảnh 17. Nói cách khác, khối 203 có thể bao gồm, chẳng hạn, một mảng mẫu (chẳng hạn, mảng độ sáng trong trường hợp ảnh đơn sắc 17, hoặc mảng độ sáng hoặc sắc độ trong trường hợp ảnh màu) hoặc ba mảng mẫu (chẳng hạn, mảng độ sáng và hai mảng sắc độ trong trường hợp ảnh màu 17) hoặc số lượng khác bất kỳ và/hoặc loại mảng tùy thuộc vào định dạng màu được áp dụng. Số lượng mẫu theo chiều ngang và dọc (hoặc trục) của khối 203 định nghĩa kích thước của khối 203. Tương ứng, khối có thể, chẳng hạn, mảng mẫu $M \times N$ (M cột x N hàng), hoặc mảng $M \times N$ của các hệ số biến đổi.

Theo các phương án thực hiện, bộ mã hóa video 20 như được thể hiện trên Fig. 2 có thể được tạo cấu hình để mã hóa ảnh 17 theo từng khối, chẳng hạn, mã hóa và dự báo được thực hiện theo khối 203.

Tính toán phần dư

Khối tính toán phần dư 204 có thể được tạo cấu hình để tính toán khối dư 205 (cũng được gọi là phần dư 205) dựa trên khối ảnh 203 và khối dự báo 265 (các chi tiết khác về khối dự báo 265 được nêu dưới đây), chẳng hạn lấy các giá trị mẫu của khối ảnh 203 trừ đi các giá trị mẫu của khối dự báo 265, theo từng mẫu (theo từng pixel để thu được khối dư 205 trong miền mẫu).

Biến đổi

Khối xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng phép biến đổi, chẳng hạn, biến đổi côsin rời rạc (discrete cosine transform, DCT) hoặc biến đổi hình sin rời rạc (discrete sine transform, DST), trên các giá trị mẫu của khối dư 205 để thu được các hệ số biến đổi 207 trong miền biến đổi. Các hệ số biến đổi 207 có thể cũng được gọi là các hệ số dư biến đổi và biểu diễn khối dư 205 trong miền biến đổi.

Khối xử lý biến đổi 206 có thể được tạo cấu hình để áp dụng các phép xấp xỉ số nguyên của DCT/DST, chẳng hạn các phép biến đổi được xác định cho H.265/HEVC. So với phép biến đổi DCT trực giao, các phép xấp xỉ số nguyên này thường được chia tỷ lệ theo hệ số cụ thể. Để bảo toàn chuẩn của khối dư được xử lý bằng các phép biến đổi thuận và nghịch, các hệ số chia tỷ lệ bổ sung được áp dụng như là một phần của quá trình biến đổi. Các hệ số chia tỷ lệ thường được chọn dựa trên các ràng buộc cụ thể như các hệ số chia tỷ lệ là lũy thừa của hai đối với các phép dịch, chiều sâu bit của các hệ số biến đổi, cân bằng giữa độ chính xác và chi phí triển khai, v.v.. Các hệ số chia tỷ lệ cụ thể, chẳng hạn, được xác định cho biến đổi ngược, chẳng hạn, bằng khối xử lý biến đổi ngược 212 (và phép biến đổi ngược tương ứng, chẳng hạn, bằng khối xử lý biến đổi ngược 312 ở bộ giải mã video 30) và các hệ số chia tỷ lệ tương ứng đối với phép biến đổi

thuận, chẳng hạn, bằng khối xử lý biến đổi 206, ở bộ mã hóa 20 có thể được xác định tương ứng.

Theo các phương án thực hiện, bộ mã hóa video 20 (tương ứng là khối xử lý biến đổi 206) có thể được tạo cấu hình để xuất ra các tham số biến đổi, chẳng hạn, loại biến đổi hoặc các phép biến đổi, chẳng hạn, trực tiếp hoặc được nén hoặc được mã hóa qua khối mã hóa entropy 270, sao cho, chẳng hạn, bộ giải mã video 30 có thể nhận và sử dụng các tham số biến đổi để giải mã.

Lượng tử

Khối lượng tử 208 có thể được tạo cấu hình để lượng tử các hệ số biến đổi 207 để thu được các hệ số được lượng tử 209, chẳng hạn, bằng cách áp dụng lượng tử vô hướng hoặc lượng tử vector. Các hệ số được lượng tử 209 có thể cũng được gọi là các hệ số biến đổi được lượng tử 209 hoặc các hệ số dư được lượng tử 209.

Quá trình lượng tử có thể giảm chiều sâu bit được liên kết với một số hoặc tất cả các hệ số biến đổi 207. Chẳng hạn, hệ số biến đổi n bit có thể được làm tròn xuống hệ số biến đổi m bit trong khi lượng tử, trong đó n lớn hơn m . Mức độ lượng tử có thể được chỉnh sửa bằng cách điều chỉnh tham số lượng tử (quantization parameter, QP). Chẳng hạn để lượng tử vô hướng, phép chia tỷ lệ khác nhau có thể được áp dụng để lượng tử mịn hơn hoặc thô hơn. Các kích thước bậc lượng tử nhỏ hơn tương ứng với lượng tử mịn hơn, trong đó các kích thước bậc lượng tử lớn hơn tương ứng với lượng tử thô hơn. Kích thước bậc lượng tử áp dụng được có thể được chỉ báo bởi QP. QP có thể chẳng hạn là chỉ số đến tập hợp định trước của các kích thước bậc lượng tử có thể áp dụng. Chẳng hạn, các QP nhỏ có thể tương ứng với lượng tử mịn (các kích thước bậc lượng tử nhỏ) và các QP lớn có thể tương ứng với lượng tử thô (các kích thước bậc lượng tử lớn) hoặc ngược lại. Lượng tử có thể bao gồm phép chia cho kích thước

bậc lượng tử và lượng tử ngược tương ứng, chẳng hạn, bằng khối lượng tử ngược 210, có thể bao gồm phép nhân với kích thước bậc lượng tử. Các phương án thực hiện theo một số chuẩn, chẳng hạn, HEVC, có thể được tạo cấu hình để sử dụng QP để xác định kích thước bậc lượng tử. Nói chung, kích thước bậc lượng tử có thể được tính toán dựa trên QP nhờ sử dụng phép xấp xỉ dấu chấm tĩnh của phương trình bao gồm phép chia. Các hệ số chia tỷ lệ bổ sung có thể được đưa vào để lượng tử và lượng tử ngược để khôi phục chuẩn của khối dư, mà có thể đã được chỉnh sửa do phép chia tỷ lệ được sử dụng trong phép xấp xỉ dấu chấm tĩnh của phương trình đối với kích thước bậc lượng tử và QP. Trong một cách triển khai lấy làm ví dụ, phép chia tỷ lệ của phép biến đổi ngược và phép lượng tử ngược có thể được kết hợp. Theo cách khác, các bảng lượng tử được tùy chỉnh có thể được sử dụng và được báo hiệu từ bộ mã hóa đến bộ giải mã, chẳng hạn trong dòng bit. Lượng tử là phép toán tổn hao, trong đó tổn hao tăng với việc tăng các kích thước bậc lượng tử.

Theo các phương án thực hiện, bộ mã hóa video 20 (tương ứng với khối lượng tử 208) có thể được tạo cấu hình để xuất ra các QP, chẳng hạn, trực tiếp hoặc được mã hóa qua khối mã hóa entropy 270, sao cho, chẳng hạn, bộ giải mã video 30 có thể nhận và áp dụng các QP để giải mã.

Lượng tử ngược

Khối lượng tử ngược 210 được tạo cấu hình để áp dụng lượng tử ngược của khối lượng tử 208 lên các hệ số được lượng tử để thu được các hệ số được lượng tử ngược 211, chẳng hạn, bằng áp dụng đảo của phương tiện lượng tử được áp dụng bởi khối lượng tử 208 dựa trên hoặc sử dụng kích thước bậc lượng tử giống như khối lượng tử 208. Các hệ số được lượng tử ngược 211 có thể cũng được gọi là các hệ số dư được lượng tử ngược 211 và tương ứng – mặc dù không hoàn

toàn giống hệt với các hệ số biến đổi do tổn hao bởi phép lượng tử - với các hệ số biến đổi 207.

Biến đổi ngược

Khối xử lý biến đổi ngược 212 được tạo cấu hình để áp dụng biến đổi ngược của phép biến đổi được áp dụng bởi khối xử lý biến đổi 206, chẳng hạn, phép biến đổi cô sin rời rạc ngược (discrete cosine transform, DCT) hoặc biến đổi sin rời rạc (discrete sine transform, DST) ngược hoặc các phép biến đổi ngược khác, để thu được khối dư được tái tạo 213 (hoặc các hệ số được lượng tử ngược 213 tương ứng) trong miền mẫu. Khối dư được tái tạo 213 có thể cũng được gọi là khối biến đổi 213.

Tái tạo

Khối tái tạo 214 (chẳng hạn, bộ cộng 214) được tạo cấu hình để bổ sung khối biến đổi 213 (khối dư được tái tạo 213) vào khối dự báo 265 để thu được khối được tái tạo 215 trong miền mẫu, chẳng hạn, bằng cách bổ sung – theo từng mẫu - các giá trị mẫu của khối dư được tái tạo 213 và các giá trị mẫu của khối dự báo 265.

Lọc

Khối lọc vòng 220 (hoặc gọi tắt là “lọc vòng” 220), được tạo cấu hình để lọc khối được tái tạo 215 để thu được khối được lọc 221, hoặc nói chung, để lọc các mẫu được tái tạo để thu được các mẫu được lọc. Khối lọc vòng, chẳng hạn, được tạo cấu hình để làm mượt các dịch chuyển pixel, hoặc ngược lại cải thiện chất lượng video. Khối lọc vòng 220 có thể bao gồm một hoặc nhiều bộ lọc vòng, chẳng hạn, bộ lọc tách khối (de-blocking filter, DBF), bộ lọc độ lệch mẫu thích

ứng (sample-adaptive offset, SAO) hoặc một hoặc nhiều bộ lọc khác, chẳng hạn, bộ lọc hai bên, bộ lọc vòng thích ứng (adaptive loop filter, ALF), bộ lọc làm nét, bộ lọc làm mượt hoặc bộ lọc cộng tác, hoặc kết hợp bất kỳ của nó. Mặc dù khối lọc vòng 220 được thể hiện trên Fig.2 như là bộ lọc trong vòng, trong các cấu hình khác, khối lọc vòng 220 có thể được thực hiện như là bộ lọc hậu vòng. khối được lọc 221 có thể cũng được gọi là khối được tái tạo được lọc 221.

Theo các phương án thực hiện, bộ mã hóa video 20 (tương ứng là khối lọc vòng 220) có thể được tạo cấu hình để xuất ra các tham số lọc vòng (chẳng hạn, thông tin SAO), chẳng hạn, trực tiếp hoặc được mã hóa qua khối mã hóa entropy 270, sao cho, chẳng hạn, bộ giải mã 30 có thể nhận và áp dụng các tham số lọc vòng tương tự hoặc các bộ lọc vòng tương ứng để giải mã.

Bộ đệm ảnh được giải mã

DPB 230 có thể là bộ nhớ lưu trữ các ảnh tham chiếu, hoặc nói chung dữ liệu ảnh tham chiếu, để mã hóa dữ liệu video bởi bộ mã hóa video 20. DPB 230 có thể được tạo bởi thiết bị bất kỳ trong các thiết bị nhớ, chẳng hạn, bộ nhớ truy nhập ngẫu nhiên động (dynamic random access memory, DRAM), bao gồm DRAM đồng bộ (synchronous DRAM, SDRAM), RAM từ trở (magnetoresistive RAM, MRAM), RAM điện trở (resistive RAM, RRAM), hoặc các loại thiết bị nhớ khác. DPB 230 có thể được tạo cấu hình để lưu trữ một hoặc nhiều khối được lọc 221. DPB 230 có thể còn được tạo cấu hình để lưu trữ các khối được lọc khác trước đó, chẳng hạn, các khối được lọc và được tái tạo trước đó 221, của cùng ảnh hiện tại hoặc của các ảnh khác nhau, chẳng hạn, các ảnh được tái tạo trước đó, và có thể tạo các ảnh được tái tạo hoàn chỉnh trước đó, chẳng hạn, được giải mã (và các khối tham chiếu tương ứng và các mẫu) và/hoặc ảnh hiện tại được tái tạo một phần (và các khối và các mẫu tham chiếu tương ứng), chẳng

hạn để dự báo ngoài. DPB 230 có thể cũng được tạo cấu hình để lưu trữ một hoặc nhiều khối được tái tạo không được lọc 215, hoặc nói chung các mẫu được tái tạo không được lọc, chẳng hạn, nếu khối được tái tạo 215 không được lọc bởi khối lọc vòng 220, hoặc phiên bản được xử lý khác nữa của các khối được tái tạo hoặc các mẫu.

Chọn chế độ (Phân vùng & Dự báo)

Khối chọn chế độ 260 bao gồm khối phân vùng 262, khối dự báo ngoài 244 và khối dự báo trong 254, và được tạo cấu hình để nhận hoặc thu được dữ liệu ảnh ban đầu, chẳng hạn, khối ban đầu 203 (khối hiện tại 203 của ảnh hiện tại 17), và dữ liệu ảnh được tái tạo, chẳng hạn, các mẫu được tái tạo được lọc và/hoặc không được lọc hoặc các khối của cùng ảnh (hiện tại) và/hoặc từ một hoặc các ảnh được giải mã trước đó, chẳng hạn, từ bộ đệm ảnh được giải mã 230 hoặc các bộ đệm khác (chẳng hạn, bộ đệm dòng, không được thể hiện). Dữ liệu ảnh được tái tạo được sử dụng làm dữ liệu ảnh tham chiếu để dự báo, chẳng hạn, dự báo ngoài hoặc dự báo trong, để thu được khối dự báo 265 hoặc bộ dự báo 265.

Khối chọn chế độ 260 có thể được tạo cấu hình để xác định hoặc chọn phân vùng cho chế độ dự báo khối hiện tại (không bao gồm phân vùng) và chế độ dự báo (chẳng hạn, chế độ dự báo trong hoặc ngoài) và tạo khối dự báo 265 tương ứng, được sử dụng để tính toán khối dư 205 và để tái tạo khối được tái tạo 215.

Theo các phương án thực hiện, khối chọn chế độ 260 có thể được tạo cấu hình để chọn chế độ phân vùng và dự báo (chẳng hạn, từ các chế độ được hỗ trợ hoặc khả dụng cho khối chọn chế độ 260), cho so khớp tốt nhất hoặc nói cách khác phần dư nhỏ nhất (phần dư nhỏ nhất nghĩa là nén tốt hơn để truyền hoặc lưu trữ), hoặc phụ tải báo hiệu nhỏ nhất (phụ tải báo hiệu nhỏ nhất nghĩa là nén tốt hơn để truyền hoặc lưu trữ), hoặc xem xét hoặc cân bằng cả hai. Khối chọn

chế độ 260 có thể được tạo cấu hình để xác định chế độ phân vùng và dự báo dựa trên (rate distortion optimization, RDO), chẳng hạn, lựa chọn chế độ dự báo cho méo dạng tốc độ nhỏ nhất. Các cụm từ như “tốt nhất”, “nhỏ nhất”, “tối ưu” v.v. trong ngữ cảnh này không nhất thiết đề cập đến toàn bộ “tốt nhất”, “nhỏ nhất”, “tối ưu”, v.v. nhưng cũng có thể đề cập đến việc hoàn thiện tiêu chí kết thúc hoặc chọn giống như giá trị lớn hơn hoặc nhỏ hơn ngưỡng hoặc các ràng buộc hác có thể dẫn đến “lựa chọn tối ưu phụ” nhưng giảm độ phức tạp và thời gian xử lý.

Nói cách khác, khối phân vùng 262 có thể được tạo cấu hình để phân vùng khối 203 thành các phân vùng khối nhỏ hơn hoặc các khối phụ (lại tạo thành các khối), chẳng hạn, sử dụng lặp lại phân vùng cây tứ phân (quad-tree, QT), phân vùng cây nhị phân (binary tree, BT) hoặc phân vùng cây tam phân (triple-tree, TT) hoặc kết hợp bất kỳ của nó, và để thực hiện, chẳng hạn, dự báo cho mỗi trong số các phân vùng khối hoặc các khối phụ, trong đó việc chọn chế độ bao gồm lựa chọn cấu trúc cây của khối được phân vùng 203 và các chế độ dự báo được áp dụng cho mỗi trong số các phân vùng khối hoặc các khối phụ.

Dưới đây, quá trình phân vùng (chẳng hạn, bằng khối phân vùng 260) và dự báo (bởi khối dự báo ngoài 244 và khối dự báo trong 254) được thực hiện bởi bộ mã hóa video 20 ví dụ sẽ được giải thích chi tiết hơn.

Phân vùng

Khối phân vùng 262 có thể phân vùng (hoặc tách) khối hiện tại 203 thành các phân vùng nhỏ hơn, chẳng hạn các khối nhỏ hơn có kích thước hình vuông hoặc hình chữ nhật. Các khối nhỏ hơn này (có thể cũng được gọi là các khối phụ) có thể còn được phân vùng thành các phân vùng thậm chí nhỏ hơn. Đây cũng được gọi là phân vùng cây hoặc phân vùng cây phân cấp, trong đó khối gốc, chẳng

hạn, ở mức gốc cây 0 (mức phân cấp 0, chiều sâu 0), có thể được phân vùng đệ quy, chẳng hạn, được phân vùng thành hai hoặc nhiều khối của mức cây thấp hơn tiếp theo, chẳng hạn, các nút ở mức cây 1 (mức phân cấp 1, chiều sâu 1), trong đó các khối này có thể lại được phân vùng thành hai hoặc nhiều khối của mức thấp hơn tiếp theo, chẳng hạn, mức cây 2 (mức phân cấp 2, chiều sâu 2), v.v. cho đến khi kết thúc phân vùng, chẳng hạn, do hoàn thành tiêu chí kết thúc, chẳng hạn, chiều sâu cây lớn nhất hoặc kích thước khối nhỏ nhất đạt đến. Các khối không được phân vùng tiếp cũng được gọi là các khối lá hoặc các nút lá của cây. Cây sử dụng phân vùng thành hai phân vùng được gọi là cây nhị phân (binary-tree, BT), cây sử dụng phân vùng thành ba phân vùng được gọi là cây tam phân (ternary-tree, TT), và cây sử dụng phân vùng thành bốn phân vùng được gọi là cây tứ phân (quad-tree, QT).

Như được nêu trước đó, cụm từ “khối” như được sử dụng ở đây có thể là phần, cụ thể là phần vuông hoặc chữ nhật, của ảnh. Chẳng hạn, dựa vào HEVC và VVC, khối có thể là hoặc tương ứng với đơn vị cây tạo mã (coding tree unit, CTU), đơn vị tạo mã (coding unit, CU), đơn vị dự báo (prediction unit, PU), và đơn vị biến đổi (transform unit, TU) và/hoặc với các khối tương ứng, chẳng hạn, khối cây mã (coding tree block, CTB), khối tạo mã (Coding Block, CB), khối biến đổi (Transform Block, TB) hoặc khối dự báo (prediction block, PB).

Chẳng hạn, CTU có thể là hoặc bao gồm CTB của các mẫu độ sáng, hai CTB tương ứng của các mẫu sắc độ của ảnh có ba mảng mẫu, hoặc CTB của các mẫu của ảnh đơn sắc hoặc ảnh được mã hóa nhờ sử dụng ba mặt phẳng màu riêng rẽ và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu. Một cách tương ứng, CTB có thể là khối $N \times N$ của các mẫu đối với giá trị bất kỳ của N sao cho việc phân chia thành phần thành các CTB là cách phân vùng. CU có thể là hoặc bao gồm khối mã của các mẫu độ sáng, hai CB tương ứng của các mẫu sắc độ của

ảnh có ba mảng mẫu, hoặc khối mã của các mẫu của ảnh đơn sắc hoặc ảnh được mã hóa nhờ sử dụng ba mặt phẳng màu riêng rẽ và các cấu trúc cú pháp được sử dụng để mã hóa các mẫu. Một cách tương ứng, CB có thể là khối $M \times N$ của các mẫu đối với một số giá trị của M và N sao cho việc phân chia CTB thành các CB là phân vùng.

Theo các phương án thực hiện, chẳng hạn, theo HEVC, CTU có thể được tách thành các CU nhờ sử dụng cấu trúc QT được ký hiệu là cây mã. Quyết định liệu có mã hóa diện tích ảnh nhờ sử dụng dự báo ngoài ảnh (thời gian) hoặc trong ảnh (không gian) được thực hiện ở mức CU. Mỗi CU có thể còn được tách thành một, hai hoặc bốn PU theo loại tách PU. Bên trong một PU, quá trình dự báo tương tự được áp dụng và thông tin liên quan được truyền đến bộ giải mã dựa trên PU. Sau khi thu được khối dư bằng cách áp dụng quá trình dự báo dựa trên loại tách PU, CU có thể được phân vùng thành các TU theo cấu trúc QT khác giống như cây mã cho CU.

Theo các phương án thực hiện, chẳng hạn, theo chuẩn tạo mã video mới nhất đang được phát triển, được gọi là VVC, phân vùng cây tứ phân và cây nhị phân (Quad-tree and binary tree, QTBT) được sử dụng để phân vùng khối mã. Trong cấu trúc khối QTBT, CU có thể có hình vuông hoặc chữ nhật. Chẳng hạn, CTU trước hết được phân vùng bằng cấu trúc QT. Các nút lá QT còn được phân vùng bằng cấu trúc BT hoặc TT. Các nút lá cây phân vùng được gọi là các CU, và phân đoạn đó được sử dụng để xử lý dự báo và biến đổi mà không phân vùng tiếp. Điều này nghĩa là CU, PU và TU có cùng kích thước khối trong cấu trúc CB QTBT. Một cách song song, nhiều phân vùng, chẳng hạn, phân vùng TT cũng được đề xuất để được sử dụng với cấu trúc khối QTBT.

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

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

Dự báo trong

Tập hợp chế độ dự báo trong có thể bao gồm 35 chế độ dự báo trong khác nhau, chẳng hạn, các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ định hướng, chẳng hạn, như được định nghĩa trong HEVC, hoặc có thể bao gồm 67 chế độ dự báo trong khác nhau, chẳng hạn, các chế độ vô hướng như chế độ DC (hoặc trung bình) và chế độ phẳng, hoặc các chế độ hướng, chẳng hạn, như được định nghĩa cho VVC.

Khối dự báo trong 254 được tạo cấu hình để sử dụng các mẫu được tái tạo của các khối lân cận của cùng ảnh hiện tại để tạo khối dự báo trong 265 theo chế độ dự báo trong của tập hợp chế độ dự báo trong.

Khối dự báo trong 254 (hoặc nói chung khối chọn chế độ 260) còn được tạo cấu hình để xuất các tham số dự báo trong (hoặc nói chung thông tin chỉ báo chế độ dự báo trong được chọn cho khối) ra khối mã hóa entropy 270 ở dạng của các phần tử cú pháp 266 để đưa vào dữ liệu ảnh được mã hóa 21, sao cho, chẳng hạn, bộ giải mã video 30 có thể nhận và sử dụng các tham số dự báo để giải mã.

Dự báo ngoài

Tập hợp chế độ dự báo ngoài (hoặc khả thi) phụ thuộc vào các ảnh tham chiếu khả dụng (chẳng hạn, các ảnh được giải mã ít nhất một phần trước đó, chẳng hạn,

được lưu trữ trong DBP 230) và các tham số dự báo ngoài khác, chẳng hạn, liệu toàn bộ ảnh tham chiếu hoặc chỉ một phần, chẳng hạn, diện tích cửa sổ tìm kiếm quanh diện tích của khối hiện tại, của ảnh tham chiếu được sử dụng để tìm kiếm khối tham chiếu so khớp tốt nhất, và/hoặc chẳng hạn, liệu nội suy pixel có được áp dụng, chẳng hạn, nội suy nửa pel và/hoặc $\frac{1}{4}$ pel, hay không.

Bên cạnh các chế độ dự báo trên, chế độ bỏ qua và/hoặc chế độ trực tiếp có thể được áp dụng.

Khối dự báo ngoài 244 có thể bao gồm khối ước tính chuyển động (motion estimation, ME) và khối bù chuyển động (motion compensation, MC) (cả hai không được thể hiện trên Fig.2). Khối ước tính chuyển động có thể được tạo cấu hình để nhận hoặc thu được khối ảnh 203 (khối ảnh hiện tại 203 của ảnh hiện tại 17) và ảnh được giải mã 231, hoặc ít nhất một hoặc các khối được tái tạo trước đó, chẳng hạn, các khối tái tạo của một hoặc các ảnh được giải mã trước 231 khác nhau, để ước tính chuyển động. Chẳng hạn, chuỗi video có thể bao gồm ảnh hiện tại và các ảnh được giải mã trước đó 231, hoặc nói cách khác, ảnh hiện tại và các ảnh được giải mã trước đó 231 có thể là một phần hoặc tạo chuỗi ảnh tạo chuỗi video.

Bộ mã hóa 20 có thể, chẳng hạn,, được tạo cấu hình để chọn khối tham chiếu từ các khối tham chiếu của các ảnh giống nhau hoặc khác nhau của các ảnh khác và tạo ảnh tham chiếu (hoặc chỉ số ảnh tham chiếu) và/hoặc độ lệch (độ lệch không gian) giữa vị trí (các tọa độ x, y) của khối tham chiếu và vị trí của khối hiện tại như là các tham số dự báo ngoài đến khối ước tính chuyển động. Độ lệch này cũng được gọi là vector chuyển động (motion vector, MV).

Khối bù chuyển động được tạo cấu hình để thu được, chẳng hạn, nhận, tham số dự báo ngoài và để thực hiện dự báo ngoài dựa trên hoặc sử dụng tham số dự báo ngoài để thu được khối dự báo ngoài 265. Bù chuyển động, được thực hiện

bởi khối bù chuyển động, có thể bao gồm nạp hoặc tạo khối dự báo dựa trên vector chuyển động/vector khối được xác định bởi ước tính chuyển động, có thể thực hiện các phép nội suy đến độ chính xác của pixel con. Phép lọc nội suy có thể tạo thêm các mẫu pixel từ các mẫu pixel đã biết, do vậy có thể tăng số lượng khối dự báo ứng viên có thể được sử dụng để mã hóa khối ảnh. Khi nhận MV cho PU của khối ảnh hiện tại, khối bù chuyển động có thể định vị khối dự báo mà MV trỏ đến ở một trong các danh sách ảnh tham chiếu.

Khối bù chuyển động có thể cũng tạo các phần tử cú pháp được liên kết với các khối và lát video để sử dụng bởi bộ giải mã video 30 khi giải mã các khối ảnh của lát video.

Tạo mã Entropy

Khối mã hóa entropy 270 được tạo cấu hình để áp dụng, chẳng hạn, thuật toán hoặc phương tiện mã hóa entropy (chẳng hạn, phương tiện tạo mã có độ dài biến thiên (variable length coding, VLC), phương tiện VLC có ngữ cảnh thích ứng (context adaptive VLC scheme, CAVLC), phương tiện mã hóa số học, phép nhị phân hóa, tạo mã số học nhị phân có ngữ cảnh thích ứng (context adaptive binary arithmetic coding, CABAC), CABAC dựa trên cú pháp (syntax-based context-adaptive binary arithmetic coding, SBAC), mã hóa entropy phân vùng khoảng xác suất (probability interval partitioning entropy, PIPE) hoặc phương pháp hoặc kỹ thuật mã hóa entropy khác) hoặc đi vòng (không nén) trên các hệ số được lượng tử 209, các tham số dự báo ngoài, các tham số dự báo trong, các tham số lọc vòng và/hoặc các phần tử cú pháp khác để thu được dữ liệu ảnh được mã hóa 21 có thể được xuất ra qua đầu ra 272, chẳng hạn, ở dạng dòng bit được mã hóa 21, sao cho, chẳng hạn, bộ giải mã video 30 có thể nhận và sử dụng các tham số để giải mã. Dòng bit được mã hóa 21 có thể được truyền đến bộ giải mã

video 30, hoặc được lưu trữ trong bộ nhớ để truyền hoặc truy xuất sau này bởi bộ giải mã video 30.

Các biến thể cấu trúc khác của bộ mã hóa video 20 có thể được sử dụng để mã hóa dòng video. Chẳng hạn, bộ mã hóa không biến đổi 20 có thể lượng tử phần dư trực tiếp không qua khối xử lý biến đổi 206 cho các khối hoặc các khung cụ thể. Theo triển khai khác, bộ mã hóa 20 có thể khiến khối lượng tử 208 và khối lượng tử ngược 210 được kết hợp thành một khối.

Bộ giải mã và phương pháp giải mã

Fig.3 thể hiện ví dụ của bộ giải mã video 30 được tạo cấu hình để thực hiện các kỹ thuật của sáng chế. Bộ giải mã video 30 được tạo cấu hình để nhận dữ liệu ảnh được mã hóa 21 (chẳng hạn, dòng bit được mã hóa 21), chẳng hạn, được mã hóa bằng bộ mã hóa 20, để thu được ảnh được giải mã 331. Dữ liệu ảnh được mã hóa hoặc dòng bit bao gồm thông tin để giải mã dữ liệu ảnh được mã hóa, chẳng hạn, dữ liệu biểu diễn các khối ảnh của lát video được mã hóa và các phần tử cú pháp liên kết.

Trong ví dụ trên Fig.3, bộ giải mã 30 bao gồm khối giải mã entropy 304, khối lượng tử ngược 310, khối xử lý biến đổi ngược 312, khối tái tạo 314 (chẳng hạn, bộ cộng 314), khối lọc vòng 320, DBP 330, khối dự báo ngoài 344 và khối dự báo trong 354. Khối dự báo ngoài 344 có thể là hoặc bao gồm khối bù chuyển động. Bộ giải mã video 30 có thể, trong một số ví dụ, thực hiện bước giải mã nói chung nghịch đảo với bước mã hóa được mô tả với bộ mã hóa video 100 từ Fig.2.

Như được giải thích dựa vào bộ mã hóa 20, khối lượng tử ngược 210, khối xử lý biến đổi ngược 212, khối tái tạo 214 bộ lọc vòng 220, DPB 230, khối dự báo ngoài 344 và khối dự báo trong 354 cũng được gọi là tạo “bộ giải mã cài

sẵn” của bộ mã hóa video 20. Tương ứng, khối lượng tử ngược 310 có thể có chức năng giống hệt với khối lượng tử ngược 110, khối xử lý biến đổi ngược 312 có thể có chức năng giống hệt với khối xử lý biến đổi ngược 212, khối tái tạo 314 có thể có chức năng giống hệt với khối tái tạo 214, bộ lọc vòng 320 có thể có chức năng giống hệt với bộ lọc vòng 220, và DPB 330 có thể có chức năng giống hệt với DPB 230. Do vậy, các phần giải thích được nêu cho các khối và các chức năng của bộ mã hóa video 20 áp dụng một cách tương ứng cho các khối và các chức năng tương ứng của bộ giải mã video 30.

Giải mã Entropy

Khối giải mã entropy 304 được tạo cấu hình để phân tách dòng bit 21 (hoặc nói chung dữ liệu ảnh được mã hóa 21) và thực hiện, chẳng hạn, giải mã entropy cho dữ liệu ảnh được mã hóa 21 để thu được, chẳng hạn, các hệ số được lượng tử 309 và/hoặc các tham số tạo mã được giải mã (không được thể hiện trên Fig.3), chẳng hạn, bất kỳ hoặc tất cả các tham số dự báo ngoài (chẳng hạn, chỉ số ảnh tham chiếu và vector chuyển động), tham số dự báo trong (chẳng hạn, chế độ hoặc chỉ số dự báo trong), các tham số biến đổi, các tham số lượng tử, các tham số lọc vòng, và/hoặc các phần tử cú pháp khác. Khối giải mã entropy 304 có thể được tạo cấu hình để áp dụng các thuật toán hoặc các phương tiện giải mã tương ứng với các phương tiện mã hóa như được mô tả theo khối mã hóa entropy 270 của bộ mã hóa 20. Khối giải mã entropy 304 có thể còn được tạo cấu hình để cấp các tham số dự báo ngoài, tham số dự báo trong và/hoặc các phần tử cú pháp khác cho khối chọn chế độ 360 và các tham số khác cho các khối khác của bộ giải mã 30. Bộ giải mã video 30 có thể nhận các phần tử cú pháp ở mức lát video và/hoặc mức khối video.

Lượng tử ngược

Khối lượng tử ngược 310 có thể được tạo cấu hình để nhận các QP (hoặc nói chung thông tin liên quan đến lượng tử ngược) và các hệ số được lượng tử từ dữ liệu ảnh được mã hóa 21 (chẳng hạn, bằng cách phân tách và/hoặc giải mã, chẳng hạn, bằng khối giải mã entropy 304) và dựa trên các QP áp dụng lượng tử ngược lên các hệ số được lượng tử được giải mã 309 để thu được các hệ số được lượng tử ngược 311, có thể cũng được gọi là các hệ số biến đổi 311. Quá trình lượng tử ngược có thể bao gồm sử dụng QP được xác định bởi bộ mã hóa video 20 cho mỗi khối video trong lát video để xác định mức độ lượng tử và, tương tự, mức độ lượng tử ngược cần được áp dụng.

Biến đổi ngược

Khối xử lý biến đổi ngược 312 có thể được tạo cấu hình để nhận các hệ số được lượng tử ngược 311, cũng được gọi là các hệ số biến đổi 311, và áp dụng biến đổi cho các hệ số được lượng tử ngược 311 để thu được các khối dư được tái tạo 213 trong miền mẫu. Các khối dư được tái tạo 213 có thể cũng được gọi là các khối biến đổi 313. Phép biến đổi này có thể là biến đổi ngược, chẳng hạn,, DCT ngược, DST ngược, phép biến đổi số nguyên ngược, hoặc quá trình biến đổi ngược tương tự về khái niệm. Khối xử lý biến đổi ngược 312 có thể còn được tạo cấu hình để nhận các tham số biến đổi hoặc thông tin tương ứng từ dữ liệu ảnh được mã hóa 21 (chẳng hạn, bằng cách phân tách và/hoặc giải mã, chẳng hạn, bằng khối giải mã entropy 304) để xác định phép biến đổi cần được áp dụng cho các hệ số được lượng tử ngược 311.

Tái tạo

Khối tái tạo 314 (chẳng hạn, bộ cộng 314) có thể được tạo cấu hình để bổ sung khối dư được tái tạo 313, vào khối dự báo 365 để thu được khối được tái tạo 315 trong miền mẫu, chẳng hạn, bằng cách bổ sung các giá trị mẫu của khối dư được tái tạo 313 và các giá trị mẫu của khối dự báo 365.

Lọc

Khối lọc vòng 320 (trong vòng mã hoặc sau vòng mã) được tạo cấu hình để lọc khối được tái tạo 315 để thu được khối được lọc 321, chẳng hạn, để làm mượt cách phép dịch chuyển pixel, hoặc ngược lại cải thiện chất lượng video. Khối lọc vòng 320 có thể bao gồm một hoặc nhiều bộ lọc vòng, chẳng hạn, bộ lọc tách khối, bộ lọc SAO hoặc một hoặc nhiều bộ lọc khác, chẳng hạn, bộ lọc hai bên, bộ lọc vòng thích ứng (adaptive loop filter, ALF), bộ lọc làm nét, bộ lọc làm mượt hoặc bộ lọc cộng tác, hoặc kết hợp bất kỳ của nó. Mặc dù khối lọc vòng 320 được thể hiện trên Fig.3 như là bộ lọc trong vòng, trong các cấu hình khác, khối lọc vòng 320 có thể được thực hiện dưới dạng hậu lọc vòng.

Bộ đệm ảnh được giải mã

Sau đó, các khối video được giải mã 321 của ảnh được lưu trữ trong bộ đệm ảnh được giải mã 330, lưu trữ các ảnh được giải mã 331 dưới dạng các ảnh tham chiếu để bù chuyển động tiếp theo cho các ảnh khác và/hoặc để hiển thị đầu ra tương ứng.

Bộ giải mã 30 được tạo cấu hình để xuất ra ảnh được giải mã 311, chẳng hạn, qua đầu ra 312, để trình chiếu hoặc hiển thị đến người dùng.

Dự báo

Khối dự báo ngoài 344 có thể có chức năng giống hệt khối dự báo ngoài 244 (cụ thể với khối bù chuyển động) và khối dự báo trong 354 có thể có chức năng

giống hết khối dự báo ngoài 254, và thực hiện các quyết định tách hoặc phân vùng và dự báo dựa trên các tham số phân vùng và/hoặc dự báo hoặc thông tin tương ứng được nhận từ dữ liệu ảnh được mã hóa 21 (chẳng hạn, bằng cách phân tách và/hoặc giải mã, chẳng hạn, bằng khối giải mã entropy 304). Khối chọn chế độ 360 có thể được tạo cấu hình để thực hiện dự báo (dự báo trong hoặc ngoài) theo khối dựa trên các ảnh được tái tạo, các khối hoặc các mẫu tương ứng (được lọc hoặc không được lọc) để thu được khối dự báo 365.

Khi lát video được mã hóa dưới dạng lát được mã hóa trong (I), khối dự báo trong 354 của khối chọn chế độ 360 được tạo cấu hình để tạo khối dự báo 365 cho khối ảnh của lát video hiện tại dựa trên chế độ dự báo trong được báo hiệu và dữ liệu từ các khối được giải mã trước đó của ảnh hiện tại. Khi ảnh video được mã hóa dưới dạng lát được mã hóa ngoài (chẳng hạn, B, hoặc P), khối dự báo ngoài 344 (chẳng hạn, khối bù chuyển động) của khối chọn chế độ 360 được tạo cấu hình để tạo các khối dự báo 365 cho khối video của lát video hiện tại dựa trên các vector chuyển động và các phần tử cú pháp khác được nhận từ khối giải mã entropy 304. Để dự báo ngoài, các khối dự báo có thể được tạo từ một trong các ảnh tham chiếu trong một trong các danh sách ảnh tham chiếu. Bộ giải mã video 30 có thể tạo các danh sách khung tham chiếu, List 0 và List 1, nhờ sử dụng các kỹ thuật tạo mặc định dựa trên các ảnh tham chiếu được lưu trữ trong DPB 330.

Khối chọn chế độ 360 được tạo cấu hình để xác định thông tin dự báo cho khối video của lát video hiện tại bằng cách phân tách các vector chuyển động và các phần tử cú pháp khác, và sử dụng thông tin dự báo để tạo các khối dự báo cho khối video hiện tại được giải mã. Chẳng hạn, khối chọn chế độ 360 sử dụng một số trong số các phần tử cú pháp được nhận để xác định chế độ dự báo (chẳng hạn, dự báo trong hoặc ngoài) được sử dụng để tạo mã các khối video của lát

video, loại lát dự báo ngoài (chẳng hạn, lát B, lát P, hoặc lát GPB), thông tin xây dựng cho một hoặc nhiều trong số các danh sách ảnh tham chiếu cho lát, các vector chuyển động cho mỗi khối video được mã hóa ngoài của lát, trạng thái dự báo ngoài cho mỗi khối video được tạo mã ngoài của lát, và thông tin khác để giải mã các khối video trong lát video hiện tại.

Các biến thể khác của bộ giải mã video 30 có thể được sử dụng để giải mã dữ liệu ảnh được mã hóa 21. Chẳng hạn, bộ giải mã 30 có thể tạo dòng video đầu ra không có khối lọc vòng 320. Chẳng hạn, bộ giải mã không biến đổi 30 có thể lượng tử ngược tín hiệu dư trực tiếp không có khối xử lý biến đổi ngược 312 cho các khối hoặc các khung cụ thể. Theo triển khai khác, bộ giải mã video 30 có thể kết hợp khối lượng tử ngược 310 và khối xử lý biến đổi ngược 312 thành một khối.

Cần hiểu rằng, trong bộ mã hóa 20 và bộ giải mã 30, kết quả xử lý của bước hiện tại có thể còn được xử lý và sau đó được xuất ra bước tiếp theo. Chẳng hạn, sau khi lọc nội suy, dẫn xuất vector chuyển động hoặc lọc vòng, hoạt động khác, chẳng hạn Clip hoặc shift, có thể được thực hiện trên kết quả xử lý lọc nội suy, dẫn xuất vector chuyển động hoặc lọc vòng.

Cần lưu ý rằng các hoạt động khác có thể được áp dụng cho các vector chuyển động được dẫn xuất của khối hiện tại (bao gồm mà không giới hạn các vector chuyển động điểm điều khiển (control point motion vector, CPMV) của chế độ afin, các vector chuyển động của khối phụ trong các chế độ afin, phẳng, ATMVP, các vector chuyển động thời gian, và v.v.). Chẳng hạn, giá trị của vector chuyển động bị giới hạn ở khoảng định trước theo bit đại diện của nó. Nếu bit đại diện của vector chuyển động là bitDepth, thì khoảng này từ $-2^{(\text{bitDepth}-1)}$ đến $2^{(\text{bitDepth}-1)}-1$, trong đó “^” nghĩa là ngoại suy. Chẳng hạn, nếu bitDepth được gán bằng 16, khoảng này là từ -32768 đến 32767; nếu bitDepth được gán bằng

18, khoảng này là từ -131072 đến 131071. Ở đây nêu hai phương pháp giới hạn vector chuyển động.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

trong đó định nghĩa về function Clip3 như sau:

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

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

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

Bộ xử lý 430 được thực hiện bằng phần cứng và phần mềm. Bộ xử lý 430 có thể được thực hiện dưới dạng một hoặc nhiều chip CPU, lõi (chẳng hạn, dưới dạng bộ xử lý nhiều lõi), FPGA, ASIC, và DSP. Bộ xử lý 430 đang nối thông với các cổng vào 410, các Rx 420, các Tx 440, các cổng ra 450, và bộ nhớ 460. Bộ xử lý 430 bao gồm môđun tạo mã 470. Môđun tạo mã 470 thực hiện các phương án thực hiện được bộc lộ được mô tả nêu trên. Chẳng hạn, môđun tạo mã 470 thực hiện, xử lý, chuẩn bị, hoặc tạo các hoạt động tạo mã khác nhau. Do vậy, việc đưa vào môđun tạo mã 470 cải thiện đáng kể chức năng của thiết bị tạo mã video 400 và biến đổi thiết bị tạo mã video 400 sang trạng thái khác. Theo cách khác, môđun tạo mã 470 được thực hiện dưới dạng các lệnh được lưu trữ trong bộ nhớ 460 và được thực thi bằng bộ xử lý 430.

Bộ nhớ 460 có thể bao gồm một hoặc nhiều đĩa, băng từ, và ổ trạng thái rắn (solid-state drive, SSD) và có thể được sử dụng dưới dạng thiết bị lưu trữ dữ liệu tràn, để lưu trữ các chương trình khi các chương trình này được chọn để thực thi, và để lưu trữ các lệnh và dữ liệu được đọc khi thực thi chương trình. Bộ nhớ 460 có thể, chẳng hạn, khả biến và/hoặc bất biến và có thể là bộ nhớ chỉ đọc (read-only memory, ROM), bộ nhớ truy nhập ngẫu nhiên (random access memory, RAM), bộ nhớ có nội dung đánh địa chỉ được tam phân (ternary content-addressable memory, TCAM), và/hoặc bộ nhớ truy nhập ngẫu nhiên tĩnh (static random-access memory, SRAM).

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

Bộ xử lý 502 trong thiết bị 500 có thể là CPU. Theo cách khác, bộ xử lý 502 có thể là loại khác bất kỳ của thiết bị, hoặc nhiều thiết bị, có thể thao tác hoặc xử lý thông tin hiện có hoặc được phát triển sau này. Mặc dù các cách triển khai được bộc lộ có thể được thực thi với một bộ xử lý như được thể hiện, chẳng hạn,, bộ xử lý 502, các ưu điểm về tốc độ và hiệu suất có thể đạt được nhờ sử dụng nhiều hơn một bộ xử lý.

Bộ nhớ 504 trong thiết bị 500 có thể là ROM hoặc RAM theo cách thức triển khai. Loại thích hợp khác bất kỳ của thiết bị lưu trữ có thể được sử dụng dưới dạng bộ nhớ 504. Bộ nhớ 504 có thể bao gồm mã và dữ liệu 506 được truy nhập bởi bộ xử lý 502 nhờ sử dụng buýt 512. Bộ nhớ 504 có thể còn bao gồm hệ điều hành 508 và các chương trình ứng dụng 510, các chương trình ứng dụng 510 bao gồm ít nhất một chương trình cho phép bộ xử lý 502 để thực hiện các phương pháp được mô tả ở đây. Chẳng hạn, các chương trình ứng dụng 510 có thể bao

gồm các ứng dụng từ 1 đến N, còn bao gồm ứng dụng tạo mã video thực hiện các phương pháp được mô tả ở đây.

Thiết bị 500 cũng có thể bao gồm một hoặc nhiều bộ phận xuất, chẳng hạn bộ hiển thị 518. Bộ phận hiển thị 518 có thể là, trong một ví dụ, bộ phận cảm ứng tiếp xúc kết hợp phần hiển thị với phần tử cảm ứng tiếp xúc có thể hoạt động để cảm nhận các đầu vào tiếp xúc. Bộ phận hiển thị 518 có thể được ghép nối với bộ xử lý 502 qua bus 512.

Mặc dù được mô tả ở đây dưới dạng một bus, bus 512 của thiết bị 500 có thể gồm nhiều bus. Ngoài ra, bộ phận lưu trữ thứ cấp 514 có thể được ghép nối trực tiếp với các thành phần khác của thiết bị 500 hoặc có thể được truy nhập qua mạng và có thể bao gồm một khối tích hợp, chẳng hạn, thẻ nhớ hoặc nhiều khối, chẳng hạn, nhiều thẻ nhớ. Do vậy, thiết bị 500 có thể được thực hiện trong các loại cấu hình.

Fig.18 là sơ đồ khối thể hiện hệ thống cấp nội dung 3100 để thực hiện dịch vụ phân tán nội dung. Hệ thống cấp nội dung 3100 này bao gồm thiết bị chụp 3102, thiết bị đầu cuối 3106, và một cách tùy chọn bao gồm bộ phận hiển thị 3126. Thiết bị chụp 3102 truyền thông với thiết bị đầu cuối 3106 trên đường truyền thông 3104. Đường truyền thông có thể bao gồm kênh truyền thông 13 được mô tả nêu trên. Đường truyền thông 3104 bao gồm nhưng không bị giới hạn ở WiFi, Ethernet, cáp, không dây (3G/4G/5G), USB, hoặc loại bất kỳ của kết hợp của nó, hoặc tương tự.

Thiết bị chụp 3102 tạo dữ liệu, và có thể mã hóa dữ liệu bằng phương pháp mã hóa như được thể hiện trên các phương án thực hiện nêu trên. Theo cách khác, thiết bị chụp 3102 có thể phân tán dữ liệu đến máy chủ streaming (không được thể hiện trong các hình vẽ), và máy chủ mã hóa dữ liệu và truyền dữ liệu được mã hóa đến thiết bị đầu cuối 3106. Thiết bị chụp 3102 bao gồm mà không

bị giới hạn ở camera, điện thoại thông minh hoặc Pad, máy tính hoặc laptop, hệ thống hội nghị video, PDA, thiết bị lắp trong xe, hoặc kết hợp của chúng, hoặc tương tự. Chẳng hạn, thiết bị chụp 3102 có thể bao gồm thiết bị nguồn 12 như được mô tả trên đây. Khi dữ liệu bao gồm video, bộ mã hóa video 20 được bao gồm trong thiết bị chụp 3102 có thể thực sự thực hiện xử lý mã hóa video. Khi dữ liệu bao gồm audio (chẳng hạn, thoại), bộ mã hóa audio được bao gồm trong thiết bị chụp 3102 có thể thực sự thực hiện quá trình mã hóa audio. Đối với một số kịch bản thực tế, thiết bị chụp 3102 phân tán dữ liệu audio và video được mã hóa bằng cách ghép kênh chúng với nhau. Đối với các kịch bản thực tế khác, chẳng hạn trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được ghép kênh. Thiết bị chụp 3102 phân tán riêng rẽ dữ liệu audio được mã hóa và dữ liệu video được mã hóa đến thiết bị đầu cuối 3106.

Trong hệ thống cấp nội dung 3100, thiết bị đầu cuối 310 nhận và tái tạo dữ liệu được mã hóa. Thiết bị đầu cuối 3106 có thể là thiết bị có khả năng nhận và khôi phục dữ liệu, chẳng hạn điện thoại thông minh hoặc Pad 3108, máy tính hoặc máy tính xách tay 3110, đầu ghi video mạng (network video recorder, NVR)/đầu ghi video số (digital video recorder, DVR) 3112, TV 3114, đầu thu tín hiệu truyền hình cáp (set top box, STB) 3116, hệ thống hội nghị video 3118, hệ thống giám sát video 3120, PDA 3122, thiết bị lắp trong xe 3124, hoặc kết hợp của một trong các thiết bị đó, hoặc tương tự có thể giải mã dữ liệu được mã hóa nêu trên. Chẳng hạn, thiết bị đầu cuối 3106 có thể bao gồm thiết bị đích 14 như được mô tả trên đây. Khi dữ liệu được mã hóa bao gồm video, bộ giải mã video 30 được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện giải mã video. Khi dữ liệu được mã hóa bao gồm audio, bộ giải mã audio được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện quá trình giải mã audio.

Đối với thiết bị đầu cuối có màn hiển thị, chẳng hạn, điện thoại thông minh hoặc Pad 3108, máy tính hoặc máy tính xách tay 3110, NVR/DVR 3112, TV 3114, PDA 3122, hoặc thiết bị lắp trong xe 3124, thiết bị đầu cuối có thể cấp dữ liệu được giải mã cho bộ phận hiển thị. Đối với thiết bị đầu cuối không có màn hiển thị, chẳng hạn, STB 3116, hệ thống hội nghị video 3118, hoặc hệ thống giám sát video 3120, bộ phận hiển thị ngoài 3126 được tiếp xúc ở đó để nhận và thể hiện dữ liệu được giải mã.

Khi mỗi thiết bị trong hệ thống máy thực hiện mã hóa hoặc giải mã, thiết bị mã hóa ảnh hoặc thiết bị giải mã ảnh, như được thể hiện trên các phương án thực hiện nêu trên, có thể được sử dụng.

Fig.19 là sơ đồ thể hiện cấu trúc của ví dụ của thiết bị đầu cuối 3106. Sau khi thiết bị đầu cuối 3106 nhận stream từ thiết bị chụp 3102, khối xử lý giao thức 3202 phân tích giao thức truyền của dòng. Giao thức bao gồm mà không bị giới hạn ở giao thức phát trực tuyến thời gian thực (Real Time Streaming Protocol, RTSP), giao thức truyền siêu văn bản (Hyper Text Transfer Protocol, HTTP), giao thức truyền trực tuyến trực tiếp HTTP (HTTP Live streaming protocol, HLS), MPEG-DASH, giao thức vận tải thời gian thực (Real-time Transport protocol, RTP), giao thức nhắn tin thời gian thực (Real Time Messaging Protocol, RTMP), hoặc loại bất kỳ của kết hợp của nó, hoặc tương tự.

Sau khi khối xử lý giao thức 3202 xử lý dòng này, tệp stream được tạo. Tệp tin này được xuất ra khỏi phân kênh 3204. Khối phân kênh 3204 có thể tách riêng dữ liệu được ghép kênh thành dữ liệu audio được mã hóa và dữ liệu video được mã hóa. Như được mô tả trên đây, đối với một số kịch bản thực tế, chẳng hạn, trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được phân kênh. Trong tình huống này, dữ liệu được mã hóa được

truyền đến bộ giải mã video 3206 và bộ giải mã audio 3208 không qua khối phân kênh 3204.

Qua quá trình phân kênh, dòng video cơ bản (elementary stream, ES), audio ES, và một cách tùy chọn, phụ đề được tạo. Bộ giải mã video 3206, bao gồm bộ giải mã video 30 như được giải thích theo các phương án thực hiện nêu trên, giải mã ES video bằng phương pháp giải mã như được thể hiện trên các phương án thực hiện nêu trên để tạo khung video, và nạp dữ liệu này cho khối đồng bộ 3212. Bộ giải mã audio 3208, giải mã ES audio để tạo khung audio, và cấp dữ liệu này cho khối đồng bộ 3212. Theo cách khác, khung video có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.Y) trước khi cấp nó cho khối đồng bộ 3212. Tương tự, khung audio có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.Y) trước khi cấp nó cho khối đồng bộ 3212.

Khối đồng bộ 3212 đồng bộ khung video và khung audio, và cấp video/audio cho bộ phận hiển thị video/audio 3214. Chẳng hạn, khối đồng bộ 3212 đồng bộ hiển thị thông tin video và audio. Thông tin có thể mã hóa trong cú pháp nhờ sử dụng các nhãn thời gian liên quan đến trình chiếu dữ liệu audio và hình ảnh được mã hóa và các nhãn thời gian liên quan đến việc truyền chính dữ liệu stream.

Nếu phụ đề được bao gồm trong stream, bộ giải mã phụ đề 3210 giải mã phụ đề, và đồng bộ nhớ với khung video và khung audio, và cấp video/audio/phụ đề cho bộ phận hiển thị video/audio/phụ đề 3216.

Trong ví dụ về việc tạo danh sách ứng viên hợp nhất theo ITU-T H.265, danh sách ứng viên hợp nhất được tạo dựa trên các ứng viên sau:

1. Có tới bốn ứng viên không gian được dẫn xuất từ năm khối lân cận không gian,
2. Một ứng viên thời gian được dẫn xuất từ hai khối cùng vị trí thời gian,
3. Các ứng viên bổ sung bao gồm các ứng viên dự báo kép kết hợp và,

4. Các ứng viên MV số 0.

Ứng viên không gian

Thông tin chuyển động của các khối lân cận không gian trước hết được bổ sung vào danh sách ứng viên hợp nhất (trong ví dụ, danh sách ứng viên hợp nhất có thể là danh sách trống trước khi MV thứ nhất được bổ sung vào danh sách ứng viên hợp nhất) dưới dạng các ứng viên thông tin chuyển động. Ở đây, các khối lân cận được xem xét chèn vào danh sách hợp nhất được minh họa trên Fig.6b. Để hợp nhất khối dự báo ngoài, có tới bốn ứng viên được chèn vào danh sách hợp nhất bằng cách lần lượt kiểm tra A1, B1, B0, A0 và B2, theo thứ tự đó.

Thông tin chuyển động có thể chứa tất cả dữ liệu chuyển động bao gồm thông tin liệu một hoặc hai danh sách ảnh tham chiếu được sử dụng cũng như chỉ số tham chiếu và vectơ chuyển động cho mỗi danh sách ảnh tham chiếu.

Trong ví dụ, sau khi kiểm tra liệu khối lân cận có khả dụng hay không và chứa thông tin chuyển động, một số phép kiểm tra dư thừa bổ sung được thực hiện trước khi lấy tất cả dữ liệu chuyển động của khối lân cận làm ứng viên thông tin chuyển động. Các phép kiểm tra dư này có thể được phân chia thành hai phân loại cho hai mục đích khác nhau:

Phân loại 1, tránh có các ứng viên có dữ liệu chuyển động dư trong danh sách,

Phân loại 2, ngăn ngừa hợp nhất hai phân vùng có thể được biểu diễn bằng các phương tiện khác sẽ tạo cú pháp dư.

Các ứng viên thời gian

Fig.6a minh họa các tọa độ của các khối trong đó các ứng viên thông tin chuyển động thời gian được truy xuất từ đó. Khối cùng vị trí là khối có các tọa độ -x, -y giống nhau của khối hiện tại, nhưng trên ảnh khác nhau (một trong các ảnh tham chiếu). Các ứng viên thông tin chuyển động thời gian được bổ sung

vào danh sách hợp nhất nếu danh sách này không đầy (trong ví dụ, danh sách hợp nhất không đầy khi số lượng ứng viên trong danh sách hợp nhất nhỏ hơn ngưỡng, chẳng hạn, ngưỡng này có thể bằng 4, 5, 6, và v.v.).

Các ứng viên được tạo

Sau khi chèn các ứng viên thông tin chuyển động không gian và thời gian, nếu danh sách hợp nhất vẫn không đầy, các ứng viên được tạo được bổ sung để điền đầy danh sách. Kích thước của danh sách được chỉ báo trong tập hợp tham số chuỗi và được cố định trong suốt toàn bộ chuỗi video được mã hóa.

Quá trình tạo danh sách hợp nhất trong ITU-T H.265 và trong VVC xuất ra danh sách ứng viên thông tin chuyển động. Quá trình tạo danh sách hợp nhất của VVC được mô tả trong phần “8.3.2.2 Derivation process for luma motion vectors for merge mode” của tài liệu JVET-L1001_v2 Versatile Video Coding (Draft 3), được công khai rộng rãi trong <http://phenix.it-sudparis.eu/jvet/>. Cụm từ thông tin chuyển động đề cập đến dữ liệu chuyển động cần để thực hiện quá trình dự báo bù chuyển động. Thông tin chuyển động thường đề cập đến thông tin sau:

- Liệu khối có áp dụng dự báo đơn hoặc dự báo kép
- ID của ảnh tham chiếu được sử dụng trong dự báo. (2 ID nếu khối áp dụng dự báo kép).
- Vector chuyển động (2 MV nếu khối được dự báo kép)
- Thông tin bổ sung

Trong chuẩn VVC và H.265, danh sách ứng viên vốn là đầu ra của quá trình tạo danh sách hợp nhất bao gồm N thông tin chuyển động ứng viên. Số N được bao gồm trong dòng bit và có thể là số nguyên dương, chẳng hạn, 5, 6 v.v.. Các ứng viên được bao gồm trong danh sách hợp nhất được tạo có thể bao gồm thông tin dự báo đơn hoặc thông tin dự báo kép. Điều này nghĩa là ứng viên được chọn từ danh sách hợp nhất có thể chỉ báo hoạt động dự báo kép.

Dự báo kép

Chế độ đặc biệt dự báo ngoài được gọi là “dự báo kép”, trong đó 2 MV được sử dụng để dự báo khối. Các vector chuyển động có thể trỏ đến các ảnh tham chiếu tương tự hoặc khác nhau, trong đó ảnh tham chiếu có thể được chỉ báo bởi ID danh sách ảnh tham chiếu và chỉ số ảnh tham chiếu. Chẳng hạn, MV thứ nhất có thể trỏ đến ảnh thứ nhất trong danh sách ảnh tham chiếu L0 và MV thứ hai có thể trỏ đến ảnh thứ nhất trong danh sách ảnh tham chiếu L1. Hai danh sách ảnh tham chiếu (chẳng hạn, L0 và L1) có thể được duy trì và ảnh được trỏ bởi MV thứ nhất được chọn từ danh sách L0 và ảnh được trỏ bởi MV thứ hai được chọn từ danh sách L1.

Trong ví dụ, nếu thông tin chuyển động chỉ báo dự báo kép, thì thông tin chuyển động bao gồm hai phần:

- Phần L0: MV và chỉ số ảnh tham chiếu trỏ đến mục trong danh sách ảnh tham chiếu L0.
- Phần L1: MV và chỉ số ảnh tham chiếu trỏ đến mục trong danh sách ảnh tham chiếu L1.

Số đếm thứ tự ảnh (Picture Order Count, POC): Biến được liên kết với mỗi ảnh, nhận diện duy nhất ảnh được liên kết trong số tất cả các ảnh trong chuỗi video được tạo mã (Coded Video Sequence, CVS), và, khi ảnh được liên kết cần được xuất ra từ DPB, chỉ báo vị trí của ảnh được liên kết theo thứ tự xuất ra so với các vị trí của thứ tự xuất ra của các ảnh khác trong cùng CVS cần được xuất ra từ DPB.

Mỗi danh sách trong các danh sách ảnh tham chiếu L0 và L1 có thể bao gồm một hoặc nhiều ảnh tham chiếu mà mỗi ảnh được nhận diện bằng POC. Liên kết với mỗi chỉ số tham chiếu và giá trị POC có thể được báo hiệu trong dòng bit.

Như là ví dụ, các danh sách ảnh tham chiếu L0 và L1 có thể bao gồm các ảnh tham chiếu sau:

Danh sách ảnh tham chiếu	Chỉ số tham chiếu	POC
L0	0	12
L0	1	13
L1	0	13
L1	1	14

Trong ví dụ nêu trên, mục thứ nhất (được chỉ báo bởi chỉ số tham chiếu 0) trong danh sách ảnh tham chiếu L1 là ảnh tham chiếu có giá trị POC 13. Mục thứ hai (được chỉ báo bởi chỉ số tham chiếu 1) trong danh sách ảnh tham chiếu L1 là ảnh tham chiếu có giá trị POC 14.

Chế độ dự báo tam giác

Khái niệm chế độ dự báo tam giác sẽ bộc lộ phân vùng tam giác để dự báo bù chuyển động. Như là ví dụ được thể hiện Fig.7, hai khối dự báo tam giác được sử dụng cho CU, theo đường chéo hoặc đường chéo ngược. Mỗi khối dự báo tam giác trong CU được dự báo ngoài, nhờ sử dụng MV dự báo đơn và chỉ số khung tham chiếu được dẫn xuất từ danh sách ứng viên dự báo đơn. Quá trình đánh trọng số thích ứng được thực hiện với cạnh đường chéo, sau khi các mẫu được liên kết với mỗi khối dự báo tam giác đã được dự báo bằng cách bù chuyển động hoặc dự báo trong ảnh. Sau đó, quá trình biến đổi và lượng tử được áp dụng cho CU. Cần lưu ý rằng chế độ này được áp dụng cho chế độ bỏ qua và chế độ hợp nhất.

Trong chế độ dự báo tam giác, khối được tách thành hai phần tam giác (như trên Fig.7), mỗi phần có thể được dự báo nhờ sử dụng một MV. MV được sử dụng để dự báo một phần tam giác (được ký hiệu bằng PU1) có thể khác với MV

được sử dụng để dự báo phần tam giác khác (được ký hiệu bằng PU2). Trong ví dụ, cần lưu ý rằng mỗi phần có thể được dự báo nhờ sử dụng một MV (dự báo đơn), để giảm độ phức tạp thực hiện chế độ dự báo tam giác. Nói cách khác, PU1 và PU2 có thể không được dự báo nhờ sử dụng dự báo kép, bao gồm hai MV.

Chế độ dự báo khối phụ

Chế độ dự báo tam giác là trường hợp đặc biệt dự báo khối phụ. Trong dự báo khối phụ, khối được phân chia thành hai khối. Trong các ví dụ nêu trên, hai hướng phân chia khối được minh họa (các phân vùng 45° và 135°), các góc phân vùng khác và các tỷ lệ phân vùng cũng khả thi (các ví dụ trên Fig.8).

Trong một số ví dụ, khối được tách thành 2 phần và mỗi phần được áp dụng bằng dự báo đơn. Dự báo khối phụ biểu diễn phiên bản tổng quát của dự báo tam giác.

Chế độ dự báo tam giác là trường hợp đặc biệt của chế độ dự báo khối phụ, trong đó khối được phân chia thành hai khối. Trong các ví dụ nêu trên, hai hướng phân chia khối được minh họa (các phân vùng 45° và 135°). Các góc phân vùng và các tỷ lệ phân vùng cũng khả thi (Chẳng hạn, các ví dụ trên Fig.8).

Trong một số ví dụ, khối được tách thành 2 khối phụ và mỗi phần (khối phụ) được dự báo với dự báo đơn.

Trong ví dụ, theo chế độ phân vùng khối phụ, các bước sau được áp dụng để thu được các mẫu dự báo:

- Bước 1: khối mã được xem xét phân chia thành 2 khối phụ, theo mô hình hình học. Mô hình này có thể dẫn đến tách khối bằng đường tách (chẳng hạn, đường thẳng) như được lấy ví dụ trên các hình vẽ từ Fig.9 đến Fig.12. Trong bước này, theo mô hình hình học, các mẫu trong khối mã được xem xét được đặt thành hai khối phụ. Khối phụ A hoặc khối phụ B bao gồm phần mẫu (nhưng

không phải tất cả) trong khối mã hiện tại. Các khái niệm Khối phụ A hoặc Khối phụ B có thể được chỉ báo bởi tham số liên quan đến đường tách.

- Bước 2: thu được chế độ dự báo thứ nhất đối với khối phụ thứ nhất và chế độ dự báo thứ hai đối với khối phụ thứ hai. Trong ví dụ, chế độ dự báo thứ nhất không giống như chế độ dự báo thứ hai. Trong ví dụ, chế độ dự báo (chế độ dự báo thứ nhất hoặc chế độ dự báo thứ hai) có thể là chế độ dự báo ngoài, thông tin đối với chế độ dự báo ngoài có thể bao gồm chỉ số ảnh tham chiếu và MV. Trong ví dụ khác, chế độ dự báo có thể là chế độ dự báo trong, thông tin cho chế độ dự báo trong có thể bao gồm chỉ số chế độ dự báo trong.
- Bước 3: Tạo các giá trị dự báo thứ nhất và các giá trị dự báo thứ hai, sử dụng lần lượt chế độ dự báo thứ nhất và chế độ dự báo thứ hai.
- Bước 4: thu được các giá trị kết hợp của các mẫu dự báo theo kết hợp của các giá trị dự báo thứ nhất và các giá trị dự báo thứ hai.

Trong ví dụ, trong bước 1, khối mã được xem xét phân chia thành 2 khối phụ theo các cách khác nhau. Fig. 9 thể hiện ví dụ về phân vùng của khối mã, đường tách 1250 phân chia khối thành 2 khối phụ. Để mô tả đường tách 1250, hai tham số được báo hiệu, một tham số là góc alpha 1210 và tham số còn lại là khoảng cách dist 1230.

Theo một số phương án thực hiện, góc, như được chỉ báo trên Fig.9, được đo giữa trục x và đường tách, trong đó khoảng cách được đo bằng chiều dài của vectơ vuông góc với đường tách và đi qua tâm của khối hiện tại.

Trong ví dụ khác, Fig.10 thể hiện cách khác biểu diễn đường tách, trong đó ví dụ về góc và khoảng cách khác với các ví dụ được thể hiện trên Fig. 9.

Trong ví dụ bất kỳ, ở bước 4, việc phân chia được bộc lộ trong Bước 1 được sử dụng để kết hợp các giá trị dự báo thứ nhất và các giá trị dự báo thứ hai, để thu được các giá trị dự báo cuối cùng. Trong ví dụ, hoạt động kết hợp được áp

dụng trong bước 4 để bỏ qua bất kỳ giả tượng nào (mép hoặc gồ ghề dọc theo đường tách). Hoạt động kết hợp có thể được mô tả dưới dạng hoạt động lọc dọc theo đường tách.

Ở phía bộ mã hóa, đường tách (các tham số định nghĩa đường, chẳng hạn, góc và khoảng cách) được xác định dựa trên hàm chi phí dựa trên méo dạng tốc độ. Các tham số đường được xác định được mã hóa thành dòng bit. Ở phía bộ giải mã, các tham số đường được giải mã (thu được) theo dòng bit.

Trong ví dụ, trong trường hợp 3 kênh video, bao gồm thành phần độ sáng và hai thành phần sắc độ, dự báo thứ nhất và dự báo thứ hai được tạo cho mỗi kênh.

Do có nhiều khả năng phân chia khối mã thành 2 khối phụ, báo hiệu (mã hóa) phép chia đòi hỏi quá nhiều bit. Do các giá trị góc và khoảng cách có thể có nhiều giá trị khác nhau, đòi hỏi quá nhiều thông tin phụ cần được báo hiệu trong dòng bit, phương tiện lượng tử được áp dụng cho thông tin phụ góc và khoảng cách để cải thiện hiệu suất mã.

Trong ví dụ, tham số góc được lượng tử α_{Idx} và tham số khoảng cách được lượng tử $distance_{Idx}$ được báo hiệu cho các tham số phân vùng.

Trong ví dụ, trong phương tiện lượng tử, giá trị góc và giá trị khoảng cách có thể được lượng tử bởi bộ lượng tử tuyến tính, đồng nhất theo

$$\alpha_{Idx} = \text{Round}(\alpha / \Delta\alpha),$$

$$distance_{Idx} = \text{Round}(dist / \Delta dist),$$

trong đó các biến $\Delta\alpha$ và $\Delta dist$ là các kích thước bậc lượng tử cho tham số góc và tham số khoảng cách của đường tách. Tương ứng, góc α và khoảng cách $dist$ có thể được tái tạo nhờ sử dụng các giá trị của α_{Idx} và $distance_{Idx}$ thu được từ dòng bit như là

$$\alpha = \alpha_{Idx} * \Delta\alpha,$$

$$dist = distance_{Idx} * \Delta dist.$$

Trong ví dụ, nếu kích thước bậc lượng tử góc bằng $\Delta\alpha = 11.25^\circ$, giá trị của α_{Idx} có thể nằm trong khoảng từ 0 đến 31 (bao gồm 0 và 31) để bao phủ các góc được phép trong mặt phẳng Oclit. Trong ví dụ khác, giá trị của kích thước bậc lượng tử góc bằng $\Delta\alpha = 22.5^\circ$, nhờ đó giá trị của α_{Idx} có thể nằm trong khoảng từ 0 đến 15 (bao gồm 0 và 15).

Trong một số ví dụ, phương pháp được đề xuất để lượng tử thông tin khoảng cách phụ thuộc vào góc, trong trường hợp bộ lượng tử tuyến tính và đồng nhất được sử dụng để lượng tử góc, nghĩa là giá trị của $\Delta\alpha$ là không đổi đối với khối mã cụ thể (giải mã hoặc mã hóa).

Mục đích của phương tiện lượng tử được phép là nhằm phân phối đồng đều các đường tách được phép đối với khối mã cụ thể, có thể có các kích thước và các khuôn dạng khác nhau. Không gian được phân phối đều của các tham số góc và khoảng cách được lượng tử cải thiện hiệu suất mã của các thành phần phi chữ nhật và cho phép biểu diễn nhỏ gọn các tham số đường tách.

Trong một số ví dụ, phương pháp phân vùng khối mã chữ nhật bằng đường thẳng được bộc lộ, trong đó đường thẳng được tham số hóa bằng cặp tham số, biểu diễn góc được lượng tử và giá trị khoảng cách được lượng tử và trong đó giá trị khoảng cách được lượng tử được dẫn xuất theo quá trình lượng tử, tùy thuộc vào giá trị của góc và khuôn dạng của khối mã.

Trong ví dụ, khoảng cách có thể được lượng tử theo cách này, sao cho khoảng giá trị cụ thể của $distance_{Idx}$ sẽ được thỏa mãn, chẳng hạn, khoảng giá trị từ 0 đến 3 (bao gồm 0 và 3). Trong ví dụ khác, khoảng cách có thể được lượng tử cho khối cụ thể theo sách sao cho các đường tách đối với các cặp giá trị cụ thể $angle_{Idx}$ và $distance_{Idx}$ được phân tán đồng đều và đường tách sẽ không nằm ngoài khu vực của khối mã cụ thể.

Ở bước thứ nhất, khoảng cách lớn nhất ρ_{max} có thể được dẫn xuất tùy thuộc

vào góc, sao cho tất cả các đường tách có giá trị khoảng cách $0 < \text{dist} < \rho_{\max}$ bị giới hạn ở khối mã (chẳng hạn, chúng đang giao với đường biên của khối mã). Điều này được lấy ví dụ trên Fig.15 đối với khối mã có kích thước mẫu độ sáng 16x4.

Trong ví dụ, khoảng cách lớn nhất $\rho_{\max}$ có thể được suy ra dưới dạng hàm, tùy thuộc vào góc αR và kích thước của khối mã theo:

$$\rho_{\max}(\alpha R, w, h) = \cos(\alpha R) \left(\frac{h}{2 \tan(\frac{\pi}{2} - \alpha R)} + \frac{w}{2} \right), \quad 0 \leq \alpha R < \frac{\pi}{2},$$

trong đó αR là góc được ký hiệu trước đó là α theo các đơn vị radian, w là chiều rộng và h là chiều cao của khối theo các mẫu độ sáng. Trong ví dụ, giá trị của kích thước bậc lượng tử khoảng cách phụ thuộc góc $\Delta \text{dist}(\alpha)$ có thể được dẫn xuất dưới dạng

$$\Delta \text{dist}(\alpha R, w, h) = \rho_{\max}(\alpha R, w, h) / N,$$

trong đó $N-1$ là giá trị lớn nhất của distanceIdx . Điều này được lấy ví dụ trên Fig.16 với các giá trị $\Delta \text{angle} = 22,5^\circ$ và $N=4$.

Trong ví dụ khác, giá trị của kích thước bậc lượng tử khoảng cách phụ thuộc góc có thể được dẫn xuất là

$$\Delta \text{dist}(\alpha R, w, h) = (\rho_{\max}(\alpha R, w, h) - \rho_{\text{th}}) / N,$$

trong đó ρ_{th} là tham số được sử dụng để chỉnh sửa khoảng cách giữa các đường tách cho khối mã. Trong ví dụ, giá trị của ρ_{th} được gán bằng $\rho_{\text{th}} = 1,5$.

Trong ví dụ khác, khoảng cách lớn nhất $\rho_{\max}$ có thể được dẫn xuất dưới dạng hàm, tùy thuộc vào góc và kích thước của khối mã theo:

$$\rho_{\max}(\alpha R, w, h) = \frac{1}{2 \sqrt{\left(\frac{\cos(\alpha R)}{w} \right)^2 + \left(\frac{\sin(\alpha R)}{h} \right)^2}}, \quad 0 \leq \alpha R < \frac{\pi}{2},$$

trong đó αR là góc theo các đơn vị radian, w là chiều rộng và h là chiều cao của khối theo các mẫu độ sáng. Điều này được lấy ví dụ trên Fig.17 với các giá trị $\Delta \text{angle} = 22,5^\circ$ và $N=4$.

Trong ví dụ, các giá trị $\Delta dist$ phụ thuộc vào giá trị của tham số biểu diễn góc, tham số biểu diễn chiều rộng và tham số biểu diễn chiều cao có thể được lưu trữ trong bảng tra cứu được tính trước, để tránh tính toán lặp lại $\Delta dist$ trong suốt quá trình mã hóa hoặc giải mã.

Trong ví dụ, các giá trị $\Delta dist$ có thể được chia tỷ lệ và làm tròn, để sử dụng số học số nguyên theo:

$$stepD = Round(\Delta dist * 2^{prec}),$$

trong đó $stepD$ là kích thước bước khoảng cách được chia tỷ lệ tuyến tính và $prec$ là biến được sử dụng để điều chỉnh độ chính xác của quá trình chia tỷ lệ. Trong ví dụ, giá trị của $prec$ được gán bằng $prec=7$.

Trong ví dụ, còn lưu trữ các giá trị được tính toán trước của $stepD$ dựa trên khuôn dạng được ký hiệu là $whRatio$, phụ thuộc vào chiều rộng và chiều cao của khối mã. Ngoài ra, các giá trị được tính toán trước của $stepD$ được lưu trữ dựa trên giá trị góc (được chuẩn hóa) $angleN$, vốn là giá trị chỉ số liên quan đến các góc ở phần tư thứ nhất của mặt phẳng Oclit (chẳng hạn, $0 \leq angleN * \Delta alpha \leq 90^\circ$). Ví dụ về bảng tra cứu này đã được áp dụng được thể hiện trên Fig.13.

Trong một ví dụ, các bước sau được áp dụng để thu được các giá trị dự báo đối với các mẫu của khối mã.

Bước 1: Đối với mẫu trong khối mã hiện tại (khối giải mã hoặc khối mã hóa), khoảng cách mẫu ($sample_dist$) được tính toán.

Trong một số ví dụ, khoảng cách mẫu có thể là khoảng cách ngang hoặc khoảng cách dọc, hoặc kết hợp của khoảng cách ngang và dọc, của mẫu nêu trên đến đường tách (đường tách được sử dụng để chỉ báo khối mã được phân chia thành hai khối phụ). Mẫu đó được biểu diễn bằng các tọa độ (x,y) với mẫu trên bên trái của khối mã. Tọa độ mẫu và $sample_dist$ được lấy ví dụ trên Fig.11 và Fig.12. Các khối này không nhất thiết là hình chữ nhật, chúng có thể có hình tam

giác hoặc hình thang.

Trong ví dụ, tham số thứ nhất là giá trị góc được lượng tử (`angleIdx`) và tham số thứ hai là giá trị khoảng cách được lượng tử (`distanceIdx`). Hai tham số mô tả phương trình đường thẳng. Trong ví dụ, khoảng cách 1230 có thể thu được theo `distanceIdx` (tham số thứ hai), và góc alpha (1210) có thể thu được theo `angleIdx` (tham số thứ nhất). Khoảng cách 1230 có thể là khoảng cách đếm tâm của khối mã, và góc này có thể là góc giữa đường tách và đường ngang (hoặc đường dọc tương đương) đi qua điểm giữa của khối mã.

Trong ví dụ, ở bước 1, khối mã được xem là phân chia thành 2 khối phụ theo các cách khác nhau. Fig.9 thể hiện ví dụ về phân vùng của khối mã, đường tách 1250 được sử dụng để chỉ báo khối được phân chia thành 2 khối phụ. Để mô tả đường 1250, một tham số góc alpha 1210 được báo hiệu trong dòng bit.

Theo một số phương án thực hiện, góc, như được chỉ báo trên Fig.9, được đo giữa trục x và đường tách, trong đó khoảng cách được đo bằng chiều dài của vectơ vuông góc với đường tách và đi qua tâm của khối hiện tại.

Trong ví dụ khác, Fig.10 thể hiện cách khác biểu diễn đường tách, trong đó các ví dụ về góc và khoảng cách khác với các ví dụ được thể hiện trên Fig. 9.

Bước 2: `sample_dist` được tính toán được sử dụng để tính toán các hệ số trọng số, các hệ số trọng số được sử dụng để kết hợp giá trị dự báo thứ nhất và giá trị dự báo thứ hai tương ứng với mẫu nêu trên. Trong ví dụ, các hệ số trọng số được ký hiệu là `sampleWeight1` và `sampleWeight2`, đề cập đến trọng số tương ứng với giá trị dự báo thứ nhất và trọng số tương ứng với giá trị dự báo thứ hai.

Bước 3: Giá trị kết hợp của mẫu dự báo ở tọa độ mẫu (x,y) được tính toán theo, giá trị dự báo thứ nhất ở tọa độ (x,y), giá trị dự báo thứ hai ở tọa độ (x,y), `sampleWeight1` và `sampleWeight2`.

Trong ví dụ, bước 1 trong các ví dụ nêu trên có thể bao gồm các bước sau:

Bước 1.1: thu được giá trị chỉ số của tham số góc (α_N) đối với khối hiện tại, giá trị của chiều rộng của khối hiện tại (W), giá trị của chiều cao của khối hiện tại (H). W và H là chiều rộng và chiều cao của khối hiện tại trong số lượng mẫu. Chẳng hạn, khối mã có chiều rộng và chiều cao đều bằng 8 là khối vuông bao gồm 64 mẫu. Trong ví dụ khác, W và H là chiều rộng và chiều cao của khối hiện tại theo số lượng mẫu độ sáng.

Bước 1.2: thu được giá trị tỷ lệ $whRatio$ theo giá trị của W và giá trị của H , giá trị của $whRatio$ là tỷ lệ giữa chiều rộng và chiều cao của khối mã hiện tại.

Bước 1.3: Thu được giá trị $stepD$ theo bảng tra cứu, giá trị của α và giá trị của $whRatio$, trong ví dụ, như được thể hiện trên Fig.13, giá trị của α và giá trị của $whRatio$ được sử dụng làm các giá trị chỉ số của bảng tra cứu.

Bước 1.4: giá trị của $sample_dist$ được tính toán theo giá trị $stepD$.

Trong ví dụ khác, bước 1 trong các ví dụ nêu trên có thể bao gồm các bước sau:

Bước 1.1: thu được giá trị của tham số góc (α_N) đối với khối hiện tại, giá trị của chỉ số khoảng cách ($distanceIdx$), giá trị của chiều rộng của khối hiện tại (W), giá trị của chiều cao của khối hiện tại (H).

Bước 1.2: thu được giá trị của tỷ lệ $whRatio$ theo giá trị của W và giá trị của H , giá trị của $whRatio$ là tỷ lệ giữa chiều rộng và chiều cao của khối mã hiện tại.

Bước 1.3: Thu được giá trị $stepD$ theo bảng tra cứu, giá trị của α và giá trị của $whRatio$, trong ví dụ, như được thể hiện trên Fig.13, giá trị của α_N và giá trị của $whRatio$ được sử dụng làm các giá trị chỉ số của bảng tra cứu. Trong ví dụ, giá trị $stepD$ là kích thước bậc lượng tử đối với quá trình tính toán khoảng cách mẫu.

Bước 1.4: giá trị của $sample_dist$ được tính toán theo giá trị $stepD$, giá trị của $distanceIdx$, giá trị của góc (α_N), giá trị của W và giá trị của H .

Trong một ví dụ, giá trị của whRatio thu được nhờ sử dụng công thức:

$$\text{whRatio} = (\text{wIdx} \geq \text{hIdx}) ? \text{wIdx} - \text{hIdx} : \text{hIdx} - \text{wIdx}$$

trong đó giá trị của wIdx bằng $\log_2(W) - 3$, và giá trị của hIdx bằng $\log_2(H) - 3$.

Trong ví dụ khác, giá trị của whRatio được tính toán bằng $\text{whRatio} = (W \geq H) ? W/H : H/W$.

Trong ví dụ, giá trị của angle alpha có thể thu được từ dòng bit (trong bộ giải mã). Trong một ví dụ, khoảng giá trị đối với góc là khoảng giá trị được lượng tử hóa giữa 0 và 31 (bao gồm 0 và 31) được ký hiệu là angleIdx. Trong một ví dụ, giá trị góc được lượng tử chỉ lấy 32 giá trị riêng khác nhau (do vậy giá trị giữa 0 đến 31 đủ để biểu diễn giá trị góc nào được chọn). Trong ví dụ khác, khoảng giá trị của giá trị góc có thể nằm giữa 0 đến 15, nghĩa là 16 giá trị góc được lượng tử khác nhau có thể được chọn. Lưu ý rằng nói chung giá trị góc có thể là giá trị nguyên lớn hơn hoặc bằng 0.

Trong một ví dụ, giá trị của alphaN là giá trị chỉ số thu được từ dòng bit, hoặc giá trị của alpha được tính toán dựa trên giá trị của bộ chỉ báo thu được từ dòng bit. Chẳng hạn, giá trị của alphaN có thể được tính toán như theo công thức:

$$\begin{aligned} \alpha N &= \text{angleIdx} && \text{nếu } \text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq \text{Lim}, \text{ hoặc} \\ \alpha N &= 2 * \text{Lim} - \text{angleIdx} && \text{nếu } \text{angleIdx} > \text{Lim} \ \&\& \ \text{angleIdx} \leq 2 * \text{Lim}, \text{ hoặc} \\ \alpha N &= \text{angleIdx} - 2 * \text{Lim} && \text{nếu } \text{angleIdx} > 2 * \text{Lim} \ \&\& \ \text{angleIdx} \leq 3 * \text{Lim}, \\ &&& \text{hoặc} \\ \alpha N &= 4 * \text{Lim} - \text{angleIdx} && \text{ngược lại.} \end{aligned}$$

trong đó angleIdx là giá trị của bộ chỉ báo thu được từ dòng bit. Giá trị của gocIdx có thể là giá trị số nguyên, khoảng giá trị đối với angleIdx giữa 0 đến 31 (bao gồm 0 và 31).

Trong ví dụ khác, giá trị alphaN có thể được tính toán theo một trong các

công thức sau:

$$\begin{aligned} \alpha N &= \text{angleIdx} && , \text{ hoặc} \\ \alpha N &= 2 * \text{Lim} - \text{angleIdx} && , \text{ hoặc} \\ \alpha N &= \text{angleIdx} - 2 * \text{Lim} && , \text{ hoặc} \\ \alpha N &= 4 * \text{Lim} - \text{angleIdx}. \end{aligned}$$

Trong công thức nêu trên, Lim là giá trị định trước không đổi liên quan đến số lượng giá trị góc được lượng tử. Chẳng hạn, nếu có 32 giá trị góc được lượng tử khác nhau, số lượng Lim có thể bằng 8 (4*Lim bằng 32). Trong ví dụ khác, Lim có thể bằng 4, tương ứng với tổng cộng 16 giá trị góc được lượng tử khác nhau.

Trong ví dụ, giá trị của sample_dist thu được theo công thức:

$$\text{sample_dist} = ((2*x) - W + K)*f1(\alpha) + ((2*y) - H + K)*f2(\alpha) - \text{distanceIdx}*stepD*scaleStep, \text{ trong đó:}$$

- f1() và f2() là các hàm số, và alpha là đầu vào cho các hàm. Trong ví dụ, f1() và f2() được thực hiện làm các bảng tra cứu, giá trị của alpha là giá trị chỉ số cho các bảng tra cứu.
- giá trị của scaleStep bằng giá trị của chiều rộng của khối mã (giải mã hoặc mã hóa) hoặc giá trị của chiều cao của khối mã.
- x và y là các tọa độ của mẫu nêu trên với mẫu trên bên trái của khối mã.
- K là giá trị độ lệch số nguyên. Trong ví dụ, giá trị cho K bằng 1.

Trong ví dụ, các hàm f1() và f2() được thực hiện dưới dạng các bảng tra cứu. Trong ví dụ, các hàm f1() và f2() biểu diễn thay đổi tăng lên trong giá trị sample_dist với việc thay đổi các giá trị x và y. Trong một số ví dụ, f1(index) biểu diễn giá trị sample_dist được thay đổi với việc tăng lên một đơn vị trong giá trị của x (đơn vị có thể tăng lên 1), trong đó f2(index) biểu diễn giá trị sample_dist được thay đổi với việc tăng lên một đơn vị trong giá trị của y. Giá trị của chỉ số có thể thu được từ giá trị của bộ chỉ báo trong dòng bit.

Trong ví dụ khác, giá trị của `sample_dist` thu được theo công thức:

$$\text{sample_dist} = ((2*x) - W + K) * f(\text{dispIdxN}) + ((2*y) - H + K) * f(\text{dispIdxS})$$

– `distanceIdx*stepD*scaleStep`, trong đó:

- $f()$ là hàm số, trong ví dụ, hàm số $f()$ được thực hiện dưới dạng bảng tra cứu.
- `dispIdxN` hoặc `dispIdxS` là các giá trị chỉ số thu được từ dòng bit hoặc được tính toán dựa trên giá trị của bộ chỉ báo trong dòng bit.
- giá trị của `scaleStep` bằng giá trị của chiều rộng của khối mã (giải mã hoặc mã hóa) hoặc giá trị của chiều cao của khối mã.
- x và y là các tọa độ của mẫu nêu trên với mẫu trên bên trái của khối mã.
- K là giá trị độ lệch số nguyên. Trong ví dụ, giá trị cho K bằng 1.

Trong ví dụ, hàm $f()$ được thực hiện dưới dạng bảng tra cứu. Hàm $f()$ biểu diễn thay đổi tăng lên trong giá trị `sample_dist` với việc thay đổi các giá trị x và y . Trong ví dụ, $f(\text{index1})$ biểu diễn giá trị `sample_dist` được thay đổi với việc tăng lên một đơn vị trong giá trị của x , trong đó $f(\text{index2})$ biểu diễn giá trị `sample_dist` được thay đổi với việc tăng lên một đơn vị trong giá trị của y . Giá trị của `index1` và `index2` là chỉ số với bảng (có giá trị số nguyên lớn hơn hoặc bằng 0), có thể thu được theo giá trị của bộ chỉ báo trong dòng bit.

Trong ví dụ, việc triển khai hàm $f()$ được minh họa trên Fig.14. Trong ví dụ này, giá trị của `idx` là tham số đầu vào (có thể là `index1` hoặc `index2`), và đầu ra của hàm được minh họa là $f(\text{idx})$. Trong ví dụ, $f()$ là triển khai của hàm cosin nhờ sử dụng số học số nguyên, trong đó `idx` (giá trị chỉ số đầu vào) biểu diễn giá trị góc được lượng tử.

Theo một phương án thực hiện, giá trị `stepD` biểu diễn giá trị khoảng cách được lượng tử để tính toán khoảng cách mẫu.

Trong ví dụ, như được thể hiện trên Fig.13, giá trị `stepD` thu được theo giá trị

của whRatio và giá trị của góc (alpha). Trong ví dụ, giá trị của stepD có thể thu được như là $\text{stepD} = \text{lookupTable}[\text{alphaN}][\text{whRatio}]$, trong đó giá trị của alphaN là giá trị chỉ số thu được từ dòng bit, hoặc giá trị của alphaN được tính toán dựa trên giá trị của bộ chỉ báo thu được từ dòng bit. Chẳng hạn, alpha có thể được tính toán dưới dạng theo công thức:

$$\begin{aligned} \alpha N &= \text{angleIdx} && \text{nếu } \text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq 8, \text{ hoặc} \\ \alpha N &= 16 - \text{angleIdx} && \text{nếu } \text{angleIdx} > 8 \ \&\& \ \text{angleIdx} \leq 16, \text{ hoặc} \\ \alpha N &= \text{angleIdx} - 16 && \text{nếu } \text{angleIdx} > 16 \ \&\& \ \text{angleIdx} \leq 24, \text{ hoặc} \\ \alpha N &= 32 - \text{angleIdx} && \text{ngược lại,} \end{aligned}$$

trong đó angleIdx là giá trị chỉ số thu được theo dòng bit.

Trong ví dụ khác:

$$\begin{aligned} \alpha N &= \text{angleIdx} && \text{nếu } \text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq \text{Lim}, \text{ hoặc} \\ \alpha N &= 2 * \text{Lim} - \text{angleIdx} && \text{nếu } \text{angleIdx} > \text{Lim} \ \&\& \ \text{angleIdx} \leq 2 * \text{Lim}, \\ &&& \text{hoặc} \\ \alpha N &= \text{angleIdx} - 2 * \text{Lim} && \text{nếu } \text{angleIdx} > 2 * \text{Lim} \ \&\& \ \text{angleIdx} \\ &&& \leq 3 * \text{Lim}, \text{ hoặc} \\ \alpha N &= 4 * \text{Lim} - \text{angleIdx} && \text{ngược lại.} \end{aligned}$$

Hoặc trong ví dụ khác:

$$\begin{aligned} \alpha N &= \text{angleIdx} && , \text{ hoặc} \\ \alpha N &= 2 * \text{Lim} - \text{angleIdx} && , \text{ hoặc} \\ \alpha N &= \text{angleIdx} - 2 * \text{Lim} && , \text{ hoặc} \\ \alpha N &= 4 * \text{Lim} - \text{angleIdx} \end{aligned}$$

Trong công thức nêu trên, Lim là giá trị định trước hằng số liên quan đến số lượng giá trị góc được lượng tử. Chẳng hạn, nếu có 32 giá trị góc được lượng tử khác nhau, thì số Lim có thể bằng 8 (4*Lim bằng 32). Trong ví dụ khác, Lim có thể bằng 4, tương ứng với tổng số 16 giá trị góc được lượng tử khác nhau.

Trong một ví dụ, giá trị của `sample_dist` thu được theo $\text{distanceIdx} * \text{stepD} * \text{scaleStep}$, trong đó `distanceIdx` là giá trị chỉ số thu được theo dòng bit, và giá trị của `scaleStep` thu được theo giá trị của chiều rộng của khối hoặc giá trị của chiều cao của khối. Kết quả của phép nhân biểu diễn khoảng cách của đường tách đến tâm điểm của khối mã (có các tọa độ $x=W/2$ và $y=H/2$).

Trong ví dụ, bảng tra cứu theo bảng định trước. Bảng định trước có các ưu điểm sau:

- Thu được khoảng cách của mẫu đến đường tách thường là phức tạp, yêu cầu giải các phương trình lượng giác không chấp nhận được khi thực hiện chuẩn tạo mã video, nhằm vào các sản phẩm tiêu dùng sản xuất với khối lượng lớn.

Trong một số ví dụ, khoảng cách mẫu thu được theo bảng tra cứu (có thể được định trước), mà tính toán trước các kết quả trung gian theo `whRatio` và `alpha`, mà đã theo số học số nguyên (do vậy, trong ví dụ, tất cả các giá trị `stepD` đều là số nguyên). Kết quả trung gian mà thu được nhờ sử dụng bảng tra cứu được chọn lựa cẩn thận do:

- Bảng tra cứu bao gồm các kết quả tính toán trung gian đối với các phép tính phức tạp (các tính toán lượng giác), do vậy độ phức tạp triển khai được giảm.
- Kích thước của bảng được duy trì nhỏ (đòi hỏi bộ nhớ).

Trong ví dụ khác, giá trị của `sample_dist` thu được theo $\text{distanceIdx} * (\text{stepD} + T) * \text{scaleStep}$, trong đó `T` là giá trị độ lệch có giá trị nguyên. Trong ví dụ, giá trị của `T` bằng 32.

Trong ví dụ khác (theo khía cạnh bộ giải mã và bộ mã hóa):

Giá trị của `sample_dist` trong các ví dụ nêu trên ký hiệu là khoảng cách giữa mẫu trong khối phụ mã hiện tại và đường tách, như được thể hiện trên Fig.11 hoặc Fig.12.

Hai khối phụ được đánh chỉ số 0 và 1. Khối phụ thứ nhất được đánh chỉ số 0

và khối phụ thứ hai được đánh chỉ số 1.

Ở chế độ phân vùng hình học, thứ tự xử lý của hai khối phụ được quyết định dựa trên chỉ số của các khối phụ. Khối phụ có chỉ số 0 sẽ được xử lý trước, và khối phụ có chỉ số 1 sẽ xử lý về sau. Thứ tự xử lý có thể được sử dụng trong quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vectơ chuyển động, v.v..

Trong một ví dụ, giá trị chỉ số của khối phụ được xác định dựa trên giá trị `sample_dist` của mẫu trong khối phụ.

Nếu giá trị `sample_dist` của mỗi mẫu trong một khối phụ nhỏ hơn (hoặc bằng) 0, thì giá trị chỉ số đối với khối phụ được gán bằng 0.

Nếu giá trị `sample_dist` của mỗi mẫu trong một khối phụ lớn hơn (hoặc bằng) 0, giá trị chỉ số đối với khối phụ này được gán bằng 1.

Trong các ví dụ tương tự, giá trị chỉ số đối với khối phụ có thể được chỉnh sửa (chẳng hạn, được đảo). Trong một ví dụ, giá trị chỉ số của khối phụ có thể được chỉnh sửa dựa trên vị trí mẫu định trước.

Trong một ví dụ, giá trị chỉ số cho khối phụ, mà vị trí định trước (chẳng hạn, vị trí dưới bên trái) của khối mã được đặt trong đó, luôn được gán bằng 0.

Trong ví dụ, nếu vị trí mẫu định trước thuộc khối phụ thứ hai có giá trị chỉ số 1, giá trị chỉ số đối với khối phụ này sẽ được chỉnh sửa bằng 0 (giá trị chỉ số đối với khối phụ khác sẽ được chỉnh sửa bằng 1).

Trong ví dụ, nếu vị trí mẫu định trước thuộc khối phụ thứ nhất có chỉ số 0, các chỉ số của các khối phụ không được thay đổi.

Như được thể hiện trên Fig.20 và Fig.21, vị trí mẫu định trước là vị trí mẫu dưới bên trái của khối mã hiện tại.

Trên Fig.20, mẫu dưới bên trái thuộc khối phụ thứ nhất có chỉ số 0, các chỉ số của các khối phụ không được thay đổi.

Trên Fig.21, mẫu dưới bên trái thuộc khối phụ thứ hai có chỉ số 1 (Fig.21a), do vậy, các chỉ số của hai khối phụ được hoán đổi hoặc đảo, như được thể hiện trên Fig.21b, thứ tự xử lý sẽ dựa trên chỉ số được hoán hoặc được đảo.

Trong một ví dụ, quá trình gán và đảo chỉ số có thể được thực hiện với các bước sau:

Bước1: Tách khối mã hiện tại thành hai khối phụ dựa trên chế độ tách;

Bước 2: tính toán giá trị `sample_dist` cho mẫu trong một khối phụ;

Nếu giá trị `sample_dist` là giá trị âm hoặc bằng 0, giá trị chỉ số cho khối phụ này được gán bằng 0 (tương ứng với khối phụ thứ nhất), và giá trị chỉ số cho khối phụ khác được gán bằng 1 (tương ứng với khối phụ thứ hai).

hoặc

Nếu giá trị `sample_dist` là giá trị dương, giá trị chỉ số cho khối phụ này được gán bằng 1 (tương ứng với khối phụ thứ hai), và giá trị chỉ số cho khối phụ khác được gán bằng 0 (tương ứng với khối phụ thứ nhất).

Bước 3: nếu giá trị `sample_dist` của vị trí mẫu định trước là giá trị dương, hoán đổi các chỉ số của hai khối phụ.

Nếu giá trị `sample_dist` của vị trí mẫu định trước là âm, các chỉ số của hai khối phụ không đổi.

Theo một phương án thực hiện sáng chế, phương pháp tạo mã được thực hiện bằng thiết bị giải mã hoặc thiết bị mã hóa được bộc lộ. Phương pháp bao gồm các bước:

S101: thu được chế độ tách (hoặc phân vùng) đối với khối mã hiện tại.

Chế độ tách được sử dụng để tách khối mã thành các khối phụ. Chế độ tách có thể được định nghĩa từ các ví dụ có góc và/hoặc khoảng cách. Góc liên quan đến phương ngang được mô tả như là phương vuông góc với đường tách. Đường tách được sử dụng để chỉ báo rằng khối mã hiện tại được phân chia thành hai

khối phụ. Khoảng cách đã mô tả khoảng cách từ tâm điểm của khối mã hiện tại đến đường tách.

Trong một số ví dụ, chế độ tách được biểu diễn theo giá trị của bộ chỉ báo trong dòng bit. Giá trị của bộ chỉ báo được sử dụng để thu được giá trị tham số góc và giá trị tham số khoảng cách.

Theo một ví dụ, các cặp (tham số thứ nhất (angle), tham số thứ hai (distance)) có thể được dẫn xuất từ phương pháp hoặc hàm định trước, nói cách khác, bộ chỉ báo của chế độ phân vùng được phân tách từ dòng bit của bộ giải mã. Bộ chỉ báo này được định nghĩa là đầu vào phương pháp hoặc hàm. Đầu ra của phương pháp hoặc hàm là các cặp (tham số thứ nhất và tham số thứ hai).

Trong một ví dụ về bộ chỉ báo (geo_partition_idx) và phương pháp hoặc hàm định trước dưới đây.

Hàm 1

Bộ chỉ báo X là đầu vào (từ 0 đến 63).

if (X<4)

Angle = 0, Dist = (X>=2)?X-4:X+1;

Else if (X>=60)

Angle = 8, Dist = (X>61)?X-64:X-59;

Else

Z=(X-4)/14;

K = Z+1+Z/3;

Angle = (X-4)%14<7 ? K: 16-K;

Dist = (X-4)%7 - 3.

Bộ chỉ báo X có giá trị giữa 0 và 63, được sử dụng để dẫn xuất tham số thứ nhất (Angle) và tham số thứ hai (Dist). Ví dụ về các kết quả đầu ra được mô tả trong Bảng 2.

Bảng 2: bảng đầu vào và đầu ra của hàm 1 trong đó geo_partition_idx (chẳng hạn, bộ chỉ báo X) là chỉ số của bảng và angleIdx (chẳng hạn, góc) là tham số thứ nhất và distanceIdx (chẳng hạn, Dist) là tham số thứ hai.

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	0	-2
3	0	-1
4	1	-3
5	1	-2
6	1	-1
7	1	0
8	1	1
9	1	2
10	1	3
11	15	-3
12	15	-2
13	15	-1
14	15	0
15	15	1
16	15	2
17	15	3
18	2	-3
19	2	-2
20	2	-1
21	2	0
22	2	1
23	2	2
24	2	3
25	14	-3
26	14	-2
27	14	-1
28	14	0
29	14	1
30	14	2
31	14	3
32	3	-3

geo_partition_idx	angleIdx	distanceIdx
33	3	-2
34	3	-1
35	3	0
36	3	1
37	3	2
38	3	3
39	13	-3
40	13	-2
41	13	-1
42	13	0
43	13	1
44	13	2
45	13	3
46	5	-3
47	5	-2
48	5	-1
49	5	0
50	5	1
51	5	2
52	5	3
53	11	-3
54	11	-2
55	11	-1
56	11	0
57	11	1
58	11	2
59	11	3
60	8	1
61	8	2
62	8	-2
63	8	-1

Ví dụ khác của bộ chỉ báo (geo_partition_idx) và phương pháp hoặc hàm định trước dưới đây.

Hàm 2

Bộ chỉ báo X là đầu vào is the input (0-63).

if ($X < 4$)

Angle = 0, Dist = $(X \geq 2) ? X - 4 : X + 1$;

Else if ($X \geq 60$)

Angle = 8, Dist = $(X > 61) ? X - 64 : X - 59$;

Else

$Z = (X - 4) / 14$;

$K = Z + 1 + Z / 3$;

Angle = $(X - 4) \% 14 < 7 ? K : 16 - K$;

Dist = $(X - 4) \% 7 - 3$.

If (Dist < 0)

Angle = Angle + 16;

Dist = - Dist.

Bộ chỉ báo X có giá trị giữa 0 và 63, được sử dụng để dẫn xuất tham số thứ nhất (Angle) và tham số thứ hai (Dist). Ví dụ của các kết quả đầu ra được mô tả trong Bảng 3.

Bảng 3: bảng đầu vào và đầu ra của hàm 2 trong đó geo_partition_idx (chẳng hạn, bộ chỉ là X) là chỉ số to của bảng và angleIdx (chẳng hạn, Angle) là tham số thứ nhất và distanceIdx (chẳng hạn, Dist) là tham số thứ hai.

geo_partition_idx	angleIdx	distanceIdx
0	0	1
1	0	2
2	16	2
3	16	1
4	17	3
5	17	2
6	17	1
7	1	0
8	1	1
9	1	2

geo_partition_idx	angleIdx	distanceIdx
10	1	3
11	31	3
12	31	2
13	31	1
14	15	0
15	15	1
16	15	2
17	15	3
18	18	3
19	18	2
20	18	1
21	2	0
22	2	1
23	2	2
24	2	3
25	30	3
26	30	2
27	30	1
28	14	0
29	14	1
30	14	2
31	14	3
32	19	3
33	19	2
34	19	1
35	3	0
36	3	1
37	3	2
38	3	3
39	29	3
40	29	2
41	29	1
42	13	0
43	13	1
44	13	2
45	13	3
46	21	3

geo_partition_idx	angleIdx	distanceIdx
47	21	2
48	21	1
49	5	0
50	5	1
51	5	2
52	5	3
53	27	3
54	27	2
55	27	1
56	11	0
57	11	1
58	11	2
59	11	3
60	8	1
61	8	2
62	24	2
63	24	1

S102: khối mã hiện tại được xem xét tách thành hai khối phụ (khối phụ A, khối phụ B) dựa trên chế độ tách.

Trong bước này, theo chế độ tách, các mẫu trong khối mã được xem xét là được đặt vào hai khối phụ. Khối phụ A hoặc khối phụ B bao gồm phần mẫu (nhưng không phải tất cả) trong khối mã hiện tại. Khối phụ A hoặc khối phụ B có thể được biểu diễn theo dấu của sample_dist của mỗi mẫu. sample_dist có thể thu được theo các ví dụ nêu trên và các phương án thực hiện.

Trong các ví dụ tương tự, bước này có thể đề cập đến các phương án thực hiện hoặc các ví dụ nêu trên về việc phân chia khối mã thành hai khối phụ. Chẳng hạn, khối mã có thể được phân chia thành 2 khối phụ theo mô hình hình học. Mô hình này có thể dẫn đến việc tách khối bằng đường tách (chẳng hạn, đường thẳng) như được lấy ví dụ trên các hình vẽ từ Fig.9 đến Fig.12.

Fig.9 thể hiện ví dụ về phân vùng khối mã, khối được xem xét phân chia thành 2 khối phụ theo đường tách 1250. Để mô tả đường 1250, hai tham số được

báo hiệu, một tham số là góc α 1210 và tham số còn lại là khoảng cách dist 1230.

Theo một số phương án thực hiện, góc, như được chỉ báo trên Fig.9, được đo giữa trục x và đường tách, trong đó khoảng cách được đo bằng chiều dài của vectơ vuông góc với đường tách và đi qua tâm của khối hiện tại.

Trong ví dụ khác, Fig.10 thể hiện cách khác biểu diễn đường tách, trong đó các ví dụ về góc và khoảng cách là khác nhau với các ví dụ được thể hiện trên Fig.9.

Ở phía bộ mã hóa, đường tách (các tham số định nghĩa đường, chẳng hạn, góc và khoảng cách) được xác định dựa trên hàm chi phí dựa trên méo dạng tốc độ. Các tham số đường được xác định được mã hóa thành dòng bit. Ở phía bộ giải mã, các tham số đường được giải mã (thu được) theo dòng bit.

Trong ví dụ, trong trường hợp 3 kênh video, bao gồm thành phần độ sáng (luma) và hai thành phần sắc độ (chroma). Quá trình tách này có thể được thực hiện cho độ sáng hoặc sắc độ.

Theo một phương án thực hiện, đối với mẫu trong khối mã, giá trị khoảng cách được sử dụng để quyết định trong khối phụ nào (khối phụ A hoặc khối phụ B) mẫu này được đặt. Trong ví dụ, đối với mỗi mẫu trong khối mã, giá trị khoảng cách thu được.

Trong ví dụ, khi giá trị khoảng cách tương ứng với mẫu nhỏ hơn hoặc bằng 0, thì mẫu này nằm trong khối phụ A (hoặc khối phụ B); khi giá trị khoảng cách tương ứng với mẫu lớn hơn 0, thì mẫu này nằm trong khối phụ B (hoặc khối phụ A).

Trong ví dụ khác, khi giá trị khoảng cách tương ứng với mẫu nhỏ hơn 0, thì mẫu này nằm trong khối phụ A (hoặc khối phụ B); khi giá trị khoảng cách tương ứng với mẫu lớn hơn hoặc bằng 0, thì mẫu này nằm trong khối phụ B (hoặc khối

phụ A).

Các quá trình thu thập giá trị khoảng cách (chẳng hạn, `sample_dist`) tương ứng với mẫu có thể đề cập đến các ví dụ nêu trên và các phương án thực hiện, chẳng hạn, `sample_dist` là giá trị khoảng cách tương ứng với mẫu.

S103: thiết lập giá trị chỉ số của khối phụ A theo vị trí mẫu định trước trong khối mã hiện tại.

Trong một số ví dụ, vị trí mẫu định trước có thể là vị trí dưới bên trái, hoặc vị trí dưới bên phải, hoặc vị trí trên bên trái hoặc trên bên phải hoặc vị trí tâm của khối mã hiện tại.

Khi vị trí mẫu định trước được đặt ở khối phụ A, giá trị chỉ số của khối phụ A được gán bằng giá trị thứ nhất, trong ví dụ này, giá trị chỉ số của khối phụ B sẽ được gán bằng giá trị thứ hai.

hoặc

Khi vị trí mẫu định trước không được đặt trong khối phụ A (chẳng hạn, vị trí mẫu định trước được đặt trong khối phụ B), giá trị chỉ số của khối phụ A được gán bằng giá trị thứ hai, trong ví dụ này, giá trị chỉ số của khối phụ B sẽ được gán bằng giá trị thứ nhất.

Do thứ tự xử lý tương ứng với giá trị thứ nhất đứng trước thứ tự xử lý tương ứng với giá trị thứ hai, do vậy, khối phụ trong đó đặt vị trí mẫu định trước luôn được xử lý trước khối phụ còn lại. Bước xử lý có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vector chuyển động, v.v.. Trong một số ví dụ, trong chế độ phân vùng hình học, thứ tự xử lý của hai khối phụ được quyết định dựa trên chỉ số của các khối phụ. Khối phụ có chỉ số 0 sẽ được xử lý trước, và khối phụ có chỉ số 1 sẽ được xử lý sau.

Giá trị thứ nhất hoặc giá trị thứ hai có thể là giá trị nguyên. Trong ví dụ, giá

trị thứ nhất có thể bằng 0 hoặc 1 hoặc các giá trị khác, giá trị thứ hai có thể bằng 0 hoặc 1 hoặc các giá trị khác. Giá trị thứ nhất và giá trị thứ hai là khác nhau.

Trong ví dụ, liệu vị trí mẫu định trước có được đặt trong khối phụ A có thể được xác định dựa trên giá trị khoảng cách mẫu (chẳng hạn, `sample_dist`) đối với vị trí mẫu định trước. Chẳng hạn, nếu giá trị khoảng cách mẫu tương ứng với mẫu trong khối phụ A lớn hơn 0, và giá trị khoảng cách mẫu tương ứng với vị trí mẫu định trước nhỏ hơn 0, thì vị trí mẫu định trước không nằm trong khối phụ A, nhưng trong khối phụ B. Trong ví dụ này, mỗi mẫu trong khối phụ B có giá trị khoảng cách mẫu nhỏ hơn 0. Trong một số ví dụ khác, mỗi mẫu trong khối phụ B có thể có giá trị khoảng cách mẫu nhỏ hơn hoặc bằng 0, nó phụ thuộc liệu mẫu tương ứng với giá trị khoảng cách mẫu bằng 0 được xem xét trong khối phụ B hay không.

Trong ví dụ khác, liệu vị trí mẫu định trước được đặt trong khối phụ A có thể được xác định dựa trên giá trị tọa độ đối với vị trí mẫu định trước.

Phương án thực hiện 2

Theo phương án thực hiện khác, chỉ số của khối phụ hoặc khối được gán bằng hàm định trước hoặc bảng tra cứu, giá trị đầu vào của hàm định trước hoặc bảng tra cứu có thể là biến hoặc giá trị tham số, đầu ra của hàm định trước hoặc bảng tra cứu là giá trị chỉ số của khối phụ hoặc khối.

Trong ví dụ, chỉ số của khối là chỉ số tách biểu diễn dưới dạng `isFlip`, giá trị chỉ số `isFlip` thỏa mãn

$$\text{isFlip} = (\text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27) ? 1 : 0.$$

Trong một ví dụ, giá trị đầu vào của hàm định trước hoặc bảng tra cứu có thể được phân tách từ dòng bit.

Trong một ví dụ, giá trị đầu vào của hàm định trước hoặc bảng tra cứu có thể

được dẫn xuất dựa trên giá trị của thông tin chỉ báo.

Trong một ví dụ, giá trị đầu vào là giá trị của chỉ số góc của đường biên tách, được dẫn xuất từ cú pháp geo_partition_idx.

Trong một ví dụ, giá trị chỉ số (index) của khối phụ A của phân vùng hình học có thể được dẫn xuất từ bảng tra cứu sau. Trong đó giá trị của partIdx[AngleIdx] bằng 0 nghĩa là chỉ số phần của khối phụ A được tách từ phân vùng hình học bằng 0, giá trị của partIdx[AngleIdx] bằng 1 nghĩa là chỉ số phần của khối phụ A được tách từ phân vùng hình học bằng 1.

Bảng 4 - - Bảng tra cứu PartIdx cho khối phụ A dựa trên AngleIdx.

AngleIdx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
partIdx[AngleIdx]	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
AngleIdx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
partIdx[AngleIdx]	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

Bảng 4 có thể được biến đổi ở dạng hàm hoặc phương trình dưới dạng:

$$\text{partIdx} = \text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27 ? 1 : 0.$$

PartIdx được sử dụng để tách phần đảo cho chế độ phân vùng hình học.

Trong ví dụ, do khối phụ A và khối phụ B được tách từ một phân vùng hình học, khối phụ B sẽ có chỉ số đối (chẳng hạn, giá trị 1 tương ứng với giá trị 0, hoặc giá trị 0 tương ứng với giá trị 1) của khối phụ A.

Trong một số ví dụ, thứ tự xử lý tương ứng với giá trị thứ nhất (0 trong ví dụ nêu trên) đứng trước thứ tự xử lý tương ứng với giá trị thứ hai (1 trong ví dụ nêu trên). Do vậy, khối phụ đặt vị trí mẫu định trước luôn được xử lý trước khối phụ còn lại. Bước xử lý có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, hoặc quá trình dẫn xuất vector chuyển động, v.v.. Trong một số ví dụ, trong chế độ phân vùng hình học, thứ tự xử lý của hai khối phụ được quyết định dựa trên chỉ số của khối phụ. Khối phụ có chỉ số 0 sẽ được

xử lý trước, và khối phụ có chỉ số 1 sẽ được xử lý sau.

Trong một số ví dụ, trong chế độ phân vùng hình học, giá trị của bộ dự báo bằng $\text{predictorA} * \text{Mask1} + \text{predictorB} * \text{Mask2}$, trong đó giá trị của bộ dự báo A hoặc giá trị của bộ dự báo B có thể được dẫn xuất từ chế độ khác (chẳng hạn, chế độ hợp nhất), Mask 1 và Mask 2 là các mặt nạ kết hợp cho các khối phụ được tách từ phân vùng hình học. Các mặt nạ kết hợp được tính toán trước dựa trên các bộ chỉ báo khác. Trong một ví dụ, angleIdx và distanceIdx được dẫn xuất từ geo_partition_idx , geo_partition_idx được phân tách từ dòng bit. Khoảng cách giữa vị trí mẫu và đường biên được tách được tính toán dựa trên angleIdx và distanceIdx , mặt nạ kết hợp tùy thuộc vào khoảng cách này nhờ sử dụng bảng tra cứu định trước.

Trong ví dụ, giá trị chỉ số (chẳng hạn, 0 hoặc 1) của khối phụ được tách từ phân vùng hình học có thể được sử dụng khi lựa chọn các mặt nạ bộ chỉ báo cho các khối phụ.

Trong một ví dụ, nếu giá trị chỉ số của khối phụ là giá trị thứ nhất (chẳng hạn, 0), Mask 1 được sử dụng cho khối phụ này để tạo giá trị của bộ dự báo; nếu giá trị chỉ số của khối phụ là giá trị thứ hai (chẳng hạn, 1), Mask 2 được sử dụng cho khối phụ này để tạo giá trị của bộ dự báo.

Trong một ví dụ, nếu giá trị chỉ số của khối phụ A bằng 0, thì Mask 1 được sử dụng cho khối phụ A, Mask 2 được sử dụng cho khối phụ B (trong ví dụ này, giá trị chỉ số của khối phụ B bằng 1).

$$\text{Final predictor} = \text{predictorA} * \text{Mask1} + \text{predictorB} * \text{Mask2}.$$

Trong một ví dụ, nếu giá trị chỉ số của khối phụ A bằng 1, thì Mask 2 được sử dụng cho khối phụ A, Mask1 được sử dụng cho khối phụ B (trong ví dụ này, giá trị chỉ số của khối phụ B bằng 0).

$$\text{Final predictor} = \text{predictorA} * \text{Mask2} + \text{predictorB} * \text{Mask1}.$$

Ở dạng tổng quát:

Final predictor = PartIdx[AngleIdx] ?

predictorA * Mask2 + predictorB * Mask1 :

predictorA * Mask1 + predictorB * Mask2.

Fig. 22 thể hiện phương pháp tạo mã được thực hiện bằng thiết bị giải mã, phương pháp bao gồm:

S2201: thu được giá trị chỉ số chế độ tách đối với khối mã hiện tại.

Theo một phương án thực hiện, chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng đối với khối mã hiện tại. Chẳng hạn, geo_partition_idx hoặc merge_gpm_partition_idx.

Trong ví dụ, merge_gpm_partition_idx[x0][y0] hoặc geo_partition_idx xác định hình dạng phân vùng của chế độ hợp nhất phân vùng hình học. Các chỉ số mảng x0, y0 xác định vị trí (x0, y0) của mẫu độ sáng trên bên trái của khối mã được xem xét so với mẫu độ sáng trên bên trái của ảnh.

Khi merge_gpm_partition_idx[x0][y0] hoặc geo_partition_idx[x0][y0] không hiện có, được suy ra bằng 0.

Trong ví dụ, giá trị chỉ số chế độ tách có thể thu được bằng cách phân tách giá trị chỉ số được mã hóa trong dòng bit video, hoặc giá trị chỉ số chế độ tách có thể được xác định theo giá trị cú pháp được phân tách từ dòng bit video.

Dòng bit có thể thu được theo mạng không dây hoặc mạng hữu tuyến. Dòng bit có thể được truyền từ website, máy chủ, hoặc nguồn từ xa khác nhờ sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn, đường thuê bao số (digital subscriber line, DSL), hoặc các công nghệ không dây, chẳng hạn, hồng ngoại, vô tuyến, vi sóng, WIFI, Bluetooth, LTE hoặc 5G.

Theo phương án thực hiện, dòng bit là chuỗi bit, chẳng hạn, ở dạng dòng đơn vị NAL hoặc dòng byte, biểu diễn chuỗi đơn vị truy nhập (access unit, (AU) tạo một hoặc nhiều CVS.

Theo một số phương án thực hiện, đối với quá trình giải mã, phía bộ giải mã đọc dòng bit và dẫn xuất các ảnh được giải mã từ dòng bit; đối với quá trình mã hóa, phía bộ mã hóa tạo dòng bit.

Thông thường, dòng bit sẽ bao gồm các phần tử cú pháp được tạo bởi cấu trúc cú pháp.

Phần tử cú pháp: Phần tử của dữ liệu được biểu diễn trong dòng bit.

Cấu trúc cú pháp: Không hoặc nhiều phần tử cú pháp cùng xuất hiện trong dòng bit theo thứ tự cụ thể.

Trong ví dụ cụ thể, các định dạng dòng bit xác định mối quan hệ giữa dòng đơn vị NAL và dòng byte, một trong số đó được gọi là dòng bit.

Dòng bit có thể, chẳng hạn, ở một trong hai định dạng: định dạng dòng đơn vị NAL hoặc định dạng dòng byte. Định dạng dòng đơn vị NAL về mặt khái niệm là loại “cơ bản” hơn. Định dạng dòng đơn vị NAL bao gồm chuỗi cấu trúc cú pháp được gọi là các đơn vị NAL. Chuỗi này được sắp thứ tự theo thứ tự giải mã. Có các ràng buộc đối với thứ tự giải mã (và nội dung) của các đơn vị NAL trong dòng đơn vị NAL.

Định dạng dòng byte có thể được tạo từ định dạng dòng đơn vị NAL bằng cách sắp thứ tự các đơn vị NAL theo thứ tự giải mã và tạo tiền tố mỗi đơn vị NAL với tiền tố mã bắt đầu và 0 hoặc nhiều byte có giá trị 0 để tạo dòng byte. Định dạng dòng đơn vị NAL có thể được trích xuất từ định dạng dòng byte bằng cách tìm kiếm vị trí của mẫu hình tiền tố mã bắt đầu duy nhất trong dòng byte này.

Theo phương án thực hiện, mệnh đề này xác định mối quan hệ giữa các ảnh nguồn và được giải mã được tạo qua dòng bit.

Nguồn video được biểu diễn bằng dòng bit là chuỗi ảnh theo thứ tự giải mã.

Thông thường, giá trị của `merge_gpm_partition_idx[x0][y0]` được giải mã từ dòng bit. Trong ví dụ, khoảng giá trị đối với `merge_gpm_partition_idx[][]` từ 0 đến 63, bao gồm 0 và 63. Trong ví dụ, quá trình giải mã cho `merge_gpm_partition_idx[][]` là “bypass”.

Khi `merge_gpm_partition_idx[x0][y0]` không hiện có, được suy ra bằng 0.

S2202: thu được giá trị chỉ số góc `angleIdx` đối với khối mã hiện tại theo giá trị chỉ số chế độ tách và bảng;

Theo một phương án thực hiện, trong đó `angleIdx` được sử dụng đối với phân vùng hình học của khối mã hiện tại.

Trong ví dụ, `angleIdx` xác định chỉ số góc của phân vùng hình học.

Giá trị của tham số góc đối với khối hiện tại thu được theo giá trị của giá trị chỉ số chế độ tách và bảng tra cứu định trước.

Theo phương án thực hiện, biến góc phân vùng `angleIdx` (tham số góc) và biến khoảng cách `distanceIdx` của chế độ phân vùng hình học được thiết lập theo giá trị của `merge_gpm_partition_idx[xCb][yCb]` (bộ chỉ báo) như được xác định trong bảng dưới đây. Theo cách thực hiện, mối quan hệ này có thể được thực hiện theo Bảng 1 hoặc theo hàm.

Bảng 1: Mô tả `angleIdx` và `distanceIdx` dựa trên `merge_gpm_partition_idx`.

<code>merge_gpm_partition_idx</code>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
<code>angleIdx</code>	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
<code>distanceIdx</code>	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
<code>merge_gpm_partition_idx</code>	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
<code>angleIdx</code>	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
<code>distanceIdx</code>	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
<code>merge_gpm_partition_idx</code>	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47

angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
merge_gpm_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

S2203: thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx;

Theo một phương án thực hiện, trong đó giá trị chỉ số partIdx thỏa mãn:

$\text{partIdx} = (\text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2}) ? 1 : 0,$

trong đó threshold1 và threshold2 là các giá trị nguyên, và threshold1 nhỏ hơn threshold2.

Trong ví dụ, trong đó threshold1 bằng 13 và threshold2 bằng 27.

Trong ví dụ, giá trị chỉ số có thể là chỉ số tách biểu diễn dưới dạng isFlip, giá trị chỉ số isFlip also thỏa mãn

$\text{isFlip} = (\text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27) ? 1 : 0.$

S2204: giải mã khối mã hiện tại theo giá trị chỉ số partIdx.

Quá trình giải mã có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vector chuyển động và v.v.. Do vậy, hiệu suất sử dụng và giải mã bộ đệm đã được cải thiện.

Theo một phương án thực hiện, trong đó việc giải mã khối mã hiện tại bao gồm bước lưu trữ thông tin chuyển động đối với khối hiện tại theo giá trị chỉ số partIdx.

Theo các phương án thực hiện nêu trên, khối mã được giải mã theo giá trị chỉ số. Quá trình giải mã có thể là quá trình dẫn xuất các trọng số của mẫu, quá trình lưu trữ thông tin chuyển động, quá trình dẫn xuất vector chuyển động và v.v.. Do vậy, hiệu suất sử dụng và giải mã bộ đệm đã được cải thiện.

Fig. 23 thể hiện bộ giải mã video, trong đó bộ giải mã bao gồm: khối phân tách 2301, được tạo cấu hình để thu được giá trị chỉ số chế độ tách đối với khối

mã hiện tại; khối nhận giá trị chỉ số góc 2302, được tạo cấu hình để thu được giá trị chỉ số góc angleIdx đối với khối mã hiện tại theo giá trị chỉ số chế độ tách và băng; khối thiết lập 2303, được tạo cấu hình để thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; khối xử lý 2304, được tạo cấu hình để giải mã khối mã hiện tại theo giá trị chỉ số partIdx.

Phương pháp theo Fig.22 có thể được thực hiện bằng bộ giải mã theo Fig.23. Các dấu hiệu và các dạng triển khai khác của các phương pháp nêu trên tương ứng với các dấu hiệu và các dạng triển khai của bộ giải mã.

Trong một ví dụ cụ thể, đối với quá trình lưu trữ MV cho chế độ phân vùng hình học.

Quá trình này được gọi là giải mã đơn vị mã với MergeGpmFlag[xCb][yCb] bằng 1.

Các đầu vào cho quá trình này là:

vị trí độ sáng (xCb, yCb) xác định mẫu trên bên trái của khối mã hiện tại so với mẫu độ sáng trên bên trái của ảnh hiện tại,

biến cbWidth xác định chiều rộng của khối mã hiện tại trong các mẫu độ sáng,

biến cbHeight xác định chiều cao của khối mã hiện tại trong các mẫu độ sáng,

biến angleIdx xác định chỉ số góc của phân vùng hình học,

biến distanceIdx xác định chỉ số khoảng cách của phân vùng hình học,

các MV độ sáng có độ chính xác mẫu phân số 1/16 mvA và mvB,

các chỉ số tham chiếu refIdxA và refIdxB,

các cờ danh sách dự báo predListFlagA và predListFlagB.

Các biến numSbX và numSbY, xác định số lượng khối 4×4 trong khối mã hiện tại theo các phương ngang và dọc, một cách lần lượt, được thiết lập bằng $cbWidth >> 2$ và $cbHeight >> 2$, một cách lần lượt.

Các biến, displacementX, displacementY, isFlip và shiftHor được dẫn xuất như sau:

```
displacementX = angleIdx
displacementY = (angleIdx + 8) % 32
isFlip = (angleIdx >= 13 && angleIdx <= 27) ? 1 : 0
shiftHor = (angleIdx % 16 == 8 || (angleIdx % 16 != 0 &&
cbHeight >= cbWidth)) ? 0 : 1
```

Các biến offsetX và offsetY được dẫn xuất như sau:

Nếu shiftHor bằng 0, phương trình sau đúng:

```
offsetX = (-cbWidth) >> 1
offsetY = ((-cbHeight) >> 1) +
(angleIdx < 16 ? (distanceIdx * cbHeight) >> 3 : -((distanceIdx * cbHeight) >> 3))
```

Ngược lại (shiftHor bằng 1), phương trình sau đúng:

```
offsetX = ((-cbWidth) >> 1) +
(angleIdx < 16 ? (distanceIdx * cbWidth) >> 3 : -((distanceIdx * cbWidth) >> 3))
offsetY = (-cbHeight) >> 1
```

Đối với mỗi khối phụ 4×4 ở chỉ số khối phụ (xSbIdx, ySbIdx) có xSbIdx = 0..numSbX - 1, và ySbIdx = 0..numSbY - 1, phương trình sau đúng:

Biến motionIdx được tính toán dựa trên mảng disLut được xác định trong Bảng 37 như sau:

```
motionIdx = (((4 * xSbIdx + offsetX) << 1) + 5) * disLut[displacementX]
+ (((4 * ySbIdx + offsetY) << 1) + 5) * disLut[displacementY]
```

Biến sType được dẫn xuất như sau:

$sType = \text{abs}(\text{motionIdx}) < 32 ? 2 : (\text{motionIdx} \leq 0 ? (1 - \text{isFlip}) : \text{isFlip})$

Tùy thuộc vào giá trị của $sType$, thực hiện các phép gán sau:

Nếu $sType$ bằng 0, phương trình sau đúng:

$\text{predFlagL0} = (\text{predListFlagA} == 0) ? 1 : 0$

$\text{predFlagL1} = (\text{predListFlagA} == 0) ? 0 : 1$

$\text{refIdxL0} = (\text{predListFlagA} == 0) ? \text{refIdxA} : -1$

$\text{refIdxL1} = (\text{predListFlagA} == 0) ? -1 : \text{refIdxA}$

$\text{mvL0}[0] = (\text{predListFlagA} == 0) ? \text{mvA}[0] : 0$

$\text{mvL0}[1] = (\text{predListFlagA} == 0) ? \text{mvA}[1] : 0$

$\text{mvL1}[0] = (\text{predListFlagA} == 0) ? 0 : \text{mvA}[0]$

$\text{mvL1}[1] = (\text{predListFlagA} == 0) ? 0 : \text{mvA}[1]$

Ngược lại, nếu $sType$ bằng 1 hoặc ($sType$ bằng 2 và $\text{predListFlagA} + \text{predListFlagB}$ không bằng 1), phương trình sau áp dụng:

$\text{predFlagL0} = (\text{predListFlagB} == 0) ? 1 : 0$

$\text{predFlagL1} = (\text{predListFlagB} == 0) ? 0 : 1$

$\text{refIdxL0} = (\text{predListFlagB} == 0) ? \text{refIdxB} : -1$

$\text{refIdxL1} = (\text{predListFlagB} == 0) ? -1 : \text{refIdxB}$

$\text{mvL0}[0] = (\text{predListFlagB} == 0) ? \text{mvB}[0] : 0$

$\text{mvL0}[1] = (\text{predListFlagB} == 0) ? \text{mvB}[1] : 0$

$\text{mvL1}[0] = (\text{predListFlagB} == 0) ? 0 : \text{mvB}[0]$

$\text{mvL1}[1] = (\text{predListFlagB} == 0) ? 0 : \text{mvB}[1]$

Ngược lại ($sType$ bằng 2 và $\text{predListFlagA} + \text{predListFlagB}$ bằng 1), phương trình sau áp dụng:

$\text{predFlagL0} = 1$

$\text{predFlagL1} = 1$

$\text{refIdxL0} = (\text{predListFlagA} == 0) ? \text{refIdxA} : \text{refIdxB}$

$$\text{refIdxL1} = (\text{predListFlagA} == 0) ? \text{refIdxB} : \text{refIdxA}$$

$$\text{mvL0}[0] = (\text{predListFlagA} == 0) ? \text{mvA}[0] : \text{mvB}[0]$$

$$\text{mvL0}[1] = (\text{predListFlagA} == 0) ? \text{mvA}[1] : \text{mvB}[1]$$

$$\text{mvL1}[0] = (\text{predListFlagA} == 0) ? \text{mvB}[0] : \text{mvA}[0]$$

$$\text{mvL1}[1] = (\text{predListFlagA} == 0) ? \text{mvB}[1] : \text{mvA}[1]$$

Thực hiện các phép gán sau cho $x = 0..3$ và $y = 0..3$:

$$\text{MvL0}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{mvL0}$$

$$\text{MvL1}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{mvL1}$$

$$\text{MvDmvrL0}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{mvL0}$$

$$\text{MvDmvrL1}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{mvL1}$$

$$\text{RefIdxL0}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{refIdxL0}$$

$$\text{RefIdxL1}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{refIdxL1}$$

$$\text{PredFlagL0}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{predFlagL0}$$

$$\text{PredFlagL1}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = \text{predFlagL1}$$

$$\text{BcwIdx}[(\text{xSbIdx} \ll 2) + x][(\text{ySbIdx} \ll 2) + y] = 0$$

Theo phương án thực hiện, Fig.24 là sơ đồ khối thể hiện bộ giải mã theo sáng chế. Bộ giải mã 2400 bao gồm bộ xử lý 2401 và vật ghi máy tính đọc được bất biến 2402 được ghép nối với bộ xử lý 2401 và lưu trữ các lệnh chương trình để thực thi bằng bộ xử lý 2401, trong đó chương trình các lệnh, khi được thực thi bằng bộ xử lý, tạo cấu hình bộ giải mã để thực hiện phương pháp theo khía cạnh thứ nhất hoặc triển khai bất kỳ hoặc phương án thực hiện của nó.

Ví dụ 1. Phương pháp tạo mã được thực hiện bằng thiết bị giải mã hoặc thiết bị mã hóa, phương pháp bao gồm các bước:

thu được chế độ tách đối với khối mã hiện tại;

tách khối mã hiện tại thành hai khối phụ (khối phụ A, khối phụ B) dựa trên chế độ tách;

thiết lập giá trị chỉ số của khối phụ A theo hàm định trước hoặc bảng định trước.

Ví dụ 2. Phương pháp theo ví dụ 1, trong đó giá trị đầu vào cho hàm định trước hoặc bảng định trước là giá trị chỉ số góc đối với khối mã hiện tại (chẳng hạn, giá trị góc được sử dụng đối với phân vùng hình học của khối mã hiện tại).

Ví dụ 3. Phương pháp theo ví dụ 1 hoặc 2, trong đó hàm định trước là,

$\text{partIdx} = \text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2} ? 1 : 0,$

trong đó AngleIdx là giá trị chỉ số góc đối với khối mã hiện tại, partIdx[AngleIdx] là giá trị chỉ số của khối phụ A, threshold 1 và threshold2 là các giá trị nguyên, và threshold1 (chẳng hạn, giá trị của threshold 1 bằng 13) nhỏ hơn threshold2 (chẳng hạn, giá trị của threshold2 bằng 27).

Ví dụ 4. Phương pháp theo ví dụ bất kỳ trong các ví dụ từ 1 đến 3, trong đó bảng định trước là,

AngleIdx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
partIdx[A ngleIdx]	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
AngleIdx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
partIdx[A ngleIdx]	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

trong đó AngleIdx là giá trị chỉ số góc đối với khối mã hiện tại, partIdx[AngleIdx] là giá trị chỉ số của khối phụ A.

Ví dụ 5. Phương pháp tạo mã được thực hiện bằng thiết bị giải mã hoặc thiết bị mã hóa, trong đó phương pháp bao gồm các bước:

thu được giá trị của tham số góc cho khối hiện tại;

thu được giá trị của tham số khoảng cách đối với khối hiện tại;

tính toán giá trị khoảng cách mẫu cho mẫu định trước (chẳng hạn, mẫu đặt ở vị trí dưới bên trái, hoặc dưới bên phải, hoặc trên bên trái hoặc trên bên phải

trong khối hiện tại) trong khối hiện tại theo giá trị của tham số góc và tham số khoảng cách;

thiết lập hai giá trị chỉ số cho hai khối phụ trong khối hiện tại, dựa trên giá trị khoảng cách mẫu cho mẫu định trước.

Ví dụ 6. Phương pháp theo ví dụ 5, trong đó việc thiết lập hai giá trị chỉ số cho hai khối phụ trong khối hiện tại, dựa trên giá trị khoảng cách mẫu cho mẫu định trước bao gồm:

thiết lập giá trị chỉ số cho một khối phụ, mà mẫu định trước được đặt trong đó, bằng giá trị thứ nhất;

thiết lập giá trị chỉ số đối với khối phụ còn lại bằng giá trị thứ hai; trong đó thứ tự xử lý tương ứng với giá trị thứ nhất đứng trước thứ tự xử lý tương ứng với giá trị thứ hai.

Ví dụ 7. Phương pháp theo ví dụ bất kỳ trong các ví dụ từ 1 đến 6, trong đó phương pháp còn bao gồm bước:

lựa chọn giá trị hoặc mặt nạ kết hợp (chẳng hạn, Mask 1 hoặc Mask 2) dựa trên giá trị chỉ số cho khối phụ.

Ví dụ 8. Bộ giải mã (30) hoặc bộ mã hóa bao gồm hệ mạch xử lý để thực hiện phương pháp theo ví dụ bất kỳ trong các ví dụ từ 1 đến 7.

Ví dụ 9. Sản phẩm chương trình máy tính bao gồm mã chương trình để thực hiện phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 7.

Ví dụ 10. Bộ giải mã hoặc bộ mã hóa, bao gồm:

một hoặc nhiều bộ xử lý; và

vật ghi máy tính đọc được bất biến được ghép nối với các bộ xử lý and lưu trữ chương trình để thực thi bằng các bộ xử lý, trong đó chương trình, khi được thực thi bằng các bộ xử lý, tạo cấu hình bộ giải mã để thực hiện phương pháp theo ví dụ bất kỳ trong các ví dụ từ 1 đến 7.

Các toán tử số học

Các toán tử số học được sử dụng trong bản mô tả giống như các toán tử được sử dụng trong ngôn ngữ lập trình C. Tuy nhiên, các kết quả chia số nguyên và các phép dịch số học được định nghĩa chính xác hơn, và các phép cộng được định nghĩa, chẳng hạn nâng lũy thừa và phép chia giá trị thực. Các quy ước đánh số và đếm thường bắt đầu từ 0, chẳng hạn, “thứ nhất” tương đương với thứ 0, “thứ hai” tương đương với thứ nhất, v.v..

Các toán tử số học

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

+	Phép cộng
–	Phép trừ (như là toán tử hai đối số) hoặc phép trừ (như là toán tử tiền tố đơn nguyên)
*	Phép nhân, bao gồm phép nhân ma trận
x^y	Nâng lũy thừa. Xác định x nâng lên lũy thừa y. Trong các ngữ cảnh khác, ký pháp này được sử dụng làm chỉ số trên sẽ không được hiểu như là nâng lũy thừa.
/	Phép chia số nguyên có rút gọn kết quả về 0. Chẳng hạn, $7 / 4$ và $-7 / -4$ được rút gọn thành 1 và $-7 / 4$ và $7 / -4$ được rút gọn thành -1.
÷	Được sử dụng để ký hiệu phép chia thành các phương trình số học trong đó không nhằm rút gọn hoặc làm tròn.
$\frac{x}{y}$	Được sử dụng để ký hiệu phép chia trong các phương trình toán học trong đó không nhằm rút gọn hoặc làm tròn.
$\sum_{i=x}^y f(i)$	Tính tổng $f(i)$ với i lấy tất cả các giá trị số nguyên từ x lên đến y và bao gồm y.

$x \% y$	Modulus. Phần dư của x chia cho y, được định nghĩa chỉ cho các số nguyên x và y với $x \geq 0$ và $y > 0$.
----------	---

Các toán tử logic

Các toán tử logic sau được định nghĩa như sau:

$x \&\& y$ Logic boolean “and” của x và y

$x || y$ Logic boolean “or” của x và y

$!$ Logic boolean “not”

$x ? y : z$ nếu x là TRUE hoặc không bằng 0, nâng lên giá trị của y; ngược lại, nâng lên giá trị của z.

Các toán tử quan hệ

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

$>$ Lớn hơn

$>=$ Lớn hơn hoặc bằng

$<$ Nhỏ hơn

$<=$ Nhỏ hơn hoặc bằng

$==$ Bằng

$!=$ Không bằng

Khi toán tử quan hệ được áp dụng cho phần tử cú pháp hoặc biến đã được gán giá trị “na” (không áp dụng), giá trị “na” được xử lý như là giá trị riêng cho phần tử cú pháp hoặc biến. Giá trị “na” được xem xét không bằng bất kỳ giá trị khác.

Các toán tử theo bit

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

$\&$ “and” theo bit. Khi hoạt động trên các đối số nguyên, hoạt động trên biểu diễn bù 2 của giá trị nguyên. Khi hoạt động trên đối số nhị phân chứa ít bit

hơn đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung nhiều bit có trọng số bằng 0.

| “or” theo bit. Khi hoạt động trên các đối số nguyên, hoạt động trên biểu diễn bù 2 của giá trị nguyên. Khi hoạt động trên đối số nhị phân chứa ít bit hơn đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung nhiều bit có trọng số bằng 0.

^ “exclusive or” theo bit. Khi hoạt động trên các đối số nguyên, hoạt động trên biểu diễn bù 2 của giá trị nguyên. Khi hoạt động trên đối số nhị phân chứa ít bit hơn đối số khác, đối số ngắn hơn được mở rộng bằng cách bổ sung nhiều bit có trọng số bằng 0.

$x \gg y$ Dịch phải số học của biểu diễn số nguyên bù 2 của các số nhị phân x sang y . Hàm này được định nghĩa chỉ cho các giá trị số nguyên không âm của y . Các bit được dịch thành các bit có trọng số lớn nhất (most significant bit, MSB) làm kết quả dịch phải có giá trị bằng MSB của x trước phép toán dịch.

$x \ll y$ Dịch trái số học của biểu diễn số nguyên bù 2 của các số nhị phân x sang y . Hàm này được định nghĩa chỉ cho các giá trị số nguyên không âm của y . Các bit được dịch thành các bit có trọng số nhỏ nhất (least significant bit, LSB) làm kết quả dịch trái có giá trị bằng 0.

Các toán tử gán

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

= Toán tử gán

++ Tăng, tức là, $x++$ tương đương $x = x + 1$; khi được sử dụng trong chỉ số mảng, nâng lên giá trị của biến trước phép tăng.

-- Giảm, tức là, $x--$ tương đương $x = x - 1$; khi được sử dụng trong chỉ số mảng, nâng lên giá trị của biến trước khi phép giảm.

$+=$ Tăng bởi lượng được xác định, tức là, $x += 3$ tương đương $x = x + 3$,
và

$x += (-3)$ tương đương $x = x + (-3)$.

$-=$ Giảm bởi lượng được xác định, tức là, $x -= 3$ tương đương $x = x - 3$,
và

$x -= (-3)$ tương đương $x = x - (-3)$.

Ký pháp khoảng

Ký pháp sau được sử dụng để xác định khoảng giá trị:

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

Các hàm số học

Các hàm số học sau được định nghĩa:

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

$\text{Asin}(x)$ là hàm sin đảo lượng giác, hoạt động trên đối số x nằm trong khoảng từ $-1,0$ đến $1,0$, bao gồm hai giá trị đầu mút, có giá trị đầu ra trong khoảng từ $-\pi/2$ đến $\pi/2$, bao gồm hai giá trị đầu mút, theo các đơn vị radian

$\text{Atan}(x)$ là hàm tang ngược lượng giác, hoạt động trên đối số x , có giá trị đầu ra trong khoảng từ $-\pi/2$ đến $\pi/2$, bao gồm hai giá trị đầu mút, theo các đơn vị radian

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

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

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

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

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

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

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

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

$\text{Ln}(x)$ lôgarit tự nhiên của x (lôgarit cơ số e , trong đó e là hằng số cơ số lôgarit tự nhiên 2,718 281 828...).

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

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

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

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

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5)$$

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

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

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

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

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

Thứ tự ưu tiên phép toán

Khi thứ tự ưu tiên trong biểu thức không được chỉ báo tường minh nhờ sử dụng các dấu ngoặc, các quy tắc sau áp dụng:

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

Bảng dưới đây xác định độ ưu tiên của các phép toán từ cao nhất đến thấp nhất; vị trí cao hơn trong bảng chỉ báo độ ưu tiên cao hơn.

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

Bảng: Ưu tiên phép toán từ cao nhất (trên cùng của bảng) đến thấp nhất (dưới cùng của bảng)

Các phép toán (với các toán hạng x, y, và z)
"x++", "x--"
"!x", "-x" (làm toán tử tiền tố đơn nguyên)
x^y
"x * y", "x / y", "x ÷ y", " $\frac{x}{y}$ ", "x % y"
"x + y", "x - y" (làm toán tử hai đối số), " $\sum_{i=x}^y f(i)$ "
"x << y", "x >> y"
"x < y", "x <= y", "x > y", "x >= y"
"x == y", "x != y"
"x & y"
"x y"
"x && y"
"x y"
"x ? y : z"
"x..y"
"x = y", "x += y", "x -= y"

Mô tả bằng văn bản các phép toán logic

Theo văn bản, mệnh đề của các phép toán logic như sẽ được mô tả bằng toán học ở dạng dưới đây:

```

if(condition 0)
    statement 0
else if(condition 1)
    statement 1
...
else /* nhận xét mang thông tin ở điều kiện còn lại*/
    statement n

```

có thể được mô tả theo cách dưới đây:

...như sau / ... điều kiện sau áp dụng:

- If condition 0, statement 0

- Ngược lại, if condition 1, statement 1
- ...
- Ngược lại (nhận xét mang thông tin ở điều kiện còn lại),
statement n

Mỗi mệnh đề “Nếu ... Ngược lại, nếu ... Ngược lại, ...” bằng văn bản được nêu với “...như sau” hoặc “... điều kiện sau áp dụng” tiếp sau ngay bằng “Nếu ...”. Điều kiện cuối cùng của “Nếu ... Ngược lại, nếu ... Ngược lại, ...” luôn là “Ngược lại, ...”. Các mệnh đề đan xen “nếu ... Ngược lại, nếu... Ngược lại, ...” có thể được nhận diện bằng cách so khớp “... như sau” hoặc “... điều kiện sau áp dụng” với kết thúc “Ngược lại, ...”.

Trong văn bản, mệnh đề của các phép lô gic như sẽ được mô tả về mặt toán học ở dạng dưới đây:

```

if(condition 0a  &&  condition 0b)
    statement 0
else if(condition 1a  ||  condition 1b)
    statement 1
...
else
    statement n

```

có thể được mô tả theo cách dưới đây:

... như sau/ ... điều kiện sau áp dụng:

- Nếu tất cả các điều kiện sau là đúng, statement 0:
–condition 0a
–condition 0b
- Ngược lại, nếu một hoặc nhiều điều kiện sau là đúng,
statement 1:

- condition 1a
- condition 1b
- ...
- Ngược lại, statement n

Theo văn bản, mệnh đề của các phép toán lô gic như sẽ được mô tả về mặt toán học ở dạng dưới đây:

if(condition 0)

statement 0

if(condition 1)

statement 1

có thể được mô tả theo cách dưới đây:

When condition 0, statement 0

When condition 1, statement 1.

Fig.18 là sơ đồ khối thể hiện hệ thống cấp nội dung 3100 để thực hiện dịch vụ phân tán nội dung. Hệ thống cấp nội dung 3100 này bao gồm thiết bị chụp 3102, thiết bị đầu cuối 3106, và một cách tùy chọn bao gồm bộ phận hiển thị 3126. Thiết bị chụp 3102 truyền thông với thiết bị đầu cuối 3106 trên đường truyền thông 3104. Đường truyền thông có thể bao gồm kênh truyền thông 13 được mô tả trên đây. Đường truyền thông 3104 bao gồm mà không bị giới hạn ở WIFI, Ethernet, cáp, không dây (3G/4G/5G), USB, hoặc loại bất kỳ của kết hợp của nó, hoặc tương tự.

Thiết bị chụp 3102 tạo dữ liệu, và có thể mã hóa dữ liệu bằng phương pháp mã hóa như được thể hiện trên các phương án thực hiện nêu trên. Theo cách khác, thiết bị chụp 3102 có thể phân tán dữ liệu đến máy chủ streaming (không được thể hiện trên các hình vẽ), và máy chủ mã hóa dữ liệu và truyền dữ liệu được mã hóa đến thiết bị đầu cuối 3106. Thiết bị chụp 3102 bao gồm mà không bị giới

hạn ở camera, điện thoại thông minh hoặc Pad, máy tính hoặc máy tính xách tay, hệ thống hội nghị video, PDA, thiết bị lắp trong xe, hoặc kết hợp của một trong các thiết bị đó, hoặc tương tự. Chẳng hạn, thiết bị chụp 3102 có thể bao gồm thiết bị nguồn 12 như được mô tả trên đây. Khi dữ liệu bao gồm video, bộ mã hóa video 20 được bao gồm trong thiết bị chụp 3102 có thể thực sự thực hiện xử lý mã hóa video. Khi dữ liệu bao gồm audio (tức là, voice), bộ mã hóa audio được bao gồm trong thiết bị chụp 3102 có thể thực sự thực hiện xử lý mã hóa audio. Đối với một số kịch bản thực tế, thiết bị chụp 3102 phân tán dữ liệu video và audio được mã hóa bằng cách ghép kênh chúng với nhau. Đối với các kịch bản thực tế khác, chẳng hạn trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được ghép kênh. Thiết bị chụp 3102 phân tán riêng rẽ dữ liệu audio được mã hóa và dữ liệu video được mã hóa đến thiết bị đầu cuối 3106.

Trong hệ thống cấp nội dung 3100, thiết bị đầu cuối 310 nhận và tái tạo dữ liệu được mã hóa. Thiết bị đầu cuối 3106 có thể là thiết bị có khả năng nhận và khôi phục dữ liệu, chẳng hạn điện thoại thông minh hoặc Pad 3108, máy tính hoặc máy tính xách tay 3110, NVR/DVR 3112, TV 3114, STB 3116, hệ thống hội nghị video 3118, hệ thống giám sát video 3120, PDA 3122, thiết bị lắp trong xe 3124, hoặc kết hợp của một trong các thiết bị đó, hoặc tương tự có thể giải mã dữ liệu được mã hóa nêu trên. Chẳng hạn, thiết bị đầu cuối 3106 có thể bao gồm thiết bị đích 14 như được mô tả trên đây. Khi dữ liệu được mã hóa bao gồm video, bộ giải mã video 30 được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện giải mã video. Khi dữ liệu được mã hóa bao gồm audio, bộ giải mã audio được bao gồm trong thiết bị đầu cuối được ưu tiên để thực hiện quá trình giải mã audio.

Đối với thiết bị đầu cuối có bộ phận hiển thị, chẳng hạn, điện thoại thông minh hoặc Pad 3108, máy tính hoặc máy tính xách tay 3110, NVR/DVR 3112, TV 3114, PDA 3122, hoặc thiết bị lắp trong xe 3124, thiết bị đầu cuối có thể cấp dữ liệu được giải mã cho bộ phận hiển thị. Đối với thiết bị đầu cuối không có bộ phận hiển thị, chẳng hạn, STB 3116, hệ thống hội nghị video 3118, hoặc hệ thống giám sát video 3120, bộ phận hiển thị ngoài 3126 được tiếp xúc ở đó để nhận và thể hiện dữ liệu được giải mã.

Khi mỗi thiết bị trong hệ thống này thực hiện mã hóa hoặc giải mã, thiết bị mã hóa ảnh hoặc thiết bị giải mã ảnh, như được thể hiện trên các phương án thực hiện nêu trên, có thể được sử dụng.

Fig.19 là sơ đồ thể hiện cấu trúc của ví dụ của thiết bị đầu cuối 3106. Sau khi thiết bị đầu cuối 3106 nhận stream từ thiết bị chụp 3102, khối xử lý giao thức 3202 phân tích giao thức truyền của stream. Giao thức bao gồm mà không bị giới hạn ở RTSP, HTTP, HLS, MPEG-DASH, RTP, RTMP, hoặc loại bất kỳ của kết hợp của nó, hoặc tương tự.

Sau khi khối xử lý giao thức 3202 xử lý stream, tệp tin stream được tạo. Tệp tin này được xuất ra khỏi phân kênh 3204. Khối phân kênh 3204 có thể tách riêng dữ liệu được ghép kênh thành dữ liệu audio được mã hóa và dữ liệu video được mã hóa. Như được mô tả trên đây, đối với một số kịch bản thực tế, chẳng hạn trong hệ thống hội nghị video, dữ liệu audio được mã hóa và dữ liệu video được mã hóa không được ghép kênh. Trong tình huống này, dữ liệu được mã hóa được truyền đến bộ giải mã video 3206 và bộ giải mã audio 3208 mà không qua khối phân kênh 3204.

Qua quá trình phân kênh, ES video, ES audio, và một cách tùy chọn phụ đề được tạo. Bộ giải mã video 3206, bao gồm bộ giải mã video 30 như được giải thích theo các phương án thực hiện nêu trên, giải mã ES video bằng phương pháp

giải mã như được thể hiện theo các phương án thực hiện nêu trên để tạo khung video, và cấp dữ liệu này cho khối đồng bộ 3212. Bộ giải mã audio 3208, giải mã ES audio để tạo khung audio, và cấp dữ liệu này cho khối đồng bộ 3212. Theo cách khác, khung video có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.19) trước khi cấp nó cho khối đồng bộ 3212. Tương tự, khung audio có thể lưu trữ trong bộ đệm (không được thể hiện trên Fig.19) trước khi cấp nó cho khối đồng bộ 3212.

Khối đồng bộ 3212 đồng bộ khung video và khung audio, và cấp video/audio cho bộ phận hiển thị video/audio 3214. Chẳng hạn, khối đồng bộ 3212 đồng bộ hiển thị thông tin video và audio. Thông tin có thể mã hóa trong cú pháp nhờ sử dụng các nhãn thời gian liên quan đến hiển thị dữ liệu audio và hình ảnh được mã hóa và các nhãn thời gian liên quan đến chính việc truyền dòng dữ liệu.

Nếu phụ đề được bao gồm trong dòng, bộ giải mã phụ đề 3210 giải mã phụ đề, và đồng bộ nó với khung video và khung audio, và cấp video/audio/phụ đề cho bộ phận hiển thị video/audio/phụ đề 3216.

Sáng chế không bị giới hạn ở hệ thống nêu trên, và thiết bị mã hóa ảnh hoặc thiết bị giải mã ảnh theo các phương án thực hiện nêu trên có thể được đưa vào hệ thống khác, chẳng hạn, hệ thống xe hơi.

Mặc dù các phương án thực hiện sáng chế đã được mô tả chủ yếu dựa trên tạo mã video, cần lưu ý rằng theo các phương án thực hiện, hệ thống tạo mã 10, bộ mã hóa 20 và bộ giải mã 30 (và một cách tương ứng là hệ thống 10) và các phương án thực hiện khác được mô tả ở đây có thể cũng được tạo cấu hình để xử lý hoặc tạo mã ảnh tĩnh, chẳng hạn, bước xử lý hoặc tạo mã ảnh cá nhân độc lập với ảnh đứng trước hoặc đứng sau bất kỳ như khi tạo mã video. Nói chung, chỉ các khối dự báo ngoài 244 (bộ mã hóa) và 344 (bộ giải mã) có thể không khả dụng trong trường hợp tạo mã xử lý ảnh bị giới hạn ở một ảnh 17. Tất cả các

chức năng khác (cũng được gọi là công cụ hoặc công nghệ) của bộ mã hóa video 20 và bộ giải mã video 30 có thể được sử dụng cân bằng để xử lý ảnh tĩnh, chẳng hạn, tính toán phần dư 204/304, biến đổi 206, lượng tử 208, lượng tử ngược 210/310, biến đổi (ngược) 212/312, phân vùng 262/362, dự báo trong 254/354, và/hoặc lọc vòng 220, 320, và tạo mã entropy 270 và giải mã entropy 304.

Theo các phương án thực hiện, chẳng hạn, bộ mã hóa 20 và bộ giải mã 30, và các chức năng được mô tả ở đây, chẳng hạn, dựa vào bộ mã hóa 20 và bộ giải mã 30, có thể được thực hiện trong phần cứng, phần mềm, firmware, hoặc kết hợp bất kỳ của nó. Nếu được thực hiện trong phần mềm, các chức năng có thể được lưu trữ trên phương tiện máy tính đọc được hoặc được truyền trên phương tiện truyền thông dưới dạng một hoặc nhiều lệnh hoặc mã và được thực thi bằng khối xử lý dựa trên phần cứng. Phương tiện máy tính đọc được có thể bao gồm các phương tiện lưu trữ máy tính đọc được, tương ứng với phương tiện hữu hình, chẳng hạn, các phương tiện lưu trữ dữ liệu, hoặc phương tiện truyền thông bao gồm bất kỳ phương tiện để truyền chương trình máy tính từ một điểm sang điểm khác, chẳng hạn, theo giao thức truyền thông. Theo cách này, phương tiện máy tính đọc được nói chung có thể tương ứng với (1) các phương tiện lưu trữ máy tính đọc được hữu hình vốn bất biến hoặc (2) phương tiện truyền thông, chẳng hạn, tín hiệu hoặc sóng mang. Phương tiện lưu trữ dữ liệu có thể là phương tiện khả dụng bất kỳ có thể được truy nhập bởi một hoặc nhiều máy tính hoặc một hoặc nhiều bộ xử lý để truy xuất các lệnh, mã và/hoặc các cấu trúc dữ liệu để thực hiện các kỹ thuật được mô tả theo sáng chế. Sản phẩm chương trình máy tính có thể bao gồm phương tiện máy tính đọc được.

Theo ví dụ, và không giới hạn, các phương tiện lưu trữ máy tính đọc được này có thể bao gồm RAM, ROM, EEPROM, CD-ROM hoặc bộ lưu trữ đĩa quang khác, bộ lưu trữ đĩa từ, hoặc các thiết bị lưu trữ từ tính khác, bộ nhớ nhanh, hoặc

phương tiện khác bất kỳ có thể được sử dụng để lưu trữ mã chương trình mong muốn ở dạng lệnh hoặc cấu trúc dữ liệu và có thể được truy nhập bằng máy tính. Bất kỳ kết nối nào cũng được gọi đúng là phương tiện máy tính đọc được. Chẳng hạn, nếu các lệnh được truyền từ website, máy chủ, hoặc nguồn từ xa khác nhờ sử dụng cáp đồng trục, cáp sợi quang, cáp xoắn, DSL, hoặc các công nghệ không dây chẳng hạn hồng ngoại, vô tuyến, và vi sóng, sau đó là cáp đồng trục, cáp sợi quang, cáp xoắn, DSL, hoặc các công nghệ không dây, chẳng hạn, hồng ngoại, vô tuyến, và vi sóng được bao gồm trong định nghĩa về phương tiện. Tuy nhiên, cần hiểu rằng phương tiện lưu trữ máy tính đọc được và phương tiện lưu trữ dữ liệu không bao gồm các kết nối, sóng kênh mang, tín hiệu, hoặc phương tiện khả biến khác, nhưng thay vào đó được dành cho phương tiện lưu trữ hữu hình bất biến. Disk và disc, như được sử dụng ở đây, bao gồm đĩa nén (compact disc, CD), đĩa laze, đĩa quang, đĩa đa dụng số (digital versatile disc, DVD), đĩa mềm floppy và đĩa Blu-ray, trong đó disk thường tái tạo dữ liệu dạng từ tính, trong khi disc tái tạo dữ liệu dạng quang bằng tia laze. Các kết hợp của các thiết bị nêu trên cũng cần được bao gồm trong phạm vi của phương tiện máy tính đọc được.

Các lệnh có thể được thực thi bằng một hoặc nhiều bộ xử lý, chẳng hạn một hoặc nhiều DSP, bộ vi xử lý đa năng, ASIC, FPGA, hoặc hệ mạch tích hợp hoặc logic rời rạc tương đương khác. Tương ứng, cụm từ “bộ xử lý,” như được sử dụng ở đây có thể đề cập đến bất kỳ cấu trúc trong cấu trúc nêu trên hoặc cấu trúc khác bất kỳ thích hợp để thực hiện các kỹ thuật được mô tả ở đây. Ngoài ra, theo một số khía cạnh, chức năng được mô tả ở đây có thể được đặt trong phần cứng và/hoặc môđun phần mềm dành riêng được tạo cấu hình để mã hóa và giải mã, hoặc được đưa vào codec kết hợp. Các kỹ thuật này cũng có thể được thực hiện trong một hoặc nhiều mạch hoặc phần tử logic.

Các kỹ thuật của sáng chế có thể được thực hiện trong các thiết bị hoặc bộ phận khác nhau, bao gồm, điện thoại cầm tay không dây, mạch tích hợp (integrated circuit, IC) hoặc tập hợp IC (chẳng hạn, chip set). Các thành phần, môđun, hoặc các đơn vị khác nhau được mô tả theo sáng chế để nhấn mạnh các khía cạnh chức năng của các thiết bị được tạo cấu hình để thực hiện các kỹ thuật được bộc lộ, nhưng không nhất thiết yêu cầu thực hiện bằng các đơn vị phần cứng khác nhau. Thay vào đó, như được mô tả trên đây, các khối khác nhau có thể được kết hợp trong khối phần cứng codec hoặc được tạo bằng tập hợp các khối phần cứng liên tác, bao gồm một hoặc nhiều bộ xử lý như được mô tả trên đây, kết hợp với phần mềm và/hoặc firmware thích hợp.

YÊU CẦU BẢO HỘ

1. Phương pháp tạo mã được triển khai bởi thiết bị giải mã bao gồm các bước:

thu được giá trị chỉ số chế độ tách đối với khối tạo mã hiện tại;

thu được giá trị chỉ số góc angleIdx đối với khối tạo mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc angleIdx dựa trên giá trị chỉ số chế độ tách;

thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; và

giải mã khối tạo mã hiện tại theo giá trị chỉ số partIdx.

2. Phương pháp theo điểm 1, trong đó giá trị chỉ số partIdx bằng 1 nếu giá trị chỉ số góc angleIdx lớn hơn hoặc bằng ngưỡng thứ nhất và giá trị chỉ số góc angleIdx nhỏ hơn hoặc bằng ngưỡng thứ hai, ngược lại giá trị chỉ số partIdx bằng 0, trong đó các ngưỡng thứ nhất và thứ hai là các giá trị số nguyên và ngưỡng thứ nhất nhỏ hơn ngưỡng thứ hai.

3. Phương pháp theo điểm 2, trong đó ngưỡng thứ nhất bằng 13 và ngưỡng thứ hai bằng 27.

4. Phương pháp theo điểm 1, trong đó giải mã khối tạo mã hiện tại bao gồm bước lưu trữ thông tin chuyển động cho khối tạo mã hiện tại theo giá trị chỉ số partIdx.

5. Phương pháp theo điểm 1, trong đó giá trị chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng cho khối tạo mã hiện tại.

6. Phương pháp theo điểm 1, trong đó giá trị chỉ số góc angleIdx được sử dụng để phân vùng hình học khối tạo mã hiện tại.

7. Bộ giải mã video bao gồm:

khối phân tích được tạo cấu hình để thu được giá trị chỉ số chế độ tách cho khối tạo mã hiện tại;

khối nhận giá trị chỉ số góc được tạo cấu hình để thu được giá trị chỉ số góc angleIdx cho khối tạo mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác

định giá trị chỉ số góc `angleIdx` dựa trên giá trị chỉ số chế độ tách;

khối thiết lập được tạo cấu hình để thiết lập giá trị chỉ số `partIdx` theo giá trị chỉ số góc `angleIdx`; và

khối xử lý được tạo cấu hình để giải mã khối tạo mã hiện tại theo giá trị chỉ số `partIdx`.

8. Bộ giải mã video theo điểm 7, trong đó giá trị chỉ số `partIdx` bằng 1 nếu giá trị chỉ số góc `angleIdx` lớn hơn hoặc bằng ngưỡng thứ nhất và giá trị chỉ số góc `angleIdx` nhỏ hơn hoặc bằng ngưỡng thứ hai, ngược lại giá trị chỉ số `partIdx` bằng 0, trong đó các ngưỡng thứ nhất và thứ hai là các giá trị số nguyên và ngưỡng thứ nhất nhỏ hơn ngưỡng thứ hai.

9. Bộ giải mã video theo điểm 8, trong đó ngưỡng thứ nhất bằng 13 và ngưỡng thứ hai bằng 27.

10. Bộ giải mã video theo điểm 7, trong đó khối xử lý được tạo cấu hình để lưu trữ thông tin chuyển động cho khối tạo mã hiện tại theo giá trị chỉ số `partIdx`.

11. Bộ giải mã video theo điểm 7, trong đó giá trị chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng cho khối tạo mã hiện tại.

12. Bộ giải mã video theo điểm 7, trong đó giá trị chỉ số góc `angleIdx` được sử dụng để phân vùng hình học khối tạo mã hiện tại.

13. Vật ghi máy tính đọc được đã lưu trữ các lệnh mà, khi được thực thi bởi một hoặc nhiều bộ xử lý, khiến một hoặc nhiều bộ xử lý thực hiện các hoạt động bao gồm:

thu được giá trị chỉ số chế độ tách cho khối tạo mã hiện tại;

thu được giá trị chỉ số góc `angleIdx` cho khối tạo mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc `angleIdx` dựa trên giá trị chỉ số chế độ tách;

thiết lập giá trị chỉ số partIdx theo giá trị chỉ số góc angleIdx; và

giải mã khối tạo mã hiện tại theo giá trị chỉ số partIdx.

14. Vật ghi máy tính đọc được bất biến theo điểm 13, trong đó giá trị chỉ số partIdx bằng 1 nếu giá trị chỉ số góc angleIdx lớn hơn hoặc bằng ngưỡng thứ nhất và giá trị chỉ số góc angleIdx nhỏ hơn hoặc bằng ngưỡng thứ hai, ngược lại giá trị chỉ số partIdx bằng 0, trong đó các ngưỡng thứ nhất và thứ hai là các giá trị số nguyên và ngưỡng thứ nhất nhỏ hơn ngưỡng thứ hai.

15. Vật ghi máy tính đọc được bất biến theo điểm 14, trong đó ngưỡng thứ nhất bằng 13 và ngưỡng thứ hai bằng 27.

16. Vật ghi máy tính đọc được bất biến theo điểm 13, trong đó giải mã khối tạo mã hiện tại bao gồm storing thông tin chuyển động cho khối tạo mã hiện tại theo giá trị chỉ số partIdx.

17. Vật ghi máy tính đọc được bất biến theo điểm 13, trong đó giá trị chỉ số chế độ tách được sử dụng để chỉ báo chế độ phân vùng hình học nào được sử dụng cho khối tạo mã hiện tại.

18. Vật ghi máy tính đọc được bất biến theo điểm 13, trong đó giá trị chỉ số góc angleIdx được sử dụng để phân vùng hình học khối tạo mã hiện tại.

19. Bộ giải mã bao gồm:

một hoặc nhiều bộ xử lý; và

vật ghi máy tính đọc được bất biến được ghép nối với một hoặc nhiều bộ xử lý và lưu trữ các lệnh, mà khi được thực thi bằng một hoặc nhiều bộ xử lý, tạo cấu hình bộ giải mã để:

thu được giá trị chỉ số chế độ tách đối với khối tạo mã hiện tại;

thu được giá trị chỉ số góc angleIdx cho khối tạo mã hiện tại theo giá trị chỉ số chế độ tách và bảng mà xác định giá trị chỉ số góc angleIdx dựa trên giá trị chỉ số chế độ tách;

thiết lập giá trị chỉ số `partIdx` theo giá trị chỉ số góc `angleIdx`; và

giải mã khối tạo mã hiện tại theo giá trị chỉ số `partIdx`.

20. Bộ giải mã theo điểm 19, trong đó giá trị chỉ số `partIdx` bằng bằng 1 nếu giá trị chỉ số góc `angleIdx` lớn hơn hoặc bằng ngưỡng thứ nhất và giá trị chỉ số góc `angleIdx` nhỏ hơn hoặc bằng ngưỡng thứ hai, ngược lại giá trị chỉ số `partIdx` bằng 0, trong đó các ngưỡng thứ nhất và thứ hai là các giá trị số nguyên và ngưỡng thứ nhất nhỏ hơn ngưỡng thứ hai.

1/25

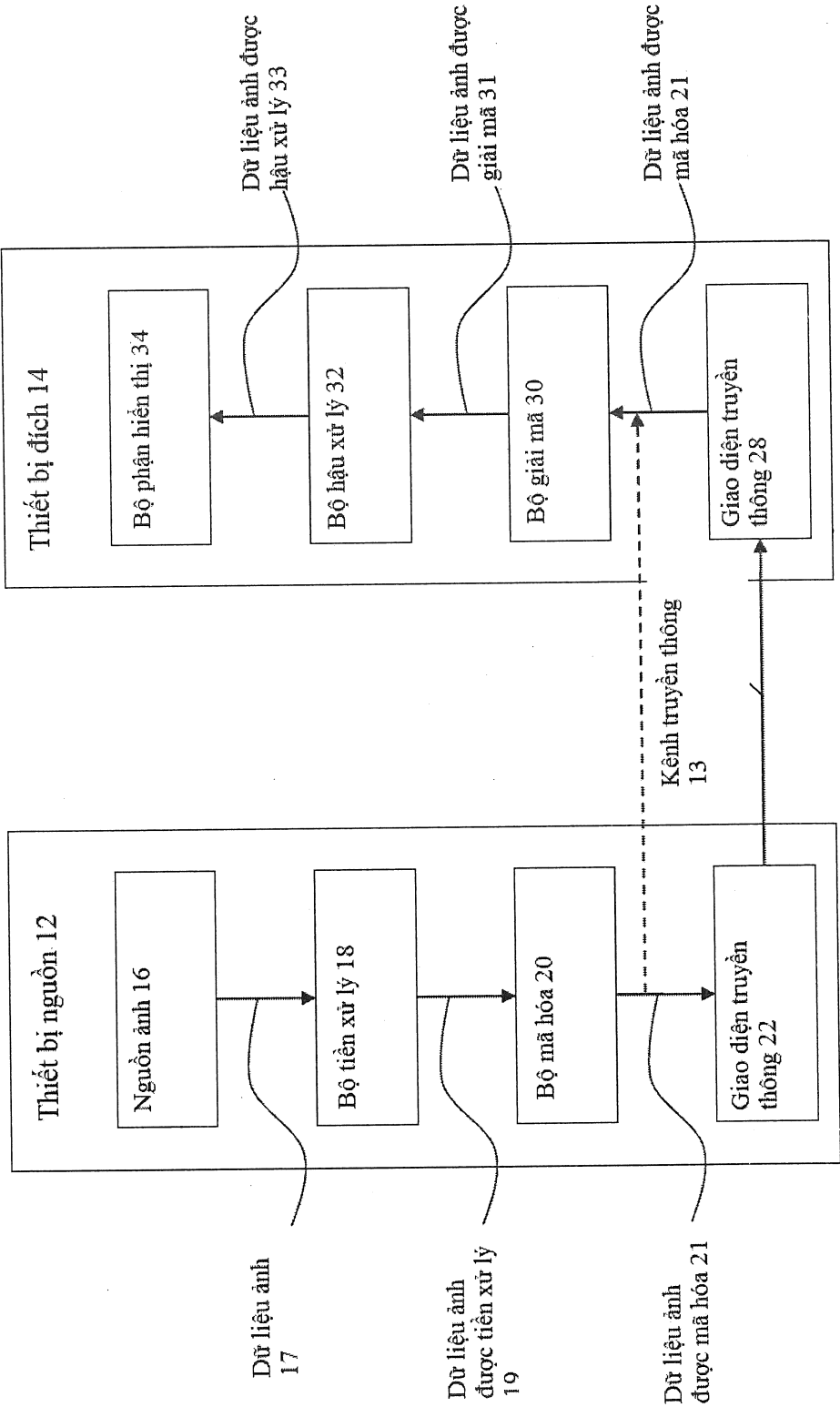


Fig.1A

2/25

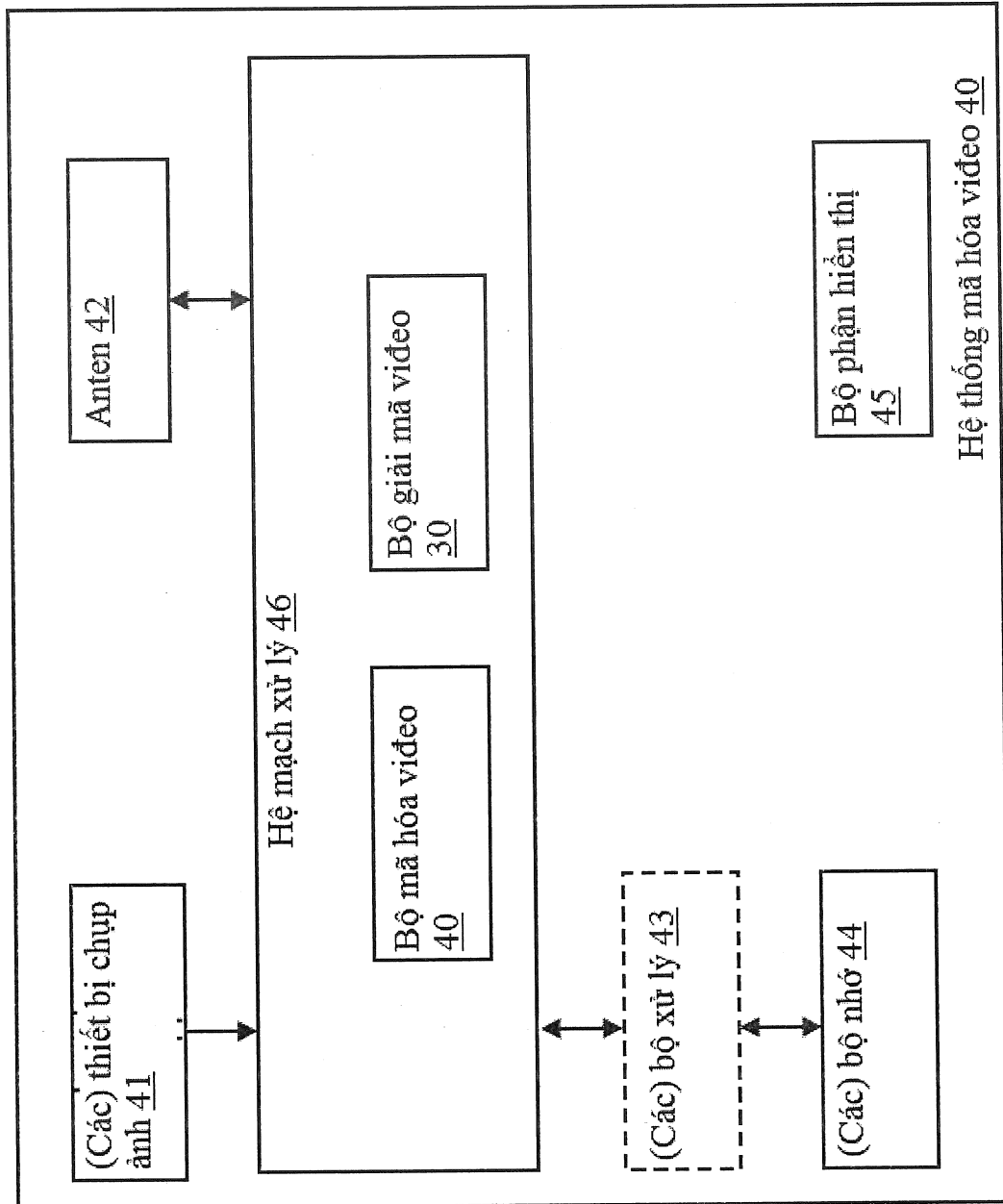


Fig. 1B

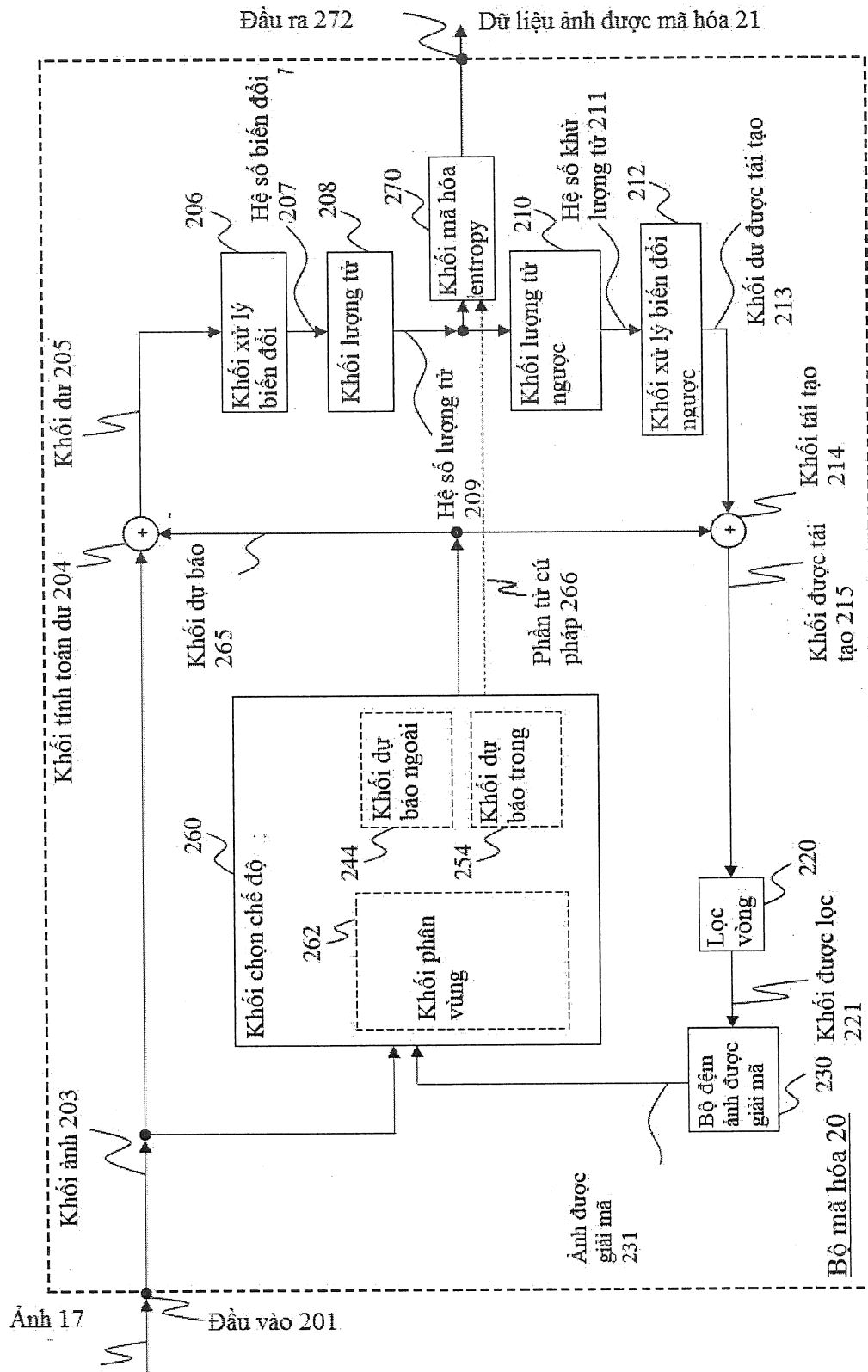


Fig. 2

4/25

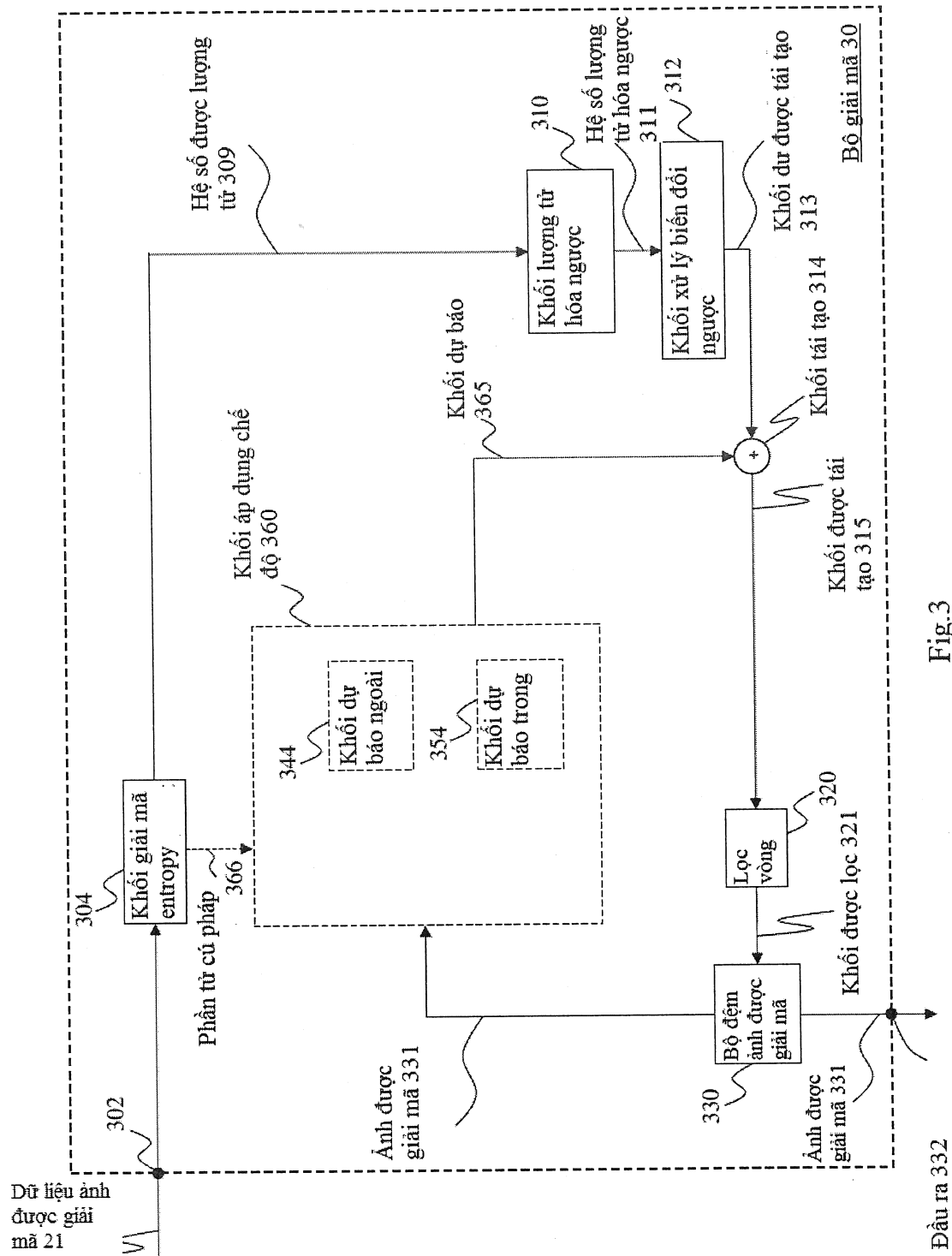


Fig.3

5/25

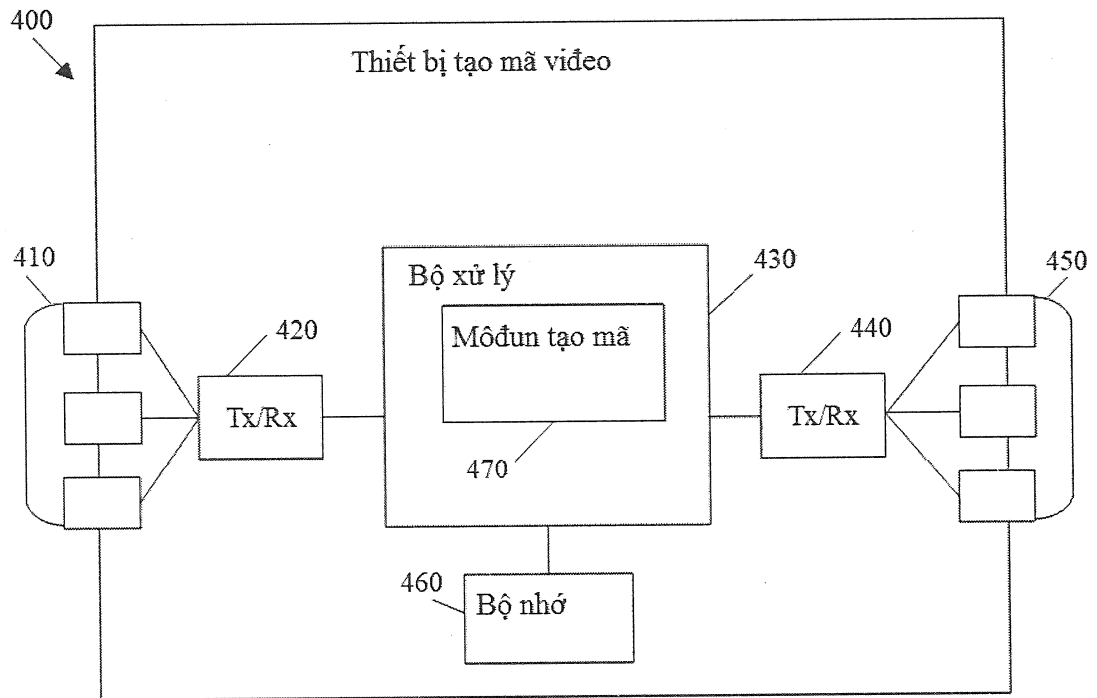


Fig.4

6/25

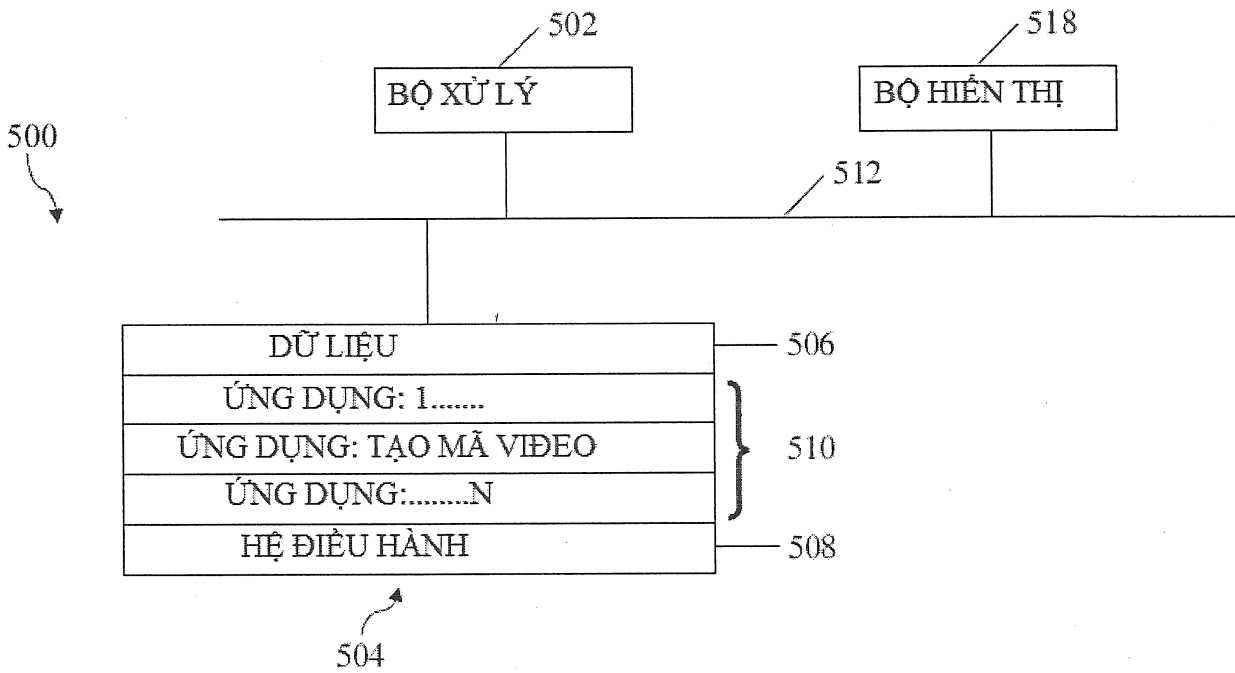


Fig.5

7/25

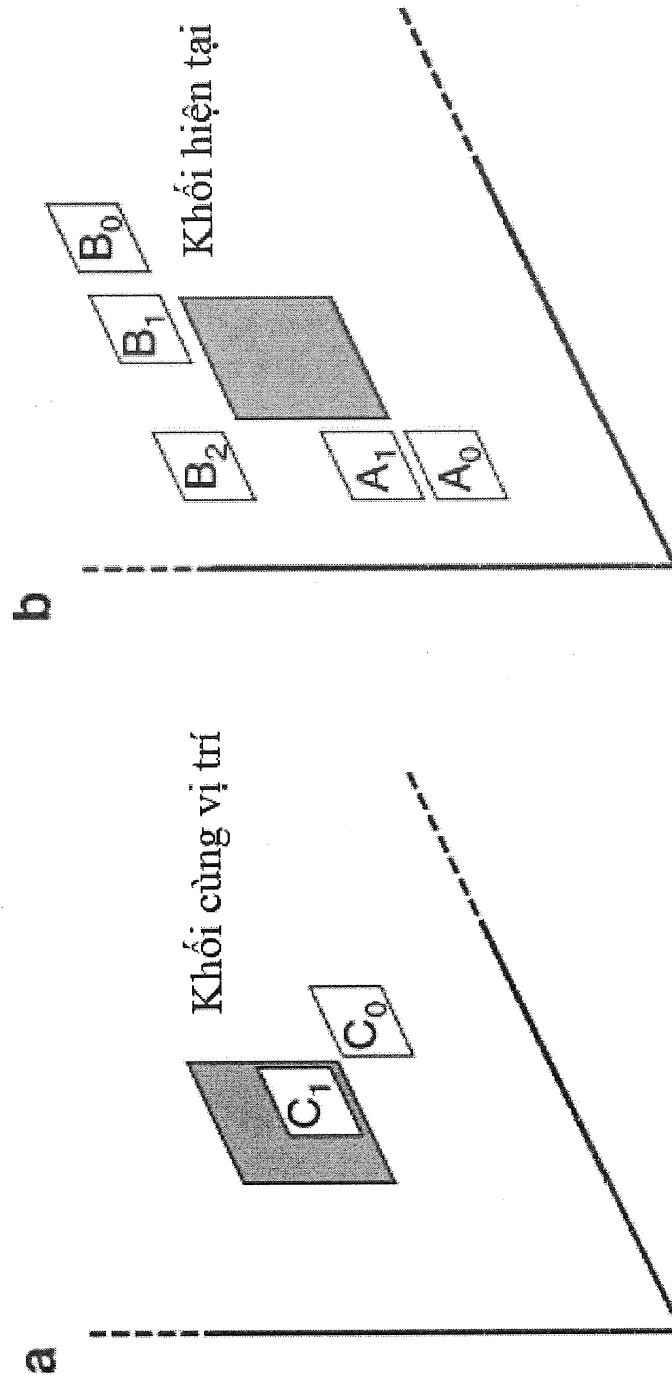
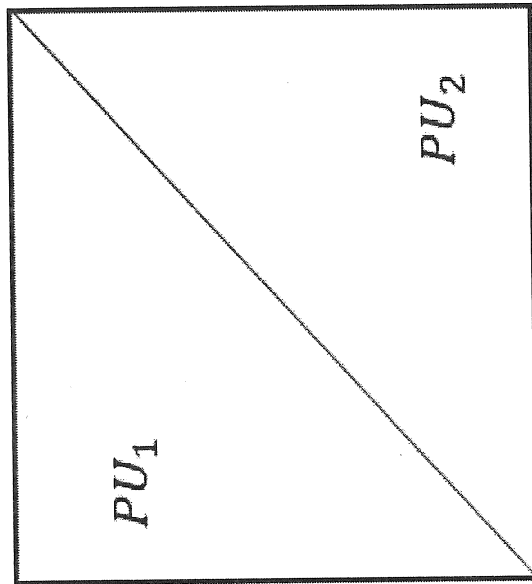


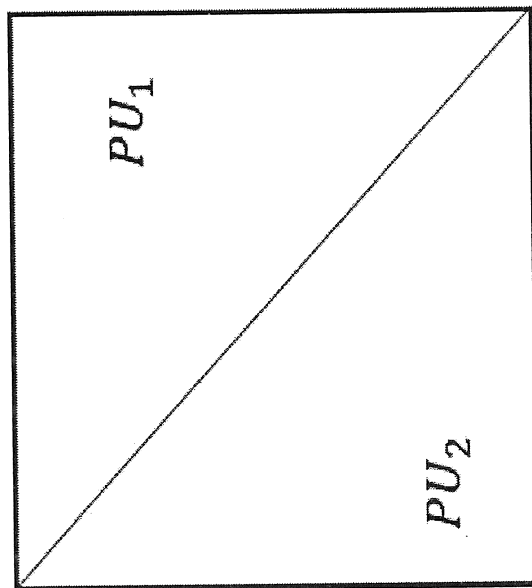
Fig.6b

Fig.6a

8/25



Tách từ góc trên bên phải
đến góc dưới bên trái



Tách từ góc trên bên trái đến
góc dưới bên phải

Fig.7

9/25

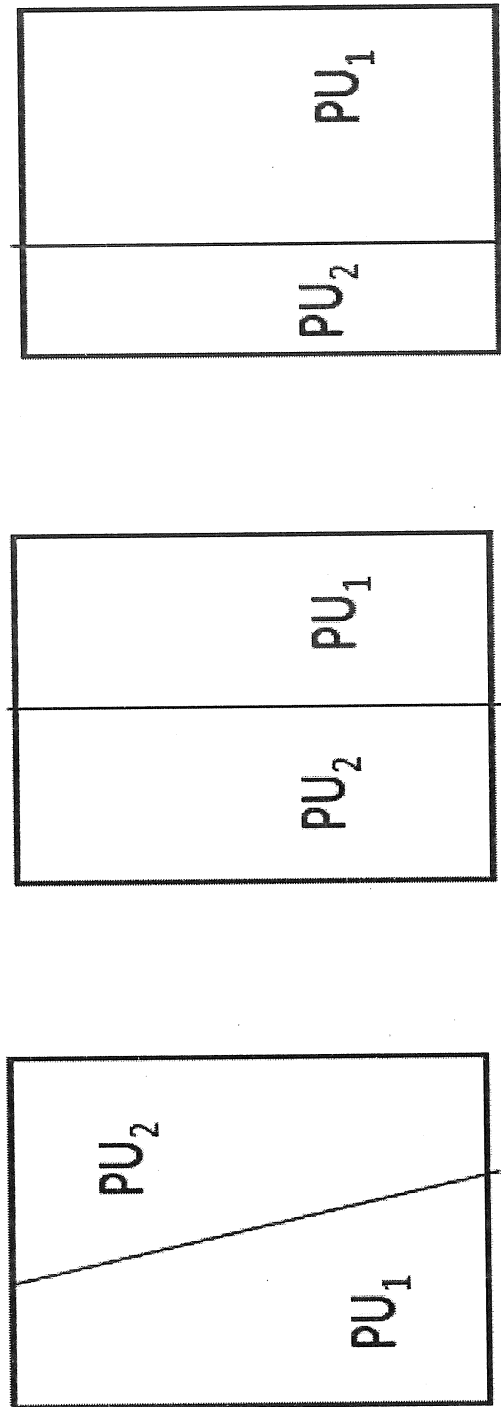


Fig.8

10/25

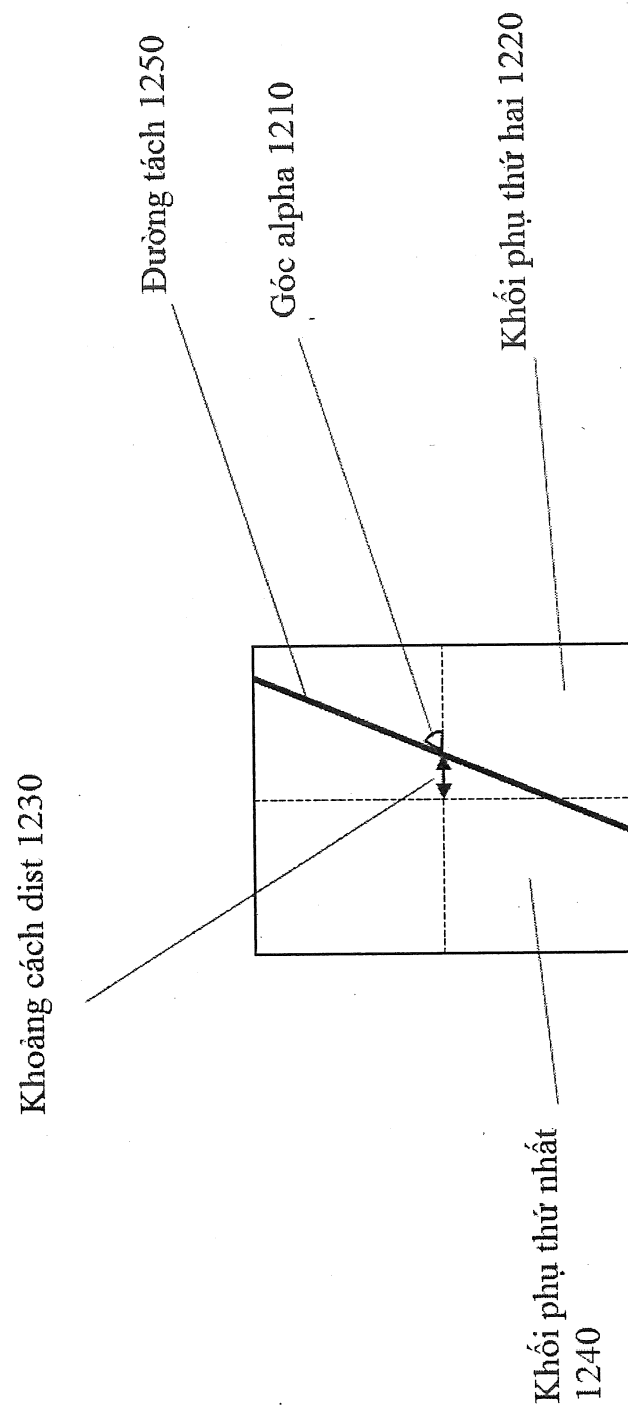


Fig.9

11/25

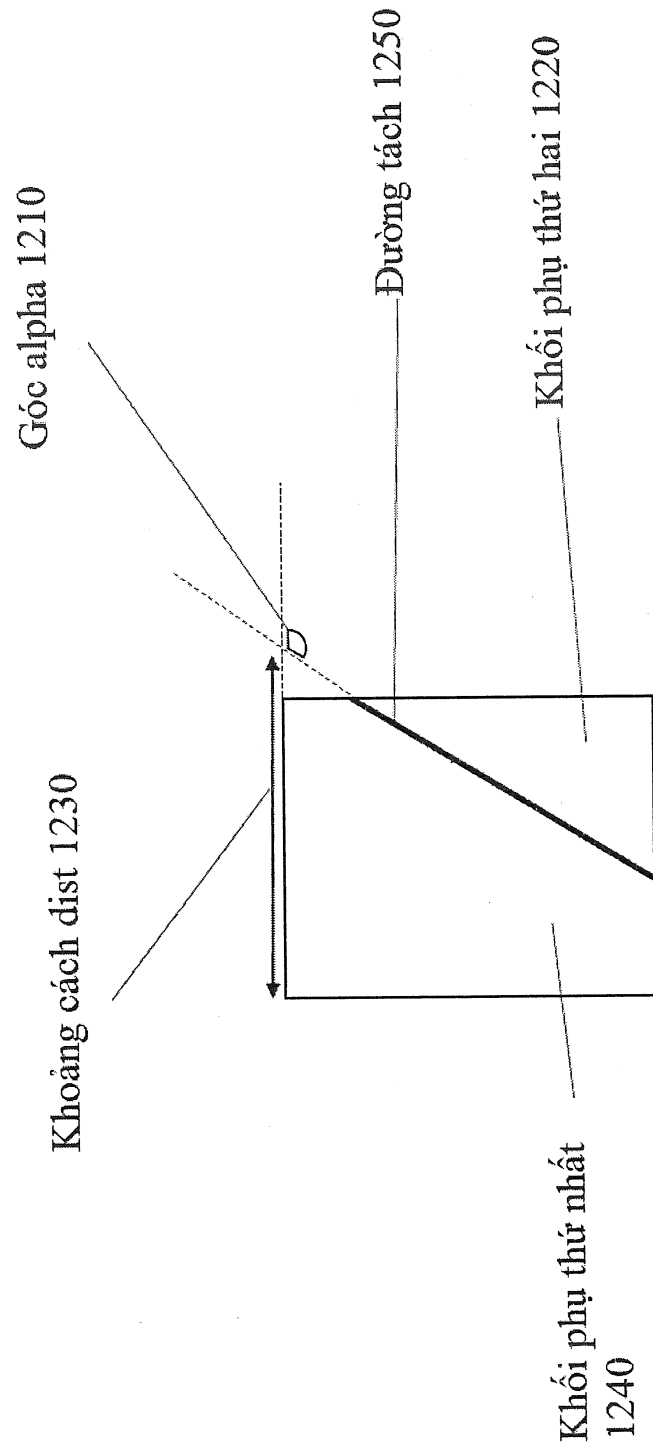


Fig.10

12/25

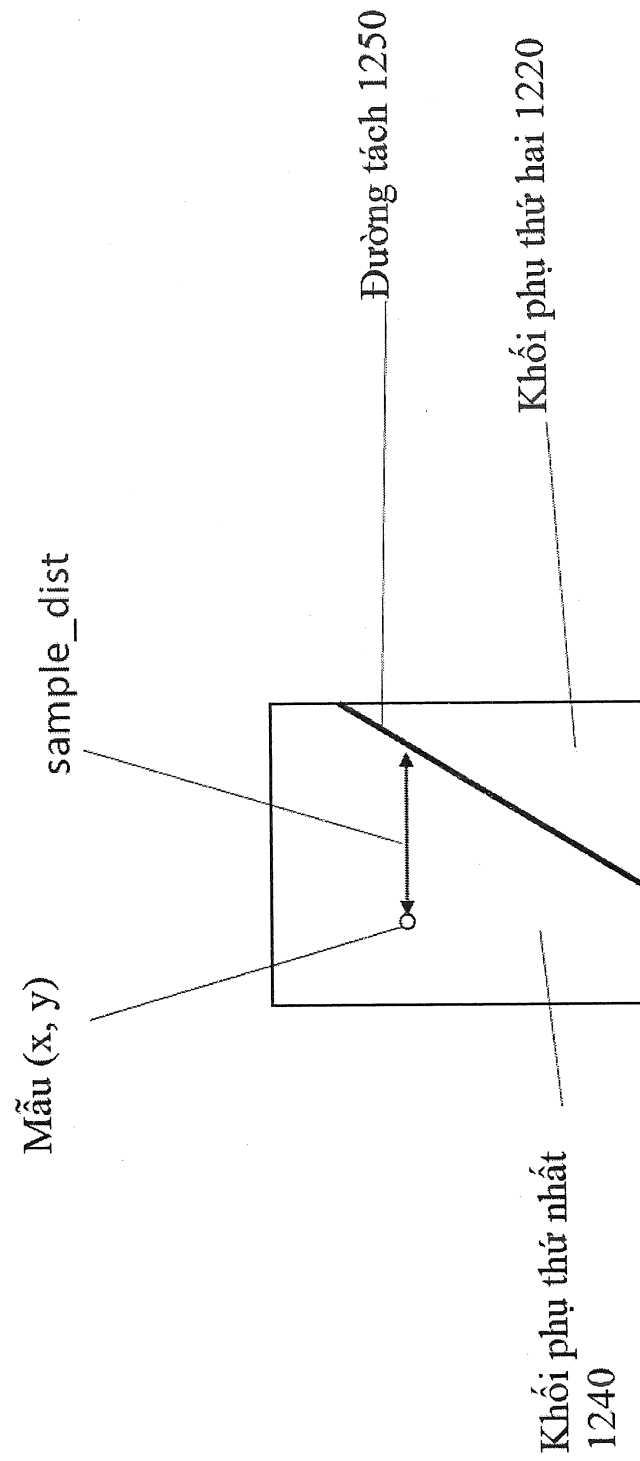


Fig.11

13/25

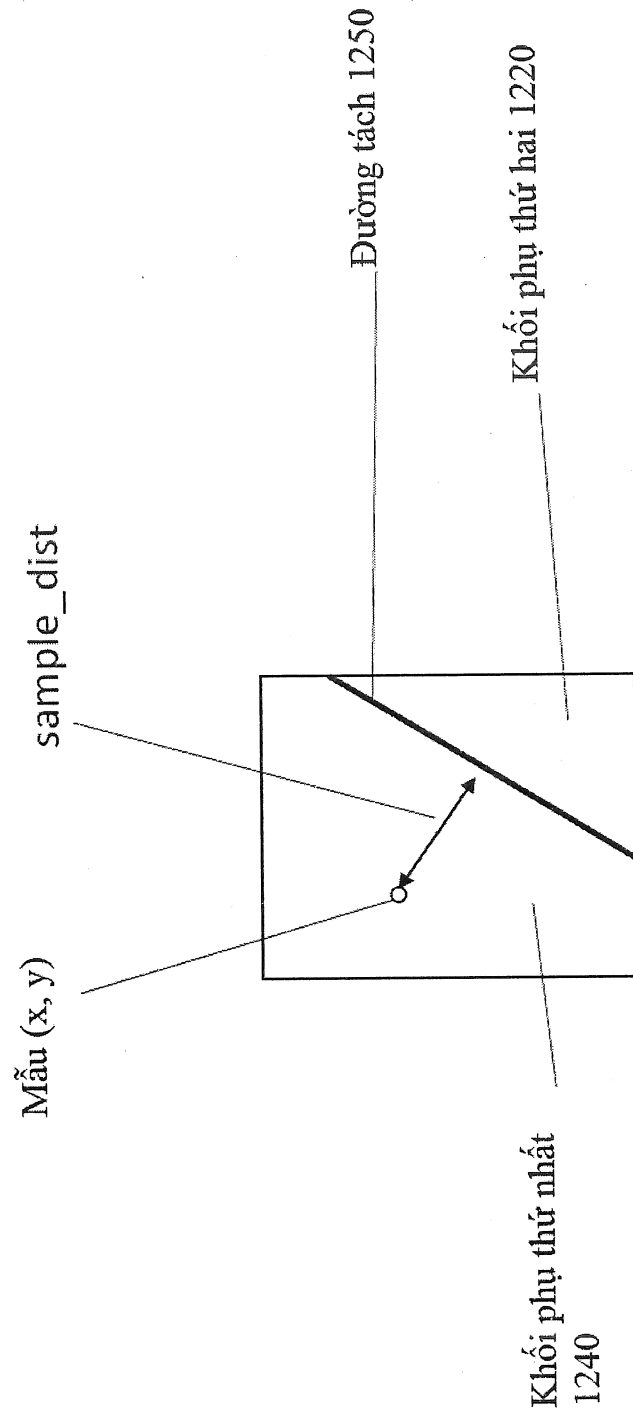


Fig.12

14/25

whRatio	alphaN	stepD
3	0	0
3	1	158
3	2	306
3	3	438
3	4	549
3	5	636
3	6	694
3	7	721
3	8	717
4	0	0
4	1	318
4	2	619
4	3	893
4	4	1129
4	5	1317
4	6	1450
4	7	1524
4	8	1536

whRatio	alphaN	stepD
0	0	0
0	1	18
0	2	31
0	3	40
0	4	42
0	5	40
0	6	31
0	7	18
0	8	0
1	0	0
1	1	38
1	2	71
1	3	97
1	4	115
1	5	125
1	6	126
1	7	118
1	8	102
2	0	0
2	1	78
2	2	149
2	3	210
2	4	260
2	5	295
2	6	315
2	7	319
2	8	307

Fig.13

15/25

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
f[idx]	64	63	59	53	45	36	24	12	0	-12	-24	-36	-45	-53	-59	-63
idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
f[idx]	-64	-63	-59	-53	-45	-36	-24	-12	0	12	24	36	45	53	59	63

Fig.14

16/25

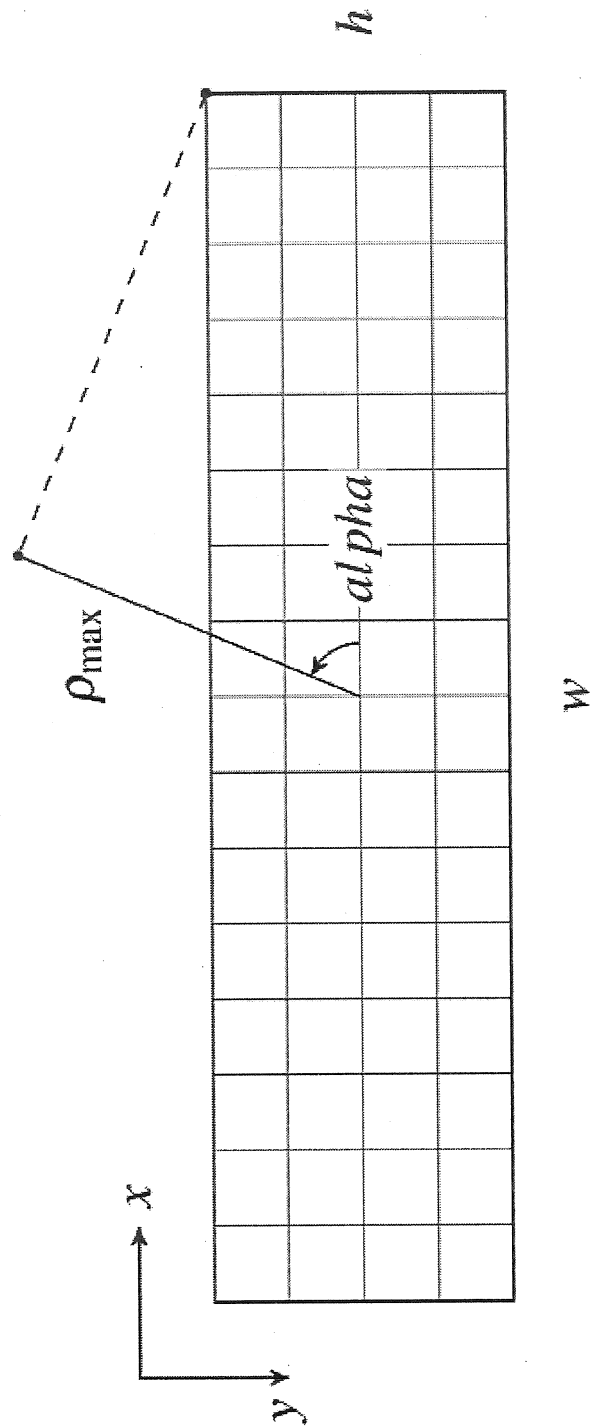


Fig.15

17/25

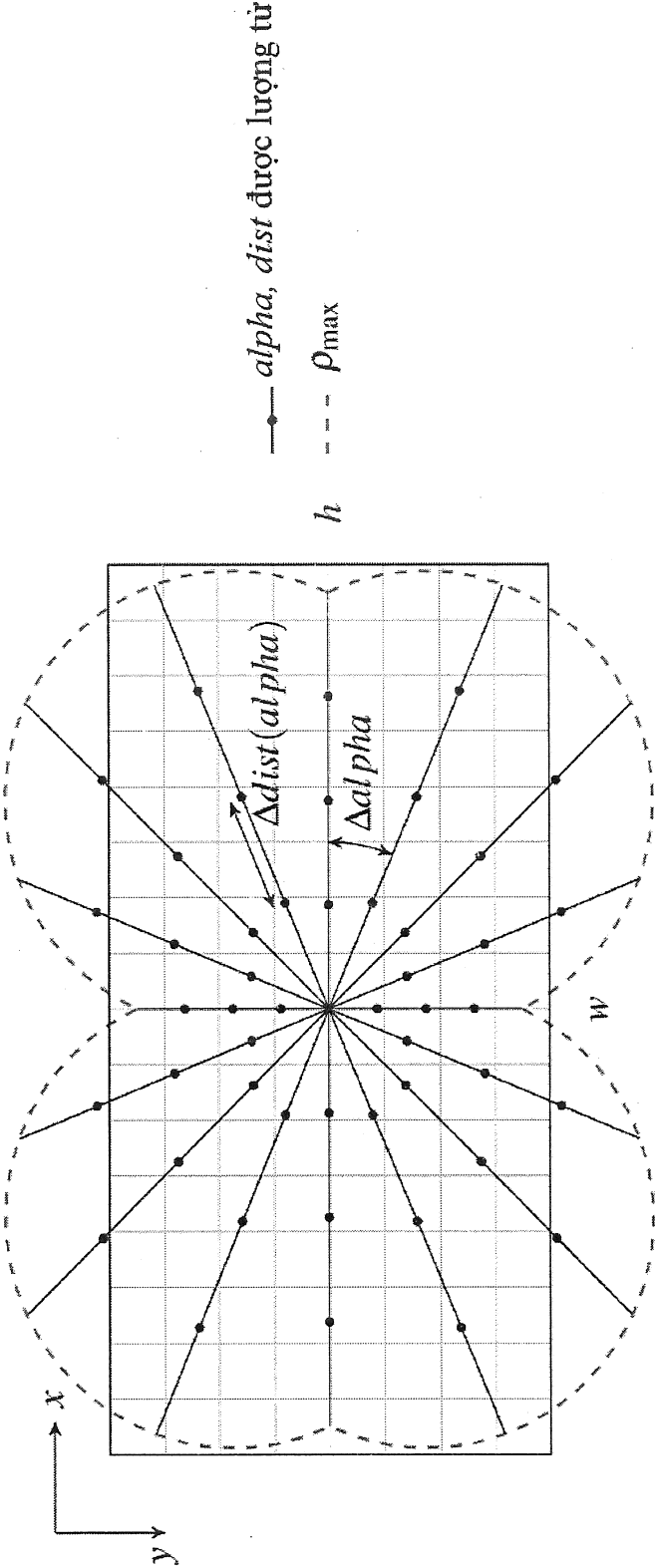


Fig.16

18/25

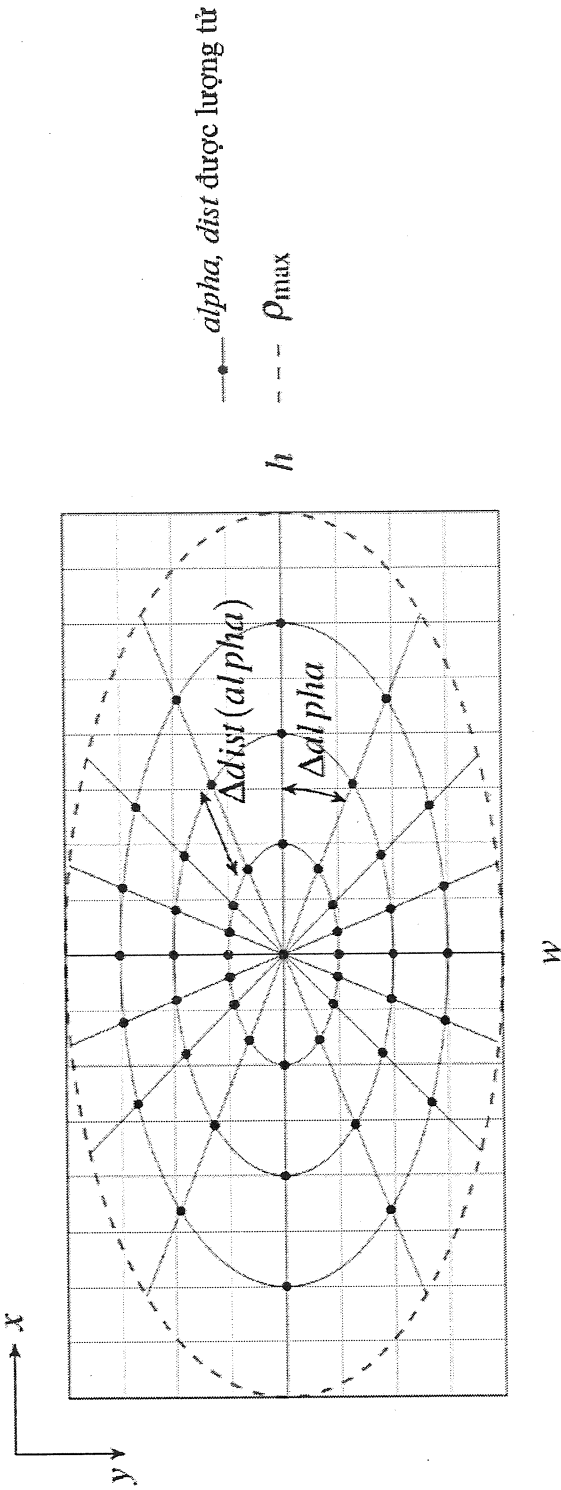


Fig.17

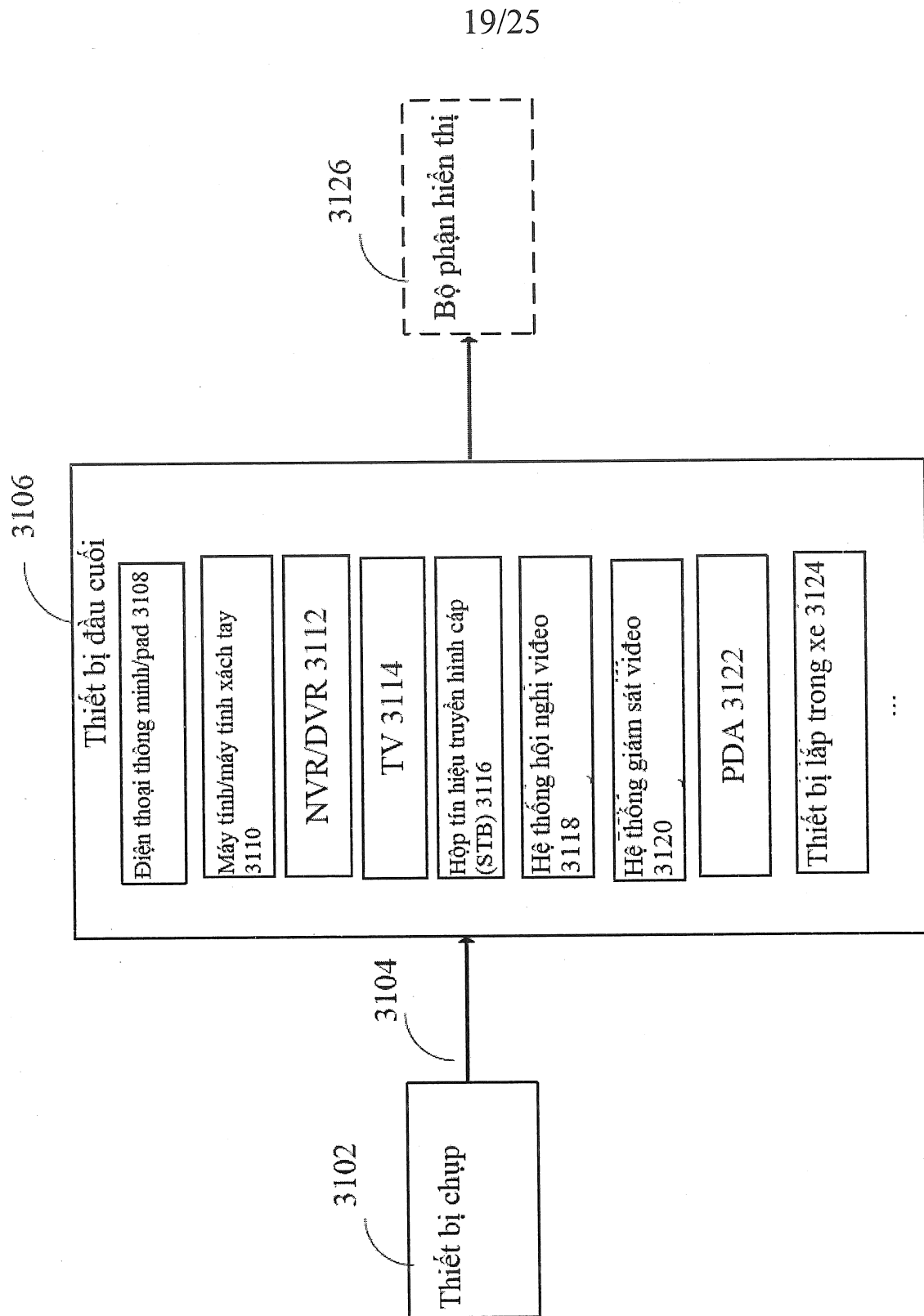


Fig.18

20/25

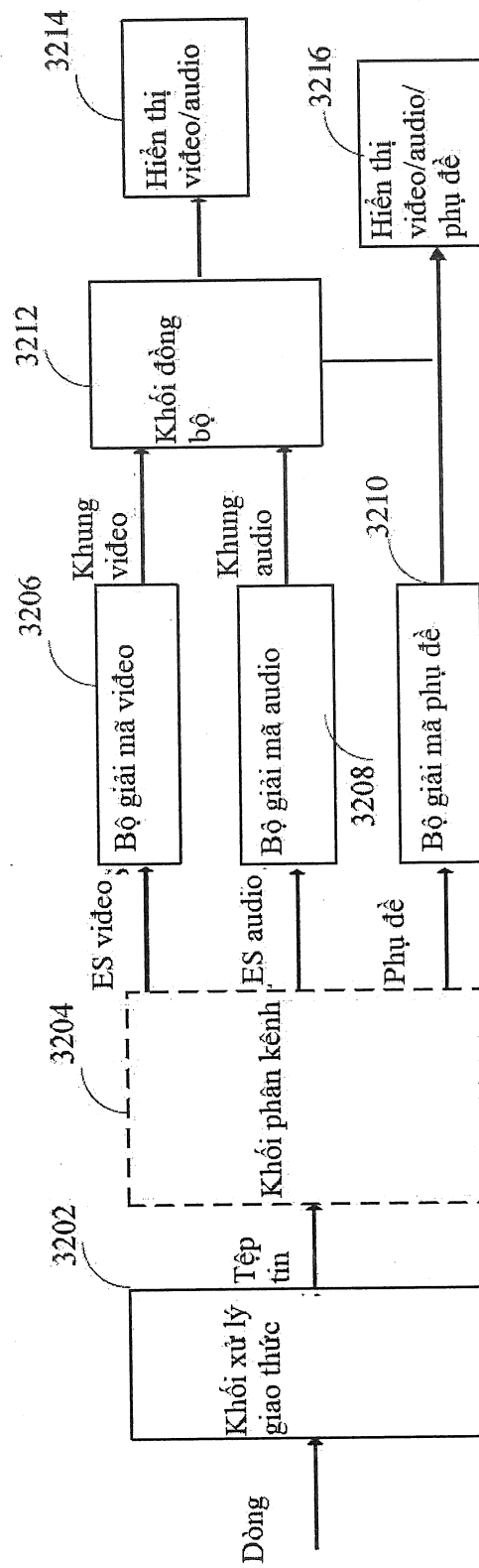
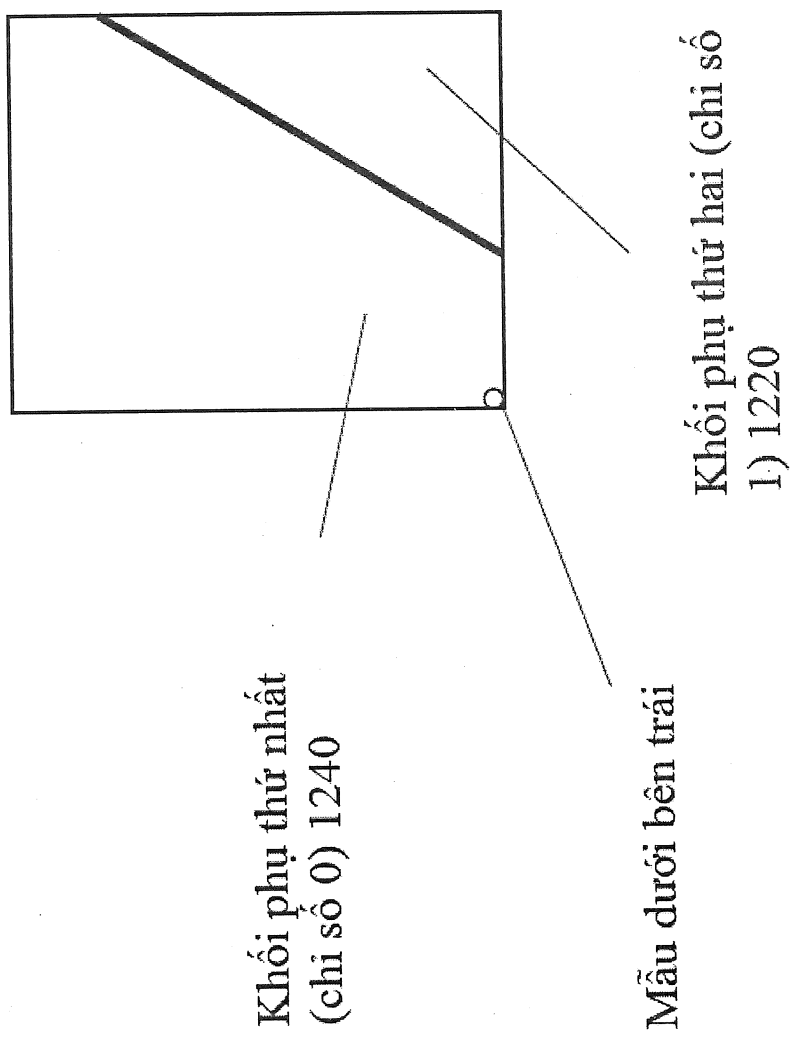


Fig.19

21/25



Đảo phần không tách

Fig.20

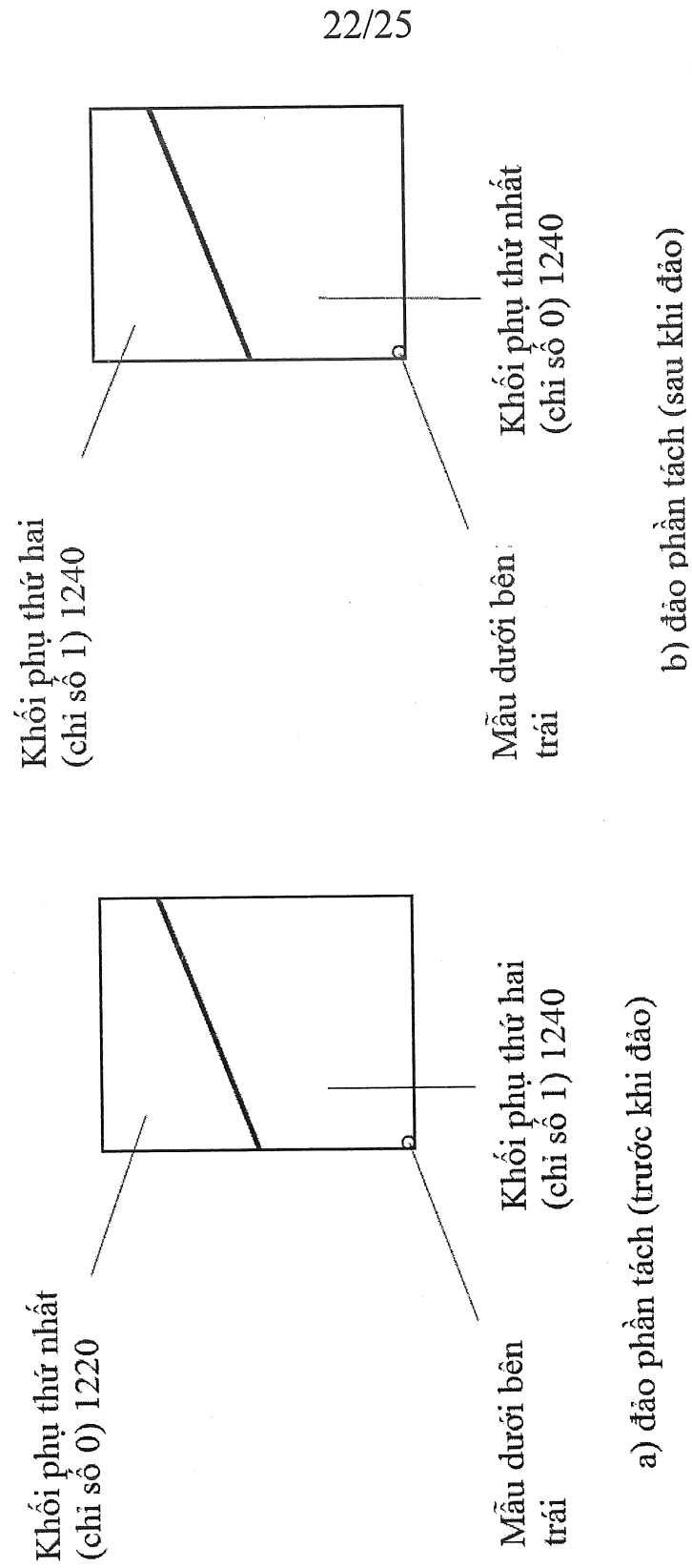


Fig.21

23/25

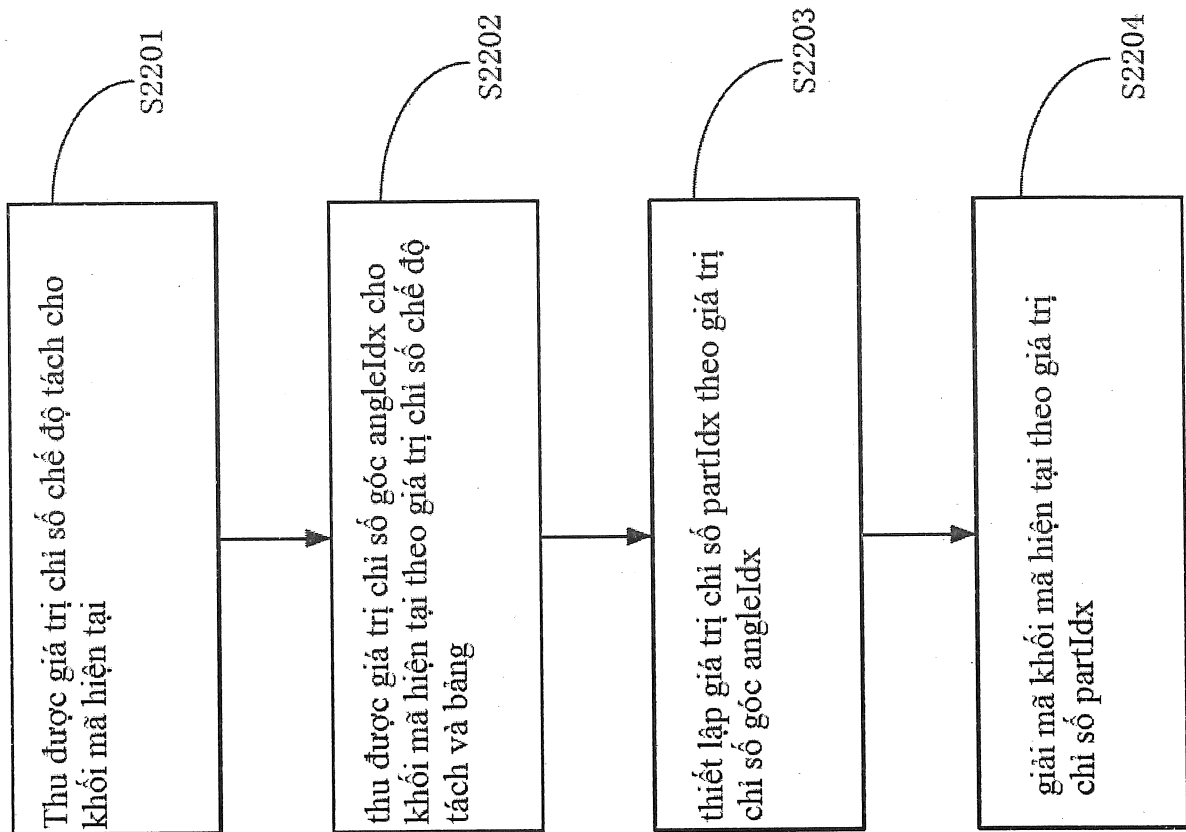


Fig.22

24/25

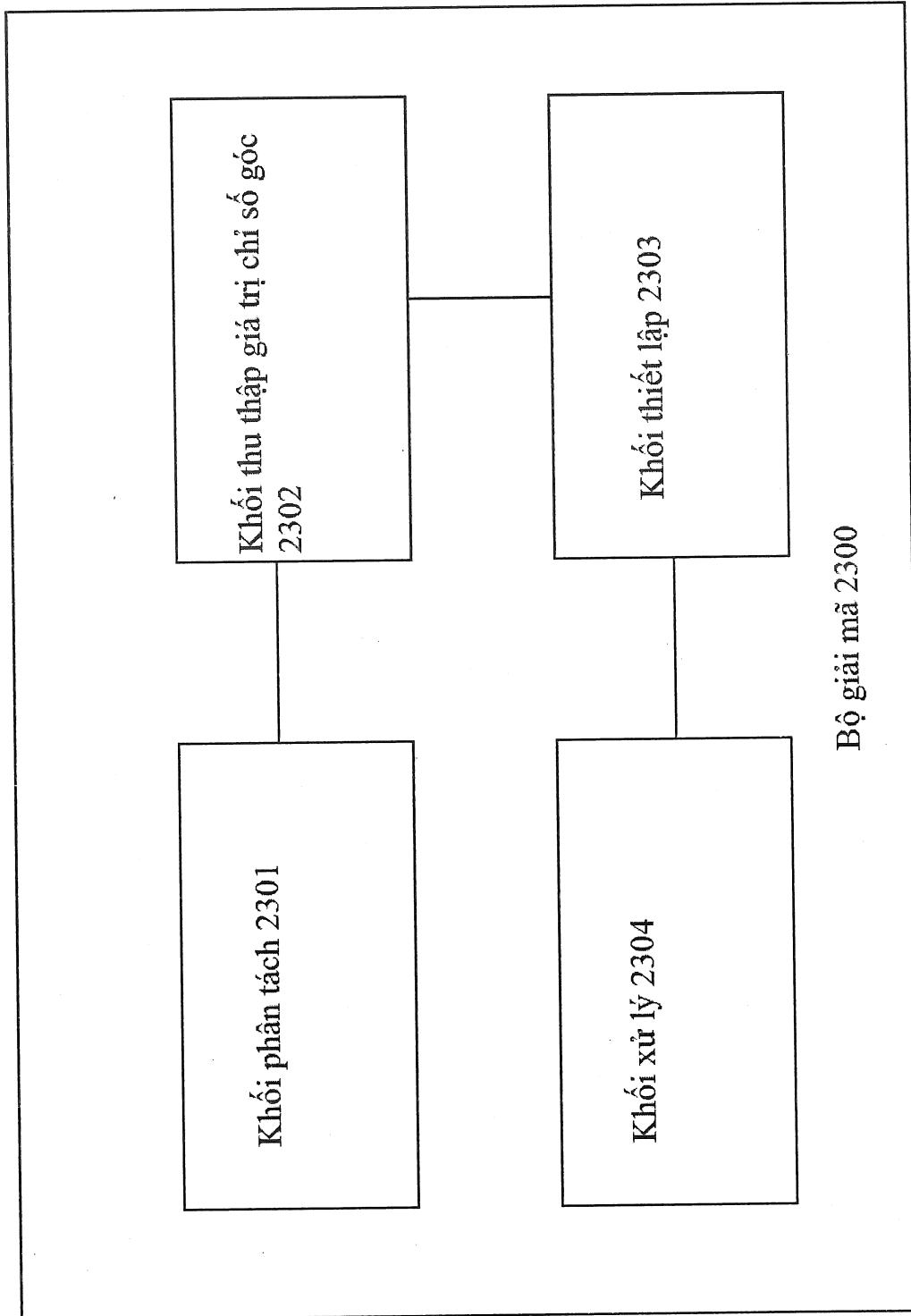


Fig.23

25/25

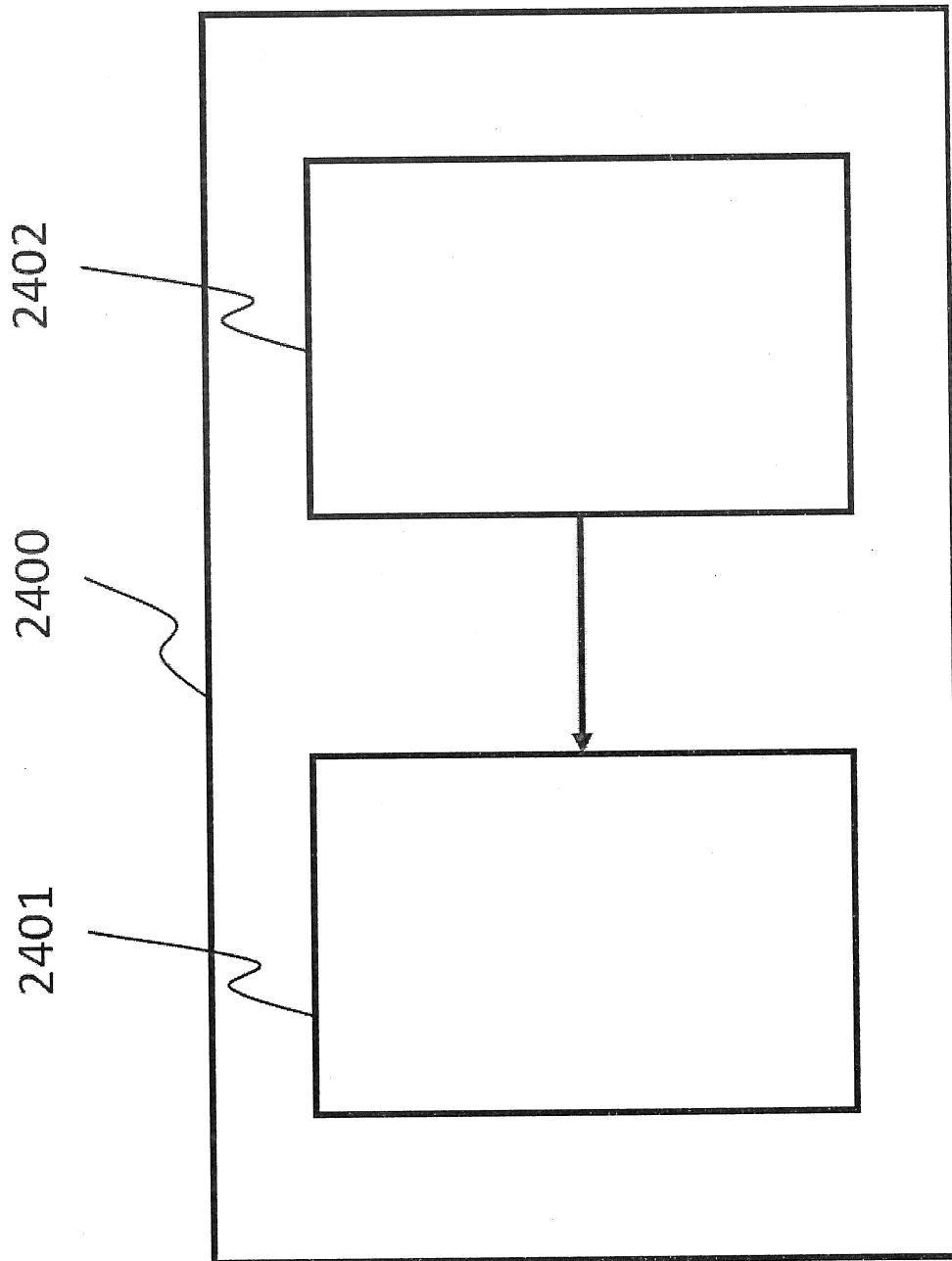


Fig.24