(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) **Cộng hòa xã hội chủ nghĩa Việt Nam (VN)**
CỤC SỞ HỮU TRÍ TUỆ

(11) **1-0050751**

(51)[2020.01] **G06T 9/00**

(13) **B**

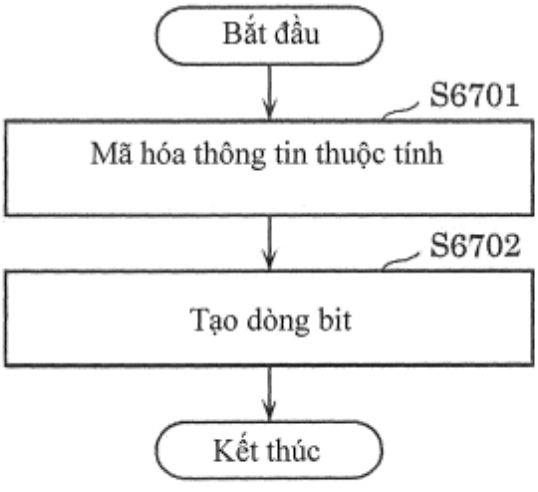
(21) 1-2021-03741
(22) 26/12/2019
(86) PCT/JP2019/051277 26/12/2019
(87) WO2020/138353 A1 02/07/2020
(30) 62/784998 26/12/2018 US
(45) 25/08/2025 449
(43) 25/10/2021 403A
(73) Panasonic Intellectual Property Corporation Of America (US)
2050 W 190th Street Suite 450, Torrance, California 90504 United States of America
(72) IGUCHI, Noritaka (JP); SUGIO, Toshiyasu (JP).
(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP MÃ HÓA DỮ LIỆU BA CHIỀU, PHƯƠNG PHÁP GIẢI MÃ DỮ LIỆU BA CHIỀU, THIẾT BỊ MÃ HÓA DỮ LIỆU BA CHIỀU VÀ THIẾT BỊ GIẢI MÃ DỮ LIỆU BA CHIỀU

(21) 1-2021-03741

(57) Sáng chế đề cập đến phương pháp mã hóa dữ liệu ba chiều bao gồm các bước: mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số (S6701); và tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất (S6702). Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

FIG. 129



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến phương pháp mã hóa dữ liệu ba chiều, phương pháp giải mã dữ liệu ba chiều, thiết bị mã hóa dữ liệu ba chiều, và thiết bị giải mã dữ liệu ba chiều.

Tình trạng kỹ thuật của sáng chế

Các thiết bị hoặc các dịch vụ ứng dụng dữ liệu ba chiều dự kiến sẽ cho thấy được việc sử dụng rộng rãi của chúng trong nhiều lĩnh vực, chẳng hạn như thị giác máy tính mà cho phép các thao tác tự quản của các xe ô tô hoặc các rôbôt, thông tin bản đồ, giám sát, kiểm tra cơ sở hạ tầng, và phân phối viđeo. Dữ liệu ba chiều được thu nhận qua các phương tiện khác nhau bao gồm bộ cảm biến khoảng cách chẳng hạn như máy đo khoảng cách, cũng như camera âm thanh nổi và sự kết hợp của các camera một mắt.

Các phương pháp biểu diễn dữ liệu ba chiều bao gồm phương pháp đã được biết đến là sơ đồ đám mây điểm mà biểu diễn hình dạng của cấu trúc ba chiều bởi đám mây điểm theo không gian ba chiều. Trong sơ đồ đám mây điểm, các vị trí và các màu sắc của đám mây điểm được lưu trữ. Trong khi đám mây điểm được kỳ vọng là phương pháp chủ đạo trong việc biểu diễn dữ liệu ba chiều, lượng lớn dữ liệu của đám mây điểm đòi hỏi việc nén của lượng dữ liệu ba chiều bằng cách mã hóa cho việc tích lũy và truyền, như trong trường hợp ảnh động hai chiều (các ví dụ bao gồm nhóm chuyên gia ảnh động (Moving Picture Experts Group, viết tắt là MPEG)- tạo mã viđeo nâng cao 4 (4 Advanced Video Coding, viết tắt là MPEG-4 AVC) và tạo mã viđeo hiệu quả cao (High Efficiency Video Coding, viết tắt là HEVC) được chuẩn hóa bởi MPEG).

Các phương pháp biểu diễn dữ liệu ba chiều bao gồm phương pháp đã được biết đến là sơ đồ đám mây điểm mà biểu diễn hình dạng của cấu trúc ba chiều bởi đám mây điểm theo không gian ba chiều. Trong sơ đồ đám mây điểm, các vị trí và các màu sắc của đám mây điểm được lưu trữ. Trong khi đám mây điểm được kỳ vọng là phương pháp chủ đạo trong việc biểu diễn dữ liệu ba chiều, lượng lớn dữ liệu của đám mây điểm đòi hỏi việc nén của lượng dữ liệu ba chiều bằng cách mã hóa cho việc tích lũy và việc truyền, như trong trường hợp ảnh động hai chiều (các ví dụ bao gồm nhóm chuyên gia ảnh động (Moving Picture Experts Group)- tạo mã viđeo nâng cao 4 (4 Advanced Video Coding, viết

tắt là MPEG-4 AVC) và tạo mã viđeo hiệu quả cao (High Efficiency Video Coding, viết tắt là HEVC) được chuẩn hóa bởi MPEG).

Hơn nữa, kỹ thuật tìm kiếm và hiển thị phương tiện nằm trong các vùng lân cận của phương tiện giao thông nhờ sử dụng dữ liệu bản đồ ba chiều đã được biết đến (ví dụ, xem tài liệu sáng chế (Patent Literature, viết tắt là PTL) 1).

Danh mục tài liệu trích dẫn

Tài liệu sáng chế

PTL 1: Công bố quốc tế số WO 2014/020663

Bản chất kỹ thuật của sáng chế

Vấn đề cần được giải quyết bởi sáng chế

Đã có nhu cầu đối với việc giải mã thông tin thuộc tính của điểm ba chiều một cách chính xác trong quy trình mã hóa dữ liệu ba chiều và giải mã dữ liệu ba chiều.

Mục đích của sáng chế là đề xuất phương pháp mã hóa dữ liệu ba chiều, phương pháp giải mã dữ liệu ba chiều, thiết bị mã hóa dữ liệu ba chiều, hoặc thiết bị giải mã dữ liệu ba chiều mà có khả năng giải mã thông tin thuộc tính của điểm ba chiều một cách chính xác.

Cách thức giải quyết vấn đề

Phương pháp mã hóa dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm: mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số; và tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất. Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Phương pháp giải mã dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm: thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit; và giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng. Dòng bit bao gồm thông tin điều

khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất, thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Hiệu quả của sáng chế

Sáng chế đề xuất phương pháp mã hóa dữ liệu ba chiều, phương pháp giải mã dữ liệu ba chiều, thiết bị mã hóa dữ liệu ba chiều, hoặc thiết bị giải mã dữ liệu ba chiều mà có khả năng giải mã thông tin thuộc tính của điểm ba chiều một cách chính xác.

Mô tả vắn tắt các hình vẽ

Fig.1 là sơ đồ minh họa cấu hình của hệ thống mã hóa và giải mã dữ liệu ba chiều theo phương án 1.

Fig.2 là sơ đồ minh họa ví dụ về cấu trúc của dữ liệu đám mây điểm theo phương án 1.

Fig.3 là sơ đồ minh họa ví dụ về cấu trúc của tệp dữ liệu chỉ báo dữ liệu đám mây điểm theo phương án 1.

Fig.4 là sơ đồ minh họa các loại của dữ liệu đám mây điểm theo phương án 1.

Fig.5 là sơ đồ minh họa cấu trúc của bộ mã hóa thứ nhất theo phương án 1.

Fig.6 là sơ đồ khối minh họa bộ mã hóa thứ nhất theo phương án 1.

Fig.7 là sơ đồ minh họa cấu trúc của bộ giải mã thứ nhất theo phương án 1.

Fig.8 là sơ đồ khối minh họa bộ giải mã thứ nhất theo phương án 1.

Fig.9 là sơ đồ minh họa cấu trúc của bộ mã hóa thứ hai theo phương án 1.

Fig.10 là sơ đồ khối minh họa bộ mã hóa thứ hai theo phương án 1.

Fig.11 là sơ đồ minh họa cấu trúc của bộ giải mã thứ hai theo phương án 1.

Fig.12 là sơ đồ khối minh họa bộ giải mã thứ hai theo phương án 1.

Fig.13 là sơ đồ minh họa ngăn xếp giao thức liên quan đến dữ liệu được mã hóa PCC theo phương án 1.

Fig.14 là sơ đồ khối của bộ mã hóa theo phương án 1.

Fig.15 là sơ đồ khối của bộ giải mã theo phương án 1.

Fig.16 là lưu đồ của quy trình mã hóa theo phương án 1.

Fig.17 là lưu đồ của quy trình giải mã theo phương án 1.

Fig.18 là sơ đồ minh họa cấu trúc cơ bản của ISOBMFF theo phương án 2.

Fig.19 là sơ đồ minh họa ngăn xếp giao thức theo phương án 2.

Fig.20 là sơ đồ minh họa ví dụ trong đó bộ phận NAL được lưu trữ trong tệp dùng cho cođec 1 theo phương án 2.

Fig.21 là sơ đồ minh họa ví dụ trong đó bộ phận NAL được lưu trữ trong tệp dùng cho cođec 2 theo phương án 2.

Fig.22 là sơ đồ minh họa cấu trúc của bộ đa hợp thứ nhất theo phương án 2.

Fig.23 là sơ đồ minh họa cấu trúc của bộ giải đa hợp thứ nhất theo phương án 2.

Fig.24 là sơ đồ minh họa cấu trúc của bộ đa hợp thứ hai theo phương án 2.

Fig.25 là sơ đồ minh họa cấu trúc của bộ giải đa hợp thứ hai theo phương án 2.

Fig.26 là lưu đồ của quy trình xử lý được thực hiện bởi bộ đa hợp thứ nhất theo phương án 2.

Fig.27 là lưu đồ của quy trình xử lý được thực hiện bởi bộ đa hợp thứ hai theo phương án 2.

Fig.28 là lưu đồ của quy trình xử lý được thực hiện bởi bộ giải đa hợp thứ nhất và bộ giải mã thứ nhất theo phương án 2.

Fig.29 là lưu đồ của quy trình xử lý được thực hiện bởi bộ giải đa hợp thứ hai và bộ giải mã thứ hai theo phương án 2.

Fig.30 là sơ đồ minh họa các cấu trúc của bộ mã hóa và bộ đa hợp thứ ba theo phương án 3.

Fig.31 là sơ đồ minh họa các cấu trúc của bộ giải đa hợp thứ ba và bộ giải mã theo phương án 3.

Fig.32 là lưu đồ của quy trình xử lý được thực hiện bởi bộ đa hợp thứ ba theo phương án 3.

Fig.33 là lưu đồ của quy trình xử lý được thực hiện bởi bộ giải đa hợp thứ ba và

bộ giải mã theo phương án 3.

Fig.34 là lưu đồ của quy trình xử lý được thực hiện bởi dữ liệu ba chiều thiết bị lưu trữ theo phương án 3.

Fig.35 là lưu đồ của quy trình xử lý được thực hiện bởi thiết bị thu nhận dữ liệu ba chiều theo phương án 3.

Fig.36 là sơ đồ minh họa các cấu trúc của bộ mã hóa và bộ đa hợp theo phương án 4.

Fig.37 là sơ đồ minh họa ví dụ về cấu trúc của dữ liệu được mã hóa theo phương án 4.

Fig.38 là sơ đồ minh họa ví dụ về cấu trúc của dữ liệu được mã hóa và bộ phận NAL theo phương án 4.

Fig.39 là sơ đồ minh họa ví dụ ngữ nghĩa của pcc_nal_unit_type theo phương án 4.

Fig.40 là sơ đồ minh họa ví dụ về thứ tự truyền của các bộ phận NAL theo phương án 4.

Fig.41 là lưu đồ của quy trình xử lý được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều theo phương án 4.

Fig.42 là lưu đồ của quy trình xử lý được thực hiện bởi thiết bị giải mã dữ liệu ba chiều theo phương án 4.

Fig.43 là lưu đồ của quy trình đa hợp theo phương án 4.

Fig.44 là lưu đồ của quy trình giải đa hợp theo phương án 4.

Fig.45 là lưu đồ của quy trình xử lý được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều theo phương án 4.

Fig.46 là lưu đồ của quy trình xử lý được thực hiện bởi thiết bị giải mã dữ liệu ba chiều theo phương án 4.

Fig.47 là sơ đồ khối minh họa bộ chia theo phương án 5.

Fig.48 là sơ đồ minh họa ví dụ về việc chia lát và việc chia mảnh theo phương án 5.

Fig.49 là sơ đồ minh họa ví dụ về mẫu hình chia lát và mẫu hình chia mảnh theo

phương án 5.

Fig.50 là sơ đồ khối của bộ mã hóa thứ nhất theo phương án 6.

Fig.51 là sơ đồ khối của bộ giải mã thứ nhất theo phương án 6.

Fig.52 là sơ đồ minh họa các ví dụ về hình dạng mảnh theo phương án 6.

Fig.53 là sơ đồ minh họa ví dụ về các mảnh và các lát theo phương án 6.

Fig.54 là sơ đồ khối của bộ chia theo phương án 6.

Fig.55 là sơ đồ minh họa ví dụ về bản đồ theo hình chiếu từ trên xuống của dữ liệu đám mây điểm theo phương án 6.

Fig.56 là sơ đồ minh họa ví dụ về việc chia mảnh theo phương án 6.

Fig.57 là sơ đồ minh họa ví dụ về việc chia mảnh theo phương án 6.

Fig.58 là sơ đồ minh họa ví dụ về việc chia mảnh theo phương án 6.

Fig.59 là sơ đồ minh họa ví dụ về dữ liệu của các mảnh được lưu trữ trong máy chủ theo phương án 6.

Fig.60 là sơ đồ minh họa hệ thống liên quan đến việc chia mảnh theo phương án 6.

Fig.61 là sơ đồ minh họa ví dụ về việc chia lát theo phương án 6.

Fig.62 là sơ đồ minh họa ví dụ về các mối tương quan phụ thuộc theo phương án 6.

Fig.63 là sơ đồ minh họa ví dụ về thứ tự giải mã của dữ liệu theo phương án 6.

Fig.64 là sơ đồ minh họa ví dụ về dữ liệu được mã hóa của các mảnh theo phương án 6.

Fig.65 là sơ đồ khối của bộ kết hợp theo phương án 6.

Fig.66 là sơ đồ minh họa ví dụ cấu trúc của dữ liệu được mã hóa và các bộ phận NAL theo phương án 6.

Fig.67 là lưu đồ của quy trình mã hóa theo phương án 6.

Fig.68 là lưu đồ của quy trình giải mã theo phương án 6.

Fig.69 là sơ đồ minh họa ví dụ về cú pháp của thông tin bổ sung mảnh theo phương án 6.

Fig.70 là sơ đồ khối của hệ thống mã hóa và giải mã theo phương án 6.

Fig.71 là sơ đồ minh họa ví dụ về cú pháp của thông tin bổ sung lát theo phương án 6.

Fig.72 là lưu đồ của quy trình mã hóa theo phương án 6.

Fig.73 là lưu đồ của quy trình giải mã theo phương án 6.

Fig.74 là lưu đồ của quy trình mã hóa theo phương án 6.

Fig.75 là lưu đồ của quy trình giải mã theo phương án 6.

Fig.76 là lưu đồ của quy trình khởi tạo lại công cụ mã hóa/giải mã CABAC đáp lại cờ khởi tạo CABAC trong việc mã hóa hoặc giải mã theo phương án 7.

Fig.77 là sơ đồ khối minh họa cấu hình của bộ mã hóa thứ nhất được bao gồm trong thiết bị mã hóa dữ liệu ba chiều theo phương án 7.

Fig.78 là sơ đồ khối minh họa cấu hình của bộ chia theo phương án 7.

Fig.79 là sơ đồ khối minh họa cấu hình của bộ mã hóa thông tin hình học và bộ mã hóa thông tin thuộc tính theo phương án 7.

Fig.80 là sơ đồ khối minh họa cấu hình của bộ giải mã thứ nhất theo phương án 7.

Fig.81 là sơ đồ khối minh họa cấu hình của bộ giải mã thông tin hình học và bộ giải mã thông tin thuộc tính theo phương án 7.

Fig.82 là lưu đồ minh họa ví dụ về quy trình xử lý được kết hợp với việc khởi tạo của CABAC trong việc mã hóa của thông tin hình học hoặc mã hóa của thông tin thuộc tính theo phương án 7.

Fig.83 là sơ đồ minh họa ví dụ về các thời điểm của việc khởi tạo CABAC đối với dữ liệu đám mây điểm ở dạng dòng bit theo phương án 7.

Fig.84 là sơ đồ minh họa cấu hình của dữ liệu được mã hóa và phương pháp lưu trữ dữ liệu được mã hóa trong bộ phận NAL theo phương án 7.

Fig.85 là lưu đồ minh họa ví dụ về quy trình xử lý được kết hợp với việc khởi tạo của CABAC trong việc giải mã của thông tin hình học hoặc giải mã của thông tin thuộc tính theo phương án 7.

Fig.86 là lưu đồ của quy trình mã hóa dữ liệu đám mây điểm theo phương án 7.

Fig.87 là lưu đồ minh họa ví dụ về quy trình cập nhật thông tin bổ sung theo

phương án 7.

Fig.88 là lưu đồ minh họa ví dụ về quy trình khởi tạo CABAC theo phương án 7.

Fig.89 là lưu đồ minh họa quy trình giải mã dữ liệu đám mây điểm theo phương án 7.

Fig.90 là lưu đồ minh họa ví dụ về quy trình khởi tạo bộ giải mã CABAC theo phương án 7.

Fig.91 là sơ đồ minh họa ví dụ về các mảnh và các lát theo phương án 7.

Fig.92 là lưu đồ minh họa ví dụ về phương pháp xác định xem có khởi tạo CABAC hay không và xác định trị số khởi tạo ngữ cảnh theo phương án 7.

Fig.93 là sơ đồ minh họa ví dụ về trường hợp trong đó bản đồ, mà là hình chiếu từ trên xuống của dữ liệu đám mây điểm được thu nhận bởi LiDAR, được chia thành các mảnh theo phương án 7.

Fig.94 là lưu đồ minh họa ví dụ khác về phương pháp xác định xem có khởi tạo CABAC hay không và xác định trị số khởi tạo ngữ cảnh theo phương án 7.

Fig.95 là sơ đồ mô tả quy trình được thực hiện bởi bộ lượng tử hóa và bộ lượng tử hóa ngược theo phương án 8.

Fig.96 là sơ đồ mô tả trị số mặc định và đenta lượng tử hóa của trị số lượng tử hóa theo phương án 8.

Fig.97 là sơ đồ khối minh họa cấu hình của bộ mã hóa thứ nhất được bao gồm trong thiết bị mã hóa dữ liệu ba chiều theo phương án 8.

Fig.98 là sơ đồ khối minh họa cấu hình của bộ chia theo phương án 8.

Fig.99 là sơ đồ khối minh họa cấu hình của bộ mã hóa thông tin hình học và bộ mã hóa thông tin thuộc tính theo phương án 8.

Fig.100 là sơ đồ khối minh họa cấu hình của bộ giải mã thứ nhất theo phương án 8.

Fig.101 là sơ đồ khối minh họa cấu hình của bộ giải mã thông tin hình học và bộ giải mã thông tin thuộc tính theo phương án 8.

Fig.102 là lưu đồ minh họa ví dụ về quy trình liên quan đến việc xác định của trị số lượng tử hóa trong việc mã hóa của thông tin hình học hoặc mã hóa của thông tin thuộc

tính theo phương án 8.

Fig.103 là lưu đồ minh họa ví dụ về quy trình giải mã thông tin hình học và thông tin thuộc tính theo phương án 8.

Fig.104 là sơ đồ mô tả ví dụ thứ nhất về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.105 là sơ đồ mô tả ví dụ thứ hai về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.106 là sơ đồ mô tả ví dụ thứ ba về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.107 là lưu đồ của quy trình mã hóa dữ liệu đám mây điểm theo phương án 8.

Fig.108 là lưu đồ minh họa ví dụ về quy trình xác định trị số QP và cập nhật thông tin bổ sung theo phương án 8.

Fig.109 là lưu đồ minh họa ví dụ về quy trình mã hóa theo phương án 8.

Fig.110 là lưu đồ minh họa quy trình giải mã dữ liệu đám mây điểm theo phương án 8.

Fig.111 là lưu đồ minh họa ví dụ về quy trình thu nhận các trị số QP và giải mã trị số QP đối với lát hoặc mảnh theo phương án 8.

Fig.112 là sơ đồ minh họa ví dụ cú pháp của GPS theo phương án 8.

Fig.113 là sơ đồ minh họa ví dụ cú pháp của APS theo phương án 8.

Fig.114 là sơ đồ minh họa ví dụ cú pháp của đoạn đầu của thông tin hình học theo phương án 8.

Fig.115 là sơ đồ minh họa ví dụ cú pháp của đoạn đầu của thông tin thuộc tính theo phương án 8.

Fig.116 là sơ đồ mô tả ví dụ khác về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.117 là sơ đồ mô tả ví dụ khác về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.118 là sơ đồ mô tả ví dụ thứ chín về phương pháp truyền thông số lượng tử hóa theo phương án 8.

Fig.119 là sơ đồ mô tả ví dụ về việc điều khiển của trị số QP theo phương án 8.

Fig.120 là lưu đồ minh họa ví dụ về phương pháp xác định trị số QP dựa vào chất lượng của đối tượng theo phương án 8.

Fig.121 là lưu đồ minh họa ví dụ về phương pháp xác định trị số QP dựa vào việc điều khiển tốc độ theo phương án 8.

Fig.122 là lưu đồ minh họa quy trình mã hóa theo phương án 8.

Fig.123 là lưu đồ minh họa quy trình giải mã theo phương án 8.

Fig.124 là sơ đồ minh họa ví dụ về phương pháp truyền thông số lượng tử hóa theo phương án 9.

Fig.125 là sơ đồ thể hiện ví dụ thứ nhất về cú pháp của APS và cú pháp của đoạn đầu của thông tin thuộc tính theo phương án 9.

Fig.126 là sơ đồ thể hiện ví dụ thứ hai về cú pháp của APS theo phương án 9.

Fig.127 là sơ đồ thể hiện ví dụ thứ hai về cú pháp của đoạn đầu của thông tin thuộc tính theo phương án 9.

Fig.128 là sơ đồ thể hiện mối tương quan giữa SPS, APS, và đoạn đầu của thông tin thuộc tính theo phương án 9.

Fig.129 là lưu đồ của quy trình mã hóa theo phương án 9.

Fig.130 là lưu đồ của quy trình giải mã theo phương án 9.

Mô tả chi tiết sáng chế

Phương pháp mã hóa dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm: mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số; và tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất. Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì dòng bit bao gồm thông tin nhận dạng thứ nhất dùng để

nhận dạng loại của thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ nhất tương ứng được tạo ra, thiết bị giải mã dữ liệu ba chiều đã thu dòng bit có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, các đoạn thông tin loại có thể được lưu trữ trong thông tin điều khiển theo trình tự được định trước, và thông tin nhận dạng thứ nhất có thể chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

Với cấu hình như vậy, vì thông tin loại được chỉ báo theo trình tự được định trước mà không có thông tin chỉ báo thông tin loại, lượng dữ liệu của dòng bit có thể được giảm, và lượng của dòng bit được truyền có thể được giảm.

Ví dụ, dòng bit có thể còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai có thể bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

Với cấu hình như vậy, vì mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số, thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ hai tương ứng có thể được mã hóa bằng cách sử dụng trị số tham chiếu.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất có thể bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số.

Với cấu hình như vậy, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, dòng bit có thể còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai có thể bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì dòng bit bao gồm thông tin nhận dạng thứ hai dùng để nhận dạng loại của thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ hai tương ứng được tạo ra, có khả năng tạo ra dòng bit mà có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất có thể bao gồm N trường trong đó N thông số được lưu trữ, N lớn hơn hoặc bằng 2, và trong thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính

thứ nhất, một trong số N trường có thể bao gồm trị số chỉ báo tính không hợp lệ, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

Với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều đã thu dòng bit có thể nhận dạng loại của thông tin thuộc tính thứ nhất bằng cách sử dụng thông tin nhận dạng thứ nhất và bỏ qua quy trình giải mã trong trường hợp thông tin điều khiển thuộc tính thứ nhất cụ thể, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, trong việc mã hóa, các đoạn thông tin thuộc tính có thể được lượng tử hóa bằng cách sử dụng các thông số lượng tử hóa như các thông số.

Với cấu hình như vậy, vì thông số được biểu diễn bằng cách sử dụng độ chênh lệch với trị số tham chiếu, có khả năng nâng cao hiệu quả mã hóa đối với việc lượng tử hóa.

Phương pháp giải mã dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm: thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit; và giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng. Dòng bit bao gồm thông tin điều khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất, thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì có khả năng nhận dạng loại của thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, các đoạn thông tin loại có thể được lưu trữ trong thông tin điều khiển theo trình tự được định trước, và thông tin nhận dạng thứ nhất có thể chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

Với cấu hình như vậy, vì thông tin loại được chỉ báo theo trình tự được định trước mà không có thông tin chỉ báo thông tin loại, lượng dữ liệu của dòng bit có thể được giảm, và lượng của dòng bit được truyền có thể được giảm.

Ví dụ, dòng bit có thể còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai có thể bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

Với cấu hình như vậy, vì có khả năng giải mã thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ hai nhờ sử dụng trị số tham chiếu, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất có thể bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số.

Với cấu hình như vậy, vì có khả năng giải mã thông tin thuộc tính nhờ sử dụng trị số tham chiếu và thông tin sai lệch, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, dòng bit có thể còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai có thể bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì có khả năng nhận dạng loại của thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ hai nhờ sử dụng thông tin nhận dạng thứ hai, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất có thể bao gồm các trường trong đó các thông số được lưu trữ, và trong việc giải mã, thông số được lưu trữ trong trường cụ thể trong số các trường của thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính thứ nhất có thể được bỏ qua, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

Với cấu hình như vậy, vì có khả năng nhận dạng loại của thông tin thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất và bỏ qua quy trình giải mã trong trường hợp thông tin điều khiển thuộc tính thứ nhất cụ thể, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, trong việc giải mã, các đoạn thông tin thuộc tính được mã hóa có thể được lượng tử hóa ngược nhờ sử dụng các thông số lượng tử hóa như các thông số.

Với cấu hình như vậy, có khả năng giải mã một cách chính xác thông tin thuộc tính

trên điểm ba chiều.

Thiết bị mã hóa dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm bộ xử lý và bộ nhớ. Nhờ sử dụng bộ nhớ, bộ xử lý: mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số; và tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất. Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất tương ứng với đoạn thông tin thuộc tính khác trong số các đoạn thông tin thuộc tính và được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì dòng bit bao gồm thông tin nhận dạng thứ nhất dùng để nhận dạng loại của thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ nhất tương ứng được tạo ra, có khả năng tạo ra dòng bit mà có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Thiết bị giải mã dữ liệu ba chiều theo một khía cạnh của sáng chế bao gồm bộ xử lý và bộ nhớ. Nhờ sử dụng bộ nhớ, bộ xử lý: thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit; và giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng. Dòng bit bao gồm thông tin điều khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất, thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, và mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất tương ứng với đoạn thông tin thuộc tính khác trong số các đoạn thông tin thuộc tính và được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì có khả năng nhận dạng loại của thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Lưu ý rằng các khía cạnh chung hoặc cụ thể này có thể được thực hiện dưới dạng hệ thống, phương pháp, mạch tích hợp, chương trình máy tính, hoặc phương tiện ghi đọc được bởi máy tính chẳng hạn như CD-ROM, hoặc có thể được thực hiện dưới dạng sự kết

hợp bất kỳ của hệ thống, phương pháp, mạch tích hợp, chương trình máy tính, và phương tiện ghi.

Phần dưới đây mô tả các phương án dựa vào các hình vẽ. Lưu ý rằng các phương án dưới đây thể hiện các phương án ví dụ về sáng chế. Các giá trị số, các hình dạng, các vật liệu, các thành phần cấu trúc, sự bố trí và sự liên kết của các thành phần cấu trúc, các bước, thứ tự xử lý của các bước, v.v., được thể hiện theo các phương án dưới đây chỉ là các ví dụ, và do đó không nhằm giới hạn sáng chế. Trong số các thành phần cấu trúc được mô tả theo các phương án dưới đây, các thành phần cấu trúc không được trích dẫn trong bất kỳ một trong số các điểm yêu cầu bảo hộ độc lập mà chỉ báo các khái niệm rộng nhất sẽ được mô tả như các thành phần cấu trúc tùy chọn.

Phương án 1

Khi nhờ sử dụng dữ liệu được mã hóa của đám mây điểm trong thiết bị hoặc cho dịch vụ thực tế, thông tin được yêu cầu đối với ứng dụng mong muốn được truyền và được thu để làm giảm độ rộng dải mạng. Tuy nhiên, các cấu trúc mã hóa thông thường đối với dữ liệu ba chiều không có chức năng như vậy, và cũng không có phương pháp mã hóa đối với chức năng như vậy.

Phương án 1 được mô tả dưới đây liên quan đến phương pháp mã hóa dữ liệu ba chiều và thiết bị mã hóa dữ liệu ba chiều đối với dữ liệu được mã hóa của đám mây điểm ba chiều mà cung cấp chức năng truyền và thu thông tin được yêu cầu đối với ứng dụng, phương pháp giải mã dữ liệu ba chiều và thiết bị giải mã dữ liệu ba chiều dùng để giải mã dữ liệu được mã hóa, phương pháp đa hợp dữ liệu ba chiều để đa hợp dữ liệu được mã hóa, và phương pháp truyền dữ liệu ba chiều để truyền dữ liệu được mã hóa.

Cụ thể, hiện tại, phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai đang được nghiên cứu là các phương pháp mã hóa (các sơ đồ mã hóa) đối với dữ liệu đám mây điểm. Tuy nhiên, không có phương pháp được định rõ để lưu trữ cấu hình của dữ liệu được mã hóa và dữ liệu được mã hóa theo định dạng hệ thống. Do vậy, có vấn đề là bộ mã hóa không thể thực hiện quy trình xử lý MUX (đa hợp), truyền, hoặc tích lũy của dữ liệu.

Ngoài ra, không có phương pháp hỗ trợ định dạng mà bao gồm hai cođec, phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai, chẳng hạn như nén đám mây điểm (point cloud compression, viết tắt là PCC).

Liên quan đến phương án này, cấu hình của dữ liệu được mã hóa PCC mà bao gồm

hai cođec, phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai, và phương pháp lưu trữ dữ liệu được mã hóa theo định dạng hệ thống sẽ được mô tả.

Cấu hình của hệ thống mã hóa và giải mã dữ liệu ba chiều (dữ liệu đám mây điểm) theo phương án này sẽ được mô tả trước tiên. Fig.1 là sơ đồ thể hiện ví dụ về cấu hình của hệ thống mã hóa và giải mã dữ liệu ba chiều theo phương án này. Như được thể hiện trên Fig.1, hệ thống mã hóa và giải mã dữ liệu ba chiều bao gồm hệ thống mã hóa dữ liệu ba chiều 4601, hệ thống giải mã dữ liệu ba chiều 4602, đầu cuối cảm biến 4603, và bộ nối bên ngoài 4604.

Hệ thống mã hóa dữ liệu ba chiều 4601 tạo ra dữ liệu được mã hóa hoặc dữ liệu được đa hợp bằng cách mã hóa dữ liệu đám mây điểm, mà là dữ liệu ba chiều. Hệ thống mã hóa dữ liệu ba chiều 4601 có thể là thiết bị mã hóa dữ liệu ba chiều được thực hiện bởi thiết bị đơn hoặc hệ thống được thực hiện bởi các thiết bị. Thiết bị mã hóa dữ liệu ba chiều có thể bao gồm phần của các bộ xử lý được bao gồm trong hệ thống mã hóa dữ liệu ba chiều 4601.

Hệ thống mã hóa dữ liệu ba chiều 4601 bao gồm hệ thống tạo dữ liệu đám mây điểm 4611, bộ trình diễn 4612, bộ mã hóa 4613, bộ đa hợp 4614, bộ phận đầu vào/đầu ra 4615, và bộ điều khiển 4616. Hệ thống tạo dữ liệu đám mây điểm 4611 bao gồm bộ thu nhận thông tin cảm biến 4617, và bộ tạo dữ liệu đám mây điểm 4618.

Bộ thu nhận thông tin cảm biến 4617 thu nhận thông tin cảm biến từ đầu cuối cảm biến 4603, và đưa ra thông tin cảm biến tới bộ tạo dữ liệu đám mây điểm 4618. Bộ tạo dữ liệu đám mây điểm 4618 tạo ra dữ liệu đám mây điểm từ thông tin cảm biến, và đưa ra dữ liệu đám mây điểm tới bộ mã hóa 4613.

Bộ trình diễn 4612 biểu diễn thông tin cảm biến hoặc dữ liệu đám mây điểm tới người dùng. Ví dụ, bộ trình diễn 4612 hiển thị thông tin hoặc hình ảnh dựa vào thông tin cảm biến hoặc dữ liệu đám mây điểm.

Bộ mã hóa 4613 mã hóa (nén) dữ liệu đám mây điểm, và đưa ra dữ liệu được mã hóa nhận được, thông tin điều khiển (thông tin báo hiệu) được thu nhận trong quá trình mã hóa, và thông tin bổ sung khác tới bộ đa hợp 4614. Thông tin bổ sung bao gồm thông tin cảm biến chẳng hạn.

Bộ đa hợp 4614 tạo ra dữ liệu được đa hợp bằng cách đa hợp dữ liệu được mã hóa, thông tin điều khiển, và thông tin bổ sung được đưa vào tới đó từ bộ mã hóa 4613. Định

dạng của dữ liệu được đa hợp là định dạng tệp cho việc tích lũy hoặc định dạng gói cho việc truyền chẳng hạn.

Bộ phận đầu vào/đầu ra 4615 (giao diện hoặc bộ phận truyền thông chẳng hạn) đưa ra dữ liệu được đa hợp tới phía ngoài. Theo cách khác dữ liệu được đa hợp có thể được tích lũy trong bộ tích lũy, chẳng hạn như bộ nhớ trong. Bộ điều khiển 4616 (hoặc bộ thi hành ứng dụng) điều khiển mỗi bộ xử lý. Nghĩa là, bộ điều khiển 4616 điều khiển mã hóa, đa hợp, hoặc quy trình xử lý khác.

Lưu ý rằng thông tin cảm biến có thể được đưa vào tới bộ mã hóa 4613 hoặc bộ đa hợp 4614. Theo cách khác bộ phận đầu vào/đầu ra 4615 có thể đưa ra dữ liệu đám mây điểm hoặc dữ liệu được mã hóa tới phía ngoài.

Tín hiệu truyền (dữ liệu được đa hợp) được đưa ra từ hệ thống mã hóa dữ liệu ba chiều 4601 được đưa vào tới hệ thống giải mã dữ liệu ba chiều 4602 qua bộ nối bên ngoài 4604.

Hệ thống giải mã dữ liệu ba chiều 4602 tạo ra dữ liệu đám mây điểm, mà là dữ liệu ba chiều, bằng cách giải mã dữ liệu được mã hóa hoặc dữ liệu được đa hợp. Lưu ý rằng hệ thống giải mã dữ liệu ba chiều 4602 có thể là thiết bị giải mã dữ liệu ba chiều được thực hiện bởi thiết bị đơn hoặc hệ thống được thực hiện bởi các thiết bị. Thiết bị giải mã dữ liệu ba chiều có thể bao gồm phần của các bộ xử lý được bao gồm trong hệ thống giải mã dữ liệu ba chiều 4602.

Hệ thống giải mã dữ liệu ba chiều 4602 bao gồm bộ thu nhận thông tin cảm biến 4621, bộ phận đầu vào/đầu ra 4622, bộ giải đa hợp 4623, bộ giải mã 4624, bộ trình diễn 4625, giao diện người dùng 4626, và bộ điều khiển 4627.

Bộ thu nhận thông tin cảm biến 4621 thu nhận thông tin cảm biến từ đầu cuối cảm biến 4603.

Bộ phận đầu vào/đầu ra 4622 thu nhận tín hiệu truyền, giải mã tín hiệu truyền thành dữ liệu được đa hợp (định dạng tệp hoặc gói), và đưa ra dữ liệu được đa hợp tới bộ giải đa hợp 4623.

Bộ giải đa hợp 4623 thu nhận dữ liệu được mã hóa, thông tin điều khiển, và thông tin bổ sung từ dữ liệu được đa hợp, và đưa ra dữ liệu được mã hóa, thông tin điều khiển, và thông tin bổ sung tới bộ giải mã 4624.

Bộ giải mã 4624 tái cấu trúc dữ liệu đám mây điểm bằng cách giải mã dữ liệu được

mã hóa.

Bộ trình diễn 4625 biểu diễn dữ liệu đám mây điểm tới người dùng. Ví dụ, bộ trình diễn 4625 hiển thị thông tin hoặc hình ảnh dựa vào dữ liệu đám mây điểm. Giao diện người dùng 4626 thu nhận chỉ báo dựa vào thao tác bởi người dùng. Bộ điều khiển 4627 (hoặc bộ thi hành ứng dụng) điều khiển mỗi bộ xử lý. Nghĩa là, bộ điều khiển 4627 điều khiển giải đa hợp, giải mã, trình diễn, hoặc quy trình xử lý khác.

Lưu ý rằng bộ phận đầu vào/đầu ra 4622 có thể thu nhận dữ liệu đám mây điểm hoặc dữ liệu được mã hóa từ phía ngoài. Bộ trình diễn 4625 có thể thu nhận thông tin bổ sung, chẳng hạn như thông tin cảm biến, và biểu diễn thông tin dựa vào thông tin bổ sung. Bộ trình diễn 4625 có thể thực hiện việc trình diễn dựa vào chỉ báo từ người dùng được thu nhận trên giao diện người dùng 4626.

Đầu cuối cảm biến 4603 tạo ra thông tin cảm biến, mà là thông tin được thu nhận bởi bộ cảm biến. Đầu cuối cảm biến 4603 là đầu cuối được bố trí bộ cảm biến hoặc camera. Ví dụ, đầu cuối cảm biến 4603 là vật di động, chẳng hạn như xe ô tô, vật thể bay, chẳng hạn như máy bay, đầu cuối di động, hoặc camera.

Thông tin cảm biến mà có thể được tạo ra bởi đầu cuối cảm biến 4603 bao gồm (1) khoảng cách giữa đầu cuối cảm biến 4603 và đối tượng hoặc năng suất phản xạ của đối tượng được thu nhận bởi LiDAR, rađa sóng milimet, hoặc bộ cảm biến hồng ngoài hoặc (2) khoảng cách giữa camera và đối tượng hoặc năng suất phản xạ của đối tượng được thu nhận bởi các hình ảnh camera một mắt hoặc hình ảnh camera âm thanh nổi chẳng hạn. Thông tin cảm biến có thể bao gồm tư thế, định hướng, con quay hồi chuyển (vận tốc góc), vị trí (độ cao hoặc thông tin GPS), vận tốc, hoặc gia tốc của bộ cảm biến chẳng hạn. Thông tin cảm biến có thể bao gồm nhiệt độ không khí, áp suất không khí, độ ẩm không khí, hoặc từ tính chẳng hạn.

Bộ nối bên ngoài 4604 được thực hiện bởi mạch tích hợp (LSI hoặc IC), bộ tích lũy bên ngoài, truyền thông với máy chủ đám mây qua Internet, hoặc phát rộng chẳng hạn.

Tiếp theo, dữ liệu đám mây điểm sẽ được mô tả. Fig.2 là sơ đồ thể hiện cấu hình của dữ liệu đám mây điểm. Fig.3 là sơ đồ thể hiện ví dụ cấu hình của tệp dữ liệu mô tả thông tin của dữ liệu đám mây điểm.

Dữ liệu đám mây điểm bao gồm dữ liệu trên các điểm. Dữ liệu trên mỗi điểm bao gồm thông tin hình học (các tọa độ ba chiều) và thông tin thuộc tính được kết hợp với thông

tin hình học. Tập hợp của các điểm như vậy được đề cập đến là đám mây điểm. Ví dụ, đám mây điểm chỉ báo hình dạng ba chiều của đối tượng.

Thông tin hình học (vị trí), chẳng hạn như các tọa độ ba chiều, có thể được đề cập đến là hình học. Dữ liệu trên mỗi điểm có thể bao gồm thông tin thuộc tính (attribute) về các loại của các thuộc tính. Loại của thuộc tính là màu sắc hoặc năng suất phản xạ chẳng hạn.

Một đoạn của thông tin thuộc tính có thể được kết hợp với một đoạn của thông tin hình học, hoặc thông tin thuộc tính về các loại khác nhau của các thuộc tính có thể được kết hợp với một đoạn của thông tin hình học. Theo cách khác các đoạn thông tin thuộc tính về cùng loại của thuộc tính có thể được kết hợp với một đoạn của thông tin hình học.

Ví dụ cấu hình của tệp dữ liệu được thể hiện trên Fig.3 là ví dụ trong đó thông tin hình học và thông tin thuộc tính được kết hợp với nhau theo mối tương quan một-một, và thông tin hình học và thông tin thuộc tính về N điểm tạo nên dữ liệu đám mây điểm được thể hiện.

Thông tin hình học là thông tin trên ba trục, cụ thể là, trục x, trục y, và trục z chẳng hạn. Thông tin thuộc tính là thông tin màu sắc RGB chẳng hạn. Tệp dữ liệu trình diễn là tệp PLY chẳng hạn.

Tiếp theo, các loại của dữ liệu đám mây điểm sẽ được mô tả. Fig.4 là sơ đồ thể hiện các loại của dữ liệu đám mây điểm. Như được thể hiện trên Fig.4, dữ liệu đám mây điểm bao gồm vật thể tĩnh và vật thể động.

Vật thể tĩnh là dữ liệu đám mây điểm ba chiều ở thời gian tùy ý (điểm thời gian). vật thể động là dữ liệu đám mây điểm ba chiều mà thay đổi theo thời gian. Trong phần dưới đây, dữ liệu đám mây điểm ba chiều được kết hợp với điểm thời gian sẽ được đề cập đến là khung PCC hoặc khung.

Đối tượng có thể là đám mây điểm mà khoảng của nó bị giới hạn ở một vài mức nhất định, chẳng hạn như dữ liệu viđeo thông thường, hoặc có thể là đám mây điểm lớn mà khoảng của nó không bị giới hạn, chẳng hạn như thông tin bản đồ.

Có dữ liệu đám mây điểm có các mật độ thay đổi. Có thể có dữ liệu đám mây điểm thưa thớt và dữ liệu đám mây điểm dày đặc.

Trong phần dưới đây, mỗi bộ xử lý sẽ được mô tả chi tiết. Thông tin cảm biến được thu nhận bởi các phương tiện khác nhau, bao gồm bộ cảm biến khoảng cách chẳng hạn như

LIDAR hoặc bộ dò tìm khoảng, camera âm thanh nổi, hoặc sự kết hợp của các camera một mắt. Bộ tạo dữ liệu đám mây điểm 4618 tạo ra dữ liệu đám mây điểm dựa vào thông tin cảm biến được thu nhận bởi bộ thu nhận thông tin cảm biến 4617. Bộ tạo dữ liệu đám mây điểm 4618 tạo ra thông tin hình học như dữ liệu đám mây điểm, và bổ sung thông tin thuộc tính được kết hợp với thông tin hình học tới thông tin hình học.

Khi tạo ra thông tin hình học hoặc bổ sung thông tin thuộc tính, bộ tạo dữ liệu đám mây điểm 4618 có thể xử lý dữ liệu đám mây điểm. Ví dụ, bộ tạo dữ liệu đám mây điểm 4618 có thể làm giảm lượng dữ liệu bằng cách bỏ qua đám mây điểm mà vị trí của nó trùng khớp với vị trí của đám mây điểm khác. Bộ tạo dữ liệu đám mây điểm 4618 có thể cũng chuyển đổi thông tin hình học (chẳng hạn như vị trí chuẩn hóa, dịch vị, hoặc quay) hoặc hoàn lại thông tin thuộc tính.

Lưu ý rằng, mặc dù Fig.1 thể hiện hệ thống tạo dữ liệu đám mây điểm 4611 như được bao gồm trong hệ thống mã hóa dữ liệu ba chiều 4601, hệ thống tạo dữ liệu đám mây điểm 4611 có thể được bố trị một cách độc lập phía ngoài hệ thống mã hóa dữ liệu ba chiều 4601.

Bộ mã hóa 4613 tạo ra dữ liệu được mã hóa bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa được định rõ trước. Nói chung, có hai loại của các phương pháp mã hóa được mô tả dưới đây. Một là phương pháp mã hóa nhờ sử dụng thông tin hình học, mà sẽ được đề cập đến là phương pháp mã hóa thứ nhất, dưới đây. Còn lại là phương pháp mã hóa nhờ sử dụng cođec viđeo, mà sẽ được đề cập đến là phương pháp mã hóa thứ hai dưới đây.

Bộ giải mã 4624 giải mã dữ liệu được mã hóa thành dữ liệu đám mây điểm nhờ sử dụng phương pháp mã hóa được định rõ trước.

Bộ đa hợp 4614 tạo ra dữ liệu được đa hợp bằng cách đa hợp dữ liệu được mã hóa theo phương pháp đa hợp hiện thời. Dữ liệu được đa hợp được tạo ra được truyền hoặc được tích lũy. Bộ đa hợp 4614 đa hợp không chỉ dữ liệu được mã hóa PCC mà còn phương tiện khác, chẳng hạn như viđeo, auđio, các tiêu đề phụ, ứng dụng, hoặc tệp, hoặc thông tin thời gian tham chiếu. Bộ đa hợp 4614 có thể còn đa hợp thông tin thuộc tính được kết hợp với thông tin cảm biến hoặc dữ liệu đám mây điểm.

Các sơ đồ đa hợp hoặc các định dạng tệp bao gồm ISOBMFF, MPEG-DASH, mà là sơ đồ truyền dựa vào các hệ thống ISOBMFF, MMT, MPEG-2 TS, hoặc RMP chẳng hạn.

Bộ giải đa hợp 4623 trích dữ liệu được mã hóa PCC, các phương tiện khác, thông tin thời gian và tương tự từ dữ liệu được đa hợp.

Bộ phận đầu vào/đầu ra 4615 truyền dữ liệu được đa hợp theo phương pháp thích hợp cho phương tiện truyền hoặc phương tiện tích lũy, chẳng hạn như phát rộng hoặc truyền thông. Bộ phận đầu vào/đầu ra 4615 có thể truyền thông với thiết bị khác qua Internet hoặc truyền thông với bộ tích lũy, chẳng hạn như máy chủ đám mây.

Đối với giao thức truyền thông, http, ftp, TCP, UDP hoặc tương tự được sử dụng. Sơ đồ truyền thông kéo hoặc sơ đồ truyền thông đẩy có thể được sử dụng.

Việc truyền nối dây hoặc việc truyền không dây có thể được sử dụng. Đối với việc truyền nối dây, Ethernet (nhãn hiệu đã đăng ký), USB, RS-232C, HDMI (nhãn hiệu đã đăng ký), hoặc cáp đồng trục được sử dụng chẳng hạn. Đối với việc truyền không dây, LAN không dây, Wi-Fi (nhãn hiệu đã đăng ký), Bluetooth (nhãn hiệu đã đăng ký), hoặc sóng milimet được sử dụng chẳng hạn.

Đối với sơ đồ phát rộng, DVB-T2, DVB-S2, DVB-C2, ATSC3.0, hoặc ISDB-S3 được sử dụng chẳng hạn.

Fig.5 là sơ đồ thể hiện cấu hình của bộ mã hóa thứ nhất 4630, mà là ví dụ về bộ mã hóa 4613 mà thực hiện việc mã hóa theo phương pháp mã hóa thứ nhất. Fig.6 là sơ đồ khối thể hiện bộ mã hóa thứ nhất 4630. Bộ mã hóa thứ nhất 4630 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ nhất. Bộ mã hóa thứ nhất 4630 bao gồm bộ mã hóa thông tin hình học 4631, bộ mã hóa thông tin thuộc tính 4632, bộ mã hóa thông tin bổ sung 4633, và bộ đa hợp 4634.

Bộ mã hóa thứ nhất 4630 khác biệt ở chỗ bằng cách thực hiện việc mã hóa bằng cách lưu ý đến cấu trúc ba chiều. Bộ mã hóa thứ nhất 4630 còn khác biệt ở chỗ bộ mã hóa thông tin thuộc tính 4632 thực hiện việc mã hóa nhờ sử dụng thông tin được thu nhận từ bộ mã hóa thông tin hình học 4631. Phương pháp mã hóa thứ nhất được đề cập đến cũng là PCC trên cơ sở hình học (geometry-based PCC, viết tắt là GPCC).

Dữ liệu đám mây điểm là dữ liệu đám mây điểm PCC giống như tệp PLY hoặc dữ liệu đám mây điểm PCC được tạo ra từ thông tin cảm biến, và bao gồm thông tin hình học (vị trí), thông tin thuộc tính (attribute), và thông tin bổ sung khác (siêu dữ liệu). Thông tin hình học được đưa vào tới bộ mã hóa thông tin hình học 4631, thông tin thuộc tính được đưa vào tới bộ mã hóa thông tin thuộc tính 4632, và thông tin bổ sung được đưa vào tới bộ

mã hóa thông tin bổ sung 4633.

Bộ mã hóa thông tin hình học 4631 tạo ra thông tin hình học được mã hóa (hình học được nén), mà là dữ liệu được mã hóa, bằng cách mã hóa thông tin hình học. Ví dụ, bộ mã hóa thông tin hình học 4631 mã hóa thông tin hình học nhờ sử dụng cấu trúc cây N-ary, chẳng hạn như cây bát phân (octree). Cụ thể là, trong trường hợp cây bát phân, không gian hiện thời được chia thành tám nút (các không gian con), thông tin 8 bit (mã cư trú) mà chỉ báo xem mỗi nút bao gồm đám mây điểm hay không được tạo ra. Nút bao gồm đám mây điểm còn được chia thành tám nút, và thông tin 8 bit mà chỉ báo xem mỗi trong số tám nút bao gồm đám mây điểm hay không được tạo ra. Quy trình này được lặp lại đến khi mức được xác định trước được đạt đến hoặc số của các đám mây điểm được bao gồm trong mỗi nút trở nên bằng hoặc nhỏ hơn ngưỡng.

Bộ mã hóa thông tin thuộc tính 4632 tạo ra thông tin thuộc tính được mã hóa (thuộc tính được nén), mà là dữ liệu được mã hóa, bằng cách mã hóa thông tin thuộc tính nhờ sử dụng cấu hình thông tin được tạo ra bởi bộ mã hóa thông tin hình học 4631. Ví dụ, bộ mã hóa thông tin thuộc tính 4632 xác định điểm tham chiếu (nút tham chiếu) mà được đề cập đến trong mã hóa điểm hiện thời (nút hiện thời) được xử lý dựa vào cây bát phân cấu trúc được tạo ra bởi bộ mã hóa thông tin hình học 4631. Ví dụ, bộ mã hóa thông tin thuộc tính 4632 đề cập đến nút mà nút cha của nó trong cây bát phân là giống như nút cha của nút hiện thời, của các nút ngoại vi hoặc các nút lân cận. Lưu ý rằng phương pháp xác định mối tương quan tham chiếu là không giới hạn ở phương pháp này.

Quy trình mã hóa thông tin thuộc tính có thể bao gồm ít nhất một trong số quy trình lượng tử hóa, quy trình dự đoán, và quy trình mã hóa số học. Trong trường hợp này, "đề cập đến" có nghĩa là việc sử dụng nút tham chiếu để tính trị số được dự đoán của thông tin thuộc tính hoặc sử dụng trạng thái của nút tham chiếu (thông tin cư trú mà chỉ báo xem nút tham chiếu bao gồm đám mây điểm hay không chẳng hạn) để xác định thông số của việc mã hóa. Ví dụ, thông số của mã hóa là thông số lượng tử hóa trong quy trình lượng tử hóa hoặc ngữ cảnh hoặc tương tự trong việc mã hóa số học.

Bộ mã hóa thông tin bổ sung 4633 tạo ra thông tin bổ sung được mã hóa (siêu dữ liệu được nén), mà là dữ liệu được mã hóa, bằng cách mã hóa dữ liệu có thể nén được của thông tin bổ sung.

Bộ đa hợp 4634 tạo ra dòng được mã hóa (dòng được nén), mà là dữ liệu được mã hóa, bằng cách đa hợp thông tin hình học được mã hóa, thông tin thuộc tính được mã hóa,

thông tin bổ sung được mã hóa, và thông tin bổ sung khác. Dòng được mã hóa được tạo ra được đưa ra tới bộ xử lý trong lớp hệ thống (không được thể hiện).

Tiếp theo, bộ giải mã thứ nhất 4640, mà là ví dụ về bộ giải mã 4624 mà thực hiện việc giải mã theo phương pháp mã hóa thứ nhất, sẽ được mô tả. Fig.7 là sơ đồ thể hiện cấu hình của bộ giải mã thứ nhất 4640. Fig.8 là sơ đồ khối thể hiện bộ giải mã thứ nhất 4640. Bộ giải mã thứ nhất 4640 tạo ra dữ liệu đám mây điểm bằng cách giải mã dữ liệu được mã hóa (dòng được mã hóa) được mã hóa theo phương pháp mã hóa thứ nhất theo phương pháp mã hóa thứ nhất. Bộ giải mã thứ nhất 4640 bao gồm bộ giải đa hợp 4641, bộ giải mã thông tin hình học 4642, bộ giải mã thông tin thuộc tính 4643, và bộ giải mã thông tin bổ sung 4644.

Dòng được mã hóa (dòng được nén), mà là dữ liệu được mã hóa, được đưa vào tới bộ giải mã thứ nhất 4640 từ bộ xử lý trong lớp hệ thống (không được thể hiện).

Bộ giải đa hợp 4641 chia tách thông tin hình học được mã hóa (hình học được nén), thông tin thuộc tính được mã hóa (thuộc tính được nén), thông tin bổ sung được mã hóa (siêu dữ liệu được nén), và thông tin bổ sung khác từ dữ liệu được mã hóa.

Bộ giải mã thông tin hình học 4642 tạo ra thông tin hình học bằng cách giải mã thông tin hình học được mã hóa. Ví dụ, bộ giải mã thông tin hình học 4642 khôi phục thông tin hình học về đám mây điểm được biểu diễn bởi các tọa độ ba chiều từ thông tin hình học được mã hóa được biểu diễn bởi cấu trúc N-ary, chẳng hạn như cây bát phân.

Bộ giải mã thông tin thuộc tính 4643 giải mã thông tin thuộc tính được mã hóa dựa vào cấu hình thông tin được tạo ra bởi bộ giải mã thông tin hình học 4642. Ví dụ, bộ giải mã thông tin thuộc tính 4643 xác định điểm tham chiếu (nút tham chiếu) mà được đề cập đến trong giải mã điểm hiện thời (nút hiện thời) được xử lý dựa vào cây bát phân cấu trúc được tạo ra bởi bộ giải mã thông tin hình học 4642. Ví dụ, bộ giải mã thông tin thuộc tính 4643 đề cập đến nút mà nút cha của nó trong cây bát phân là giống như nút cha của nút hiện thời, của các nút ngoại vi hoặc các nút lân cận. Lưu ý rằng phương pháp xác định mối tương quan tham chiếu là không giới hạn ở phương pháp này.

Quy trình giải mã thông tin thuộc tính có thể bao gồm ít nhất một trong số quy trình lượng tử hóa ngược, quy trình dự đoán, và quy trình giải mã số học. Trong trường hợp này, "đề cập đến" có nghĩa là việc sử dụng nút tham chiếu để tính trị số được dự đoán của thông tin thuộc tính hoặc sử dụng trạng thái của nút tham chiếu (thông tin cư trú mà chỉ báo xem nút tham chiếu bao gồm đám mây điểm hay không chẳng hạn) để xác định

thông số của việc giải mã. Ví dụ, thông số của việc giải mã là thông số lượng tử hóa trong quy trình lượng tử hóa ngược hoặc ngữ cảnh hoặc tương tự trong việc giải mã số học.

Bộ giải mã thông tin bổ sung 4644 tạo ra thông tin bổ sung bằng cách giải mã thông tin bổ sung được mã hóa. Bộ giải mã thứ nhất 4640 sử dụng thông tin bổ sung được yêu cầu cho quy trình giải mã đối với thông tin hình học và thông tin thuộc tính trong việc giải mã, và đưa ra thông tin bổ sung được yêu cầu đối với ứng dụng tới phía ngoài.

Tiếp theo, bộ mã hóa thứ hai 4650, mà là ví dụ về bộ mã hóa 4613 mà thực hiện việc mã hóa theo phương pháp mã hóa thứ hai, sẽ được mô tả. Fig.9 là sơ đồ thể hiện cấu hình của bộ mã hóa thứ hai 4650. Fig.10 là sơ đồ khối thể hiện bộ mã hóa thứ hai 4650.

Bộ mã hóa thứ hai 4650 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ hai. Bộ mã hóa thứ hai 4650 bao gồm bộ tạo thông tin bổ sung 4651, bộ tạo hình ảnh hình học 4652, bộ tạo hình ảnh thuộc tính 4653, bộ mã hóa viđeo 4654, bộ mã hóa thông tin bổ sung 4655, và bộ đa hợp 4656.

Bộ mã hóa thứ hai 4650 khác biệt ở chỗ bằng cách tạo ra hình ảnh hình học và hình ảnh thuộc tính bằng cách chiếu cấu trúc ba chiều lên hình ảnh hai chiều, và mã hóa hình ảnh hình học được tạo ra và hình ảnh thuộc tính trong sơ đồ mã hóa viđeo hiện thời. Phương pháp mã hóa thứ hai được đề cập đến là PCC trên cơ sở viđeo (video-based PCC, viết tắt là VPCC).

Dữ liệu đám mây điểm là dữ liệu đám mây điểm PCC giống như tệp PLY hoặc dữ liệu đám mây điểm PCC được tạo ra từ thông tin cảm biến, và bao gồm thông tin hình học (vị trí), thông tin thuộc tính (thuộc tính), và thông tin bổ sung khác (siêu dữ liệu).

Bộ tạo thông tin bổ sung 4651 tạo ra thông tin bản đồ về các hình ảnh hai chiều bằng cách chiếu cấu trúc ba chiều lên hình ảnh hai chiều.

Bộ tạo hình ảnh hình học 4652 tạo ra hình ảnh hình học dựa vào thông tin hình học và thông tin bản đồ được tạo ra bởi bộ tạo thông tin bổ sung 4651. Hình ảnh hình học là hình ảnh khoảng cách trong đó khoảng cách (độ sâu) được chỉ báo là trị số điểm ảnh chẳng hạn. Hình ảnh khoảng cách có thể là hình ảnh của các đám mây điểm được nhìn từ một điểm ngắm (hình ảnh của các đám mây điểm được chiếu lên một mặt phẳng hai chiều), các hình ảnh của các đám mây điểm được nhìn từ các điểm ngắm, hoặc hình ảnh đơn tích hợp các hình ảnh.

Bộ tạo hình ảnh thuộc tính 4653 tạo ra hình ảnh thuộc tính dựa vào thông tin thuộc tính và thông tin bản đồ được tạo ra bởi bộ tạo thông tin bổ sung 4651. Hình ảnh thuộc tính là hình ảnh trong đó thông tin thuộc tính (màu sắc (RGB) chẳng hạn) được chỉ báo là trị số điểm ảnh chẳng hạn. Hình ảnh có thể là hình ảnh của các đám mây điểm được nhìn từ một điểm ngắm (hình ảnh của các đám mây điểm được chiếu lên một mặt phẳng hai chiều), các hình ảnh của các đám mây điểm được nhìn từ các điểm ngắm, hoặc hình ảnh đơn tích hợp các hình ảnh.

Bộ mã hóa viđeo 4654 tạo ra hình ảnh hình học được mã hóa (hình ảnh hình học được nén) và hình ảnh thuộc tính được mã hóa (hình ảnh thuộc tính được nén), mà là dữ liệu được mã hóa, bằng cách mã hóa hình ảnh hình học và hình ảnh thuộc tính trong sơ đồ mã hóa viđeo. Lưu ý rằng, với sơ đồ mã hóa viđeo, phương pháp mã hóa đã được biết bất kỳ đều có thể được sử dụng. Ví dụ, sơ đồ mã hóa viđeo là AVC hoặc HEVC.

Bộ mã hóa thông tin bổ sung 4655 tạo ra thông tin bổ sung được mã hóa (siêu dữ liệu được nén) bằng cách mã hóa thông tin bổ sung, thông tin bản đồ và tương tự được bao gồm trong dữ liệu đám mây điểm.

Bộ đa hợp 4656 tạo ra dòng được mã hóa (dòng được nén), mà là dữ liệu được mã hóa, bằng cách đa hợp hình ảnh hình học được mã hóa, hình ảnh thuộc tính được mã hóa, thông tin bổ sung được mã hóa, và thông tin bổ sung khác. Dòng được mã hóa được tạo ra được đưa ra tới bộ xử lý trong lớp hệ thống (không được thể hiện).

Tiếp theo, bộ giải mã thứ hai 4660, mà là ví dụ về bộ giải mã 4624 mà thực hiện việc giải mã theo phương pháp mã hóa thứ hai, sẽ được mô tả. Fig.11 là sơ đồ thể hiện cấu hình của bộ giải mã thứ hai 4660. Fig.12 là sơ đồ khối thể hiện bộ giải mã thứ hai 4660. Bộ giải mã thứ hai 4660 tạo ra dữ liệu đám mây điểm bằng cách giải mã dữ liệu được mã hóa (dòng được mã hóa) được mã hóa theo phương pháp mã hóa thứ hai theo phương pháp mã hóa thứ hai. Bộ giải mã thứ hai 4660 bao gồm bộ giải đa hợp 4661, bộ giải mã viđeo 4662, bộ giải mã thông tin bổ sung 4663, bộ tạo thông tin hình học 4664, và bộ tạo thông tin thuộc tính 4665.

Dòng được mã hóa (dòng được nén), mà là dữ liệu được mã hóa, được đưa vào tới bộ giải mã thứ hai 4660 từ bộ xử lý trong lớp hệ thống (không được thể hiện).

Bộ giải đa hợp 4661 chia tách hình ảnh hình học được mã hóa (hình ảnh hình học được nén), hình ảnh thuộc tính được mã hóa (hình ảnh thuộc tính được nén), thông tin bổ sung được mã hóa (siêu dữ liệu được nén), và thông tin bổ sung khác từ dữ liệu được mã

hóa.

Bộ giải mã viđeo 4662 tạo ra hình ảnh hình học và hình ảnh thuộc tính bằng cách giải mã hình ảnh hình học được mã hóa và hình ảnh thuộc tính được mã hóa trong sơ đồ mã hóa viđeo. Lưu ý rằng, với sơ đồ mã hóa viđeo, phương pháp mã hóa đã được biết bất kỳ đều có thể được sử dụng. Ví dụ, sơ đồ mã hóa viđeo là AVC hoặc HEVC.

Bộ giải mã thông tin bổ sung 4663 tạo ra thông tin bổ sung bao gồm thông tin bản đồ hoặc tương tự bằng cách giải mã thông tin bổ sung được mã hóa.

Bộ tạo thông tin hình học 4664 tạo ra thông tin hình học từ hình ảnh hình học và thông tin bản đồ. Bộ tạo thông tin thuộc tính 4665 tạo ra thông tin thuộc tính từ hình ảnh thuộc tính và thông tin bản đồ.

Bộ giải mã thứ hai 4660 sử dụng thông tin bổ sung được yêu cầu dùng để giải mã trong việc giải mã, và đưa ra thông tin bổ sung được yêu cầu đối với ứng dụng tới phía ngoài.

Trong phần dưới đây, vấn đề với sơ đồ mã hóa PCC sẽ được mô tả. Fig.13 là sơ đồ thể hiện ngăn xếp giao thức liên quan đến dữ liệu được mã hóa PCC. Fig.13 thể hiện ví dụ trong đó dữ liệu được mã hóa PCC được đa hợp với dữ liệu phương tiện khác, chẳng hạn như viđeo (HEVC chẳng hạn) hoặc auđio, và được truyền hoặc được tích lũy.

Sơ đồ đa hợp và định dạng tệp có chức năng đa hợp các loại dữ liệu được mã hóa khác nhau và truyền hoặc tích lũy dữ liệu. Để truyền hoặc tích lũy dữ liệu được mã hóa, dữ liệu được mã hóa được chuyển đổi thành định dạng cho sơ đồ đa hợp. Ví dụ, với HEVC, kỹ thuật lưu trữ dữ liệu được mã hóa trong cấu trúc dữ liệu được đề cập đến là bộ phận NAL và lưu trữ bộ phận NAL trong ISOBMFF được quy định.

Hiện tại, phương pháp mã hóa thứ nhất (Codec1) và phương pháp mã hóa thứ hai (Codec2) đang được nghiên cứu là các phương pháp mã hóa đối với dữ liệu đám mây điểm. Tuy nhiên, không có phương pháp được định rõ để lưu trữ cấu hình của dữ liệu được mã hóa và dữ liệu được mã hóa theo định dạng hệ thống. Do vậy, có vấn đề là bộ mã hóa không thể thực hiện quy trình xử lý MUX (đa hợp), việc truyền, hoặc việc tích lũy của dữ liệu.

Lưu ý rằng, trong phần dưới đây, thuật ngữ "phương pháp mã hóa" có nghĩa là phương pháp bất kỳ trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai ngoại trừ phương pháp mã hóa cụ thể được định rõ.

Trong phần dưới đây, cách thức định rõ bộ phận NAL theo phương án này sẽ được

mô tả. Ví dụ, với cođec thông thường, chẳng hạn như HEVC, bộ phận NAL theo một định dạng được định rõ cho một cođec. Tuy nhiên, chưa có phương pháp hỗ trợ định dạng mà bao gồm hai cođec, nghĩa là, phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai, chẳng hạn như PCC (cođec như vậy sẽ được đề cập đến là cođec PCC dưới đây).

Thứ nhất, bộ mã hóa 4670 có các chức năng của cả bộ mã hóa thứ nhất 4630 và bộ mã hóa thứ hai 4650 nêu trên và bộ giải mã 4680 có các chức năng của cả bộ giải mã thứ nhất 4640 và bộ giải mã thứ hai 4660 nêu trên sẽ được mô tả.

Fig.14 là sơ đồ khối thể hiện bộ mã hóa 4670 theo phương án này. Bộ mã hóa 4670 bao gồm bộ mã hóa thứ nhất 4630 và bộ mã hóa thứ hai 4650 nêu trên và bộ đa hợp 4671. Bộ đa hợp 4671 đa hợp dữ liệu được mã hóa được tạo ra bởi bộ mã hóa thứ nhất 4630 và dữ liệu được mã hóa được tạo ra bởi bộ mã hóa thứ hai 4650, và đưa ra dữ liệu được mã hóa nhận được.

Fig.15 là sơ đồ khối thể hiện bộ giải mã 4680 theo phương án này. Bộ giải mã 4680 bao gồm bộ giải mã thứ nhất 4640 và bộ giải mã thứ hai 4660 nêu trên và bộ giải đa hợp 4681. Bộ giải đa hợp 4681 trích dữ liệu được mã hóa được tạo ra nhờ sử dụng phương pháp mã hóa thứ nhất và dữ liệu được mã hóa được tạo ra nhờ sử dụng phương pháp mã hóa thứ hai từ dữ liệu được mã hóa đầu vào. Bộ giải đa hợp 4681 đưa ra dữ liệu được mã hóa được tạo ra nhờ sử dụng phương pháp mã hóa thứ nhất tới bộ giải mã thứ nhất 4640, và đưa ra dữ liệu được mã hóa được tạo ra nhờ sử dụng phương pháp mã hóa thứ hai tới bộ giải mã thứ hai 4660.

Với cấu hình nêu trên, bộ mã hóa 4670 có thể mã hóa dữ liệu đám mây điểm nhờ sử dụng có lựa chọn phương pháp mã hóa thứ nhất hoặc phương pháp mã hóa thứ hai. Bộ giải mã 4680 có thể giải mã dữ liệu được mã hóa được mã hóa nhờ sử dụng phương pháp mã hóa thứ nhất, dữ liệu được mã hóa nhờ sử dụng phương pháp mã hóa thứ hai, và dữ liệu được mã hóa được mã hóa nhờ sử dụng cả phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, bộ mã hóa 4670 có thể thay đổi phương pháp mã hóa (giữa phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai) trên cơ sở dữ liệu đám mây điểm hoặc trên cơ sở khung. Theo cách khác bộ mã hóa 4670 có thể thay đổi phương pháp mã hóa trên cơ sở của bộ phận có thể mã hóa được.

Ví dụ, bộ mã hóa 4670 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bao gồm thông tin nhận dạng đối với cođec PCC.

Bộ giải đa hợp 4681 trong bộ giải mã 4680 nhận dạng dữ liệu bằng cách sử dụng thông tin nhận dạng đối với cođec PCC chẳng hạn. Khi dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất, bộ giải đa hợp 4681 đưa ra dữ liệu tới bộ giải mã thứ nhất 4640. Khi dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai, bộ giải đa hợp 4681 đưa ra dữ liệu tới bộ giải mã thứ hai 4660.

Bộ mã hóa 4670 có thể truyền, như thông tin điều khiển, thông tin chỉ báo xem cả hai phương pháp mã hóa được sử dụng hoặc bất kỳ một trong số các phương pháp mã hóa được sử dụng hay không, cùng với thông tin nhận dạng đối với cođec PCC.

Tiếp theo, quy trình mã hóa theo phương án này sẽ được mô tả. Fig.16 là lưu đồ thể hiện quy trình mã hóa theo phương án này. Việc sử dụng thông tin nhận dạng đối với cođec PCC cho phép quy trình mã hóa sẵn sàng cho các cođec.

Thứ nhất, bộ mã hóa 4670 mã hóa dữ liệu PCC theo cả hai hoặc một trong số các cođec, nghĩa là, phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai (S4681).

Khi cođec được sử dụng là phương pháp mã hóa thứ hai (nếu "phương pháp mã hóa thứ hai" ở S4682), bộ mã hóa 4670 thiết đặt pcc_codec_type trong đoạn đầu bộ phận NAL tới trị số mà chỉ báo rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai (S4683). Bộ mã hóa 4670 sau đó thiết đặt pcc_nal_unit_type trong đoạn đầu bộ phận NAL tới ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ hai (S4684). Bộ mã hóa 4670 sau đó tạo ra bộ phận NAL có đoạn đầu bộ phận NAL được thiết đặt và bao gồm dữ liệu được mã hóa trong tải trọng. Bộ mã hóa 4670 sau đó truyền bộ phận NAL được tạo ra (S4685).

Mặt khác, khi cođec được sử dụng là phương pháp mã hóa thứ nhất (nếu "phương pháp mã hóa thứ nhất" ở S4682), bộ mã hóa 4670 thiết đặt pcc_codec_type trong đoạn đầu bộ phận NAL tới trị số mà chỉ báo rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất (S4686). Bộ mã hóa 4670 sau đó thiết đặt loại bộ phận pcc nal (pcc_nal_unit_type) trong đoạn đầu bộ phận NAL tới ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ nhất (S4687). Bộ mã hóa 4670 sau đó tạo ra bộ phận NAL có đoạn đầu bộ phận NAL được thiết đặt và bao gồm dữ liệu được mã hóa trong tải trọng. Bộ mã hóa 4670 sau đó truyền bộ phận NAL được tạo ra (S4685).

Tiếp theo, quy trình giải mã theo phương án này sẽ được mô tả. Fig.17 là lưu đồ thể hiện quy trình giải mã theo phương án này. Việc sử dụng thông tin nhận dạng đối với

cođec PCC cho phép quy trình giải mã sẵn sàng cho các cođec.

Thứ nhất, bộ giải mã 4680 thu bộ phận NAL (S4691). Ví dụ, bộ phận NAL là bộ phận NAL được tạo ra trong quy trình xử lý bởi bộ mã hóa 4670 nêu trên.

Bộ giải mã 4680 sau đó xác định xem pcc_codec_type trong đoạn đầu bộ phận NAL chỉ báo phương pháp mã hóa thứ nhất hay phương pháp mã hóa thứ hai (S4692).

Khi pcc_codec_type chỉ báo phương pháp mã hóa thứ hai (nếu "phương pháp mã hóa thứ hai" ở S4692), bộ giải mã 4680 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai (S4693). Bộ giải mã 4680 sau đó nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ hai (S4694). Bộ giải mã 4680 sau đó giải mã dữ liệu PCC trong quy trình giải mã đối với phương pháp mã hóa thứ hai (S4695).

Mặt khác, khi pcc_codec_type chỉ báo phương pháp mã hóa thứ nhất (nếu "phương pháp mã hóa thứ nhất" ở S4692), bộ giải mã 4680 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất (S4696). Bộ giải mã 4680 sau đó nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ nhất (S4697). Bộ giải mã 4680 sau đó giải mã dữ liệu PCC trong quy trình giải mã đối với phương pháp mã hóa thứ nhất (S4698).

Như được nêu trên, thiết bị mã hóa dữ liệu ba chiều theo khía cạnh của sáng chế tạo ra dòng được mã hóa bằng cách mã hóa dữ liệu ba chiều (dữ liệu đám mây điểm chẳng hạn), và lưu trữ thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai (thông tin nhận dạng đối với cođec chẳng hạn) trong thông tin điều khiển (tập hợp thông số chẳng hạn) đối với dòng được mã hóa.

Với cấu hình như vậy, thiết bị giải mã dữ liệu ba chiều có thể xác định phương pháp mã hóa được sử dụng cho việc mã hóa từ thông tin được lưu trữ trong thông tin điều khiển, khi giải mã dòng được mã hóa được tạo ra bởi thiết bị mã hóa dữ liệu ba chiều. Do đó, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác dòng được mã hóa ngay cả khi các phương pháp mã hóa được sử dụng.

Dữ liệu ba chiều bao gồm thông tin hình học chẳng hạn. Trong việc mã hóa nêu

trên, thiết bị mã hóa dữ liệu ba chiều mã hóa thông tin hình học. Trong bộ phận lưu trữ nêu trên, thiết bị mã hóa dữ liệu ba chiều lưu trữ thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai trong thông tin điều khiển đối với thông tin hình học.

Dữ liệu ba chiều bao gồm thông tin hình học và thông tin thuộc tính chẳng hạn. Trong việc mã hóa nêu trên, thiết bị mã hóa dữ liệu ba chiều mã hóa thông tin hình học và thông tin thuộc tính. Trong bộ phận lưu trữ nêu trên, thiết bị mã hóa dữ liệu ba chiều lưu trữ thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai trong thông tin điều khiển đối với thông tin hình học, và lưu trữ thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin thuộc tính trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai trong thông tin điều khiển đối với thông tin thuộc tính.

Với cấu hình như vậy, các phương pháp mã hóa khác nhau có thể được sử dụng đối với thông tin hình học và thông tin thuộc tính, và do đó, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, phương pháp mã hóa dữ liệu ba chiều còn bao gồm lưu trữ dòng được mã hóa trong một hoặc nhiều bộ phận (các bộ phận NAL chẳng hạn).

Ví dụ, bộ phận bao gồm thông tin (pcc_nal_unit_type chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có định dạng mà là chung cho phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ một cách độc lập đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, bộ phận bao gồm thông tin (loại bộ phận nal codec 1 (codec1_nal_unit_type) hoặc loại bộ phận nal codec 2 (codec2_nal_unit_type) chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có các hình dạng khác nhau đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ một cách độc lập đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, bộ phận bao gồm thông tin (pcc_nal_unit_type chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có định dạng mà là chung cho phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ chung đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Thiết bị giải mã dữ liệu ba chiều theo phương án này xác định phương pháp mã hóa được sử dụng cho việc mã hóa của dòng được mã hóa được thu nhận bằng cách mã hóa của dữ liệu ba chiều dựa vào thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của dữ liệu ba chiều trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai (thông tin nhận dạng đối với cođec chẳng hạn) được bao gồm trong thông tin điều khiển (tập hợp thông số chẳng hạn) đối với dòng được mã hóa, và giải mã dòng được mã hóa nhờ sử dụng phương pháp mã hóa được xác định.

Với cấu hình như vậy, thiết bị giải mã dữ liệu ba chiều có thể xác định phương pháp mã hóa được sử dụng cho việc mã hóa từ thông tin được lưu trữ trong thông tin điều khiển, khi giải mã dòng được mã hóa. Do đó, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác dòng được mã hóa ngay cả khi các phương pháp mã hóa được sử dụng.

Dữ liệu ba chiều bao gồm thông tin hình học, và dòng được mã hóa bao gồm dữ liệu được mã hóa của thông tin hình học chẳng hạn. Trong việc xác định nêu trên, thiết bị giải mã dữ liệu ba chiều xác định phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học dựa vào thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai được bao gồm trong thông tin điều khiển đối với thông tin hình học được bao gồm trong dòng được mã hóa. Trong việc giải mã nêu trên, thiết bị giải mã dữ liệu ba chiều giải mã dữ liệu được mã hóa của thông tin hình học nhờ sử dụng phương pháp mã hóa được xác định được sử dụng cho việc mã hóa của thông tin hình học.

Dữ liệu ba chiều bao gồm thông tin hình học và thông tin thuộc tính, và dòng được mã hóa bao gồm dữ liệu được mã hóa của thông tin hình học và dữ liệu được mã hóa của thông tin thuộc tính chẳng hạn. Trong việc xác định nêu trên, thiết bị giải mã dữ liệu ba chiều xác định phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học dựa vào thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin hình học trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai được bao gồm trong thông tin điều khiển đối với thông tin hình học được bao gồm trong dòng được mã hóa, và xác định phương pháp mã hóa được sử dụng cho việc mã hóa của thông tin thuộc tính dựa vào thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã

hóa của thông tin thuộc tính trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai được bao gồm trong thông tin điều khiển đối với thông tin thuộc tính được bao gồm trong dòng được mã hóa. Trong việc giải mã nêu trên, thiết bị giải mã dữ liệu ba chiều giải mã dữ liệu được mã hóa của thông tin hình học nhờ sử dụng phương pháp mã hóa được xác định được sử dụng cho việc mã hóa của thông tin hình học, và giải mã dữ liệu được mã hóa của thông tin thuộc tính nhờ sử dụng phương pháp mã hóa được xác định được sử dụng cho việc mã hóa của thông tin thuộc tính.

Với cấu hình như vậy, các phương pháp mã hóa khác nhau có thể được sử dụng đối với thông tin hình học và thông tin thuộc tính, và do đó, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, dòng được mã hóa được lưu trữ trong một hoặc nhiều bộ phận (các bộ phận NAL chẳng hạn), và thiết bị giải mã dữ liệu ba chiều còn thu nhận dòng được mã hóa từ một hoặc nhiều bộ phận.

Ví dụ, bộ phận bao gồm thông tin (pcc_nal_unit_type chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có định dạng mà là chung cho phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ một cách độc lập đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, bộ phận bao gồm thông tin (codec1_nal_unit_type hoặc codec2_nal_unit_type chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có các hình dạng khác nhau đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ một cách độc lập đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, bộ phận bao gồm thông tin (pcc_nal_unit_type chẳng hạn) chỉ báo loại của dữ liệu được bao gồm trong bộ phận mà có định dạng mà là chung cho phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai và được định rõ chung đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Ví dụ, thiết bị giải mã dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Phương án 2

Theo phương án 2, phương pháp lưu trữ bộ phận NAL trong tệp ISOBMFF sẽ được mô tả.

ISOBMFF là chuẩn định dạng tệp được quy định trong ISO/IEC14496-12. ISOBMFF là chuẩn mà không phụ thuộc vào bất kỳ phương tiện nào, và quy định định dạng mà cho phép các loại phương tiện khác nhau, chẳng hạn như viđeo, auđio, và văn bản, được đa hợp và được lưu trữ.

Cấu trúc cơ bản (tệp) của ISOBMFF sẽ được mô tả. Bộ phận cơ bản của ISOBMFF là ô vuông. Ô vuông được tạo nên bởi loại, độ dài, và dữ liệu, và tệp là tập hợp của các loại ô vuông khác nhau.

Fig.18 là sơ đồ thể hiện cấu trúc cơ bản (tệp) của ISOBMFF. Tệp trong ISOBMFF bao gồm các ô vuông, chẳng hạn như ftyp mà chỉ báo nhãn của tệp bởi mã bốn ký tự (four-character code, viết tắt là 4CC), moov mà lưu trữ siêu dữ liệu, chẳng hạn như thông tin điều khiển (thông tin báo hiệu), và mdat mà lưu trữ dữ liệu.

Phương pháp để lưu trữ mỗi phương tiện trong tệp ISOBMFF được quy định một cách riêng biệt. Ví dụ, phương pháp lưu trữ viđeo AVC hoặc viđeo HEVC được quy định trong ISO/IEC14496-15. Ở đây, có thể được dự tính để mở rộng chức năng của ISOBMFF và sử dụng ISOBMFF để tích lũy hoặc truyền dữ liệu được mã hóa PCC. Tuy nhiên, không có quy ước để lưu trữ dữ liệu được mã hóa PCC trong tệp ISOBMFF. Theo phương án này, phương pháp lưu trữ dữ liệu được mã hóa PCC trong tệp ISOBMFF sẽ được mô tả.

Fig.19 là sơ đồ thể hiện ngăn xếp giao thức trong trường hợp trong đó cođec bộ phận NAL PCC chung trong tệp ISOBMFF. Ở đây, cođec bộ phận NAL PCC chung được lưu trữ trong tệp ISOBMFF. Mặc dù bộ phận NAL là chung cho các cođec PCC, phương pháp lưu trữ đối với mỗi cođec (sự thực hiện của Codec1, sự thực hiện của Codec2) được quy định một cách mong muốn, vì các cođec PCC được lưu trữ trong bộ phận NAL.

Tiếp theo, phương pháp lưu trữ bộ phận NAL PCC chung mà hỗ trợ các cođec PCC trong tệp ISOBMFF sẽ được mô tả. Fig.20 là sơ đồ thể hiện ví dụ trong đó bộ phận NAL PCC chung được lưu trữ trong tệp ISOBMFF đối với phương pháp lưu trữ đối với cođec 1 (sự thực hiện của Codec1). Fig.21 là sơ đồ thể hiện ví dụ trong đó bộ phận NAL PCC chung được lưu trữ trong tệp ISOBMFF đối với phương pháp lưu trữ đối với cođec 2 (sự thực hiện của Codec2).

Ở đây, ftyp là thông tin mà là quan trọng đối với việc nhận dạng của định dạng tệp, và ký hiệu nhận dạng khác nhau của ftyp được định rõ đối với mỗi cođec. Khi dữ liệu được mã hóa PCC được mã hóa theo phương pháp mã hóa thứ nhất (sơ đồ mã hóa) được lưu trữ trong tệp, ftyp được thiết đặt tới pcc1. Khi dữ liệu được mã hóa PCC được mã hóa theo

phương pháp mã hóa thứ hai được lưu trữ trong tệp, ftyp được thiết đặt tới pcc2.

Ở đây, pcc1 chỉ báo rằng cođec PCC 1 (phương pháp mã hóa thứ nhất) được sử dụng. Pcc2 chỉ báo rằng cođec PCC 2 (phương pháp mã hóa thứ hai) được sử dụng. Nghĩa là, pcc1 và pcc2 chỉ báo rằng dữ liệu là PCC (được mã hóa dữ liệu ba chiều (dữ liệu đám mây điểm)), và chỉ báo cođec PCC (phương pháp mã hóa thứ nhất hoặc phương pháp mã hóa thứ hai).

Trong phần dưới đây, phương pháp lưu trữ bộ phận NAL trong tệp ISOBMFF sẽ được mô tả. Bộ đa hợp phân tích đoạn đầu bộ phận NAL, và mô tả pcc1 trong ftyp của ISOBMFF nếu pcc_codec_type = Codec1.

Bộ đa hợp phân tích đoạn đầu bộ phận NAL, và mô tả pcc2 trong ftyp của ISOBMFF nếu pcc_codec_type = Codec2.

Nếu pcc_nal_unit_type là siêu dữ liệu, bộ đa hợp lưu trữ bộ phận NAL trong moov hoặc mdat theo cách thức được xác định trước chẳng hạn. Nếu pcc_nal_unit_type là dữ liệu, bộ đa hợp lưu trữ bộ phận NAL trong moov hoặc mdat theo cách thức được xác định trước chẳng hạn.

Ví dụ, bộ đa hợp có thể lưu trữ kích thước bộ phận NAL trong bộ phận NAL, như với HEVC.

Theo phương pháp lưu trữ này, bộ giải đa hợp (lớp hệ thống) có thể xác định xem dữ liệu được mã hóa PCC được mã hóa theo phương pháp mã hóa thứ nhất hay phương pháp mã hóa thứ hai nhờ phân tích ftyp được bao gồm trong tệp. Hơn nữa, như được nêu trên, bằng cách xác định xem dữ liệu được mã hóa PCC được mã hóa theo phương pháp mã hóa thứ nhất hay phương pháp mã hóa thứ hai, dữ liệu được mã hóa được mã hóa trong bất kỳ một trong số các phương pháp mã hóa có thể được trích từ dữ liệu bao gồm cả dữ liệu được mã hóa được mã hóa theo các phương pháp mã hóa. Do đó, khi truyền dữ liệu được mã hóa, lượng dữ liệu được truyền có thể được giảm. Ngoài ra, theo phương pháp lưu trữ này, các định dạng dữ liệu (tệp) khác nhau không cần được thiết đặt đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai, và định dạng dữ liệu chung có thể được sử dụng đối với phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Lưu ý rằng, khi thông tin nhận dạng đối với cođec, chẳng hạn như ftyp của ISOBMFF, được chỉ báo trong siêu dữ liệu của lớp hệ thống, bộ đa hợp có thể lưu trữ bộ phận NAL mà không có pcc_nal_unit_type trong tệp ISOBMFF.

Tiếp theo, các cấu hình và các thao tác của bộ đa hợp của hệ thống mã hóa dữ liệu ba chiều (thiết bị mã hóa dữ liệu ba chiều) theo phương án này và bộ giải đa hợp của hệ thống giải mã dữ liệu ba chiều (thiết bị giải mã dữ liệu ba chiều) theo phương án này sẽ được mô tả.

Fig.22 là sơ đồ thể hiện cấu hình của bộ đa hợp thứ nhất 4710. Bộ đa hợp thứ nhất 4710 bao gồm bộ chuyển đổi tệp 4711 mà tạo ra dữ liệu được đa hợp (tệp) nhờ lưu trữ dữ liệu được mã hóa được tạo ra bởi bộ mã hóa thứ nhất 4630 và thông tin điều khiển (bộ phận NAL) trong tệp ISOBMFF. Bộ đa hợp thứ nhất 4710 được bao gồm trong bộ đa hợp 4614 được thể hiện trên Fig.1 chẳng hạn.

Fig.23 là sơ đồ thể hiện cấu hình của bộ giải đa hợp thứ nhất 4720. Bộ giải đa hợp thứ nhất 4720 bao gồm bộ chuyển đổi ngược tệp 4721 mà thu nhận dữ liệu được mã hóa và thông tin điều khiển (bộ phận NAL) từ dữ liệu được đa hợp (tệp) và đưa ra dữ liệu được mã hóa được thu nhận và thông tin điều khiển tới bộ giải mã thứ nhất 4640. Bộ giải đa hợp thứ nhất 4720 được bao gồm trong bộ giải đa hợp 4623 được thể hiện trên Fig.1 chẳng hạn.

Fig.24 là sơ đồ thể hiện cấu hình của bộ đa hợp thứ hai 4730. Bộ đa hợp thứ hai 4730 bao gồm bộ chuyển đổi tệp 4731 mà tạo ra dữ liệu được đa hợp (tệp) nhờ lưu trữ dữ liệu được mã hóa được tạo ra bởi bộ mã hóa thứ hai 4650 và thông tin điều khiển (bộ phận NAL) trong tệp ISOBMFF. Bộ đa hợp thứ hai 4730 được bao gồm trong bộ đa hợp 4614 được thể hiện trên Fig.1 chẳng hạn.

Fig.25 là sơ đồ thể hiện cấu hình của bộ giải đa hợp thứ hai 4740. Bộ giải đa hợp thứ hai 4740 bao gồm bộ chuyển đổi ngược tệp 4741 mà thu nhận dữ liệu được mã hóa và thông tin điều khiển (bộ phận NAL) từ dữ liệu được đa hợp (tệp) và đưa ra dữ liệu được mã hóa được thu nhận và thông tin điều khiển tới bộ giải mã thứ hai 4660. Bộ giải đa hợp thứ hai 4740 được bao gồm trong bộ giải đa hợp 4623 được thể hiện trên Fig.1 chẳng hạn.

Fig.26 là lưu đồ thể hiện quy trình đa hợp bởi bộ đa hợp thứ nhất 4710. Thứ nhất, bộ đa hợp thứ nhất 4710 phân tích pcc_codec_type trong đoạn đầu bộ phận NAL, nhờ đó xác định xem cođec được sử dụng là phương pháp mã hóa thứ nhất hay phương pháp mã hóa thứ hai (S4701).

Khi pcc_codec_type biểu diễn phương pháp mã hóa thứ hai (nếu "phương pháp mã hóa thứ hai" ở S4702), bộ đa hợp thứ nhất 4710 không xử lý bộ phận NAL (S4703).

Mặt khác, khi pcc_codec_type biểu diễn phương pháp mã hóa thứ nhất (nếu

"phương pháp mã hóa thứ nhất" ở S4702), bộ đa hợp thứ nhất 4710 mô tả pcc1 trong ftyp (S4704). Nghĩa là, bộ đa hợp thứ nhất 4710 mô tả thông tin chỉ báo rằng dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất được lưu trữ trong tệp trong ftyp.

Bộ đa hợp thứ nhất 4710 sau đó phân tích pcc_nal_unit_type trong đoạn đầu bộ phận NAL, và lưu trữ dữ liệu trong ô vuông (moov hoặc mdat chẳng hạn) theo cách thức được xác định trước thích hợp cho loại dữ liệu được biểu diễn bởi pcc_nal_unit_type (S4705). Bộ đa hợp thứ nhất 4710 sau đó tạo ra tệp ISOBMFF bao gồm ftyp nêu trên và ô vuông nêu trên (S4706).

Fig.27 là lưu đồ thể hiện quy trình đa hợp bởi bộ đa hợp thứ hai 4730. Thứ nhất, bộ đa hợp thứ hai 4730 phân tích pcc_codec_type trong đoạn đầu bộ phận NAL, nhờ đó xác định xem cođec được sử dụng là phương pháp mã hóa thứ nhất hay phương pháp mã hóa thứ hai (S4711).

Khi pcc_codec_type biểu diễn phương pháp mã hóa thứ hai (nếu "phương pháp mã hóa thứ hai" ở S4712), bộ đa hợp thứ hai 4730 mô tả pcc2 trong ftyp (S4713). Nghĩa là, bộ đa hợp thứ hai 4730 mô tả thông tin chỉ báo rằng dữ liệu được mã hóa theo phương pháp mã hóa thứ hai được lưu trữ trong tệp trong ftyp.

Bộ đa hợp thứ hai 4730 sau đó phân tích pcc_nal_unit_type trong đoạn đầu bộ phận NAL, và lưu trữ dữ liệu trong ô vuông (moov hoặc mdat chẳng hạn) theo cách thức được xác định trước thích hợp cho loại dữ liệu được biểu diễn bởi pcc_nal_unit_type (S4714). Bộ đa hợp thứ hai 4730 sau đó tạo ra tệp ISOBMFF bao gồm ftyp nêu trên và ô vuông nêu trên (S4715).

Mặt khác, khi pcc_codec_type biểu diễn phương pháp mã hóa thứ nhất (nếu "phương pháp mã hóa thứ nhất" ở S4712), bộ đa hợp thứ hai 4730 không xử lý bộ phận NAL (S4716).

Lưu ý rằng quy trình nêu trên là ví dụ trong đó dữ liệu PCC được mã hóa trong bất kỳ một trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai. Bộ đa hợp thứ nhất 4710 và bộ đa hợp thứ hai 4730 lưu trữ bộ phận NAL được yêu cầu trong tệp nhờ nhận dạng loại cođec của bộ phận NAL. Lưu ý rằng, khi thông tin nhận dạng đối với cođec PCC được bao gồm ở vị trí khác với đoạn đầu bộ phận NAL, bộ đa hợp thứ nhất 4710 và bộ đa hợp thứ hai 4730 có thể nhận dạng loại cođec (phương pháp mã hóa thứ nhất hoặc phương pháp mã hóa thứ hai) dựa vào thông tin nhận dạng đối với cođec PCC được bao gồm ở vị trí khác với đoạn đầu bộ phận NAL ở bước S4701 hoặc S4711.

Khi lưu trữ dữ liệu trong tệp ở bước S4706 hoặc S4714, bộ đa hợp thứ nhất 4710 và bộ đa hợp thứ hai 4730 có thể lưu trữ dữ liệu trong tệp sau khi xóa pcc_nal_unit_type khỏi đoạn đầu bộ phận NAL.

Fig.28 là lưu đồ thể hiện quy trình được thực hiện bởi bộ giải đa hợp thứ nhất 4720 và bộ giải mã thứ nhất 4640. Thứ nhất, bộ giải đa hợp thứ nhất 4720 phân tích ftyp trong tệp ISOBMFF (S4721). Khi codec được biểu diễn bởi ftyp là phương pháp mã hóa thứ hai (pcc2) (nếu "phương pháp mã hóa thứ hai" ở S4722), bộ giải đa hợp thứ nhất 4720 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai (S4723). Bộ giải đa hợp thứ nhất 4720 cũng truyền kết quả xác định tới bộ giải mã thứ nhất 4640. Bộ giải mã thứ nhất 4640 không xử lý bộ phận NAL (S4724).

Mặt khác, khi codec được biểu diễn bởi ftyp là phương pháp mã hóa thứ nhất (pcc1) (nếu "phương pháp mã hóa thứ nhất" ở S4722), bộ giải đa hợp thứ nhất 4720 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất (S4725). Bộ giải đa hợp thứ nhất 4720 cũng truyền kết quả xác định tới bộ giải mã thứ nhất 4640.

Bộ giải mã thứ nhất 4640 nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ nhất (S4726). Bộ giải mã thứ nhất 4640 sau đó giải mã PCC dữ liệu bằng cách sử dụng quy trình giải mã đối với phương pháp mã hóa thứ nhất (S4727).

Fig.29 là lưu đồ thể hiện quy trình được thực hiện bởi bộ giải đa hợp thứ hai 4740 và bộ giải mã thứ hai 4660. Thứ nhất, bộ giải đa hợp thứ hai 4740 phân tích ftyp trong tệp ISOBMFF (S4731). Khi codec được biểu diễn bởi ftyp là phương pháp mã hóa thứ hai (pcc2) (nếu "phương pháp mã hóa thứ hai" ở S4732), bộ giải đa hợp thứ hai 4740 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai (S4733). Bộ giải đa hợp thứ hai 4740 cũng truyền kết quả xác định tới bộ giải mã thứ hai 4660.

Bộ giải mã thứ hai 4660 nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ hai (S4734). Bộ giải mã thứ hai 4660 sau đó giải mã PCC dữ liệu bằng cách sử dụng quy trình giải mã đối với phương pháp mã hóa thứ hai (S4735).

Mặt khác, khi cođec được biểu diễn bởi ftyp là phương pháp mã hóa thứ nhất (pcc1) (nếu "phương pháp mã hóa thứ nhất" ở S4732), bộ giải đa hợp thứ hai 4740 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất (S4736). Bộ giải đa hợp thứ hai 4740 cũng truyền kết quả xác định tới bộ giải mã thứ hai 4660. Bộ giải mã thứ hai 4660 không xử lý bộ phận NAL (S4737).

Như được nêu trên, ví dụ, vì loại cođec của bộ phận NAL được nhận dạng trong bộ giải đa hợp thứ nhất 4720 hoặc bộ giải đa hợp thứ hai 4740, loại cođec có thể được nhận dạng ở tầng sớm. Hơn nữa, bộ phận NAL được yêu cầu có thể được đưa vào tới bộ giải mã thứ nhất 4640 hoặc bộ giải mã thứ hai 4660, và bộ phận NAL không cần đến có thể được loại bỏ. Trong trường hợp này, quy trình của bộ giải mã thứ nhất 4640 hoặc bộ giải mã thứ hai 4660 phân tích thông tin nhận dạng đối với cođec có thể là không cần thiết. Lưu ý rằng quy trình tham chiếu lại tới loại bộ phận NAL và phân tích thông tin nhận dạng đối với cođec có thể được thực hiện bởi bộ giải mã thứ nhất 4640 hoặc bộ giải mã thứ hai 4660.

Hơn nữa, nếu pcc_nal_unit_type được xóa khỏi đoạn đầu bộ phận NAL bởi bộ đa hợp thứ nhất 4710 hoặc bộ đa hợp thứ hai 4730, bộ giải đa hợp thứ nhất 4720 hoặc bộ giải đa hợp thứ hai 4740 có thể đưa ra bộ phận NAL tới bộ giải mã thứ nhất 4640 hoặc bộ giải mã thứ hai 4660 sau khi bổ sung pcc_nal_unit_type tới bộ phận NAL.

Phương án 3

Theo phương án 3, bộ đa hợp và bộ giải đa hợp mà tương ứng với bộ mã hóa 4670 và bộ giải mã 4680 sẵn sàng cho các cođec nêu trên liên quan đến phương án 1 sẽ được mô tả. Fig.30 là sơ đồ thể hiện các cấu hình của bộ mã hóa 4670 và bộ đa hợp thứ ba 4750 theo phương án này.

Bộ mã hóa 4670 mã hóa dữ liệu đám mây điểm theo cả hai hoặc một trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai. Bộ mã hóa 4670 có thể thay đổi phương pháp mã hóa (giữa phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai) trên cơ sở dữ liệu đám mây điểm hoặc trên cơ sở khung. Theo cách khác bộ mã hóa 4670 có thể thay đổi phương pháp mã hóa trên cơ sở của bộ phận có thể mã hóa được.

Bộ mã hóa 4670 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bao gồm thông tin nhận dạng đối với cođec PCC.

Bộ đa hợp thứ ba 4750 bao gồm bộ chuyển đổi tệp 4751. Bộ chuyển đổi tệp 4751

chuyển đổi bộ phận NAL được đưa ra từ bộ mã hóa 4670 thành tệp dữ liệu PCC. Bộ chuyển đổi tệp 4751 phân tích thông tin nhận dạng cođec được bao gồm trong đoạn đầu bộ phận NAL, và xác định xem dữ liệu được mã hóa PCC là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất, dữ liệu được mã hóa theo phương pháp mã hóa thứ hai, hoặc dữ liệu được mã hóa theo cả hai phương pháp mã hóa hay không. Bộ chuyển đổi tệp 4751 mô tả tên nhãn mà cho phép việc nhận dạng cođec trong ftyp. Ví dụ, khi chỉ báo dữ liệu được mã hóa theo cả hai phương pháp mã hóa, pcc3 được mô tả trong ftyp.

Lưu ý rằng, khi bộ mã hóa 4670 mô tả thông tin nhận dạng cođec PCC ở vị trí khác với bộ phận NAL, bộ chuyển đổi tệp 4751 có thể xác định cođec PCC (phương pháp mã hóa) dựa vào thông tin nhận dạng.

Fig.31 là sơ đồ thể hiện các cấu hình của bộ giải đa hợp thứ ba 4760 và bộ giải mã 4680 theo phương án này.

Bộ giải đa hợp thứ ba 4760 bao gồm bộ chuyển đổi ngược tệp 4761. Bộ chuyển đổi ngược tệp 4761 phân tích ftyp được bao gồm trong tệp, và xác định xem dữ liệu được mã hóa PCC là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất, dữ liệu được mã hóa theo phương pháp mã hóa thứ hai, hoặc dữ liệu được mã hóa theo cả hai phương pháp mã hóa hay không.

Khi dữ liệu được mã hóa PCC là dữ liệu được mã hóa trong bất kỳ một trong số các phương pháp mã hóa, dữ liệu được đưa vào tới bộ giải mã thích hợp trong số bộ giải mã thứ nhất 4640 và bộ giải mã thứ hai 4660, và không được đưa vào tới bộ giải mã khác. Khi dữ liệu được mã hóa PCC là dữ liệu được mã hóa theo cả hai phương pháp mã hóa, dữ liệu được đưa vào tới bộ giải mã 4680 sẵn sàng cho cả các phương pháp mã hóa.

Bộ giải mã 4680 giải mã dữ liệu được mã hóa PCC theo cả hai hoặc một trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai.

Fig.32 là lưu đồ thể hiện quy trình được thực hiện bởi bộ đa hợp thứ ba 4750 theo phương án này.

Thứ nhất, bộ đa hợp thứ ba 4750 phân tích pcc_codec_type trong đoạn đầu bộ phận NAL, nhờ đó xác định xem (các) cođec được sử dụng là phương pháp mã hóa thứ nhất, phương pháp mã hóa thứ hai, hoặc cả phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai hay không (S4741).

Khi phương pháp mã hóa thứ hai được sử dụng (nếu Có ở S4742 và "phương pháp

mã hóa thứ hai" ở S4743), bộ đa hợp thứ ba 4750 mô tả pcc2 trong ftyp (S4744). Nghĩa là, bộ đa hợp thứ ba 4750 mô tả thông tin chỉ báo rằng dữ liệu được mã hóa theo phương pháp mã hóa thứ hai được lưu trữ trong tệp trong ftyp.

Bộ đa hợp thứ ba 4750 sau đó phân tích pcc_nal_unit_type trong đoạn đầu bộ phận NAL, và lưu trữ dữ liệu trong ô vuông (moov hoặc mdat chẳng hạn) theo cách thức được xác định trước thích hợp cho loại dữ liệu được biểu diễn bởi pcc_nal_unit_type (S4745). Bộ đa hợp thứ ba 4750 sau đó tạo ra tệp ISOBMFF bao gồm ftyp nêu trên và ô vuông nêu trên (S4746).

Khi phương pháp mã hóa thứ nhất được sử dụng (nếu Có ở S4742 và "phương pháp mã hóa thứ nhất" ở S4743), bộ đa hợp thứ ba 4750 mô tả pcc1 trong ftyp (S4747). Nghĩa là, bộ đa hợp thứ ba 4750 mô tả thông tin chỉ báo rằng dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất được lưu trữ trong tệp trong ftyp.

Bộ đa hợp thứ ba 4750 sau đó phân tích pcc_nal_unit_type trong đoạn đầu bộ phận NAL, và lưu trữ dữ liệu trong ô vuông (moov hoặc mdat chẳng hạn) theo cách thức được xác định trước thích hợp cho loại dữ liệu được biểu diễn bởi pcc_nal_unit_type (S4748). Bộ đa hợp thứ ba 4750 sau đó tạo ra tệp ISOBMFF bao gồm ftyp nêu trên và ô vuông nêu trên (S4746).

Khi cả phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai được sử dụng (nếu Không ở S4742), bộ đa hợp thứ ba 4750 mô tả pcc3 trong ftyp (S4749). Nghĩa là, bộ đa hợp thứ ba 4750 mô tả thông tin chỉ báo rằng dữ liệu được mã hóa theo cả hai phương pháp mã hóa được lưu trữ trong tệp trong ftyp.

Bộ đa hợp thứ ba 4750 sau đó phân tích pcc_nal_unit_type trong đoạn đầu bộ phận NAL, và lưu trữ dữ liệu trong ô vuông (moov hoặc mdat chẳng hạn) theo cách thức được xác định trước thích hợp cho loại dữ liệu được biểu diễn bởi pcc_nal_unit_type (S4750). Bộ đa hợp thứ ba 4750 sau đó tạo ra tệp ISOBMFF bao gồm ftyp nêu trên và ô vuông nêu trên (S4746).

Fig.33 là lưu đồ thể hiện quy trình được thực hiện bởi bộ giải đa hợp thứ ba 4760 và bộ giải mã 4680. Thứ nhất, bộ giải đa hợp thứ ba 4760 phân tích ftyp được bao gồm trong tệp ISOBMFF (S4761). Khi codec được biểu diễn bởi ftyp là phương pháp mã hóa thứ hai (pcc2) (nếu Có ở S4762 và "phương pháp mã hóa thứ hai" ở S4763), bộ giải đa hợp thứ ba 4760 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai (S4764). Bộ giải đa hợp thứ ba 4760

cũng truyền kết quả xác định tới bộ giải mã 4680.

Bộ giải mã 4680 nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ hai (S4765). Bộ giải mã 4680 sau đó giải mã PCC dữ liệu bằng cách sử dụng quy trình giải mã đối với phương pháp mã hóa thứ hai (S4766).

Khi codec được biểu diễn bởi ftyp là phương pháp mã hóa thứ nhất (pcc1) (nếu Có ở S4762 và "phương pháp mã hóa thứ nhất" ở S4763), bộ giải đa hợp thứ ba 4760 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất (S4767). Bộ giải đa hợp thứ ba 4760 cũng truyền kết quả xác định tới bộ giải mã 4680.

Bộ giải mã 4680 nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với phương pháp mã hóa thứ nhất (S4768). Bộ giải mã 4680 sau đó giải mã PCC dữ liệu bằng cách sử dụng quy trình giải mã đối với phương pháp mã hóa thứ nhất (S4769).

Khi ftyp chỉ báo rằng cả các phương pháp mã hóa được sử dụng (pcc3) (nếu Không ở S4762), bộ giải đa hợp thứ ba 4760 xác định rằng dữ liệu được bao gồm trong tải trọng của bộ phận NAL là dữ liệu được mã hóa theo cả phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai (S4770). Bộ giải đa hợp thứ ba 4760 cũng truyền kết quả xác định tới bộ giải mã 4680.

Bộ giải mã 4680 nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type trong đoạn đầu bộ phận NAL là ký hiệu nhận dạng của bộ phận NAL đối với các codec được mô tả theo pcc_codec_type (S4771). Bộ giải mã 4680 sau đó giải mã PCC dữ liệu bằng cách sử dụng các quy trình giải mã cho cả hai phương pháp mã hóa (S4772). Nghĩa là, bộ giải mã 4680 giải mã dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất nhờ sử dụng quy trình giải mã đối với phương pháp mã hóa thứ nhất, và giải mã dữ liệu được mã hóa theo phương pháp mã hóa thứ hai nhờ sử dụng quy trình giải mã đối với phương pháp mã hóa thứ hai.

Trong phần dưới đây, các biến thể của phương án này sẽ được mô tả. Như các loại nhãn được biểu diễn bởi ftyp, các loại được mô tả dưới đây có thể được chỉ báo bởi thông tin nhận dạng. Hơn nữa, sự kết hợp của các loại được mô tả dưới đây có thể cũng được chỉ báo bởi thông tin nhận dạng.

Thông tin nhận dạng có thể chỉ báo xem đối tượng dữ liệu gốc vẫn là PCC được mã hóa là đám mây điểm mà khoảng của nó bị giới hạn hoặc đám mây điểm lớn mà khoảng của nó không bị giới hạn hay không, chẳng hạn như thông tin bản đồ.

Thông tin nhận dạng có thể chỉ báo xem dữ liệu gốc vẫn là PCC được mã hóa là vật thể tĩnh hay vật thể động.

Như được nêu trên, thông tin nhận dạng có thể chỉ báo xem dữ liệu được mã hóa PCC là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất hay dữ liệu được mã hóa theo phương pháp mã hóa thứ hai.

Thông tin nhận dạng có thể chỉ báo thuật toán được sử dụng trong việc mã hóa PCC. Ở đây, "thuật toán" có nghĩa là phương pháp mã hóa mà có thể được sử dụng theo phương pháp mã hóa thứ nhất hoặc phương pháp mã hóa thứ hai chẳng hạn.

Thông tin nhận dạng có thể chỉ báo sự khác nhau giữa các phương pháp lưu trữ dữ liệu được mã hóa PCC trong tệp ISOBMFF. Ví dụ, thông tin nhận dạng có thể chỉ báo xem phương pháp lưu trữ được sử dụng là phương pháp lưu trữ cho việc tích lũy hoặc phương pháp lưu trữ cho việc truyền theo thời gian thực, chẳng hạn như tạo dòng động, hay không.

Mặc dù ví dụ trong đó ISOBMFF được sử dụng như định dạng tệp đã được mô tả theo các phương án 2 và 3, các định dạng khác có thể cũng được sử dụng. Ví dụ, phương pháp theo phương án này có thể cũng được sử dụng khi dữ liệu được mã hóa PCC được lưu trữ trong các hệ thống TS MPEG-2, MPEG-DASH, MMT, hoặc RMP.

Mặc dù ví dụ trong đó siêu dữ liệu, chẳng hạn như thông tin nhận dạng, được lưu trữ trong ftyp đã được thể hiện như trên, siêu dữ liệu có thể cũng được lưu trữ ở vị trí khác với ftyp. Ví dụ, siêu dữ liệu có thể được lưu trữ trong moov.

Như được nêu trên, thiết bị lưu trữ dữ liệu ba chiều (hoặc dữ liệu ba chiều đa hợp thiết bị hoặc thiết bị mã hóa dữ liệu ba chiều) thực hiện quy trình được thể hiện trên Fig.34.

Thứ nhất, thiết bị lưu trữ dữ liệu ba chiều (mà bao gồm bộ đa hợp thứ nhất 4710, bộ đa hợp thứ hai 4730 hoặc bộ đa hợp thứ ba 4750 chẳng hạn) thu được một hoặc nhiều bộ phận (các bộ phận NAL chẳng hạn) mà lưu trữ dòng được mã hóa, mà là dữ liệu đám mây điểm được mã hóa (S4781). Thiết bị lưu trữ dữ liệu ba chiều sau đó lưu trữ một hoặc nhiều bộ phận trong tệp (tệp ISOBMFF chẳng hạn) (S4782). Trong bộ phận lưu trữ (S4782), thiết bị lưu trữ dữ liệu ba chiều cũng lưu trữ thông tin chỉ báo rằng dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa (pcc1, pcc2, hoặc pcc3 chẳng hạn) trong

thông tin điều khiển (ftyp chẳng hạn) (được đề cập đến cũng như thông tin báo hiệu) đối với tệp.

Với cấu hình như vậy, thiết bị mà xử lý tệp được tạo ra bởi thiết bị lưu trữ dữ liệu ba chiều có thể nhanh chóng xác định xem dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa hay không bằng cách tham chiếu tới thông tin điều khiển đối với tệp. Do đó, lượng xử lý của thiết bị có thể được giảm, hoặc tốc độ xử lý của thiết bị có thể được tăng lên.

Ví dụ, thông tin chỉ báo phương pháp mã hóa được sử dụng cho việc mã hóa của dữ liệu đám mây điểm trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai. Lưu ý rằng thực tế là dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa và phương pháp mã hóa được sử dụng cho việc mã hóa của dữ liệu đám mây điểm trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai có thể được chỉ báo bởi đoạn đơn của thông tin hoặc các đoạn khác nhau của thông tin.

Với cấu hình như vậy, thiết bị mà xử lý tệp được tạo ra bởi thiết bị lưu trữ dữ liệu ba chiều có thể nhanh chóng xác định cođec được sử dụng đối với dữ liệu được lưu trữ trong tệp bằng cách tham chiếu tới thông tin điều khiển đối với tệp. Do đó, lượng xử lý của thiết bị có thể được giảm, hoặc tốc độ xử lý của thiết bị có thể được tăng lên.

Ví dụ, phương pháp mã hóa thứ nhất là phương pháp (GPCC) mà mã hóa thông tin hình học mà biểu diễn vị trí của dữ liệu đám mây điểm như cây N-ary (N biểu diễn số nguyên bằng hoặc lớn hơn 2) và mã hóa thông tin thuộc tính nhờ sử dụng thông tin hình học, và phương pháp mã hóa thứ hai là phương pháp (VPCC) mà tạo ra hình ảnh hai chiều từ dữ liệu đám mây điểm và mã hóa hình ảnh hai chiều trong phương pháp mã hóa viđeo.

Ví dụ, tệp nêu trên là phù hợp với ISOBMFF (định dạng tệp phương tiện trên cơ sở ISO).

Ví dụ, thiết bị lưu trữ dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Như được nêu trên, thiết bị thu nhận dữ liệu ba chiều (hoặc thiết bị giải đa hợp dữ liệu ba chiều hoặc thiết bị giải mã dữ liệu ba chiều) thực hiện quy trình được thể hiện trên Fig.35.

Thiết bị thu nhận dữ liệu ba chiều (mà bao gồm bộ giải đa hợp thứ nhất 4720, bộ giải đa hợp thứ hai 4740, hoặc bộ giải đa hợp thứ ba 4760 chẳng hạn) thu được tệp (tệp

ISOBMFF chẳng hạn) mà lưu trữ một hoặc nhiều bộ phận (các bộ phận NAL chẳng hạn) mà lưu trữ dòng được mã hóa, mà là dữ liệu đám mây điểm được mã hóa (S4791). Thiết bị thu nhận dữ liệu ba chiều thu được một hoặc nhiều bộ phận từ tệp (S4792). Thông tin điều khiển (ftyp chẳng hạn) đối với tệp bao gồm thông tin chỉ báo rằng dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa (pcc1, pcc2, hoặc pcc3 chẳng hạn).

Ví dụ, thiết bị thu nhận dữ liệu ba chiều xác định xem dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa hay không bằng cách tham chiếu tới thông tin. Khi thiết bị thu nhận dữ liệu ba chiều xác định rằng dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa, thiết bị thu nhận dữ liệu ba chiều tạo ra dữ liệu đám mây điểm bằng cách giải mã dữ liệu đám mây điểm được mã hóa được bao gồm trong một hoặc nhiều bộ phận. Theo cách khác khi thiết bị thu nhận dữ liệu ba chiều xác định rằng dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa, thiết bị thu nhận dữ liệu ba chiều đưa ra thông tin chỉ báo rằng dữ liệu được bao gồm trong một hoặc nhiều bộ phận là dữ liệu đám mây điểm được mã hóa tới bộ xử lý ở tầng tiếp theo (bộ giải mã thứ nhất 4640, bộ giải mã thứ hai 4660, hoặc bộ giải mã 4680 chẳng hạn) (hoặc thông báo bộ xử lý ở tầng tiếp theo rằng dữ liệu được bao gồm trong một hoặc nhiều bộ phận là dữ liệu đám mây điểm được mã hóa).

Với cấu hình như vậy, thiết bị thu nhận dữ liệu ba chiều có thể nhanh chóng xác định xem dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa hay không bằng cách tham chiếu tới thông tin điều khiển đối với tệp. Do đó, lượng xử lý của thiết bị thu nhận dữ liệu ba chiều hoặc thiết bị ở tầng tiếp theo có thể được giảm, hoặc tốc độ xử lý của thiết bị thu nhận dữ liệu ba chiều hoặc thiết bị ở tầng tiếp theo có thể được tăng lên.

Ví dụ, thông tin biểu diễn phương pháp mã hóa được sử dụng cho việc mã hóa trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai. Lưu ý rằng thực tế là dữ liệu được lưu trữ trong tệp là dữ liệu đám mây điểm được mã hóa và phương pháp mã hóa được sử dụng cho việc mã hóa của dữ liệu đám mây điểm trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai có thể được chỉ báo bởi đoạn đơn của thông tin hoặc các đoạn khác nhau của thông tin.

Với cấu hình như vậy, thiết bị thu nhận dữ liệu ba chiều có thể nhanh chóng xác định cođec được sử dụng đối với dữ liệu được lưu trữ trong tệp bằng cách tham chiếu tới thông tin điều khiển đối với tệp. Do đó, lượng xử lý của thiết bị thu nhận dữ liệu ba chiều hoặc thiết bị ở tầng tiếp theo có thể được giảm, hoặc tốc độ xử lý của thiết bị thu nhận dữ

liệu ba chiều hoặc thiết bị ở tầng tiếp theo có thể được tăng lên.

Ví dụ, dựa vào thông tin, thiết bị thu nhận dữ liệu ba chiều thu được dữ liệu được mã hóa trong bất kỳ một trong số phương pháp mã hóa thứ nhất và phương pháp mã hóa thứ hai từ dữ liệu đám mây điểm được mã hóa bao gồm dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất và dữ liệu được mã hóa theo phương pháp mã hóa thứ hai.

Ví dụ, phương pháp mã hóa thứ nhất là phương pháp (GPCC) mà mã hóa thông tin hình học mà biểu diễn vị trí của dữ liệu đám mây điểm như cây N-ary (N biểu diễn số nguyên bằng hoặc lớn hơn 2) và mã hóa thông tin thuộc tính nhờ sử dụng thông tin hình học, và phương pháp mã hóa thứ hai là phương pháp (VPCC) mà tạo ra hình ảnh hai chiều từ dữ liệu đám mây điểm và mã hóa hình ảnh hai chiều trong phương pháp mã hóa viđeo.

Ví dụ, tệp nêu trên là phù hợp với ISOBMFF (định dạng tệp phương tiện trên cơ sở ISO).

Ví dụ, thiết bị thu nhận dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Phương án 4

Theo phương án 4, các loại của dữ liệu được mã hóa (thông tin hình học (hình học), thông tin thuộc tính (attribute), và thông tin bổ sung (siêu dữ liệu)) được tạo ra bởi bộ mã hóa thứ nhất 4630 hoặc bộ mã hóa thứ hai 4650 nêu trên, phương pháp tạo ra thông tin bổ sung (siêu dữ liệu), và quy trình đa hợp trong bộ đa hợp sẽ được mô tả. Thông tin bổ sung (siêu dữ liệu) có thể được đề cập đến là tập hợp thông số hoặc thông tin điều khiển (thông tin báo hiệu).

Theo phương án này, vật thể động (dữ liệu đám mây điểm ba chiều mà thay đổi theo thời gian) nêu trên dựa vào Fig.4 sẽ được mô tả chẳng hạn. Tuy nhiên, phương pháp như nhau có thể cũng được sử dụng cho vật thể tĩnh (dữ liệu đám mây điểm ba chiều được kết hợp với điểm thời gian tùy ý).

Fig.36 là sơ đồ thể hiện các cấu hình của bộ mã hóa 4801 và bộ đa hợp 4802 trong thiết bị mã hóa dữ liệu ba chiều theo phương án này. Bộ mã hóa 4801 tương ứng với bộ mã hóa thứ nhất 4630 hoặc bộ mã hóa thứ hai 4650 nêu trên chẳng hạn. Bộ đa hợp 4802 tương ứng với bộ đa hợp 4634 hoặc 4656 nêu trên.

Bộ mã hóa 4801 mã hóa các khung PCC (nén đám mây điểm) của dữ liệu đám mây điểm để tạo ra các đoạn dữ liệu được mã hóa (nhiều dữ liệu được nén) của thông tin

hình học, thông tin thuộc tính, và thông tin bổ sung.

Bộ đa hợp 4802 tích hợp các loại của dữ liệu (thông tin hình học, thông tin thuộc tính, và thông tin bổ sung) trong bộ phận NAL, nhờ đó chuyển đổi dữ liệu thành cấu hình dữ liệu mà lấy việc truy cập dữ liệu trong thiết bị giải mã đưa vào xem xét.

Fig.37 là sơ đồ thể hiện ví dụ cấu hình của dữ liệu được mã hóa được tạo ra bởi bộ mã hóa 4801. Các mũi tên trên hình vẽ chỉ báo sự phụ thuộc được bao gồm trong việc giải mã của dữ liệu được mã hóa. Nguồn của mũi tên tùy thuộc vào dữ liệu của đích mũi tên. Nghĩa là, thiết bị giải mã giải mã dữ liệu của đích mũi tên, và giải mã dữ liệu của nguồn mũi tên nhờ sử dụng dữ liệu được giải mã. Nói cách khác, "thực thể thứ nhất tùy thuộc vào thực thể thứ hai" có nghĩa là dữ liệu của thực thể thứ hai được đề cập đến (được sử dụng) trong việc xử lý (mã hóa, giải mã, hoặc tương tự) của dữ liệu của thực thể thứ nhất.

Thứ nhất, quy trình tạo ra dữ liệu được mã hóa của thông tin hình học sẽ được mô tả. Bộ mã hóa 4801 mã hóa thông tin hình học của mỗi khung để tạo ra dữ liệu hình học được mã hóa (dữ liệu hình học được nén) đối với mỗi khung. Dữ liệu hình học được mã hóa được biểu thị bởi G(i). i biểu thị số khung hoặc điểm thời gian của khung chẳng hạn.

Hơn nữa, bộ mã hóa 4801 tạo ra tập hợp thông số hình học (GPS(i)) đối với mỗi khung. Tập hợp thông số hình học bao gồm thông số mà có thể được sử dụng để giải mã của dữ liệu hình học được mã hóa. Dữ liệu hình học được mã hóa đối với mỗi khung tùy thuộc vào tập hợp thông số hình học được kết hợp.

Dữ liệu hình học được mã hóa được tạo nên bởi các khung được định rõ là chuỗi hình học. Bộ mã hóa 4801 tạo ra chuỗi tập hợp thông số hình học (cũng được đề cập đến là chuỗi hình học PS hoặc hình học SPS) mà lưu trữ thông số được nhờ sử dụng chung cho quy trình giải mã đối với các khung theo chuỗi hình học. Chuỗi hình học tùy thuộc vào hình học SPS.

Tiếp theo, quy trình tạo ra dữ liệu được mã hóa của thông tin thuộc tính sẽ được mô tả. Bộ mã hóa 4801 mã hóa thông tin thuộc tính của mỗi khung để tạo ra dữ liệu thuộc tính được mã hóa (dữ liệu thuộc tính được nén) đối với mỗi khung. Dữ liệu thuộc tính được mã hóa được biểu thị bởi A(i). Fig.37 thể hiện ví dụ trong đó có thuộc tính X và thuộc tính Y, và dữ liệu thuộc tính được mã hóa đối với thuộc tính X được biểu thị bởi AX(i), và dữ liệu thuộc tính được mã hóa đối với thuộc tính Y được biểu thị bởi AY(i).

Hơn nữa, bộ mã hóa 4801 tạo ra tập hợp thông số thuộc tính (APS(i)) đối với mỗi

khung. Tập hợp thông số thuộc tính đối với thuộc tính X được biểu thị bởi AXPS(i), và tập hợp thông số thuộc tính đối với thuộc tính Y được biểu thị bởi AYPS(i). Tập hợp thông số thuộc tính bao gồm thông số mà có thể được sử dụng để giải mã của thông tin thuộc tính được mã hóa. Dữ liệu thuộc tính được mã hóa tùy thuộc vào tập hợp thông số thuộc tính được kết hợp.

Dữ liệu thuộc tính được mã hóa được tạo nên bởi các khung được định rõ là chuỗi thuộc tính. Bộ mã hóa 4801 tạo ra tập hợp thông số chuỗi thuộc tính (cũng được đề cập đến là chuỗi thuộc tính PS hoặc thuộc tính SPS) mà lưu trữ thông số được nhờ sử dụng chung cho quy trình giải mã đối với các khung theo chuỗi thuộc tính. Chuỗi thuộc tính tùy thuộc vào thuộc tính SPS.

Theo phương pháp mã hóa thứ nhất, dữ liệu thuộc tính được mã hóa tùy thuộc vào dữ liệu hình học được mã hóa.

Fig.37 thể hiện ví dụ trong đó có hai loại của thông tin thuộc tính (thuộc tính X và thuộc tính Y). Khi có hai loại của thông tin thuộc tính, ví dụ, hai bộ mã hóa tạo ra dữ liệu và siêu dữ liệu cho hai loại của thông tin thuộc tính. Ví dụ, chuỗi thuộc tính được định rõ đối với mỗi loại của thông tin thuộc tính, và thuộc tính SPS được tạo ra đối với mỗi loại của thông tin thuộc tính.

Lưu ý rằng, mặc dù Fig.37 thể hiện ví dụ trong đó có một loại của thông tin hình học, và có hai loại của thông tin thuộc tính, sáng chế là không giới hạn ở đó. Có thể có một loại của thông tin thuộc tính hoặc ba hoặc nhiều hơn ba loại của thông tin thuộc tính. Trong các trường hợp như vậy, dữ liệu được mã hóa có thể được tạo ra theo cách thức như nhau. Nếu dữ liệu đám mây điểm không có thông tin thuộc tính, có thể không có thông tin thuộc tính. Trong trường hợp như vậy, bộ mã hóa 4801 không phải tạo ra tập hợp thông số được kết hợp với thông tin thuộc tính.

Tiếp theo, quy trình tạo ra dữ liệu được mã hóa của thông tin bổ sung (siêu dữ liệu) sẽ được mô tả. Bộ mã hóa 4801 tạo ra dòng PCC PS (cũng được đề cập đến là dòng PCC PS hoặc dòng PS), mà là tập hợp thông số đối với toàn bộ dòng PCC. Bộ mã hóa 4801 lưu trữ thông số mà có thể được nhờ sử dụng chung đối với quy trình giải mã cho một hoặc nhiều chuỗi hình học và một hoặc nhiều chuỗi thuộc tính trong dòng PS. Ví dụ, dòng PS bao gồm thông tin nhận dạng chỉ báo cođec đối với dữ liệu đám mây điểm và thông tin chỉ báo thuật toán được sử dụng cho việc mã hóa chẳng hạn. Chuỗi hình học và chuỗi thuộc tính phụ thuộc vào dòng PS.

Tiếp theo, bộ phận truy cập và GOF sẽ được mô tả. Theo phương án này, các khái niệm của bộ phận truy cập (access unit, viết tắt là AU) và nhóm của các khung (group of frame, viết tắt là GOF) mới được đưa vào.

Bộ phận truy cập là bộ phận cơ bản để truy cập dữ liệu trong việc giải mã, và được tạo nên bởi một hoặc nhiều đoạn dữ liệu và một hoặc nhiều đoạn siêu dữ liệu. Ví dụ, bộ phận truy cập được tạo nên bởi thông tin hình học và một hoặc nhiều đoạn thông tin thuộc tính được kết hợp với cùng điểm thời gian. GOF là bộ phận truy cập ngẫu nhiên, và được tạo nên bởi một hoặc nhiều bộ phận truy cập.

Bộ mã hóa 4801 tạo ra đoạn đầu bộ phận truy cập (đoạn đầu AU) như thông tin nhận dạng chỉ báo trên đỉnh của bộ phận truy cập. Bộ mã hóa 4801 lưu trữ thông số liên quan đến bộ phận truy cập trong đoạn đầu bộ phận truy cập. Ví dụ, đoạn đầu bộ phận truy cập bao gồm cấu hình của hoặc thông tin về dữ liệu được mã hóa được bao gồm trong bộ phận truy cập. Đoạn đầu bộ phận truy cập còn bao gồm thông số được nhờ sử dụng chung cho dữ liệu được bao gồm trong bộ phận truy cập, chẳng hạn như thông số liên quan đến giải mã của dữ liệu được mã hóa.

Lưu ý rằng bộ mã hóa 4801 có thể tạo ra ký hiệu phân cách bộ phận truy cập mà không bao gồm thông số liên quan đến bộ phận truy cập, thay vì đoạn đầu bộ phận truy cập. Ký hiệu phân cách bộ phận truy cập được sử dụng như thông tin nhận dạng chỉ báo trên đỉnh của bộ phận truy cập. Thiết bị giải mã nhận dạng trên đỉnh của bộ phận truy cập bằng cách phát hiện đoạn đầu bộ phận truy cập hoặc ký hiệu phân cách bộ phận truy cập.

Tiếp theo, việc tạo ra của thông tin nhận dạng đối với đỉnh của GOF sẽ được mô tả. Như thông tin nhận dạng chỉ báo trên đỉnh của GOF, bộ mã hóa 4801 tạo ra đoạn đầu GOF. Bộ mã hóa 4801 lưu trữ thông số liên quan đến GOF trong đoạn đầu GOF. Ví dụ, đoạn đầu GOF bao gồm cấu hình của hoặc thông tin về dữ liệu được mã hóa được bao gồm trong GOF. Đoạn đầu GOF còn bao gồm thông số được nhờ sử dụng chung cho dữ liệu được bao gồm trong GOF, chẳng hạn như thông số liên quan đến giải mã của dữ liệu được mã hóa.

Lưu ý rằng bộ mã hóa 4801 có thể tạo ra ký hiệu phân cách GOF mà không bao gồm thông số liên quan đến GOF, thay vì đoạn đầu GOF. Ký hiệu phân cách GOF được sử dụng như thông tin nhận dạng chỉ báo trên đỉnh của GOF. Thiết bị giải mã nhận dạng trên đỉnh của GOF bằng cách phát hiện đoạn đầu GOF hoặc ký hiệu phân cách GOF.

Trong dữ liệu được mã hóa PCC, bộ phận truy cập được định rõ là bộ phận khung

PCC chẳng hạn. Thiết bị giải mã truy cập khung PCC dựa vào thông tin nhận dạng đối với đỉnh của bộ phận truy cập.

Ví dụ, GOF được định rõ là một bộ phận truy cập ngẫu nhiên. Thiết bị giải mã truy cập bộ phận truy cập ngẫu nhiên dựa vào thông tin nhận dạng đối với đỉnh của GOF. Ví dụ, nếu các khung PCC là độc lập với nhau và có thể được giải mã riêng biệt, khung PCC có thể được định rõ là bộ phận truy cập ngẫu nhiên.

Lưu ý rằng hai hoặc nhiều hơn hai khung PCC có thể được gán tới một bộ phận truy cập, và các bộ phận truy cập ngẫu nhiên có thể được gán tới một GOF.

Bộ mã hóa 4801 có thể định rõ và tạo ra tập hợp thông số hoặc siêu dữ liệu khác với tập hợp thông số hoặc siêu dữ liệu nêu trên. Ví dụ, bộ mã hóa 4801 có thể tạo ra thông tin nâng cao phụ (supplemental enhancement information, viết tắt là SEI) mà lưu trữ thông số (thông số tùy chọn) mà luôn không được sử dụng để giải mã.

Tiếp theo, cấu hình của dữ liệu được mã hóa và phương pháp lưu trữ dữ liệu được mã hóa trong bộ phận NAL sẽ được mô tả.

Ví dụ, định dạng dữ liệu được định rõ đối với mỗi loại của dữ liệu được mã hóa. Fig.38 là sơ đồ thể hiện ví dụ về dữ liệu được mã hóa và bộ phận NAL.

Ví dụ, như được thể hiện trên Fig.38, dữ liệu được mã hóa bao gồm đoạn đầu và tải trọng. Dữ liệu được mã hóa có thể bao gồm thông tin độ dài chỉ báo độ dài (lượng dữ liệu) của dữ liệu được mã hóa, đoạn đầu, hoặc tải trọng. Dữ liệu được mã hóa có thể không bao gồm đoạn đầu.

Đoạn đầu bao gồm thông tin nhận dạng dùng để nhận dạng dữ liệu chẳng hạn. Thông tin nhận dạng chỉ báo loại dữ liệu hoặc số khung chẳng hạn.

Đoạn đầu bao gồm thông tin nhận dạng chỉ báo mối tương quan tham chiếu chẳng hạn. Thông tin nhận dạng được lưu trữ trong đoạn đầu khi có sự phụ thuộc mối tương quan giữa dữ liệu, ví dụ, và cho phép thực thể đề cập đến thực thể khác. Ví dụ, đoạn đầu của thực thể được đề cập đến bao gồm thông tin nhận dạng dùng để nhận dạng dữ liệu. Đoạn đầu của thực thể tham chiếu bao gồm thông tin nhận dạng chỉ báo thực thể được tham chiếu tới.

Lưu ý rằng, khi thực thể được đề cập đến hoặc thực thể tham chiếu có thể được nhận dạng hoặc được xác định từ thông tin khác, thông tin nhận dạng dùng để nhận dạng dữ liệu hoặc thông tin nhận dạng chỉ báo mối tương quan tham chiếu có thể được bỏ qua.

Bộ đa hợp 4802 lưu trữ dữ liệu được mã hóa trong tải trọng của bộ phận NAL. Đoạn đầu bộ phận NAL bao gồm pcc_nal_unit_type, mà là thông tin nhận dạng đối với dữ liệu được mã hóa. FIG 39 là sơ đồ thể hiện ví dụ ngữ nghĩa của pcc_nal_unit_type.

Như được thể hiện trên Fig.39, khi pcc_codec_type là codec 1 (Codec1: phương pháp mã hóa thứ nhất), các trị số từ 0 đến 10 của pcc_nal_unit_type được gán tới dữ liệu hình học được mã hóa (hình học), dữ liệu thuộc tính X được mã hóa (thuộc tính X (AttributeX)), dữ liệu thuộc tính Y được mã hóa (thuộc tính Y (AttributeY)), hình học PS (Geom. PS), thuộc tính XPS (AttrX. S), thuộc tính YPS (AttrY. PS), hình học SPS (chuỗi hình học PS), thuộc tính X SPS (chuỗi thuộc tính X PS), thuộc tính Y SPS (chuỗi thuộc tính Y PS), đoạn đầu AU (AU header), và đoạn đầu GOF (GOF header) trong codec 1. Các trị số 11 và lớn hơn được dành riêng trong codec 1.

Khi pcc_codec_type là codec 2 (Codec2: phương pháp mã hóa thứ hai), các trị số từ 0 đến 2 của pcc_nal_unit_type được gán tới dữ liệu A (DataA), siêu dữ liệu A (MetaDataA), và siêu dữ liệu B (MetaDataB) trong codec. Các trị số 3 và lớn hơn được dành riêng trong codec 2.

Tiếp theo, thứ tự truyền của dữ liệu sẽ được mô tả. Trong phần dưới đây, các giới hạn về thứ tự truyền của các bộ phận NAL sẽ được mô tả.

Bộ đa hợp 4802 truyền các bộ phận NAL trên cơ sở GOF hoặc trên cơ sở AU. Bộ đa hợp 4802 bố trí đoạn đầu GOF ở đỉnh của GOF, và bố trí đoạn đầu AU ở đỉnh của AU.

Để cho phép thiết bị giải mã giải mã next AU và các AU dưới đây ngay cả khi dữ liệu bị mất bởi vì tổn thất gói hoặc tương tự, bộ đa hợp 4802 có thể bố trí tập hợp thông số chuỗi (sequence parameter set, viết tắt là SPS) trong mỗi AU.

Khi có mối tương quan phụ thuộc dùng để giải mã giữa dữ liệu được mã hóa, thiết bị giải mã giải mã dữ liệu của thực thể được đề cập đến và sau đó giải mã dữ liệu của thực thể tham chiếu. Để cho phép thiết bị giải mã thực hiện việc giải mã theo thứ tự thu mà không có việc sắp xếp lại dữ liệu, bộ đa hợp 4802 đầu tiên truyền dữ liệu của thực thể được tham chiếu tới.

Fig.40 là sơ đồ thể hiện các ví dụ về thứ tự truyền của các bộ phận NAL. Fig.40 thể hiện ba ví dụ, nghĩa là, thứ tự thông tin hình học thứ nhất, thứ tự thông số thứ nhất, và thứ tự dữ liệu được tích hợp.

Thứ tự thông tin hình học thứ nhất của việc truyền là ví dụ trong đó thông tin liên

quan đến thông tin hình học được truyền cùng nhau, và thông tin liên quan đến thông tin thuộc tính được truyền cùng nhau. Trong trường hợp thứ tự truyền này, việc truyền của thông tin liên quan đến thông tin hình học kết thúc sớm hơn so với việc truyền của thông tin liên quan đến thông tin thuộc tính.

Ví dụ, theo thứ tự truyền này được sử dụng, khi thiết bị giải mã không giải mã thông tin thuộc tính, thiết bị giải mã có thể có khả năng có thời gian nghỉ vì thiết bị giải mã có thể bỏ qua việc giải mã của thông tin thuộc tính. Khi thiết bị giải mã được yêu cầu để giải mã thông tin hình học sớm, thiết bị giải mã có thể có khả năng giải mã thông tin hình học sớm hơn vì thiết bị giải mã thu nhận dữ liệu được mã hóa của thông tin hình học sớm hơn.

Lưu ý rằng, mặc dù trên Fig.40 thuộc tính X SPS và thuộc tính Y SPS được tích hợp và được thể hiện như thuộc tính SPS, thuộc tính X SPS và thuộc tính Y SPS có thể được bố trí một cách riêng biệt.

Trong tập hợp thứ tự thông số thứ nhất của việc truyền, tập hợp thông số được truyền đầu tiên, và dữ liệu sau đó được truyền.

Như được nêu trên, đến khi các giới hạn về thứ tự truyền của các bộ phận NAL được đáp ứng, bộ đa hợp 4802 có thể truyền các bộ phận NAL theo thứ tự bất kỳ. Ví dụ, thông tin nhận dạng thứ tự có thể được định rõ, và bộ đa hợp 4802 có thể có chức năng truyền các bộ phận NAL theo các thứ tự. Ví dụ, thông tin nhận dạng thứ tự đối với các bộ phận NAL được lưu trữ trong dòng PS.

Thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã dựa vào thông tin nhận dạng thứ tự. Thiết bị giải mã dữ liệu ba chiều có thể chỉ báo thứ tự mong muốn của việc truyền tới thiết bị mã hóa dữ liệu ba chiều, và thiết bị mã hóa dữ liệu ba chiều (bộ đa hợp 4802) có thể điều khiển thứ tự truyền theo thứ tự được chỉ báo của việc truyền.

Lưu ý rằng bộ đa hợp 4802 có thể tạo ra dữ liệu được mã hóa có các chức năng được hợp nhất với nhau như trong trường hợp thứ tự dữ liệu được tích hợp của việc truyền, đến khi các giới hạn về thứ tự truyền được đáp ứng. Ví dụ, như được thể hiện trên Fig.40, đoạn đầu GOF và đoạn đầu AU có thể được tích hợp, hoặc AXPS và AYPS có thể được tích hợp. Trong trường hợp như vậy, ký hiệu nhận dạng mà chỉ báo dữ liệu có các chức năng được định rõ trong pcc_nal_unit_type.

Trong phần dưới đây, các biến thể của phương án này sẽ được mô tả. Có các mức

của các PS, chẳng hạn như mức khung PS, mức chuỗi PS, và PCC mức chuỗi PS. Miễn là mức chuỗi PCC là mức cao hơn, và mức khung là mức thấp hơn, các thông số có thể được lưu trữ theo cách thức được mô tả dưới đây.

Trị số của PS mặc định được chỉ báo trong PS ở mức cao hơn. Nếu trị số của PS ở mức thấp hơn khác với trị số của PS ở mức cao hơn, trị số của PS được chỉ báo trong PS ở mức thấp hơn. Theo cách khác trị số của PS không được mô tả trong PS ở mức cao hơn nhưng được mô tả trong PS ở mức thấp hơn. Theo cách khác thông tin chỉ báo xem trị số của PS được chỉ báo trong PS ở mức thấp hơn, ở mức cao hơn, hoặc ở cả hai mức được chỉ báo theo cả hai hoặc một trong số PS ở mức thấp hơn và PS ở mức cao hơn hay không. Theo cách khác, PS ở mức thấp hơn có thể được hợp nhất với PS ở mức cao hơn. Nếu PS ở mức thấp hơn và PS ở mức cao hơn chồng lấn với nhau, bộ đa hợp 4802 có thể bỏ qua việc truyền của một trong số các PS.

Lưu ý rằng bộ mã hóa 4801 hoặc bộ đa hợp 4802 có thể chia dữ liệu thành các lát hoặc các mảnh và truyền mỗi trong số các lát hoặc các mảnh được chia như dữ liệu được chia. Dữ liệu được chia bao gồm thông tin dùng để nhận dạng dữ liệu được chia, và thông số được sử dụng để giải mã của dữ liệu được chia được bao gồm trong tập hợp thông số. Trong trường hợp này, ký hiệu nhận dạng mà chỉ báo rằng dữ liệu là dữ liệu liên quan đến mảnh hoặc lát hoặc dữ liệu lưu trữ thông số được định rõ trong pcc_nal_unit_type.

Trong phần dưới đây, quy trình liên quan đến thông tin nhận dạng thứ tự sẽ được mô tả. Fig.41 là lưu đồ thể hiện quy trình được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều (bộ mã hóa 4801 và bộ đa hợp 4802) mà bao gồm thứ tự truyền của các bộ phận NAL.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định thứ tự truyền của các bộ phận NAL (thông tin hình học thứ nhất hoặc tập hợp thông số thứ nhất) (S4801). Ví dụ, thiết bị mã hóa dữ liệu ba chiều xác định thứ tự truyền dựa vào thông số kỹ thuật từ người dùng hoặc thiết bị bên ngoài (thiết bị giải mã dữ liệu ba chiều chẳng hạn).

Nếu thứ tự được xác định của việc truyền là thông tin hình học thứ nhất (nếu "thông tin hình học thứ nhất" ở S4802), thiết bị mã hóa dữ liệu ba chiều thiết đặt thông tin nhận dạng thứ tự được bao gồm trong dòng PS tới thông tin hình học thứ nhất (S4803). Nghĩa là, trong trường hợp này, thông tin nhận dạng thứ tự chỉ báo rằng các bộ phận NAL được truyền theo thứ tự thông tin hình học thứ nhất. Thiết bị mã hóa dữ liệu ba chiều sau đó truyền các bộ phận NAL theo thứ tự thông tin hình học thứ nhất (S4804).

Mặt khác, nếu thứ tự được xác định của việc truyền là tập hợp thông số thứ nhất (nếu "tập hợp thông số thứ nhất" ở S4802), thiết bị mã hóa dữ liệu ba chiều thiết đặt thông tin nhận dạng thứ tự được bao gồm trong dòng PS tới tập hợp thông số thứ nhất (S4805). Nghĩa là, trong trường hợp này, thông tin nhận dạng thứ tự chỉ báo rằng các bộ phận NAL được truyền trong tập hợp thứ tự thông số thứ nhất. Thiết bị mã hóa dữ liệu ba chiều sau đó truyền các bộ phận NAL trong tập hợp thứ tự thông số thứ nhất (S4806).

Fig.42 là lưu đồ thể hiện quy trình được thực hiện bởi thiết bị giải mã dữ liệu ba chiều mà bao gồm thứ tự truyền của các bộ phận NAL. Thứ nhất, thiết bị giải mã dữ liệu ba chiều phân tích thông tin nhận dạng thứ tự được bao gồm trong dòng PS (S4811).

Nếu thứ tự truyền được chỉ báo bởi thông tin nhận dạng thứ tự là thông tin hình học thứ nhất (nếu "thông tin hình học thứ nhất" ở S4812), thiết bị giải mã dữ liệu ba chiều giải mã các bộ phận NAL dựa vào việc xác định rằng thứ tự truyền của các bộ phận NAL là thông tin hình học thứ nhất (S4813).

Mặt khác, nếu thứ tự truyền được chỉ báo bởi thông tin nhận dạng thứ tự là tập hợp thông số thứ nhất (nếu "tập hợp thông số thứ nhất" ở S4812), thiết bị giải mã dữ liệu ba chiều giải mã các bộ phận NAL dựa vào việc xác định rằng thứ tự truyền của các bộ phận NAL là tập hợp thông số thứ nhất (S4814).

Ví dụ, nếu thiết bị giải mã dữ liệu ba chiều không giải mã thông tin thuộc tính, ở bước S4813, thiết bị giải mã dữ liệu ba chiều không thu nhận toàn bộ các bộ phận NAL nhưng có thể thu nhận phần của bộ phận NAL liên quan đến thông tin hình học và giải mã bộ phận NAL được thu nhận để thu nhận thông tin hình học.

Tiếp theo, quy trình liên quan đến việc tạo ra của AU và GOF sẽ được mô tả. Fig.43 là lưu đồ thể hiện quy trình được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều (bộ đa hợp 4802) mà liên quan đến việc tạo ra của AU và GOF trong việc đa hợp của các bộ phận NAL.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định loại của dữ liệu được mã hóa (S4821). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định xem dữ liệu được mã hóa được xử lý là dữ liệu AU thứ nhất, dữ liệu GOF thứ nhất, hoặc dữ liệu khác hay không.

Nếu dữ liệu được mã hóa là dữ liệu GOF thứ nhất (nếu “GOF thứ nhất" ở S4822), thiết bị mã hóa dữ liệu ba chiều tạo ra các bộ phận NAL bằng cách bố trí đoạn đầu GOF và đoạn đầu AU ở đỉnh của dữ liệu được mã hóa thuộc về GOF (S4823).

Nếu dữ liệu được mã hóa là dữ liệu AU thứ nhất (nếu "AU thứ nhất" ở S4822), thiết bị mã hóa dữ liệu ba chiều tạo ra các bộ phận NAL bằng cách bố trí đoạn đầu AU ở đỉnh của dữ liệu được mã hóa thuộc về AU (S4824).

Nếu dữ liệu được mã hóa không là dữ liệu GOF thứ nhất cũng không là dữ liệu AU thứ nhất (nếu "khác với GOF thứ nhất và AU thứ nhất" ở S4822), thiết bị mã hóa dữ liệu ba chiều tạo ra các bộ phận NAL bằng cách bố trí dữ liệu được mã hóa để theo sau đoạn đầu AU của AU trong đó dữ liệu được mã hóa thuộc về (S4825).

Tiếp theo, quy trình liên quan đến việc truy cập tới AU và GOF sẽ được mô tả. Fig.44 là lưu đồ thể hiện quy trình được thực hiện bởi thiết bị giải mã dữ liệu ba chiều mà bao gồm truy cập tới AU và GOF trong việc giải đa hợp của bộ phận NAL.

Thứ nhất, thiết bị giải mã dữ liệu ba chiều xác định loại của dữ liệu được mã hóa được bao gồm trong bộ phận NAL nhờ phân tích nal_unit_type trong bộ phận NAL (S4831). Cụ thể là, thiết bị giải mã dữ liệu ba chiều xác định xem dữ liệu được mã hóa được bao gồm trong bộ phận NAL là dữ liệu AU thứ nhất, dữ liệu GOF thứ nhất, hoặc dữ liệu khác hay không.

Nếu dữ liệu được mã hóa được bao gồm trong bộ phận NAL là dữ liệu GOF thứ nhất (nếu "GOF thứ nhất" ở S4832), thiết bị giải mã dữ liệu ba chiều xác định rằng bộ phận NAL là vị trí bắt đầu của việc truy cập ngẫu nhiên, truy cập bộ phận NAL, và bắt đầu quy trình giải mã (S4833).

Nếu dữ liệu được mã hóa được bao gồm trong bộ phận NAL là dữ liệu AU thứ nhất (nếu "AU thứ nhất" ở S4832), thiết bị giải mã dữ liệu ba chiều xác định rằng bộ phận NAL là AU-thứ nhất, truy cập dữ liệu được bao gồm trong bộ phận NAL, và giải mã AU (S4834).

Nếu dữ liệu được mã hóa được bao gồm trong bộ phận NAL không phải dữ liệu GOF thứ nhất cũng không phải là dữ liệu AU thứ nhất (nếu "khác với GOF thứ nhất và AU thứ nhất" ở S4832), thiết bị giải mã dữ liệu ba chiều không xử lý bộ phận NAL.

Như được nêu trên, thiết bị mã hóa dữ liệu ba chiều thực hiện quy trình được thể hiện trên Fig.45. Thiết bị mã hóa dữ liệu ba chiều mã hóa sắp xếp theo thứ tự thời gian dữ liệu ba chiều (dữ liệu đám mây điểm trên vật thể động chẳng hạn). Dữ liệu ba chiều bao gồm thông tin hình học và thông tin thuộc tính được kết hợp với mỗi điểm thời gian.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều mã hóa thông tin hình học (S4841).

Thiết bị mã hóa dữ liệu ba chiều sau đó mã hóa thông tin thuộc tính được xử lý bằng cách tham chiếu tới thông tin hình học được kết hợp với cùng điểm thời gian như thông tin thuộc tính được xử lý (S4842). Ở đây, như được thể hiện trên Fig.37, thông tin hình học và thông tin thuộc tính được kết hợp với cùng điểm thời gian tạo nên bộ phận truy cập (AU). Nghĩa là, thiết bị mã hóa dữ liệu ba chiều mã hóa thông tin thuộc tính được xử lý bằng cách tham chiếu tới thông tin hình học được bao gồm trong cùng bộ phận truy cập như thông tin thuộc tính được xử lý.

Theo cách này, thiết bị mã hóa dữ liệu ba chiều có thể nắm lợi thế của bộ phận truy cập để tạo thuận lợi cho việc điều khiển của sự tham chiếu trong việc mã hóa. Do đó, thiết bị mã hóa dữ liệu ba chiều có thể làm giảm lượng xử lý của quy trình mã hóa.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều tạo ra dòng bit bao gồm thông tin hình học được mã hóa (dữ liệu hình học được mã hóa), thông tin thuộc tính được mã hóa (dữ liệu thuộc tính được mã hóa), và thông tin chỉ báo thông tin hình học của thực thể được đề cập đến khi việc mã hóa thông tin thuộc tính được xử lý.

Ví dụ, dòng bit bao gồm tập hợp thông số hình học (hình học PS) mà bao gồm thông tin điều khiển đối với thông tin hình học được kết hợp với mỗi điểm thời gian và tập hợp thông số thuộc tính (thuộc tính PS) mà bao gồm thông tin điều khiển đối với thông tin thuộc tính được kết hợp với mỗi điểm thời gian.

Ví dụ, dòng bit bao gồm chuỗi tập hợp thông số hình học (hình học SPS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau và tập hợp thông số chuỗi thuộc tính (thuộc tính SPS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau.

Ví dụ, dòng bit bao gồm dòng tập hợp thông số (dòng PS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau và các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau.

Ví dụ, dòng bit bao gồm đoạn đầu bộ phận truy cập (đoạn đầu AU) mà bao gồm thông tin điều khiển mà là chung trong bộ phận truy cập.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều thực hiện việc mã hóa theo cách sao cho các nhóm của các khung (các GOF) được tạo nên bởi một hoặc nhiều bộ phận truy cập có

thể được giải mã một cách độc lập. Nghĩa là, GOF là bộ phận truy cập ngẫu nhiên.

Ví dụ, dòng bit bao gồm đoạn đầu GOF mà bao gồm thông tin điều khiển mà là chung trong GOF.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Như được nêu trên, thiết bị giải mã dữ liệu ba chiều thực hiện quy trình được thể hiện trên Fig.46. Thiết bị giải mã dữ liệu ba chiều giải mã sắp xếp theo thứ tự thời gian dữ liệu ba chiều (dữ liệu đám mây điểm trên vật thể động chẳng hạn). Dữ liệu ba chiều bao gồm thông tin hình học và thông tin thuộc tính được kết hợp với mỗi điểm thời gian. Thông tin hình học và thông tin thuộc tính được kết hợp với cùng điểm thời gian tạo nên bộ phận truy cập (AU).

Thứ nhất, thiết bị giải mã dữ liệu ba chiều giải mã dòng bit để thu nhận thông tin hình học (S4851). Nghĩa là, thiết bị giải mã dữ liệu ba chiều tạo ra thông tin hình học bằng cách giải mã thông tin hình học được mã hóa (dữ liệu hình học được mã hóa) được bao gồm trong dòng bit.

Thiết bị giải mã dữ liệu ba chiều sau đó giải mã dòng bit để thu nhận thông tin thuộc tính được xử lý bằng cách tham chiếu tới thông tin hình học được kết hợp với cùng điểm thời gian như thông tin thuộc tính được xử lý (S4852). Nghĩa là, thiết bị giải mã dữ liệu ba chiều tạo ra thông tin thuộc tính bằng cách giải mã thông tin thuộc tính được mã hóa (dữ liệu thuộc tính được mã hóa) được bao gồm trong dòng bit. Trong quy trình này, thiết bị giải mã dữ liệu ba chiều đề cập đến thông tin hình học được giải mã được bao gồm trong bộ phận truy cập như thông tin thuộc tính.

Theo cách này, thiết bị giải mã dữ liệu ba chiều có thể nắm lợi thế của bộ phận truy cập để tạo thuận lợi cho việc điều khiển của sự tham chiếu trong việc giải mã. Do đó, thiết bị giải mã dữ liệu ba chiều có thể làm giảm lượng xử lý của quy trình giải mã.

Ví dụ, thiết bị giải mã dữ liệu ba chiều thu nhận, từ dòng bit, thông tin chỉ báo thông tin hình học của thực thể được đề cập đến khi giải mã thông tin thuộc tính được xử lý, và giải mã thông tin thuộc tính được xử lý bằng cách tham chiếu tới thông tin hình học của thực thể được đề cập đến được chỉ báo bởi thông tin được thu nhận.

Ví dụ, dòng bit bao gồm tập hợp thông số hình học (hình học PS) mà bao gồm thông tin điều khiển đối với thông tin hình học được kết hợp với mỗi điểm thời gian và tập

hợp thông số thuộc tính (thuộc tính PS) mà bao gồm thông tin điều khiển đối với thông tin thuộc tính được kết hợp với mỗi điểm thời gian. Nghĩa là, thiết bị giải mã dữ liệu ba chiều sử dụng thông tin điều khiển được bao gồm trong tập hợp thông số hình học được kết hợp với điểm thời gian được nhắm tới việc xử lý để giải mã thông tin hình học được kết hợp với điểm thời gian được nhắm tới việc xử lý, và sử dụng thông tin điều khiển được bao gồm trong tập hợp thông số thuộc tính được kết hợp với điểm thời gian được nhắm tới việc xử lý để giải mã thông tin thuộc tính được kết hợp với điểm thời gian được nhắm tới việc xử lý.

Ví dụ, dòng bit bao gồm chuỗi tập hợp thông số hình học (hình học SPS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau và tập hợp thông số chuỗi thuộc tính (thuộc tính SPS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau. Nghĩa là, thiết bị giải mã dữ liệu ba chiều sử dụng thông tin điều khiển được bao gồm theo chuỗi tập hợp thông số hình học để giải mã các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau, và sử dụng thông tin điều khiển được bao gồm trong tập hợp thông số chuỗi thuộc tính để giải mã các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau.

Ví dụ, dòng bit bao gồm dòng tập hợp thông số (dòng PS) mà bao gồm thông tin điều khiển mà là chung cho các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau và các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau. Nghĩa là, thiết bị giải mã dữ liệu ba chiều sử dụng thông tin điều khiển được bao gồm trong dòng tập hợp thông số để giải mã các đoạn của thông tin hình học được kết hợp với các điểm thời gian khác nhau và các đoạn thông tin thuộc tính được kết hợp với các điểm thời gian khác nhau.

Ví dụ, dòng bit bao gồm đoạn đầu bộ phận truy cập (đoạn đầu AU) mà bao gồm thông tin điều khiển mà là chung trong bộ phận truy cập. Nghĩa là, thiết bị giải mã dữ liệu ba chiều sử dụng thông tin điều khiển được bao gồm trong đoạn đầu bộ phận truy cập để giải mã thông tin hình học và thông tin thuộc tính được bao gồm trong bộ phận truy cập.

Ví dụ, thiết bị giải mã dữ liệu ba chiều giải mã một cách độc lập các nhóm của các khung (các GOF) được tạo nên bởi một hoặc nhiều bộ phận truy cập. Nghĩa là, GOF là bộ phận truy cập ngẫu nhiên.

Ví dụ, dòng bit bao gồm đoạn đầu GOF mà bao gồm thông tin điều khiển mà là

chung trong GOF. Nghĩa là, thiết bị giải mã dữ liệu ba chiều giải mã thông tin hình học và thông tin thuộc tính được bao gồm trong GOF nhờ sử dụng thông tin điều khiển được bao gồm trong đoạn đầu GOF.

Ví dụ, thiết bị giải mã dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện các quy trình xử lý nêu trên nhờ sử dụng bộ nhớ.

Phương án 5

Tiếp theo, cấu hình của bộ chia 4911 sẽ được mô tả. Fig.47 là sơ đồ khối minh họa bộ chia 4911. Bộ chia 4911 bao gồm bộ chia lát 4931, bộ chia mảnh thông tin hình học (bộ chia mảnh hình học) 4932, và bộ chia mảnh thuộc tính thông tin (bộ chia mảnh thuộc tính) 4933.

Bộ chia lát 4931 tạo ra các đoạn của lát thông tin hình học bằng cách chia thông tin hình học (vị trí (hình học)) thành các lát. Bộ chia lát 4931 cũng tạo ra các đoạn của thông tin thuộc tính lát bằng cách chia thông tin thuộc tính (attribute) thành các lát. Bộ chia lát 4931 cũng đưa ra thông tin bổ sung lát (siêu dữ liệu lát) bao gồm thông tin liên quan đến việc chia lát và thông tin được tạo ra trong việc chia lát.

Bộ chia mảnh thông tin hình học 4932 tạo ra các đoạn của thông tin hình học phân chia (các đoạn của thông tin hình học mảnh) bằng cách chia các đoạn của lát thông tin hình học thành các mảnh. Bộ chia mảnh thông tin hình học 4932 cũng đưa ra thông tin bổ sung mảnh hình học (hình học siêu dữ liệu mảnh) bao gồm thông tin liên quan đến việc chia mảnh của thông tin hình học và thông tin được tạo ra trong việc chia mảnh của thông tin hình học.

Bộ chia mảnh thuộc tính thông tin 4933 tạo ra các đoạn của thông tin thuộc tính phân chia (các đoạn của thông tin thuộc tính mảnh) bằng cách chia các đoạn của thông tin thuộc tính lát thành các mảnh. Bộ chia mảnh thuộc tính thông tin 4933 cũng đưa ra thông tin bổ sung mảnh thuộc tính (thuộc tính siêu dữ liệu mảnh) bao gồm thông tin liên quan đến việc chia mảnh của thông tin thuộc tính và thông tin được tạo ra trong việc chia mảnh của thông tin thuộc tính.

Lưu ý rằng số của các lát hoặc các mảnh được tạo ra bởi việc phân chia là bằng hoặc lớn hơn 1. Nghĩa là, việc chia lát hoặc việc chia mảnh có thể không được thực hiện.

Mặc dù ví dụ trong đó việc chia mảnh được thực hiện sau khi việc chia lát đã được thể hiện ở đây, việc chia lát có thể được thực hiện sau khi việc chia mảnh. Theo cách khác,

các bộ phận khác của việc phân chia có thể được định rõ cùng với lát và mảnh, và việc phân chia có thể được thực hiện dựa vào ba hoặc nhiều bộ phận của việc phân chia.

Trong phần dưới đây, phương pháp chia dữ liệu đám mây điểm sẽ được mô tả. Fig.48 là sơ đồ minh họa ví dụ về việc chia lát và việc chia mảnh.

Thứ nhất, phương pháp của việc chia lát sẽ được mô tả. Bộ chia 4911 chia dữ liệu đám mây điểm ba chiều thành các đám mây điểm tùy ý trong các bộ phận của các lát. Trong việc chia lát, bộ chia 4911 không phân tách thông tin hình học và thông tin thuộc tính trên điểm và chia một cách tập hợp thông tin hình học và thông tin thuộc tính trên điểm thành các lát. Nghĩa là, bộ chia 4911 thực hiện việc chia lát theo cách sao cho thông tin hình học và thông tin thuộc tính trên điểm bất kỳ thuộc về cùng lát. Lưu ý rằng dữ liệu đám mây điểm có thể được chia theo cách thức bất kỳ của các lát theo cách bất ký đến khi yêu cầu này được đáp ứng. Bộ phận nhỏ nhất của việc phân chia là điểm. Ví dụ, thông tin hình học và thông tin thuộc tính được chia thành cùng số của các lát. Ví dụ, sau khi việc chia lát, thông tin hình học tương ứng với điểm ba chiều và thông tin thuộc tính tương ứng với điểm ba chiều được bao gồm trong cùng lát.

Trong việc chia lát, bộ chia 4911 cũng tạo ra thông tin bổ sung lát, mà là thông tin bổ sung liên quan đến số của các lát và phương pháp phân chia. Thông tin bổ sung lát là chung cho thông tin hình học và thông tin thuộc tính. Ví dụ, thông tin bổ sung lát bao gồm thông tin chỉ báo kích thước hoặc vị trí tọa độ tham chiếu hoặc độ dài của phía của ô vuông giới hạn sau khi việc phân chia. Thông tin bổ sung lát cũng bao gồm thông tin chỉ báo số của các lát, loại của việc phân chia và tương tự.

Tiếp theo, phương pháp của việc chia mảnh sẽ được mô tả. Bộ chia 4911 chia dữ liệu nhận được từ việc chia lát thành lát thông tin hình học (lát G) và thông tin thuộc tính lát (lát), và chia mỗi trong số lát thông tin hình học và thông tin thuộc tính lát thành các mảnh.

Lưu ý rằng mặc dù Fig.48 thể hiện ví dụ trong đó cây bát phân cấu trúc được sử dụng cho việc phân chia, dữ liệu đám mây điểm có thể được chia thành số bất kỳ của các lát hoặc các mảnh theo phương pháp phân chia bất kỳ.

Bộ chia 4911 có thể chia thông tin hình học và thông tin thuộc tính theo các phương pháp chia khác nhau hoặc theo phương pháp chia giống nhau. Bộ chia 4911 có thể chia các lát thành các mảnh theo các phương pháp chia khác nhau hoặc theo phương pháp chia giống nhau.

Trong việc chia mảnh, bộ chia 4911 cũng tạo ra thông tin bổ sung mảnh liên quan đến số của các mảnh và phương pháp phân chia. Thông tin bổ sung mảnh là khác nhau giữa thông tin hình học và thông tin thuộc tính (có thông tin bổ sung mảnh hình học và thông tin bổ sung mảnh thuộc tính). Ví dụ, thông tin bổ sung mảnh bao gồm thông tin chỉ báo kích thước hoặc vị trí tọa độ tham chiếu hoặc độ dài của phía của ô vuông giới hạn sau khi việc phân chia. Thông tin bổ sung mảnh cũng bao gồm thông tin chỉ báo số của các mảnh, loại của việc phân chia và tương tự.

Tiếp theo, ví dụ về phương pháp chia dữ liệu đám mây điểm thành các lát hoặc các mảnh sẽ được mô tả. Với phương pháp chia lát hoặc mảnh, bộ chia 4911 có thể sử dụng phương pháp được xác định trước hoặc thay đổi một cách thích ứng phương pháp được sử dụng tùy thuộc vào dữ liệu đám mây điểm.

Trong việc chia lát, bộ chia 4911 chia không gian ba chiều mà không có việc phân tách thông tin hình học và thông tin thuộc tính. Ví dụ, bộ chia 4911 xác định hình dạng của đối tượng và chia không gian ba chiều thành các lát dựa vào hình dạng của đối tượng. Ví dụ, bộ chia 4911 trích đối tượng, chẳng hạn như cây hoặc công trình xây dựng, và thực hiện việc phân chia trên cơ sở đối tượng. Ví dụ, bộ chia 4911 thực hiện việc chia lát theo cách sao cho toàn bộ của một hoặc nhiều đối tượng được bao gồm trong một lát. Theo cách khác bộ chia 4911 có thể chia một đối tượng thành các lát.

Trong trường hợp này, thiết bị mã hóa có thể sử dụng phương pháp mã hóa khác nhau đối với mỗi lát. Ví dụ, thiết bị mã hóa có thể sử dụng phương pháp nén chất lượng cao đối với đối tượng cụ thể hoặc một số đối tượng cụ thể. Trong trường hợp đó, thiết bị mã hóa có thể lưu trữ thông tin chỉ báo phương pháp mã hóa đối với mỗi lát trong thông tin bổ sung (siêu dữ liệu).

Bộ chia 4911 có thể cũng thực hiện việc chia lát theo cách sao cho mỗi lát tương ứng với không gian tọa độ được xác định trước, dựa vào thông tin bản đồ hoặc thông tin hình học.

Trong việc chia mảnh, bộ chia 4911 chia một cách độc lập thông tin hình học và thông tin thuộc tính. Ví dụ, bộ chia 4911 chia lát thành các mảnh tùy thuộc vào lượng dữ liệu hoặc lượng xử lý. Ví dụ, bộ chia 4911 xác định xem lượng dữ liệu của lát (số của các điểm ba chiều được bao gồm trong lát chẳng hạn) là lớn hơn ngưỡng được xác định trước. Nếu lượng dữ liệu của lát là lớn hơn ngưỡng, bộ chia 4911 chia lát thành các mảnh. Nếu lượng dữ liệu của lát là nhỏ hơn ngưỡng, bộ chia 4911 không chia lát thành các mảnh.

Ví dụ, bộ chia 4911 chia lát thành các mảnh theo cách sao cho lượng xử lý hoặc thời gian xử lý của thiết bị giải mã nằm trong khoảng được xác định trước (nghĩa là, là bằng hoặc nhỏ hơn trị số được xác định trước). Theo cách này, lượng xử lý đối với mỗi mảnh của thiết bị giải mã được tạo nên không đổi, và việc xử lý được phân bố trong thiết bị giải mã được tạo thuận lợi.

Khi lượng xử lý đối với thông tin hình học và lượng xử lý đối với thông tin thuộc tính là khác nhau, ví dụ, lượng xử lý đối với thông tin hình học là lớn hơn lượng xử lý đối với thông tin thuộc tính, bộ chia 4911 chia thông tin hình học thành nhiều mảnh hơn so với thông tin thuộc tính.

Tùy thuộc vào nội dung, nếu thiết bị giải mã có thể giải mã và hiển thị thông tin hình học trước khi giải mã và hiển thị thông tin thuộc tính, ví dụ, bộ chia 4911 có thể chia thông tin hình học thành nhiều mảnh hơn so với thông tin thuộc tính. Trong trường hợp đó, thiết bị giải mã có thể làm tăng tính song song của việc xử lý của thông tin hình học so với việc xử lý của thông tin thuộc tính, và do đó có thể xử lý thông tin hình học nhanh hơn so với thông tin thuộc tính.

Lưu ý rằng thiết bị giải mã không phải xử lý các lát hoặc các mảnh của dữ liệu song song, và có thể xác định xem có xử lý các lát hoặc các mảnh của dữ liệu song song hay không dựa vào số hoặc dung lượng của các bộ xử lý giải mã.

Bằng cách thực hiện các sự phân chia như được nêu trên, mã hóa thích ứng có thể được thực hiện tùy thuộc vào nội dung hoặc đối tượng. Ngoài ra, quy trình giải mã song song có thể được thực hiện. Theo cách này, tính linh hoạt của hệ thống mã hóa dữ liệu đám mây điểm hoặc hệ thống giải mã dữ liệu đám mây điểm được nâng cao.

Fig.49 là sơ đồ minh họa ví dụ về mẫu hình chia lát và mẫu hình chia mảnh. Trên hình vẽ, DU biểu diễn đơn vị dữ liệu (DataUnit), và thể hiện mảnh hoặc lát của dữ liệu. Mỗi DU bao gồm chỉ số lát (SliceIndex) và chỉ số mảnh (TileIndex). Trên hình vẽ, chỉ số bên trên của DU chỉ báo chỉ số lát, và chỉ số bên trên của DU chỉ báo chỉ số mảnh.

Trong mẫu hình 1, trong việc chia lát, thông tin hình học và thông tin thuộc tính được chia thành cùng số của các lát G và các lát A theo phương pháp chia giống nhau. Trong việc chia mảnh, lát G và lát A được chia thành các số khác nhau của các mảnh theo các phương pháp chia khác nhau. Các lát G được chia thành cùng số của các mảnh theo phương pháp chia giống nhau. Các lát A được chia thành cùng số của các mảnh theo phương pháp chia giống nhau.

Theo mẫu hình 2, trong việc chia lát, thông tin hình học và thông tin thuộc tính được chia thành cùng số của các lát G và các lát A theo phương pháp chia giống nhau. Trong việc chia mảnh, lát G và lát A được chia thành các số khác nhau của các mảnh theo các phương pháp chia khác nhau. Các lát G được chia thành các số khác nhau của các mảnh theo các phương pháp chia khác nhau. Các lát A được chia thành các số khác nhau của các mảnh theo các phương pháp chia khác nhau.

Phương án 6

Dưới đây, ví dụ thực hiện việc chia lát sau khi việc chia mảnh sẽ được mô tả. Ứng dụng tự quản cho việc dẫn động được tự động hóa của phương tiện giao thông, v.v., không yêu cầu dữ liệu đám mây điểm của tất cả các vùng nhưng dữ liệu đám mây điểm của vùng bao quanh phương tiện giao thông hoặc vùng theo hướng di chuyển của phương tiện giao thông. Ở đây, các mảnh và các lát có thể được sử dụng để giải mã một cách có lựa chọn dữ liệu đám mây điểm ban đầu. Có khả năng đạt được việc nâng cao về hiệu quả mã hóa hoặc việc xử lý song song bằng cách chia dữ liệu đám mây điểm ba chiều thành các mảnh và còn chia các mảnh thành các lát. Khi dữ liệu được chia, thông tin bổ sung (siêu dữ liệu) được tạo ra, và thông tin bổ sung được tạo ra được truyền tới bộ đa hợp.

Fig.50 là sơ đồ khối minh họa cấu hình của bộ mã hóa thứ nhất 5010 được bao gồm trong thiết bị mã hóa dữ liệu ba chiều theo phương án này. Bộ mã hóa thứ nhất 5010 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách mã hóa dữ liệu đám mây điểm nhờ sử dụng phương pháp mã hóa thứ nhất (PCC trên cơ sở hình học (GPCC)). Bộ mã hóa thứ nhất 5010 bao gồm bộ chia 5011, các bộ mã hóa thông tin hình học 5012, bộ mã hóa thông tin các thuộc tính 5013, bộ mã hóa thông tin bổ sung 5014, và bộ đa hợp 5015.

Bộ chia 5011 tạo ra các đoạn dữ liệu được chia bằng cách chia dữ liệu đám mây điểm. Cụ thể là, bộ chia 5011 tạo ra các đoạn dữ liệu được chia bằng cách chia không gian của dữ liệu đám mây điểm thành các không gian con. Ở đây, không gian con là một trong số mảnh và lát, hoặc sự kết hợp của mảnh và lát. Cụ thể hơn là, dữ liệu đám mây điểm bao gồm thông tin hình học, thông tin thuộc tính, và thông tin bổ sung. Bộ chia 5011 chia thông tin hình học thành các đoạn của thông tin hình học được chia và thông tin thuộc tính thành các đoạn của thông tin thuộc tính được chia. Ngoài ra, bộ chia 5011 tạo ra thông tin bổ sung liên quan đến việc phân chia.

Ví dụ, thứ nhất, bộ chia 5011 chia đám mây điểm thành các mảnh. Tiếp theo, bộ chia 5011 còn chia các mảnh được thu nhận thành các lát.

Các bộ mã hóa thông tin hình học 5012 tạo ra các đoạn của thông tin hình học được mã hóa bằng cách mã hóa các đoạn của thông tin hình học được chia. Ví dụ, các bộ mã hóa thông tin hình học 5012 xử lý song song các đoạn của thông tin hình học được chia.

Bộ mã hóa thông tin các thuộc tính 5013 tạo ra các đoạn của thông tin thuộc tính được mã hóa bằng cách mã hóa các đoạn của thông tin thuộc tính được chia. Ví dụ, bộ mã hóa thông tin các thuộc tính 5013 xử lý song song các đoạn của thông tin hình học được chia.

Bộ mã hóa thông tin bổ sung 5014 tạo ra thông tin bổ sung được mã hóa bằng cách mã hóa thông tin bổ sung được bao gồm trong dữ liệu đám mây điểm và thông tin bổ sung liên quan đến việc phân chia dữ liệu được tạo ra ở thời gian chia bởi bộ chia 5011.

Bộ đa hợp 5015 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách đa hợp các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa, và truyền dữ liệu được mã hóa được tạo ra. Thông tin bổ sung được mã hóa cũng được sử dụng ở thời gian giải mã.

Cần lưu ý rằng mặc dù Fig.50 thể hiện hai bộ mã hóa thông tin hình học 5012 và hai bộ mã hóa thông tin các thuộc tính 5013 như ví dụ, số của các bộ mã hóa thông tin hình học 5012 và số của bộ mã hóa thông tin các thuộc tính 5013 có thể là một hoặc ít nhất ba. Hơn nữa, các đoạn dữ liệu được chia có thể được xử lý song song trong các chip giống nhau, chẳng hạn như các lõi trong CPU, trong lõi của mỗi trong số các chip, hoặc trong các lõi của mỗi trong số các chip.

Phần dưới đây mô tả quy trình giải mã. Fig.51 là sơ đồ khối minh họa cấu hình của bộ giải mã thứ nhất 5020. Bộ giải mã thứ nhất 5020 khôi phục dữ liệu đám mây điểm bằng cách giải mã dữ liệu được mã hóa (dòng được mã hóa) được tạo ra bằng cách mã hóa dữ liệu đám mây điểm nhờ sử dụng phương pháp mã hóa thứ nhất (GPCC). Bộ giải mã thứ nhất 5020 bao gồm bộ giải đa hợp 5021, các bộ giải mã thông tin hình học 5022, bộ giải mã thông tin các thuộc tính 5023, bộ giải mã thông tin bổ sung 5024, và bộ kết hợp 5025.

Bộ giải đa hợp 5021 tạo ra các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa bằng cách giải đa hợp dữ liệu được mã hóa (dòng được mã hóa).

Các bộ giải mã thông tin hình học 5022 tạo ra các đoạn của thông tin hình học được chia bằng cách giải mã các đoạn của thông tin hình học được mã hóa. Ví dụ, các bộ

giải mã thông tin hình học 5022 xử lý các đoạn của thông tin hình học được mã hóa song song.

Bộ giải mã thông tin các thuộc tính 5023 tạo ra các đoạn của thông tin thuộc tính được chia bằng cách giải mã các đoạn của thông tin thuộc tính được mã hóa. Ví dụ, bộ giải mã thông tin các thuộc tính 5023 xử lý các đoạn của thông tin thuộc tính được mã hóa song song.

Bộ giải mã thông tin bổ sung 5024 tạo ra thông tin bổ sung bằng cách giải mã thông tin bổ sung được mã hóa.

Bộ kết hợp 5025 tạo ra thông tin hình học bằng cách kết hợp các đoạn của thông tin hình học được chia nhờ sử dụng thông tin bổ sung. Bộ kết hợp 5025 cũng tạo ra thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin thuộc tính được chia nhờ sử dụng thông tin bổ sung. Ví dụ, thứ nhất, bộ kết hợp 5025 tạo ra các đoạn dữ liệu đám mây điểm tương ứng với các mảnh bằng cách kết hợp các đoạn của dữ liệu đám mây điểm được giải mã tương ứng với các lát nhờ sử dụng thông tin bổ sung lát. Tiếp theo, bộ kết hợp 5025 khôi phục dữ liệu đám mây điểm ban đầu bằng cách kết hợp các đoạn dữ liệu đám mây điểm tương ứng với các mảnh nhờ sử dụng thông tin bổ sung mảnh.

Cần lưu ý rằng mặc dù Fig.50 thể hiện hai bộ giải mã thông tin hình học 5022 và hai bộ giải mã thông tin các thuộc tính 5023 như ví dụ, số của các bộ giải mã thông tin hình học 5022 và số của bộ giải mã thông tin các thuộc tính 5023 có thể là một hoặc ít nhất ba. Hơn nữa, các đoạn dữ liệu được chia có thể được xử lý song song trong các chip giống nhau, chẳng hạn như các lõi trong CPU, trong lõi của mỗi trong số các chip, hoặc trong các lõi của mỗi trong số các chip.

Phần dưới đây mô tả phương pháp chia dữ liệu đám mây điểm. Ứng dụng tự quản cho việc dẫn động được tự động hóa của phương tiện giao thông, v.v., không yêu cầu dữ liệu đám mây điểm của tất cả các vùng nhưng dữ liệu đám mây điểm của vùng bao quanh phương tiện giao thông hoặc vùng theo hướng di chuyển của phương tiện giao thông.

Fig.52 là sơ đồ minh họa các ví dụ về hình dạng mảnh. Như được thể hiện trên Fig.52, các ví dụ về hình dạng mảnh có thể bao gồm các loại các hình dạng khác nhau chẳng hạn như hình tròn, hình chữ nhật, hoặc hình elip.

Fig.53 là sơ đồ minh họa ví dụ về các mảnh và các lát. Thành phần của các lát có thể khác nhau giữa các mảnh. Ví dụ, thành phần của các mảnh hoặc các lát có thể được tối

ưu hóa dựa vào khối lượng dữ liệu. Theo cách khác thành phần của các mảnh hoặc các lát có thể được tối ưu hóa dựa vào tốc độ giải mã.

Việc chia mảnh có thể được thực hiện dựa vào thông tin hình học. Trong trường hợp này, thông tin thuộc tính được chia theo cách thức như nhau như thông tin hình học tương ứng.

Hơn nữa, trong việc chia lát sau khi việc chia mảnh, thông tin hình học và thông tin thuộc tính có thể được chia thành các lát nhờ sử dụng các phương pháp khác nhau. Ví dụ, phương pháp chia lát trong mỗi mảnh có thể được lựa chọn khi yêu cầu từ ứng dụng. Phương pháp chia lát khác nhau hoặc phương pháp chia mảnh khác nhau có thể được sử dụng dựa vào yêu cầu từ ứng dụng.

Ví dụ, bộ chia 5011 chia dữ liệu đám mây điểm ba chiều thành một hoặc nhiều mảnh theo hình dạng hai chiều được thu nhận bằng cách xem dữ liệu đám mây điểm ba chiều từ bên trên, dựa vào thông tin vị trí chẳng hạn như thông tin bản đồ. Bộ chia 5011 chia mỗi trong số một hoặc nhiều mảnh thành một hoặc nhiều lát sau đó.

Cần lưu ý rằng bộ chia 5011 có thể chia thông tin hình học (hình học) và thông tin thuộc tính (attribute) thành các lát nhờ sử dụng phương pháp như nhau.

Cần lưu ý rằng mỗi trong số thông tin hình học và thông tin thuộc tính có thể là của một loại hoặc hai hoặc nhiều hơn hai loại. Ngoài ra, khi dữ liệu đám mây điểm không có thông tin thuộc tính, thông tin thuộc tính có thể là không cần thiết.

Fig.54 là sơ đồ khối của bộ chia 5011. Bộ chia 5011 bao gồm bộ chia mảnh 5031, bộ chia lát thông tin hình học (bộ chia lát hình học) 5032, và bộ chia lát thông tin thuộc tính (bộ chia lát thuộc tính) 5033.

Bộ chia mảnh 5031 tạo ra các đoạn của thông tin hình học mảnh bằng cách chia thông tin hình học (vị trí (hình học)) thành các mảnh. Ngoài ra, bộ chia mảnh 5031 tạo ra các đoạn của thông tin thuộc tính mảnh bằng cách chia thông tin thuộc tính (attribute) thành các mảnh. Ngoài ra, bộ chia mảnh 5031 đưa ra thông tin bổ sung mảnh (siêu dữ liệu mảnh) bao gồm thông tin liên quan đến việc chia mảnh và thông tin được tạo ra trong việc chia mảnh.

Bộ chia lát thông tin hình học 5032 tạo ra các đoạn của thông tin hình học được chia (các đoạn của lát thông tin hình học) bằng cách chia các đoạn của thông tin hình học mảnh thành các lát. Ngoài ra, bộ chia lát thông tin hình học 5032 đưa ra thông tin bổ sung

lát hình học (siêu dữ liệu lát hình học) bao gồm thông tin liên quan đến việc chia lát của thông tin hình học và thông tin được tạo ra trong việc chia lát của thông tin hình học.

Bộ chia lát thông tin thuộc tính 5033 tạo ra các đoạn của thông tin thuộc tính được chia (các đoạn của thông tin thuộc tính lát) bằng cách chia các đoạn của thông tin thuộc tính mảnh thành các lát. Ngoài ra, bộ chia lát thông tin thuộc tính 5033 đưa ra thông tin bổ sung lát thuộc tính (siêu dữ liệu lát thuộc tính) bao gồm thông tin liên quan đến việc chia lát của thông tin thuộc tính và thông tin được tạo ra trong việc chia lát của thông tin thuộc tính.

Phần dưới đây mô tả các ví dụ về hình dạng mảnh. Toàn bộ bản đồ ba chiều (bản đồ 3D) được chia thành các mảnh. Dữ liệu của các mảnh được truyền một cách có lựa chọn tới thiết bị giải mã dữ liệu ba chiều. Theo cách khác dữ liệu của các mảnh được truyền tới thiết bị giải mã dữ liệu ba chiều theo thứ tự giảm dần của mức độ quan trọng. Hình dạng mảnh có thể được lựa chọn từ các hình dạng theo tình huống.

Fig.55 là sơ đồ minh họa ví dụ về bản đồ nhìn từ trên xuống của dữ liệu đám mây điểm được thu nhận bởi LiDAR. ví dụ được thể hiện trên Fig.55 là dữ liệu đám mây điểm của xa lộ và bao gồm cầu vượt (cầu qua đường)

Fig.56 là sơ đồ minh họa ví dụ về việc chia dữ liệu đám mây điểm được thể hiện trên Fig.55 thành các mảnh hình vuông. Dễ dàng thực hiện việc phân chia như vậy thành các hình vuông trong máy chủ bản đồ. Đối với tuyến đường thông thường, độ cao của mảnh được thiết đặt thấp. Độ cao của các mảnh được thiết đặt cao hơn cho cầu vượt so với cho tuyến đường thông thường sao cho các mảnh chứa cầu vượt.

Fig.57 là sơ đồ minh họa ví dụ về việc chia dữ liệu đám mây điểm được thể hiện trên Fig.55 thành các mảnh hình tròn. Trong trường hợp này, các mảnh lân cận có thể chồng lấn với nhau theo hình chiếu bằng. Khi phương tiện giao thông yêu cầu dữ liệu đám mây điểm của vùng xung quanh, thiết bị mã hóa dữ liệu ba chiều truyền, tới phương tiện giao thông, dữ liệu đám mây điểm của vùng bao gồm các cột (các hình tròn theo hình chiếu từ trên xuống) xung quanh phương tiện giao thông.

Đối với với ví dụ được thể hiện trên Fig.56, đối với tuyến đường thông thường, độ cao của mảnh được thiết đặt thấp. Độ cao của các mảnh được thiết đặt cao hơn cho cầu vượt so với cho tuyến đường thông thường sao cho các mảnh chứa cầu vượt.

Thiết bị mã hóa dữ liệu ba chiều có thể thay đổi độ cao của mảnh theo, ví dụ, hình

dạng hoặc độ cao của tuyến đường hoặc công trình xây dựng. Ngoài ra, thiết bị mã hóa dữ liệu ba chiều có thể thay đổi độ cao của mảnh theo thông tin vị trí hoặc thông tin vùng. Ngoài ra, thiết bị mã hóa dữ liệu ba chiều có thể thay đổi độ cao của mỗi mảnh. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể thay đổi độ cao của các mảnh đối với mỗi khu vực bao gồm các mảnh. Theo cách khác, thiết bị mã hóa dữ liệu ba chiều có thể thiết đặt các mảnh trong khu vực tới độ cao như nhau. Hơn nữa, các mảnh có các độ cao khác nhau có thể chồng lấn với nhau theo hình chiếu từ trên xuống.

Fig.58 là sơ đồ minh họa ví dụ về việc chia mảnh khi các mảnh có các loại các hình dạng khác nhau, các kích thước, và các độ cao được sử dụng. Mảnh bất kỳ có thể có hình dạng hoặc kích thước bất kỳ, hoặc sự kết hợp của chúng.

Ví dụ, cùng với việc thực hiện việc phân chia thành các mảnh hình vuông không chồng lấn và việc thực hiện việc phân chia thành các mảnh hình tròn chồng lấn như được nêu trên, thiết bị mã hóa dữ liệu ba chiều có thể thực hiện việc phân chia thành các mảnh hình vuông chồng lần. Hơn nữa, hình dạng mảnh không cần là hình vuông hoặc hình tròn, và có thể là hình đa giác có ba hoặc nhiều hơn ba đỉnh, hoặc hình dạng không có đỉnh.

Hơn nữa, hình dạng mảnh có thể là hai hoặc nhiều hơn hai loại, và các mảnh có các hình dạng khác nhau có thể chồng lấn với nhau. Ngoài ra, hình dạng mảnh có thể là của một hoặc nhiều loại; và khi hình dạng như nhau được sử dụng cho các mảnh được chia, hình dạng như nhau có thể bao gồm các hình dạng khác nhau về kích thước hoặc các hình dạng như vậy có thể chồng lấn với nhau.

Ví dụ, mảnh được sử dụng là lớn hơn đối với vùng không bao gồm đối tượng chẳng hạn như tuyến đường so với vùng bao gồm đối tượng. Hơn nữa, thiết bị mã hóa dữ liệu ba chiều có thể thay đổi một cách thích ứng hình dạng mảnh hoặc kích thước theo đối tượng.

Hơn nữa, ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể thiết đặt các mảnh theo hướng di chuyển của xe ô tô (phương tiện giao thông) tới kích thước lớn bởi vì việc đọc của các mảnh ở khoảng cách lớn phía trước của xe ô tô theo hướng di chuyển là thường được cần đến; và thiết đặt các mảnh ở phía bên cạnh xe ô tô tới kích thước nhỏ hơn so với các mảnh theo hướng di chuyển bởi vì xe ô tô thường ít di chuyển tới phía bên.

Fig.59 là sơ đồ minh họa ví dụ về dữ liệu của các mảnh được lưu trữ trong máy chủ. Ví dụ, dữ liệu đám mây điểm được chia thành các mảnh và được mã hóa trước, và dữ liệu được mã hóa được thu nhận được lưu trữ trong máy chủ. Người dùng thu nhận dữ liệu của các mảnh mong muốn từ máy chủ khi cần. Theo cách khác máy chủ (thiết bị mã hóa

dữ liệu ba chiều) có thể thực hiện việc chia mảnh và mã hóa sao cho các mảnh bao gồm dữ liệu được mong muốn bởi người dùng, đáp lại lệnh từ người dùng.

Ví dụ, khi vật thể di chuyển được (phương tiện giao thông) di chuyển ở tốc độ cao, có thể nhận thấy rằng dữ liệu đám mây điểm rộng hơn được cần đến. Vì lý do này, máy chủ có thể xác định hình dạng mảnh và kích thước dựa vào tốc độ xe được đánh giá trước (chẳng hạn, tốc độ hợp pháp trên đường, tốc độ xe được đánh giá từ độ rộng hoặc hình dạng của tuyến đường, hoặc tốc độ xe theo thống kê), và thực hiện việc chia mảnh. Theo cách khác như được thể hiện trên Fig.59, máy chủ có thể mã hóa các mảnh có hình dạng hoặc kích thước trước, và lưu trữ dữ liệu được thu nhận. Vật thể di chuyển được có thể thu nhận dữ liệu của các mảnh có hình dạng hoặc kích thước thích hợp theo hướng di chuyển và tốc độ của vật thể di chuyển được.

Fig.60 là sơ đồ minh họa ví dụ về hệ thống liên quan đến việc chia mảnh. Như được thể hiện trên Fig.60, hình dạng mảnh và vùng có thể được xác định dựa vào vị trí của anten (trạm gốc) mà là phương tiện truyền thông truyền dữ liệu đám mây điểm, hoặc ở vùng truyền thông được hỗ trợ bởi anten. Theo cách khác, khi dữ liệu đám mây điểm được tạo ra bởi bộ cảm biến chẳng hạn như camera, hình dạng mảnh và vùng có thể được xác định dựa vào vị trí hoặc khoảng đích (khoảng phát hiện) của bộ cảm biến.

Một mảnh có thể được gán tới một anten hoặc một bộ cảm biến, hoặc một mảnh có thể được gán tới các anten hoặc các bộ cảm biến. Ngoài ra, các mảnh có thể được gán tới một anten hoặc một bộ cảm biến. Anten hoặc bộ cảm biến có thể được cố định hoặc di chuyển được.

Ví dụ, dữ liệu được mã hóa được chia thành các mảnh có thể được quản lý bởi máy chủ được kết nối tới anten hoặc bộ cảm biến đối với vùng được gán tới các mảnh. Máy chủ có thể quản lý dữ liệu được mã hóa của vùng và thông tin mảnh của vùng lân cận. Các đoạn dữ liệu được mã hóa của các mảnh có thể được quản lý trong máy chủ trung tâm (đám mây) mà quản lý các máy chủ đều tương ứng với một trong số các mảnh khác nhau. Theo cách khác, thay vì cung cấp các máy chủ đều tương ứng với một trong số các mảnh khác nhau, các anten hoặc các bộ cảm biến có thể được kết nối trực tiếp tới máy chủ trung tâm.

Cần lưu ý rằng khoảng đích của anten hoặc bộ cảm biến có thể thay đổi tùy thuộc vào công suất các sóng radio, các độ chênh lệch giữa các thiết bị, và các điều kiện lắp đặt, và hình dạng mảnh và kích thước có thể thay đổi phù hợp với chúng. Thay vì mảnh, lát hoặc khung PCC có thể được gán dựa vào khoảng đích của anten hoặc bộ cảm biến.

Phần dưới đây mô tả phương pháp chia mảnh thành các lát. Có khả năng nâng cao hiệu quả mã hóa bằng cách gán các đối tượng tương tự tới cùng lát.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể nhận diện các đối tượng (chẳng hạn, tuyến đường, công trình xây dựng, cây) nhờ sử dụng các đặc điểm của dữ liệu đám mây điểm, và thực hiện việc chia lát bởi việc tạo chùm các đám mây điểm đối với mỗi trong số các đối tượng.

Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể phân loại các đối tượng có cùng thuộc tính thành các nhóm, và thực hiện việc chia lát bằng cách gán lát tới mỗi trong số các nhóm. Ở đây, thuộc tính là, ví dụ, thông tin liên quan đến chuyển động. Các đối tượng được phân loại thành các nhóm theo thông tin động về những người đi bộ, các xe ô tô, v.v., thông tin tựa động về các vụ tai nạn, tắc đường, v.v., thông tin tựa tĩnh về điều khiển giao thông, công việc trên đường, v.v., và thông tin tĩnh về các bề mặt đường, các cấu trúc, v.v.

Cần lưu ý rằng các lát có thể có dữ liệu chồng lấn. Ví dụ, khi việc chia lát được thực hiện đối với mỗi đối tượng nhóm, đối tượng bất kỳ có thể thuộc về một đối tượng nhóm hoặc hai hoặc nhiều hơn hai nhóm đối tượng.

Fig.61 là sơ đồ minh họa ví dụ về việc chia lát này. Ví dụ, mảnh là tựa hình khối theo ví dụ được thể hiện trên Fig.61. Cần lưu ý rằng, mảnh có thể ở dạng cột hoặc có hình dạng khác.

Các đám mây điểm được bao gồm trong mảnh được phân loại thành các nhóm đối tượng chẳng hạn như tuyến đường, công trình xây dựng, và cây. Sau đó, việc chia lát được thực hiện sao cho mỗi đối tượng nhóm được bao gồm trong một trong số các lát khác nhau. Sau đó, các lát được mã hóa một cách riêng biệt.

Phần dưới đây mô tả phương pháp mã hóa dữ liệu được chia. Thiết bị mã hóa dữ liệu ba chiều (bộ mã hóa thứ nhất 5010) mã hóa mỗi dữ liệu được chia. Khi thiết bị mã hóa dữ liệu ba chiều mã hóa thông tin thuộc tính, thiết bị mã hóa dữ liệu ba chiều tạo ra, như thông tin bổ sung, thông tin mối tương quan phụ thuộc chỉ báo dựa vào việc mã hóa thông tin thành phần (thông tin hình học, thông tin bổ sung, hoặc thông tin thuộc tính khác) đã được thực hiện. Nói cách khác, thông tin mối tương quan phụ thuộc chỉ báo, ví dụ, thông tin thành phần của đích tham chiếu (dependee). Trong trường hợp này, thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin mối tương quan phụ thuộc dựa vào thông tin thành phần tương ứng với hình dạng được chia đối với thông tin thuộc tính. Cần lưu ý rằng thiết bị mã hóa

dữ liệu ba chiều có thể tạo ra thông tin mối tương quan phụ thuộc dựa vào thông tin thành phần tương ứng với các hình dạng được chia.

Thông tin mối tương quan phụ thuộc có thể được tạo ra bởi thiết bị mã hóa dữ liệu ba chiều, và thông tin mối tương quan phụ thuộc có thể được truyền tới thiết bị giải mã dữ liệu ba chiều. Theo cách khác thiết bị giải mã dữ liệu ba chiều có thể tạo ra thông tin mối tương quan phụ thuộc, và thiết bị mã hóa dữ liệu ba chiều không cần truyền thông tin mối tương quan phụ thuộc. Ngoài ra, mối tương quan phụ thuộc được sử dụng bởi thiết bị mã hóa dữ liệu ba chiều có thể được xác định trước, và thiết bị mã hóa dữ liệu ba chiều không cần truyền thông tin mối tương quan phụ thuộc.

Fig.62 là sơ đồ minh họa ví dụ về mối tương quan phụ thuộc của mỗi dữ liệu. Đầu nhọn của mũi tên trên hình vẽ chỉ báo đích tham chiếu, và đầu còn lại của mũi tên chỉ báo gốc tham chiếu. Thiết bị giải mã dữ liệu ba chiều giải mã dữ liệu theo thứ tự từ đích tham chiếu tới gốc tham chiếu. Dữ liệu được chỉ báo bởi đường liền nét trên hình vẽ là dữ liệu được truyền thực tế, và dữ liệu được chỉ báo bởi đường đứt nét là dữ liệu không được truyền.

Trên hình vẽ, G biểu thị thông tin hình học, và A biểu thị thông tin thuộc tính. Gt1 biểu thị thông tin hình học cho số mảnh 1, và Gt2 biểu thị thông tin hình học cho số mảnh 2. Gt1s1 biểu thị thông tin hình học cho số mảnh 1 và số lát 1, Gt1s2 biểu thị thông tin hình học cho số mảnh 1 và số lát 2, Gt2s1 biểu thị thông tin hình học cho số mảnh 2 và số lát 1, và Gt2s2 biểu thị thông tin hình học cho số mảnh 2 và số lát 2. Tương tự, At1 biểu thị thông tin thuộc tính cho số mảnh 1, và At2 biểu thị thông tin thuộc tính cho số mảnh 2. At1s1 biểu thị thông tin thuộc tính cho số mảnh 1 và số lát 1, At1s2 biểu thị thông tin thuộc tính cho số mảnh 1 và số lát 2, At2s1 biểu thị thông tin thuộc tính cho số mảnh 2 và số lát 1, và At2s2 biểu thị thông tin thuộc tính cho số mảnh 2 và số lát 2.

Mtile biểu thị thông tin bổ sung mảnh, MGslice biểu thị thông tin bổ sung lát hình học, và MAslice biểu thị thông tin bổ sung lát thuộc tính. Dt1s1 biểu thị thông tin mối tương quan phụ thuộc của thông tin thuộc tính At1s1, và Dt2s1 biểu thị thông tin mối tương quan phụ thuộc của thông tin thuộc tính At2s1.

Cần lưu ý rằng cấu trúc khác nhau nhận được từ việc chia mảnh hoặc việc chia lát có thể được sử dụng theo ứng dụng v.v..

Thiết bị mã hóa dữ liệu ba chiều có thể bố trí lại dữ liệu theo thứ tự giải mã sao cho thiết bị giải mã dữ liệu ba chiều không cần bố trí lại dữ liệu. Cần lưu ý rằng thiết bị

giải mã dữ liệu ba chiều có thể bố trí lại dữ liệu, hoặc cả thiết bị mã hóa dữ liệu ba chiều và thiết bị giải mã dữ liệu ba chiều có thể bố trí lại dữ liệu.

Fig.63 là sơ đồ minh họa ví dụ về thứ tự giải mã của dữ liệu. Theo ví dụ được thể hiện trên Fig.63, dữ liệu được giải mã theo thứ tự từ bên trái. Thiết bị giải mã dữ liệu ba chiều giải mã, trong số dữ liệu có mối tương quan phụ thuộc với nhau, dữ liệu của đích tham chiếu trước tiên. Ví dụ, thiết bị mã hóa dữ liệu ba chiều bố trí lại dữ liệu theo thứ tự này và truyền dữ liệu. Cần lưu ý rằng thứ tự bất kỳ có thể được sử dụng miễn là dữ liệu của đích tham chiếu nắm quyền ưu tiên. Hơn nữa, thiết bị mã hóa dữ liệu ba chiều có thể truyền thông tin bổ sung và thông tin mối tương quan phụ thuộc trước dữ liệu.

Hơn nữa, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách có lựa chọn các mảnh dựa vào yêu cầu từ ứng dụng và thông tin được thu nhận từ đoạn đầu bộ phận NAL. Fig.64 là sơ đồ minh họa ví dụ về dữ liệu được mã hóa của các mảnh. Ví dụ, thứ tự giải mã của các mảnh là tùy chọn. Nói cách khác, các mảnh không cần có mối tương quan phụ thuộc với nhau.

Phần dưới đây mô tả cấu hình của bộ kết hợp 5025 được bao gồm trong bộ giải mã thứ nhất 5020. Fig.65 là sơ đồ khối minh họa cấu hình của bộ kết hợp 5025. Bộ kết hợp 5025 bao gồm bộ kết hợp lát thông tin hình học (bộ kết hợp lát hình học) 5041, bộ kết hợp thông tin thuộc tính lát (bộ kết hợp lát thuộc tính) 5042, và bộ kết hợp mảnh 5043.

Bộ kết hợp lát thông tin hình học 5041 tạo ra các đoạn của thông tin hình học mảnh bằng cách kết hợp các đoạn của thông tin hình học được chia nhờ sử dụng thông tin bổ sung lát hình học. Bộ kết hợp thông tin thuộc tính lát 5042 tạo ra các đoạn của thông tin thuộc tính mảnh bằng cách kết hợp các đoạn của thông tin thuộc tính được chia nhờ sử dụng thông tin bổ sung lát thuộc tính.

Bộ kết hợp mảnh 5043 tạo ra thông tin hình học bằng cách kết hợp các đoạn của thông tin hình học mảnh nhờ sử dụng thông tin bổ sung mảnh. Bên cạnh đó, bộ kết hợp mảnh 5043 tạo ra thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin thuộc tính mảnh nhờ sử dụng thông tin bổ sung mảnh.

Cần lưu ý rằng số của các lát hoặc các mảnh được chia ít nhất là một. Nói cách khác, việc chia lát hoặc việc chia mảnh không cần được thực hiện.

Phần dưới đây mô tả cấu trúc của dữ liệu được mã hóa được trải qua việc chia lát hoặc việc chia mảnh, và phương pháp lưu trữ dữ liệu được mã hóa trong bộ phận NAL

(phương pháp đa hợp). Fig.66 là sơ đồ minh họa cấu trúc của dữ liệu được mã hóa và phương pháp lưu trữ dữ liệu được mã hóa trong bộ phận NAL.

Dữ liệu được mã hóa (thông tin hình học được chia hoặc thông tin thuộc tính được chia) được lưu trữ trong bộ phận NAL tải trọng.

Dữ liệu được mã hóa bao gồm đoạn đầu và tải trọng. Đoạn đầu bao gồm thông tin nhận dạng dùng để nhận dạng dữ liệu được bao gồm trong tải trọng. Các ví dụ về thông tin nhận dạng bao gồm loại của việc chia lát hoặc việc chia mảnh (slice_type, tile_type), thông tin chỉ số dùng để nhận dạng lát hoặc mảnh (slice_idx, tile_idx), thông tin hình học của dữ liệu (lát hoặc mảnh), hoặc địa chỉ của dữ liệu (address). Thông tin chỉ số dùng để nhận dạng lát cũng được đề cập đến là chỉ số lát (SliceIndex). Thông tin chỉ số dùng để nhận dạng mảnh cũng được đề cập đến là chỉ số mảnh (TileIndex). Loại phân chia chỉ báo, ví dụ, phương pháp dựa vào hình dạng đối tượng như được nêu trên, phương pháp dựa vào thông tin bản đồ hoặc thông tin vị trí, hoặc phương pháp dựa vào khối lượng dữ liệu hoặc lượng xử lý.

Hơn nữa, đoạn đầu của dữ liệu được mã hóa bao gồm thông tin nhận dạng chỉ báo mối tương quan phụ thuộc. Theo cách khác, khi dữ liệu có mối tương quan phụ thuộc với nhau, đoạn đầu bao gồm thông tin nhận dạng dùng cho gốc tham chiếu để đề cập đến đích tham chiếu. Ví dụ, đoạn đầu của dữ liệu của đích tham chiếu bao gồm thông tin nhận dạng dùng để nhận dạng dữ liệu. Đoạn đầu của dữ liệu của gốc tham chiếu bao gồm thông tin nhận dạng chỉ báo đích tham chiếu. Cần lưu ý rằng khi thông tin nhận dạng dùng để nhận dạng dữ liệu, thông tin bổ sung liên quan đến việc chia lát hoặc việc chia mảnh, và thông tin nhận dạng chỉ báo mối tương quan phụ thuộc có thể nhận dạng được hoặc có thể dẫn ra được từ thông tin khác, các đoạn của thông tin này có thể được bỏ qua.

Phần dưới đây mô tả các thủ tục của quy trình mã hóa dữ liệu đám mây điểm và quy trình giải mã dữ liệu đám mây điểm theo phương án này. Fig.67 là lưu đồ của quy trình mã hóa dữ liệu đám mây điểm theo phương án này.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định phương pháp phân chia được sử dụng (S5011). Các ví dụ về phương pháp phân chia bao gồm việc chia mảnh và việc chia lát. Phương pháp phân chia có thể bao gồm số phân chia, loại phân chia, v.v., khi việc chia mảnh hoặc việc chia lát được thực hiện. Loại phân chia chỉ báo, ví dụ, phương pháp dựa vào hình dạng đối tượng như được nêu trên, phương pháp dựa vào thông tin bản đồ hoặc thông tin hình học, hoặc phương pháp dựa vào khối lượng dữ liệu hoặc lượng xử lý.

Cần lưu ý rằng phương pháp phân chia có thể được xác định trước.

Khi việc chia mảnh được thực hiện (Có ở S5012), thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách chia thông tin hình học và thông tin thuộc tính một cách tập hợp (S5013). Bên cạnh đó, thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin bổ sung mảnh liên quan đến việc chia mảnh. Cần lưu ý rằng thiết bị mã hóa dữ liệu ba chiều có thể chia thông tin hình học và thông tin thuộc tính một cách riêng biệt.

Khi việc chia lát được thực hiện (Có ở S5014), thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học được chia và các đoạn của thông tin thuộc tính được chia bằng cách chia các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh (hoặc thông tin hình học và thông tin thuộc tính) một cách riêng biệt (S5015). Ngoài ra, thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính liên quan đến việc chia lát. Cần lưu ý rằng thiết bị mã hóa dữ liệu ba chiều có thể chia thông tin hình học mảnh và thông tin thuộc tính mảnh một cách tập hợp.

Tiếp theo, thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học được mã hóa và các đoạn của thông tin thuộc tính được mã hóa nhờ lần lượt mã hóa các đoạn của thông tin hình học được chia và các đoạn của thông tin thuộc tính được chia (S5016). Ngoài ra, thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin mối tương quan phụ thuộc.

Cuối cùng, thiết bị mã hóa dữ liệu ba chiều tạo ra dữ liệu được mã hóa (dòng được mã hóa) nhờ lưu trữ trong các bộ phận NAL (đa hợp) các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung (S5017). Ngoài ra, thiết bị mã hóa dữ liệu ba chiều truyền dữ liệu được mã hóa được tạo ra.

Fig.68 là lưu đồ của quy trình giải mã dữ liệu đám mây điểm theo phương án này. Thứ nhất, thiết bị giải mã dữ liệu ba chiều xác định phương pháp phân chia nhờ phân tích thông tin bổ sung (thông tin bổ sung mảnh, thông tin bổ sung lát hình học, thông tin bổ sung lát thuộc tính) liên quan đến phương pháp phân chia được bao gồm trong dữ liệu được mã hóa (dòng được mã hóa) (S5021). Các ví dụ về phương pháp phân chia bao gồm việc chia mảnh và việc chia lát. Phương pháp phân chia có thể bao gồm số phân chia, loại phân chia, v.v., khi việc chia mảnh hoặc việc chia lát được thực hiện.

Tiếp theo, thiết bị giải mã dữ liệu ba chiều tạo ra thông tin hình học được chia và thông tin thuộc tính được chia bằng cách giải mã các đoạn của thông tin hình học được mã

hóa và các đoạn của thông tin thuộc tính được mã hóa được bao gồm trong dữ liệu được mã hóa, nhờ sử dụng thông tin mối tương quan phụ thuộc được bao gồm trong dữ liệu được mã hóa (S5022).

Khi thông tin bổ sung chỉ báo rằng việc chia lát đã được thực hiện (Có ở S5023), thiết bị giải mã dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách kết hợp các đoạn của thông tin hình học được chia và kết hợp các đoạn của thông tin thuộc tính được chia, nhờ sử dụng các phương pháp tương ứng, dựa vào thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính (S5024). Cần lưu ý rằng thiết bị giải mã dữ liệu ba chiều có thể kết hợp các đoạn của thông tin hình học được chia và kết hợp các đoạn của thông tin thuộc tính được chia, nhờ sử dụng phương pháp như nhau.

Khi thông tin bổ sung chỉ báo rằng việc chia mảnh đã được thực hiện (Có ở S5025), thiết bị giải mã dữ liệu ba chiều tạo ra thông tin hình học và thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin hình học mảnh (các đoạn của thông tin hình học được chia) và kết hợp các đoạn của thông tin thuộc tính mảnh (các đoạn của thông tin thuộc tính được chia), nhờ sử dụng phương pháp như nhau, dựa vào thông tin bổ sung mảnh (S5026). Cần lưu ý rằng thiết bị giải mã dữ liệu ba chiều có thể kết hợp các đoạn của thông tin hình học mảnh và kết hợp các đoạn của thông tin thuộc tính mảnh, nhờ sử dụng các phương pháp tương ứng.

Phần dưới đây mô tả thông tin bổ sung mảnh. Thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin bổ sung mảnh mà là siêu dữ liệu liên quan đến phương pháp chia mảnh, và truyền thông tin bổ sung được tạo ra mảnh tới thiết bị giải mã dữ liệu ba chiều.

Fig.69 là sơ đồ minh họa ví dụ về cú pháp của thông tin bổ sung mảnh (TileMetaData). Như được thể hiện trên Fig.69, ví dụ, thông tin bổ sung mảnh bao gồm thông tin phương pháp phân chia (type_of_divide), thông tin hình dạng (hình dạng nhìn từ bên trên), cờ chồng lấn (tile_overlap_flag), thông tin chồng lấn (type_of_overlap), thông tin độ cao (tile_height), số mảnh (tile_number), và thông tin vị trí mảnh (global_position, relative_position).

Thông tin phương pháp phân chia (type_of_divide) chỉ báo phương pháp chia mảnh. Ví dụ, thông tin phương pháp phân chia chỉ báo xem phương pháp chia mảnh là việc phân chia dựa vào thông tin bản đồ hay không, nghĩa là, việc phân chia dựa vào hình chiếu từ trên xuống (top_view) hoặc việc phân chia khác (còn lại).

Thông tin hình dạng (hình dạng nhìn từ bên trên) được bao gồm trong thông tin bổ sung mảnh khi phương pháp chia mảnh là, ví dụ, việc phân chia dựa vào hình chiếu từ trên xuống. Thông tin hình dạng chỉ báo hình dạng theo hình chiếu từ trên xuống của mảnh. Các ví dụ về hình dạng bao gồm hình vuông và hình tròn. Hơn nữa, các ví dụ về hình dạng có thể bao gồm hình elip, hình chữ nhật, hoặc hình đa giác khác với hình tứ giác, hoặc có thể bao gồm hình dạng khác với chúng. Cần lưu ý rằng thông tin hình dạng có thể chỉ báo không chỉ hình dạng theo hình chiếu từ trên xuống của mảnh mà còn hình dạng ba chiều (chẳng hạn, hình lập phương, cột tròn) của mảnh.

Cờ chồng lấn (tile_overlap_flag) chỉ báo xem các mảnh chồng lấn với nhau hay không. Ví dụ, cờ chồng lấn được bao gồm trong thông tin bổ sung mảnh khi phương pháp chia mảnh là việc phân chia dựa vào hình chiếu từ trên xuống. Trong trường hợp này, cờ chồng lấn chỉ báo xem các mảnh chồng lấn với nhau hay không theo hình chiếu từ trên xuống. Cần lưu ý rằng cờ chồng lấn có thể chỉ báo xem các mảnh chồng lấn với nhau hay không theo không gian ba chiều.

Thông tin chồng lấn (type_of_overlap) được bao gồm trong thông tin bổ sung mảnh khi, ví dụ, các mảnh chồng lấn với nhau. Thông tin chồng lấn chỉ báo, ví dụ, cách thức các mảnh chồng lấn với nhau như thế nào. Ví dụ, thông tin chồng lấn chỉ báo kích thước của khu vực chồng lấn.

Thông tin độ cao (tile_height) chỉ báo độ cao của mảnh. Cần lưu ý rằng thông tin độ cao có thể bao gồm thông tin chỉ báo hình dạng mảnh. Ví dụ, khi hình dạng của mảnh theo hình chiếu từ trên xuống là hình chữ nhật, thông tin có thể chỉ báo độ dài của phía (độ dài theo chiều dọc, độ dài theo chiều ngang) của hình chữ nhật. Khi hình dạng của mảnh theo hình chiếu từ trên xuống là hình tròn, thông tin có thể chỉ báo đường kính hoặc bán kính của hình tròn.

Hơn nữa, thông tin độ cao có thể chỉ báo độ cao của mỗi mảnh hoặc độ cao chung cho các mảnh. Ngoài ra, các loại độ cao chẳng hạn như các tuyến đường và các cầu vượt có thể được thiết đặt trước, và thông tin độ cao có thể chỉ báo độ cao của mỗi trong số các loại độ cao và độ cao loại của mỗi mảnh. Theo cách khác độ cao của mỗi loại độ cao có thể được quy định trước, và thông tin độ cao có thể chỉ báo độ cao loại của mỗi mảnh. Nói cách khác, thông tin độ cao không cần chỉ báo độ cao của mỗi loại độ cao.

Số mảnh (tile_number) chỉ báo số của các mảnh. Cần lưu ý rằng thông tin bổ sung mảnh có thể bao gồm thông tin chỉ báo khoảng cách giữa các mảnh.

Thông tin vị trí mảnh (global_position, relative_position) là thông tin dùng để nhận dạng vị trí của mỗi mảnh. Ví dụ, thông tin vị trí mảnh chỉ báo các tọa độ tuyệt đối hoặc các tọa độ liên quan của mỗi mảnh.

Cần lưu ý rằng một phần hoặc tất cả của thông tin nêu trên có thể được cung cấp đối với mỗi mảnh hoặc mỗi nhóm của các mảnh (chẳng hạn, đối với mỗi khung hoặc nhóm của các khung).

Thiết bị mã hóa dữ liệu ba chiều có thể bao gồm thông tin bổ sung mảnh trong thông tin nâng cao phụ (SEI) và truyền SEI. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể lưu trữ thông tin bổ sung mảnh trong tập hợp thông số hiện thời (PPS, GPS, hoặc APS, v.v.) và truyền tập hợp thông số.

Ví dụ, khi thông tin bổ sung mảnh thay đổi đối với mỗi khung, thông tin bổ sung mảnh có thể được lưu trữ trong tập hợp thông số đối với mỗi khung (GPS hoặc APS v.v.). Khi thông tin bổ sung mảnh không thay đổi theo trình tự, thông tin bổ sung mảnh có thể được lưu trữ trong tập hợp thông số đối với trình tự (hình học SPS hoặc thuộc tính SPS). Hơn nữa, khi thông tin phân mảnh như nhau được sử dụng đối với thông tin hình học và thông tin thuộc tính, thông tin bổ sung mảnh có thể được lưu trữ trong tập hợp thông số đối với dòng PCC (dòng PS).

Hơn nữa, thông tin bổ sung mảnh có thể được lưu trữ trong bất kỳ một trong số tập hợp thông số nêu trên hoặc trong các tập hợp thông số. Ngoài ra, thông tin bổ sung mảnh có thể được lưu trữ trong đoạn đầu của dữ liệu được mã hóa. Ngoài ra, thông tin bổ sung mảnh có thể được lưu trữ trong đoạn đầu của bộ phận NAL.

Hơn nữa, một phần hoặc tất cả của thông tin bổ sung mảnh có thể được lưu trữ trong một trong số đoạn đầu của thông tin hình học được chia và đoạn đầu của thông tin thuộc tính được chia, và không cần được lưu trữ ở phần khác. Ví dụ, khi thông tin bổ sung mảnh như nhau được sử dụng đối với thông tin hình học và thông tin thuộc tính, thông tin bổ sung mảnh có thể được bao gồm trong đoạn đầu của một trong số thông tin hình học và thông tin thuộc tính. Ví dụ, khi thông tin thuộc tính tùy thuộc vào thông tin hình học, thông tin hình học được xử lý đầu tiên. Vì lý do này, thông tin bổ sung mảnh có thể được bao gồm trong đoạn đầu của thông tin hình học, và không cần được bao gồm trong đoạn đầu của thông tin thuộc tính. Trong trường hợp này, ví dụ, thiết bị giải mã dữ liệu ba chiều xác định rằng thông tin thuộc tính của gốc tham chiếu thuộc về mảnh giống như mảnh có thông tin hình học của đích tham chiếu.

Thiết bị giải mã dữ liệu ba chiều tái cấu trúc dữ liệu đám mây điểm được trải qua việc chia mảnh, dựa vào thông tin bổ sung mảnh. Khi có các đoạn dữ liệu đám mây điểm chồng lấn, thiết bị giải mã dữ liệu ba chiều định rõ các đoạn dữ liệu đám mây điểm chồng lấn và lựa chọn một trong số các đoạn dữ liệu đám mây điểm chồng lấn hoặc hợp nhất các đoạn dữ liệu đám mây điểm.

Hơn nữa, thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã nhờ sử dụng thông tin bổ sung mảnh. Ví dụ, khi các mảnh chồng lấn với nhau, thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã đối với mỗi mảnh, thực hiện việc xử lý (chẳng hạn, làm nhẵn hoặc lọc) nhờ sử dụng các đoạn dữ liệu được giải mã, và tạo ra dữ liệu đám mây điểm. Điều này khiến có thể thực hiện việc giải mã chính xác cao.

Fig.70 là sơ đồ minh họa ví dụ cấu hình của hệ thống bao gồm thiết bị mã hóa dữ liệu ba chiều và thiết bị giải mã dữ liệu ba chiều. Bộ chia mảnh 5051 chia dữ liệu đám mây điểm bao gồm thông tin hình học và thông tin thuộc tính thành mảnh thứ nhất và mảnh thứ hai. Ngoài ra, bộ chia mảnh 5051 truyền thông tin bổ sung mảnh liên quan đến việc chia mảnh tới bộ giải mã 5053 và bộ kết hợp mảnh 5054.

Bộ mã hóa 5052 tạo ra dữ liệu được mã hóa bằng cách mã hóa mảnh thứ nhất và mảnh thứ hai.

Bộ giải mã 5053 khôi phục mảnh thứ nhất và mảnh thứ hai bằng cách giải mã dữ liệu được mã hóa được tạo ra bởi bộ mã hóa 5052. Bộ kết hợp mảnh 5054 khôi phục dữ liệu đám mây điểm (thông tin hình học và thông tin thuộc tính) bằng cách kết hợp mảnh thứ nhất và mảnh thứ hai nhờ sử dụng thông tin bổ sung mảnh.

Phần dưới đây mô tả thông tin bổ sung lát. Thiết bị mã hóa dữ liệu ba chiều tạo ra thông tin bổ sung lát mà là siêu dữ liệu liên quan đến phương pháp chia lát, và truyền thông tin bổ sung được tạo ra lát tới thiết bị giải mã dữ liệu ba chiều.

Fig.71 là sơ đồ minh họa ví dụ về cú pháp của thông tin bổ sung lát (SliceMetaData). Như được thể hiện trên Fig.71, ví dụ, thông tin bổ sung lát bao gồm thông tin phương pháp phân chia (type_of_divide), cờ chồng lấn (slice_overlap_flag), thông tin chồng lấn (type_of_overlap), số lát (slice_number), thông tin vị trí lát (global_position, relative_position), và thông tin kích thước lát (slice_bounding_box_size).

Thông tin phương pháp phân chia (type_of_divide) chỉ báo phương pháp chia lát. Ví dụ, thông tin phương pháp phân chia chỉ báo xem phương pháp chia lát là việc phân

chia dựa vào thông tin về đối tượng (đối tượng) như được thể hiện trên Fig.61 hay không. Cần lưu ý rằng, thông tin bổ sung lát có thể bao gồm thông tin chỉ báo đối tượng phương pháp phân chia. Ví dụ, thông tin này chỉ báo xem một đối tượng được chia thành các lát hoặc được gán tới một lát hay không. Ngoài ra, thông tin có thể chỉ báo, ví dụ, số phân chia khi một đối tượng được chia thành các lát.

Cờ chồng lấn (slice_overlap_flag) chỉ báo xem các lát chồng lấn với nhau hay không. Thông tin chồng lấn (type_of_overlap) được bao gồm trong thông tin bổ sung lát khi, ví dụ, các lát chồng lấn với nhau. Thông tin chồng lấn chỉ báo, ví dụ, các lát chồng lấn với nhau như thế nào. Ví dụ, thông tin chồng lấn chỉ báo kích thước của khu vực chồng lấn.

Số lát (slice_number) chỉ báo số của các lát.

Thông tin vị trí lát (global_position, relative_position) và thông tin kích thước lát (slice_bounding_box_size) là thông tin về vùng của lát. Thông tin vị trí lát là thông tin dùng để nhận dạng vị trí của mỗi lát. Ví dụ, thông tin vị trí lát chỉ báo các tọa độ tuyệt đối hoặc các tọa độ liên quan của mỗi lát. Thông tin kích thước lát (slice_bounding_box_size) chỉ báo kích thước của mỗi lát. Ví dụ, thông tin kích thước lát chỉ báo kích thước của ô vuông giới hạn của mỗi lát.

Thiết bị mã hóa dữ liệu ba chiều có thể bao gồm thông tin bổ sung lát ở SEI và truyền SEI. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể lưu trữ thông tin bổ sung lát trong tập hợp thông số hiện thời (PPS, GPS, hoặc APS, v.v.) và truyền tập hợp thông số.

Ví dụ, khi thông tin bổ sung lát thay đổi đối với mỗi khung, thông tin bổ sung lát có thể được lưu trữ trong tập hợp thông số đối với mỗi khung (GPS hoặc APS v.v.). Khi thông tin bổ sung lát không thay đổi theo trình tự, thông tin bổ sung lát có thể được lưu trữ trong tập hợp thông số đối với trình tự (hình học SPS hoặc thuộc tính SPS). Hơn nữa, khi thông tin chia lát được sử dụng đối với thông tin hình học và thông tin thuộc tính, thông tin bổ sung lát có thể được lưu trữ trong tập hợp thông số đối với dòng PCC (dòng PS).

Hơn nữa, thông tin bổ sung lát có thể được lưu trữ trong bất kỳ một trong số tập hợp thông số nêu trên hoặc trong tập hợp thông số. Ngoài ra, thông tin bổ sung lát có thể được lưu trữ trong đoạn đầu của dữ liệu được mã hóa. Ngoài ra, thông tin bổ sung lát có thể được lưu trữ trong đoạn đầu của bộ phận NAL.

Hơn nữa, một phần hoặc tất cả của thông tin bổ sung lát có thể được lưu trữ trong một trong số đoạn đầu của thông tin hình học được chia và đoạn đầu của thông tin thuộc tính được chia, và không cần được lưu trữ ở phần khác. Ví dụ, khi thông tin bổ sung lát như nhau được sử dụng đối với thông tin hình học và thông tin thuộc tính, thông tin bổ sung lát có thể được bao gồm trong đoạn đầu của một trong số thông tin hình học và thông tin thuộc tính. Ví dụ, khi thông tin thuộc tính tùy thuộc vào thông tin hình học, thông tin hình học được xử lý đầu tiên. Vì lý do này, thông tin bổ sung lát có thể được bao gồm trong đoạn đầu của thông tin hình học, và không cần được bao gồm trong đoạn đầu của thông tin thuộc tính. Trong trường hợp này, ví dụ, thiết bị giải mã dữ liệu ba chiều xác định rằng thông tin thuộc tính của gốc tham chiếu thuộc về cùng lát như lát có thông tin hình học của đích tham chiếu.

Thiết bị giải mã dữ liệu ba chiều tái cấu trúc dữ liệu đám mây điểm được trải qua việc chia lát, dựa vào thông tin bổ sung lát. Khi có các đoạn dữ liệu đám mây điểm chồng lấn, thiết bị giải mã dữ liệu ba chiều định rõ các đoạn dữ liệu đám mây điểm chồng lấn và lựa chọn một trong số các đoạn dữ liệu đám mây điểm chồng lấn hoặc hợp nhất các đoạn dữ liệu đám mây điểm.

Hơn nữa, thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã nhờ sử dụng thông tin bổ sung lát. Ví dụ, khi các lát chồng lấn với nhau, thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã đối với mỗi lát, thực hiện việc xử lý (chẳng hạn, làm nhẵn hoặc lọc) nhờ sử dụng các đoạn dữ liệu được giải mã, và tạo ra dữ liệu đám mây điểm. Điều này khiến có thể thực hiện việc giải mã chính xác cao.

Fig.72 là lưu đồ của quy trình mã hóa dữ liệu ba chiều bao gồm quy trình tạo thông tin bổ sung mảnh được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều theo phương án này.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định phương pháp phân chia được sử dụng (S5031). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định xem phương pháp phân chia dựa vào hình chiếu từ trên xuống (top_view) hoặc phương pháp khác (còn lại) được sử dụng như phương pháp chia mảnh hay không. Ngoài ra, thiết bị mã hóa dữ liệu ba chiều xác định hình dạng mảnh khi phương pháp phân chia dựa vào hình chiếu từ trên xuống được sử dụng. Ngoài ra, thiết bị mã hóa dữ liệu ba chiều xác định xem các mảnh chồng lấn với các mảnh khác.

Khi phương pháp chia mảnh được xác định ở bước S5031 là phương pháp phân chia dựa vào hình chiếu từ trên xuống (Có ở S5032), thiết bị mã hóa dữ liệu ba chiều bao

gồm kết quả của việc xác định rằng phương pháp chia mảnh là phương pháp phân chia dựa vào hình chiếu từ trên xuống (top_view), trong thông tin bổ sung mảnh (S5033).

Mặt khác, khi phương pháp chia mảnh được xác định ở bước S5031 là phương pháp khác với phương pháp phân chia dựa vào hình chiếu từ trên xuống (Không ở S5032), thiết bị mã hóa dữ liệu ba chiều bao gồm kết quả của việc xác định rằng phương pháp chia mảnh là phương pháp khác với phương pháp phân chia dựa vào hình chiếu từ trên xuống (top_view), trong thông tin bổ sung mảnh (S5034).

Hơn nữa, khi hình dạng theo hình chiếu từ trên xuống của mảnh được xác định ở bước S5031 là hình vuông (SQUARE ở S5035), thiết bị mã hóa dữ liệu ba chiều bao gồm kết quả của việc xác định rằng hình dạng theo hình chiếu từ trên xuống của mảnh là hình vuông, trong thông tin bổ sung mảnh (S5036). Ngược lại, khi hình dạng theo hình chiếu từ trên xuống của mảnh được xác định ở bước S5031 là hình tròn (CIRCLE ở S5035), thiết bị mã hóa dữ liệu ba chiều bao gồm kết quả của việc xác định rằng hình dạng theo hình chiếu từ trên xuống của mảnh là hình tròn, trong thông tin bổ sung mảnh (S5037).

Tiếp theo, thiết bị mã hóa dữ liệu ba chiều xác định xem các mảnh chồng lấn với các mảnh khác (S5038) hay không. Khi các mảnh chồng lấn với các mảnh khác (Có ở S5038), thiết bị mã hóa dữ liệu ba chiều bao gồm kết quả của việc xác định rằng các mảnh chồng lấn với các mảnh khác, trong thông tin bổ sung mảnh (S5039). Mặt khác, khi các mảnh không chồng lấn với các mảnh khác (Không ở S5038), thiết bị mã hóa dữ liệu ba chiều bao gồm kết quả của việc xác định rằng các mảnh không chồng lấn với các mảnh khác, trong thông tin bổ sung mảnh (S5040).

Cuối cùng, thiết bị mã hóa dữ liệu ba chiều chia các mảnh dựa vào phương pháp chia mảnh được xác định ở bước S5031, mã hóa mỗi trong số các mảnh, và truyền dữ liệu được mã hóa được tạo ra và thông tin bổ sung mảnh (S5041).

Fig.73 là lưu đồ của quy trình giải mã dữ liệu ba chiều được thực hiện bởi thiết bị giải mã dữ liệu ba chiều theo phương án này nhờ sử dụng thông tin bổ sung mảnh.

Thứ nhất, thiết bị giải mã dữ liệu ba chiều phân tích thông tin bổ sung mảnh được bao gồm trong dòng bit (S5051).

Khi thông tin bổ sung mảnh chỉ báo rằng các mảnh không chồng lấn với các mảnh khác (Không ở S5052), thiết bị giải mã dữ liệu ba chiều tạo ra dữ liệu đám mây điểm của mỗi mảnh bằng cách giải mã mảnh (S5053). Cuối cùng, thiết bị giải mã dữ liệu ba chiều

tái cấu trúc dữ liệu đám mây điểm từ dữ liệu đám mây điểm của mỗi mảnh, dựa vào phương pháp chia mảnh và hình dạng mảnh được chỉ báo bởi thông tin bổ sung mảnh (S5054).

Ngược lại, khi thông tin bổ sung mảnh chỉ báo rằng các mảnh chồng lấn với các mảnh khác (Có ở S5052), thiết bị giải mã dữ liệu ba chiều tạo ra dữ liệu đám mây điểm của mỗi mảnh bằng cách giải mã mảnh. Ngoài ra, thiết bị giải mã dữ liệu ba chiều nhận dạng các phần chồng lấn của các mảnh dựa vào thông tin bổ sung mảnh (S5055). Cần lưu ý rằng, liên quan đến các phần chồng lấn, thiết bị giải mã dữ liệu ba chiều có thể thực hiện việc giải mã nhờ sử dụng các đoạn của thông tin chồng lấn. Cuối cùng, thiết bị giải mã dữ liệu ba chiều tái cấu trúc dữ liệu đám mây điểm từ dữ liệu đám mây điểm của mỗi mảnh, dựa vào phương pháp chia mảnh, hình dạng mảnh, và thông tin chồng lấn được chỉ báo bởi thông tin bổ sung mảnh (S5056).

Phần dưới đây mô tả, ví dụ, các biến thể liên quan đến lát. Thiết bị mã hóa dữ liệu ba chiều có thể truyền, như thông tin bổ sung, thông tin chỉ báo loại (tuyến đường, công trình xây dựng, cây, v.v.) hoặc thuộc tính (thông tin động, thông tin tĩnh, v.v.) của đối tượng. Theo cách khác thông số tạo mã có thể được xác định trước theo đối tượng, và thiết bị mã hóa dữ liệu ba chiều có thể thông báo thông số tạo mã tới thiết bị giải mã dữ liệu ba chiều bằng cách truyền loại hoặc thuộc tính của đối tượng.

Các phương pháp sau đây có thể được sử dụng liên quan đến thứ tự mã hóa dữ liệu lát và thứ tự truyền. Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể mã hóa dữ liệu lát theo thứ tự giảm dần nhằm dễ dàng cho nhận diện đối tượng hoặc việc tạo chùm. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể mã hóa dữ liệu lát theo thứ tự trong đó việc tạo chùm được kết thúc. Hơn nữa, thiết bị mã hóa dữ liệu ba chiều có thể truyền dữ liệu lát theo thứ tự trong đó dữ liệu lát được mã hóa. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể truyền dữ liệu lát theo thứ tự giảm dần của quyền ưu tiên dùng để giải mã trong ứng dụng. Ví dụ, khi thông tin động có quyền ưu tiên cao dùng để giải mã, thiết bị mã hóa dữ liệu ba chiều có thể truyền dữ liệu lát theo thứ tự trong đó các lát được tạo nhóm nhờ sử dụng thông tin động.

Hơn nữa, khi thứ tự dữ liệu được mã hóa là khác với thứ tự của quyền ưu tiên dùng để giải mã, thiết bị mã hóa dữ liệu ba chiều có thể truyền dữ liệu được mã hóa sau khi bố trí lại dữ liệu được mã hóa. Ngoài ra, khi lưu trữ dữ liệu được mã hóa, thiết bị mã hóa dữ liệu ba chiều có thể lưu trữ dữ liệu được mã hóa sau khi bố trí lại dữ liệu được mã hóa.

Ứng dụng (thiết bị giải mã dữ liệu ba chiều) yêu cầu máy chủ (thiết bị mã hóa dữ

liệu ba chiều) để truyền các lát bao gồm dữ liệu mong muốn. Máy chủ có thể truyền lát dữ liệu được yêu cầu bởi ứng dụng, và không cần truyền dữ liệu lát không cần thiết đối với ứng dụng.

Ứng dụng yêu cầu máy chủ để truyền mảnh bao gồm dữ liệu mong muốn. Máy chủ có thể truyền mảnh dữ liệu được yêu cầu bởi ứng dụng, và không cần truyền dữ liệu mảnh không cần thiết đối với ứng dụng.

Như được nêu trên, thiết bị mã hóa dữ liệu ba chiều theo phương án này thực hiện quy trình được thể hiện trên Fig.74. Thứ nhất, thiết bị mã hóa dữ liệu ba chiều mã hóa các không gian con (chẳng hạn, các mảnh) được thu nhận bằng cách chia không gian hiện thời mà bao gồm các điểm ba chiều, để tạo ra các đoạn dữ liệu được mã hóa (S5061). Thiết bị mã hóa dữ liệu ba chiều tạo ra dòng bit bao gồm các đoạn dữ liệu được mã hóa và thông tin thứ nhất (chẳng hạn, hình dạng nhìn từ bên trên (topview_shape)) chỉ báo hình dạng của mỗi trong số các không gian con (S5062).

Theo đó, vì thiết bị mã hóa dữ liệu ba chiều có thể lựa chọn hình dạng bất kỳ từ các loại khác nhau của các hình dạng của các không gian con, thiết bị mã hóa dữ liệu ba chiều có thể nâng cao hiệu quả mã hóa.

Ví dụ, hình dạng là hình dạng hai chiều hoặc hình dạng ba chiều của mỗi trong số các không gian con. Ví dụ, hình dạng là hình dạng theo hình chiếu từ trên xuống của không gian con. Theo cách khác, thông tin thứ nhất chỉ báo hình dạng của không gian con được nhìn từ hướng cụ thể (chẳng hạn, hướng bên trên). Tóm lại, thông tin thứ nhất chỉ báo hình dạng trên hình vẽ nhìn từ bên trên của không gian con. Ví dụ, hình dạng là hình chữ nhật hoặc hình tròn.

Ví dụ, dòng bit bao gồm thông tin thứ hai (chẳng hạn, cờ chồng lần mảnh (tile_overlap_flag)) chỉ báo xem các không gian con chồng lấn hay không.

Theo đó, vì thiết bị mã hóa dữ liệu ba chiều cho phép các không gian con chồng lấn, thiết bị mã hóa dữ liệu ba chiều có thể tạo ra các không gian con mà không tạo nên hình dạng của mỗi trong số tổ hợp các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ ba (chẳng hạn, loại chia (type_of_divide)) chỉ báo xem phương pháp phân chia được sử dụng để thu nhận các không gian con là phương pháp phân chia nhờ sử dụng hình chiếu từ trên xuống hay không.

Ví dụ, dòng bit bao gồm thông tin thứ tư (chẳng hạn, độ cao mảnh (tile_height))

chỉ báo ít nhất một trong số độ cao, độ rộng, độ sâu, hoặc bán kính của mỗi trong số các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ năm (chẳng hạn, vị trí toàn cầu (global_position) hoặc vị trí liên quan (relative_position)) chỉ báo vị trí của mỗi trong số các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ sáu (chẳng hạn, số mảnh (tile_number)) chỉ báo tổng số của các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ bảy chỉ báo khoảng cách giữa các không gian con.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Hơn nữa, thiết bị giải mã dữ liệu ba chiều theo phương án này thực hiện quy trình được thể hiện trên Fig.75. Thứ nhất, thiết bị giải mã dữ liệu ba chiều giải mã các đoạn dữ liệu được mã hóa được bao gồm trong dòng bit và được tạo ra bằng cách mã hóa các không gian con (chẳng hạn, các mảnh) được thu nhận bằng cách chia không gian hiện thời mà bao gồm các điểm ba chiều, để khôi phục các không gian con (S5071). Thiết bị giải mã dữ liệu ba chiều khôi phục không gian hiện thời bằng cách kết hợp các không gian con nhờ sử dụng thông tin thứ nhất (chẳng hạn, hình dạng nhìn từ bên trên) mà được bao gồm trong dòng bit và chỉ báo hình dạng của mỗi trong số các không gian con (S5072). Ví dụ, thiết bị giải mã dữ liệu ba chiều có thể xác định vị trí và khoảng của mỗi trong số các không gian con theo không gian hiện thời bằng cách nhận diện hình dạng của không gian con nhờ sử dụng thông tin thứ nhất. Thiết bị giải mã dữ liệu ba chiều có thể kết hợp các không gian con dựa vào các vị trí và các khoảng được xác định của các không gian con. Theo đó, thiết bị giải mã dữ liệu ba chiều có thể kết hợp các không gian con một cách chính xác.

Ví dụ, hình dạng là hình dạng hai chiều hoặc hình dạng ba chiều của mỗi trong số các không gian con. Ví dụ, hình dạng là hình chữ nhật hoặc hình tròn.

Ví dụ, dòng bit bao gồm thông tin thứ hai (chẳng hạn, tile_overlap_flag) chỉ báo xem các không gian con chồng lấn hay không. Trong việc khôi phục của không gian hiện thời, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con bởi còn nhờ sử dụng thông tin thứ hai. Ví dụ, thiết bị giải mã dữ liệu ba chiều xác định xem các không gian con chồng lấn hay không, nhờ sử dụng thông tin thứ hai. Khi các không gian con chồng lấn,

thiết bị giải mã dữ liệu ba chiều nhận dạng các khu vực chồng lấn và thực hiện quy trình được xác định trước trên các khu vực được nhận dạng.

Ví dụ, dòng bit bao gồm thông tin thứ ba (chẳng hạn, type_of_divide) chỉ báo xem phương pháp phân chia được sử dụng để thu nhận các không gian con là phương pháp phân chia nhờ sử dụng hình chiếu từ trên xuống hay không. Trong việc khôi phục của không gian hiện thời, khi thông tin thứ ba chỉ báo rằng phương pháp phân chia được sử dụng để thu nhận các không gian con là phương pháp phân chia nhờ sử dụng hình chiếu từ trên xuống, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con nhờ sử dụng thông tin thứ nhất.

Ví dụ, dòng bit bao gồm thông tin thứ tư (chẳng hạn, tile_height) chỉ báo ít nhất một trong số độ cao, độ rộng, độ sâu, hoặc bán kính của mỗi trong số các không gian con. Trong việc khôi phục của không gian hiện thời, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con bởi còn nhờ sử dụng thông tin thứ tư. Ví dụ, thiết bị giải mã dữ liệu ba chiều có thể xác định vị trí và khoảng của mỗi trong số các không gian con theo không gian hiện thời bằng cách nhận diện độ cao của không gian con nhờ sử dụng thông tin thứ tư. Thiết bị giải mã dữ liệu ba chiều có thể kết hợp các không gian con dựa vào các vị trí và các khoảng được xác định của các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ năm (chẳng hạn, global_position hoặc relative_position) chỉ báo vị trí của mỗi trong số các không gian con. Trong việc khôi phục của không gian hiện thời, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con bởi còn nhờ sử dụng thông tin thứ năm. Ví dụ, thiết bị giải mã dữ liệu ba chiều có thể xác định vị trí của mỗi trong số các không gian con theo không gian hiện thời bằng cách nhận diện vị trí của không gian con nhờ sử dụng thông tin thứ năm. Thiết bị giải mã dữ liệu ba chiều có thể kết hợp các không gian con dựa vào các vị trí được xác định của các không gian con.

Ví dụ, dòng bit bao gồm thông tin thứ sáu (chẳng hạn, tile_number) chỉ báo tổng số của các không gian con. Trong việc khôi phục của không gian hiện thời, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con bởi còn nhờ sử dụng thông tin thứ sáu.

Ví dụ, dòng bit bao gồm thông tin thứ bảy chỉ báo khoảng cách giữa các không gian con. Trong việc khôi phục của không gian hiện thời, thiết bị giải mã dữ liệu ba chiều kết hợp các không gian con bởi còn nhờ sử dụng thông tin thứ bảy. Ví dụ, thiết bị giải mã dữ liệu ba chiều có thể xác định vị trí và khoảng của mỗi trong số các không gian con theo không gian hiện thời bằng cách nhận diện khoảng cách giữa các không gian con nhờ sử

dụng thông tin thứ bảy. Thiết bị giải mã dữ liệu ba chiều có thể kết hợp các không gian con dựa vào các vị trí và các khoảng được xác định của các không gian con.

Ví dụ, thiết bị giải mã dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Phương án 7

Để chia dữ liệu đám mây điểm thành các mảnh và các lát và mã hóa hoặc giải mã một cách hữu hiệu dữ liệu phân chia, việc điều khiển thích hợp được cần đến ở phía bộ mã hóa và phía bộ giải mã. Bằng cách thực hiện việc mã hóa và giải mã của mỗi đoạn dữ liệu phân chia độc lập, thay vì phụ thuộc, từ các đoạn khác của dữ liệu phân chia, bộ xử lý đa lõi hoặc đa luồng có thể được sử dụng để xử lý các đoạn dữ liệu phân chia trong các luồng/các lõi tương ứng song song, và hiệu suất được nâng cao.

Có các phương pháp khác nhau về việc chia dữ liệu đám mây điểm thành các mảnh và các lát. Ví dụ, có phương pháp chia dữ liệu đám mây điểm dựa vào thuộc tính của đối tượng, chẳng hạn như bề mặt đường, của dữ liệu đám mây điểm hoặc đặc tính, chẳng hạn như thông tin màu sắc chẳng hạn như màu xanh da trời, của dữ liệu đám mây điểm.

CABAC là từ viết tắt của tạo mã số học nhị phân thích ứng ngữ cảnh (context-based adaptive binary arithmetic coding), mà là phương pháp mã hóa mà thực hiện việc mã hóa số học (mã hóa entrôpi) với tỷ lệ nén cao bằng cách làm tăng xác suất chính xác nhờ cập nhật thành công ngữ cảnh (mô hình đánh giá xác suất xuất hiện của ký hiệu nhị phân đầu vào) dựa vào thông tin được mã hóa .

Để xử lý các đoạn dữ liệu phân chia chẳng hạn như các mảnh hoặc các lát song song, mỗi đoạn dữ liệu phân chia cần được mã hóa hoặc được giải mã một cách độc lập. Để tạo nên CABAC cho các đoạn dữ liệu phân chia độc lập với nhau, CABAC cần được khởi tạo ở đỉnh của mỗi đoạn dữ liệu phân chia. Tuy nhiên, không có cơ chế cho nó.

Cờ khởi tạo CABAC được sử dụng để khởi tạo CABAC trong việc mã hóa và giải mã CABAC.

Fig.76 là lưu đồ của quy trình khởi tạo CABAC đáp lại cờ khởi tạo CABAC.

Thiết bị mã hóa dữ liệu ba chiều hoặc thiết bị giải mã dữ liệu ba chiều xác định xem cờ khởi tạo CABAC là 1 hay không trong việc mã hóa hoặc giải mã (S5201).

Khi cờ khởi tạo CABAC là 1 (nếu Có ở S5201), thiết bị mã hóa dữ liệu ba chiều hoặc thiết bị giải mã dữ liệu ba chiều khởi tạo CABAC bộ mã hóa/bộ giải mã tới trạng thái

mặc định (S5202), và tiếp tục mã hóa hoặc giải mã.

Khi cờ khởi tạo CABAC không phải 1 (nếu Không ở S5201), thiết bị mã hóa dữ liệu ba chiều hoặc thiết bị giải mã dữ liệu ba chiều không thực hiện việc khởi tạo, và tiếp tục mã hóa hoặc giải mã.

Nghĩa là, khi khởi tạo CABAC, cabac_init_flag được thiết đặt tới 1, và CABAC bộ mã hóa hoặc bộ giải mã CABAC được khởi tạo hoặc được khởi tạo lại. Khi khởi tạo CABAC, trị số ban đầu (trạng thái mặc định) của ngữ cảnh được sử dụng đối với quy trình CABAC được thiết đặt.

Quy trình mã hóa sẽ được mô tả. Fig.77 là sơ đồ khối minh họa cấu hình của bộ mã hóa thứ nhất 5200 được bao gồm trong thiết bị mã hóa dữ liệu ba chiều theo phương án này. Fig.78 là sơ đồ khối minh họa cấu hình của bộ chia 5201 theo phương án này. Fig.79 là sơ đồ khối minh họa cấu hình của bộ mã hóa thông tin hình học 5202 và bộ mã hóa thông tin thuộc tính 5203 theo phương án này.

Bộ mã hóa thứ nhất 5200 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ nhất (PCC trên cơ sở hình học (GPCC)). Bộ mã hóa thứ nhất 5200 bao gồm bộ chia 5201, các bộ mã hóa thông tin hình học 5202, các bộ mã hóa thông tin các thuộc tính 5203, bộ mã hóa thông tin bổ sung 5204, và bộ đa hợp 5205.

Bộ chia 5201 tạo ra các đoạn dữ liệu phân chia bằng cách chia dữ liệu đám mây điểm. Cụ thể là, bộ chia 5201 tạo ra các đoạn dữ liệu phân chia bằng cách chia không gian của dữ liệu đám mây điểm thành các không gian con. Ở đây, không gian con là sự kết hợp của các mảnh hoặc các lát hoặc sự kết hợp của các mảnh và các lát. Cụ thể hơn là, dữ liệu đám mây điểm bao gồm thông tin hình học, thông tin thuộc tính, và thông tin bổ sung. Bộ chia 5201 chia thông tin hình học thành các đoạn của thông tin hình học phân chia, và chia thông tin thuộc tính thành các đoạn của thông tin thuộc tính phân chia. Bộ chia 5201 cũng tạo ra thông tin bổ sung liên quan đến việc phân chia.

Như được minh họa trên Fig.78, bộ chia 5201 bao gồm bộ chia mảnh 5211 và bộ chia lát 5212. Ví dụ, bộ chia mảnh 5211 chia đám mây điểm thành các mảnh. Bộ chia mảnh 5211 có thể xác định trị số lượng tử hóa được sử dụng đối với mỗi mảnh phân chia như thông tin bổ sung mảnh.

Bộ chia lát 5212 còn chia mảnh được thu nhận bởi bộ chia mảnh 5211 thành các

lát. Bộ chia lát 5212 có thể xác định trị số lượng tử hóa được sử dụng đối với mỗi lát phân chia như thông tin bổ sung lát.

Các bộ mã hóa thông tin hình học 5202 tạo ra các đoạn của thông tin hình học được mã hóa bằng cách mã hóa các đoạn của thông tin hình học phân chia. Ví dụ, các bộ mã hóa thông tin hình học 5202 xử lý các đoạn của thông tin hình học phân chia song song.

Như được minh họa trên Fig.79, bộ mã hóa thông tin hình học 5202 bao gồm bộ khởi tạo CABAC 5221 và bộ mã hóa entrôpi 5222. Bộ khởi tạo CABAC 5221 khởi tạo hoặc khởi tạo lại CABAC đáp lại cờ khởi tạo CABAC. Bộ mã hóa entrôpi 5222 mã hóa thông tin hình học phân chia theo CABAC.

Các bộ mã hóa thông tin các thuộc tính 5203 tạo ra các đoạn của thông tin thuộc tính được mã hóa bằng cách mã hóa các đoạn của thông tin thuộc tính phân chia. Ví dụ, các bộ mã hóa thông tin các thuộc tính 5203 xử lý các đoạn của thông tin thuộc tính phân chia song song.

Như được minh họa trên Fig.79, bộ mã hóa thông tin thuộc tính 5203 bao gồm bộ khởi tạo CABAC 5231 và bộ mã hóa entrôpi 5232. Bộ khởi tạo CABAC 5231 khởi tạo hoặc khởi tạo lại CABAC đáp lại cờ khởi tạo CABAC. Bộ mã hóa entrôpi 5232 mã hóa thông tin thuộc tính phân chia theo CABAC.

Bộ mã hóa thông tin bổ sung 5204 tạo ra thông tin bổ sung được mã hóa bằng cách mã hóa thông tin bổ sung được bao gồm trong dữ liệu đám mây điểm và thông tin bổ sung liên quan đến dữ liệu việc phân chia được tạo ra trong việc phân chia bởi bộ chia 5201.

Bộ đa hợp 5205 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách đa hợp các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa, và truyền dữ liệu được mã hóa được tạo ra. Thông tin bổ sung được mã hóa được sử dụng để giải mã.

Lưu ý rằng, mặc dù Fig.77 thể hiện ví dụ trong đó có hai bộ mã hóa thông tin hình học 5202 và hai bộ mã hóa thông tin các thuộc tính 5203, số của các bộ mã hóa thông tin hình học 5202 và số của bộ mã hóa thông tin các thuộc tính 5203 có thể là một, hoặc ba hoặc nhiều hơn. Các đoạn dữ liệu phân chia có thể được xử lý song song trong cùng chip, chẳng hạn như bởi các lõi của CPU, được xử lý song song bởi các lõi của các chip, hoặc được xử lý song song bởi các lõi của các chip.

Tiếp theo, quy trình giải mã sẽ được mô tả. Fig.80 là sơ đồ khối minh họa cấu hình

của bộ giải mã thứ nhất 5240. Fig.81 là sơ đồ khối minh họa cấu hình của bộ giải mã thông tin hình học 5242 và bộ giải mã thông tin thuộc tính 5243.

Bộ giải mã thứ nhất 5240 tái tạo dữ liệu đám mây điểm bằng cách giải mã dữ liệu được mã hóa (dòng được mã hóa) được tạo ra bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ nhất (GPCC). Bộ giải mã thứ nhất 5240 bao gồm bộ giải đa hợp 5241, các bộ giải mã thông tin hình học 5242, các bộ giải mã thông tin các thuộc tính 5243, bộ giải mã thông tin bổ sung 5244, và bộ kết hợp 5245.

Bộ giải đa hợp 5241 tạo ra các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa bằng cách giải đa hợp dữ liệu được mã hóa (dòng được mã hóa).

Các bộ giải mã thông tin hình học 5242 tạo ra các đoạn của thông tin hình học được lượng tử hóa bằng cách giải mã các đoạn của thông tin hình học được mã hóa. Ví dụ, các bộ giải mã thông tin hình học 5242 xử lý các đoạn của thông tin hình học được mã hóa song song.

Như được minh họa trên Fig.81, bộ giải mã thông tin hình học 5242 bao gồm bộ khởi tạo CABAC 5251 và bộ giải mã entrôpi 5252. Bộ khởi tạo CABAC 5251 khởi tạo hoặc khởi tạo lại CABAC đáp lại cờ khởi tạo CABAC. Bộ giải mã entrôpi 5252 giải mã thông tin hình học theo CABAC.

Các bộ giải mã thông tin các thuộc tính 5243 tạo ra các đoạn của thông tin thuộc tính phân chia bằng cách giải mã các đoạn của thông tin thuộc tính được mã hóa. Ví dụ, các bộ giải mã thông tin các thuộc tính 5243 xử lý các đoạn của thông tin thuộc tính được mã hóa song song.

Như được minh họa trên Fig.81, bộ giải mã thông tin thuộc tính 5243 bao gồm bộ khởi tạo CABAC 5261 và bộ giải mã entrôpi 5262. Bộ khởi tạo CABAC 5261 khởi tạo hoặc khởi tạo lại CABAC đáp lại cờ khởi tạo CABAC. Bộ giải mã entrôpi 5262 giải mã thông tin thuộc tính theo CABAC.

Các bộ giải mã thông tin bổ sung 5244 tạo ra thông tin bổ sung bằng cách giải mã thông tin bổ sung được mã hóa.

Bộ kết hợp 5245 tạo ra thông tin hình học bằng cách kết hợp các đoạn của thông tin hình học phân chia nhờ sử dụng thông tin bổ sung. Bộ kết hợp 5245 tạo ra thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin thuộc tính phân chia nhờ sử dụng

thông tin bổ sung. Ví dụ, bộ kết hợp 5245 đầu tiên tạo ra dữ liệu đám mây điểm được kết hợp với mảnh bằng cách kết hợp dữ liệu đám mây điểm được giải mã được kết hợp với các lát nhờ sử dụng thông tin bổ sung lát. Bộ kết hợp 5245 sau đó tái tạo dữ liệu đám mây điểm ban đầu bằng cách kết hợp dữ liệu đám mây điểm được kết hợp với các mảnh nhờ sử dụng thông tin bổ sung mảnh.

Lưu ý rằng, mặc dù Fig.80 thể hiện ví dụ trong đó có hai bộ giải mã thông tin hình học 5242 và hai bộ giải mã thông tin các thuộc tính 5243, số của các bộ giải mã thông tin hình học 5242 và số của bộ giải mã thông tin các thuộc tính 5243 có thể là một, hoặc ba hoặc nhiều hơn. Các đoạn dữ liệu phân chia có thể được xử lý song song trong cùng chip, chẳng hạn như bởi các lõi của CPU, được xử lý song song bởi các lõi của các chip, hoặc được xử lý song song bởi các lõi của các chip.

Fig.82 là lưu đồ minh họa ví dụ về quy trình xử lý được kết hợp với việc khởi tạo của CABAC trong việc mã hóa của thông tin hình học hoặc mã hóa của thông tin thuộc tính.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định, đối với mỗi lát, xem có khởi tạo CABAC trong việc mã hóa của thông tin hình học đối với lát dựa vào điều kiện được xác định trước (S5201) hay không.

Khi được xác định để khởi tạo CABAC (nếu Có ở S5202), thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh được sử dụng cho việc mã hóa của thông tin hình học (S5203). Trị số khởi tạo ngữ cảnh được thiết đặt nhờ xét đến các đặc tính mã hóa. Trị số ban đầu có thể là trị số được xác định trước hoặc có thể được xác định một cách thích ứng tùy thuộc vào các đặc tính của dữ liệu trong lát.

Thiết bị mã hóa dữ liệu ba chiều sau đó thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1, và thiết đặt trị số khởi tạo ngữ cảnh (S5204). Khi khởi tạo CABAC, quy trình khởi tạo được thực hiện nhờ sử dụng trị số khởi tạo ngữ cảnh trong việc mã hóa của thông tin hình học.

Mặt khác, khi được xác định không khởi tạo CABAC (nếu Không ở S5202), thiết bị mã hóa dữ liệu ba chiều thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0 (S5205).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định, đối với mỗi lát, xem có khởi tạo CABAC trong việc mã hóa của thông tin thuộc tính đối với lát dựa vào điều kiện được xác

định trước (S5206) hay không.

Khi được xác định để khởi tạo CABAC (nếu Có ở S5207), thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh được sử dụng cho việc mã hóa của thông tin thuộc tính (S5208). Trị số khởi tạo ngữ cảnh được thiết đặt nhờ xét đến các đặc tính mã hóa. Trị số ban đầu có thể là trị số được xác định trước hoặc có thể được xác định một cách thích ứng tùy thuộc vào các đặc tính của dữ liệu trong lát.

Thiết bị mã hóa dữ liệu ba chiều sau đó thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1, và thiết đặt trị số khởi tạo ngữ cảnh (S5209). Khi khởi tạo CABAC, quy trình khởi tạo được thực hiện nhờ sử dụng trị số khởi tạo ngữ cảnh trong việc mã hóa của thông tin thuộc tính.

Mặt khác, khi được xác định không khởi tạo CABAC (nếu Không ở S5207), thiết bị mã hóa dữ liệu ba chiều thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0 (S5210).

Lưu ý rằng, trong lưu đồ trên Fig.82, việc xử lý liên quan đến thông tin hình học và việc xử lý liên quan đến thông tin thuộc tính có thể được thực hiện theo thứ tự ngược hoặc song song.

Lưu ý rằng, mặc dù lưu đồ trên Fig.82 thể hiện việc xử lý trên cơ sở lát như ví dụ, việc xử lý trên cơ sở mảnh hoặc quy trình dựa trên các bộ phận dữ liệu khác có thể được thực hiện theo cách thức như nhau như việc xử lý trên cơ sở lát. Nghĩa là, lát trong lưu đồ trên Fig.82 có thể được thay thế với mảnh hoặc các bộ phận dữ liệu khác.

Điều kiện được xác định trước đối với thông tin hình học và điều kiện được xác định trước đối với thông tin thuộc tính có thể là điều kiện như nhau hoặc các điều kiện khác nhau.

Fig.83 là sơ đồ minh họa ví dụ về các thời điểm của việc khởi tạo CABAC đối với dữ liệu đám mây điểm ở dạng dòng bit.

Dữ liệu đám mây điểm bao gồm thông tin hình học và không hoặc nhiều hơn các đoạn thông tin thuộc tính. Nghĩa là, dữ liệu đám mây điểm có thể không bao gồm thông tin thuộc tính hoặc bao gồm các đoạn thông tin thuộc tính.

Ví dụ, như thông tin thuộc tính trên điểm ba chiều, dữ liệu đám mây điểm có thể bao gồm thông tin màu sắc, có thể bao gồm thông tin màu sắc và thông tin phản xạ, hoặc có thể bao gồm một hoặc nhiều đoạn của thông tin màu sắc mỗi được liên kết tới một hoặc

nhiều đoạn của thông tin điểm ngắm.

Theo cấu hình bất kỳ, phương pháp được mô tả theo phương án này có thể được áp dụng.

Tiếp theo, điều kiện xác định của việc xem có khởi tạo CABAC hay không sẽ được mô tả.

Có thể được xác định để khởi tạo CABAC trong việc mã hóa của thông tin hình học hoặc thông tin thuộc tính khi bất kỳ trong số các điều kiện được mô tả dưới đây được thỏa mãn.

Ví dụ, CABAC có thể được khởi tạo ở dữ liệu dẫn đầu của thông tin hình học hoặc thông tin thuộc tính (mỗi đoạn của thông tin thuộc tính nếu có các đoạn thông tin thuộc tính). Ví dụ, CABAC có thể được khởi tạo ở đỉnh của dữ liệu tạo nên khung PCC mà có thể được giải mã đơn lẻ. Nghĩa là, như được minh họa ở phần (a) trên Fig.83, nếu các khung PCC có thể được giải mã trên cơ sở khung, CABAC có thể được khởi tạo ở dữ liệu dẫn đầu của khung PCC.

Ví dụ, như được minh họa ở phần (b) trên Fig.83, nếu các khung không thể được giải mã đơn lẻ, chẳng hạn như khi dự đoán liên kết được sử dụng giữa các khung PCC, CABAC có thể được khởi tạo ở dữ liệu dẫn đầu của bộ phận truy cập ngẫu nhiên (GOF chẳng hạn).

Ví dụ, như được minh họa ở phần (c) trên Fig.83, CABAC có thể được khởi tạo ở đỉnh của một hoặc nhiều đoạn của dữ liệu lát phân chia, ở đỉnh của một hoặc nhiều đoạn dữ liệu mảnh phân chia, hoặc ở đỉnh của dữ liệu phân chia khác.

Mặc dù phần (c) trên Fig.83 thể hiện các mảnh như ví dụ, phần mô tả này vẫn đúng đối với các lát. CABAC có thể luôn được khởi tạo ở đỉnh của mảnh hoặc lát hoặc có thể luôn không được khởi tạo ở đỉnh của mảnh hoặc lát.

Fig.84 là sơ đồ minh họa cấu hình của dữ liệu được mã hóa và phương pháp lưu trữ dữ liệu được mã hóa trong bộ phận NAL.

Thông tin khởi tạo có thể được lưu trữ trong đoạn đầu của dữ liệu được mã hóa hoặc trong siêu dữ liệu. Thông tin khởi tạo có thể cũng được lưu trữ trong cả đoạn đầu và siêu dữ liệu. Thông tin khởi tạo là cabac_init_flag, CABAC trị số ban đầu, hoặc chỉ số của bảng có khả năng nhận dạng trị số ban đầu.

Theo phương án này, "siêu dữ liệu" trong phần mô tả mà các thứ được lưu trữ trong

siêu dữ liệu có thể được thay thế bằng "đoạn đầu của dữ liệu được mã hóa" hoặc ngược lại.

Khi thông tin khởi tạo được lưu trữ trong đoạn đầu của dữ liệu được mã hóa, thông tin khởi tạo có thể được lưu trữ trong bộ phận NAL thứ nhất trong dữ liệu được mã hóa chẳng hạn. Thông tin khởi tạo về việc mã hóa của thông tin hình học được lưu trữ trong thông tin hình học, và thông tin khởi tạo về việc mã hóa của thông tin thuộc tính được lưu trữ trong thông tin thuộc tính.

Cabac_init_flag cho việc mã hóa của thông tin thuộc tính và cabac_init_flag cho việc mã hóa của thông tin hình học có thể được thiết đặt là trị số như nhau hoặc các trị số khác nhau. Khi các cờ được thiết đặt là trị số như nhau, cabac_init_flag có thể được chia sẻ đối với thông tin hình học và thông tin thuộc tính. Khi các cờ được thiết đặt là các trị số khác nhau, cabac_init_flag đối với thông tin hình học và cabac_init_flag đối với thông tin thuộc tính chỉ báo các trị số khác nhau.

Thông tin khởi tạo đối với thông tin hình học và thông tin khởi tạo đối với thông tin thuộc tính có thể được lưu trữ trong siêu dữ liệu chung, ít nhất một trong số siêu dữ liệu riêng của thông tin hình học và siêu dữ liệu riêng của thông tin thuộc tính, hoặc cả siêu dữ liệu chung và siêu dữ liệu riêng. Cờ có thể được sử dụng để chỉ báo siêu dữ liệu nào trong số siêu dữ liệu riêng đối với thông tin hình học, siêu dữ liệu riêng đối với thông tin thuộc tính, và siêu dữ liệu chung thông tin khởi tạo được lưu trữ.

Fig.85 là lưu đồ minh họa ví dụ về quy trình xử lý được kết hợp với việc khởi tạo của CABAC trong việc giải mã của thông tin hình học hoặc giải mã của thông tin thuộc tính.

Thiết bị giải mã dữ liệu ba chiều phân tích dữ liệu được mã hóa để thu nhận cờ khởi tạo CABAC đối với thông tin hình học, cờ khởi tạo CABAC đối với thông tin thuộc tính, và trị số khởi tạo ngữ cảnh (S5211).

Thiết bị giải mã dữ liệu ba chiều sau đó xác định xem cờ khởi tạo CABAC đối với thông tin hình học là 1 hay không (S5512).

Khi cờ khởi tạo CABAC đối với thông tin hình học là 1 (nếu Có ở S5212), thiết bị giải mã dữ liệu ba chiều khởi tạo CABAC giải mã đối với thông tin hình học được mã hóa nhờ sử dụng trị số khởi tạo ngữ cảnh trong việc mã hóa của thông tin hình học (S5213).

Mặt khác, khi cờ khởi tạo CABAC đối với thông tin hình học là 0 (nếu Không ở S5212), thiết bị giải mã dữ liệu ba chiều không khởi tạo CABAC giải mã đối với thông tin

hình học được mã hóa (S5214).

Thiết bị giải mã dữ liệu ba chiều sau đó xác định xem cờ khởi tạo CABAC đối với thông tin thuộc tính là 1 hay không (S5215).

Khi cờ khởi tạo CABAC đối với thông tin thuộc tính là 1 (nếu Có ở S5215), thiết bị giải mã dữ liệu ba chiều khởi tạo CABAC giải mã đối với thông tin thuộc tính được mã hóa nhờ sử dụng trị số khởi tạo ngữ cảnh trong việc mã hóa của thông tin thuộc tính (S5216).

Mặt khác, khi cờ khởi tạo CABAC đối với thông tin thuộc tính là 0 (nếu Không ở S5215), thiết bị giải mã dữ liệu ba chiều không khởi tạo CABAC giải mã đối với thông tin thuộc tính được mã hóa (S5217).

Lưu ý rằng, trong lưu đồ trên Fig.85, việc xử lý liên quan đến thông tin hình học và việc xử lý liên quan đến thông tin thuộc tính có thể được thực hiện theo thứ tự ngược hoặc song song.

Lưu ý rằng lưu đồ trên Fig.85 có thể được áp dụng tới bất kỳ trong số trường hợp chia lát và trường hợp chia mảnh.

Tiếp theo, tiến trình của quy trình mã hóa dữ liệu đám mây điểm và tiến trình của quy trình giải mã dữ liệu đám mây điểm theo phương án này sẽ được mô tả. Fig.86 là lưu đồ của quy trình mã hóa dữ liệu đám mây điểm theo phương án này.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định phương pháp phân chia được sử dụng (S5221). Phương pháp phân chia bao gồm việc xác định xem có thực hiện việc chia mảnh hay không và việc xác định xem có thực hiện việc chia lát hay không. Phương pháp phân chia có thể bao gồm số của các mảnh hoặc các lát trong trường hợp trong đó việc chia mảnh hoặc việc chia lát được thực hiện, và loại của việc phân chia chẳng hạn. Loại của việc phân chia là sơ đồ dựa vào hình dạng đối tượng, sơ đồ dựa vào thông tin bản đồ hoặc thông tin hình học, hoặc sơ đồ dựa vào lượng dữ liệu hoặc lượng xử lý chẳng hạn. Phương pháp phân chia có thể được xác định trước.

Khi việc chia mảnh được thực hiện (nếu Có ở S5222), thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách chia thông tin hình học và thông tin thuộc tính về cơ sở mảnh (S5223). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin bổ sung mảnh liên quan đến việc chia mảnh.

Khi việc chia lát được thực hiện (nếu Có ở S5224), thiết bị mã hóa dữ liệu ba chiều

tạo ra các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia bằng cách chia các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh (hoặc thông tin hình học và thông tin thuộc tính) (S5225). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính liên quan đến việc chia lát.

Thiết bị mã hóa dữ liệu ba chiều sau đó tạo ra các đoạn của thông tin hình học được mã hóa và các đoạn của thông tin thuộc tính được mã hóa bằng cách mã hóa mỗi trong số các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia (S5226). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin phụ thuộc.

Thiết bị mã hóa dữ liệu ba chiều sau đó tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách tích hợp (đa hợp) các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa và thông tin bổ sung trong bộ phận NAL (S5227). Thiết bị mã hóa dữ liệu ba chiều cũng truyền dữ liệu được mã hóa được tạo ra.

Fig.87 là lưu đồ minh họa ví dụ về quy trình xác định trị số của cờ khởi tạo CABAC và cập nhật thông tin bổ sung trong việc chia mảnh (S5223) và việc chia lát (S5525).

Ở các bước S5223 và S5225, thông tin hình học mảnh và thông tin thuộc tính mảnh và/hoặc lát thông tin hình học và thông tin thuộc tính lát có thể được chia một cách độc lập theo các cách thức tương ứng, hoặc có thể là một cách tập hợp được chia theo cách thức chung. Theo cách này, thông tin bổ sung được chia trên cơ sở mảnh và/hoặc trên cơ sở lát được tạo ra.

Ở các bước này, thiết bị mã hóa dữ liệu ba chiều xác định xem có thiết đặt cờ khởi tạo CABAC tới 1 hoặc 0 (S5231) hay không.

Thiết bị mã hóa dữ liệu ba chiều sau đó cập nhật thông tin bổ sung để bao gồm cờ khởi tạo CABAC được xác định (S5232).

Fig.88 là lưu đồ minh họa ví dụ về quy trình khởi tạo CABAC trong việc xử lý mã hóa (S5226).

Thiết bị mã hóa dữ liệu ba chiều xác định xem cờ khởi tạo CABAC là 1 hay không (S5241).

Khi cờ khởi tạo CABAC là 1 (nếu Có ở S5241), thiết bị mã hóa dữ liệu ba chiều khởi tạo lại CABAC bộ mã hóa tới trạng thái mặc định (S5242).

Thiết bị mã hóa dữ liệu ba chiều sau đó tiếp tục quy trình mã hóa đến khi điều kiện

để dừng quy trình mã hóa được thỏa mãn, chẳng hạn như đến khi không có dữ liệu được mã hóa (S5243).

Fig.89 là lưu đồ minh họa quy trình giải mã dữ liệu đám mây điểm theo phương án này. Thứ nhất, thiết bị giải mã dữ liệu ba chiều xác định phương pháp phân chia nhờ phân tích thông tin bổ sung (thông tin bổ sung mảnh, thông tin bổ sung lát hình học, và thông tin bổ sung lát thuộc tính) liên quan đến phương pháp phân chia được bao gồm trong dữ liệu được mã hóa (dòng được mã hóa) (S5251). Phương pháp phân chia bao gồm việc xác định xem có thực hiện việc chia mảnh hay không và việc xác định xem có thực hiện việc chia lát hay không. Phương pháp phân chia có thể bao gồm, ví dụ, số của các mảnh hoặc các lát và loại của việc phân chia trong trường hợp trong đó việc chia mảnh hoặc việc chia lát được thực hiện.

Thiết bị giải mã dữ liệu ba chiều sau đó tạo ra thông tin hình học phân chia và thông tin thuộc tính phân chia bằng cách giải mã các đoạn của thông tin hình học được mã hóa và các đoạn của thông tin thuộc tính được mã hóa được bao gồm trong dữ liệu được mã hóa nhờ sử dụng thông tin phụ thuộc được bao gồm trong dữ liệu được mã hóa (S5252).

Nếu thông tin bổ sung chỉ báo rằng việc chia lát đã được thực hiện (nếu Có ở S5253), thiết bị giải mã dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách kết hợp các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia dựa vào thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính (S5254).

Nếu thông tin bổ sung chỉ báo rằng việc chia mảnh đã được thực hiện (nếu Có ở S5255), thiết bị giải mã dữ liệu ba chiều tạo ra thông tin hình học và thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh (các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia) dựa vào thông tin bổ sung mảnh (S5256).

Fig.90 là lưu đồ minh họa ví dụ về quy trình khởi tạo bộ giải mã CABAC trong việc kết hợp (S5254) của thông tin được chia thành các lát và kết hợp (S5256) của thông tin được chia thành các mảnh.

Các đoạn của lát thông tin hình học và các đoạn của thông tin thuộc tính lát hoặc các đoạn của thông tin hình học mảnh hoặc các đoạn của thông tin thuộc tính mảnh có thể được kết hợp theo các cách thức tương ứng hoặc theo cách thức như nhau.

Thiết bị giải mã dữ liệu ba chiều thu nhận cờ khởi tạo CABAC bằng cách giải mã thông tin bổ sung trong dòng được mã hóa.

Thiết bị giải mã dữ liệu ba chiều sau đó xác định xem cờ khởi tạo CABAC là 1 hay không (S5262).

Khi cờ khởi tạo CABAC là 1 (nếu Có ở S5262), thiết bị giải mã dữ liệu ba chiều khởi tạo lại bộ giải mã CABAC tới trạng thái mặc định (S5263).

Mặt khác, khi cờ khởi tạo CABAC không phải 1 (nếu Không ở S5262), thiết bị giải mã dữ liệu ba chiều không khởi tạo lại bộ giải mã CABAC và chuyển tới bước S5264.

Thiết bị giải mã dữ liệu ba chiều sau đó tiếp tục quy trình giải mã đến khi điều kiện để dừng quy trình giải mã được thỏa mãn, chẳng hạn như đến khi không có dữ liệu được giải mã (S5264).

Tiếp theo, các điều kiện khác liên quan đến việc xác định xem có khởi tạo CABAC hay không sẽ được mô tả.

Việc xem có khởi tạo việc mã hóa của thông tin hình học hoặc mã hóa của thông tin thuộc tính hay không có thể được xác định nhờ xét đến hiệu quả mã hóa trên cơ sở của các bộ phận dữ liệu, chẳng hạn như các mảnh hoặc các lát. Trong trường hợp đó, CABAC có thể được khởi tạo ở dữ liệu dẫn đầu của mảnh hoặc lát mà thỏa mãn điều kiện được xác định trước.

Tiếp theo, các điều kiện liên quan đến việc xác định xem có khởi tạo CABAC hay không trong việc mã hóa của thông tin hình học sẽ được mô tả.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể xác định mật độ của dữ liệu đám mây điểm đối với mỗi lát, nghĩa là, số của các điểm trên mỗi vùng bộ phận thuộc về mỗi lát, so sánh mật độ dữ liệu của lát với mật độ dữ liệu của lát khác, và xác định rằng hiệu quả mã hóa là tốt hơn khi CABAC không được khởi tạo và xác định không khởi tạo CABAC nếu sự thay đổi về mật độ dữ liệu thỏa mãn điều kiện được xác định trước. Mặt khác, nếu sự thay đổi về mật độ dữ liệu không thỏa mãn điều kiện được xác định trước, thiết bị mã hóa dữ liệu ba chiều có thể xác định rằng hiệu quả mã hóa là tốt hơn khi CABAC được khởi tạo, và xác định để khởi tạo CABAC.

Ở đây, "lát khác" có thể là lát đứng trước theo thứ tự giải mã hoặc lát lân cận theo không gian chẳng hạn. Thiết bị mã hóa dữ liệu ba chiều có thể không thực hiện việc so sánh về mật độ dữ liệu với mật độ dữ liệu của lát khác và có thể xác định xem có khởi tạo

CABAC hay không dựa vào việc xem mật độ dữ liệu của lát là mật độ dữ liệu được xác định trước hay không.

Khi được xác định để khởi tạo CABAC, thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh được sử dụng cho việc mã hóa của thông tin hình học. Trị số khởi tạo ngữ cảnh được thiết đặt ở trị số mà đưa ra các đặc tính mã hóa tốt đáp lại mật độ dữ liệu. Thiết bị mã hóa dữ liệu ba chiều có thể giữ lại trị số ban đầu bảng đối với mật độ dữ liệu trước và lựa chọn trị số ban đầu tối ưu từ bảng.

Lưu ý rằng thiết bị mã hóa dữ liệu ba chiều có thể xác định xem có khởi tạo CABAC hay không dựa vào số của các điểm, việc phân bố của các điểm, hoặc sự không cân bằng của các điểm, ví dụ, thay vì dựa vào mật độ của lát nêu trên như ví dụ. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể xác định xem có khởi tạo CABAC hay không dựa vào số lượng đặc điểm hoặc số của các điểm đặc điểm được thu nhận từ thông tin về các điểm hoặc dựa vào đối tượng được nhận diện. Trong trường hợp đó, tiêu chuẩn xác định có thể được giữ lại trong bộ nhớ ở dạng bảng mà kết hợp tiêu chuẩn xác định với số lượng đặc điểm hoặc số của các điểm đặc điểm được thu nhận từ thông tin về các điểm hoặc đối tượng được nhận diện dựa vào thông tin về các điểm.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể xác định đối tượng được kết hợp với thông tin hình học của thông tin bản đồ và xác định xem có khởi tạo CABAC hay không dựa vào đối tượng dựa vào thông tin hình học. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể xác định xem có khởi tạo CABAC hay không dựa vào thông tin hoặc số lượng đặc điểm được thu nhận bằng cách chiếu dữ liệu ba chiều trên mặt phẳng hai chiều.

Tiếp theo, các điều kiện liên quan đến việc xác định xem có khởi tạo CABAC hay không trong việc mã hóa của thông tin thuộc tính sẽ được mô tả.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể so sánh đặc tính màu sắc của lát liên quan với đặc tính màu sắc của lát đứng trước, và xác định rằng hiệu quả mã hóa là tốt hơn khi CABAC không được khởi tạo và xác định không khởi tạo CABAC nếu sự thay đổi về đặc tính màu sắc thỏa mãn điều kiện được xác định trước. Mặt khác, nếu sự thay đổi về đặc tính màu sắc không thỏa mãn điều kiện được xác định trước, thiết bị mã hóa dữ liệu ba chiều có thể xác định rằng hiệu quả mã hóa là tốt hơn khi CABAC được khởi tạo, và xác định để khởi tạo CABAC. Đặc tính màu sắc là độ chói (luminance), độ màu (chromaticity), hoặc sắc độ, biểu đồ của nó, hoặc tính liên tục màu sắc chẳng hạn.

Ở đây, "lát khác" có thể là lát đứng trước theo thứ tự giải mã hoặc lát lân cận theo

không gian chẳng hạn. Thiết bị mã hóa dữ liệu ba chiều có thể không thực hiện việc so sánh về mật độ dữ liệu với mật độ dữ liệu của lát khác và có thể xác định xem có khởi tạo CABAC hay không dựa vào việc xem mật độ dữ liệu của lát là mật độ dữ liệu được xác định trước hay không.

Khi được xác định để khởi tạo CABAC, thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh được sử dụng cho việc mã hóa của thông tin thuộc tính. Trị số khởi tạo ngữ cảnh được thiết đặt ở trị số mà đưa ra các đặc tính mã hóa tốt đáp lại mật độ dữ liệu. Thiết bị mã hóa dữ liệu ba chiều có thể giữ lại trị số ban đầu bảng đối với mật độ dữ liệu trước và lựa chọn trị số ban đầu tối ưu từ bảng.

Khi thông tin thuộc tính là năng suất phản xạ, thiết bị mã hóa dữ liệu ba chiều có thể xác định xem có khởi tạo CABAC hay không dựa vào thông tin dựa trên năng suất phản xạ.

Khi điểm ba chiều có các đoạn thông tin thuộc tính, thiết bị mã hóa dữ liệu ba chiều có thể xác định một cách độc lập thông tin khởi tạo đối với mỗi đoạn của thông tin thuộc tính dựa vào đoạn của thông tin thuộc tính, có thể xác định thông tin khởi tạo đối với các đoạn thông tin thuộc tính dựa vào một trong số các đoạn thông tin thuộc tính, hoặc có thể xác định thông tin khởi tạo đối với các đoạn thông tin thuộc tính nhờ sử dụng các đoạn thông tin thuộc tính.

Mặc dù ví dụ đã được mô tả trong đó thông tin khởi tạo đối với thông tin hình học được xác định dựa vào thông tin hình học, và thông tin khởi tạo đối với thông tin thuộc tính được xác định dựa vào thông tin thuộc tính, thông tin khởi tạo đối với thông tin hình học và thông tin thuộc tính có thể được xác định dựa vào thông tin hình học, dựa vào thông tin thuộc tính, hoặc dựa vào cả thông tin hình học và thông tin thuộc tính.

Thiết bị mã hóa dữ liệu ba chiều có thể xác định thông tin khởi tạo dựa vào kết quả của việc mô phỏng về hiệu quả mã hóa được thực hiện bởi bật và tắt cabac_init_flag hoặc lựa chọn một hoặc nhiều trị số ban đầu từ trị số ban đầu bảng chẳng hạn.

Khi phương pháp phân chia dữ liệu thành các mảnh, các lát hoặc tương tự được xác định dựa vào thông tin hình học hoặc thông tin thuộc tính, thiết bị mã hóa dữ liệu ba chiều có thể xác định thông tin khởi tạo dựa vào thông tin giống như thông tin dựa vào việc xác định phương pháp phân chia.

Fig.91 là sơ đồ minh họa ví dụ về các mảnh và các lát.

Ví dụ, các lát trong một mảnh có phần của dữ liệu PCC được nhận diện như được chỉ báo bởi các chú thích. Cờ khởi tạo CABAC có thể được sử dụng để xác định xem việc khởi tạo lại của ngữ cảnh được cần đến hay không trong các lát liên tiếp. Ví dụ, trên Fig.91, khi một mảnh bao gồm dữ liệu lát được chia dựa trên các đối tượng (chẳng hạn như vật thể di chuyển, vỉa hè, công trình xây dựng, cây hoặc các đối tượng khác), các cờ khởi tạo CABAC cho các lát của vật thể di chuyển, vỉa hè, và cây được thiết đặt là 1, và các cờ khởi tạo CABAC cho các lát của công trình xây dựng và các đối tượng khác được thiết đặt là 0. Điều này có nghĩa là, nếu vỉa hè và công trình xây dựng có thể là cả các cấu trúc thường xuyên dày đặc và có các hiệu quả tạo mã tương tự, hiệu quả mã hóa có thể có khả năng được nâng cao nhờ tránh khỏi việc khởi tạo lại của CABAC giữa các lát đối với vỉa hè và công trình xây dựng. Mặt khác, nếu công trình xây dựng và cây có thể là khác nhau đáng kể về mật độ và hiệu quả mã hóa, hiệu quả mã hóa có thể có khả năng được nâng cao bằng cách khởi tạo CABAC giữa các lát đối với công trình xây dựng và cây.

Fig.92 là lưu đồ minh họa ví dụ về phương pháp xác định xem có khởi tạo CABAC hay không và xác định trị số khởi tạo ngữ cảnh.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều chia dữ liệu đám mây điểm thành các lát dựa vào đối tượng được xác định từ thông tin hình học (S5271).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định, đối với mỗi lát, xem có khởi tạo CABAC hay không cho việc mã hóa của thông tin hình học và mã hóa của thông tin thuộc tính dựa vào mật độ dữ liệu của đối tượng của lát (S5272). Nói cách khác, thiết bị mã hóa dữ liệu ba chiều xác định thông tin khởi tạo CABAC (cờ khởi tạo CABAC) cho việc mã hóa của thông tin hình học và mã hóa của thông tin thuộc tính dựa vào thông tin hình học. Thiết bị mã hóa dữ liệu ba chiều xác định việc khởi tạo với hiệu quả mã hóa cao dựa vào dữ liệu đám mây điểm mật độ chẳng hạn. Thông tin khởi tạo CABAC có thể được chỉ báo bởi cabac_init_flag mà là chung cho thông tin hình học và thông tin thuộc tính.

Khi được xác định để khởi tạo CABAC (nếu Có ở S5273), thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh cho việc mã hóa của thông tin hình học (S5274).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số khởi tạo ngữ cảnh cho việc mã hóa của thông tin thuộc tính (S5275).

Thiết bị mã hóa dữ liệu ba chiều sau đó thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1, thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin hình học, thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1, và thiết đặt trị số khởi tạo ngữ cảnh đối

với thông tin thuộc tính (S5276). Lưu ý rằng khi khởi tạo CABAC, thiết bị mã hóa dữ liệu ba chiều thực hiện quy trình khởi tạo nhờ sử dụng trị số khởi tạo ngữ cảnh trong mỗi trong số mã hóa của thông tin hình học và mã hóa của thông tin thuộc tính.

Mặt khác, khi được xác định không khởi tạo CABAC (nếu Không ở S5273), thiết bị mã hóa dữ liệu ba chiều thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0, và thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0 (S5277).

Fig.93 là sơ đồ minh họa ví dụ về trường hợp trong đó bản đồ, mà là hình chiếu từ trên xuống của dữ liệu đám mây điểm được thu nhận bởi LiDAR, được chia thành các mảnh. Fig.94 là lưu đồ minh họa ví dụ khác về phương pháp xác định xem có khởi tạo CABAC hay không và xác định trị số khởi tạo ngữ cảnh.

Trong dữ liệu bản đồ cỡ lớn, thiết bị mã hóa dữ liệu ba chiều chia dữ liệu đám mây điểm thành một hoặc nhiều mảnh dựa vào thông tin hình học theo cách thức phân chia hình chiếu từ trên xuống hai chiều (S5281). Thiết bị mã hóa dữ liệu ba chiều có thể chia dữ liệu đám mây điểm thành các vùng hình vuông như được minh họa trên Fig.93 chẳng hạn. Thiết bị mã hóa dữ liệu ba chiều có thể cũng chia dữ liệu đám mây điểm thành các mảnh có các hình dạng hoặc các kích thước khác nhau. Việc phân chia thành các mảnh có thể được thực hiện trong một hoặc nhiều phương pháp được xác định trước hoặc có thể được thực hiện một cách thích ứng.

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định đối tượng trong mỗi mảnh, và xác định xem có khởi tạo CABAC hay không trong việc mã hóa của thông tin hình học đối với mảnh hoặc mã hóa của thông tin thuộc tính đối với mảnh (S5282). Lưu ý rằng, trong việc phân chia thành các lát, thiết bị mã hóa dữ liệu ba chiều nhận diện đối tượng (cây, con người là, vật thể di chuyển, hoặc công trình xây dựng), và xác định xem có thực hiện việc chia lát và xác định trị số ban đầu dựa vào đối tượng hay không.

Khi được xác định để khởi tạo CABAC (nếu Có ở S5283), thiết bị mã hóa dữ liệu ba chiều xác định trị số khởi tạo ngữ cảnh cho việc mã hóa của thông tin hình học (S5284).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số khởi tạo ngữ cảnh cho việc mã hóa của thông tin thuộc tính (S5285).

Ở các bước S5284 và S5285, trị số ban đầu đối với mảnh có các đặc tính mã hóa cụ thể có thể được lưu trữ như trị số ban đầu và được sử dụng như trị số ban đầu đối với mảnh có các đặc tính mã hóa như nhau.

Thiết bị mã hóa dữ liệu ba chiều sau đó thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1, thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin hình học, thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1, và thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin thuộc tính (S5286). Lưu ý rằng khi khởi tạo CABAC, thiết bị mã hóa dữ liệu ba chiều thực hiện quy trình khởi tạo nhờ sử dụng trị số khởi tạo ngữ cảnh trong mỗi trong số mã hóa của thông tin hình học và mã hóa của thông tin thuộc tính.

Mặt khác, khi được xác định không khởi tạo CABAC (nếu Không ở S5283), thiết bị mã hóa dữ liệu ba chiều thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0, và thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0 (S5287).

Phương án 8

Tiếp theo, thông số lượng tử hóa sẽ được mô tả.

Để chia dữ liệu đám mây điểm dựa vào các đặc tính và các vị trí liên quan đến dữ liệu đám mây điểm, lát và mảnh được sử dụng. Ở đây, chất lượng khác nhau có thể được yêu cầu đối với mỗi trong số các đoạn dữ liệu đám mây điểm phân chia, bởi vì của các sự hạn chế phần cứng hoặc các yêu cầu cho việc xử lý theo thời gian thực chẳng hạn. Ví dụ, khi mã hóa dữ liệu đám mây điểm bằng cách chia dữ liệu đám mây điểm thành các lát trên cơ sở đối tượng, dữ liệu lát bao gồm máy móc là ít quan trọng, sao cho giải pháp (chất lượng) của dữ liệu lát có thể được giảm bởi việc lượng tử hóa. Mặt khác, giải pháp (chất lượng) của dữ liệu lát quan trọng có thể được tăng lên bằng cách thiết đặt trị số lượng tử hóa ở trị số thấp hơn. Thông số lượng tử hóa được sử dụng để cho phép điều khiển của trị số lượng tử hóa như vậy.

Ở đây, dữ liệu được lượng tử hóa, mức độ được sử dụng cho việc lượng tử hóa, và dữ liệu được lượng tử hóa, mà là kết quả tính bởi việc lượng tử hóa, được biểu diễn bởi các phương trình G1 và G2 dưới đây.

Dữ liệu được lượng tử hóa = dữ liệu/mức độ (phương trình G1)

Dữ liệu = dữ liệu được lượng tử hóa*mức độ (phương trình G2)

Fig.95 là sơ đồ mô tả quy trình được thực hiện bởi bộ lượng tử hóa 5323 mà lượng tử hóa dữ liệu và bộ lượng tử hóa ngược 5333 mà lượng tử hóa ngược dữ liệu được lượng tử hóa.

Bộ lượng tử hóa 5323 lượng tử hóa dữ liệu bằng cách sử dụng mức độ. Nghĩa là, bộ lượng tử hóa 5323 tính dữ liệu được lượng tử hóa, mà là dữ liệu được lượng tử hóa,

bằng cách thực hiện quy trình theo phương trình G1.

Bộ lượng tử hóa ngược 5333 lượng tử hóa ngược được lượng tử hóa dữ liệu bằng cách sử dụng mức độ. Nghĩa là, bộ lượng tử hóa ngược tính dữ liệu được lượng tử hóa được lượng tử hóa ngược bằng cách thực hiện quy trình theo phương trình G2.

Mức độ và trị số lượng tử hóa (trị số thông số lượng tử hóa (quantization parameter, viết tắt là QP)) được biểu diễn bởi phương trình G3 dưới đây.

Trị số lượng tử hóa (trị số QP) = log(mức độ) (phương trình G3)

Trị số lượng tử hóa (trị số QP) = trị số mặc định (trị số tham chiếu) + đenta lượng tử hóa (thông tin sai lệch) (phương trình G4)

Các thông số này thường được đề cập đến là thông số lượng tử hóa.

Ví dụ, như được minh họa trên Fig.96, trị số lượng tử hóa là trị số đối với trị số mặc định, và được tính bằng cách bổ sung đenta lượng tử hóa tới trị số mặc định. Nếu trị số lượng tử hóa là nhỏ hơn trị số mặc định, đenta lượng tử hóa là trị số âm. Nếu trị số lượng tử hóa là lớn hơn trị số mặc định, đenta lượng tử hóa là trị số dương. Nếu trị số lượng tử hóa là bằng trị số mặc định, đenta lượng tử hóa là 0. Khi đenta lượng tử hóa là 0, đenta lượng tử hóa có thể được bỏ qua.

Quy trình mã hóa sẽ được mô tả. Fig.97 là sơ đồ khối minh họa cấu hình của bộ mã hóa thứ nhất 5300 được bao gồm trong thiết bị mã hóa dữ liệu ba chiều theo phương án này. Fig.98 là sơ đồ khối minh họa cấu hình của bộ chia 5301 theo phương án này. Fig.99 là sơ đồ khối minh họa cấu hình của bộ mã hóa thông tin hình học 5302 và bộ mã hóa thông tin thuộc tính 5303 theo phương án này.

Bộ mã hóa thứ nhất 5300 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ nhất (PCC trên cơ sở hình học (GPCC)). Bộ mã hóa thứ nhất 5300 bao gồm bộ chia 5301, các bộ mã hóa thông tin hình học 5302, các bộ mã hóa thông tin các thuộc tính 5303, bộ mã hóa thông tin bổ sung 5304, và bộ đa hợp 5305.

Bộ chia 5301 tạo ra các đoạn dữ liệu phân chia bằng cách chia dữ liệu đám mây điểm. Cụ thể là, bộ chia 5301 tạo ra các đoạn dữ liệu phân chia bằng cách chia không gian của dữ liệu đám mây điểm thành các không gian con. Ở đây, không gian con là sự kết hợp của các mảnh hoặc các lát, hoặc sự kết hợp của các mảnh và các lát. Cụ thể hơn là, dữ liệu đám mây điểm bao gồm thông tin hình học, thông tin thuộc tính, và thông tin bổ sung. Bộ

chia 5301 chia thông tin hình học thành các đoạn của thông tin hình học phân chia, và chia thông tin thuộc tính thành các đoạn của thông tin thuộc tính phân chia. Bộ chia 5301 cũng tạo ra thông tin bổ sung liên quan đến việc phân chia.

Như được minh họa trên Fig.98, bộ chia 5301 bao gồm bộ chia mảnh 5311 và bộ chia lát 5312. Ví dụ, bộ chia mảnh 5311 chia đám mây điểm thành các mảnh. Bộ chia mảnh 5311 có thể xác định trị số lượng tử hóa được sử dụng đối với mỗi mảnh phân chia như thông tin bổ sung mảnh.

Bộ chia lát 5312 còn chia mảnh được thu nhận bởi bộ chia mảnh 5311 thành các lát. Bộ chia lát 5312 có thể xác định trị số lượng tử hóa được sử dụng đối với mỗi lát phân chia như thông tin bổ sung lát.

Các bộ mã hóa thông tin hình học 5302 tạo ra các đoạn của thông tin hình học được mã hóa bằng cách mã hóa các đoạn của thông tin hình học phân chia. Ví dụ, các bộ mã hóa thông tin hình học 5302 xử lý các đoạn của thông tin hình học phân chia song song.

Như được minh họa trên Fig.99, bộ mã hóa thông tin hình học 5302 bao gồm bộ tính trị số lượng tử hóa 5321 và bộ mã hóa entrôpi 5322. Bộ tính trị số lượng tử hóa 5321 tạo ra trị số lượng tử hóa (thông số lượng tử hóa) của thông tin hình học phân chia được mã hóa. Bộ mã hóa entrôpi 5322 tính thông tin hình học được lượng tử hóa bằng cách lượng tử hóa thông tin hình học phân chia nhờ sử dụng trị số lượng tử hóa (thông số lượng tử hóa) được tạo ra bởi bộ tính trị số lượng tử hóa 5321.

Các bộ mã hóa thông tin các thuộc tính 5303 tạo ra các đoạn của thông tin thuộc tính được mã hóa bằng cách mã hóa các đoạn của thông tin thuộc tính phân chia. Ví dụ, các bộ mã hóa thông tin các thuộc tính 5303 xử lý các đoạn của thông tin thuộc tính phân chia song song.

Như được minh họa trên Fig.99, bộ mã hóa thông tin thuộc tính 5303 bao gồm bộ tính trị số lượng tử hóa 5331 và bộ mã hóa entrôpi 5332. Bộ tính trị số lượng tử hóa 5321 tạo ra trị số lượng tử hóa (thông số lượng tử hóa) của thông tin thuộc tính phân chia được mã hóa. Bộ mã hóa entrôpi 5332 tính thông tin thuộc tính được lượng tử hóa bằng cách lượng tử hóa thông tin thuộc tính phân chia nhờ sử dụng trị số lượng tử hóa (thông số lượng tử hóa) được tạo ra bởi bộ tính trị số lượng tử hóa 5331.

Bộ mã hóa thông tin bổ sung 5304 tạo ra thông tin bổ sung được mã hóa bằng cách mã hóa thông tin bổ sung được bao gồm trong dữ liệu đám mây điểm và thông tin bổ sung

liên quan đến dữ liệu việc phân chia được tạo ra trong việc phân chia bởi bộ chia 5301.

Bộ đa hợp 5305 tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách đa hợp các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa, và truyền dữ liệu được mã hóa được tạo ra. Thông tin bổ sung được mã hóa được sử dụng để giải mã.

Lưu ý rằng, mặc dù Fig.97 thể hiện ví dụ trong đó có hai bộ mã hóa thông tin hình học 5302 và hai bộ mã hóa thông tin các thuộc tính 5303, số của các bộ mã hóa thông tin hình học 5302 và số của bộ mã hóa thông tin các thuộc tính 5303 có thể là một, hoặc ba hoặc nhiều hơn. Các đoạn dữ liệu phân chia có thể được xử lý song song trong cùng chip, chẳng hạn như bởi các lõi của CPU, được xử lý song song bởi các lõi của các chip, hoặc được xử lý song song bởi các lõi của các chip.

Tiếp theo, quy trình giải mã sẽ được mô tả. Fig.100 là sơ đồ khối minh họa cấu hình của bộ giải mã thứ nhất 5340. Fig.101 là sơ đồ khối minh họa cấu hình của bộ giải mã thông tin hình học 5342 và bộ giải mã thông tin thuộc tính 5343.

Bộ giải mã thứ nhất 5340 tái tạo dữ liệu đám mây điểm bằng cách giải mã dữ liệu được mã hóa (dòng được mã hóa) được tạo ra bằng cách mã hóa dữ liệu đám mây điểm theo phương pháp mã hóa thứ nhất (GPCC). Bộ giải mã thứ nhất 5340 bao gồm bộ giải đa hợp 5341, các bộ giải mã thông tin hình học 5342, các bộ giải mã thông tin các thuộc tính 5343, bộ giải mã thông tin bổ sung 5344, và bộ kết hợp 5345.

Bộ giải đa hợp 5341 tạo ra các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa, và thông tin bổ sung được mã hóa bằng cách giải đa hợp dữ liệu được mã hóa (dòng được mã hóa).

Các bộ giải mã thông tin hình học 5342 tạo ra các đoạn của thông tin hình học được lượng tử hóa bằng cách giải mã các đoạn của thông tin hình học được mã hóa. Ví dụ, các bộ giải mã thông tin hình học 5342 xử lý các đoạn của thông tin hình học được mã hóa song song.

Như được minh họa trên Fig.101, bộ giải mã thông tin hình học 5342 bao gồm bộ tính trị số lượng tử hóa 5351 và bộ giải mã entrôpi 5352. Bộ tính trị số lượng tử hóa 5351 tạo ra trị số lượng tử hóa của thông tin hình học được lượng tử hóa. Bộ giải mã entrôpi 5352 tính thông tin hình học bằng cách lượng tử hóa ngược thông tin hình học được lượng tử hóa nhờ sử dụng trị số lượng tử hóa được tạo ra bởi bộ tính trị số lượng tử hóa 5351.

Các bộ giải mã thông tin các thuộc tính 5343 tạo ra các đoạn của thông tin thuộc tính phân chia bằng cách giải mã các đoạn của thông tin thuộc tính được mã hóa. Ví dụ, các bộ giải mã thông tin các thuộc tính 5343 xử lý các đoạn của thông tin thuộc tính được mã hóa song song.

Như được minh họa trên Fig.101, bộ giải mã thông tin thuộc tính 5343 bao gồm bộ tính trị số lượng tử hóa 5361 và bộ giải mã entrôpi 5362. Bộ tính trị số lượng tử hóa 5361 tạo ra trị số lượng tử hóa của thông tin thuộc tính được lượng tử hóa. Bộ giải mã entrôpi 5362 tính thông tin thuộc tính bằng cách lượng tử hóa ngược thông tin thuộc tính được lượng tử hóa nhờ sử dụng trị số lượng tử hóa được tạo ra bởi bộ tính trị số lượng tử hóa 5361.

Các bộ giải mã thông tin bổ sung 5344 tạo ra thông tin bổ sung bằng cách giải mã thông tin bổ sung được mã hóa.

Bộ kết hợp 5345 tạo ra thông tin hình học bằng cách kết hợp các đoạn của thông tin hình học phân chia nhờ sử dụng thông tin bổ sung. Bộ kết hợp 5345 tạo ra thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin thuộc tính phân chia nhờ sử dụng thông tin bổ sung. Ví dụ, bộ kết hợp 5345 đầu tiên tạo ra dữ liệu đám mây điểm được kết hợp với mảnh bằng cách kết hợp dữ liệu đám mây điểm được giải mã được kết hợp với các lát nhờ sử dụng thông tin bổ sung lát. Bộ kết hợp 5345 sau đó tái tạo dữ liệu đám mây điểm ban đầu bằng cách kết hợp dữ liệu đám mây điểm được kết hợp với các mảnh nhờ sử dụng thông tin bổ sung mảnh.

Lưu ý rằng, mặc dù Fig.100 thể hiện ví dụ trong đó có hai bộ giải mã thông tin hình học 5342 và hai bộ giải mã thông tin các thuộc tính 5343, số của các bộ giải mã thông tin hình học 5342 và số của bộ giải mã thông tin các thuộc tính 5343 có thể là một, hoặc ba hoặc nhiều hơn. Các đoạn dữ liệu phân chia có thể được xử lý song song trong cùng chip, chẳng hạn như bởi các lõi của CPU, được xử lý song song bởi các lõi của các chip, hoặc được xử lý song song bởi các lõi của các chip.

[Phương pháp xác định thông số lượng tử hóa]

Fig.102 là lưu đồ minh họa ví dụ về quy trình liên quan đến việc xác định của trị số lượng tử hóa (trị số thông số lượng tử hóa: trị số QP) trong việc mã hóa của thông tin hình học (hình học) hoặc mã hóa của thông tin thuộc tính (attribute).

Trị số QP được xác định nhờ xét đến hiệu quả mã hóa trên cơ sở của các bộ phận

dữ liệu của thông tin hình học hoặc thông tin thuộc tính tạo nên khung PCC chẳng hạn. Khi bộ phận dữ liệu là mảnh hoặc lát nhận được từ việc phân chia, trị số QP được xác định dựa trên dữ liệu phân chia các bộ phận nhờ xét đến hiệu quả mã hóa của dữ liệu phân chia các bộ phận. Trị số QP có thể được xác định trên cơ sở của các bộ phận dữ liệu trước khi việc phân chia.

Như được minh họa trên Fig.102, thiết bị mã hóa dữ liệu ba chiều xác định trị số QP được sử dụng cho việc mã hóa của thông tin hình học (S5301). Thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP đối với mỗi trong số các các lát phân chia theo cách thức được xác định trước. Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định trị số QP dựa vào các đặc tính hoặc chất lượng của dữ liệu của thông tin hình học. Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể xác định mật độ của dữ liệu đám mây điểm đối với mỗi đơn vị dữ liệu, nghĩa là, số của các điểm trên mỗi vùng bộ phận thuộc về mỗi lát, và xác định trị số tương ứng với mật độ của dữ liệu đám mây điểm như trị số QP. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể xác định, như trị số QP, bất kỳ trong số các trị số sau đây tương ứng với thông tin hình học: số của các điểm của dữ liệu đám mây điểm, việc phân bố của các điểm của dữ liệu đám mây điểm, sự không cân bằng của các điểm của dữ liệu đám mây điểm, số lượng đặc điểm được thu nhận từ thông tin về các điểm, số của các điểm đặc điểm, hoặc đối tượng được nhận diện. Thiết bị mã hóa dữ liệu ba chiều có thể cũng xác định đối tượng được kết hợp với thông tin hình học của bản đồ và xác định trị số QP dựa vào đối tượng dựa vào thông tin hình học, hoặc có thể xác định trị số QP dựa vào thông tin hoặc số lượng đặc điểm được thu nhận bằng cách chiếu đám mây điểm ba chiều trên mặt phẳng hai chiều. Trị số QP tương ứng có thể được lưu trữ trong bộ nhớ trước ở dạng bảng mà kết hợp trị số QP với mật độ, số của các điểm, việc phân bố của các điểm, hoặc sự không cân bằng của các điểm của dữ liệu đám mây điểm. Trị số QP tương ứng có thể cũng được lưu trữ trong bộ nhớ trước ở dạng bảng mà kết hợp trị số QP với số lượng đặc điểm hoặc số của các điểm đặc điểm được thu nhận từ thông tin về các điểm hoặc đối tượng được nhận diện dựa vào thông tin về các điểm. Trị số QP tương ứng có thể được xác định dựa vào kết quả của việc mô phỏng về hiệu quả mã hóa hoặc tương tự nhờ sử dụng các trị số QP khác nhau trong việc mã hóa của thông tin hình học liên quan đến dữ liệu đám mây điểm.

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số tham chiếu (trị số mặc định) của và thông tin sai lệch (đenta lượng tử hóa) về trị số QP đối với thông tin hình học (S5302). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định trị số tham chiếu và thông tin sai lệch được truyền nhờ sử dụng trị số QP được xác định theo cách thức được xác định

trước, và thiết đặt (bổ sung) trị số tham chiếu và thông tin sai lệch được xác định trong ít nhất một trong số thông tin bổ sung hoặc đoạn đầu của dữ liệu.

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số QP được sử dụng cho việc mã hóa của thông tin thuộc tính (S5303). Thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP đối với mỗi trong số các các lát phân chia theo cách thức được xác định trước. Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định trị số QP dựa vào các đặc tính hoặc chất lượng của dữ liệu của thông tin thuộc tính. Ví dụ, thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP trên cơ sở của các bộ phận dữ liệu dựa vào các đặc tính của thông tin thuộc tính. Màu sắc các đặc tính bao gồm độ chói, độ màu, và sắc độ, biểu đồ của nó, và tính liên tục màu sắc chẳng hạn. Khi thông tin thuộc tính là năng suất phản xạ, trị số QP có thể được xác định dựa vào thông tin dựa vào năng suất phản xạ. Ví dụ, khi mặt được phát hiện như đối tượng từ dữ liệu đám mây điểm, thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP chất lượng cao đối với dữ liệu đám mây điểm tạo nên đối tượng được phát hiện như khuôn mặt. Theo cách này, thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP đối với dữ liệu đám mây điểm tạo nên đối tượng tùy thuộc vào loại của đối tượng.

Khi điểm ba chiều có các đoạn thông tin thuộc tính, thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP khác nhau đối với mỗi đoạn của thông tin thuộc tính dựa vào đoạn của thông tin thuộc tính. Theo cách khác thiết bị mã hóa dữ liệu ba chiều có thể xác định trị số QP đối với các đoạn thông tin thuộc tính dựa vào bất kỳ một trong số các đoạn thông tin thuộc tính, hoặc xác định trị số QP đối với các đoạn thông tin thuộc tính dựa vào các đoạn thông tin thuộc tính.

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số tham chiếu (trị số mặc định) của và thông tin sai lệch (đenta lượng tử hóa) về trị số QP đối với thông tin thuộc tính (S5304). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định trị số tham chiếu và thông tin sai lệch được truyền nhờ sử dụng trị số QP được xác định theo cách thức được xác định trước, và thiết đặt (bổ sung) trị số tham chiếu và thông tin sai lệch được xác định trong ít nhất một trong số thông tin bổ sung hoặc đoạn đầu của dữ liệu.

Thiết bị mã hóa dữ liệu ba chiều sau đó lượng tử hóa và mã hóa thông tin hình học và thông tin thuộc tính dựa vào các trị số QP được xác định lần lượt đối với thông tin hình học và thông tin thuộc tính (S5305).

Lưu ý rằng mặc dù ví dụ đã được mô tả trong đó trị số QP đối với thông tin hình học được xác định dựa vào thông tin hình học, và trị số QP đối với thông tin thuộc tính

được xác định dựa vào thông tin thuộc tính, sáng chế là không giới hạn ở đó. Ví dụ, các trị số QP đối với thông tin hình học và thông tin thuộc tính có thể được xác định dựa vào thông tin hình học, dựa vào thông tin thuộc tính, hoặc dựa vào thông tin hình học và thông tin thuộc tính.

Lưu ý rằng các trị số QP đối với thông tin hình học và thông tin thuộc tính có thể được điều chỉnh nhờ xét đến sự cân bằng giữa chất lượng của thông tin hình học và chất lượng của thông tin thuộc tính trong dữ liệu đám mây điểm. Ví dụ, các trị số QP đối với thông tin hình học và thông tin thuộc tính có thể được thiết đặt theo cách sao cho chất lượng của thông tin hình học là cao, và chất lượng của thông tin thuộc tính là thấp hơn so với chất lượng của thông tin hình học. Ví dụ, trị số QP đối với thông tin thuộc tính có thể được xác định dưới sự hạn chế mà trị số QP đối với thông tin thuộc tính là bằng hoặc lớn hơn trị số QP đối với thông tin hình học.

Trị số QP có thể được điều chỉnh sao cho dữ liệu được mã hóa được tạo ra nằm trong khoảng được xác định trước của tỉ lệ. Ví dụ, khi lượng mã của việc mã hóa của đơn vị dữ liệu đứng trước được kỳ vọng vượt quá tỉ lệ được xác định trước, nghĩa là, khi độ chênh lệch với tỉ lệ được xác định trước là nhỏ hơn độ sai lệch thứ nhất, trị số QP có thể được điều chỉnh để làm giảm chất lượng tạo mã sao cho độ sai lệch giữa tỉ lệ được xác định trước và lượng mã của bộ phận dữ liệu là nhỏ hơn độ sai lệch thứ nhất. Mặt khác, khi độ chênh lệch với tỉ lệ được xác định trước là lớn hơn độ sai lệch thứ hai, mà là lớn hơn độ sai lệch thứ nhất, và có độ sai lệch thực chất, trị số QP có thể được điều chỉnh để nâng cao chất lượng tạo mã của bộ phận dữ liệu. việc điều chỉnh giữa các bộ phận dữ liệu có thể được thực hiện giữa các khung PCC hoặc giữa các mảnh hoặc các lát. việc điều chỉnh của trị số QP đối với thông tin thuộc tính có thể được thực hiện dựa vào tốc độ việc việc mã hóa của thông tin hình học.

Lưu ý rằng, trong lưu đồ trên Fig.102, việc xử lý liên quan đến thông tin hình học và việc xử lý liên quan đến thông tin thuộc tính có thể được thực hiện theo thứ tự ngược hoặc song song.

Lưu ý rằng, mặc dù lưu đồ trên Fig.102 thể hiện việc xử lý trên cơ sở lát như ví dụ, việc xử lý trên cơ sở mảnh hoặc quy trình dựa trên các bộ phận dữ liệu khác có thể được thực hiện theo cách thức như nhau như việc xử lý trên cơ sở lát. Nghĩa là, lát trong lưu đồ trên Fig.102 có thể được thay thế với mảnh hoặc các bộ phận dữ liệu khác.

Fig.103 là lưu đồ minh họa ví dụ về quy trình giải mã thông tin hình học và thông

tin thuộc tính.

Như được minh họa trên Fig.103, thiết bị giải mã dữ liệu ba chiều thu nhận trị số tham chiếu và thông tin sai lệch mà chỉ báo trị số QP đối với thông tin hình học, và trị số tham chiếu và thông tin sai lệch mà chỉ báo trị số QP đối với thông tin thuộc tính (S5311). Cụ thể là, thiết bị giải mã dữ liệu ba chiều phân tích một hoặc cả hai trong số siêu dữ liệu được truyền hoặc đoạn đầu của dữ liệu được mã hóa được truyền, và thu nhận các trị số tham chiếu và thông tin sai lệch để dẫn ra các trị số QP.

Thiết bị giải mã dữ liệu ba chiều sau đó dẫn ra các trị số QP nhờ sử dụng các trị số tham chiếu được thu nhận và thông tin sai lệch theo cách thức được xác định trước.

Thiết bị giải mã dữ liệu ba chiều sau đó thu nhận thông tin hình học được lượng tử hóa, và thu nhận thông tin hình học bằng cách lượng tử hóa ngược thông tin hình học được lượng tử hóa nhờ sử dụng số QP được dẫn ra (S5313).

Thiết bị giải mã dữ liệu ba chiều sau đó thu nhận thông tin thuộc tính được lượng tử hóa, và thu nhận thông tin thuộc tính bằng cách lượng tử hóa ngược thông tin thuộc tính được lượng tử hóa nhờ sử dụng số QP được dẫn ra (S5314).

Tiếp theo, phương pháp truyền thông số lượng tử hóa sẽ được mô tả.

Fig.104 là sơ đồ mô tả ví dụ thứ nhất về phương pháp truyền thông số lượng tử hóa. Phần (a) trên Fig.104 thể hiện ví dụ về mối tương quan giữa các trị số QP.

Trên Fig.104, Q_G và Q_A lần lượt biểu thị trị số tuyệt đối của trị số QP được sử dụng cho việc mã hóa của thông tin hình học và trị số tuyệt đối của trị số QP được sử dụng cho việc mã hóa của thông tin thuộc tính. Q_G là ví dụ về thông số lượng tử hóa thứ nhất được sử dụng cho việc lượng tử hóa thông tin hình học trên mỗi trong số các điểm ba chiều. $\Delta(Q_A, Q_G)$ biểu thị thông tin sai lệch mà chỉ báo độ sai lệch giữa Q_A và Q_G được sử dụng để dẫn ra Q_A. Nghĩa là, Q_A được dẫn ra nhờ sử dụng Q_G và $\Delta(Q_A, Q_G)$. Theo cách này, trị số QP được tách thành trị số tham chiếu (trị số tuyệt đối) và thông tin sai lệch (trị số tương đối) cho việc truyền. Trong việc giải mã, trị số QP mong muốn được dẫn ra từ trị số tham chiếu và thông tin sai lệch được truyền.

Ví dụ, ở phần (a) trên Fig.104, trị số tuyệt đối Q_G và thông tin sai lệch $\Delta(Q_A, Q_G)$ được truyền, và trong việc giải mã, Q_A được dẫn ra bằng cách bổ sung $\Delta(Q_A, Q_G)$ vào Q_G như được thể hiện bởi phương trình G5 dưới đây.

$$Q_A = Q_G + \Delta(Q_A, Q_G) \text{ (phương trình G5)}$$

Tham chiếu tới các phần (b) và (c) trên Fig.104, phương pháp truyền các trị số QP trong trường hợp trong đó dữ liệu đám mây điểm bao gồm thông tin hình học và thông tin thuộc tính được chia thành các lát sẽ được mô tả. Phần (b) trên Fig.104 thể hiện ví dụ thứ nhất về mối tương quan giữa trị số tham chiếu và thông tin sai lệch đối với mỗi trị số QP. Phần (c) trên Fig.104 thể hiện ví dụ thứ nhất về thứ tự truyền của các trị số QP, thông tin hình học, và thông tin thuộc tính.

Đối với mỗi đoạn của thông tin hình học và mỗi đoạn của thông tin thuộc tính, các trị số QP được phân loại thành các trị số QP (các khung QP) trong các bộ phận của các khung PCC và các trị số QP (các dữ liệu QP) trong các bộ phận của các bộ phận dữ liệu. Trị số QP được sử dụng cho việc mã hóa được xác định ở bước S5301 trên Fig.102 là trị số QP trong các bộ phận của các bộ phận dữ liệu.

Ở đây, Q_G, mà là trị số QP được sử dụng cho việc mã hóa của thông tin hình học trong các bộ phận của các khung PCC, được sử dụng như trị số tham chiếu, và trị số QP trong các bộ phận của các bộ phận dữ liệu được tạo ra và được truyền như thông tin sai lệch mà chỉ báo độ chênh lệch với Q_G.

Q_G: trị số QP cho việc mã hóa của thông tin hình học đối với khung PCC, mà được truyền như trị số tham chiếu "1." nhờ sử dụng GPS.

Q_A: trị số QP cho việc mã hóa của thông tin thuộc tính đối với khung PCC, mà được truyền như thông tin sai lệch "2." nhờ sử dụng APS.

Q_{Gs1}, Q_{Gs2}: các trị số QP cho việc mã hóa của thông tin hình học của dữ liệu lát, mà lần lượt được truyền như thông tin sai lệch "3." và "5." chỉ báo độ chênh lệch với Q_G, nhờ sử dụng đoạn đầu của dữ liệu được mã hóa của thông tin hình học.

Q_{As1}, Q_{As2}: các trị số QP cho việc mã hóa của thông tin thuộc tính của dữ liệu lát, mà lần lượt được truyền như thông tin sai lệch "4." và "6." chỉ báo độ chênh lệch với Q_A, nhờ sử dụng đoạn đầu của dữ liệu được mã hóa của thông tin thuộc tính.

Lưu ý rằng thông tin được sử dụng để dẫn ra khung QP được mô tả trong siêu dữ liệu (GPS, APS) được kết hợp với khung, và thông tin được sử dụng để dẫn ra dữ liệu QP được mô tả trong siêu dữ liệu (đoạn đầu của dữ liệu được mã hóa) được kết hợp với dữ liệu.

Theo cách này, dữ liệu QP được tạo ra và được truyền như thông tin sai lệch chỉ báo độ chênh lệch với khung QP. Do đó, lượng dữ liệu của dữ liệu QP có thể được giảm.

Trong mỗi đoạn dữ liệu được mã hóa, bộ giải mã thứ nhất 5340 đề cập đến siêu dữ liệu được chỉ báo bởi mũi tên ở phần (c) trên Fig.104, và thu nhận trị số tham chiếu và thông tin sai lệch được kết hợp với dữ liệu được mã hóa. Bộ giải mã thứ nhất 5340 sau đó dẫn ra trị số QP tương ứng với dữ liệu được mã hóa được giải mã dựa vào trị số tham chiếu được thu nhận và thông tin sai lệch.

Ví dụ, bộ giải mã thứ nhất 5340 thu nhận thông tin tham chiếu "1." và thông tin sai lệch "2." và "6." được chỉ báo bởi các mũi tên ở phần (c) trên Fig.104 từ siêu dữ liệu hoặc đoạn đầu, và dẫn ra trị số QP của As_2 bằng cách bổ sung thông tin sai lệch "2." và "6." tới thông tin tham chiếu "1." như được thể hiện bởi phương trình G6 dưới đây.

$$Q_{As2} = Q_G + \Delta(Q_A, Q_G) + \Delta(Q_{As2}, Q_A) \text{ (phương trình G6)}$$

Dữ liệu đám mây điểm bao gồm thông tin hình học và không hoặc nhiều hơn các đoạn thông tin thuộc tính. Nghĩa là, dữ liệu đám mây điểm có thể không bao gồm thông tin thuộc tính hoặc các đoạn thông tin thuộc tính.

Ví dụ, một điểm ba chiều có thể có, như thông tin thuộc tính, thông tin màu sắc, thông tin màu sắc và năng suất phản xạ thông tin, hoặc một hoặc nhiều đoạn của thông tin màu sắc được liên kết tới một hoặc nhiều đoạn của thông tin điểm ngắm.

Ở đây, ví dụ về trường hợp trong đó một điểm ba chiều có hai đoạn của thông tin màu sắc và năng suất phản xạ thông tin sẽ được mô tả dựa vào Fig.105. Fig.105 là sơ đồ mô tả ví dụ thứ hai về phương pháp truyền thông số lượng tử hóa. Phần (a) trên Fig.105 là sơ đồ minh họa ví dụ thứ hai về mối tương quan giữa trị số tham chiếu và thông tin sai lệch đối với mỗi trị số QP. Phần (b) trên Fig.105 là sơ đồ minh họa ví dụ thứ hai về thứ tự truyền của các trị số QP, thông tin hình học, và thông tin thuộc tính.

Q_G là ví dụ về thông số lượng tử hóa thứ nhất như trên Fig.104.

Hai các đoạn của thông tin màu sắc được chỉ báo lần lượt bởi độ chói (luma) Y và các độ màu (chroma) Cb, Cr. Q_{Y1}, trị số QP nào được sử dụng cho việc mã hóa của độ chói Y1 của màu sắc thứ nhất, được dẫn ra từ Q_G, mà là trị số tham chiếu, và $\Delta(Q_{Y1}, Q_G)$, mà chỉ báo độ sai lệch giữa Q_{Y1} và Q_G. Độ chói Y1 là ví dụ về độ chói thứ nhất, và Q_{Y1} là ví dụ về thông số lượng tử hóa thứ hai được sử dụng cho việc lượng tử hóa độ chói Y1 là độ chói thứ nhất. $\Delta(Q_{Y1}, Q_G)$ là thông tin sai lệch "2.".

Q_{Cb1} và Q_{Cr1}, mà là các trị số QP được sử dụng cho việc mã hóa của các độ màu Cb1 và Cr1 của màu sắc thứ nhất, được dẫn ra từ Q_{Y1} và $\Delta(Q_{Cb1}, Q_{Y1})$ và $\Delta(Q_{Cr1}, Q_{Y1})$, mà

lần lượt chỉ báo độ sai lệch giữa Q_{Cb1} và Q_{Y1} và độ sai lệch giữa Q_{Cr1} và Q_{Y1}. Các độ màu Cb1 và Cr1 là các ví dụ về độ màu thứ nhất, và Q_{Cb1} và Q_{Cr1} là các ví dụ về thông số lượng tử hóa thứ ba được sử dụng cho việc lượng tử hóa các độ màu Cb1 và Cr1 như độ màu thứ nhất. $\Delta(Q_{Cb1}, Q_{Y1})$ là thông tin sai lệch "3.", và $\Delta(Q_{Cr1}, Q_{Y1})$ là thông tin sai lệch "4.". $\Delta(Q_{Cb1}, Q_{Y1})$ và $\Delta(Q_{Cr1}, Q_{Y1})$ là các ví dụ về độ sai lệch thứ nhất.

Lưu ý rằng Q_{Cb1} và Q_{Cr1} có thể là các trị số giống hệt hoặc trị số chung. Khi trị số chung được sử dụng, một trong số Q_{Cb1} và Q_{Cr1} có thể được sử dụng, và trị số khác có thể được bỏ qua.

Q_{Y1D}, mà là trị số QP được sử dụng cho việc mã hóa của độ chói Y1D của màu sắc thứ nhất trong dữ liệu lát, được dẫn ra từ Q_{Y1} và $\Delta(Q_{Y1D}, Q_{Y1})$ chỉ báo độ sai lệch giữa Q_{Y1D} và Q_{Y1}. Độ chói Y1D của màu sắc thứ nhất trong dữ liệu lát là ví dụ về độ chói thứ nhất của một hoặc nhiều điểm ba chiều được bao gồm theo không gian con, và Q_{Y1D} là ví dụ về thông số lượng tử hóa thứ năm được sử dụng cho việc lượng tử hóa độ chói Y_{1D}. $\Delta(Q_{Y1D}, Q_{Y1})$ là thông tin sai lệch "10.", và ví dụ về độ sai lệch thứ hai.

Tương tự, Q_{Cb1D} và Q_{Cr1D}, mà là các trị số QP được sử dụng cho việc mã hóa của các độ màu Cb1D và Cr1D của màu sắc thứ nhất trong dữ liệu lát, lần lượt được dẫn ra từ Q_{Cb1} và $\Delta(Q_{Cb1D}, Q_{Cb1})$ chỉ báo độ sai lệch giữa Q_{Cb1D} và Q_{Cb1} và Q_{Cr1} và $\Delta(Q_{Cr1D}, Q_{Cr1})$ chỉ báo độ sai lệch giữa Q_{Cr1D} và Q_{Cr1}. Các độ màu Cb1D và Cr1D của màu sắc thứ nhất trong dữ liệu lát là các ví dụ về độ màu thứ nhất của một hoặc nhiều điểm ba chiều được bao gồm theo không gian con, và Q_{Cb1D} và Q_{Cr1D} là các ví dụ về thông số lượng tử hóa thứ sáu được sử dụng cho việc lượng tử hóa các độ màu Cb1D và Cr1D. $\Delta(Q_{Cb1D}, Q_{Cb1})$ là thông tin sai lệch "11.", và $\Delta(Q_{Cr1D}, Q_{Cr1})$ là thông tin sai lệch "12.". $\Delta(Q_{Cb1D}, Q_{Cb1})$ và $\Delta(Q_{Cr1D}, Q_{Cr1})$ là các ví dụ về độ sai lệch thứ ba.

Mối tương quan giữa các trị số QP đối với màu sắc thứ nhất giữ nguyên đối với màu sắc thứ hai, sao cho các phần mô tả thừa sẽ được bỏ qua.

Q_R, mà là trị số QP được sử dụng cho việc mã hóa của năng suất phản xạ R, được dẫn ra từ Q_G, mà là trị số tham chiếu, và $\Delta(Q_R, Q_G)$, mà chỉ báo độ sai lệch giữa Q_R và Q_G. Q_R là ví dụ về thông số lượng tử hóa thứ tư được sử dụng cho việc lượng tử hóa năng suất phản xạ R. $\Delta(Q_R, Q_G)$ là thông tin sai lệch "8.".

Q_{RD}, mà là trị số QP được sử dụng cho việc mã hóa của năng suất phản xạ RD trong dữ liệu lát, được dẫn ra từ Q_R và $\Delta(Q_{RD}, Q_R)$, mà chỉ báo độ sai lệch giữa Q_{RD} và Q_R. $\Delta(Q_{RD}, Q_R)$ là thông tin sai lệch "16.".

Như được nêu trên, thông tin sai lệch "9." Đến "16." chỉ báo độ sai lệch giữa dữ liệu QP và khung QP.

Lưu ý rằng khi các trị số của dữ liệu QP và khung QP là như nhau, ví dụ, thông tin sai lệch có thể được thiết đặt ở 0, hoặc có thể không được truyền, và vắng mặt của việc truyền có thể được coi là thông tin sai lệch là 0.

Khi thu nhận độ màu Cr2 của màu sắc thứ hai bằng cách giải mã, ví dụ, bộ giải mã thứ nhất 5340 thu nhận thông tin tham chiếu "1." và thông tin sai lệch "5.", "7.", và "15." được chỉ báo bởi các mũi tên ở phần (b) trên Fig.105 từ siêu dữ liệu hoặc đoạn đầu, và dẫn ra trị số QP của độ màu Cr2 bằng cách bổ sung thông tin sai lệch "5.", "7.", và "15." tới thông tin tham chiếu "1." như được thể hiện bởi phương trình G7 dưới đây.

$$Q_{Cr2D} = Q_G + \Delta(Q_{Y2}, Q_G) + \Delta(Q_{Cr2}, Q_{Y2}) + \Delta(Q_{Cr2D}, Q_{Cr2}) \text{ (phương trình G7)}$$

Tiếp theo, ví dụ về trường hợp trong đó thông tin hình học và thông tin thuộc tính được chia thành hai mảnh và sau đó được chia thành hai lát sẽ được mô tả dựa vào Fig.106. Fig.106 là sơ đồ mô tả ví dụ thứ ba về phương pháp truyền thông số lượng tử hóa. Phần (a) trên Fig.106 thể hiện ví dụ thứ ba về mối tương quan giữa trị số tham chiếu và thông tin sai lệch đối với mỗi trị số QP. Phần (b) trên Fig.106 thể hiện ví dụ thứ ba về thứ tự truyền của các trị số QP, thông tin hình học, và thông tin thuộc tính. Phần (c) trên Fig.106 mô tả trị số được tạo ra trung bình đối với thông tin sai lệch theo ví dụ thứ ba.

Khi thông tin hình học và thông tin thuộc tính được chia thành các mảnh và sau đó còn được chia thành các lát, như được minh họa ở phần (c) trên Fig.106, sau khi thông tin thuộc tính được chia thành các mảnh, trị số QP (Q_{At1}) và thông tin sai lệch $\Delta(Q_{At1}, Q_A)$ đối với mỗi mảnh được tạo ra như các trị số được tạo ra trung bình. Sau khi mảnh được chia thành các lát, các trị số QP (Q_{At1s1}, Q_{At1s2}) và thông tin sai lệch ($\Delta(Q_{At1s1}, Q_{At1})$, $\Delta(Q_{At1s2}, Q_{At1})$) được tạo ra đối với mỗi lát.

Trong trường hợp này, thông tin sai lệch "4." ở phần (a) trên Fig.106 được dẫn ra theo phương trình G8 dưới đây.

$$\Delta(Q_{At1s1}, Q_A) = \Delta(Q_{At1}, Q_A) + \Delta(Q_{At1s1}, Q_{At1}) \text{ (phương trình G8)}$$

Khi thu nhận thông tin thuộc tính A_{t2s1} đối với lát 1 trong mảnh 2 bằng cách giải mã, ví dụ, bộ giải mã thứ nhất 5340 thu nhận thông tin tham chiếu "1." và thông tin sai lệch "2." và "8." được chỉ báo bởi các mũi tên ở phần (b) trên Fig.106 từ siêu dữ liệu hoặc đoạn đầu, và dẫn ra trị số QP của thông tin thuộc tính At_2s_1 bằng cách bổ sung thông tin sai lệch

"2." và "8." tới thông tin tham chiếu "1." như được thể hiện bởi phương trình G9 dưới đây.

$$Q_{At2s1} = Q_G + \Delta(Q_{At2s1}, Q_A) + \Delta(Q_A, Q_G) \text{ (phương trình G9)}$$

Tiếp theo, tiến trình của quy trình mã hóa dữ liệu đám mây điểm và tiến trình của quy trình giải mã dữ liệu đám mây điểm theo phương án này sẽ được mô tả. Fig.107 là lưu đồ của quy trình mã hóa dữ liệu đám mây điểm theo phương án này.

Thứ nhất, thiết bị mã hóa dữ liệu ba chiều xác định phương pháp phân chia được sử dụng (S5321). Phương pháp phân chia bao gồm việc xác định xem có thực hiện việc chia mảnh hay không và việc xác định xem có thực hiện việc chia lát hay không. Phương pháp phân chia có thể bao gồm số của các mảnh hoặc các lát trong trường hợp trong đó việc chia mảnh hoặc việc chia lát được thực hiện, và loại của việc phân chia chẳng hạn. Loại của việc phân chia là sơ đồ dựa vào hình dạng đối tượng, sơ đồ dựa vào thông tin bản đồ hoặc thông tin hình học, hoặc sơ đồ dựa vào lượng dữ liệu hoặc lượng xử lý chẳng hạn. Lưu ý rằng phương pháp phân chia có thể được xác định trước.

Khi việc chia mảnh được thực hiện (nếu Có ở S5322), thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách chia thông tin hình học và thông tin thuộc tính về cơ sở mảnh (S5323). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin bổ sung mảnh liên quan đến việc chia mảnh.

Khi việc chia lát được thực hiện (nếu Có ở S5324), thiết bị mã hóa dữ liệu ba chiều tạo ra các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia bằng cách chia các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh (hoặc thông tin hình học và thông tin thuộc tính) (S5325). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính liên quan đến việc chia lát.

Thiết bị mã hóa dữ liệu ba chiều sau đó tạo ra các đoạn của thông tin hình học được mã hóa và các đoạn của thông tin thuộc tính được mã hóa bằng cách mã hóa mỗi trong số các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia (S5326). Thiết bị mã hóa dữ liệu ba chiều cũng tạo ra thông tin phụ thuộc.

Thiết bị mã hóa dữ liệu ba chiều sau đó tạo ra dữ liệu được mã hóa (dòng được mã hóa) bằng cách tích hợp (đa hợp) các đoạn của thông tin hình học được mã hóa, các đoạn của thông tin thuộc tính được mã hóa và thông tin bổ sung trong bộ phận NAL (S5327).

Thiết bị mã hóa dữ liệu ba chiều cũng truyền dữ liệu được mã hóa được tạo ra.

Fig.108 là lưu đồ minh họa ví dụ về quy trình xác định trị số QP và cập nhật thông tin bổ sung trong việc chia mảnh (S5323) và việc chia lát (S5325).

Ở các bước S5323 và S5325, thông tin hình học mảnh và thông tin thuộc tính mảnh và/hoặc lát thông tin hình học và thông tin thuộc tính lát có thể được chia một cách độc lập theo các cách thức tương ứng, hoặc có thể là một cách tập hợp được chia theo cách thức chung. Theo cách này, thông tin bổ sung được chia trên cơ sở mảnh và/hoặc trên cơ sở lát được tạo ra.

Ở các bước này, thiết bị mã hóa dữ liệu ba chiều xác định trị số tham chiếu và thông tin sai lệch đối với trị số QP trên cơ sở mảnh phân chia và/hoặc trên cơ sở lát phân chia (S5331). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều xác định trị số tham chiếu và thông tin sai lệch chẳng hạn như được minh họa trên các hình vẽ từ Fig.104 đến Fig.106.

Thiết bị mã hóa dữ liệu ba chiều sau đó cập nhật thông tin bổ sung để bao gồm trị số tham chiếu và thông tin sai lệch được xác định (S5332).

Fig.109 là lưu đồ minh họa ví dụ về quy trình trong việc mã hóa (S5326).

Thiết bị mã hóa dữ liệu ba chiều mã hóa mỗi trong số các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia (S5341). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều mã hóa mỗi trong số các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia nhờ sử dụng trị số QP được xác định.

Thiết bị mã hóa dữ liệu ba chiều sau đó tiếp tục quy trình mã hóa đến khi điều kiện để dừng quy trình mã hóa được thỏa mãn, chẳng hạn như đến khi không có dữ liệu được mã hóa (S5342).

Fig.110 là lưu đồ minh họa quy trình giải mã dữ liệu đám mây điểm theo phương án này. Thứ nhất, thiết bị giải mã dữ liệu ba chiều xác định phương pháp phân chia nhờ phân tích thông tin bổ sung (thông tin bổ sung mảnh, thông tin bổ sung lát hình học, và thông tin bổ sung lát thuộc tính) liên quan đến phương pháp phân chia được bao gồm trong dữ liệu được mã hóa (dòng được mã hóa) (S5351). Phương pháp phân chia bao gồm việc xác định xem có thực hiện việc chia mảnh hay không và việc xác định xem có thực hiện việc chia lát hay không. Phương pháp phân chia có thể bao gồm số của các mảnh hoặc các lát trong trường hợp trong đó việc chia mảnh hoặc việc chia lát được thực hiện, và loại của việc phân chia chẳng hạn.

Thiết bị giải mã dữ liệu ba chiều sau đó tạo ra thông tin hình học phân chia và thông tin thuộc tính phân chia bằng cách giải mã các đoạn của thông tin hình học được mã hóa và các đoạn của thông tin thuộc tính được mã hóa được bao gồm trong dữ liệu được mã hóa nhờ sử dụng thông tin phụ thuộc được bao gồm trong dữ liệu được mã hóa (S5352).

Nếu thông tin bổ sung chỉ báo rằng việc chia lát đã được thực hiện (nếu Có ở S5353), thiết bị giải mã dữ liệu ba chiều tạo ra các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh bằng cách kết hợp các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia dựa vào thông tin bổ sung lát hình học và thông tin bổ sung lát thuộc tính (S5354).

Nếu thông tin bổ sung chỉ báo rằng việc chia mảnh đã được thực hiện (nếu Có ở S5355), thiết bị giải mã dữ liệu ba chiều tạo ra thông tin hình học và thông tin thuộc tính bằng cách kết hợp các đoạn của thông tin hình học mảnh và các đoạn của thông tin thuộc tính mảnh (các đoạn của thông tin hình học phân chia và các đoạn của thông tin thuộc tính phân chia) dựa vào thông tin bổ sung mảnh (S5356).

Fig.111 là lưu đồ minh họa ví dụ về quy trình thu nhận các trị số QP và giải mã trị số QP đối với lát hoặc mảnh trong việc kết hợp của thông tin được chia thành các lát (S5354) và kết hợp của thông tin được chia thành các mảnh (S5356).

Các đoạn của lát thông tin hình học và các đoạn của thông tin thuộc tính lát hoặc các đoạn của thông tin hình học mảnh hoặc các đoạn của thông tin thuộc tính mảnh có thể được kết hợp theo các cách thức tương ứng hoặc theo cách thức như nhau.

Thiết bị giải mã dữ liệu ba chiều thu nhận trị số tham chiếu và thông tin sai lệch bằng cách giải mã thông tin bổ sung trong dòng được mã hóa (S5361).

Thiết bị giải mã dữ liệu ba chiều sau đó tính trị số lượng tử hóa nhờ sử dụng trị số tham chiếu được giải mã và thông tin sai lệch, và cập nhật trị số QP được sử dụng cho việc lượng tử hóa ngược tới trị số QP được tính (S5362). Theo cách này, trị số QP cho việc lượng tử hóa ngược của thông tin thuộc tính được lượng tử hóa đối với mỗi mảnh hoặc lát có thể được dẫn ra.

Thiết bị giải mã dữ liệu ba chiều sau đó tiếp tục quy trình giải mã đến khi điều kiện để dừng quy trình giải mã được thỏa mãn, chẳng hạn như đến khi không có dữ liệu được giải mã (S5363).

Fig.112 là sơ đồ minh họa ví dụ cú pháp của GPS. Fig.113 là sơ đồ minh họa ví dụ

cú pháp của APS. Fig.114 là sơ đồ minh họa ví dụ cú pháp của đoạn đầu của thông tin hình học. Fig.115 là sơ đồ minh họa ví dụ cú pháp của đoạn đầu của thông tin thuộc tính.

Như được minh họa trên Fig.112, ví dụ, GPS, mà là thông tin bổ sung của thông tin hình học, bao gồm trị số QP (QP_value), mà chỉ báo trị số tuyệt đối được sử dụng làm tham chiếu để dẫn ra trị số QP. QP_value tương ứng với Q_G được minh họa trên trên các hình vẽ từ Fig.104 đến Fig.106.

Như được minh họa trên Fig.113, ví dụ, khi điểm ba chiều có các đoạn của thông tin màu sắc được kết hợp với các điểm ngắm, APS, mà là thông tin bổ sung của thông tin thuộc tính, có thể định rõ điểm ngắm mặc định, và đoạn thứ 0 của thông tin thuộc tính có thể luôn mô tả thông tin trên điểm ngắm mặc định. Ví dụ, khi giải mã hoặc hiển thị đoạn đơn của thông tin màu sắc, thiết bị mã hóa dữ liệu ba chiều có thể giải mã hoặc hiển thị đoạn thứ 0 của thông tin thuộc tính.

APS bao gồm QP_delta_Attribute_to_Geometry. QP_delta_Attribute_to_Geometry là thông tin sai lệch chỉ báo độ chênh lệch với trị số tham chiếu (QP_value) được mô tả trong GPS. Thông tin sai lệch chỉ báo độ sai lệch về độ chói khi thông tin thuộc tính là thông tin màu sắc chẳng hạn.

GPS có thể bao gồm cờ mà chỉ báo xem Geometry_header (đoạn đầu của thông tin hình học) bao gồm thông tin sai lệch được sử dụng để tính trị số QP hay không. APS có thể bao gồm cờ mà chỉ báo xem attribute_header (đoạn đầu của thông tin thuộc tính) bao gồm thông tin sai lệch được sử dụng để tính trị số QP hay không. Cờ có thể chỉ báo xem thông tin thuộc tính bao gồm thông tin sai lệch chỉ báo độ sai lệch của dữ liệu QP từ khung QP, mà được sử dụng để tính dữ liệu QP, hay không.

Khi màu sắc thứ nhất của thông tin thuộc tính được chỉ báo bởi độ chói thứ nhất và độ màu thứ nhất, trong việc lượng tử hóa của độ chói thứ nhất nhờ sử dụng thông số lượng tử hóa thứ hai và việc lượng tử hóa của độ màu thứ nhất nhờ sử dụng thông số lượng tử hóa thứ ba, nếu các việc lượng tử hóa được thực hiện nhờ sử dụng thông số lượng tử hóa thứ năm và thông số lượng tử hóa thứ sáu, dòng được mã hóa có thể bao gồm thông tin nhận dạng (flag) mà chỉ báo rằng các việc lượng tử hóa được thực hiện nhờ sử dụng thông số lượng tử hóa thứ năm và thông số lượng tử hóa thứ sáu.

Như được minh họa trên Fig.114, đoạn đầu của thông tin hình học có thể bao gồm QP_delta_data_to_frame, mà là thông tin sai lệch chỉ báo độ chênh lệch với trị số tham chiếu (QP_value) được mô tả trong GPS. Đoạn đầu của thông tin hình học có thể được chia

thành các đoạn của thông tin được kết hợp với các mảnh và/hoặc các lát, và trị số QP tương ứng với mỗi mảnh và/hoặc lát có thể được chỉ báo.

Như được minh họa trên Fig.115, đoạn đầu của thông tin thuộc tính có thể bao gồm dữ liệu đenta QP tới khung (QP_delta_data_to_frame), mà là thông tin sai lệch chỉ báo độ chênh lệch với trị số QP được mô tả trong APS.

Mặc dù trị số tham chiếu của trị số QP đã được mô tả là trị số QP của thông tin hình học đối với khung PCC dựa vào trên các hình vẽ từ Fig.104 đến Fig.106, sáng chế là không giới hạn ở đó, và các trị số khác có thể được sử dụng như trị số tham chiếu.

Fig.116 là sơ đồ mô tả ví dụ khác về phương pháp truyền thông số lượng tử hóa.

Các phần (a) và (b) trên Fig.116 minh họa ví dụ thứ tư, trong đó trị số tham chiếu chung Q được thiết đặt dựa vào các trị số QP của thông tin hình học và thông tin thuộc tính đối với khung PCC. Theo ví dụ thứ tư, trị số tham chiếu Q được lưu trữ trong GPS, thông tin sai lệch chỉ báo độ sai lệch của trị số QP (Q_G) của thông tin hình học từ trị số tham chiếu Q được lưu trữ trong GPS, và thông tin sai lệch chỉ báo các độ sai lệch của các trị số QP (Q_Y và Q_R) của thông tin thuộc tính từ trị số tham chiếu Q được lưu trữ trong APS. Lưu ý rằng trị số tham chiếu Q có thể được lưu trữ ở SPS.

Các phần (c) và (d) trên Fig.116 minh họa ví dụ thứ năm, trong đó trị số tham chiếu khác nhau được thiết đặt đối với mỗi trong số thông tin hình học và thông tin thuộc tính. Theo ví dụ thứ năm, các trị số QP tham chiếu (các trị số tuyệt đối) của thông tin hình học và thông tin thuộc tính lần lượt được lưu trữ trong GPS và APS. Nghĩa là, trị số tham chiếu Q_G được thiết đặt trong thông tin hình học, trị số tham chiếu Q_Y được thiết đặt trong thông tin màu sắc của thông tin thuộc tính, và trị số tham chiếu Q_R được thiết đặt là năng suất phản xạ của thông tin thuộc tính. Theo cách này, trị số tham chiếu của trị số QP có thể được thiết đặt đối với mỗi trong số thông tin hình học và các loại của thông tin thuộc tính. Lưu ý rằng ví dụ thứ năm có thể được kết hợp với ví dụ khác. Nghĩa là, Q_A theo ví dụ thứ nhất, hoặc Q_{Y1}, Q_{Y2}, và Q_R theo ví dụ thứ hai có thể là trị số tham chiếu của trị số QP.

Các phần (e) và (f) trên Fig.116 minh họa ví dụ thứ sáu, trong đó khi có các khung PCC, trị số tham chiếu chung Q được thiết đặt đối với các khung PCC. Theo ví dụ thứ sáu, trị số tham chiếu Q được lưu trữ ở SPS hoặc GPS, và thông tin sai lệch chỉ báo độ sai lệch giữa trị số QP của thông tin hình học và trị số tham chiếu đối với mỗi khung PCC được lưu trữ trong GPS. Lưu ý rằng, nằm trong khoảng của bộ phận truy cập ngẫu nhiên, chẳng hạn như GOF, ví dụ, leading khung của bộ phận truy cập ngẫu nhiên có thể được ấn định như

trị số tham chiếu, và thông tin sai lệch $\Delta(Q_{G(1)}, Q_{G(0)})$ chỉ báo các độ sai lệch giữa các khung PCC có thể được truyền.

Lưu ý rằng, ngay cả khi mảnh hoặc lát còn được chia, thông tin sai lệch chỉ báo độ chênh lệch với trị số QP của bộ phận của việc phân chia được lưu trữ trong dữ liệu đoạn đầu và được truyền theo cách thức như nhau.

Fig.117 là sơ đồ mô tả ví dụ khác về phương pháp truyền thông số lượng tử hóa.

Các phần (a) và (b) trên Fig.117 minh họa ví dụ thứ bảy, trong đó trị số tham chiếu chung Q_G được thiết đặt đối với thông tin hình học và thông tin thuộc tính của khung PCC. Theo ví dụ thứ bảy, trị số tham chiếu Q_G được lưu trữ trong GPS, và thông tin sai lệch chỉ báo các độ sai lệch từ thông tin hình học hoặc thông tin thuộc tính được lưu trữ các đoạn đầu dữ liệu tương ứng. Trị số tham chiếu Q_G có thể được lưu trữ ở SPS.

Các phần (c) và (d) trên Fig.117 thể hiện ví dụ thứ tám, trong đó trị số QP của thông tin thuộc tính được chỉ báo bởi thông tin sai lệch chỉ báo độ chênh lệch với trị số QP của thông tin hình học thuộc về cùng lát và mảnh. Theo ví dụ thứ tám, trị số tham chiếu Q_G có thể được lưu trữ ở SPS.

Fig.118 là sơ đồ mô tả ví dụ thứ chín về phương pháp truyền thông số lượng tử hóa.

Các phần (a) và (b) minh họa ví dụ thứ chín, trong đó các đoạn thông tin thuộc tính có trị số QP chung, và mỗi đoạn của thông tin thuộc tính chỉ báo thông tin sai lệch chỉ báo độ sai lệch giữa trị số QP chung và trị số QP của thông tin hình học và thông tin sai lệch chỉ báo độ chênh lệch với trị số của thông tin thuộc tính và trị số QP chung.

Fig.119 là sơ đồ mô tả ví dụ về việc điều khiển của trị số QP.

Đối với trị số của thông số lượng tử hóa giảm, chất lượng nâng cao, trong khi hiệu quả mã hóa giảm bởi vì nhiều hơn các bit được yêu cầu.

Ví dụ, khi mã hóa dữ liệu đám mây điểm ba chiều bằng cách chia dữ liệu đám mây điểm ba chiều thành các mảnh, nếu dữ liệu đám mây điểm được bao gồm trong mảnh là tuyến đường chính, dữ liệu đám mây điểm được mã hóa nhờ sử dụng trị số QP được định rõ trước của thông tin thuộc tính. Mặt khác, các mảnh ngoại vi không bao gồm thông tin quan trọng, và do đó, hiệu quả mã hóa có thể có khả năng được nâng cao bằng cách thiết đặt thông tin sai lệch của trị số QP ở trị số dương làm giảm chất lượng của dữ liệu.

Hơn nữa, khi mã hóa dữ liệu đám mây điểm ba chiều được chia thành các mảnh

bằng cách chia các mảnh thành các lát, vỉa hè, cây, và công trình xây dựng là quan trọng cho việc đánh giá vị trí (xác định vị trí và lập bản đồ) trong lái xe tự động, sao cho trị số QP được thiết đặt ở trị số âm. Mặt khác, vật thể di chuyển và các đối tượng khác là ít quan trọng, sao cho trị số QP được thiết đặt ở trị số dương.

Phần (b) trên Fig.119 thể hiện ví dụ trong đó thông tin sai lệch được dẫn ra trong trường hợp trong đó đenta lượng tử hóa trị số được thiết đặt trước dựa vào đối tượng được bao gồm trong mảnh hoặc lát. Ví dụ, khi dữ liệu phân chia là dữ liệu lát về "công trình xây dựng" được bao gồm trong mảnh của "tuyến đường chính", thông tin sai lệch là -5, mà được dẫn ra bằng cách lấy tổng đenta lượng tử hóa trị số là 0 của mảnh của "tuyến đường chính" và đenta lượng tử hóa trị số là -5 của dữ liệu lát về "công trình xây dựng".

Fig.120 là lưu đồ minh họa ví dụ về phương pháp xác định trị số QP dựa vào chất lượng của đối tượng.

Thiết bị mã hóa dữ liệu ba chiều chia dữ liệu đám mây điểm thành một hoặc nhiều mảnh dựa vào thông tin bản đồ, và xác định đối tượng được bao gồm trong một hoặc nhiều mảnh (S5371). Cụ thể là, thiết bị mã hóa dữ liệu ba chiều thực hiện quy trình nhận diện đối tượng để nhận diện đối tượng nào nhờ sử dụng mô hình học được thu nhận bằng cách học máy chẳng hạn.

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định xem có mã hóa mảnh được xử lý với chất lượng cao hay không (S5372). Để mã hóa với chất lượng cao có nghĩa là mã hóa ở tốc độ bit lớn hơn tỉ lệ được xác định trước chẳng hạn.

Khi mảnh được xử lý được mã hóa với chất lượng cao (nếu Có ở S5372), thiết bị mã hóa dữ liệu ba chiều sau đó thiết đặt trị số QP của mảnh sao cho hiệu quả mã hóa là cao (S5373).

Mặt khác, khi mảnh được xử lý không được mã hóa với chất lượng cao (nếu Không ở S5372), thiết bị mã hóa dữ liệu ba chiều thiết đặt trị số QP của mảnh sao cho hiệu quả mã hóa là thấp (S5374).

Theo sau bước S5373 hoặc S5374, thiết bị mã hóa dữ liệu ba chiều xác định đối tượng trong mảnh, và chia mảnh thành một hoặc nhiều lát (S5375).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định xem có mã hóa lát được xử lý với chất lượng cao hay không (S5376).

Khi lát được xử lý được mã hóa với chất lượng cao (nếu Có ở S5376), thiết bị mã

hóa dữ liệu ba chiều sau đó thiết đặt trị số QP của lát sao cho hiệu quả mã hóa là cao (S5377).

Mặt khác, khi lát được xử lý không được mã hóa với chất lượng cao (nếu Không ở S5376), thiết bị mã hóa dữ liệu ba chiều thiết đặt trị số QP của lát sao cho hiệu quả mã hóa là thấp (S5378).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số tham chiếu và thông tin sai lệch được truyền dựa vào set trị số QP theo cách thức được xác định trước, và lưu trữ trị số tham chiếu và thông tin sai lệch được xác định trong ít nhất một trong số thông tin bổ sung và đoạn đầu của dữ liệu (S5379).

Thiết bị mã hóa dữ liệu ba chiều sau đó lượng tử hóa và mã hóa thông tin hình học và thông tin thuộc tính dựa vào trị số QP được xác định (S5380).

Fig.121 là lưu đồ minh họa ví dụ về phương pháp xác định trị số QP dựa vào việc điều khiển tốc độ.

Thiết bị mã hóa dữ liệu ba chiều mã hóa liên tục dữ liệu đám mây điểm (S5381).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định việc điều khiển tốc độ status liên quan đến quy trình mã hóa từ lượng mã của dữ liệu được mã hóa và sự chiếm giữ của bộ đệm mã hóa, và xác định chất lượng của việc mã hóa tiếp theo (S5382).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định xem có làm tăng chất lượng mã hóa (S5383) hay không.

Khi chất lượng mã hóa được tăng lên (nếu Có ở S5383), thiết bị mã hóa dữ liệu ba chiều thiết đặt trị số QP của mảnh sao cho chất lượng mã hóa là cao hơn (S5384).

Mặt khác, khi chất lượng mã hóa không được tăng lên (nếu Không ở S5383), thiết bị mã hóa dữ liệu ba chiều thiết đặt trị số QP của mảnh sao cho chất lượng mã hóa là thấp hơn (S5385).

Thiết bị mã hóa dữ liệu ba chiều sau đó xác định trị số tham chiếu và thông tin sai lệch được truyền dựa vào set trị số QP theo cách thức được xác định trước, và lưu trữ trị số tham chiếu và thông tin sai lệch được xác định trong ít nhất một trong số thông tin bổ sung và đoạn đầu của dữ liệu (S5386).

Thiết bị mã hóa dữ liệu ba chiều sau đó lượng tử hóa và mã hóa thông tin hình học và thông tin thuộc tính dựa vào trị số QP được xác định (S5387).

Như được nêu trên, thiết bị mã hóa dữ liệu ba chiều theo phương án này thực hiện quy trình được minh họa trên Fig.122. Thứ nhất, thiết bị mã hóa dữ liệu ba chiều lượng tử hóa thông tin hình học trên mỗi trong số các điểm ba chiều nhờ sử dụng thông số lượng tử hóa thứ nhất (S5391). Thiết bị mã hóa dữ liệu ba chiều lượng tử hóa độ chói thứ nhất nhờ sử dụng thông số lượng tử hóa thứ hai và lượng tử hóa độ màu thứ nhất nhờ sử dụng thông số lượng tử hóa thứ ba, độ chói thứ nhất và độ màu thứ nhất chỉ báo màu sắc thứ nhất trong số thông tin thuộc tính trên mỗi trong số các điểm ba chiều (S5392). Thiết bị mã hóa dữ liệu ba chiều tạo ra dòng bit bao gồm thông tin hình học được lượng tử hóa, độ chói thứ nhất được lượng tử hóa, độ màu thứ nhất được lượng tử hóa, thông số lượng tử hóa thứ nhất, thông số lượng tử hóa thứ hai, và độ sai lệch thứ nhất giữa thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ ba (S5393).

Với cấu hình như vậy, vì thông số lượng tử hóa thứ ba được chỉ báo bởi độ chênh lệch thứ nhất với thông số lượng tử hóa thứ hai trong dòng bit, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều còn lượng tử hóa năng suất phản xạ trong số thông tin thuộc tính trên mỗi trong số các điểm ba chiều nhờ sử dụng thông số lượng tử hóa thứ tư. Hơn nữa, trong việc tạo ra nêu trên, dòng bit được tạo ra còn bao gồm năng suất phản xạ được lượng tử hóa và thông số lượng tử hóa thứ tư.

Ví dụ, trong việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ hai, đối với mỗi trong số các không gian con được thu nhận bằng cách chia không gian hiện thời bao gồm các điểm ba chiều, độ chói thứ nhất của một hoặc nhiều điểm ba chiều được bao gồm theo không gian con được lượng tử hóa còn sử dụng thông số lượng tử hóa thứ năm. Trong việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ ba, độ màu thứ nhất của một hoặc nhiều điểm ba chiều được lượng tử hóa còn sử dụng thông số lượng tử hóa thứ sáu. Trong việc tạo ra nêu trên, dòng bit được tạo ra còn bao gồm độ sai lệch thứ hai giữa thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ năm và độ sai lệch thứ ba giữa thông số lượng tử hóa thứ ba và thông số lượng tử hóa thứ sáu.

Với cấu hình như vậy, vì thông số lượng tử hóa thứ năm được chỉ báo bởi độ chênh lệch thứ hai với thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ sáu được chỉ báo bởi độ chênh lệch thứ ba với thông số lượng tử hóa thứ ba trong dòng bit, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, trong việc tạo ra nêu trên, dòng bit được tạo ra còn bao gồm thông tin nhận

dạng chỉ báo rằng thông số lượng tử hóa thứ năm và thông số lượng tử hóa thứ sáu lần lượt đã được sử dụng trong việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ hai và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ ba.

Với cấu hình như vậy, thiết bị giải mã dữ liệu ba chiều đã thu nhận dòng bit có thể xác định từ thông tin nhận dạng mà việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ năm và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ sáu đã được thực hiện, sao cho tải xử lý của quy trình giải mã có thể được giảm.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều còn lượng tử hóa độ chói thứ hai nhờ sử dụng thông số lượng tử hóa thứ bảy và lượng tử hóa độ màu thứ hai nhờ sử dụng thông số lượng tử hóa thứ tám, độ chói thứ hai và độ màu thứ hai chỉ báo màu sắc thứ hai trong số thông tin thuộc tính của mỗi trong số các điểm ba chiều. Trong việc tạo ra nêu trên, dòng bit được tạo ra còn bao gồm độ chói thứ hai được lượng tử hóa, độ màu thứ hai được lượng tử hóa, thông số lượng tử hóa thứ bảy, và độ sai lệch thứ tư giữa thông số lượng tử hóa thứ bảy và thông số lượng tử hóa thứ tám.

Với cấu hình như vậy, vì thông số lượng tử hóa thứ tám được chỉ báo bởi độ chênh lệch thứ tư với thông số lượng tử hóa thứ bảy trong dòng bit, hiệu quả mã hóa có thể được nâng cao. Ngoài ra, hai loại của thông tin màu sắc có thể được bao gồm trong thông tin thuộc tính trên điểm ba chiều.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Thiết bị giải mã dữ liệu ba chiều theo phương án này thực hiện quy trình được minh họa trên Fig.123. Thứ nhất, thiết bị giải mã dữ liệu ba chiều thu nhận thông tin hình học được lượng tử hóa, độ chói thứ nhất được lượng tử hóa, độ màu thứ nhất được lượng tử hóa, thông số lượng tử hóa thứ nhất, thông số lượng tử hóa thứ hai, và độ sai lệch thứ nhất giữa thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ ba, bằng cách thu nhận dòng bit (S5394). Thiết bị giải mã dữ liệu ba chiều tính thông tin hình học về các điểm ba chiều bằng cách lượng tử hóa ngược thông tin hình học được lượng tử hóa nhờ sử dụng thông tin lượng tử hóa thứ nhất (S5395). Trong số độ chói thứ nhất và độ màu thứ nhất chỉ báo màu sắc thứ nhất của các điểm ba chiều, thiết bị giải mã dữ liệu ba chiều tính độ chói thứ nhất bằng cách lượng tử hóa ngược độ chói thứ nhất được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ hai (S5396). Thiết bị giải mã dữ liệu ba chiều tính độ màu thứ nhất bằng cách lượng tử hóa ngược độ màu thứ nhất được lượng tử hóa nhờ sử dụng

thông số lượng tử hóa thứ ba được thu nhận từ thông số lượng tử hóa thứ hai và độ sai lệch thứ nhất (S5397).

Theo cách này, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác thông tin hình học và thông tin thuộc tính trên điểm ba chiều.

Ví dụ, trong việc thu nhận, năng suất phản xạ được lượng tử hóa và thông số lượng tử hóa thứ tư còn được thu nhận bằng cách thu nhận dòng bit. Thiết bị giải mã dữ liệu ba chiều còn tính năng suất phản xạ của các điểm ba chiều bằng cách lượng tử hóa ngược năng suất phản xạ được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ tư.

Do đó, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác năng suất phản xạ của điểm ba chiều.

Ví dụ, trong việc thu nhận, độ sai lệch thứ hai giữa thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ năm và độ sai lệch thứ ba giữa thông số lượng tử hóa thứ ba và thông số lượng tử hóa thứ sáu còn được thu nhận bằng cách thu nhận dòng bit. Trong việc tính độ chói thứ nhất, độ chói thứ nhất của một hoặc nhiều điểm ba chiều được tính bằng cách lượng tử hóa ngược độ chói thứ nhất được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ năm được thu nhận từ độ sai lệch thứ hai, một hoặc nhiều điểm ba chiều được bao gồm trong mỗi không gian con được thu nhận bằng cách chia không gian hiện thời bao gồm các điểm ba chiều, độ chói thứ nhất được lượng tử hóa độ chói được thu nhận bằng cách lượng tử hóa độ chói thứ nhất của một hoặc nhiều điểm ba chiều nhờ sử dụng thông số lượng tử hóa thứ hai và thông số lượng tử hóa thứ năm. Trong việc tính của độ màu thứ nhất, độ màu thứ nhất của một hoặc nhiều điểm ba chiều được tính bằng cách lượng tử hóa ngược độ màu thứ nhất được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ ba và thông số lượng tử hóa thứ sáu được thu nhận từ độ sai lệch thứ ba, độ màu thứ nhất được lượng tử hóa độ màu được thu nhận bằng cách lượng tử hóa độ màu thứ nhất của ít nhất một điểm ba chiều nhờ sử dụng thông số lượng tử hóa thứ ba và thông số lượng tử hóa thứ sáu.

Ví dụ, trong việc thu nhận, thông tin nhận dạng chỉ báo rằng việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ năm và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ sáu đã được thực hiện còn được thu nhận bằng cách thu nhận dòng bit. Trong việc tính của độ chói thứ nhất, khi thông tin nhận dạng chỉ báo rằng việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ năm và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ sáu đã được thực hiện, độ chói thứ nhất được lượng tử hóa được xác định là độ

chói được thu nhận bằng cách lượng tử hóa độ chói thứ nhất của một hoặc nhiều điểm ba chiều. Trong việc tính của độ màu thứ nhất, khi thông tin nhận dạng chỉ báo rằng việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ năm và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ sáu đã được thực hiện, độ màu thứ nhất được lượng tử hóa được xác định là độ màu được thu nhận bằng cách lượng tử hóa độ màu thứ nhất của một hoặc nhiều điểm ba chiều.

Với cấu hình như vậy, thiết bị giải mã dữ liệu ba chiều có thể xác định từ thông tin nhận dạng rằng việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ năm và việc lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ sáu đã được thực hiện, sao cho tải xử lý của quy trình giải mã có thể được giảm.

Ví dụ, trong việc thu nhận, độ chói thứ hai được lượng tử hóa, độ màu thứ hai được lượng tử hóa, thông số lượng tử hóa thứ bảy, và độ sai lệch thứ tư giữa thông số lượng tử hóa thứ bảy và thông số lượng tử hóa thứ tám còn được thu nhận bằng cách thu nhận dòng bit. Của độ chói thứ hai và độ màu thứ hai mà chỉ báo màu sắc thứ hai của các điểm ba chiều, thiết bị giải mã dữ liệu ba chiều còn tính độ chói thứ hai bằng cách lượng tử hóa ngược độ chói thứ hai được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ bảy. Thiết bị giải mã dữ liệu ba chiều còn tính độ màu thứ hai bằng cách lượng tử hóa ngược độ màu thứ hai được lượng tử hóa nhờ sử dụng thông số lượng tử hóa thứ tám được thu nhận từ thông số lượng tử hóa thứ bảy và độ sai lệch thứ tư.

Theo cách này, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác màu sắc thứ hai của điểm ba chiều.

Ví dụ, thiết bị giải mã dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Phương án 9

Với phương pháp mã hóa dữ liệu ba chiều, phương pháp giải mã dữ liệu ba chiều, thiết bị mã hóa dữ liệu ba chiều, và thiết bị giải mã dữ liệu ba chiều được mô tả liên quan đến phương án 8, các quy trình xử lý theo phương án này được mô tả dưới đây là cũng khả thi.

Fig.124 là sơ đồ minh họa ví dụ về phương pháp truyền thông số lượng tử hóa theo phương án 9. Phần (a) trên Fig.124 thể hiện ví dụ trong đó trị số tham chiếu đối với trị số QP được thiết đặt đối với mỗi trong số thông tin hình học và thông tin thuộc tính. Fig.124

chủ yếu khác với Fig.105 thể hiện phương án 8 ở chỗ trị số tham chiếu đối với trị số QP được thiết đặt không chỉ đối với thông tin hình học mà còn đối với thông tin thuộc tính. Nghĩa là, trị số QP đối với ít nhất một trong số các đoạn thông tin thuộc tính bao gồm màu sắc thứ nhất, màu sắc thứ hai, và năng suất phản xạ được ấn định như trị số tham chiếu, và các trị số QP đối với thông tin khác thuộc tính là chỉ báo như thông tin sai lệch đối với trị số tham chiếu chung.

Trên Fig.124, Q_{Y1}, mà là trị số QP được sử dụng cho việc mã hóa của độ chói Y1 của màu sắc thứ nhất, được thiết đặt như trị số tham chiếu chung đối với các đoạn thông tin thuộc tính bao gồm màu sắc thứ nhất, màu sắc thứ hai, và năng suất phản xạ. Q_{Y2}, mà là trị số tham chiếu đối với màu sắc thứ hai, được dẫn ra nhờ sử dụng trị số tham chiếu chung Q_{Y1} và $\Delta(Q_{Y2}, Q_{Y1})$, mà là thông tin sai lệch "5." đối với Q_{Y1}. Q_R, mà là trị số tham chiếu đối với năng suất phản xạ, được dẫn ra nhờ sử dụng trị số tham chiếu chung Q_{Y1} và $\Delta(Q_R, Q_{Y1})$, mà là thông tin sai lệch "8." đối với Q_{Y1}. Trong trường hợp này, trị số tham chiếu chung Q_{Y1} được bao gồm trong APS1, mà là APS đối với màu sắc thứ nhất.

Đối với thuộc tính thứ tư, trị số tham chiếu khác nhau so với trị số tham chiếu chung Q_{Y1} có thể được thiết đặt. Thuộc tính thứ năm có thể không có trị số QP. Nghĩa là, thông tin thuộc tính có thể bao gồm cả thông tin thuộc tính mà được lượng tử hóa nhờ sử dụng trị số tham chiếu chung để dẫn ra các trị số QP được sử dụng đối với việc lượng tử hóa của các đoạn thông tin thuộc tính và thông tin thuộc tính mà được lượng tử hóa nhờ sử dụng trị số tham chiếu khác nhau so với trị số tham chiếu chung. Thông tin thuộc tính có thể còn bao gồm thông tin thuộc tính mà được mã hóa mà không có nhờ sử dụng trị số QP.

Lưu ý rằng, mặc dù ví dụ trong đó trị số QP được sử dụng đối với việc lượng tử hóa của thông tin thuộc tính của màu sắc thứ nhất là trị số tham chiếu chung để dẫn ra các trị số QP được sử dụng đối với việc lượng tử hóa của các đoạn thông tin thuộc tính đã được mô tả dựa vào Fig.124, trị số tham chiếu chung có thể được xác định theo các quy tắc được mô tả dưới đây. Ví dụ, khi tất cả thông tin thuộc tính được mô tả trong thông tin điều khiển chẳng hạn như SPS, trị số QP được bao gồm trong thông tin thuộc tính thứ nhất được chỉ báo ở SPS trong số tất cả thông tin thuộc tính có thể được ấn định như trị số tham chiếu chung. Theo cách khác thông tin điều khiển chẳng hạn như SPS có thể chỉ báo thông tin thuộc tính mà được lượng tử hóa nhờ sử dụng trị số QP được ấn định như trị số tham chiếu chung. Theo cách khác trong thông tin điều khiển chẳng hạn như SPS, thông tin thuộc tính mà được lượng tử hóa nhờ sử dụng trị số QP được ấn định như trị số tham chiếu chung có thể được chỉ báo trước tiên trong số các đoạn thông tin thuộc tính. Trong trường hợp bất

kỳ, bằng cách biểu diễn trị số QP được sử dụng đối với việc lượng tử hóa của mỗi trong số các đoạn thông tin thuộc tính như sự kết hợp của trị số tham chiếu và thông tin sai lệch, lượng của dữ liệu được mã hóa có thể được giảm.

Lưu ý rằng các trị số tham chiếu khác nhau Q_{Y1}, Q_{Y2}, và Q_R đối với các đoạn thông tin thuộc tính khác nhau có thể được chỉ báo trong APS, Q_{Y1} có thể được sử dụng như trị số tham chiếu đối với trị số QP đối với màu sắc thứ nhất, Q_{Y2} có thể được sử dụng như trị số tham chiếu đối với trị số QP đối với màu sắc thứ hai, và Q_R có thể được sử dụng như trị số tham chiếu đối với trị số QP đối với năng suất phản xạ. Trong trường hợp đó, Q_{Y2} và Q_R được biểu diễn bởi trị số tuyệt đối, như với Q_{Y1}.

Ví dụ thứ nhất là phương pháp chỉ báo trị số QP đối với các đoạn thông tin thuộc tính khi siêu dữ liệu đối với các đoạn thông tin thuộc tính được mô tả một cách tập hợp trong một APS.

Fig.125 là sơ đồ thể hiện ví dụ thứ nhất về cú pháp của APS và cú pháp của đoạn đầu của thông tin thuộc tính.

Thứ nhất, ví dụ cú pháp của APS sẽ được mô tả.

aps_idx biểu thị số chỉ số của APS. Aps_idx chỉ báo sự tương ứng giữa APS và đoạn đầu của thông tin thuộc tính.

sps_idx biểu thị số chỉ số của SPS trong đó APS tương ứng.

num_of_attribute biểu thị số của các đoạn thông tin thuộc tính. Lưu ý rằng, khi APS được thiết đặt đối với mỗi đoạn của thông tin thuộc tính, trường hoặc vòng lặp của num_of_attribute không cần được bao gồm trong APS.

attribute_type biểu thị loại của thông tin thuộc tính hoặc, nói cách khác, loại của thông tin thuộc tính. Lưu ý rằng, khi loại của thông tin thuộc tính được mô tả ở SPS tương ứng, thông tin mà cho phép tham chiếu tới loại của thông tin thuộc tính được mô tả ở SPS có thể được bao gồm trong APS thay vì attribute_type.

Trên Fig.125, nếu câu được bao quanh bởi đường đứt nét 6701 chỉ báo trị số QP tùy thuộc vào attribute_type. Ví dụ, khi loại của thông tin thuộc tính là màu sắc, trị số QP đối với độ chói (luma) được biểu diễn bởi trị số tuyệt đối được chỉ báo là trị số tham chiếu, và các trị số QP đối với độ màu (sắc độ: Cb, Cr) được chỉ báo như thông tin sai lệch đối với trị số QP đối với độ chói.

Mặt khác, khi loại của thông tin thuộc tính là năng suất phản xạ, trị số QP đối với

năng suất phản xạ được biểu diễn bởi trị số tuyệt đối được chỉ báo. Như ví dụ khác, khi loại của thông tin thuộc tính không có trị số QP, không có trị số QP được chỉ báo.

Khi có hai hoặc nhiều hơn hai đoạn thông tin thuộc tính, trị số tham chiếu (QP_value_Luma hoặc QP_value trong ví dụ này) đối với đoạn của thông tin thuộc tính có thể được chỉ báo bởi độ chênh lệch với trị số tham chiếu đối với đoạn khác của thông tin thuộc tính. Ví dụ, trong vòng lặp của num_of_attribute, trị số tham chiếu đối với thông tin thuộc tính chung có thể được chỉ báo khi i = 0, và trị số độ sai lệch đối với thông tin thuộc tính chung có thể được chỉ báo khi i => 1.

Data_QP_delata_present_flag là cờ mà chỉ báo xem trị số QP đối với mỗi đoạn dữ liệu (lát) là có mặt trong đoạn đầu của thông tin thuộc tính hay không. Khi cờ là 1, trị số QP đối với mỗi đoạn dữ liệu (lát) được chỉ báo trong đoạn đầu của thông tin thuộc tính.

Tiếp theo, ví dụ cú pháp của đoạn đầu của thông tin thuộc tính sẽ được mô tả.

Đoạn đầu của thông tin thuộc tính cũng bao gồm aps_idx. Do đó, sự tương ứng giữa APS và đoạn đầu của thông tin thuộc tính được chỉ báo bởi APS và aps_idx được bao gồm trong đoạn đầu của thông tin thuộc tính. Nghĩa là, thực tế là APS và đoạn đầu của thông tin thuộc tính chia sẻ aps_idx chỉ báo rằng có sự tương ứng giữa APS và đoạn đầu của thông tin thuộc tính.

attribute_type chỉ báo loại của thông tin thuộc tính (loại của thông tin thuộc tính). Lưu ý rằng khi loại của thông tin thuộc tính được mô tả trong APS hoặc SPS tương ứng, thông tin mà cho phép tham chiếu tới loại của thông tin thuộc tính được mô tả trong APS hoặc SPS có thể được bao gồm trong đoạn đầu của thông tin thuộc tính, thay vì attribute_type.

Các trị số QP đối với các trường trong nếu câu được bao quanh bởi đường đứt nét 6702, cụ thể là, QP_delata_data_to_frame, QP_delta1_to_frame, và QP_delta2_to_frame, là các trị số QP đối với dữ liệu tương ứng với attribute_type. Mỗi trị số QP chỉ báo thông tin sai lệch đối với trị số được mô tả trong APS.

Ví dụ thứ hai là phương pháp chỉ báo trị số QP đối với thông tin thuộc tính khi siêu dữ liệu cho một đoạn của thông tin thuộc tính được mô tả độc lập trong một APS. Theo ví dụ thứ hai, các loại khác nhau (các loại) của thông tin thuộc tính có cấu trúc đoạn đầu chung, và do đó, thay đổi của cấu trúc cú pháp với thông tin thuộc tính có thể được tránh khỏi một cách hữu hiệu.

Fig.126 là sơ đồ thể hiện ví dụ thứ hai về cú pháp của APS. Fig.127 là sơ đồ thể hiện ví dụ thứ hai về cú pháp của đoạn đầu của thông tin thuộc tính.

APS bao gồm trị số tham chiếu và trị số độ sai lệch đối với trị số QP của khung. Khi data_QP_delta_present_flag của APS là 1, đoạn đầu của thông tin thuộc tính bao gồm thông tin sai lệch đối với trị số tham chiếu đối với APS.

Ở đây, các trường liên quan đến các trị số QP là luôn có mặt, xem loại của thông tin thuộc tính là màu sắc, năng suất phản xạ, hoặc số khung chẳng hạn hay không. APS có số thứ nhất của các trường để lưu trữ N trị số QP (N là 2 hoặc lớn hơn), mà không quan tâm đến loại của thông tin thuộc tính. Ở đây, N là 3 chẳng hạn.

Khi loại của thông tin thuộc tính là màu sắc, ví dụ, QP_value trong APS lưu trữ thông tin mà chỉ báo trị số QP đối với độ chói, và QP_delta1 và QP_delta2 lưu trữ thông tin mà chỉ báo các trị số QP đối với sắc độ. Ví dụ, QP_value là trị số tham chiếu, và QP_delta1 và QP_delta2 là thông tin sai lệch đối với QP_value. Nghĩa là, trị số QP đối với độ chói được chỉ báo bởi QP_value, và các trị số QP đối với sắc độ được chỉ báo bởi trị số được thu nhận bằng cách bổ sung QP_delta1 đến QP_value và trị số được thu nhận bằng cách bổ sung QP_delta2 đến QP_value. Theo cách này, APS bao gồm trị số tham chiếu đối với thông số lượng tử hóa cho việc lượng tử hóa thông tin thuộc tính tương ứng.

Tương tự, QP_delta_data_to_frame trong đoạn đầu của thông tin thuộc tính lưu trữ thông tin sai lệch về trị số QP đối với độ chói đối với QP_value trong APS tương ứng. QP_delta1_to_frame và QP_delta2_to_frame có thể lưu trữ thông tin sai lệch về các trị số QP đối với sắc độ lần lượt đối với QP_delta1 và QP_delta2 trong APS tương ứng.

Khi loại của thông tin thuộc tính là năng suất phản xạ, ví dụ, QP_value trong APS có thể lưu trữ thông tin mà chỉ báo trị số QP đối với năng suất phản xạ, và QP_delta1 và QP_delta2 có thể lưu trữ thông tin mà luôn chỉ báo 0 hoặc tính không hợp lệ. Tương tự, QP_delta_data_to_frame trong đoạn đầu của thông tin thuộc tính có thể lưu trữ thông tin mà chỉ báo trị số QP đối với năng suất phản xạ, và QP_delta1_to_frame và QP_delta2_to_frame có thể lưu trữ thông tin mà luôn chỉ báo 0 hoặc tính không hợp lệ. Trong trường hợp đó, thiết bị giải mã dữ liệu ba chiều không cần sử dụng để giải mã và có thể bỏ qua thông tin được lưu trữ trong QP_delta1 và QP_delta2 lưu trữ thông tin mà chỉ báo 0 hoặc tính không hợp lệ và QP_delta1_to_frame và QP_delta2_to_frame, mà không quan tâm đến thông tin.

Đối với ví dụ khác, khi loại của thông tin thuộc tính không có trị số QP, tất cả các

trường trong APS có thể lưu trữ thông tin mà chỉ báo 0 hoặc tính không hợp lệ. Trong trường hợp đó, data_AP_delta_present_flag cùng được thiết đặt ở 0. Trong trường hợp đó, thiết bị giải mã dữ liệu ba chiều không cần sử dụng để giải mã và có thể bỏ qua thông tin được lưu trữ trong QP_delta1 và QP_delta2 lưu trữ thông tin mà chỉ báo 0 hoặc tính không hợp lệ và QP_delta1_to_frame và QP_delta2_to_frame, mà không quan tâm đến thông tin. Theo cách này, thiết bị giải mã dữ liệu ba chiều có thể bỏ qua thông số được lưu trữ trong trường cụ thể của các trường trong đoạn đầu của thông tin thuộc tính cụ thể mà tương ứng với loại cụ thể của thuộc tính trong các đoạn đầu của các đoạn thông tin thuộc tính.

Với cấu hình như vậy, các trị số QP đối với các loại khác nhau của thông tin thuộc tính có thể được chỉ báo bởi các sự kết hợp của trị số tham chiếu và thông tin sai lệch trong cấu trúc cú pháp chung, sao cho hiệu quả mã hóa có thể được nâng cao.

Lưu ý rằng khi thông tin thuộc tính tương ứng với một đoạn của thông tin hình học bao gồm hai hoặc nhiều hơn hai đoạn của thông tin màu sắc, cách thức chỉ báo thông tin thuộc tính có thể được thay đổi tùy thuộc vào loại của thông tin thuộc tính. Ví dụ, thông tin màu sắc có thể được chỉ báo bởi thông tin sai lệch và trị số tham chiếu QP chung, và QP trị số tham chiếu đối với năng suất phản xạ có thể là một cách riêng biệt được chỉ báo trong APS.

Sáng chế là không giới hạn ở các phương pháp được mô tả liên quan đến các phương án 8 và 9, và trị số tham chiếu có thể được báo hiệu một cách riêng biệt từ thông tin sai lệch, hoặc thông tin sai lệch có thể được báo hiệu một cách độc lập như trị số tham chiếu. Ví dụ, sự kết hợp của trị số tham chiếu và thông tin sai lệch có thể được thay đổi một cách thích ứng tùy thuộc vào các đặc tính của dữ liệu. Ví dụ, ít nhất một trị số tham chiếu có thể được truyền đối với bộ phận mà cần được giải mã một cách độc lập, và thông tin sai lệch có thể được truyền đối với bộ phận mà không cần được giải mã một cách độc lập. Theo cách này, chức năng có thể được nâng cao, và ở cùng thời điểm lượng mã có thể được giảm.

Theo cách khác lượng của thông tin có thể được tính đối với các sự kết hợp của trị số tham chiếu và thông tin sai lệch, và sự kết hợp của trị số tham chiếu và thông tin sai lệch mà có lượng nhỏ nhất của thông tin có thể được tạo ra và được phân phối dựa vào kết quả tính. Khi thay đổi một cách thích ứng sự kết hợp của trị số tham chiếu và thông tin sai lệch, ý nghĩa (các ngữ nghĩa) của trường mà chỉ báo trị số tham chiếu và trường mà chỉ báo thông tin sai lệch có thể được thay đổi một cách thích ứng. Ví dụ, ý nghĩa của mỗi trường

có thể được thay đổi, chẳng hạn như bằng cách xác định xem việc thiết đặt mỗi trường là không hợp lệ hay không theo các quy tắc nêu trên, hoặc cờ mà chỉ báo để thay đổi ý nghĩa của mỗi trường có thể được bổ sung. Theo cách khác đích tham chiếu đối với trị số tham chiếu có thể được thay đổi một cách thích ứng. Trong trường hợp đó, cờ mà chỉ báo rằng đích tham chiếu đã được thay đổi, hoặc Id hoặc tương tự mà cho phép việc nhận dạng của đích tham chiếu có thể được chỉ báo.

Tiếp theo, dựa vào Fig.128, phương pháp chỉ báo mối tương quan giữa thông tin thuộc tính được mô tả ở SPS, APS, và attribute_header (đoạn đầu của thông tin thuộc tính) nhờ sử dụng attribute_component_id sẽ được mô tả. Fig.128 là sơ đồ thể hiện mối tương quan giữa SPS, APS, và đoạn đầu của thông tin thuộc tính. Lưu ý rằng đích của các mũi tên trên Fig.128 chỉ báo đích tham chiếu.

SPS bao gồm thông tin liên quan đến các loại của các đoạn thông tin thuộc tính. Nghĩa là, SPS có thể tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn của thông tin attribute_type mỗi trong số mà chỉ báo loại khác nhau của thông tin thuộc tính. SPS cũng bao gồm, đối với mỗi loại của thông tin thuộc tính, attribute_component_id mà chỉ báo số mà cho phép việc nhận dạng về loại của thông tin thuộc tính. Lưu ý rằng SPS là ví dụ về thông tin điều khiển. Attribute_type là ví dụ về thông tin loại. Attribute_component_id được bao gồm ở SPS là ví dụ về thông tin nhận dạng thứ nhất mà chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

APS hoặc attribute_header bao gồm attribute_component_id mà tương ứng với attribute_component_id được bao gồm ở SPS. Lưu ý rằng APS là ví dụ về thông tin điều khiển thuộc tính thứ hai. Attribute_header là ví dụ về thông tin điều khiển thuộc tính thứ nhất. Attribute_component_id được bao gồm trong APS là ví dụ về thông tin nhận dạng thứ hai mà chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Thiết bị giải mã dữ liệu ba chiều đề cập đến SPS được chỉ báo bởi sps_idx được bao gồm trong APS hoặc attribute_header. Thiết bị giải mã dữ liệu ba chiều sau đó thu nhận loại của thông tin thuộc tính tương ứng với attribute_component_id được bao gồm trong APS hoặc attribute_header từ SPS được tham chiếu như loại của thông tin thuộc tính trong đó thông tin được bao gồm trong APS hoặc attribute_header tương ứng. Lưu ý rằng một APS tương ứng với một loại của thông tin thuộc tính. Đoạn đầu của một đoạn của thông

tin thuộc tính tương ứng với một loại của thông tin thuộc tính. Mỗi trong số các APS tương ứng với (các) đoạn đầu của một hoặc nhiều đoạn thông tin thuộc tính. Nghĩa là, một APS tương ứng với (các) đoạn đầu của một hoặc nhiều đoạn thông tin thuộc tính khác với (các) đoạn đầu của một hoặc nhiều đoạn thông tin thuộc tính mà tương ứng với APS khác.

Khi attribute_component_id = 0, ví dụ, thiết bị giải mã dữ liệu ba chiều có thể thu nhận thông tin thuộc tính (chẳng hạn như attribute_type) mà tương ứng với attribute_component_id có trị số như nhau, nghĩa là, trị số là 0, từ SPS.

Lưu ý rằng, thay vì attribute_component_id, trình tự của các đoạn thông tin thuộc tính được mô tả ở SPS có thể được mô tả ở SPS. Nghĩa là, thông tin loại mà chỉ báo các loại của thông tin thuộc tính có thể được lưu trữ (được mô tả) ở SPS theo trình tự được định trước. Trong trường hợp đó, attribute_component_id được bao gồm trong APS hoặc attribute_header chỉ báo rằng APS hoặc attribute_header bao gồm attribute_component_id được kết hợp với thông tin loại ở vị trí theo trình tự được định trước.

Theo cách khác trình tự của các APS được truyền hoặc thông tin thuộc tính có thể được thực hiện để đồng nhất với trình tự của thông tin thuộc tính được mô tả ở SPS, nhờ đó cho phép thiết bị giải mã dữ liệu ba chiều dẫn ra trình tự đến của các APS hoặc thông tin thuộc tính và đề cập đến thông tin thuộc tính tương ứng với trình tự đến. Khi dữ liệu đám mây điểm bao gồm cả thông tin thuộc tính mà APS hoặc attribute_header của nó có thể hoặc có thể không có mặt tùy thuộc vào khung và thông tin thuộc tính mà APS hoặc attribute_header của nó là luôn có mặt mà không quan tâm đến khung, thông tin thuộc tính mà APS hoặc attribute_header của nó là luôn có mặt mà không quan tâm đến khung có thể được truyền đầu tiên, và sau đó thông tin thuộc tính mà APS hoặc attribute_header của nó có thể hoặc có thể không có mặt tùy thuộc vào khung có thể được truyền.

Lưu ý rằng, mặc dù các APS mỗi trong số mà tương ứng với các đoạn thông tin thuộc tính được thể hiện trong một khung trên FigS. 124 và 128, một APS có thể được sử dụng, thay vì các APS. Trong trường hợp đó, một APS bao gồm thông tin thông tin thuộc tính liên quan mà tương ứng với các đoạn thông tin thuộc tính.

aps_idx có thể bao gồm số trình tự mà tương ứng với số khung. Sự tương ứng giữa APS và attribute_header có thể được chỉ báo theo cách này. Lưu ý rằng aps_idx có thể có chức năng của attribute_component_id. Điều này cho phép thông tin về toàn bộ trình tự liên quan đến một hoặc nhiều loại của các APS hoặc thông tin thuộc tính được lưu trữ ở SPS và được đề cập đến từ mỗi APS hoặc attribute_header.

Lưu ý rằng để cho phép việc xác định về loại (attribute_type) của thông tin thuộc tính của APS hoặc attribute_header, attribute_type có thể được bao gồm trực tiếp trong APS hoặc attribute_header, hoặc có thể được bao gồm trong đoạn đầu bộ phận NAL như loại của bộ phận NAL.

Trong trường hợp bất kỳ, thông tin thuộc tính của APS hoặc attribute_header có thể được thu nhận, và loại của thuộc tính của thông tin thuộc tính có thể được xác định.

Như được nêu trên, thiết bị mã hóa dữ liệu ba chiều theo phương án này thực hiện quy trình được thể hiện trên Fig.129. Thứ nhất, thiết bị mã hóa dữ liệu ba chiều mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số (S6701). Thiết bị mã hóa dữ liệu ba chiều tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất (S6702). Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau. Hơn nữa, các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì dòng bit bao gồm thông tin nhận dạng thứ nhất dùng để nhận dạng loại của thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ nhất tương ứng được tạo ra, thiết bị giải mã dữ liệu ba chiều đã thu dòng bit có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, các đoạn thông tin loại được lưu trữ trong thông tin điều khiển theo trình tự được định trước. Thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

Với cấu hình như vậy, vì thông tin loại được chỉ báo theo trình tự được định trước mà không có thông tin chỉ báo thông tin loại, lượng dữ liệu của dòng bit có thể được giảm, và lượng của dòng bit được truyền có thể được giảm.

Ví dụ, dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

Với cấu hình như vậy, vì mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số, thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ hai tương ứng có thể được mã hóa nhờ sử dụng trị số tham chiếu. với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều đã thu dòng bit có thể nhận dạng loại của thông tin thứ hai thuộc tính nhờ sử dụng thông tin nhận dạng thứ hai, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số. Với cấu hình như vậy, hiệu quả mã hóa có thể được nâng cao.

Ví dụ, dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì dòng bit bao gồm thông tin nhận dạng thứ hai dùng để nhận dạng loại của thông tin thuộc tính trong đó thông tin điều khiển thuộc tính thứ hai tương ứng được tạo ra, có khả năng tạo ra dòng bit mà có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm N trường trong đó N thông số được lưu trữ, N lớn hơn hoặc bằng 2. Trong thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính thứ nhất, một trong số N trường bao gồm trị số chỉ báo tính không hợp lệ, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

Với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều đã thu dòng bit có thể nhận dạng loại của thông tin thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất và bỏ qua quy trình giải mã trong trường hợp thông tin điều khiển thuộc tính thứ nhất cụ thể, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, trong việc mã hóa, các đoạn thông tin thuộc tính được lượng tử hóa nhờ sử dụng các thông số lượng tử hóa như các thông số.

Với cấu hình như vậy, vì thông số được biểu diễn nhờ sử dụng độ chênh lệch với

trị số tham chiếu, có khả năng nâng cao hiệu quả mã hóa đối với việc lượng tử hóa.

Ví dụ, thiết bị mã hóa dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Thiết bị giải mã dữ liệu ba chiều theo phương án này thực hiện quy trình được thể hiện trên Fig.130. Thứ nhất, thiết bị giải mã dữ liệu ba chiều thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit (S6711). Thiết bị giải mã dữ liệu ba chiều giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng (S6712). Dòng bit bao gồm thông tin điều khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất. Thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau. Các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

Với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều có thể nhận dạng loại của thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, các đoạn thông tin loại được lưu trữ trong thông tin điều khiển theo trình tự được định trước. Thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

Với cấu hình như vậy, vì thông tin loại được chỉ báo theo trình tự được định trước mà không có thông tin chỉ báo thông tin loại, lượng dữ liệu của dòng bit có thể được giảm, và lượng của dòng bit được truyền có thể được giảm.

Ví dụ, dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

Với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều có thể giải mã thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ hai nhờ sử dụng trị số tham

chiếu, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số. Với cấu hình như vậy, vì có khả năng giải mã thông tin thuộc tính nhờ sử dụng trị số tham chiếu và thông tin sai lệch, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính. Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại. Với cấu hình như vậy, vì có khả năng nhận dạng loại của thông tin thuộc tính tương ứng với thông tin điều khiển thuộc tính thứ hai nhờ sử dụng thông tin nhận dạng thứ hai, có khả năng giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm các trường trong đó các thông số được lưu trữ. Trong việc giải mã, thông số được lưu trữ trong trường cụ thể trong số các trường của thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính thứ nhất được bỏ qua, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

Với cấu hình như vậy, vì thiết bị giải mã dữ liệu ba chiều có thể nhận dạng loại của thông tin thuộc tính thứ nhất nhờ sử dụng thông tin nhận dạng thứ nhất, thiết bị giải mã dữ liệu ba chiều có thể giải mã một cách chính xác và hiệu quả thông tin thuộc tính trên điểm ba chiều.

Ví dụ, trong việc giải mã, các đoạn thông tin thuộc tính được mã hóa được lượng tử hóa ngược nhờ sử dụng các thông số lượng tử hóa như các thông số.

Với cấu hình như vậy, có khả năng giải mã một cách chính xác thông tin thuộc tính trên điểm ba chiều.

Ví dụ, thiết bị giải mã dữ liệu ba chiều bao gồm bộ xử lý và bộ nhớ, và bộ xử lý thực hiện quy trình nêu trên nhờ sử dụng bộ nhớ.

Thiết bị mã hóa dữ liệu ba chiều, thiết bị giải mã dữ liệu ba chiều, và tương tự theo các phương án của sáng chế đã được nêu trên, nhưng sáng chế là không giới hạn ở các

phương án này.

Lưu ý rằng mỗi trong số các bộ xử lý được bao gồm trong thiết bị mã hóa dữ liệu ba chiều, thiết bị giải mã dữ liệu ba chiều, và tương tự theo các phương án nêu trên được thực hiện cụ thể như mạch tích hợp cỡ lớn (large-scale integrated, viết tắt là LSI), mà là mạch tích hợp (integrated circuit, viết tắt là IC). Chúng có thể có dạng các chip riêng biệt, hoặc có thể được bao gói một phần hoặc toàn bộ trong chip đơn.

IC như vậy là không giới hạn ở LSI, và do đó có thể được thực hiện như mạch chuyên dụng hoặc bộ xử lý đa năng. Theo cách khác mảng cổng lập trình được dạng (field programmable gate array, viết tắt là FPGA) mà cho phép lập trình sau khi sản xuất LSI, hoặc bộ xử lý có thể tái cấu hình được mà cho phép cấu hình lại sự kết nối và việc thiết đặt của các ô mạch bên trong LSI có thể được ứng dụng.

Hơn nữa, theo các phương án nêu trên, các thành phần cấu trúc có thể được thực hiện dưới dạng phần cứng chuyên dụng hoặc có thể được thực hiện bằng cách thực hiện chương trình phần mềm phù hợp với các thành phần cấu trúc như vậy. Theo cách khác các thành phần cấu trúc có thể được thực hiện bởi bộ thực hiện chương trình chẳng hạn như CPU hoặc bộ xử lý đọc ra và thực hiện chương trình phần mềm được ghi trong phương tiện ghi chẳng hạn như đĩa cứng hoặc bộ nhớ bán dẫn.

Sáng chế có thể cũng được thực hiện dưới dạng phương pháp mã hóa dữ liệu ba chiều, phương pháp giải mã dữ liệu ba chiều, hoặc tương tự được thực hiện bởi thiết bị mã hóa dữ liệu ba chiều, thiết bị giải mã dữ liệu ba chiều, và tương tự.

Ngoài ra, các sự phân chia của các khối chức năng được thể hiện trong các sơ đồ khối chỉ là các ví dụ, và do đó các khối chức năng có thể được thực hiện như khối chức năng đơn, hoặc khối chức năng đơn có thể được chia thành các khối chức năng, hoặc một hoặc nhiều chức năng có thể được di chuyển tới khối chức năng khác. Ngoài ra, các chức năng của các khối chức năng có các chức năng tương tự có thể được xử lý bởi phần cứng hoặc phần mềm đơn lẻ theo cách thức song song hóa hoặc được chia theo thời gian.

Ngoài ra, thứ tự xử lý của việc thực hiện các bước được thể hiện trong các lưu đồ chỉ là minh họa để mô tả một cách cụ thể sáng chế, và do vậy có thể có thứ tự khác với thứ tự được thể hiện. Ngoài ra, một hoặc nhiều của các bước có thể được thực hiện đồng thời (song song) với bước khác.

Thiết bị mã hóa dữ liệu ba chiều, thiết bị giải mã dữ liệu ba chiều, và tương tự theo

một hoặc nhiều khía cạnh đã được nêu trên dựa vào các phương án, nhưng sáng chế là không giới hạn ở các phương án này. Một hoặc nhiều khía cạnh do đó có thể bao gồm các dạng nhận được bằng cách thực hiện các sự sửa đổi khác nhau tới các phương án nêu trên mà có thể được nắm rõ bởi người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng, cũng như các dạng nhận được bằng cách kết hợp các thành phần cấu trúc theo các phương án khác nhau, mà không trệch khỏi tinh thần của sáng chế.

Khả năng áp dụng công nghiệp

Sáng chế có khả năng áp dụng được tới thiết bị mã hóa dữ liệu ba chiều và thiết bị giải mã dữ liệu ba chiều.

Các ký hiệu tham chiếu trên các hình vẽ

4601 hệ thống mã hóa dữ liệu ba chiều

4602 hệ thống giải mã dữ liệu ba chiều

4603 đầu cuối cảm biến

4604 bộ nối bên ngoài

4611 hệ thống tạo dữ liệu đám mây điểm

4612 bộ trình diễn

4613 bộ mã hóa

4614 bộ đa hợp

4615 bộ phận đầu vào/đầu ra

4616 bộ điều khiển

4617 bộ thu nhận thông tin cảm biến

4618 bộ tạo dữ liệu đám mây điểm

4621 bộ thu nhận thông tin cảm biến

4622 bộ phận đầu vào/đầu ra

4623 bộ giải đa hợp

4624 bộ giải mã

4625 bộ trình diễn

4626 giao diện người dùng

4627 bộ điều khiển

4630 bộ mã hóa thứ nhất

4631 bộ mã hóa thông tin hình học

4632 bộ mã hóa thông tin thuộc tính

4633 bộ mã hóa thông tin bổ sung

4634 bộ đa hợp

4640 bộ giải mã thứ nhất

4641 bộ giải đa hợp

4642 bộ giải mã thông tin hình học

4643 bộ giải mã thông tin thuộc tính

4644 bộ giải mã thông tin bổ sung

4650 bộ mã hóa thứ hai

4651 bộ tạo thông tin bổ sung

4652 bộ tạo hình ảnh hình học

4653 bộ tạo thông tin thuộc tính

4654 bộ mã hóa viđeo

4655 bộ mã hóa thông tin bổ sung

4656 bộ đa hợp

4660 bộ giải mã thứ hai

4661 bộ giải đa hợp

4662 bộ giải mã viđeo

4663 bộ giải mã thông tin bổ sung

4664 bộ tạo thông tin hình học

4665 bộ tạo thông tin bổ sung

4670 bộ mã hóa

4671 bộ đa hợp

4680 bộ giải mã

4681 bộ giải đa hợp

4710 bộ đa hợp thứ nhất

4711 bộ chuyển đổi tệp

4720 bộ giải đa hợp thứ nhất

4721 bộ chuyển đổi ngược tệp

4730 bộ đa hợp thứ hai

4731 bộ chuyển đổi tệp

4740 bộ giải đa hợp thứ hai

4741 bộ chuyển đổi ngược tệp

4750 bộ đa hợp thứ ba

4751 bộ chuyển đổi tệp

4760 bộ giải đa hợp thứ ba

4761 bộ chuyển đổi ngược tệp

4801 bộ mã hóa

4802 bộ đa hợp

4911 bộ chia

4931 bộ chia lát

4932 bộ chia mảnh thông tin hình học

4933 bộ chia mảnh thuộc tính thông tin

5010 bộ mã hóa thứ nhất

5011 bộ chia

5012 bộ mã hóa thông tin hình học

5013 bộ mã hóa thông tin thuộc tính

5014 bộ giải mã thông tin bổ sung

5015 bộ đa hợp

5020 bộ giải mã thứ nhất

5021 bộ giải đa hợp

5022 bộ giải mã thông tin hình học

5023 bộ giải mã thông tin thuộc tính

5024 bộ giải mã thông tin bổ sung

5025 bộ kết hợp

5031 bộ chia mảnh

5032 bộ chia lát thông tin hình học

5033 bộ chia lát thông tin thuộc tính

5041 bộ kết hợp lát thông tin hình học

5042 bộ kết hợp thông tin thuộc tính lát

5043 bộ kết hợp mảnh

5051 bộ chia mảnh

5052 bộ mã hóa

5053 bộ giải mã

5054 bộ kết hợp mảnh

5200 bộ mã hóa thứ nhất

5201 bộ chia

5202 bộ mã hóa thông tin hình học

5203 bộ mã hóa thông tin thuộc tính

5204 bộ mã hóa thông tin bổ sung

5205 bộ đa hợp

5211 bộ chia mảnh

5212 bộ chia lát

5221, 5231, 5251, 5261 bộ khởi tạo CABAC

5222, 5232 bộ mã hóa entrôpi

5240 bộ giải mã thứ nhất

5241 bộ giải đa hợp

5242 bộ giải mã thông tin hình học

5243 bộ giải mã thông tin thuộc tính

5244 bộ giải mã thông tin bổ sung

5245 bộ kết hợp

5252, 5262 bộ giải mã entrôpi

5300 bộ mã hóa thứ nhất

5301 bộ chia

5302 bộ mã hóa thông tin hình học

5303 bộ mã hóa thông tin thuộc tính

5304 bộ mã hóa thông tin bổ sung

5305 bộ đa hợp

5311 bộ chia mảnh

5312 bộ chia lát

5321, 5331, 5351, 5361 bộ tính trị số lượng tử hóa

5322, 5332 bộ mã hóa entrôpi

5323 bộ mã hóa

5333 bộ lượng tử hóa ngược

5340 bộ giải mã thứ nhất

5341 bộ giải đa hợp

5342 bộ giải mã thông tin hình học

5343 bộ giải mã thông tin thuộc tính

5344 bộ giải mã thông tin bổ sung

5345 bộ kết hợp

5352, 5362 bộ giải mã entrôpi

YÊU CẦU BẢO HỘ

1. Phương pháp mã hóa dữ liệu ba chiều, trong đó phương pháp này bao gồm các bước:

mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số; và

tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất,

trong đó thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau,

các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

2. Phương pháp mã hóa dữ liệu ba chiều theo điểm 1,

trong đó các đoạn thông tin loại được lưu trữ trong thông tin điều khiển theo trình tự được định trước, và

thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

3. Phương pháp mã hóa dữ liệu ba chiều theo điểm 1 hoặc điểm 2,

trong đó dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

4. Phương pháp mã hóa dữ liệu ba chiều theo điểm 3,

trong đó mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số.

5. Phương pháp mã hóa dữ liệu ba chiều theo điểm 1 hoặc điểm 2,

trong đó dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại.

6. Phương pháp mã hóa dữ liệu ba chiều theo điểm bất kỳ trong số các điểm từ 1 đến 5,

trong đó mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm N trường trong đó N thông số được lưu trữ, N lớn hơn hoặc bằng 2, và

trong thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính thứ nhất, một trong số N trường bao gồm trị số chỉ báo tính không hợp lệ, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

7. Phương pháp mã hóa dữ liệu ba chiều theo điểm bất kỳ trong số các điểm từ 1 đến 6,

trong đó trong việc mã hóa, các đoạn thông tin thuộc tính được lượng tử hóa nhờ sử dụng các thông số lượng tử hóa như các thông số.

8. Phương pháp giải mã dữ liệu ba chiều, phương pháp này bao gồm các bước:

thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit; và

giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng,

trong đó dòng bit bao gồm thông tin điều khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất,

thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau,

các đoạn thông tin điều khiển thuộc tính thứ nhất tương ứng một-một với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất được kết hợp với một trong số các đoạn thông tin loại.

9. Phương pháp giải mã dữ liệu ba chiều theo điểm 8,

trong đó các đoạn thông tin loại được lưu trữ trong thông tin điều khiển theo trình tự

được định trước, và

thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất được kết hợp với một trong số các đoạn thông tin loại mà có thứ tự theo trình tự được định trước.

10. Phương pháp giải mã dữ liệu ba chiều theo điểm 8 hoặc điểm 9,

trong đó dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm trị số tham chiếu của thông số được sử dụng cho việc mã hóa đoạn thông tin thuộc tính tương ứng trong số các đoạn thông tin thuộc tính.

11. Phương pháp giải mã dữ liệu ba chiều theo điểm 10,

trong đó mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin sai lệch mà sai lệch với trị số tham chiếu của thông số.

12. Phương pháp giải mã dữ liệu ba chiều theo điểm 8 hoặc điểm 9,

trong đó dòng bit còn bao gồm các đoạn thông tin điều khiển thuộc tính thứ hai tương ứng với các đoạn thông tin thuộc tính, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ hai bao gồm thông tin nhận dạng thứ hai chỉ báo rằng thông tin điều khiển thuộc tính thứ hai được kết hợp với một trong số các đoạn thông tin loại.

13. Phương pháp giải mã dữ liệu ba chiều theo điểm bất kỳ trong số các điểm từ 8 đến 12,

trong đó mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm các trường trong đó các thông số được lưu trữ, và

trong việc giải mã, thông số được lưu trữ trong trường cụ thể trong số các trường của thông tin điều khiển thuộc tính thứ nhất cụ thể trong số các đoạn thông tin điều khiển thuộc tính thứ nhất được bỏ qua, thông tin điều khiển thuộc tính thứ nhất cụ thể tương ứng với loại thuộc tính cụ thể.

14. Phương pháp giải mã dữ liệu ba chiều theo điểm bất kỳ trong số các điểm từ 8 đến 13,

trong đó trong việc giải mã, các đoạn thông tin thuộc tính được mã hóa được lượng tử hóa ngược nhờ sử dụng các thông số lượng tử hóa như các thông số.

15. Thiết bị mã hóa dữ liệu ba chiều, trong đó thiết bị này bao gồm:

bộ xử lý; và

bộ nhớ,

trong đó nhờ sử dụng bộ nhớ, bộ xử lý:

mã hóa các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng, nhờ sử dụng các thông số; và

tạo ra dòng bit bao gồm các đoạn thông tin thuộc tính được mã hóa, thông tin điều khiển, và các đoạn thông tin điều khiển thuộc tính thứ nhất,

thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất tương ứng với đoạn thông tin thuộc tính khác trong số các đoạn thông tin thuộc tính và được kết hợp với một trong số các đoạn thông tin loại.

16. Thiết bị giải mã dữ liệu ba chiều, trong đó thiết bị này bao gồm:

bộ xử lý; và

bộ nhớ,

trong đó nhờ sử dụng bộ nhớ, bộ xử lý:

thu nhận các đoạn thông tin thuộc tính được mã hóa và các thông số từ dòng bit; và

giải mã các đoạn thông tin thuộc tính được mã hóa nhờ sử dụng các thông số, để tạo ra các đoạn thông tin thuộc tính của các điểm ba chiều tương ứng,

dòng bit bao gồm thông tin điều khiển và các đoạn thông tin điều khiển thuộc tính thứ nhất,

thông tin điều khiển tương ứng với các đoạn thông tin thuộc tính và bao gồm các đoạn thông tin loại đều chỉ báo loại thông tin thuộc tính khác nhau, và

mỗi trong số các đoạn thông tin điều khiển thuộc tính thứ nhất bao gồm thông tin nhận dạng thứ nhất chỉ báo rằng thông tin điều khiển thuộc tính thứ nhất tương ứng với đoạn thông tin thuộc tính khác trong số các đoạn thông tin thuộc tính và được kết hợp với một trong số các đoạn thông tin loại.

FIG. 1

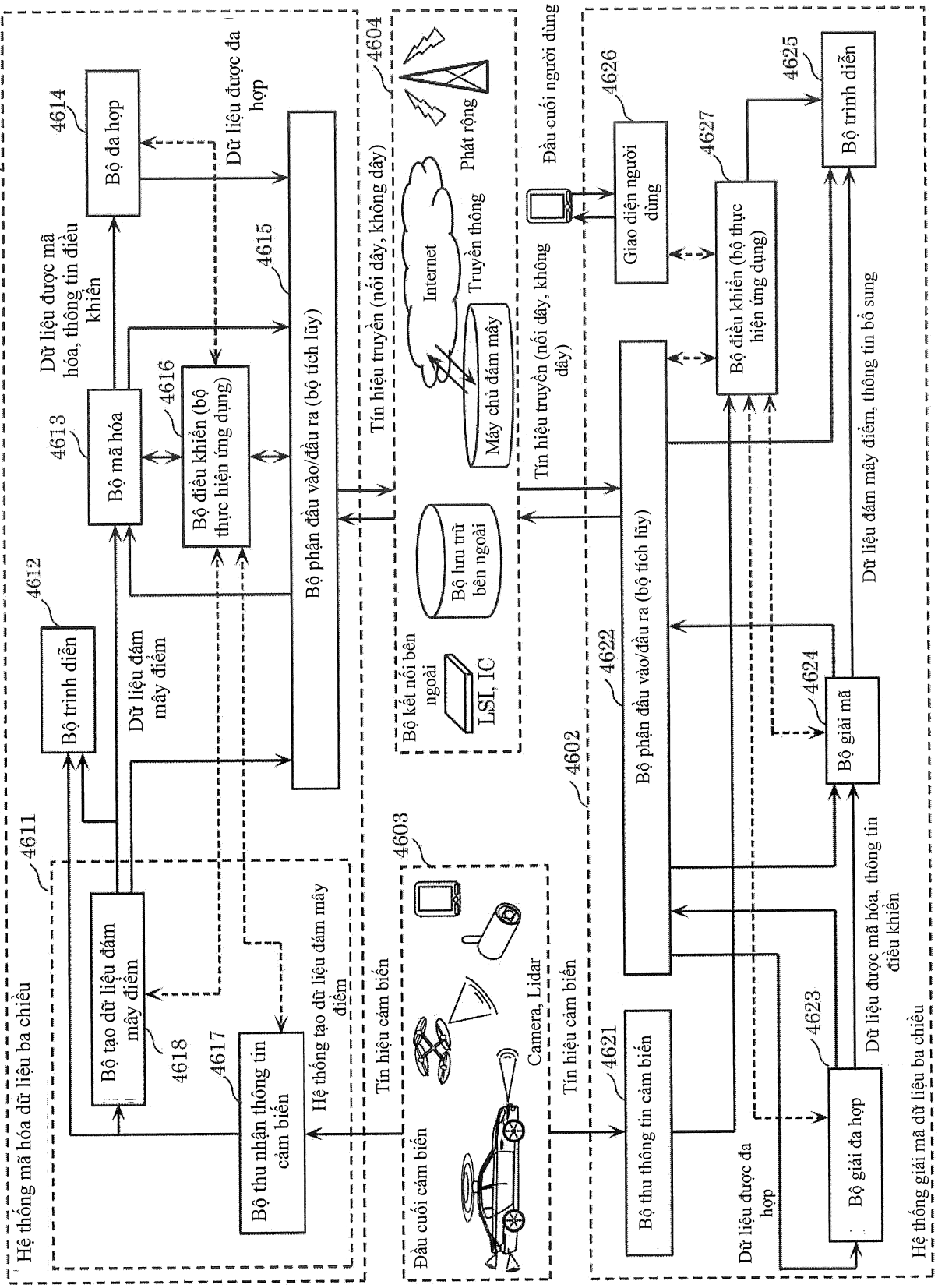

4601
Hệ thống mã hóa dữ liệu ba chiều
4611
Bộ tạo dữ liệu đám mây điểm
4618
4617
Bộ thu nhận thông tin cảm biến
Hệ thống tạo dữ liệu đám mây điểm
Tín hiệu cảm biến
4612
Bộ trình diễn
Dữ liệu đám mây điểm
4613
Bộ mã hóa
4616
Bộ điều khiển (bộ thực hiện ứng dụng)
Dữ liệu được mã hóa, thông tin điều khiển
4614
Bộ đa hợp
Dữ liệu được đa hợp
4615
Bộ phận đầu vào/đầu ra (bộ tích lũy)
4604
Tín hiệu truyền (nối dây, không dây)
Bộ kết nối bên ngoài
LSI, IC
Bộ lưu trữ bên ngoài
Internet
Truyền thông
Phát rộng
Máy chủ đám mây
4603
Đầu cuối cảm biến
Camera, Lidar
Tín hiệu cảm biến
4602
Tín hiệu truyền (nối dây, không dây)
Đầu cuối người dùng
4626
Giao diện người dùng
4627
Bộ điều khiển (bộ thực hiện ứng dụng)
4625
Bộ trình diễn
Bộ phận đầu vào/đầu ra (bộ tích lũy)
4622
4624
Bộ giải mã
Dữ liệu đám mây điểm, thông tin bổ sung
4621
Bộ thu thông tin cảm biến
4623
Bộ giải đa hợp
Dữ liệu được đa hợp
Dữ liệu được mã hóa, thông tin điều khiển
Hệ thống giải mã dữ liệu ba chiều

FIG. 2

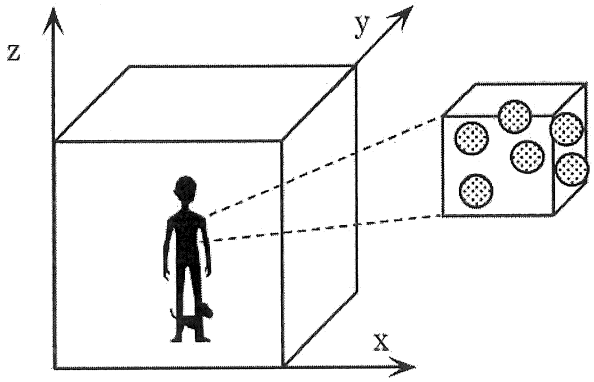


FIG. 3

	Thông tin địa lý	Thông tin thuộc tính
n = 1	$G(1) = (x_1, y_1, z_1)$	$A(1) = (R_1, G_1, B_1)$
n = 2	$G(2) = (x_2, y_2, z_2)$	$A(2) = (R_2, G_2, B_2)$
⋮	⋮	⋮
n = N	$G(N) = (x_N, y_N, z_N)$	$A(N) = (R_N, G_N, B_N)$

FIG. 4

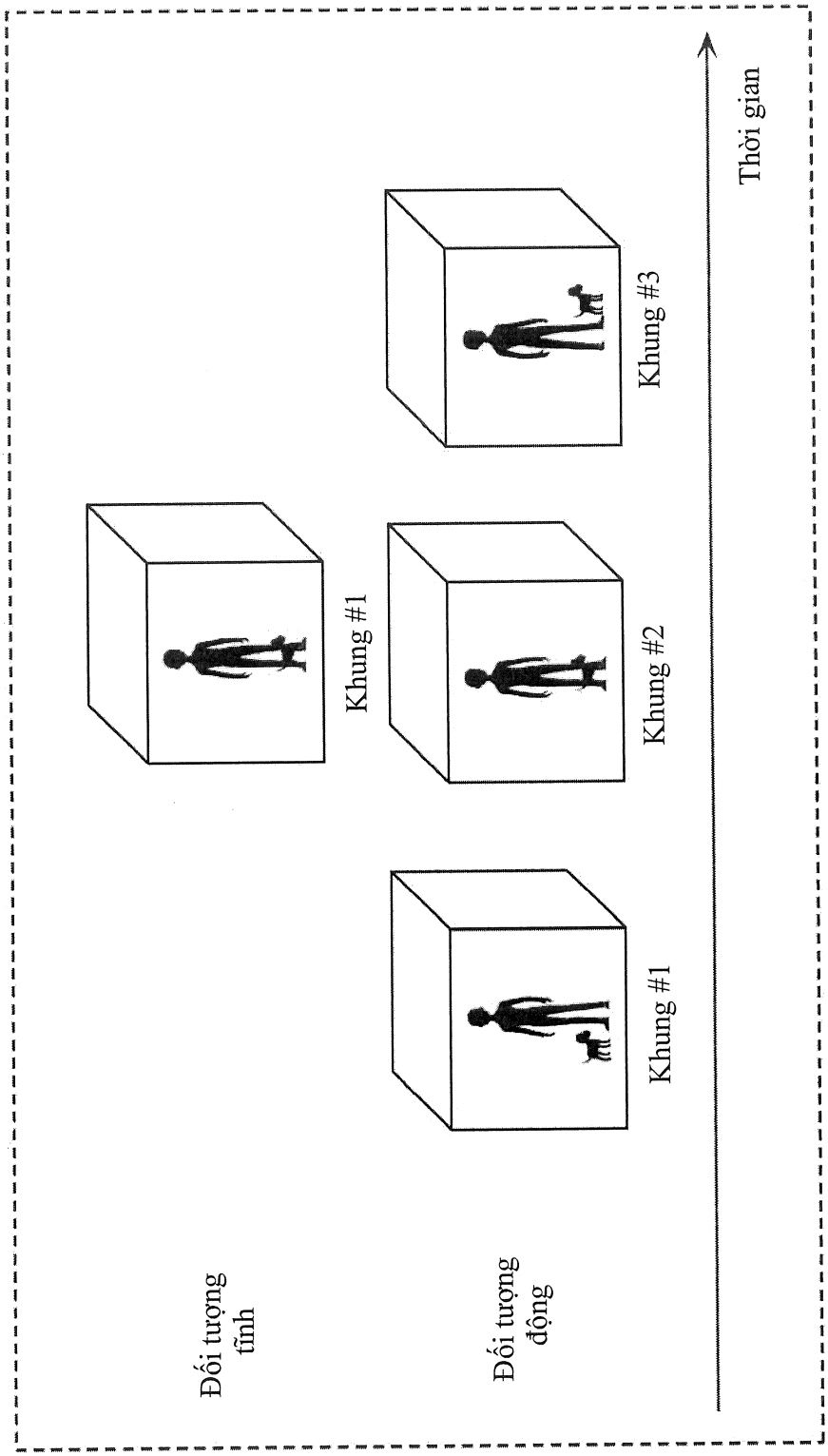

Đối tượng tĩnh
Khung #1
Đối tượng động
Khung #1
Khung #2
Khung #3
Thời gian

FIG. 5

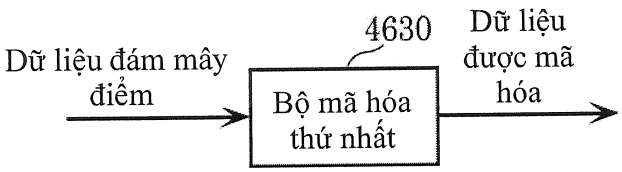


FIG. 6

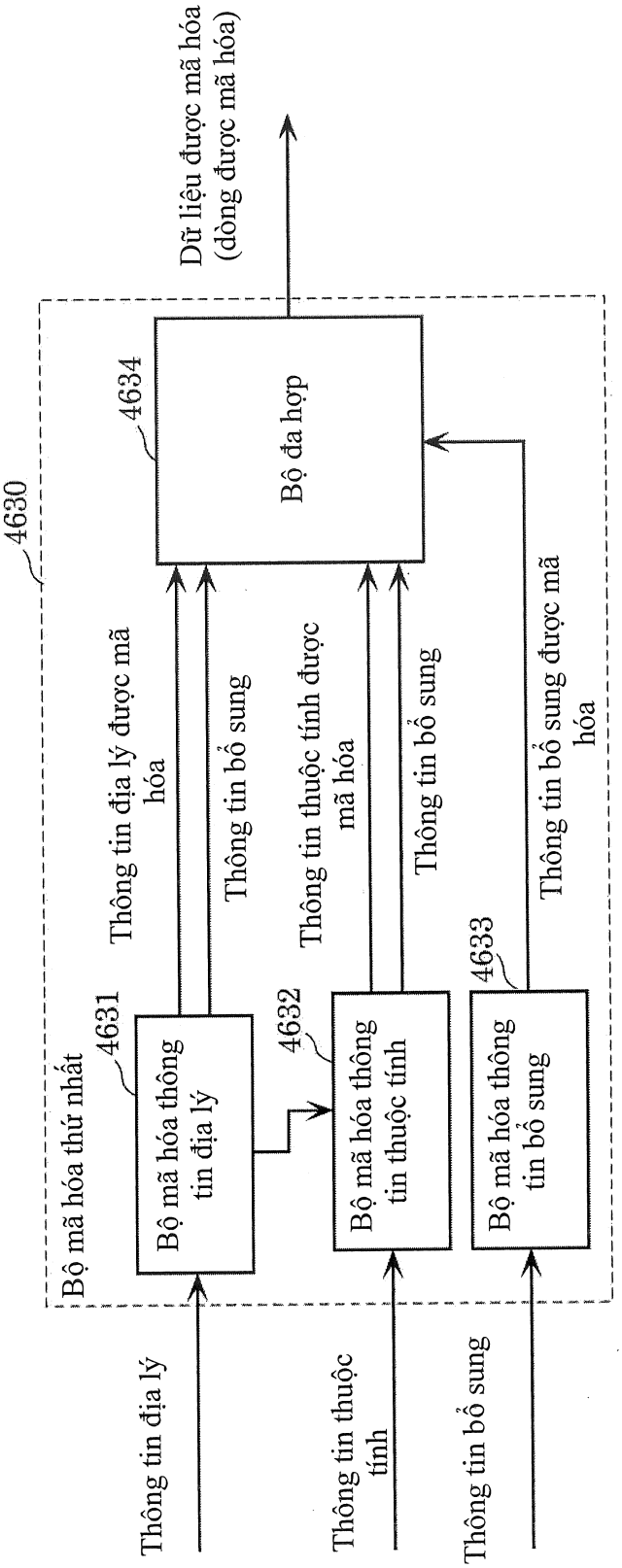

Dữ liệu được mã hóa (dòng được mã hóa)
4630
Bộ mã hóa thứ nhất
4634
Bộ đa hợp
4631
Bộ mã hóa thông tin địa lý
4632
Bộ mã hóa thông tin thuộc tính
4633
Bộ mã hóa thông tin bổ sung
Thông tin địa lý được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
Thông tin địa lý
Thông tin thuộc tính
Thông tin bổ sung

FIG. 7

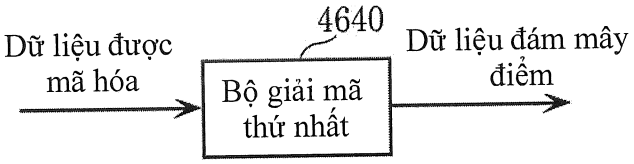

4640
Dữ liệu được mã hóa
Bộ giải mã thứ nhất
Dữ liệu đám mây điểm

FIG. 8

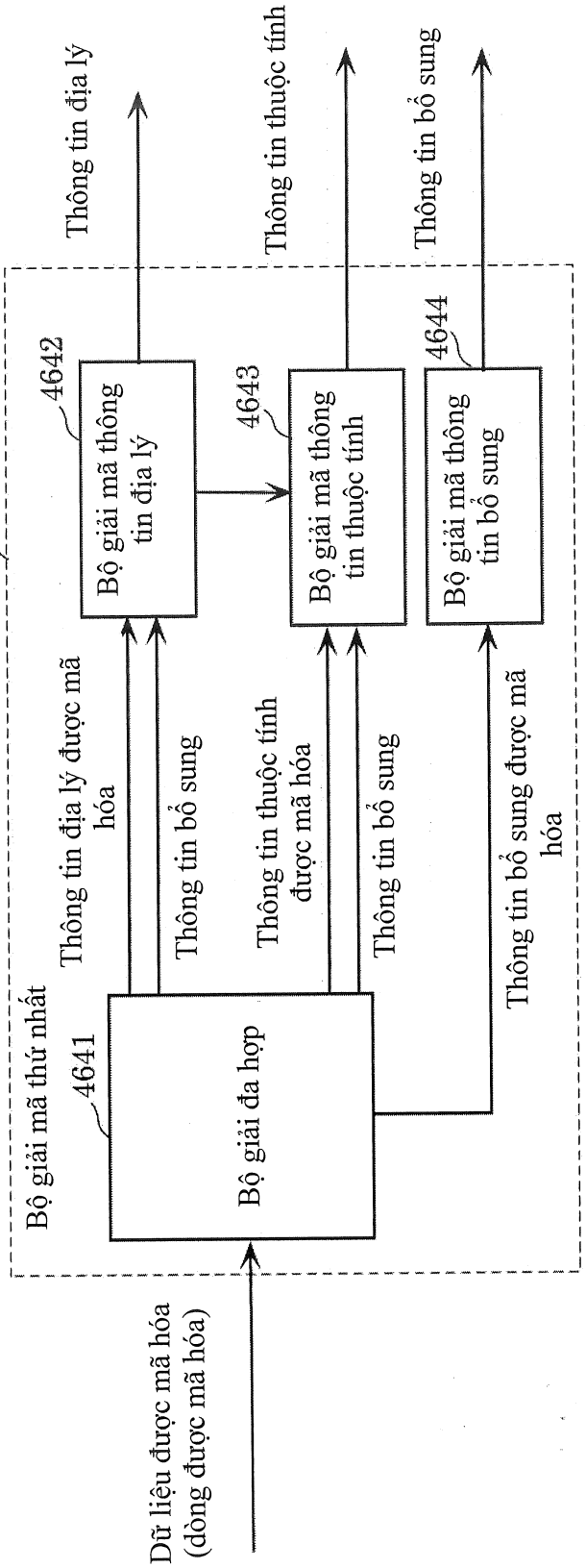

Dữ liệu được mã hóa (dòng được mã hóa)
4640
Bộ giải mã thứ nhất
4641
Bộ giải đa hợp
Thông tin địa lý được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
4642
Bộ giải mã thông tin địa lý
4643
Bộ giải mã thông tin thuộc tính
4644
Bộ giải mã thông tin bổ sung
Thông tin địa lý
Thông tin thuộc tính
Thông tin bổ sung

FIG. 9

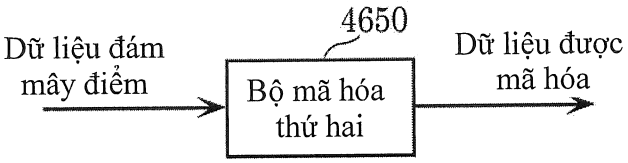


FIG. 10

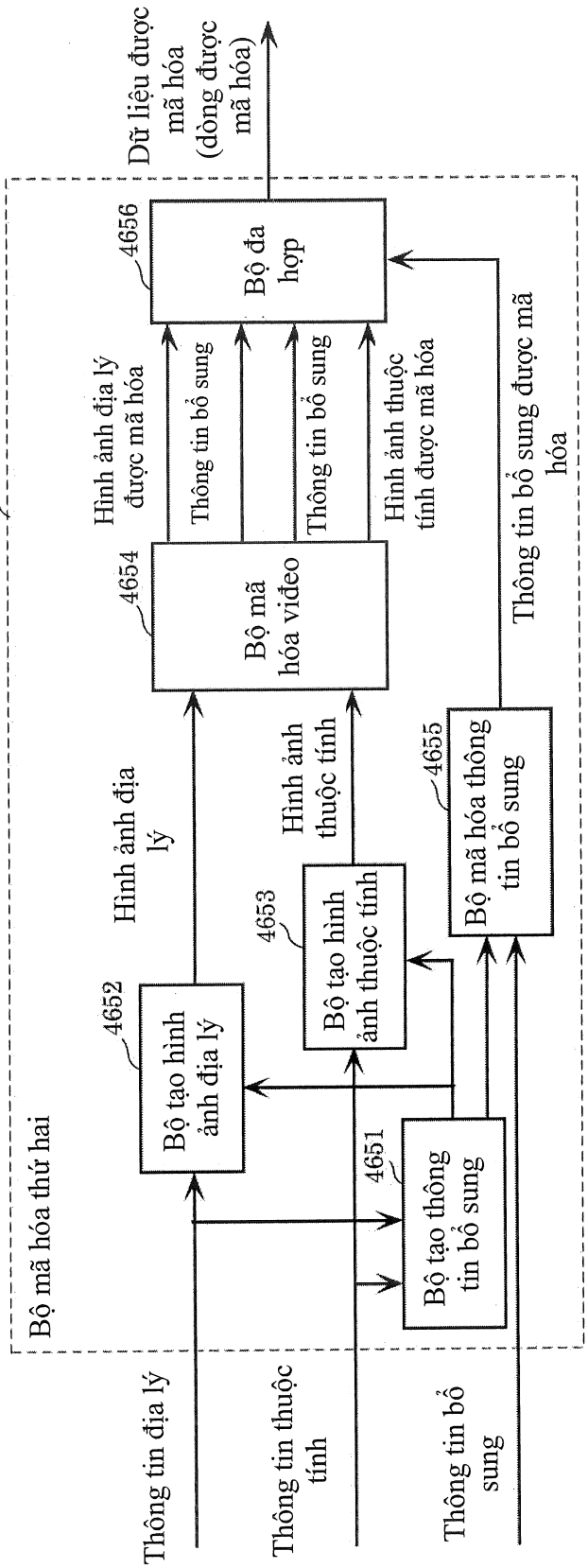

Thông tin địa lý
Thông tin thuộc tính
Thông tin bổ sung
Bộ mã hóa thứ hai
4650
4651
Bộ tạo thông tin bổ sung
4652
Bộ tạo hình ảnh địa lý
4653
Bộ tạo hình ảnh thuộc tính
4655
Bộ mã hóa thông tin bổ sung
4654
Bộ mã hóa video
4656
Bộ đa hợp
Hình ảnh địa lý
Hình ảnh thuộc tính
Thông tin bổ sung được mã hóa
Hình ảnh địa lý được mã hóa
Thông tin bổ sung
Thông tin bổ sung
Hình ảnh thuộc tính được mã hóa
Dữ liệu được mã hóa (dòng được mã hóa)

FIG. 11

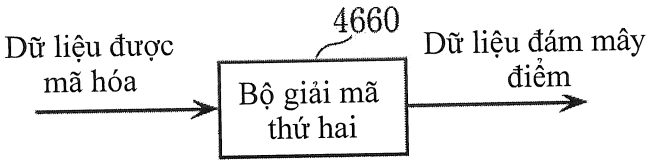


FIG. 12

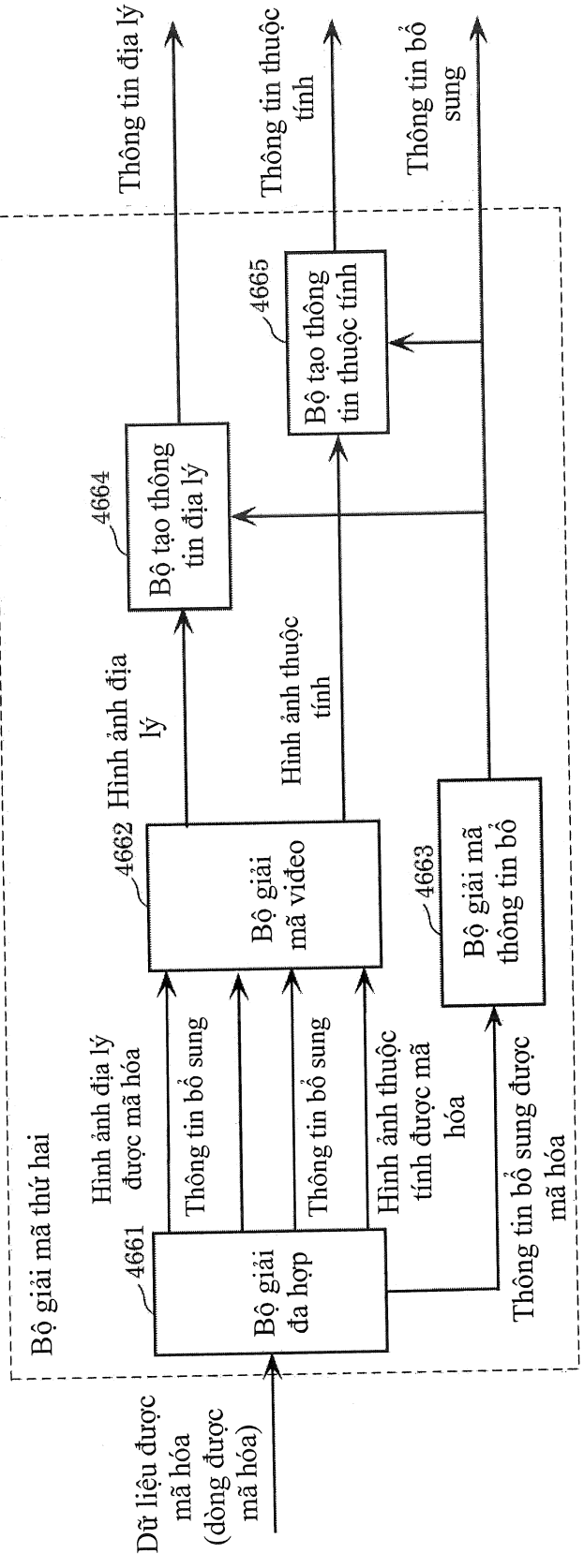

Dữ liệu được mã hóa (dòng được mã hóa)
4660
Bộ giải mã thứ hai
4661
Bộ giải đa hợp
Hình ảnh địa lý được mã hóa
Thông tin bổ sung
Thông tin bổ sung
Hình ảnh thuộc tính được mã hóa
Thông tin bổ sung được mã hóa
4662
Bộ giải mã video
Hình ảnh địa lý
Hình ảnh thuộc tính
4663
Bộ giải mã thông tin bổ
4664
Bộ tạo thông tin địa lý
4665
Bộ tạo thông tin thuộc tính
Thông tin địa lý
Thông tin thuộc tính
Thông tin bổ sung

FIG. 13

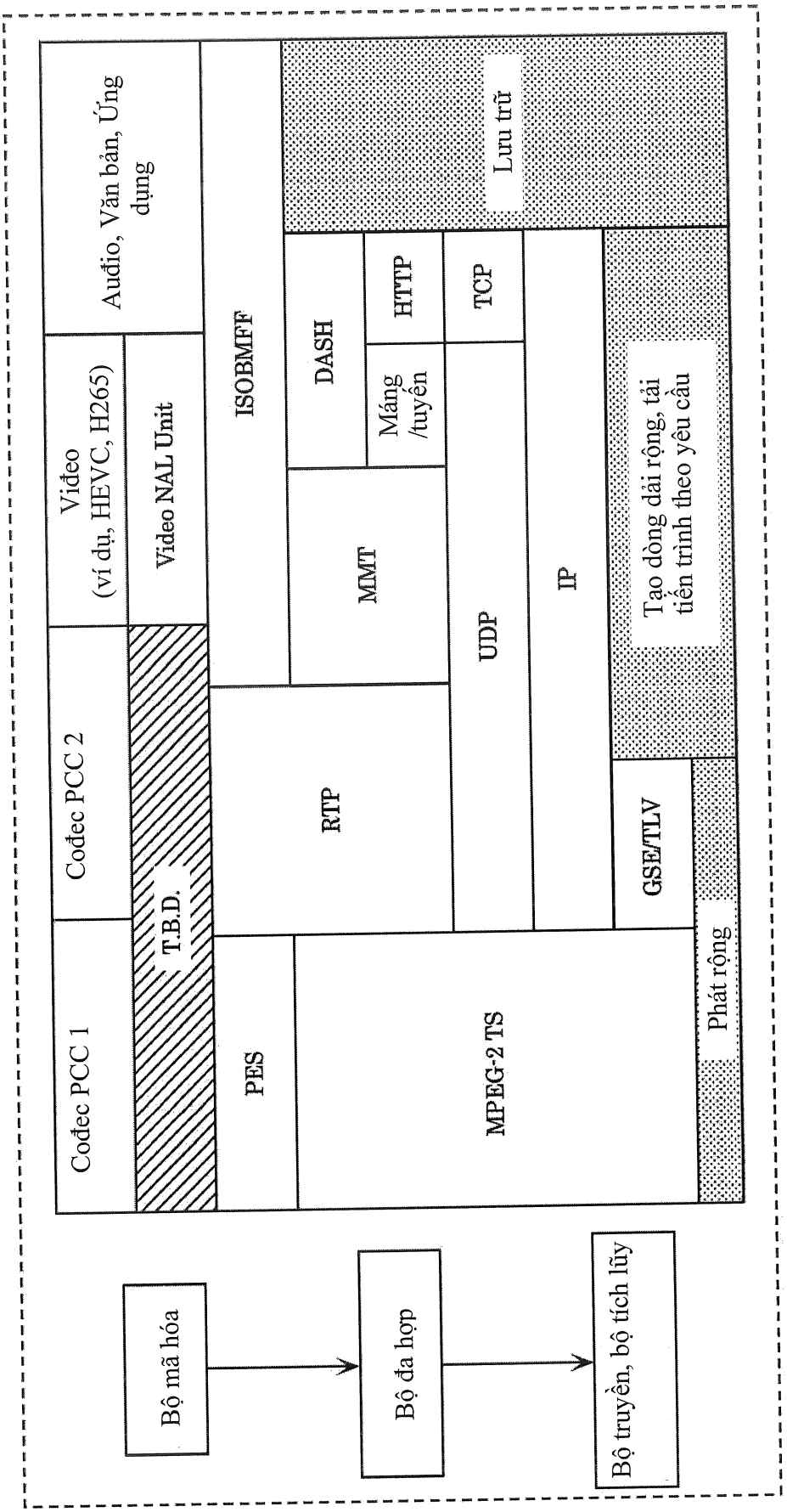

Bộ mã hóa
Bộ đa hợp
Bộ truyền, bộ tích lũy
Codec PCC 1
Codec PCC 2
Video (ví dụ, HEVC, H265)
Audio, Văn bản, Ứng dụng
T.B.D.
Video NAL Unit
PES
MPEG-2 TS
ISOBMFF
RTP
MMT
DASH
Máng /tuyến
UDP
HTTP
TCP
IP
GSE/TLV
Phát rộng
Tạo dòng dải rộng, tải tiến trình theo yêu cầu
Lưu trữ

FIG. 14

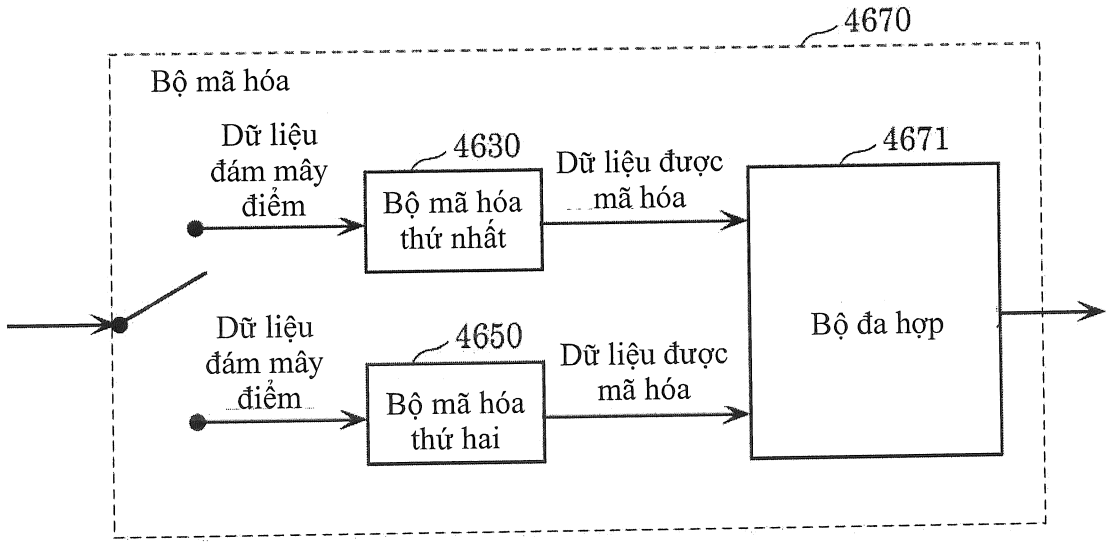

4670
Bộ mã hóa
Dữ liệu đám mây điểm
4630
Bộ mã hóa thứ nhất
Dữ liệu được mã hóa
4671
Bộ đa hợp
Dữ liệu đám mây điểm
4650
Bộ mã hóa thứ hai
Dữ liệu được mã hóa

FIG. 15

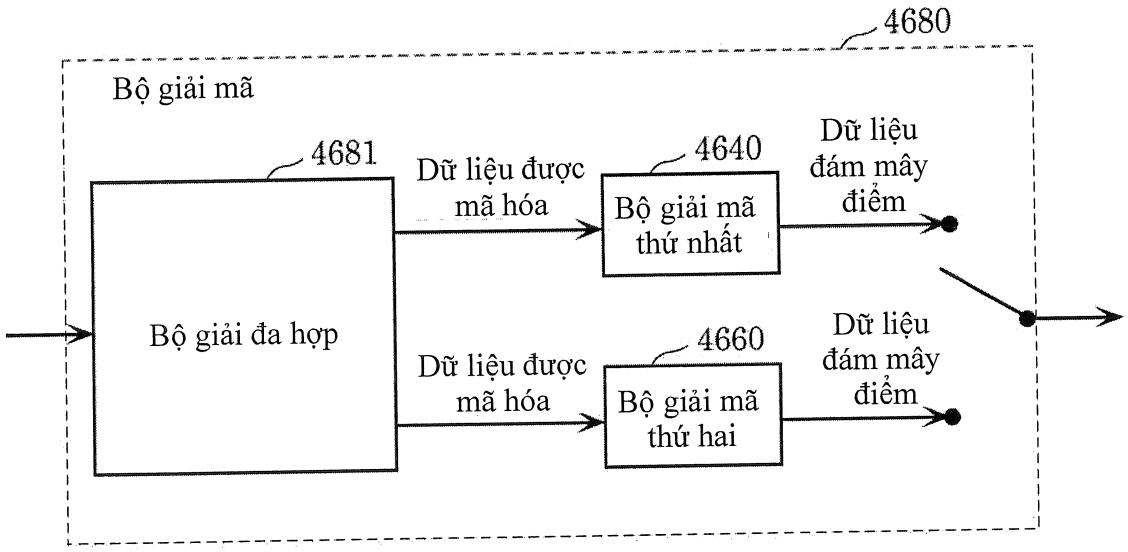

4680
Bộ giải mã
4681
Bộ giải đa hợp
Dữ liệu được mã hóa
4640
Bộ giải mã thứ nhất
Dữ liệu đám mây điểm
Dữ liệu được mã hóa
4660
Bộ giải mã thứ hai
Dữ liệu đám mây điểm

FIG. 16

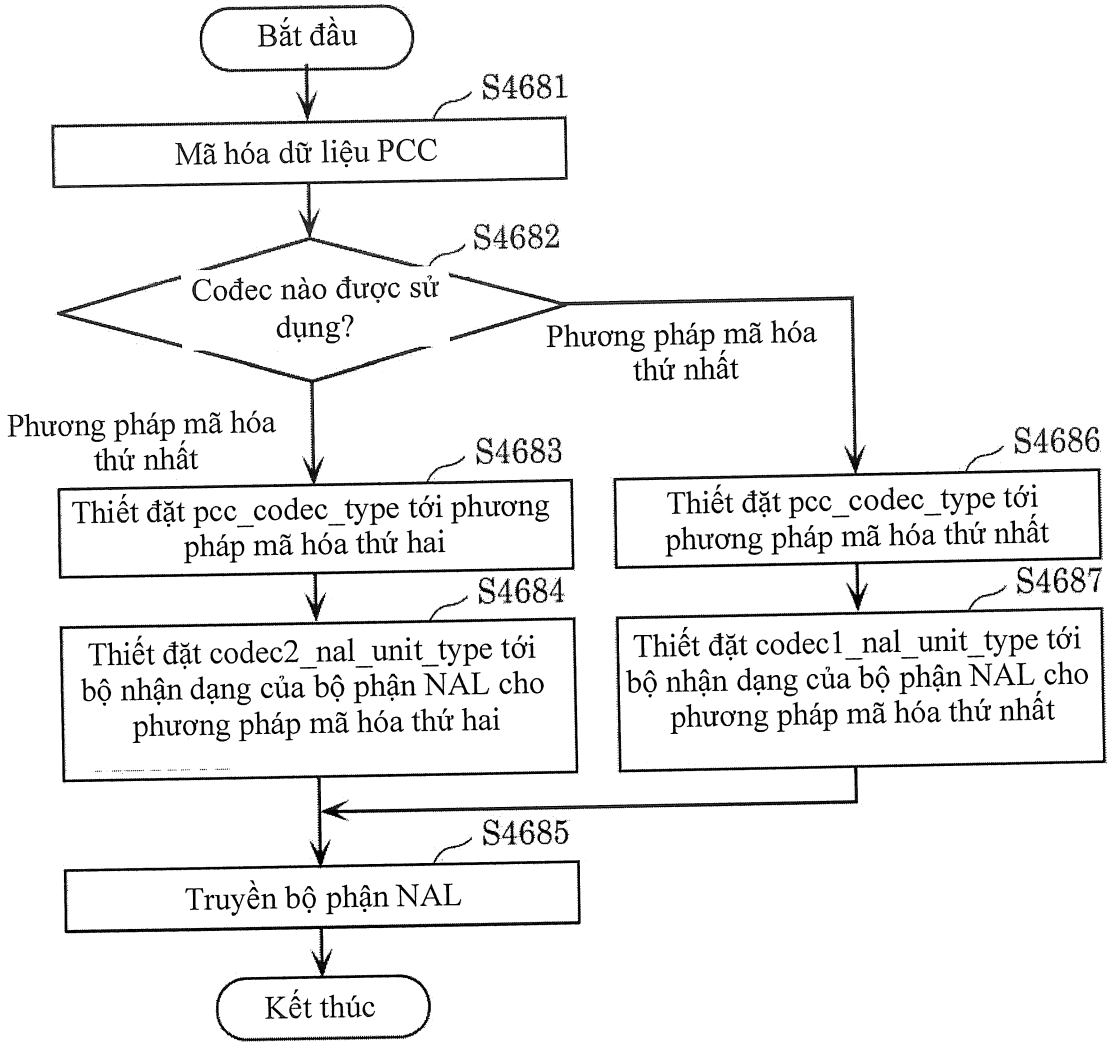

Bắt đầu
S4681
Mã hóa dữ liệu PCC
S4682
Codec nào được sử dụng?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ nhất
S4683
Thiết đặt pcc_codec_type tới phương pháp mã hóa thứ hai
S4684
Thiết đặt codec2_nal_unit_type tới bộ nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ hai
S4686
Thiết đặt pcc_codec_type tới phương pháp mã hóa thứ nhất
S4687
Thiết đặt codec1_nal_unit_type tới bộ nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ nhất
S4685
Truyền bộ phận NAL
Kết thúc

FIG. 17

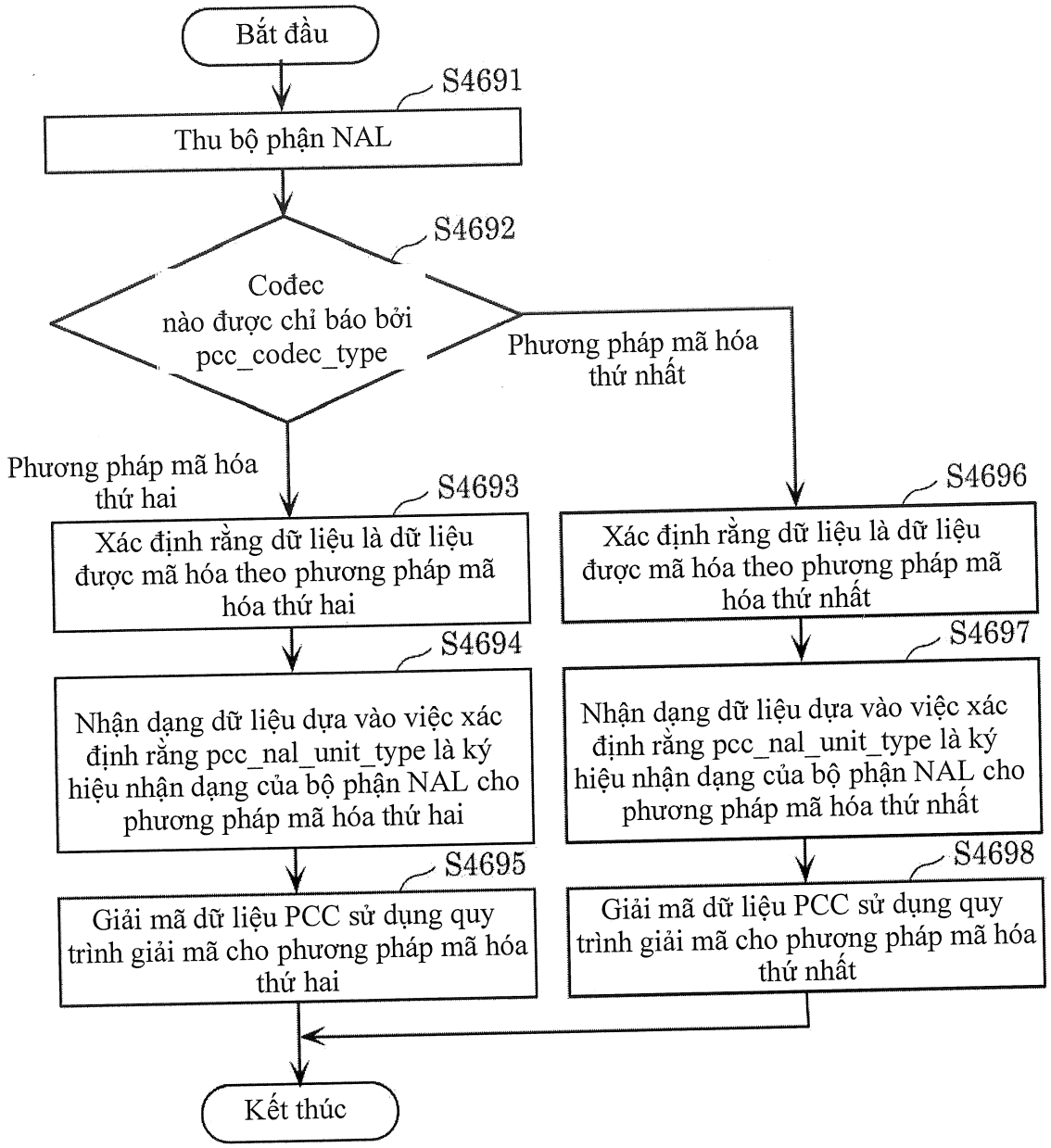

Bắt đầu
S4691
Thu bộ phận NAL
S4692
Codec nào được chỉ báo bởi pcc_codec_type
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
S4693
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai
S4694
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là ký hiệu nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ hai
S4695
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ hai
S4696
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất
S4697
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là ký hiệu nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ nhất
S4698
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ nhất
Kết thúc

FIG. 18

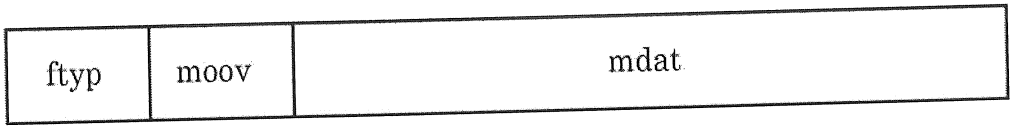

ftyp
moov
mdat

FIG. 19

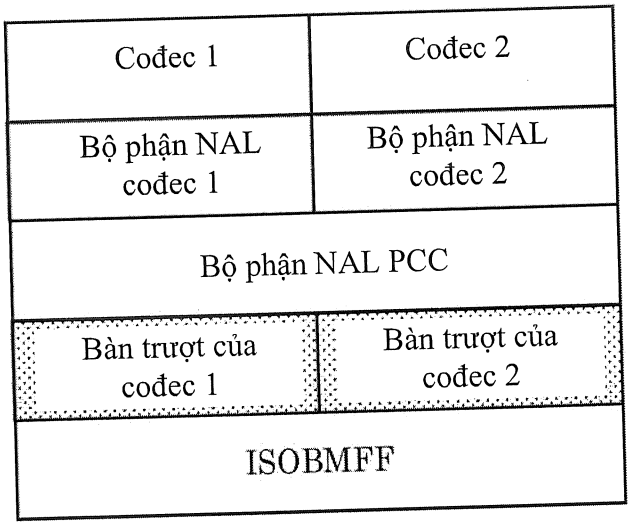

Codec 1
Codec 2
Bộ phận NAL codec 1
Bộ phận NAL codec 2
Bộ phận NAL PCC
Bàn trượt của codec 1
Bàn trượt của codec 2
ISOBMFF

FIG. 20

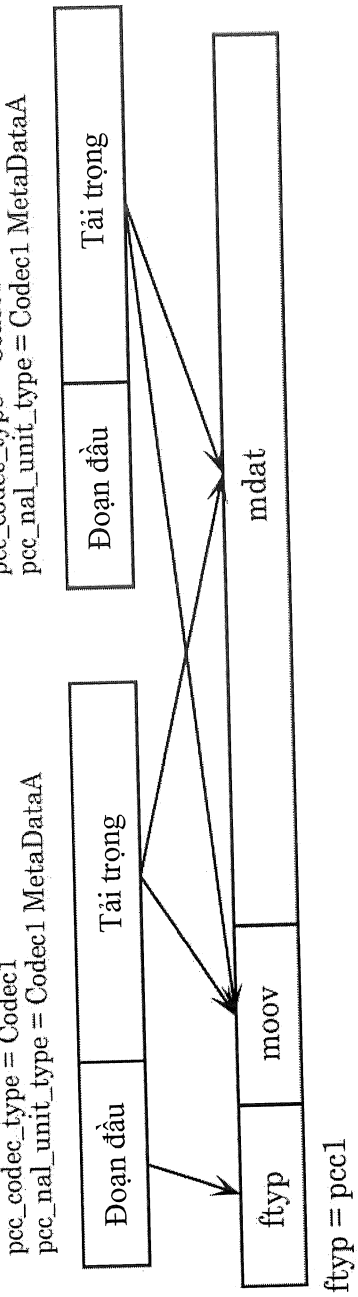

pcc_codec_type = Codec1
pcc_nal_unit_type = Codec1 MetaDataA
Đoạn đầu
Tải trọng
pcc_codec_type = Codec1
pcc_nal_unit_type = Codec1 MetaDataA
Đoạn đầu
Tải trọng
ftyp
moov
mdat
ftyp = pcc1
Bộ phận NAL PCC chung
ISOBMFF
(phương pháp lưu trữ cho
CODEC 1)

FIG. 21

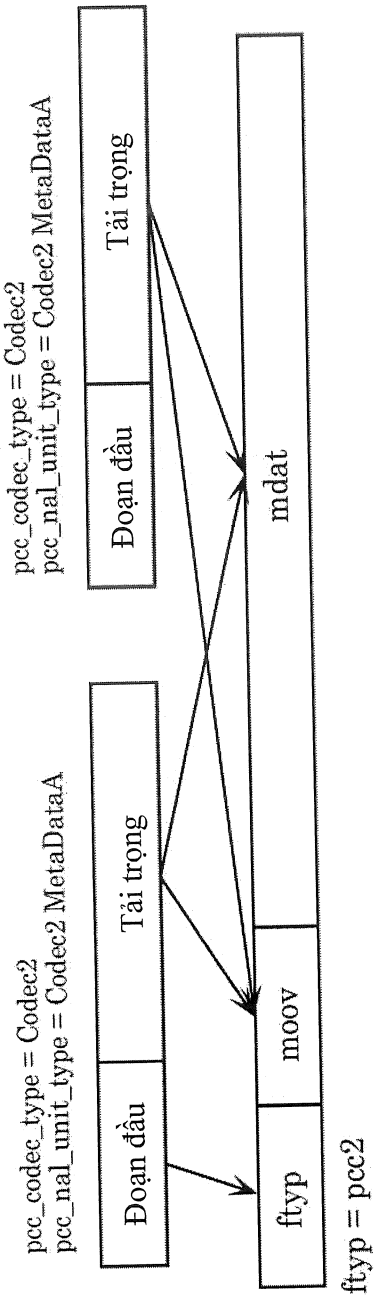

pcc_codec_type = Codec2
pcc_nal_unit_type = Codec2 MetaDataA
Đoạn đầu
Tải trọng
pcc_codec_type = Codec2
pcc_nal_unit_type = Codec2 MetaDataA
Đoạn đầu
Tải trọng
ftyp
moov
mdat
ftyp = pcc2
Bộ phận NAL PCC chung
ISOBMFF
(phương pháp lưu trữ cho
CODEC 2)

FIG. 22

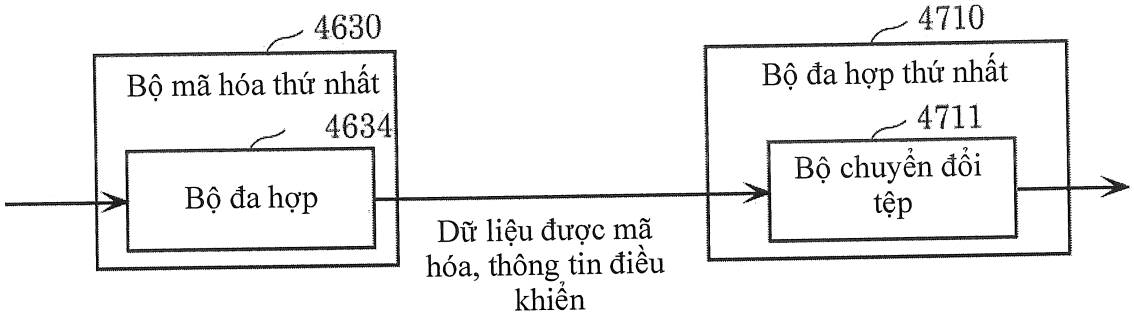


FIG. 23

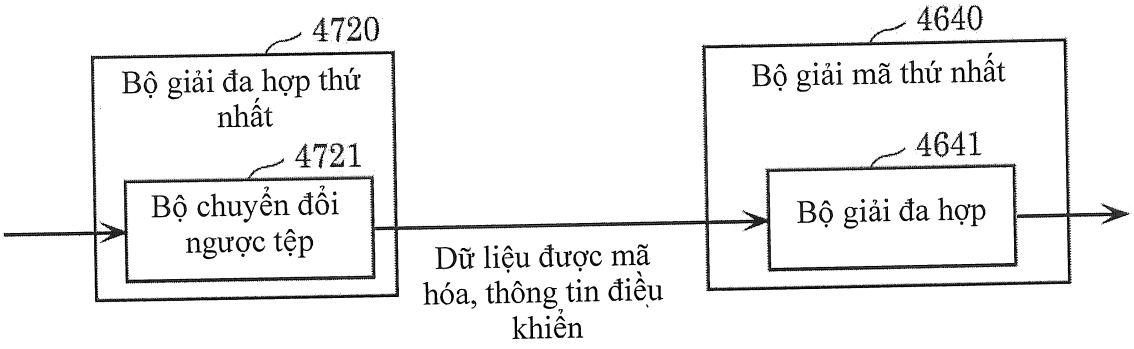


FIG. 24

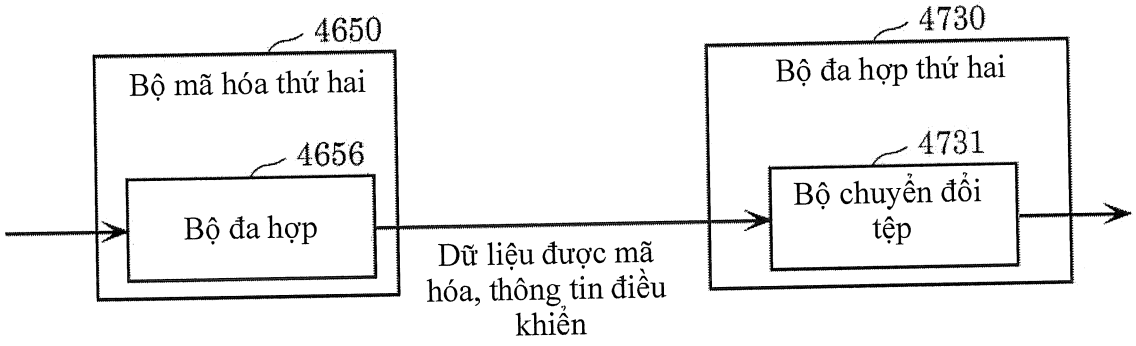


FIG. 25

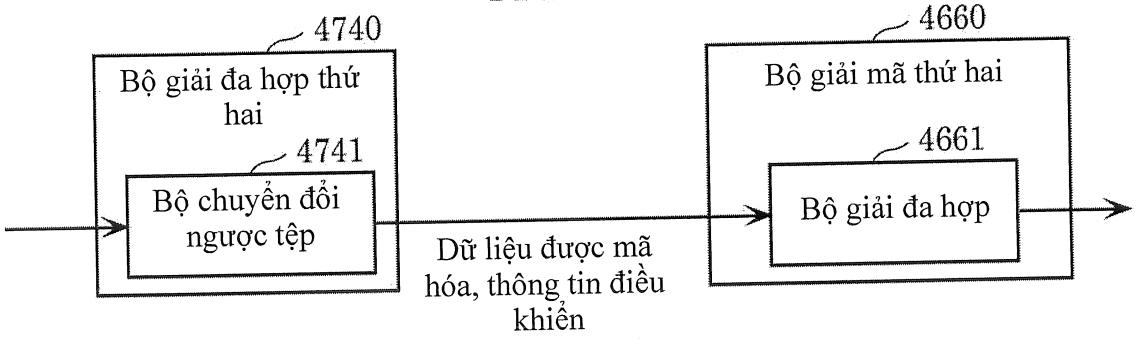


FIG. 26

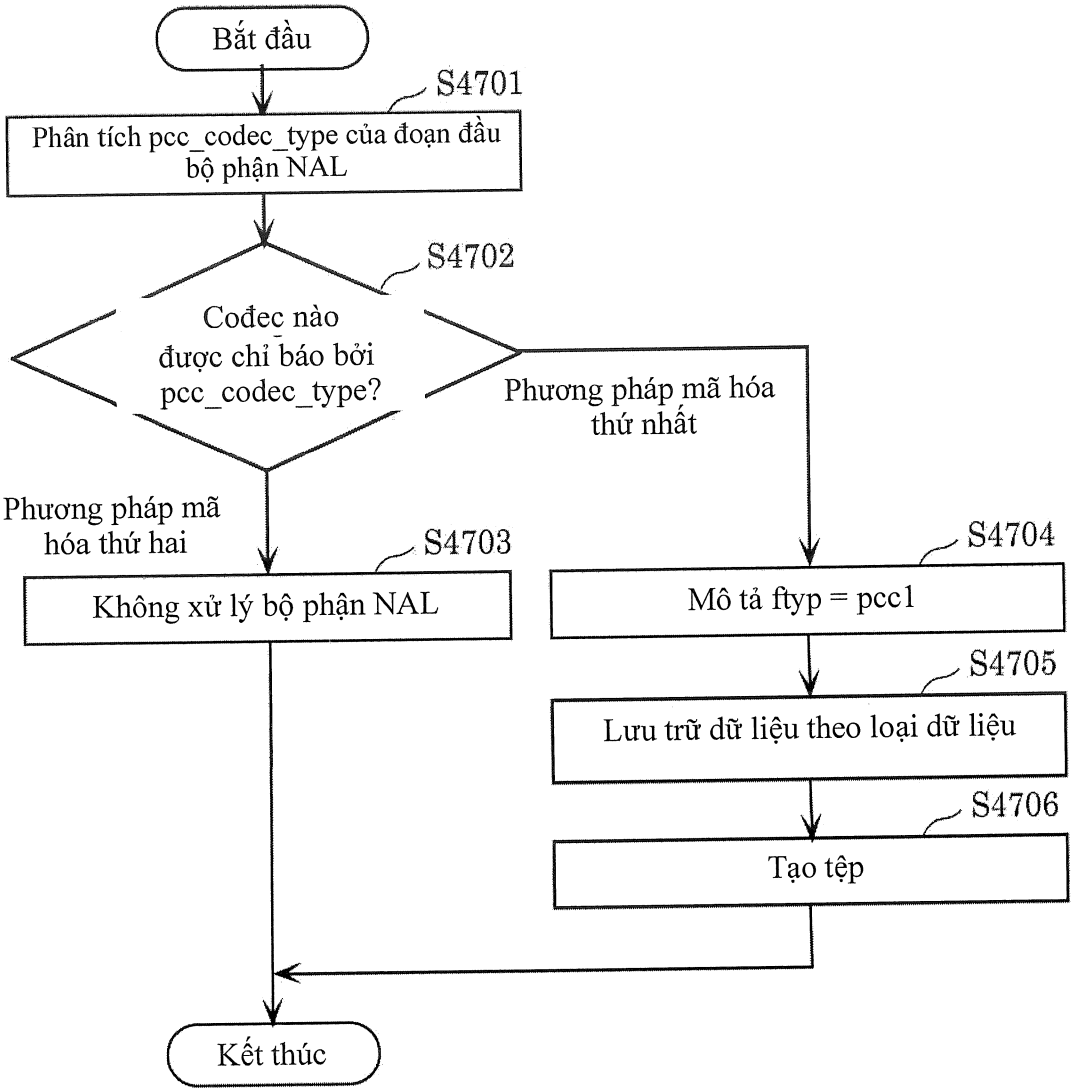

Bắt đầu
S4701
Phân tích pcc_codec_type của đoạn đầu bộ phận NAL
S4702
Codec nào được chỉ báo bởi pcc_codec_type?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
S4703
Không xử lý bộ phận NAL
S4704
Mô tả ftyp = pcc1
S4705
Lưu trữ dữ liệu theo loại dữ liệu
S4706
Tạo tệp
Kết thúc

FIG. 27

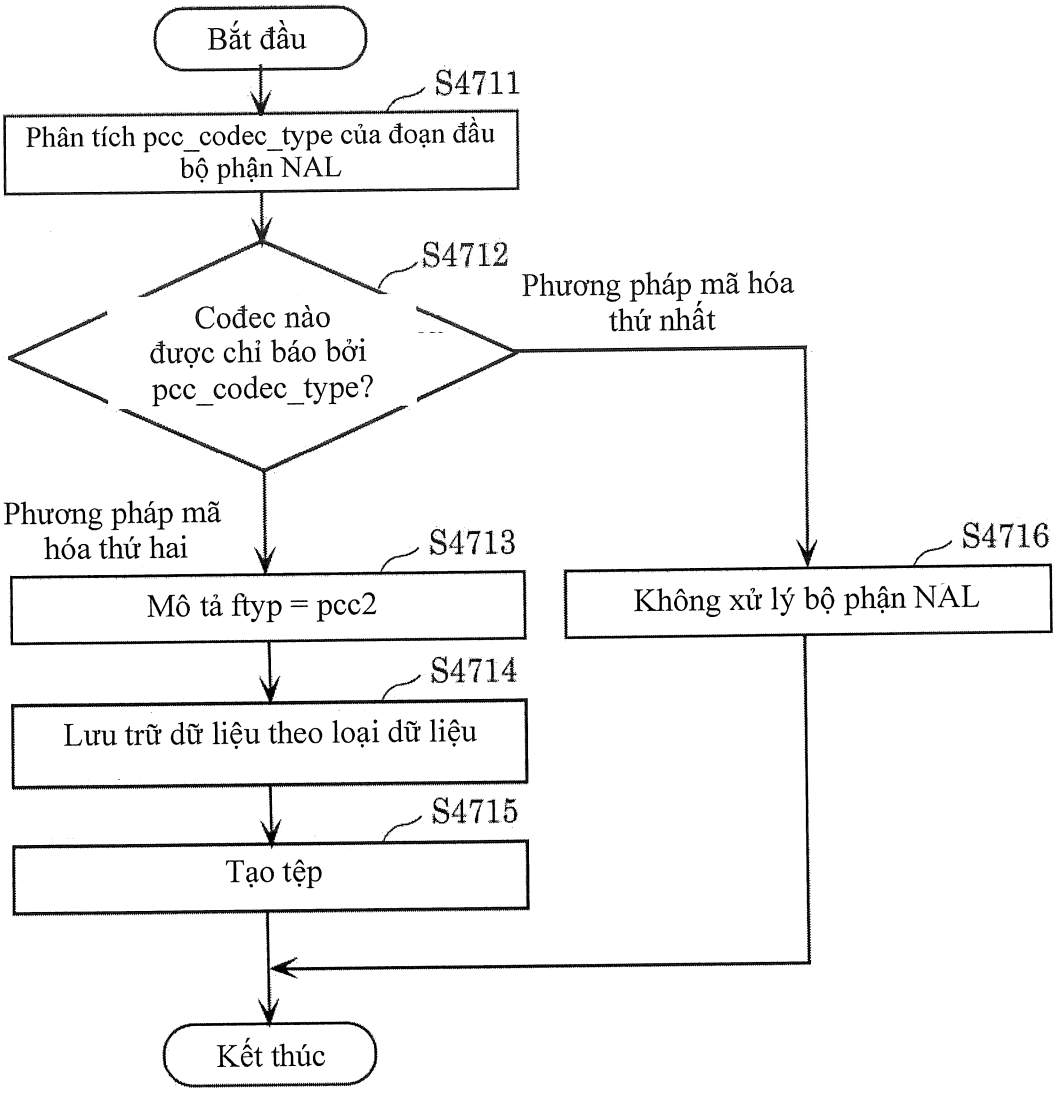

Bắt đầu
S4711
Phân tích pcc_codec_type của đoạn đầu bộ phận NAL
S4712
Codec nào được chỉ báo bởi pcc_codec_type?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
S4713
Mô tả ftyp = pcc2
S4714
Lưu trữ dữ liệu theo loại dữ liệu
S4715
Tạo tệp
S4716
Không xử lý bộ phận NAL
Kết thúc

FIG. 28

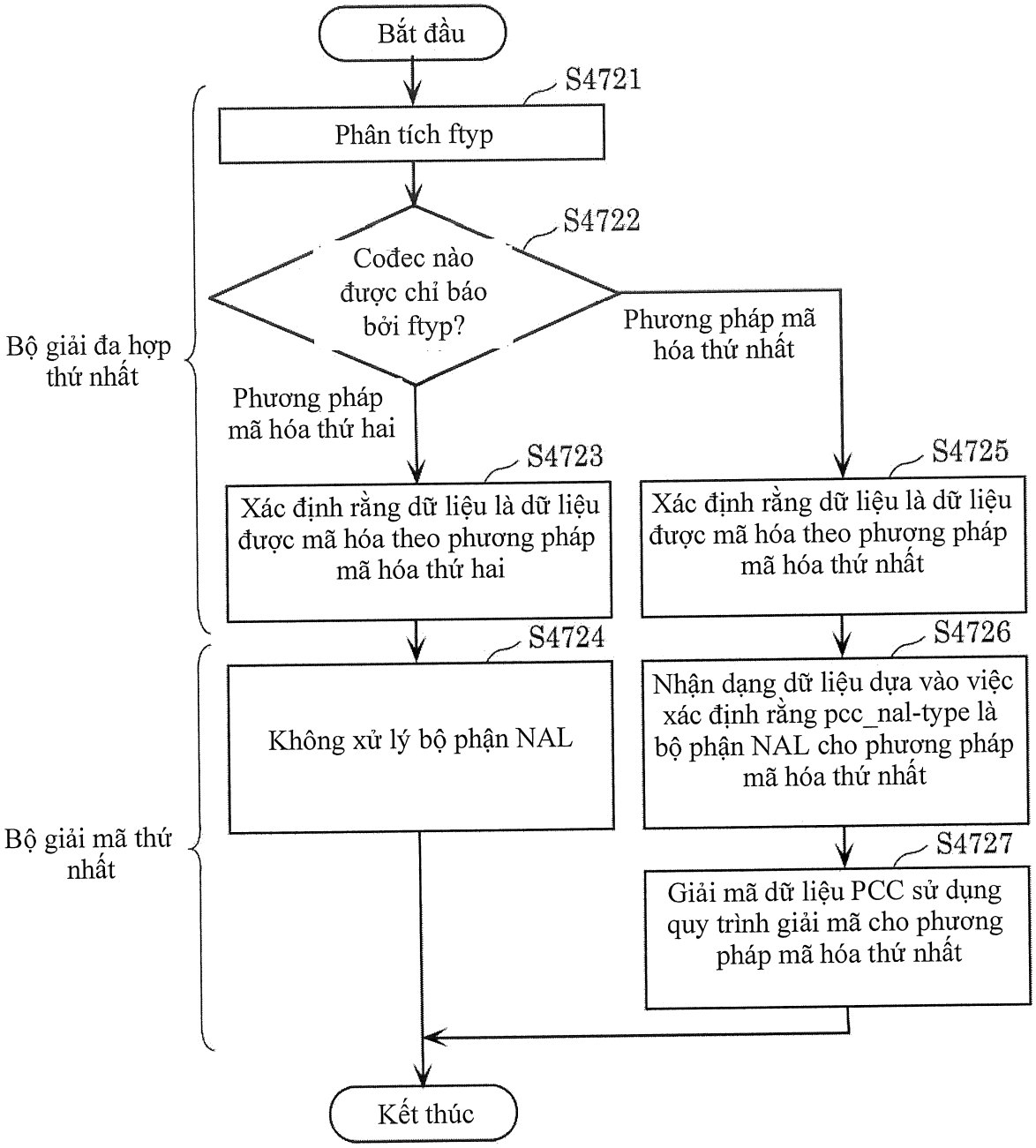

Bắt đầu
S4721
Phân tích ftyp
S4722
Cođec nào được chỉ báo bởi ftyp?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
Bộ giải đa hợp thứ nhất
S4723
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai
S4725
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất
S4724
Không xử lý bộ phận NAL
S4726
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal-type là bộ phận NAL cho phương pháp mã hóa thứ nhất
S4727
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ nhất
Bộ giải mã thứ nhất
Kết thúc

FIG. 29

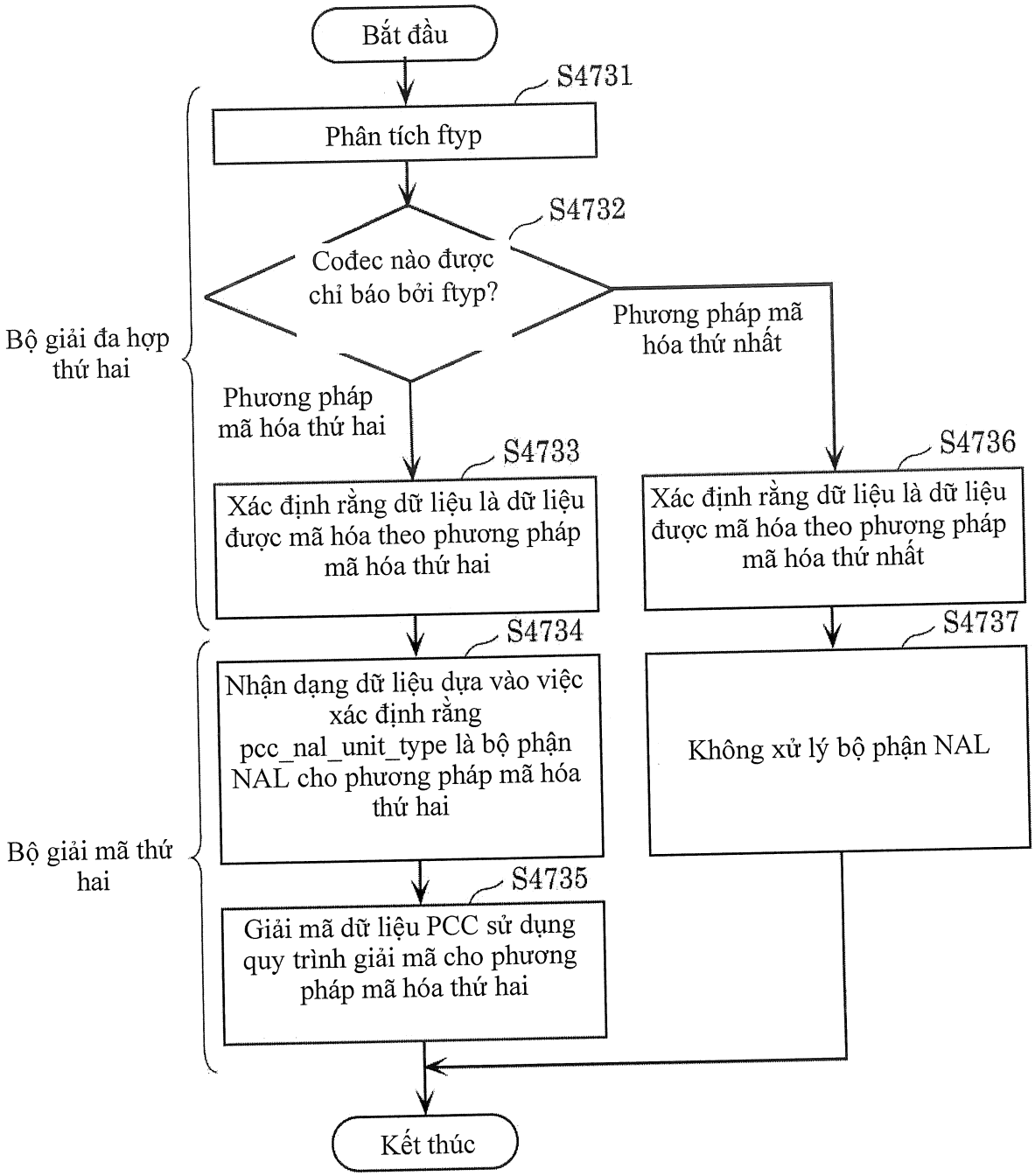

Bắt đầu
S4731
Phân tích ftyp
S4732
Cođec nào được chỉ báo bởi ftyp?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
Bộ giải đa hợp thứ hai
S4733
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai
S4736
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất
S4734
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là bộ phận NAL cho phương pháp mã hóa thứ hai
S4737
Không xử lý bộ phận NAL
Bộ giải mã thứ hai
S4735
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ hai
Kết thúc

FIG. 30

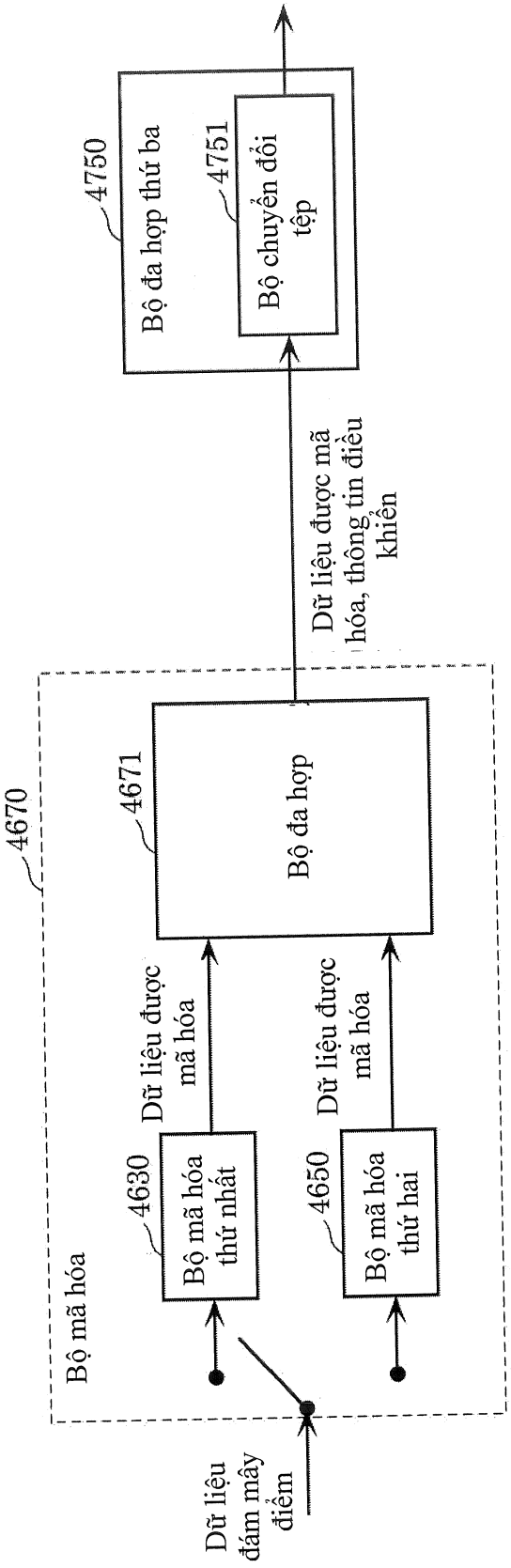

Dữ liệu đám mây điểm
Bộ mã hóa
4670
4630
Bộ mã hóa thứ nhất
4650
Bộ mã hóa thứ hai
Dữ liệu được mã hóa
Dữ liệu được mã hóa
4671
Bộ đa hợp
Dữ liệu được mã hóa, thông tin điều khiển
4750
Bộ đa hợp thứ ba
4751
Bộ chuyển đổi tệp

FIG. 31

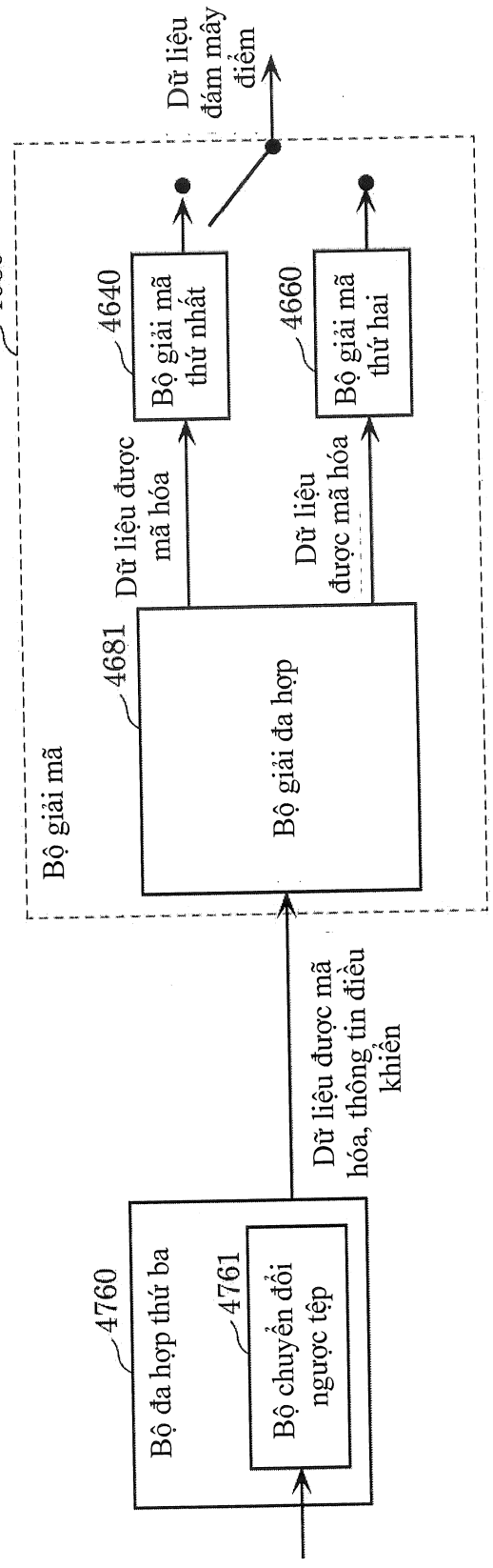

Bộ đa hợp thứ ba
4760
Bộ chuyển đổi ngược tệp
4761
Dữ liệu được mã hóa, thông tin điều khiển
Bộ giải mã
4680
Bộ giải đa hợp
4681
Dữ liệu được mã hóa
Dữ liệu được mã hóa
Bộ giải mã thứ hai
4660
Bộ giải mã thứ nhất
4640
Dữ liệu đám mây điểm

FIG. 32

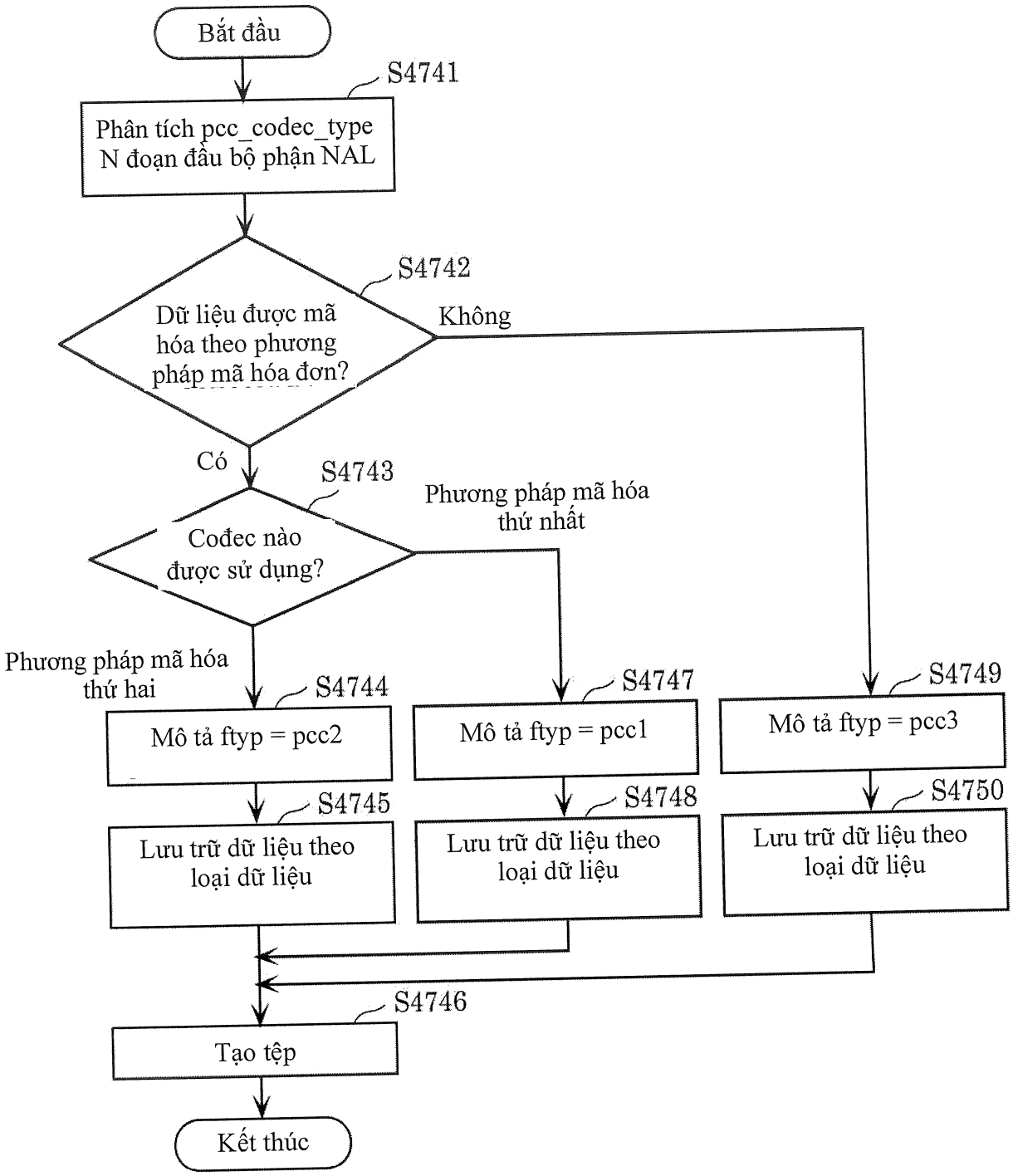

Bắt đầu
S4741
Phân tích pcc_codec_type
N đoạn đầu bộ phận NAL
S4742
Dữ liệu được mã hóa theo phương pháp mã hóa đơn?
Không
Có
S4743
Cođec nào được sử dụng?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
S4744
Mô tả ftyp = pcc2
S4745
Lưu trữ dữ liệu theo loại dữ liệu
S4747
Mô tả ftyp = pcc1
S4748
Lưu trữ dữ liệu theo loại dữ liệu
S4749
Mô tả ftyp = pcc3
S4750
Lưu trữ dữ liệu theo loại dữ liệu
S4746
Tạo tệp
Kết thúc

FIG. 33

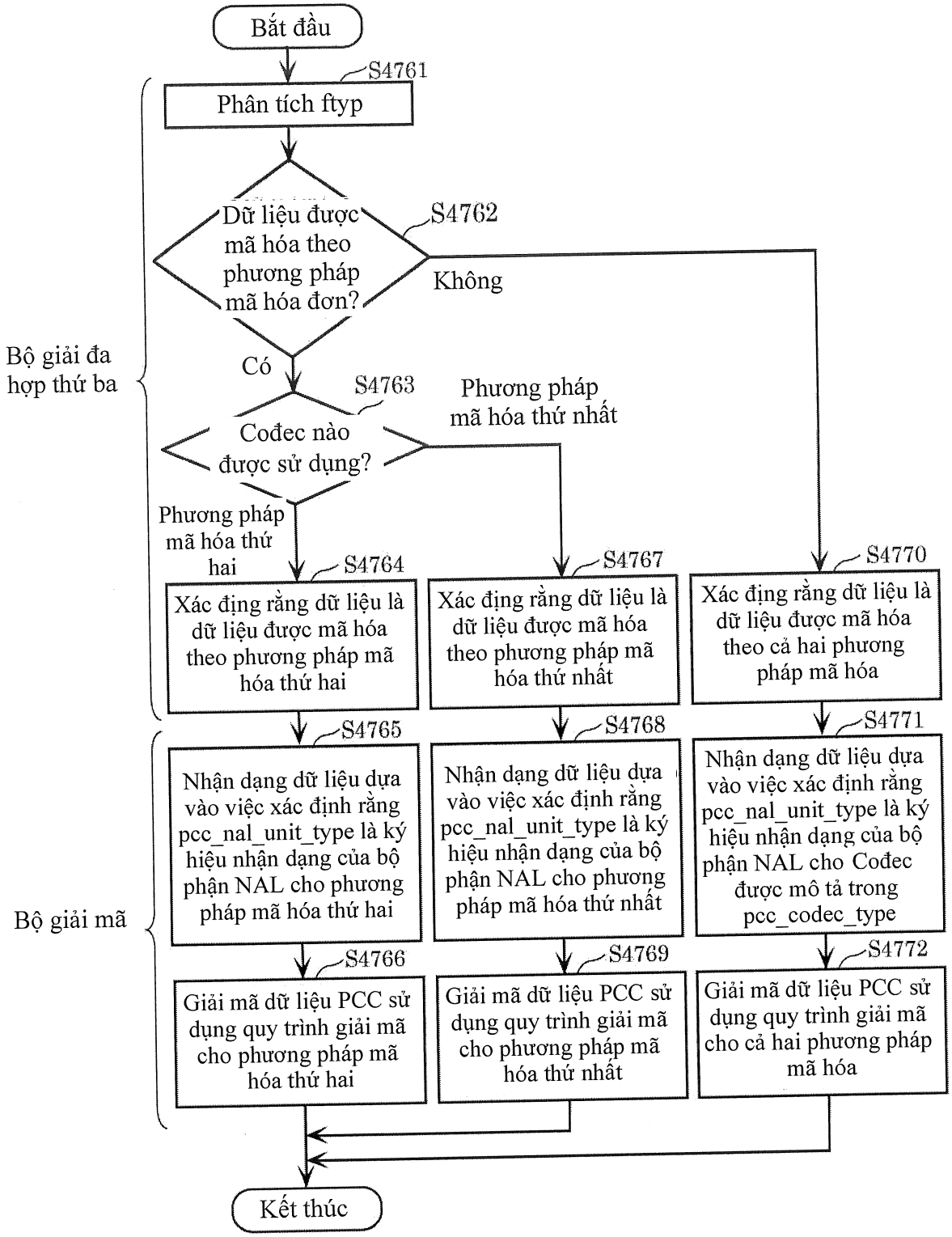

Bắt đầu
S4761
Phân tích ftyp
Dữ liệu được mã hóa theo phương pháp mã hóa đơn?
S4762
Không
Có
S4763
Codec nào được sử dụng?
Phương pháp mã hóa thứ nhất
Phương pháp mã hóa thứ hai
Bộ giải đa hợp thứ ba
S4764
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ hai
S4767
Xác định rằng dữ liệu là dữ liệu được mã hóa theo phương pháp mã hóa thứ nhất
S4770
Xác định rằng dữ liệu là dữ liệu được mã hóa theo cả hai phương pháp mã hóa
S4765
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là ký hiệu nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ hai
S4768
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là ký hiệu nhận dạng của bộ phận NAL cho phương pháp mã hóa thứ nhất
S4771
Nhận dạng dữ liệu dựa vào việc xác định rằng pcc_nal_unit_type là ký hiệu nhận dạng của bộ phận NAL cho Codec được mô tả trong pcc_codec_type
Bộ giải mã
S4766
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ hai
S4769
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho phương pháp mã hóa thứ nhất
S4772
Giải mã dữ liệu PCC sử dụng quy trình giải mã cho cả hai phương pháp mã hóa
Kết thúc

FIG. 34

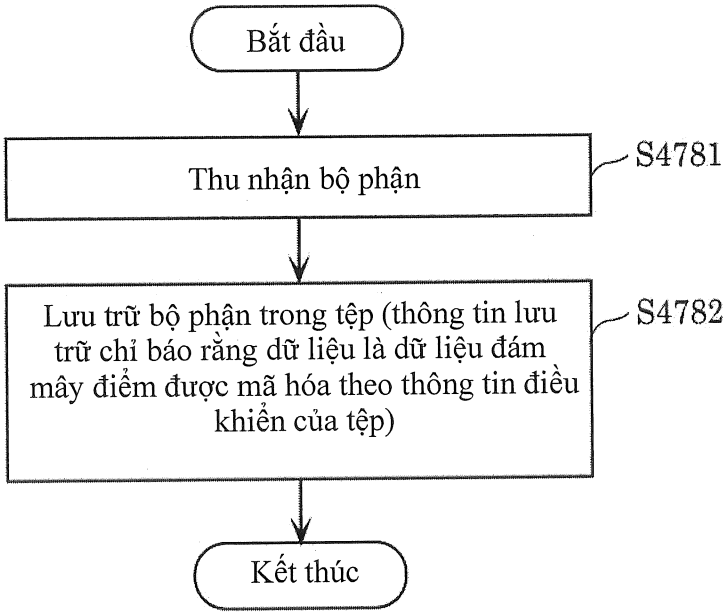

Bắt đầu
Thu nhận bộ phận
S4781
Lưu trữ bộ phận trong tệp (thông tin lưu trữ chỉ báo rằng dữ liệu là dữ liệu đám mây điểm được mã hóa theo thông tin điều khiển của tệp)
S4782
Kết thúc

FIG. 35

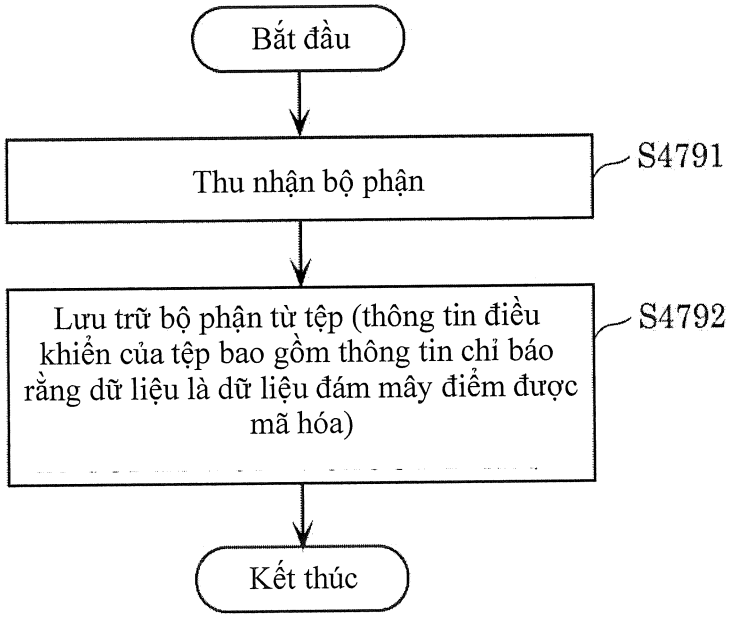

Bắt đầu
Thu nhận bộ phận
S4791
Lưu trữ bộ phận từ tệp (thông tin điều khiển của tệp bao gồm thông tin chỉ báo rằng dữ liệu là dữ liệu đám mây điểm được mã hóa)
S4792
Kết thúc

FIG. 36

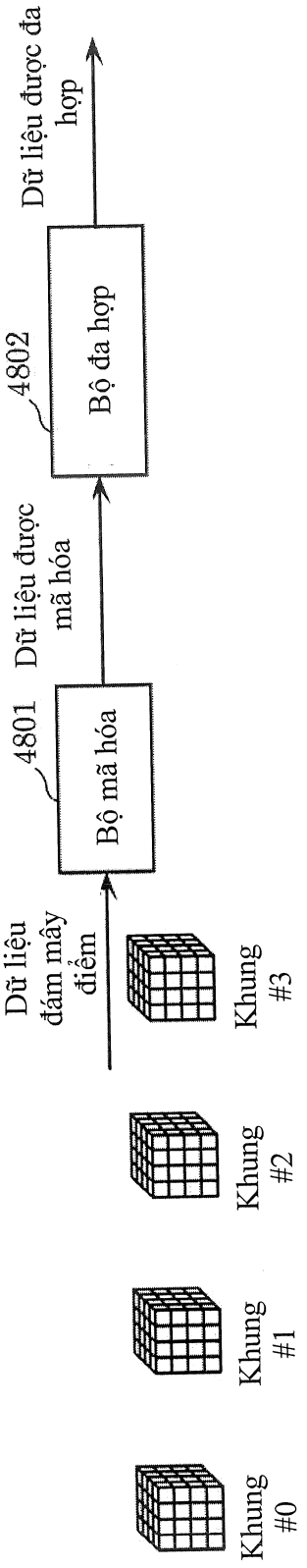

Khung #0
Khung #1
Khung #2
Khung #3
Dữ liệu đám mây điểm
4801
Bộ mã hóa
Dữ liệu được mã hóa
4802
Bộ đa hợp
Dữ liệu được đa hợp

FIG. 37

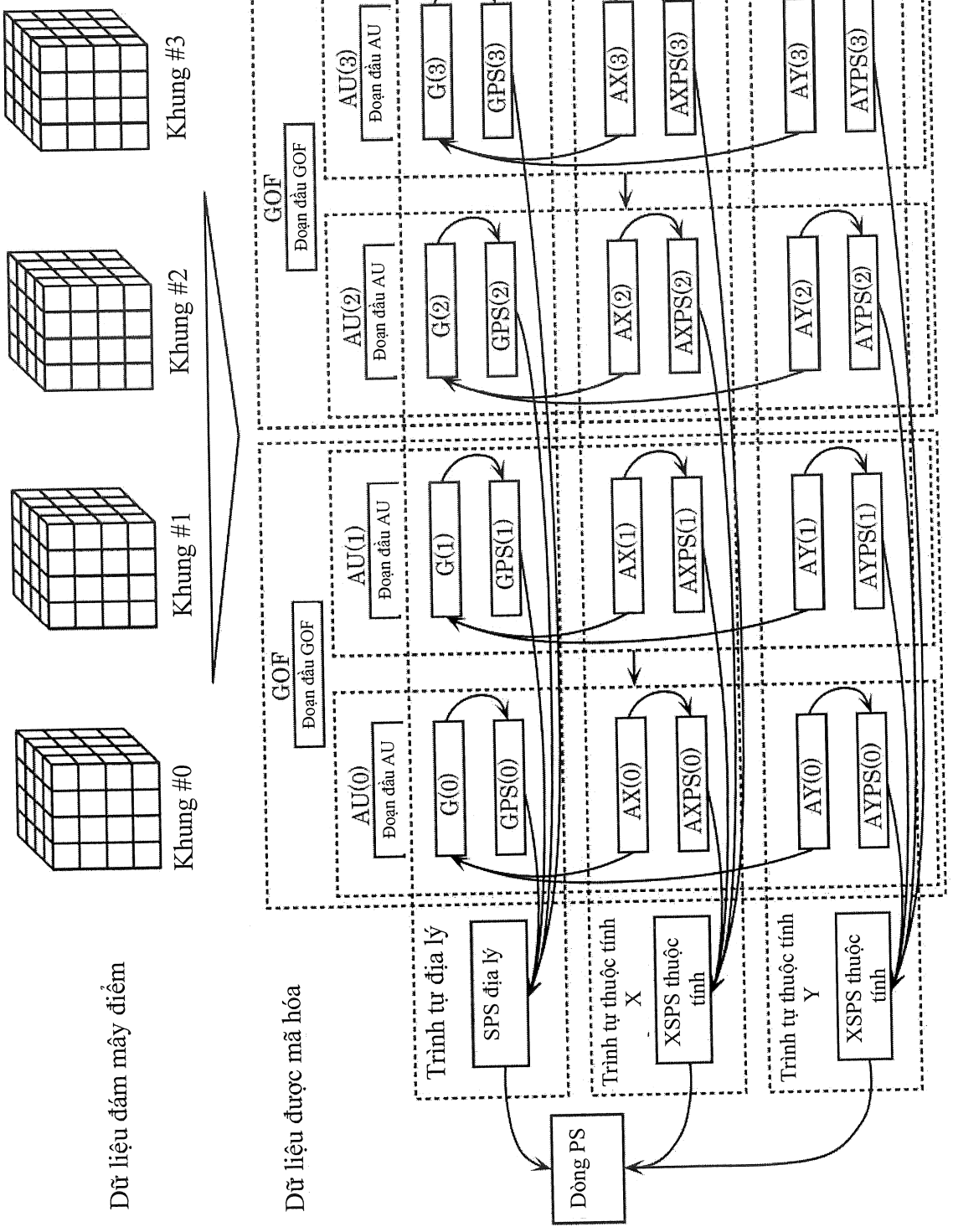

Dữ liệu đám mây điểm
Khung #0
Khung #1
Khung #2
Khung #3
Dữ liệu được mã hóa
GOF
Đoạn đầu GOF
AU(0)
Đoạn đầu AU
AU(1)
Đoạn đầu AU
AU(2)
Đoạn đầu AU
AU(3)
Đoạn đầu AU
G(0)
G(1)
G(2)
G(3)
GPS(0)
GPS(1)
GPS(2)
GPS(3)
AX(0)
AX(1)
AX(2)
AX(3)
AXPS(0)
AXPS(1)
AXPS(2)
AXPS(3)
AY(0)
AY(1)
AY(2)
AY(3)
AYPS(0)
AYPS(1)
AYPS(2)
AYPS(3)
Trình tự địa lý
SPS địa lý
Trình tự thuộc tính X
XSPS thuộc tính
Trình tự thuộc tính Y
XSPS thuộc tính
Dòng PS

FIG. 38

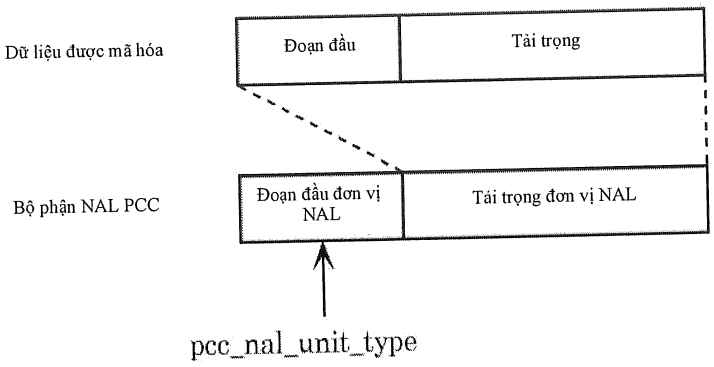

Dữ liệu được mã hóa
Đoạn đầu
Tải trọng
Bộ phận NAL PCC
Đoạn đầu đơn vị NAL
Tải trọng đơn vị NAL
pcc_nal_unit_type

FIG. 39

Các ví dụ về các nghữ nghĩa của pcc_nal_unit_type
(1) nếu pcc_codec_type == Codec 1
0: Codec1 Địa lý
1: Codec1 Thuộc tính X
2: Codec1 Thuộc tính Y
3: Codec1 PS địa lý
4: Codec1 PS thuộc tính X
5: Codec1 PS thuộc tính Y
6: Codec1 PS trình tự địa lý
7: Codec1 PS trình tự thuộc tính X
8: Codec1 PS trình tự thuộc tính Y
9: Codec1 Đoạn đầu AU
10: Codec1 Đoạn đầu GOF
11: Codec1 được dành cho việc sử dụng trong tương lai

(2) nếu ppcc_codec_type = = Codec 2
0: Codec2 Dữ liệu A
1: Codec2 Diều dữ liệu A
2: Codec2 Siêu dữ liệu B
3~: Codec2 được dành cho việc sử dụng trong tương lai

FIG. 40

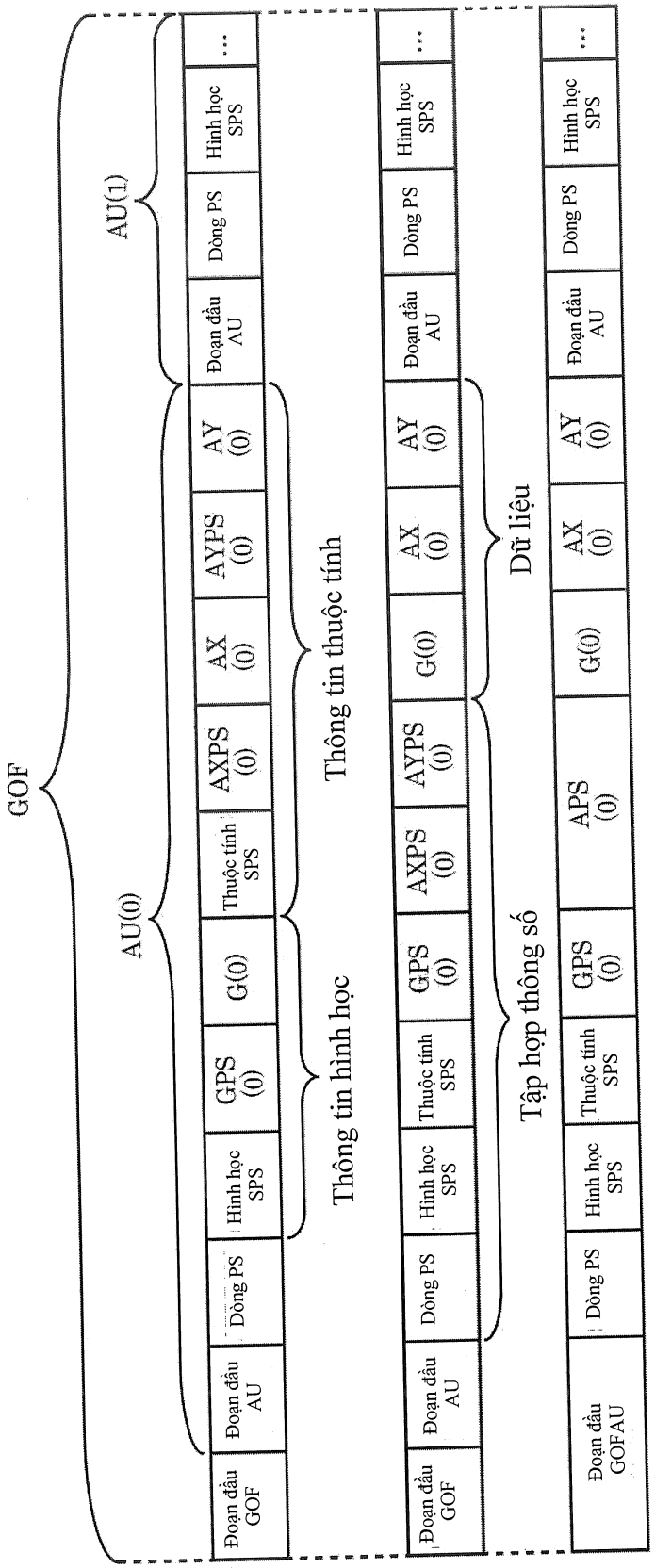

GOF
AU(0)
AU(1)
Đoạn đầu GOF
Đoạn đầu AU
Dòng PS
Hình học SPS
GPS (0)
G(0)
Thuộc tính SPS
AXPS (0)
AX (0)
AYPS (0)
AY (0)
Thông tin hình học
Thông tin thuộc tính
Dữ liệu
Tập hợp thông số
Thông tin hình học thứ nhất
Tập hợp thông số thứ nhất
Dữ liệu được tích hợp

FIG. 41

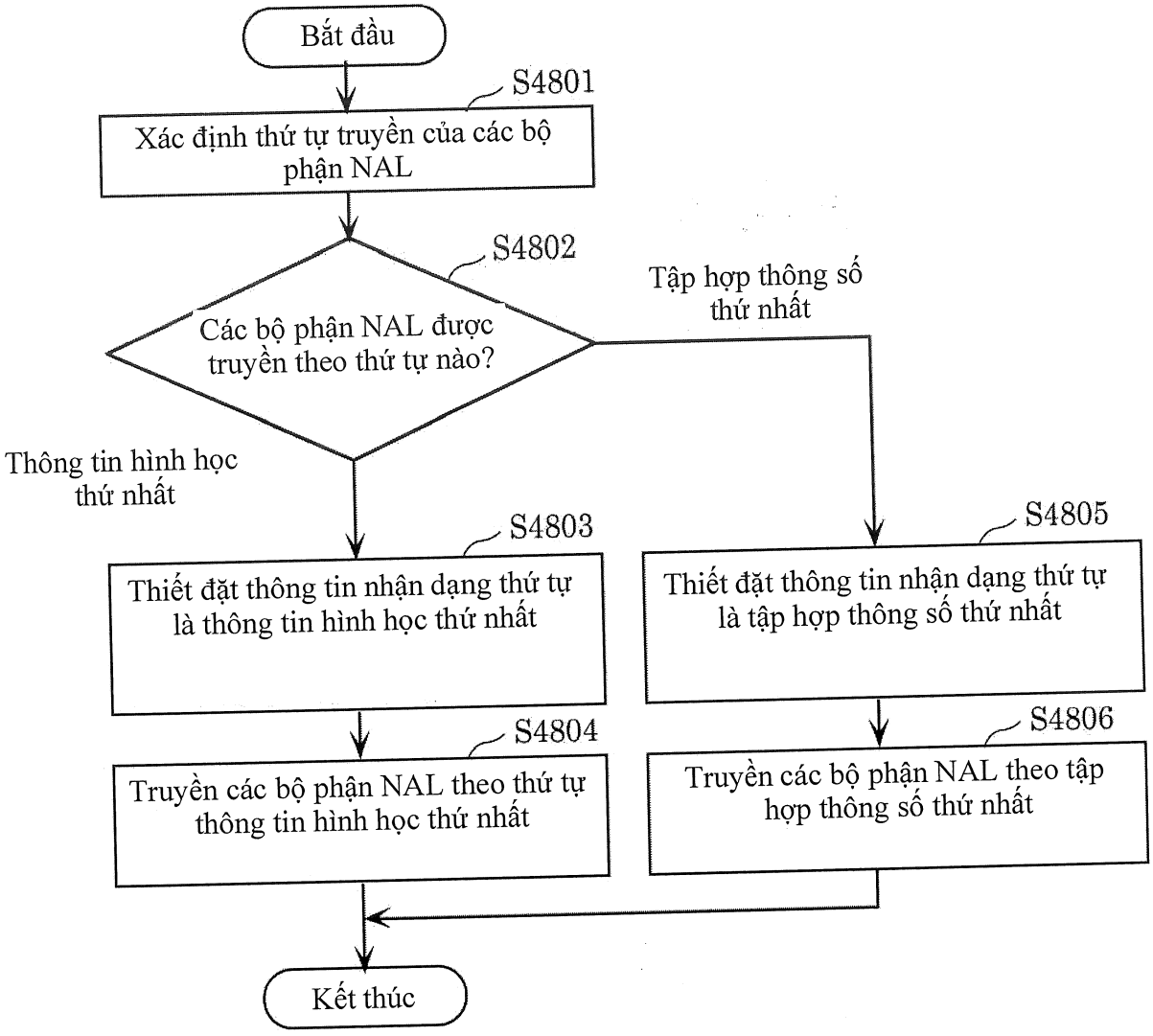

Bắt đầu
S4801
Xác định thứ tự truyền của các bộ phận NAL
S4802
Các bộ phận NAL được truyền theo thứ tự nào?
Tập hợp thông số thứ nhất
Thông tin hình học thứ nhất
S4803
Thiết đặt thông tin nhận dạng thứ tự là thông tin hình học thứ nhất
S4805
Thiết đặt thông tin nhận dạng thứ tự là tập hợp thông số thứ nhất
S4804
Truyền các bộ phận NAL theo thứ tự thông tin hình học thứ nhất
S4806
Truyền các bộ phận NAL theo tập hợp thông số thứ nhất
Kết thúc

FIG. 42

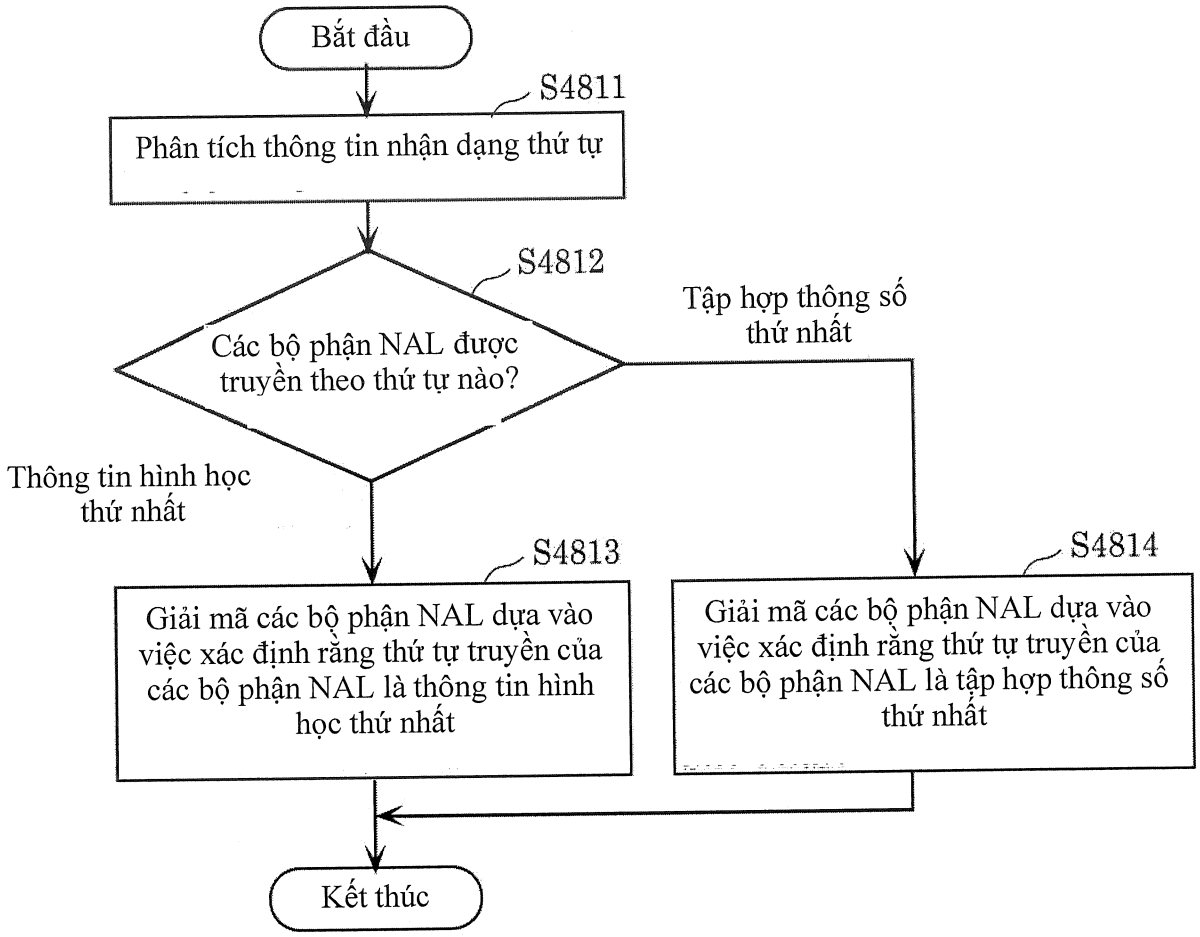

Bắt đầu
S4811
Phân tích thông tin nhận dạng thứ tự
S4812
Các bộ phận NAL được truyền theo thứ tự nào?
Tập hợp thông số thứ nhất
Thông tin hình học thứ nhất
S4813
Giải mã các bộ phận NAL dựa vào việc xác định rằng thứ tự truyền của các bộ phận NAL là thông tin hình học thứ nhất
S4814
Giải mã các bộ phận NAL dựa vào việc xác định rằng thứ tự truyền của các bộ phận NAL là tập hợp thông số thứ nhất
Kết thúc

FIG. 43

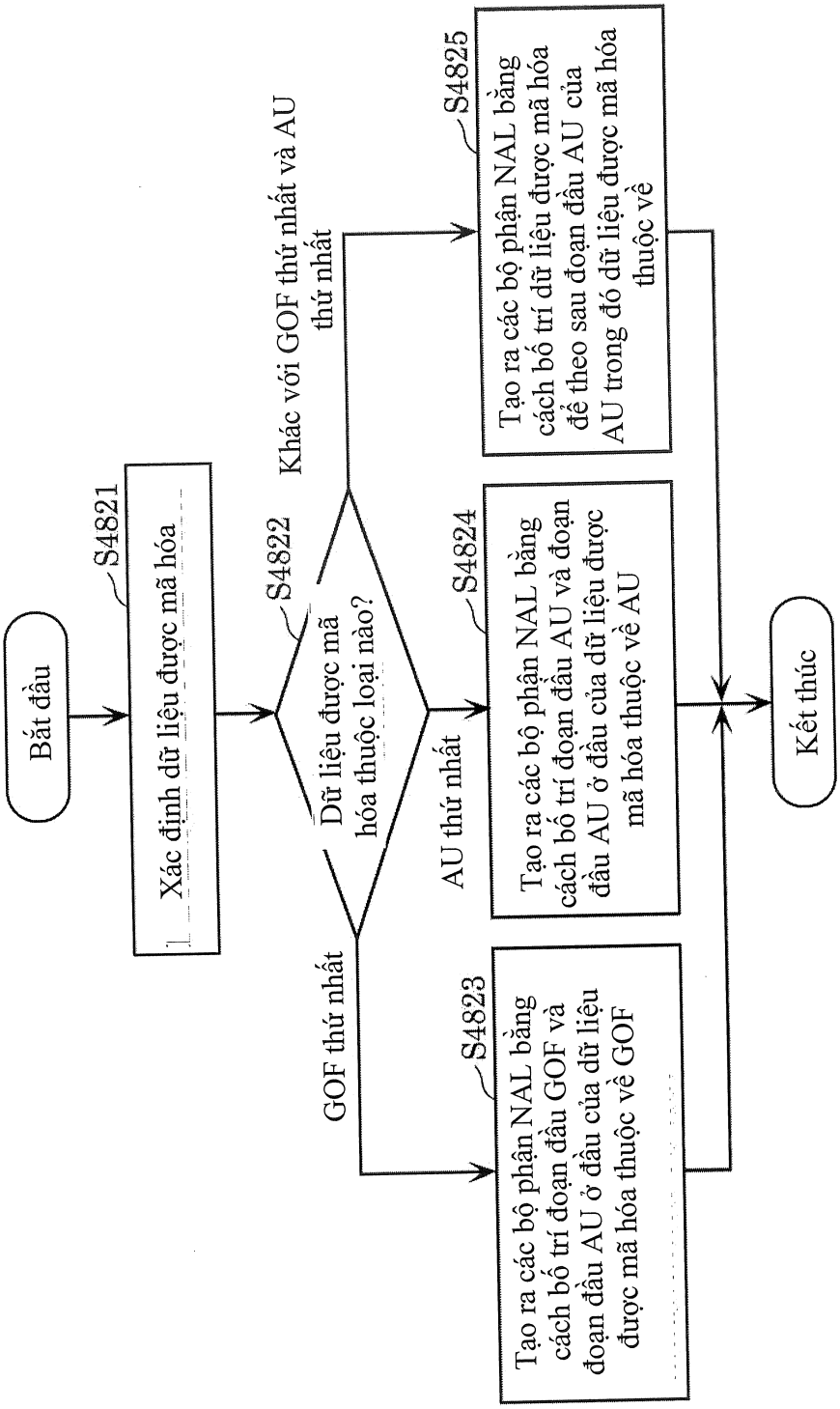

Bắt đầu
S4821
Xác định dữ liệu được mã hóa
S4822
Dữ liệu được mã hóa thuộc loại nào?
GOF thứ nhất
AU thứ nhất
Khác với GOF thứ nhất và AU thứ nhất
S4823
Tạo ra các bộ phận NAL bằng cách bố trí đoạn đầu GOF và đoạn đầu AU ở đầu của dữ liệu được mã hóa thuộc về GOF
S4824
Tạo ra các bộ phận NAL bằng cách bố trí đoạn đầu AU và đoạn đầu AU ở đầu của dữ liệu được mã hóa thuộc về AU
S4825
Tạo ra các bộ phận NAL bằng cách bố trí dữ liệu được mã hóa để theo sau đoạn đầu AU của AU trong đó dữ liệu được mã hóa thuộc về
Kết thúc

FIG. 44

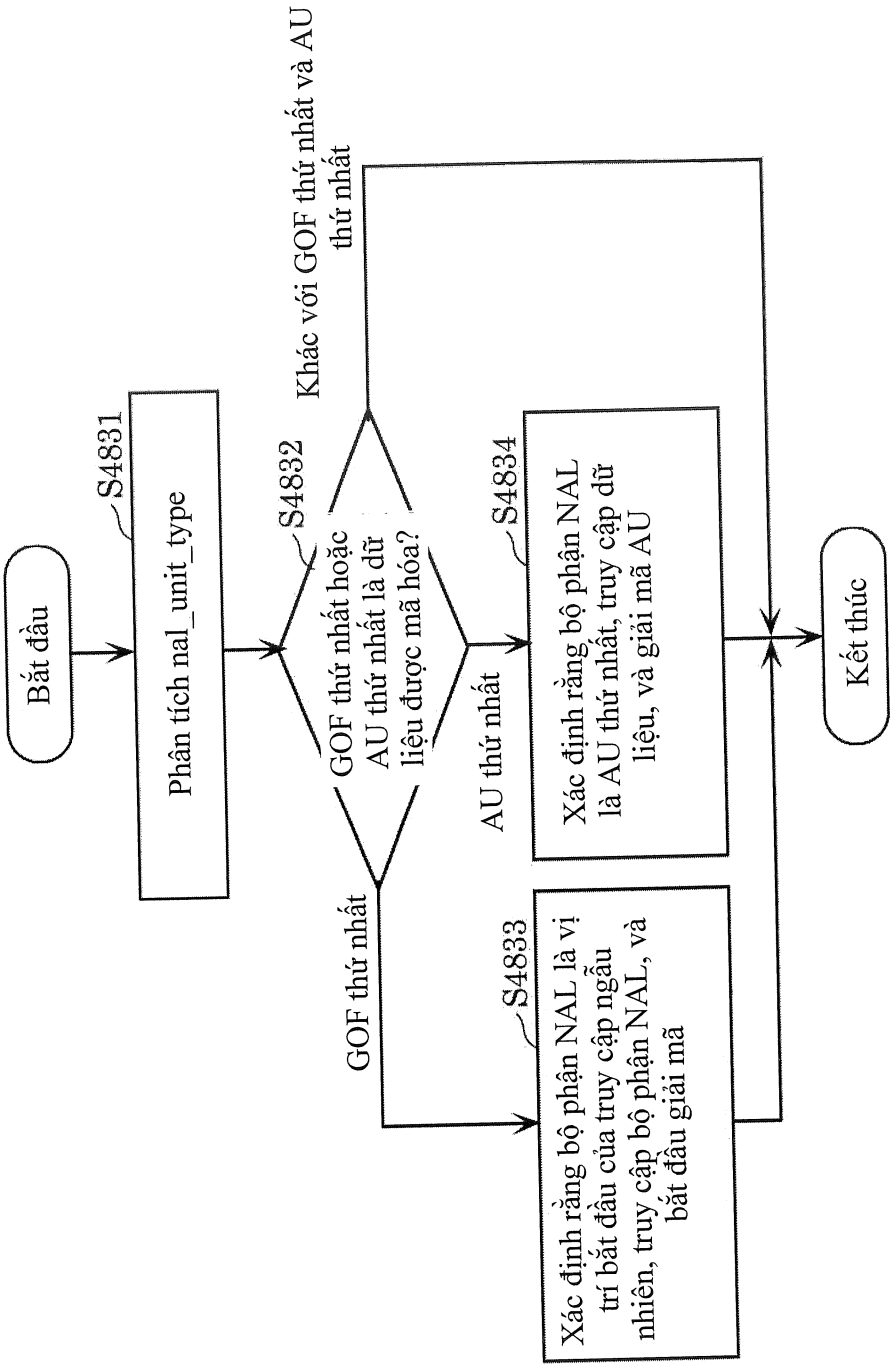

Bắt đầu
S4831
Phân tích nal_unit_type
S4832
GOF thứ nhất hoặc AU thứ nhất là dữ liệu được mã hóa?
Khác với GOF thứ nhất và AU thứ nhất
AU thứ nhất
S4834
Xác định rằng bộ phận NAL là AU thứ nhất, truy cập dữ liệu, và giải mã AU
GOF thứ nhất
S4833
Xác định rằng bộ phận NAL là vị trí bắt đầu của truy cập ngẫu nhiên, truy cập bộ phận NAL, và bắt đầu giải mã
Kết thúc

FIG. 45

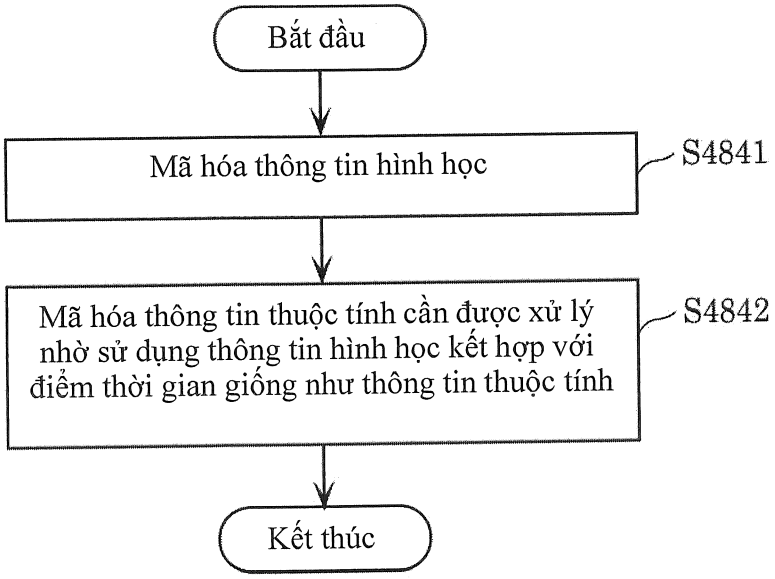

Bắt đầu
Mã hóa thông tin hình học
S4841
Mã hóa thông tin thuộc tính cần được xử lý nhờ sử dụng thông tin hình học kết hợp với điểm thời gian giống như thông tin thuộc tính
S4842
Kết thúc

FIG. 46

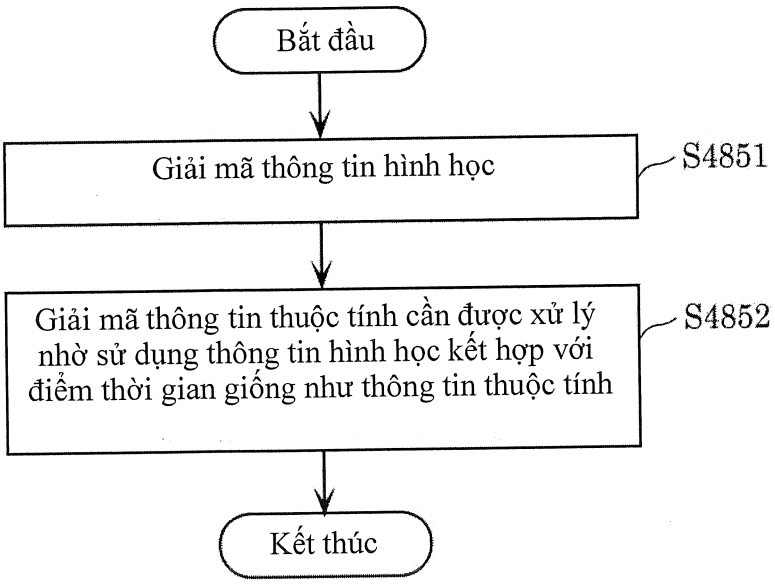

Bắt đầu
Giải mã thông tin hình học
S4851
Giải mã thông tin thuộc tính cần được xử lý nhờ sử dụng thông tin hình học kết hợp với điểm thời gian giống như thông tin thuộc tính
S4852
Kết thúc

FIG. 47

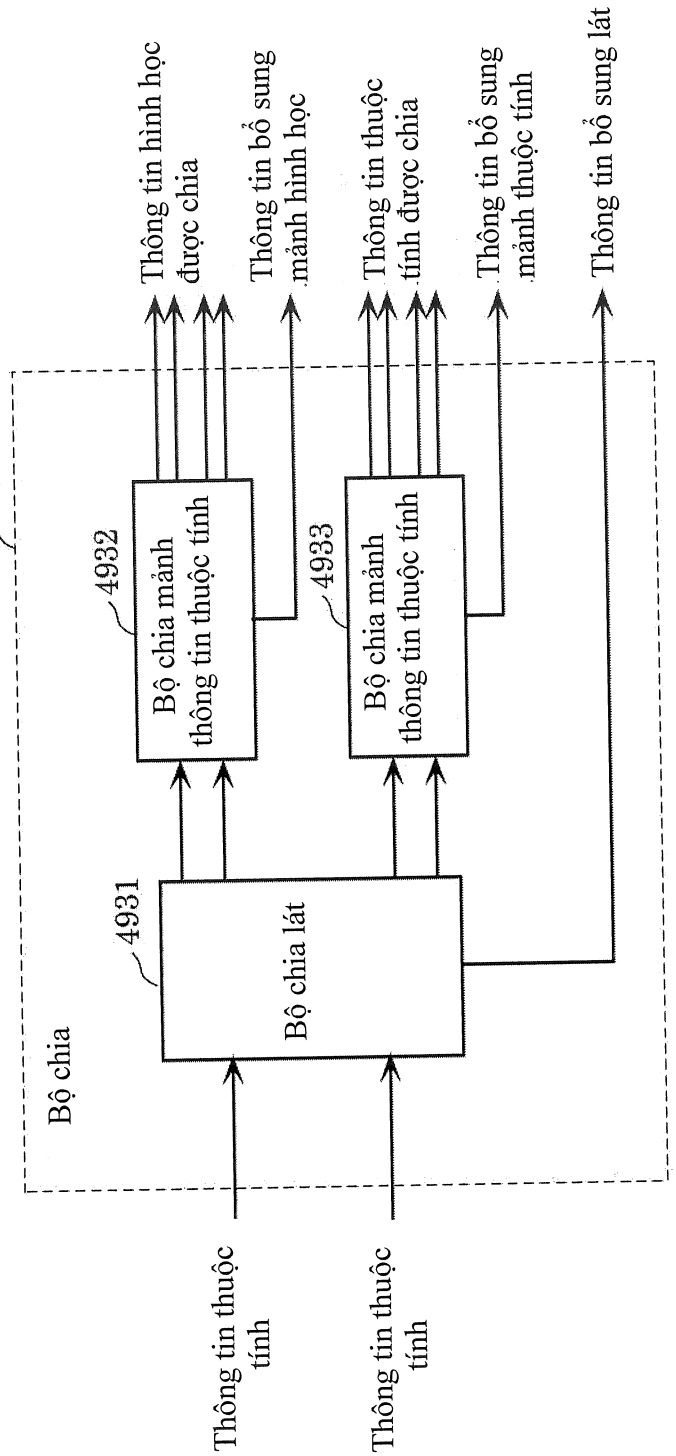

Thông tin thuộc tính
Thông tin thuộc tính
Bộ chia
4911
4931
Bộ chia lát
4932
Bộ chia mảnh thông tin thuộc tính
4933
Bộ chia mảnh thông tin thuộc tính
Thông tin hình học được chia
Thông tin bổ sung mảnh hình học
Thông tin thuộc tính được chia
Thông tin bổ sung mảnh thuộc tính
Thông tin bổ sung lát

FIG. 48

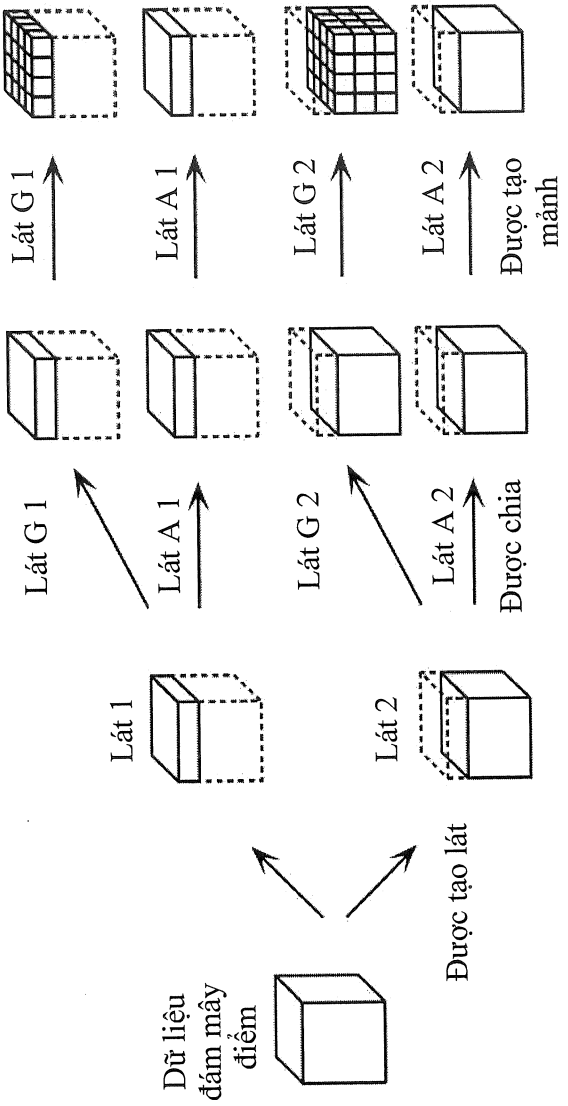

Dữ liệu đám mây điểm
Được tạo lát
Lát 1
Lát 2
Lát G 1
Lát A 1
Lát G 2
Lát A 2
Được chia
Lát G 1
Lát A 1
Lát G 2
Lát A 2
Được tạo mảnh

FIG. 49

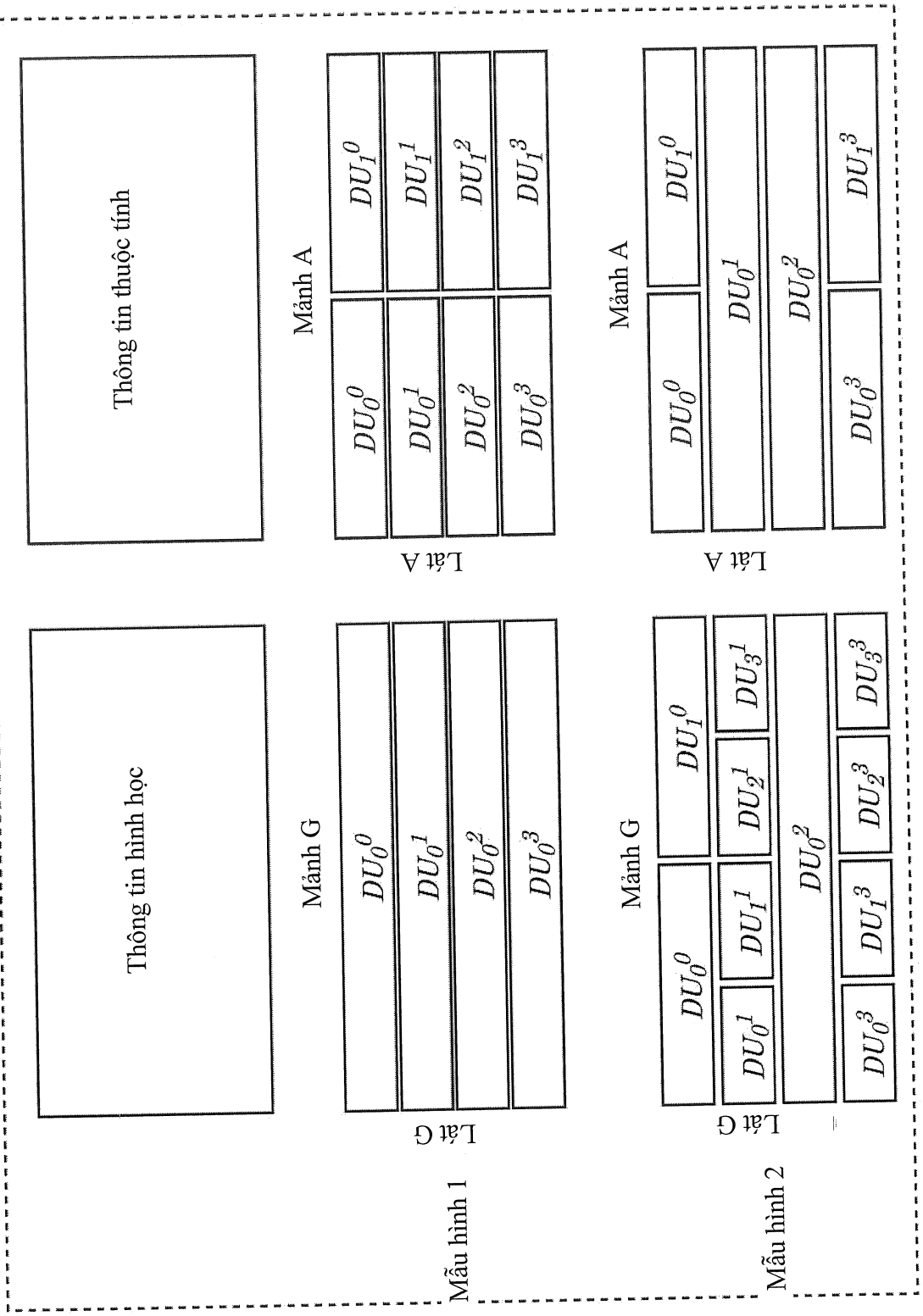

Thông tin hình học
Thông tin thuộc tính
Mảnh G
Mảnh A
Lát G
Lát A
DU_0^0
DU_0^1
DU_0^2
DU_0^3
DU_1^0
DU_1^1
DU_1^2
DU_1^3
DU_2^1
DU_2^3
DU_3^1
DU_3^3
Mẫu hình 1
Mẫu hình 2

FIG. 50

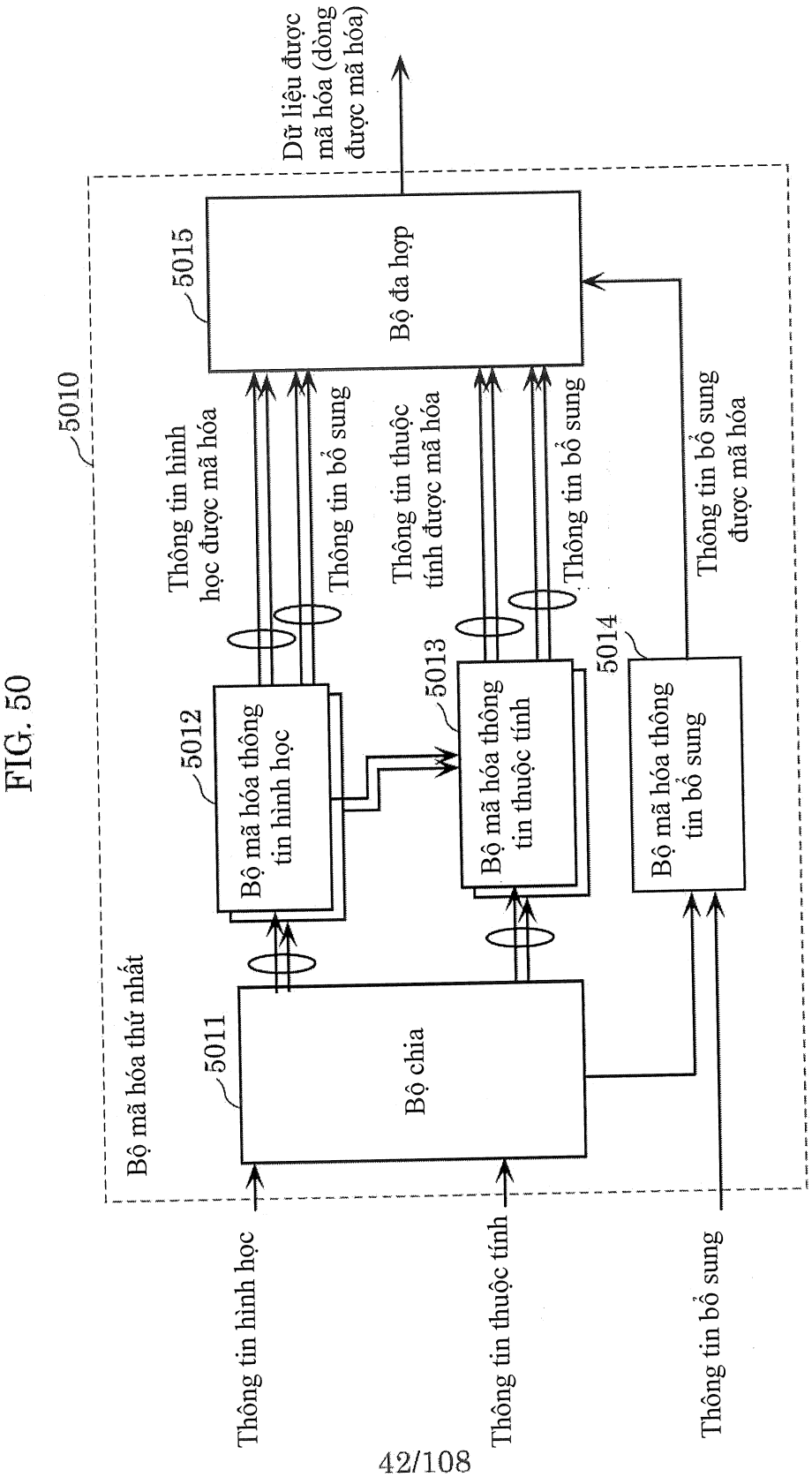

Dữ liệu được mã hóa (dòng được mã hóa)
5010
5015
Bộ đa hợp
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
5012
Bộ mã hóa thông tin hình học
5013
Bộ mã hóa thông tin thuộc tính
5014
Bộ mã hóa thông tin bổ sung
Bộ mã hóa thứ nhất
5011
Bộ chia
Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung

FIG. 51

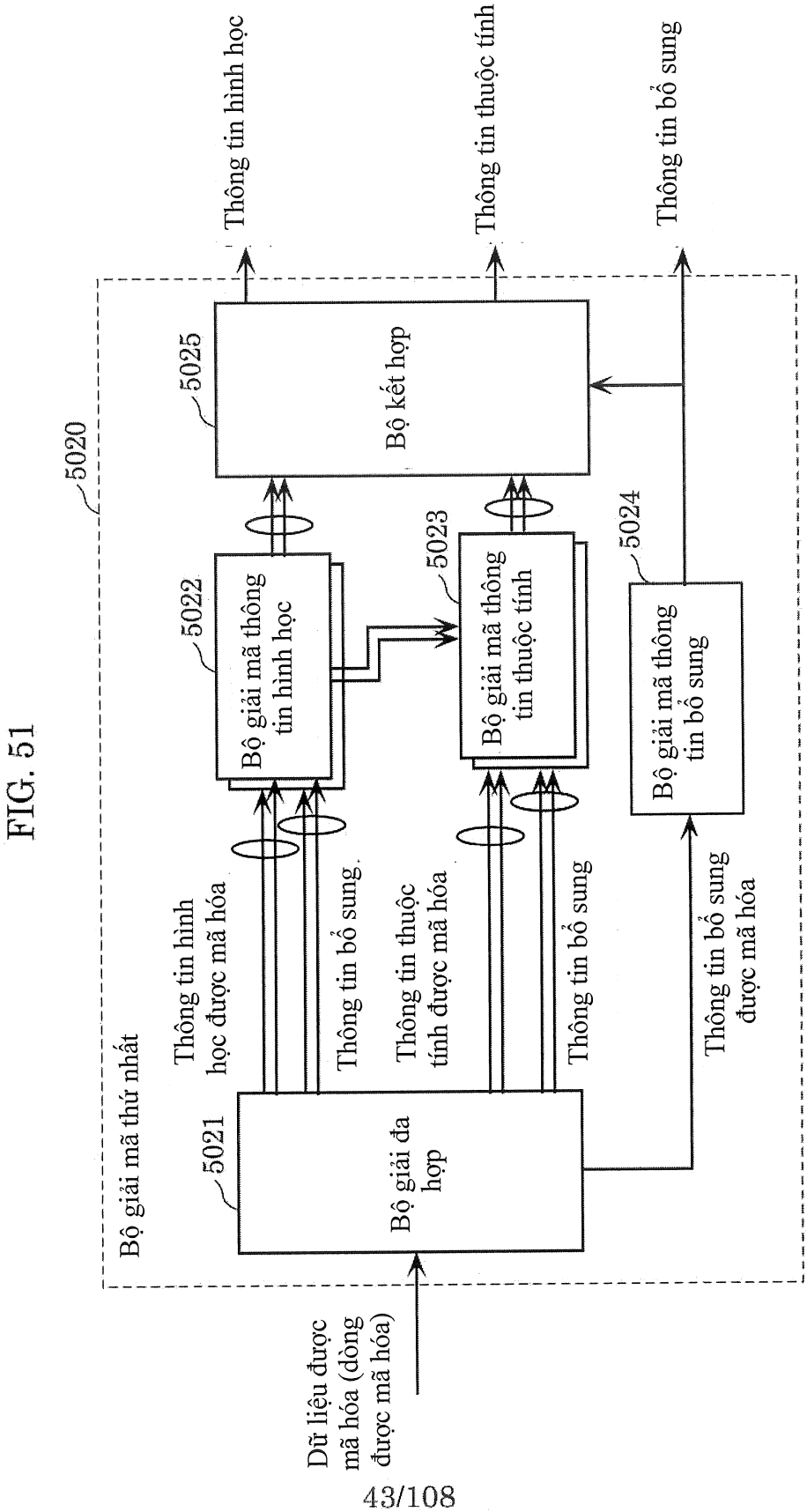

Dữ liệu được mã hóa (dòng được mã hóa)
Bộ giải mã thứ nhất
5020
5021
Bộ giải đa hợp
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
5022
Bộ giải mã thông tin hình học
5023
Bộ giải mã thông tin thuộc tính
5024
Bộ giải mã thông tin bổ sung
5025
Bộ kết hợp
Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung

FIG.52

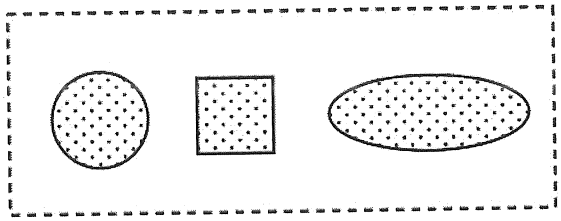


FIG.53

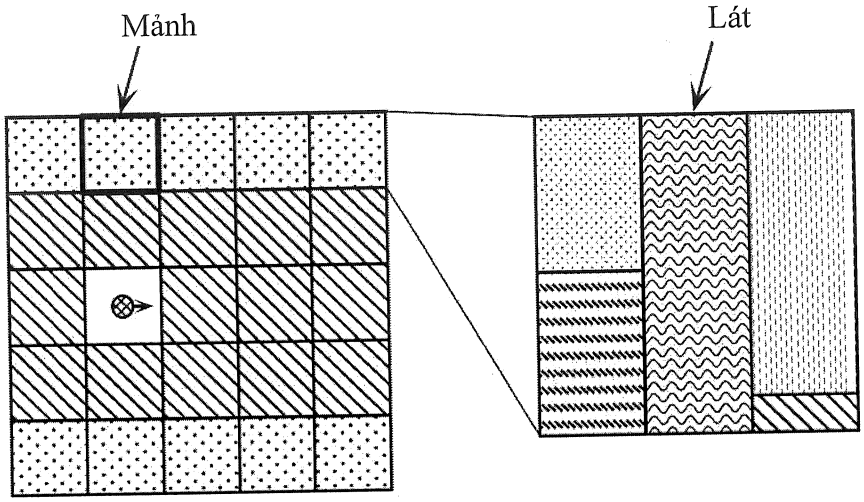


FIG. 54

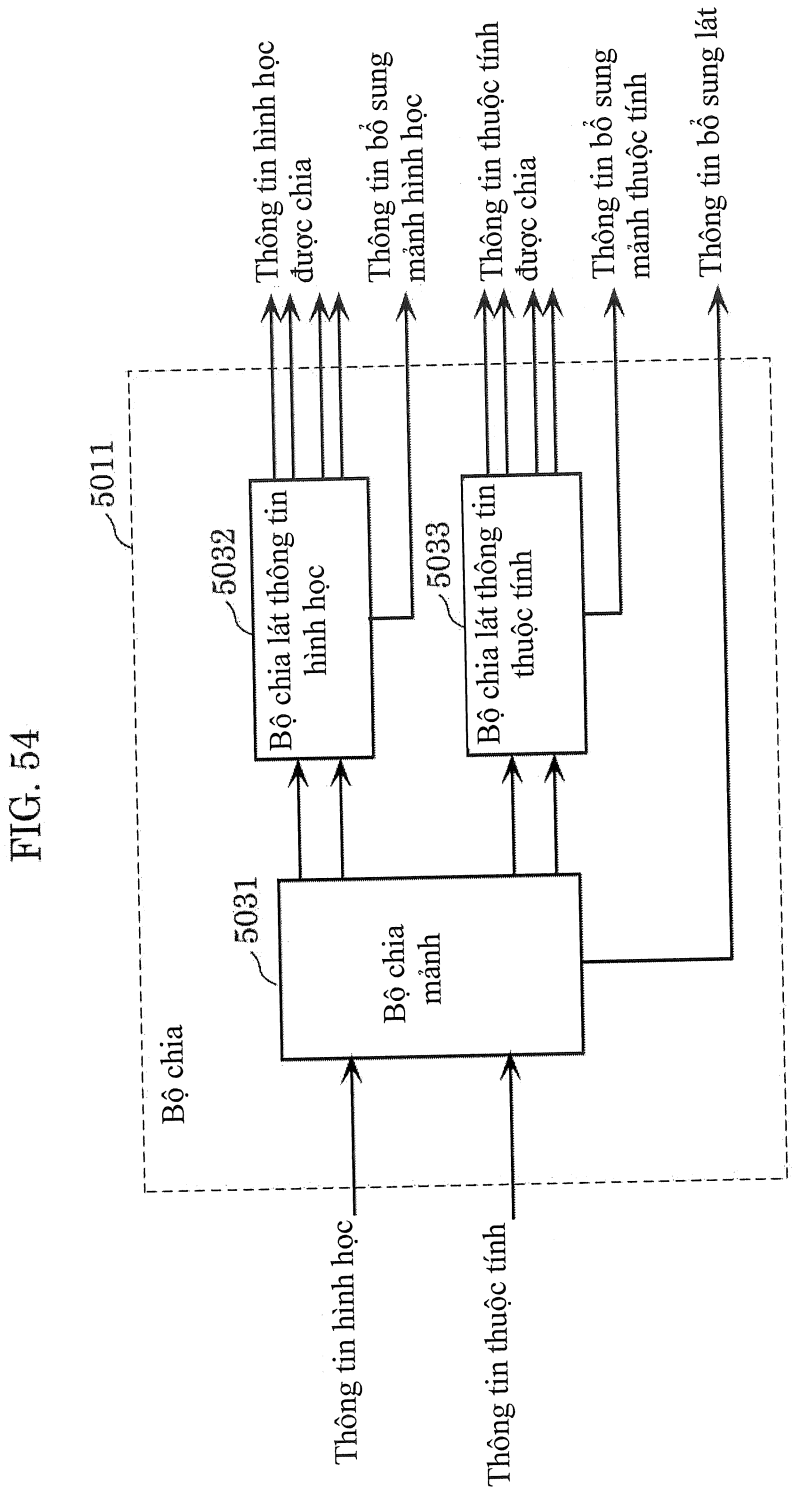

Thông tin hình học
Thông tin thuộc tính
Bộ chia
5011
5031
Bộ chia mảnh
5032
Bộ chia lát thông tin hình học
5033
Bộ chia lát thông tin thuộc tính
Thông tin hình học được chia
Thông tin bổ sung mảnh hình học
Thông tin thuộc tính được chia
Thông tin bổ sung mảnh thuộc tính
Thông tin bổ sung lát

FIG. 55

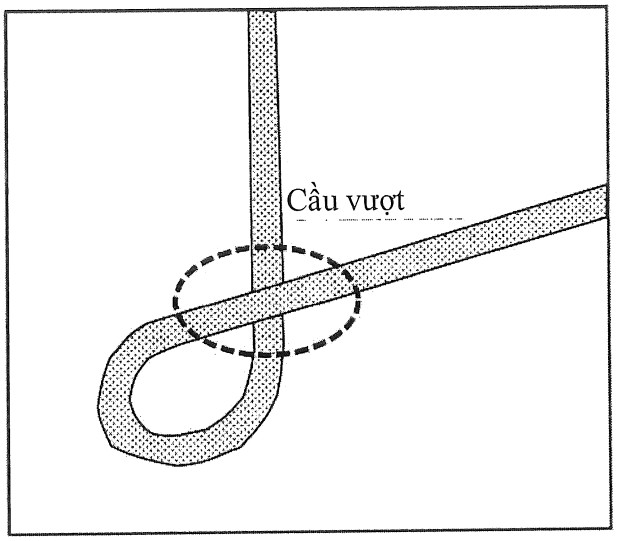


FIG. 56

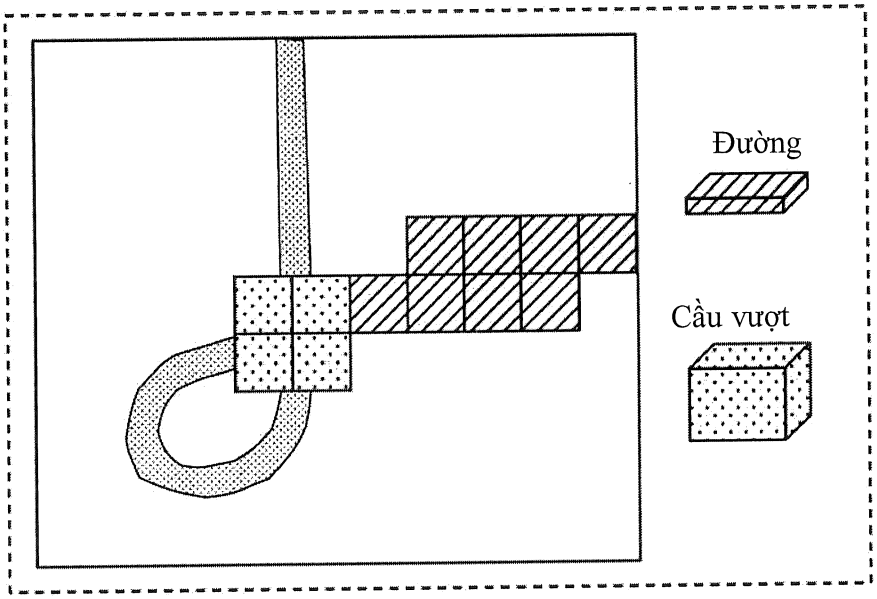


FIG. 57

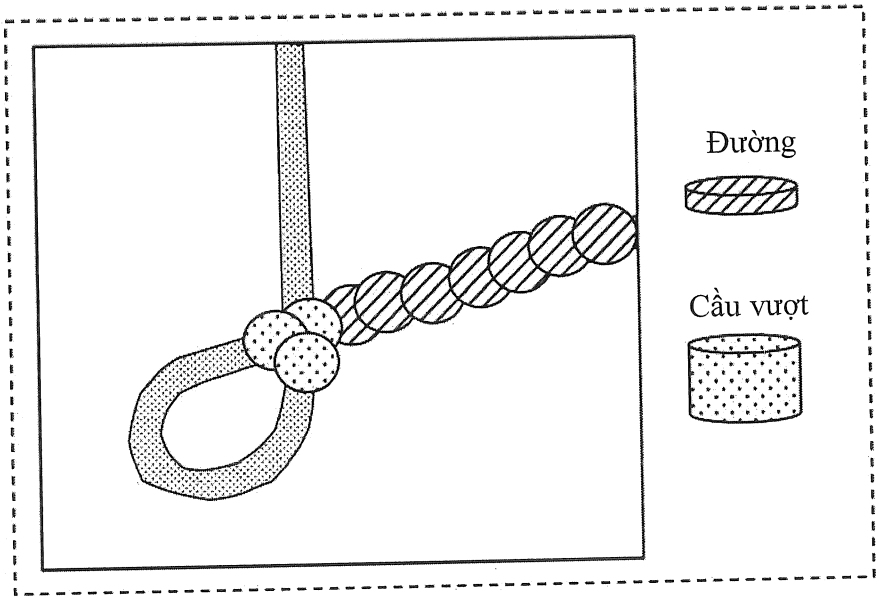


FIG. 58

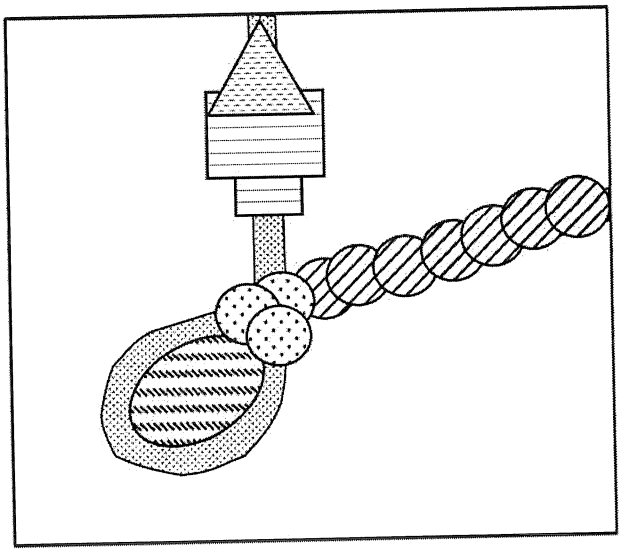


FIG. 59

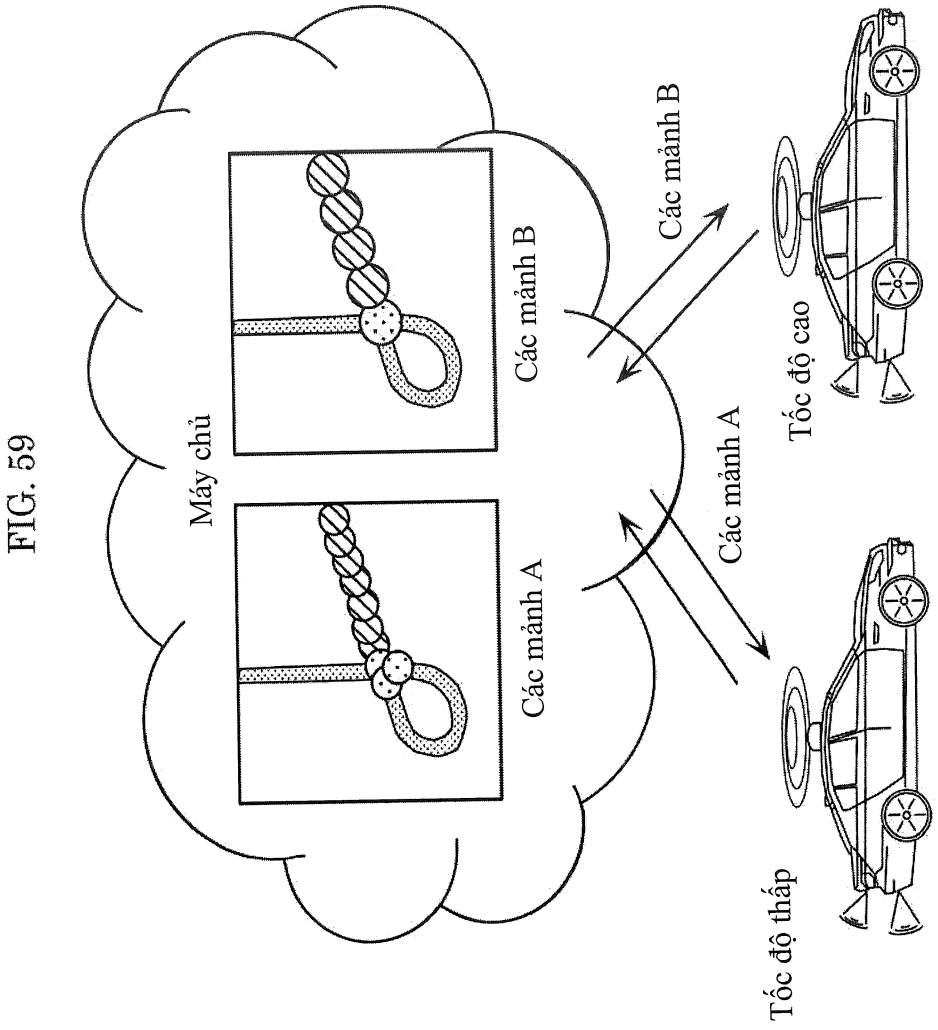

Máy chủ
Các mảnh A
Các mảnh B
Các mảnh B
Các mảnh A
Tốc độ cao
Tốc độ thấp

FIG. 60

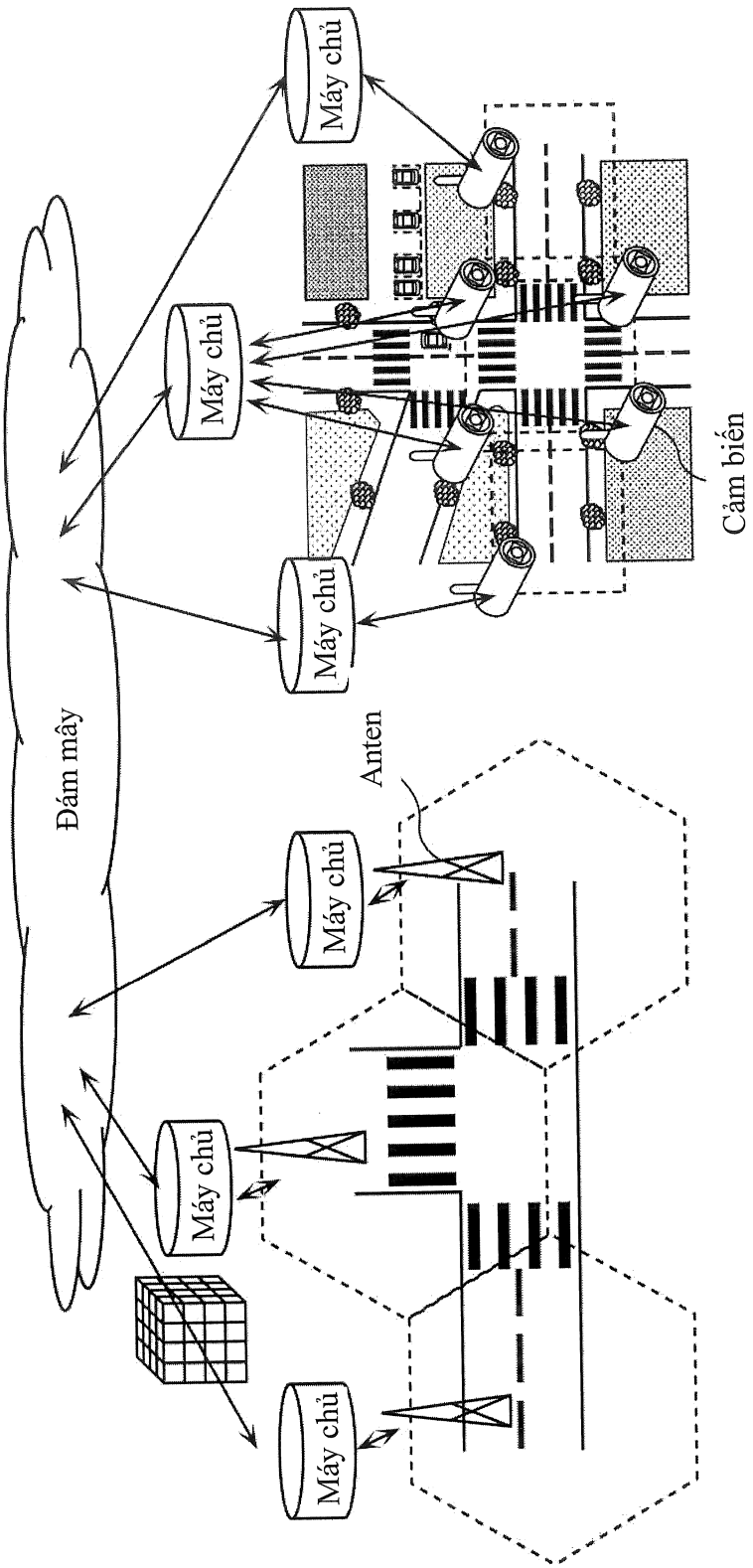

Đám mây
Máy chủ
Máy chủ
Máy chủ
Máy chủ
Máy chủ
Máy chủ
Anten
Cảm biến

FIG. 61

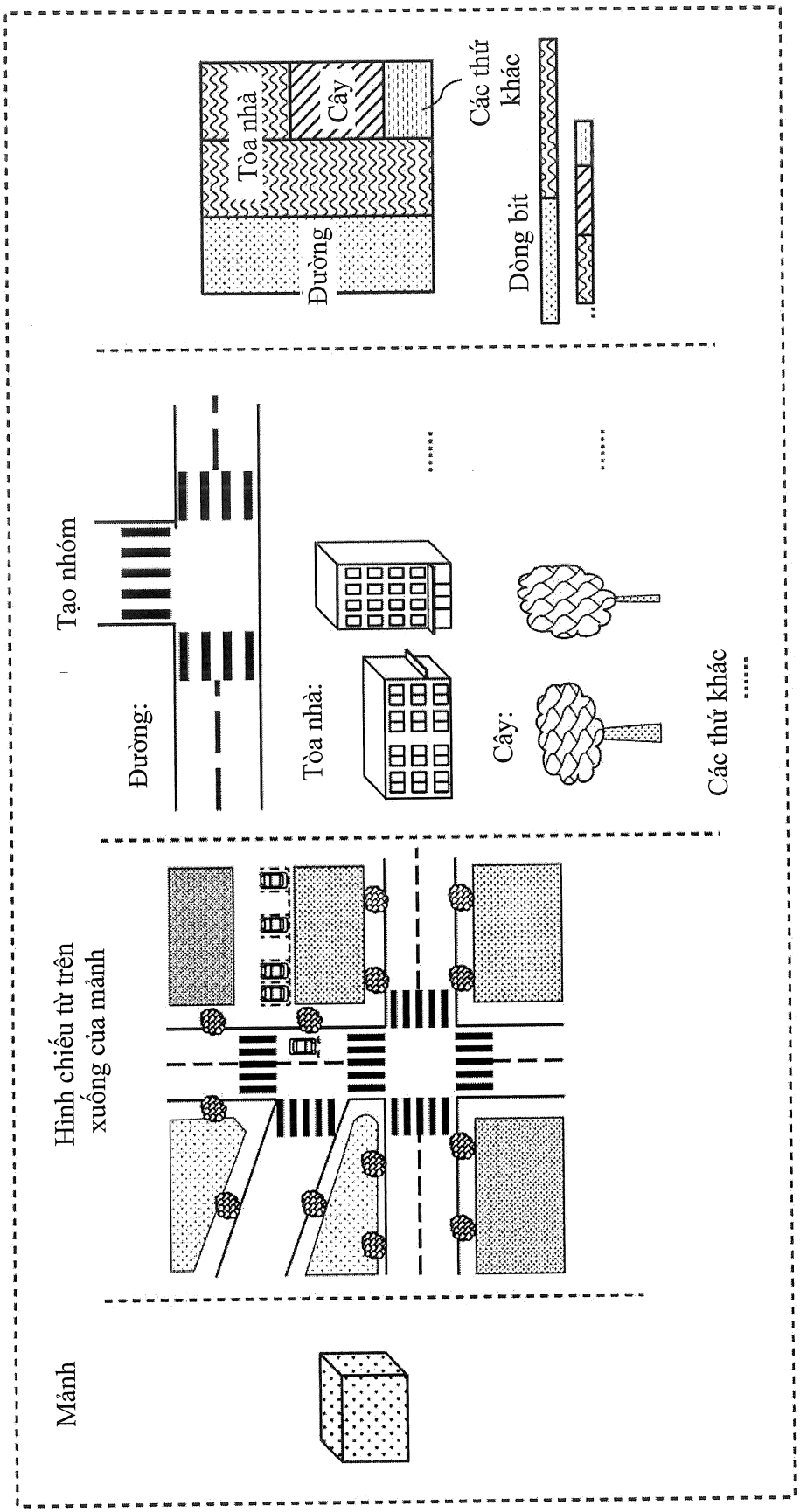

Mảnh
Hình chiếu từ trên xuống của mảnh
Tạo nhóm
Đường:
Tòa nhà:
Cây:
Các thứ khác
.....
Đường
Tòa nhà
Cây
Các thứ khác
Dòng bit

FIG. 62

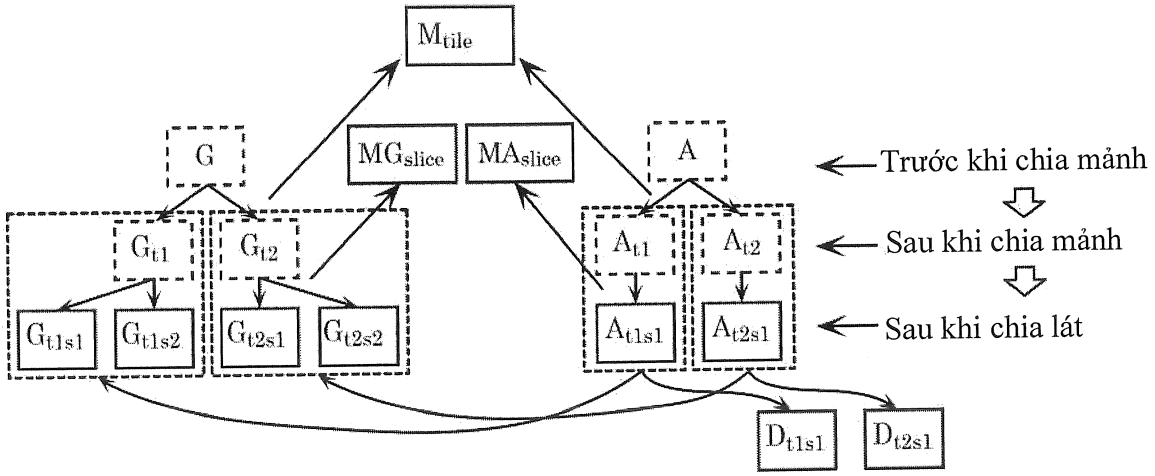


FIG. 63

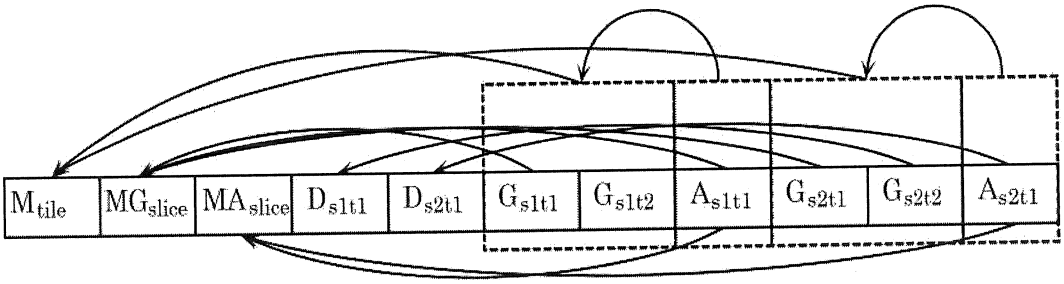


FIG. 64

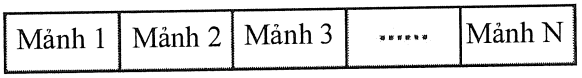


FIG. 65

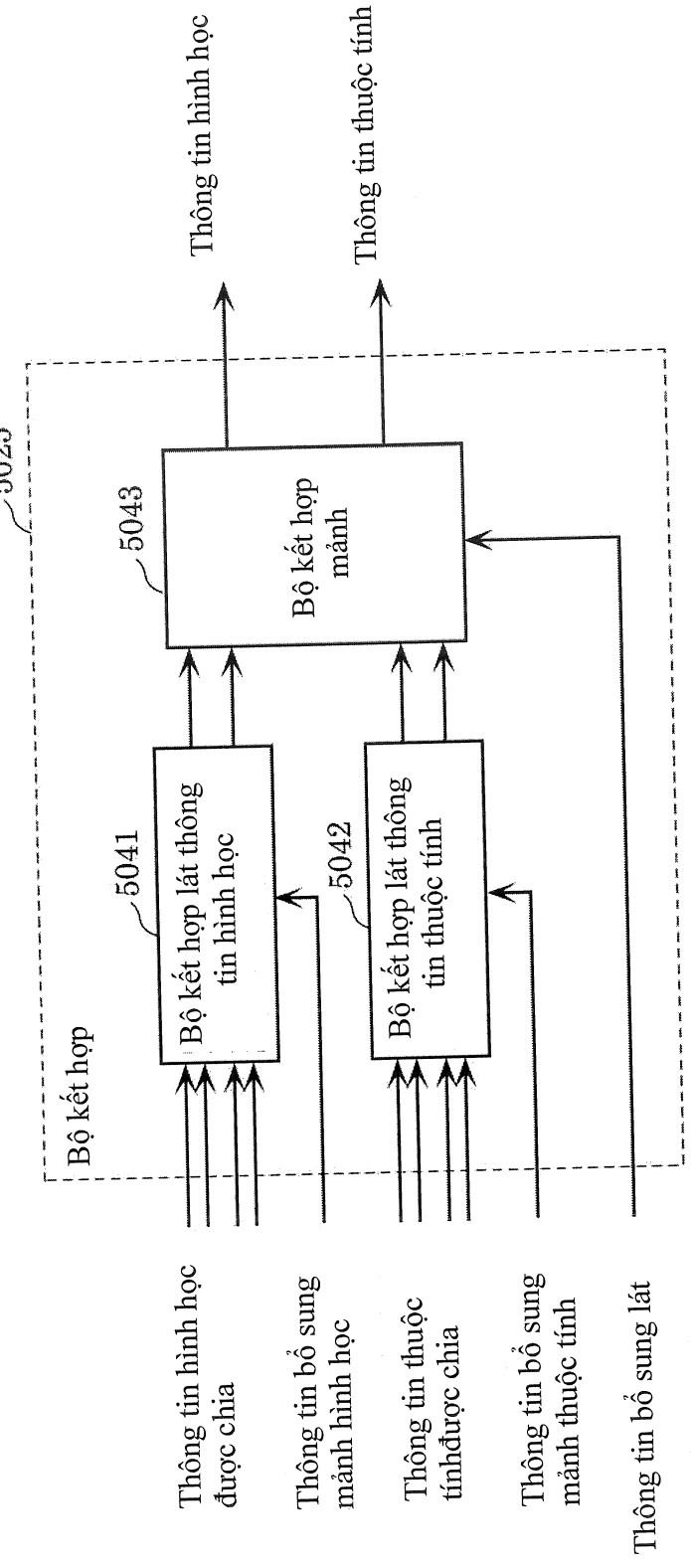

Thông tin hình học
Thông tin thuộc tính
5025
Bộ kết hợp
5043
Bộ kết hợp mảnh
5041
Bộ kết hợp lát thông tin hình học
5042
Bộ kết hợp lát thông tin thuộc tính
Thông tin hình học được chia
Thông tin bổ sung mảnh hình học
Thông tin thuộc tínhđược chia
Thông tin bổ sung mảnh thuộc tính
Thông tin bổ sung lát

FIG. 66

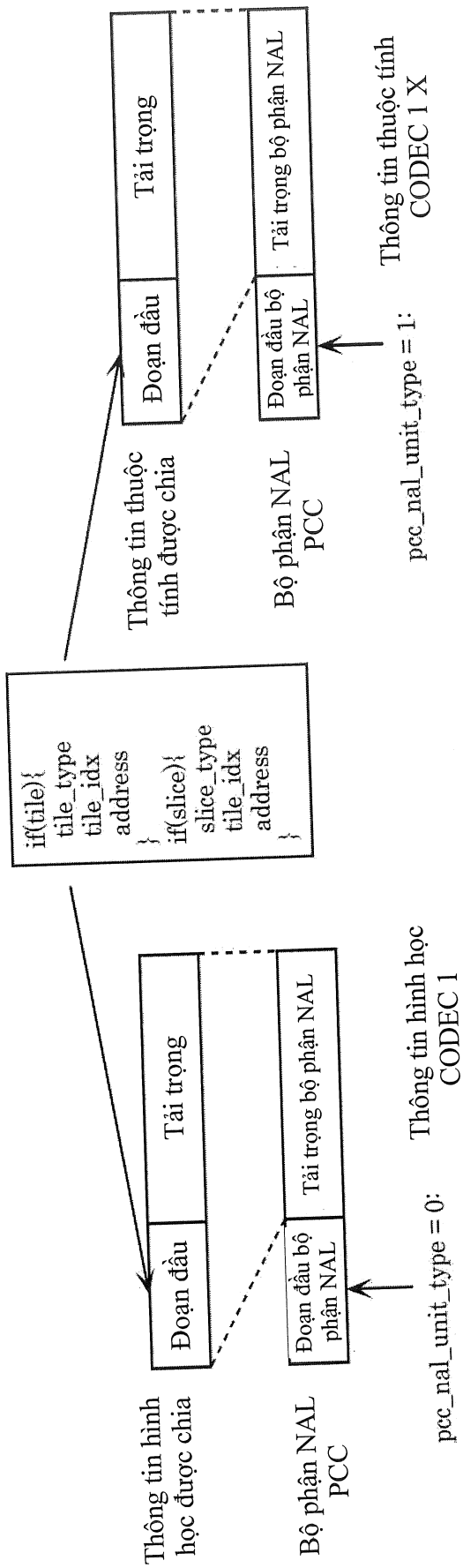

if(tile){
tile_type
tile_idx
address
}
if(slice){
slice_type
tile_idx
address
}
Thông tin hình học được chia
Đoạn đầu
Tải trọng
Bộ phận NAL PCC
Đoạn đầu bộ phận NAL
Tải trọng bộ phận NAL
pcc_nal_unit_type = 0:
Thông tin hình học CODEC 1
Thông tin thuộc tính được chia
Đoạn đầu
Tải trọng
Bộ phận NAL PCC
Đoạn đầu bộ phận NAL
Tải trọng bộ phận NAL
pcc_nal_unit_type = 1:
Thông tin thuộc tính CODEC 1 X

FIG. 67

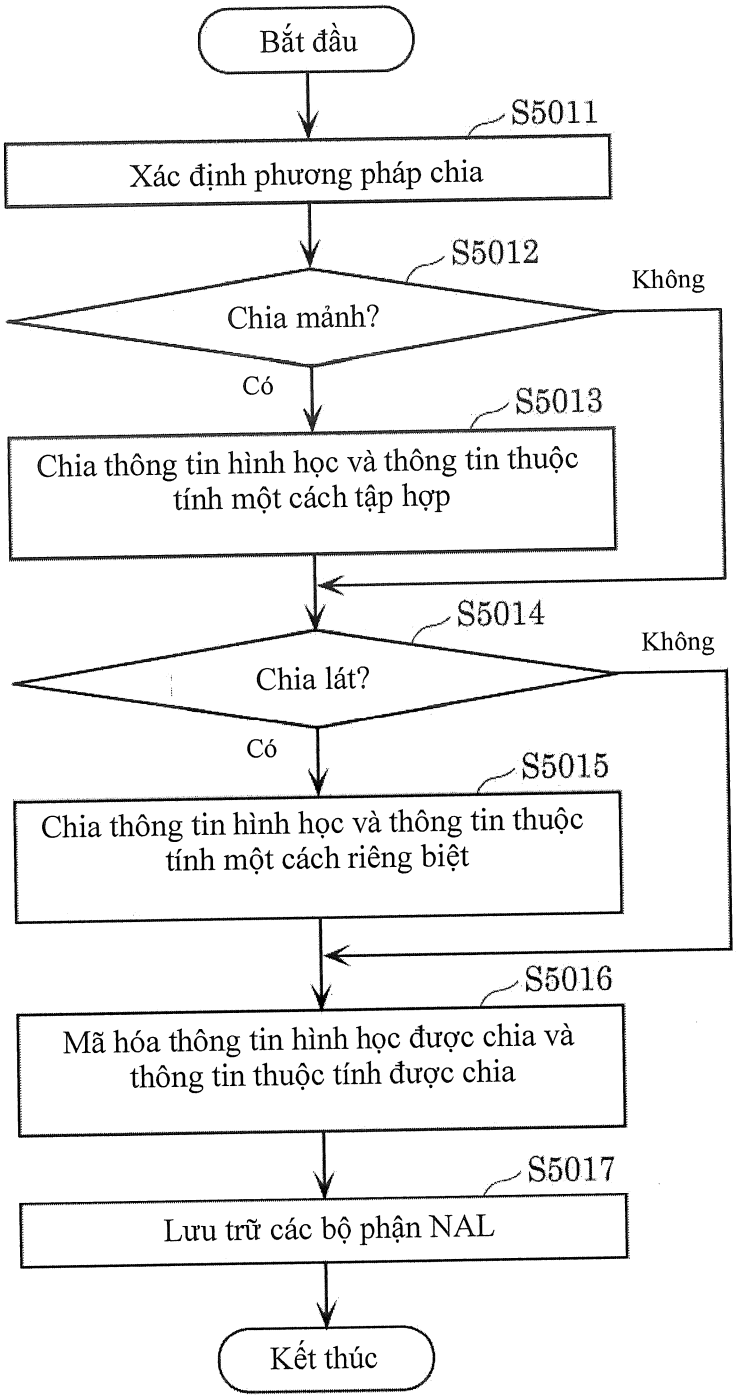

Bắt đầu
S5011
Xác định phương pháp chia
S5012
Chia mảnh?
Không
Có
S5013
Chia thông tin hình học và thông tin thuộc tính một cách tập hợp
S5014
Chia lát?
Không
Có
S5015
Chia thông tin hình học và thông tin thuộc tính một cách riêng biệt
S5016
Mã hóa thông tin hình học được chia và thông tin thuộc tính được chia
S5017
Lưu trữ các bộ phận NAL
Kết thúc

FIG. 68

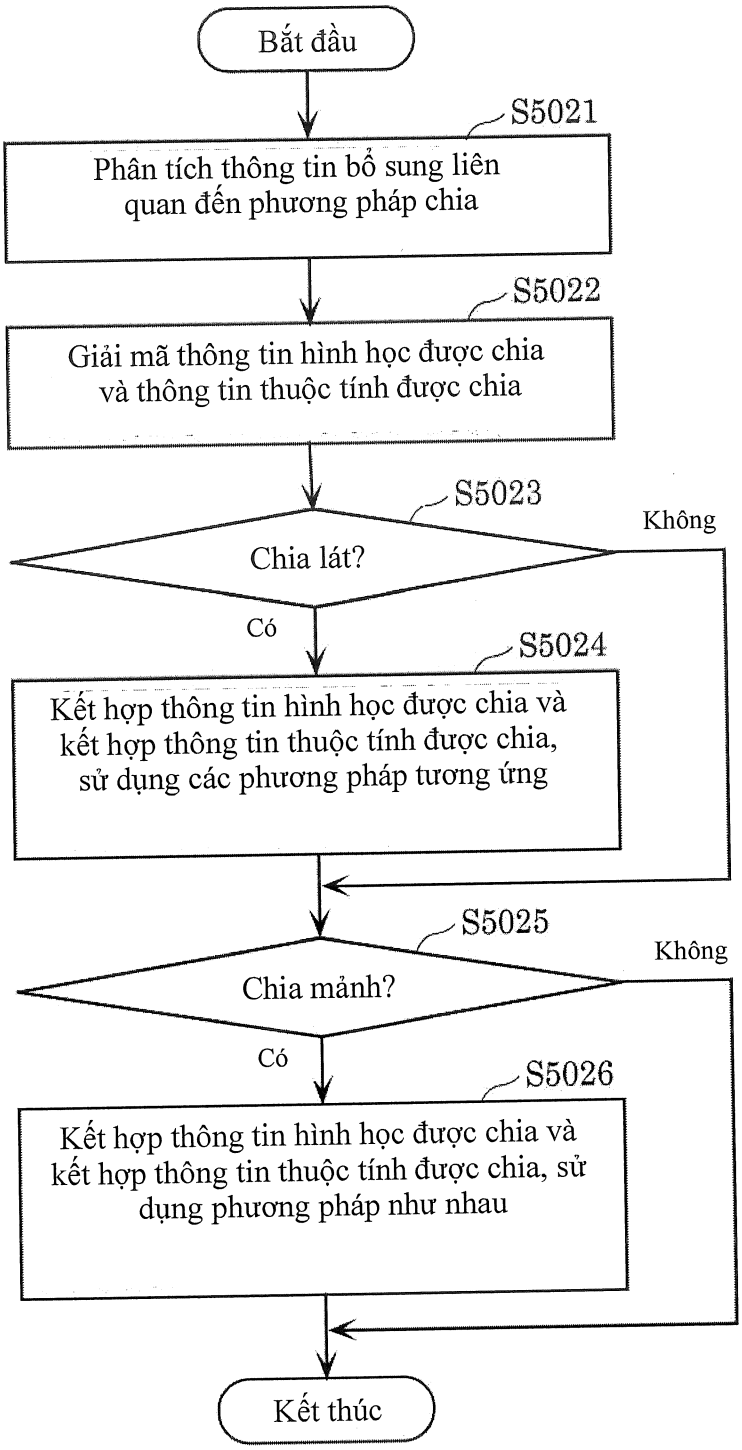

Bắt đầu
S5021
Phân tích thông tin bổ sung liên quan đến phương pháp chia
S5022
Giải mã thông tin hình học được chia và thông tin thuộc tính được chia
S5023
Chia lát?
Không
Có
S5024
Kết hợp thông tin hình học được chia và kết hợp thông tin thuộc tính được chia, sử dụng các phương pháp tương ứng
S5025
Chia mảnh?
Không
Có
S5026
Kết hợp thông tin hình học được chia và kết hợp thông tin thuộc tính được chia, sử dụng phương pháp như nhau
Kết thúc

FIG. 69

```
TileMetaData{
  type_of_divide  1:top_view, 2:other
  if(type_of_divide == 1){  //top_view
    topview_shape    //1: square, 2:circle
    tile_overlap_flag   //1:true, 2:false
    if(tile_overlap_flag){
      type_of_overlap
    }
    independent_tile_height_flag
    if(independent_tile_height_flag){
      tile_height //1: road, flyover
    }

    tile_number

    for(tile_number){
      global_position or relative_position
      if(independent_tile_height_flag){
        tile_height
      }
    }
  }
}
```

FIG. 70

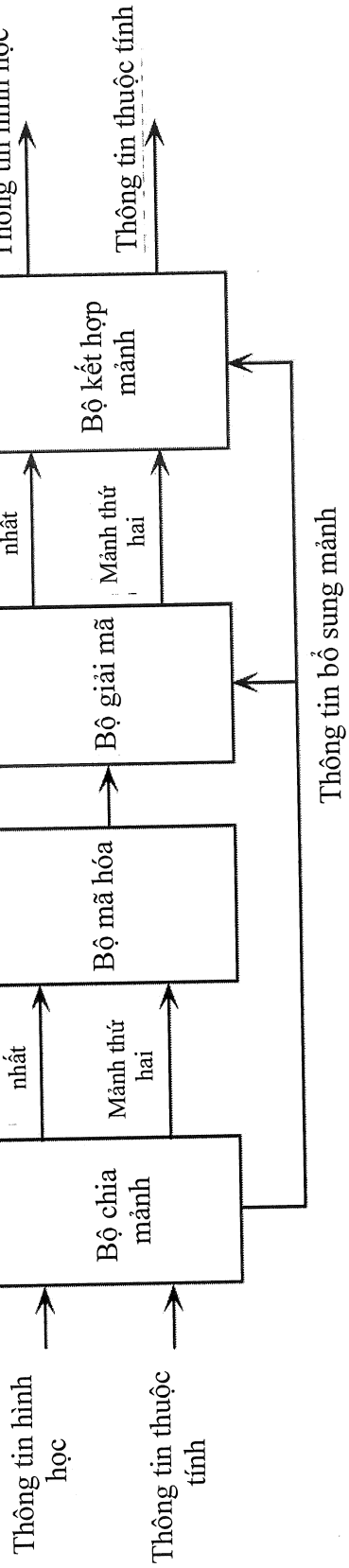

Thông tin hình học
Thông tin thuộc tính
5051
Bộ chia mảnh
Mảnh thứ nhất
Mảnh thứ hai
5052
Bộ mã hóa
5053
Bộ giải mã
Mảnh thứ nhất
Mảnh thứ hai
5054
Bộ kết hợp mảnh
Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung mảnh

FIG. 71

```
SliceMetaData{
   type_of_divide   1:object, 2:other
   if(type_of_divide == 1) {   //object
      slice_overlap_flag     //1:true, 2:false
      if(slice_overlap_flag){
         type_of_overlap
      }

      slice_number

      for(slice_number){
         global_position or relative_position
         slice_bounding_box_size
       }
   }
}
```

FIG. 72

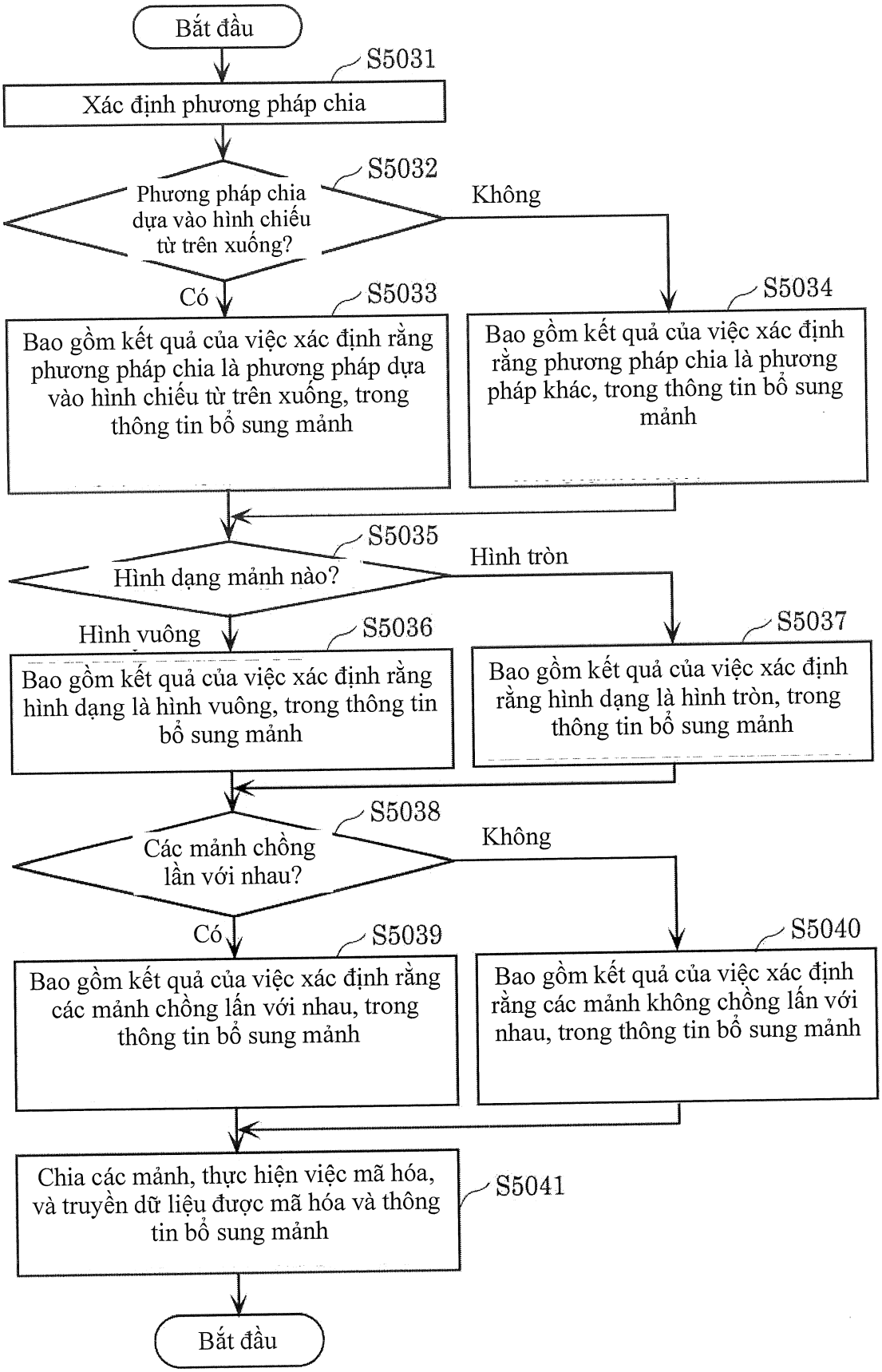

Bắt đầu
S5031
Xác định phương pháp chia
S5032
Phương pháp chia dựa vào hình chiếu từ trên xuống?
Không
Có
S5033
Bao gồm kết quả của việc xác định rằng phương pháp chia là phương pháp dựa vào hình chiếu từ trên xuống, trong thông tin bổ sung mảnh
S5034
Bao gồm kết quả của việc xác định rằng phương pháp chia là phương pháp khác, trong thông tin bổ sung mảnh
S5035
Hình dạng mảnh nào?
Hình tròn
Hình vuông
S5036
Bao gồm kết quả của việc xác định rằng hình dạng là hình vuông, trong thông tin bổ sung mảnh
S5037
Bao gồm kết quả của việc xác định rằng hình dạng là hình tròn, trong thông tin bổ sung mảnh
S5038
Các mảnh chồng lấn với nhau?
Không
Có
S5039
Bao gồm kết quả của việc xác định rằng các mảnh chồng lấn với nhau, trong thông tin bổ sung mảnh
S5040
Bao gồm kết quả của việc xác định rằng các mảnh không chồng lấn với nhau, trong thông tin bổ sung mảnh
Chia các mảnh, thực hiện việc mã hóa, và truyền dữ liệu được mã hóa và thông tin bổ sung mảnh
S5041
Bắt đầu

FIG. 73

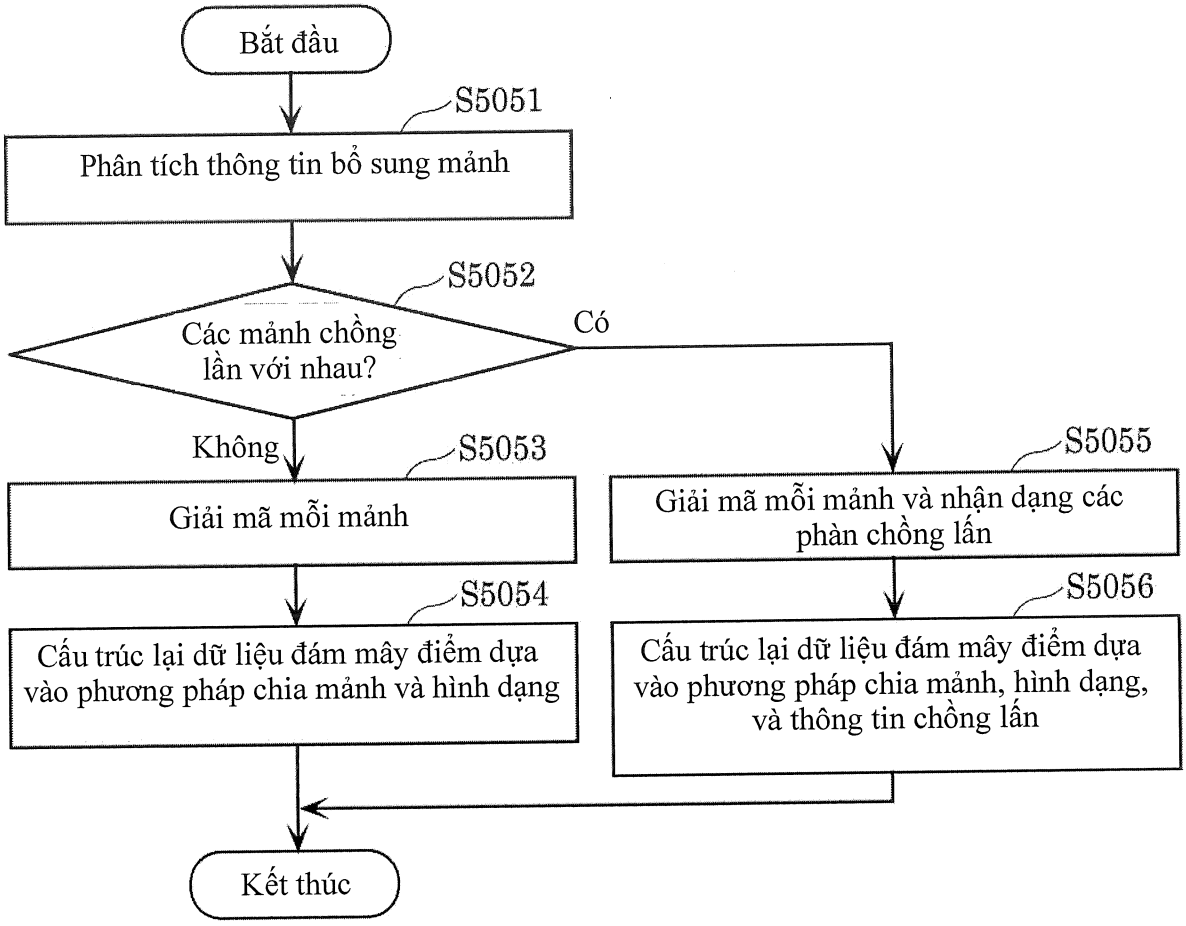

Bắt đầu
S5051
Phân tích thông tin bổ sung mảnh
S5052
Các mảnh chồng lấn với nhau?
Có
Không
S5053
Giải mã mỗi mảnh
S5054
Cấu trúc lại dữ liệu đám mây điểm dựa vào phương pháp chia mảnh và hình dạng
S5055
Giải mã mỗi mảnh và nhận dạng các phần chồng lấn
S5056
Cấu trúc lại dữ liệu đám mây điểm dựa vào phương pháp chia mảnh, hình dạng, và thông tin chồng lấn
Kết thúc

FIG. 74

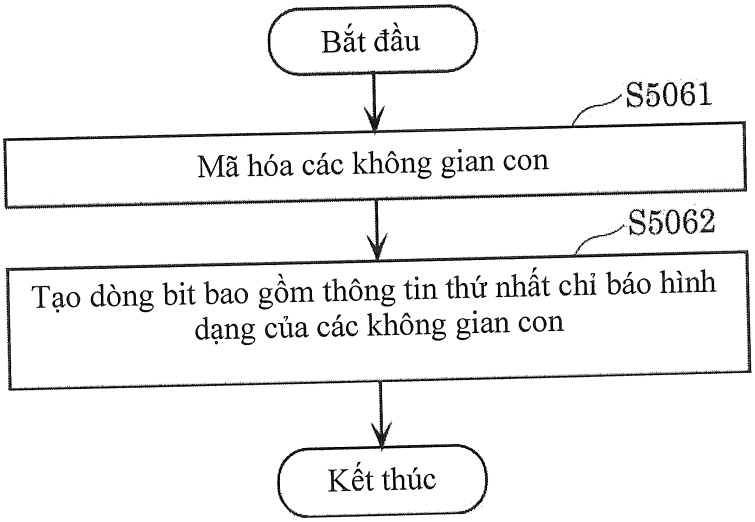

Bắt đầu
S5061
Mã hóa các không gian con
S5062
Tạo dòng bit bao gồm thông tin thứ nhất chỉ báo hình dạng của các không gian con
Kết thúc

FIG. 75

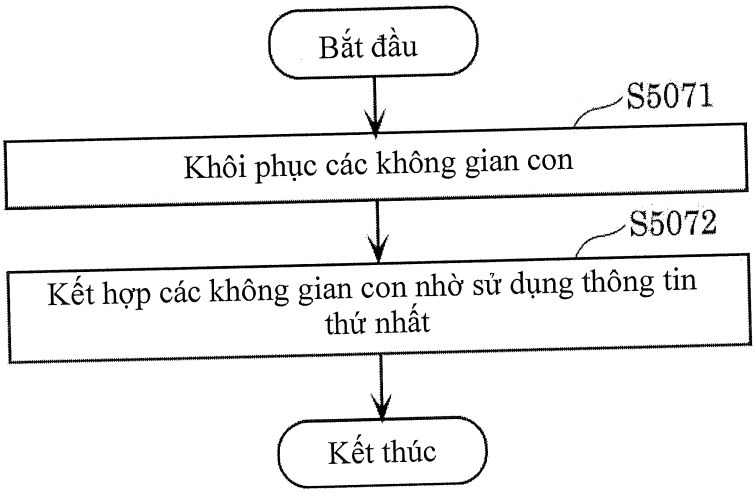

Bắt đầu
S5071
Khôi phục các không gian con
S5072
Kết hợp các không gian con nhờ sử dụng thông tin thứ nhất
Kết thúc

FIG. 76

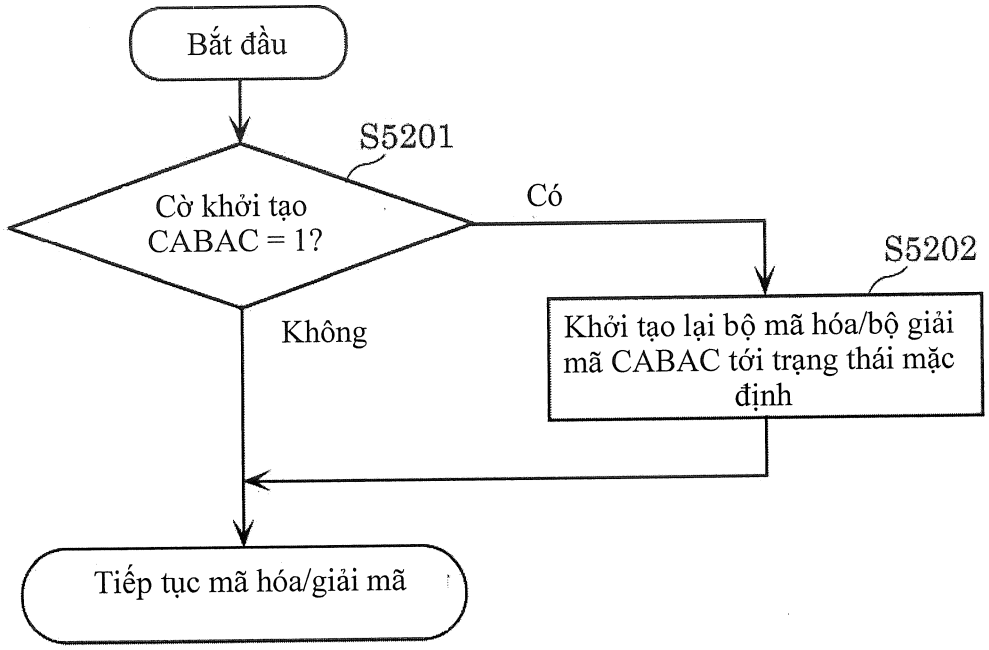

Bắt đầu
S5201
Cờ khởi tạo CABAC = 1?
Có
S5202
Khởi tạo lại bộ mã hóa/bộ giải mã CABAC tới trạng thái mặc định
Không
Tiếp tục mã hóa/giải mã

FIG. 77

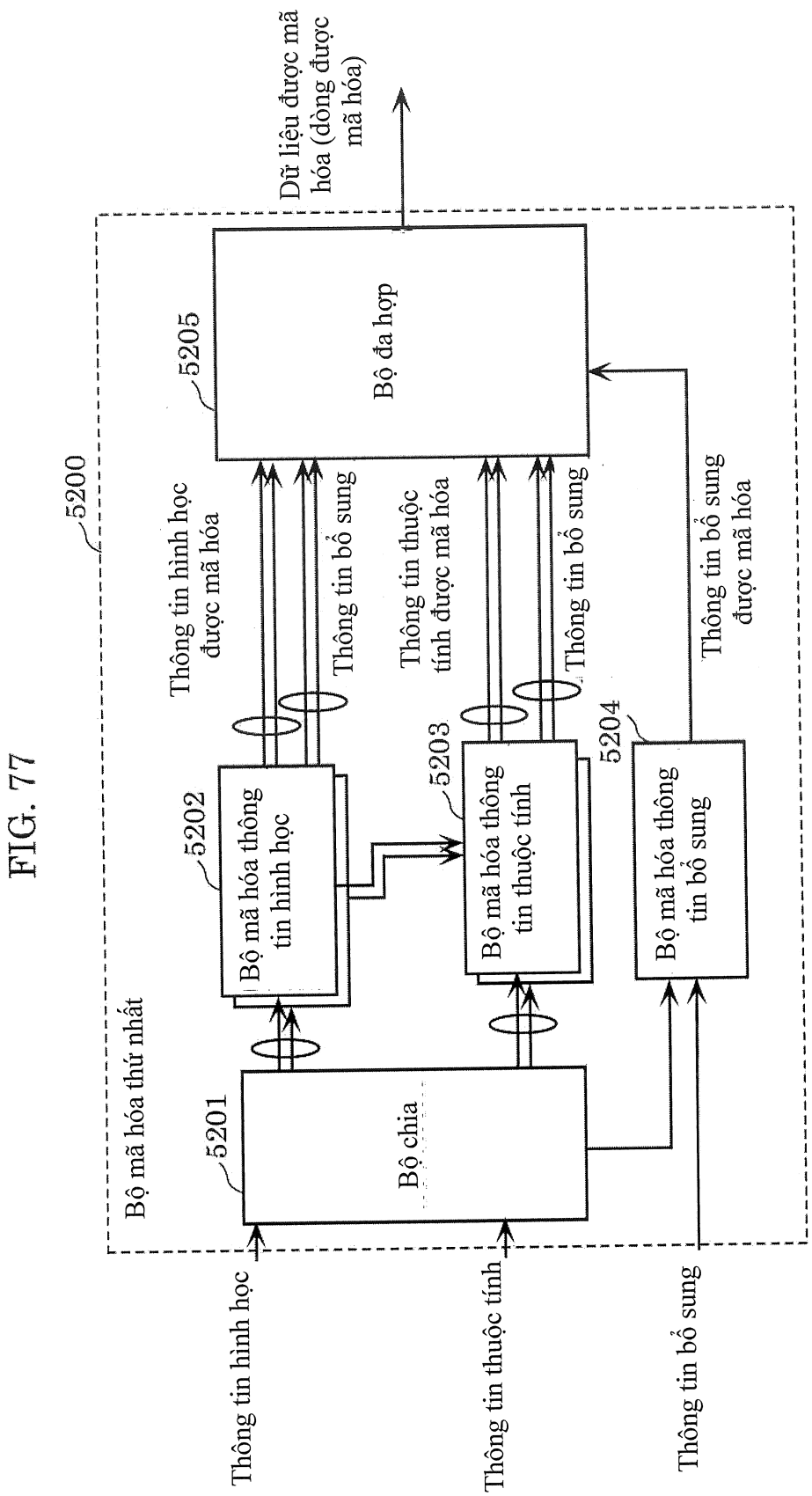

Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung
Bộ mã hóa thứ nhất
5200
Bộ chia
5201
Bộ mã hóa thông tin hình học
5202
Bộ mã hóa thông tin thuộc tính
5203
Bộ mã hóa thông tin bổ sung
5204
Bộ đa hợp
5205
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
Dữ liệu được mã hóa (dòng được mã hóa)

FIG. 78

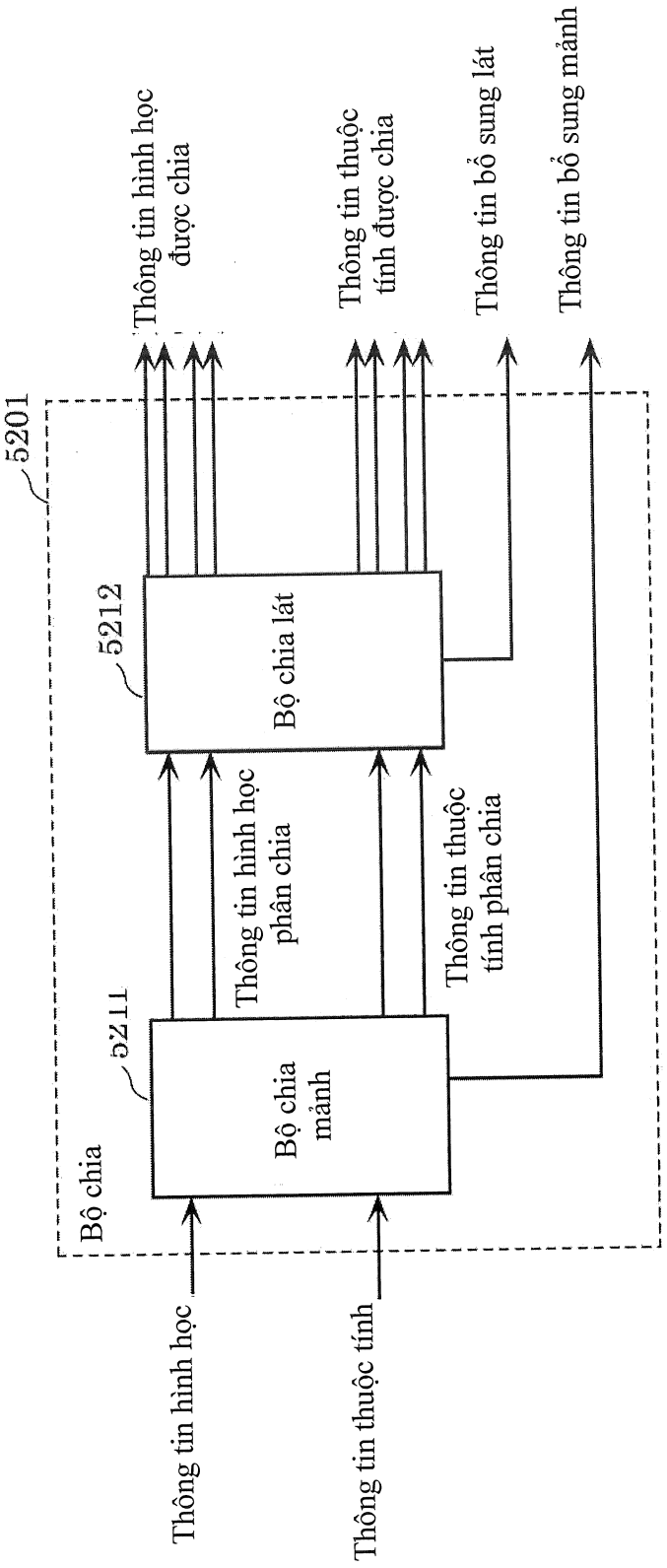

Thông tin hình học
Thông tin thuộc tính
Bộ chia
5201
Bộ chia mảnh
5211
Bộ chia lát
5212
Thông tin hình học phân chia
Thông tin thuộc tính phân chia
Thông tin hình học được chia
Thông tin thuộc tính được chia
Thông tin bổ sung lát
Thông tin bổ sung mảnh

FIG. 79

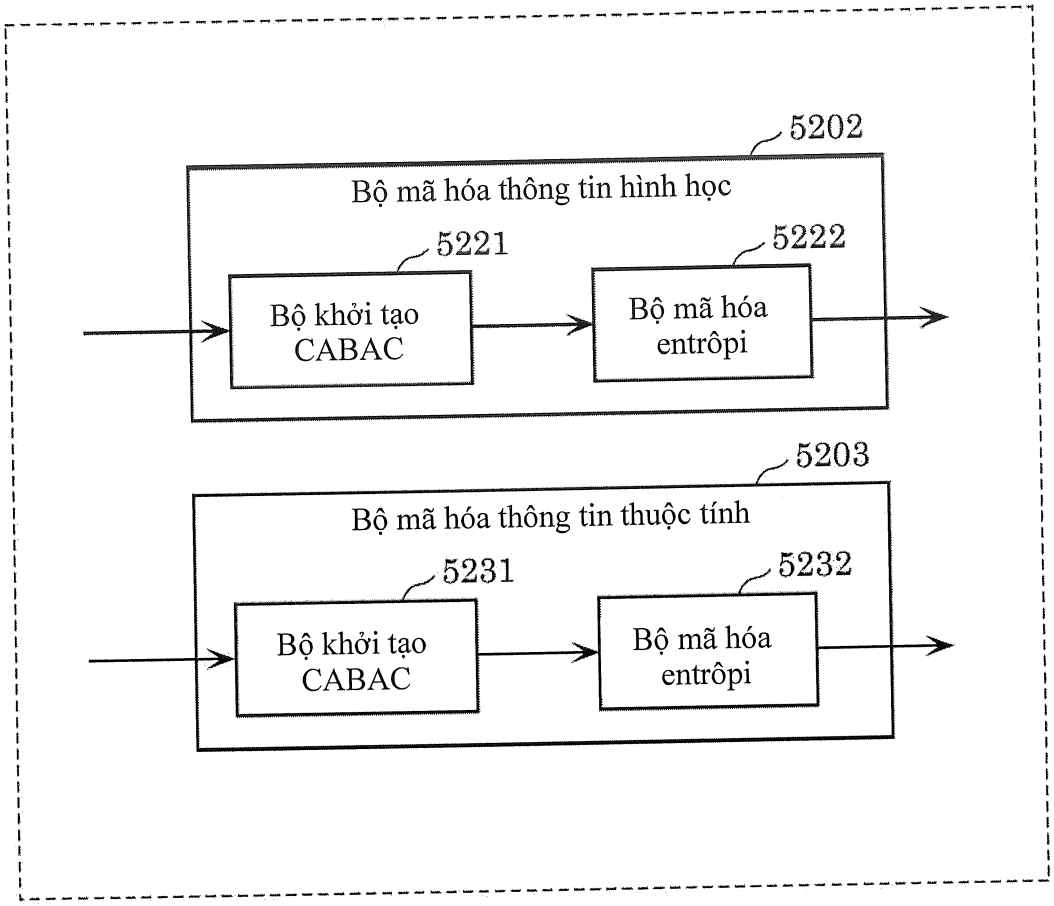

5202
Bộ mã hóa thông tin hình học
5221
5222
Bộ khởi tạo CABAC
Bộ mã hóa entrôpi
5203
Bộ mã hóa thông tin thuộc tính
5231
5232
Bộ khởi tạo CABAC
Bộ mã hóa entrôpi

FIG. 80

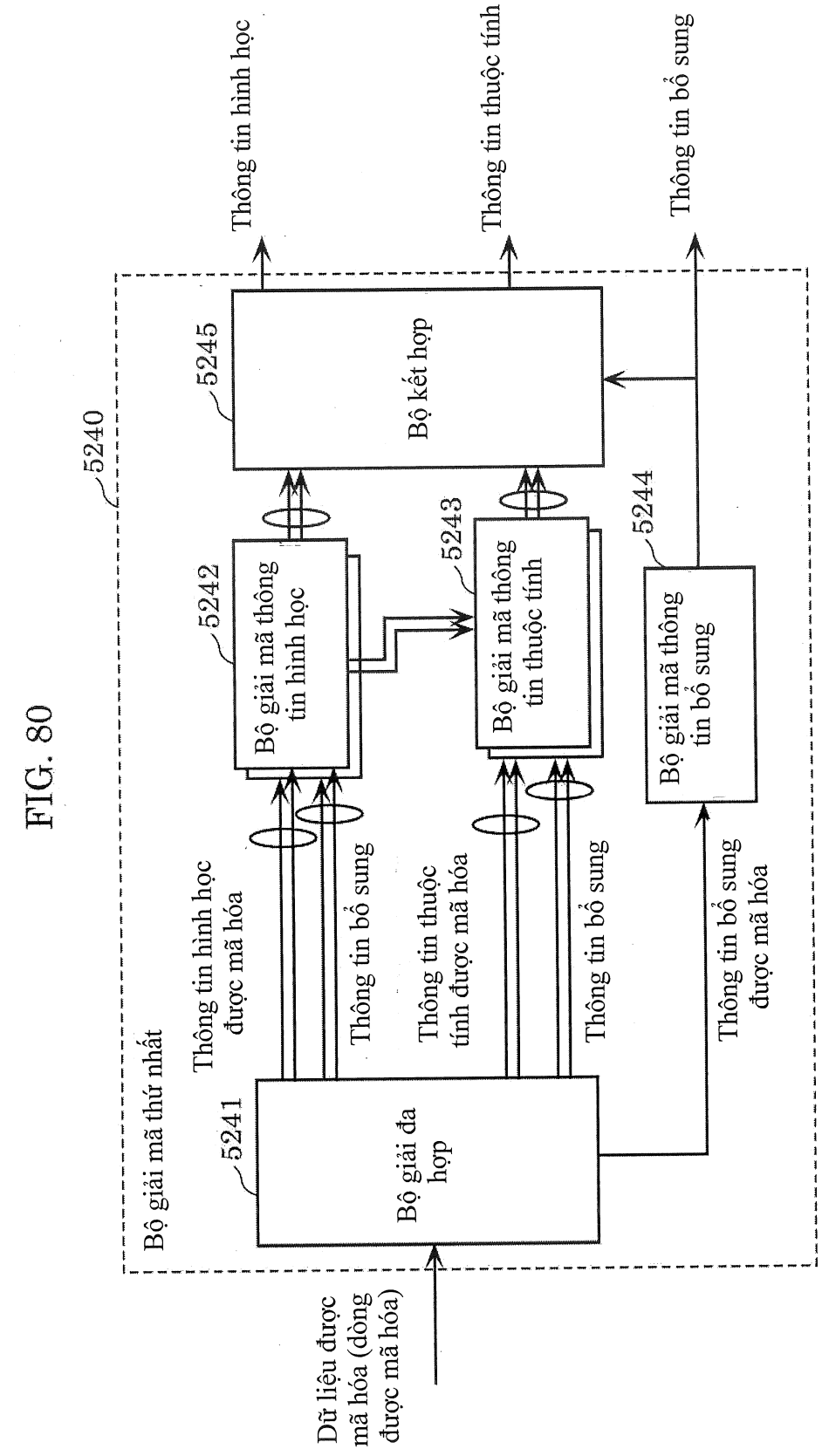

Dữ liệu được mã hóa (dòng được mã hóa)
Bộ giải mã thứ nhất
5240
Bộ giải đa hợp
5241
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
Bộ giải mã thông tin hình học
5242
Bộ giải mã thông tin thuộc tính
5243
Bộ giải mã thông tin bổ sung
5244
Bộ kết hợp
5245
Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung

FIG. 81

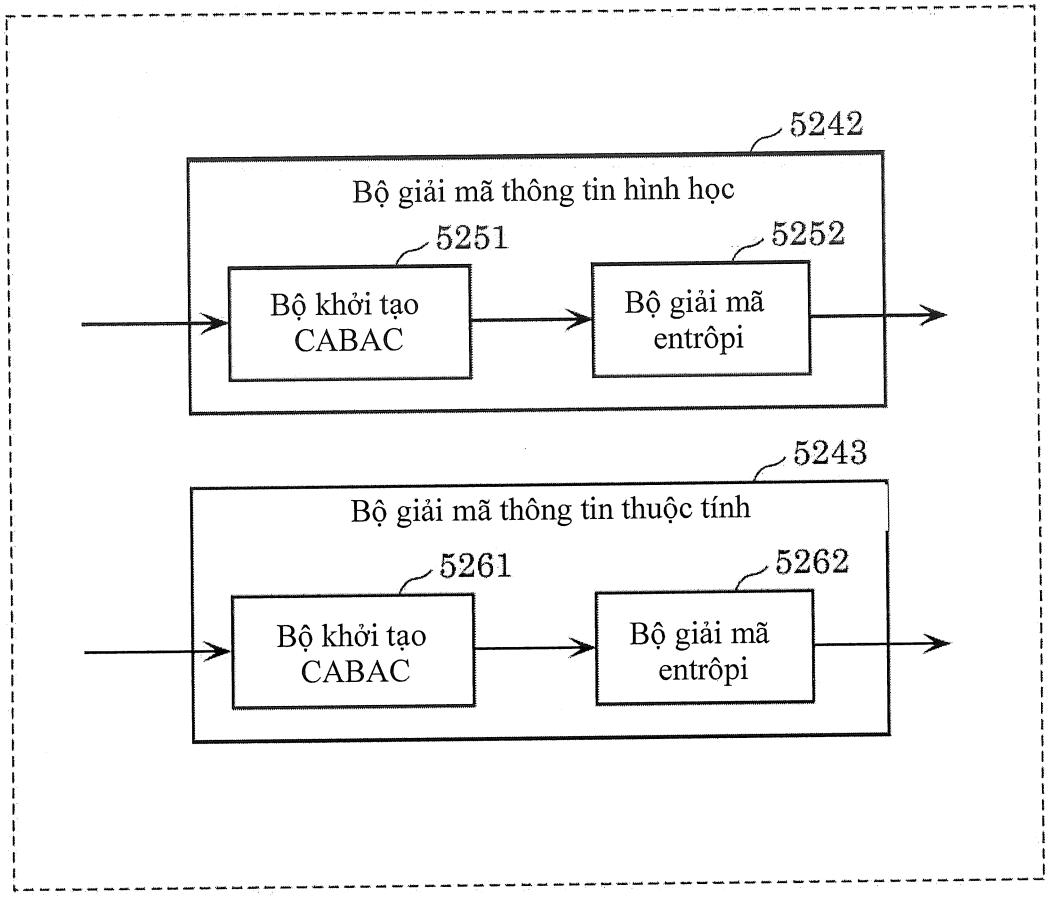

5242
Bộ giải mã thông tin hình học
5251
Bộ khởi tạo CABAC
5252
Bộ giải mã entrôpi
5243
Bộ giải mã thông tin thuộc tính
5261
Bộ khởi tạo CABAC
5262
Bộ giải mã entrôpi

FIG. 82

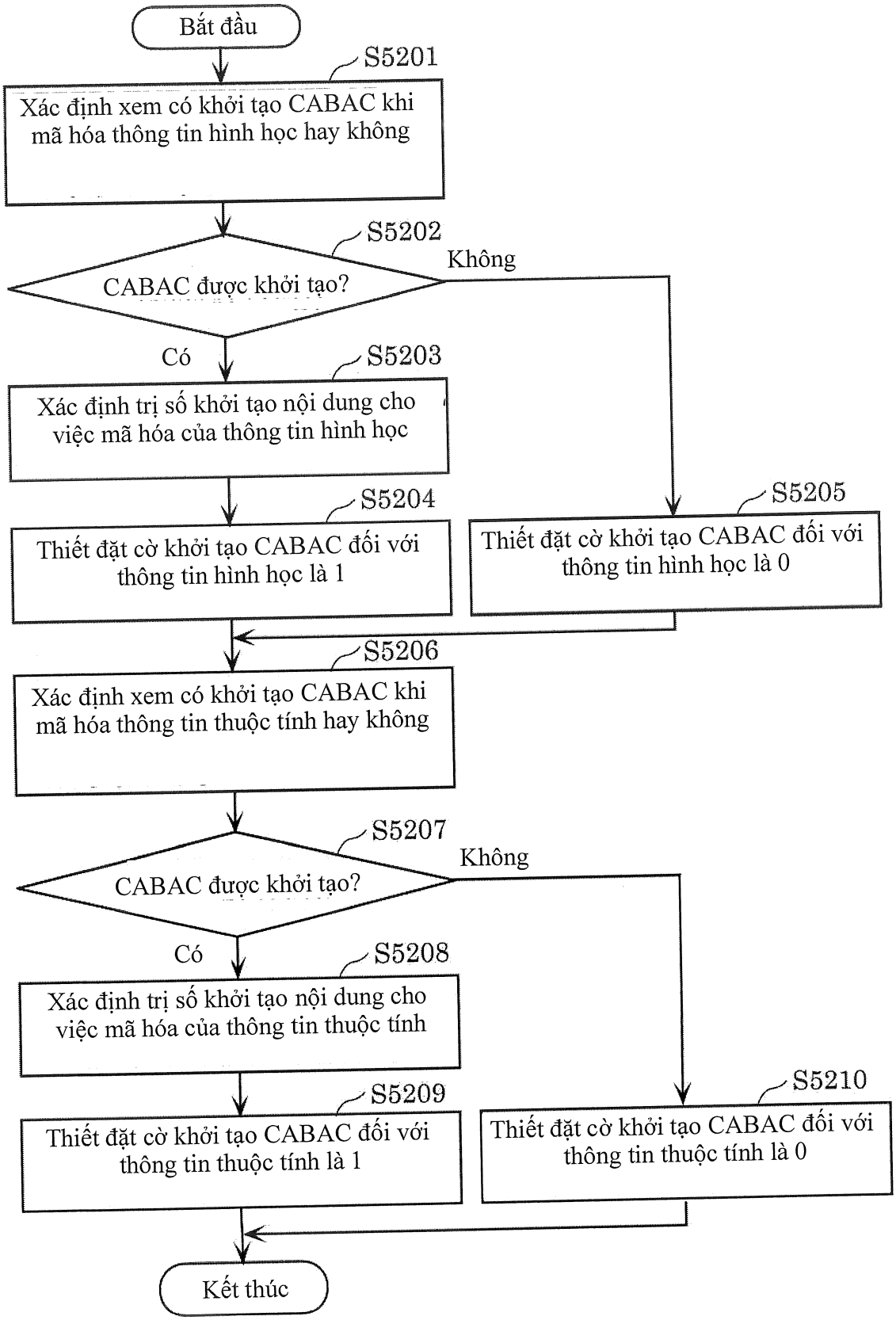

Bắt đầu
S5201
Xác định xem có khởi tạo CABAC khi mã hóa thông tin hình học hay không
S5202
CABAC được khởi tạo?
Không
Có
S5203
Xác định trị số khởi tạo nội dung cho việc mã hóa của thông tin hình học
S5204
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1
S5205
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0
S5206
Xác định xem có khởi tạo CABAC khi mã hóa thông tin thuộc tính hay không
S5207
CABAC được khởi tạo?
Không
Có
S5208
Xác định trị số khởi tạo nội dung cho việc mã hóa của thông tin thuộc tính
S5209
Thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1
S5210
Thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0
Kết thúc

FIG.83

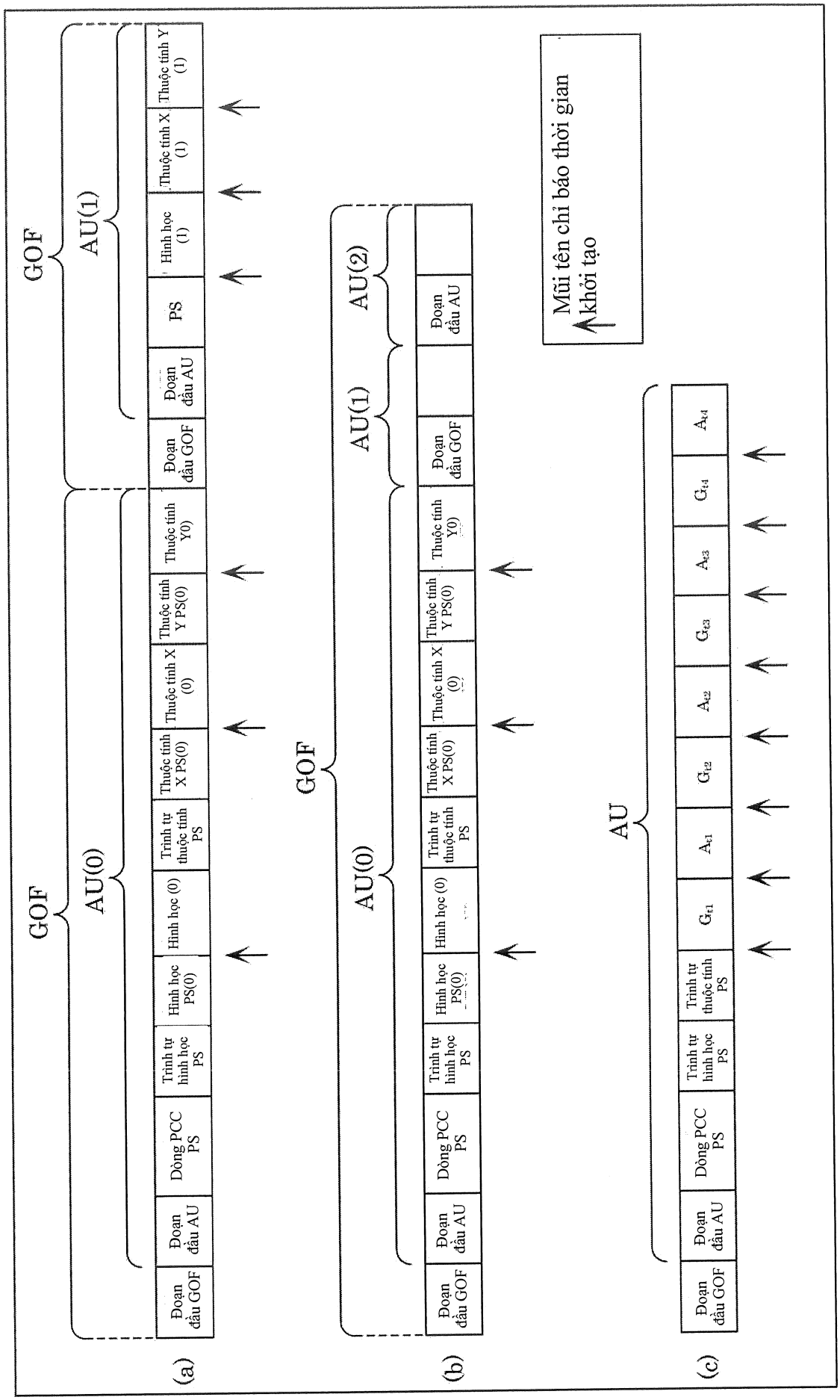

(a)
GOF
AU(0)
AU(1)
Đoạn đầu GOF
Đoạn đầu AU
Dòng PCC PS
Trình tự hình học PS
Hình học PS(0)
Hình học (0)
Trình tự thuộc tính PS
Thuộc tính X PS(0)
Thuộc tính X (0)
Thuộc tính Y PS(0)
Thuộc tính Y0)
Đoạn đầu GOF
Đoạn đầu AU
PS
Hình học (1)
Thuộc tính X (1)
Thuộc tính Y (1)
(b)
GOF
AU(0)
AU(1)
AU(2)
Đoạn đầu GOF
Đoạn đầu AU
Dòng PCC PS
Trình tự hình học PS
Hình học PS(0)
Hình học (0)
Trình tự thuộc tính PS
Thuộc tính X PS(0)
Thuộc tính X (0)
Thuộc tính Y PS(0)
Thuộc tính Y0)
Đoạn đầu GOF
Đoạn đầu AU
Mũi tên chỉ báo thời gian khởi tạo
(c)
AU
Đoạn đầu GOF
Đoạn đầu AU
Dòng PCC PS
Trình tự hình học PS
Trình tự thuộc tính PS
Gt1
At1
Gt2
At2
Gt3
At3
Gt4
At4

FIG.84

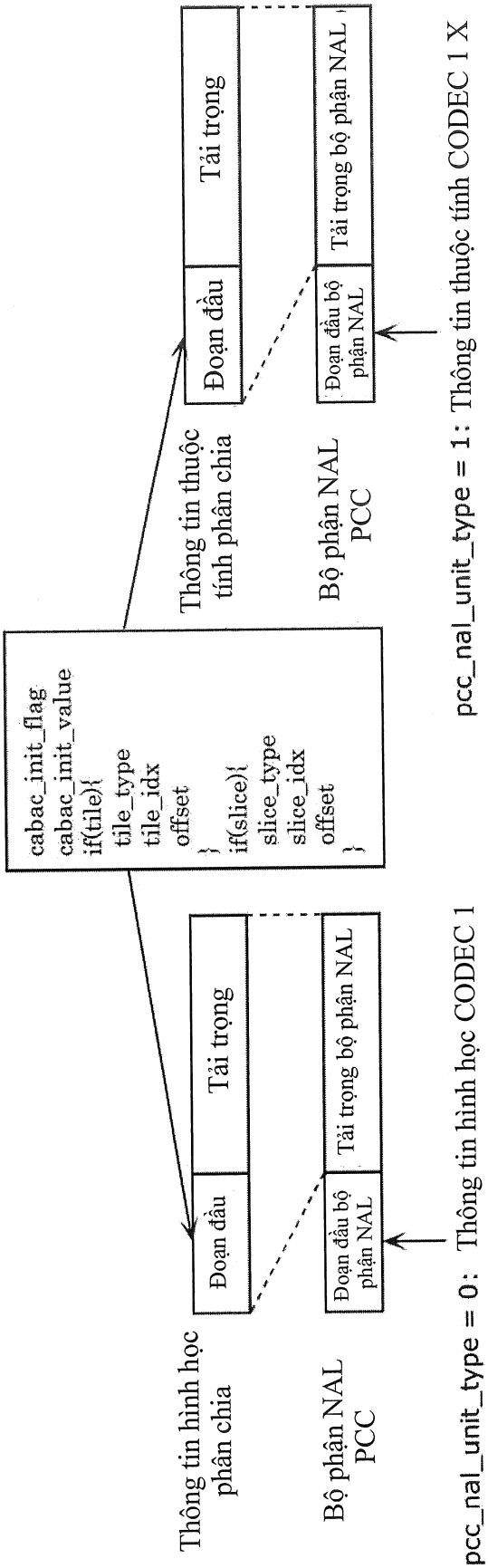

cabac_init_flag
cabac_init_value
if(tile){
tile_type
tile_idx
offset
}
if(slice){
slice_type
slice_idx
offset
}
Thông tin thuộc tính phân chia
Đoạn đầu
Tải trọng
Bộ phận NAL PCC
Đoạn đầu bộ phận NAL
Tải trọng bộ phận NAL
pcc_nal_unit_type = 1: Thông tin thuộc tính CODEC 1 X
Thông tin hình học phân chia
Đoạn đầu
Tải trọng
Bộ phận NAL PCC
Đoạn đầu bộ phận NAL
Tải trọng bộ phận NAL
pcc_nal_unit_type = 0: Thông tin hình học CODEC 1

FIG. 85

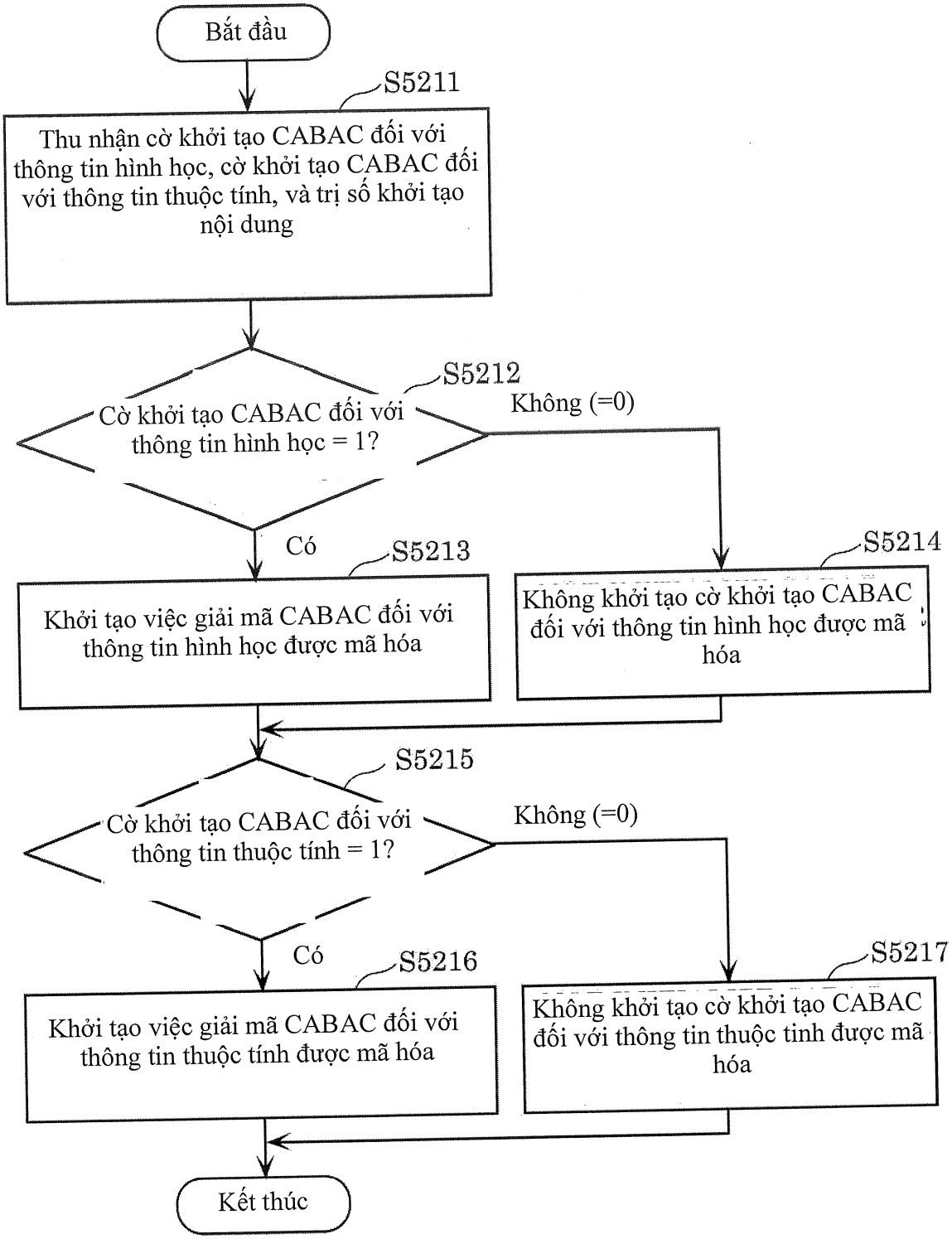

Bắt đầu
S5211
Thu nhận cờ khởi tạo CABAC đối với thông tin hình học, cờ khởi tạo CABAC đối với thông tin thuộc tính, và trị số khởi tạo nội dung
S5212
Cờ khởi tạo CABAC đối với thông tin hình học = 1?
Không (=0)
Có
S5213
Khởi tạo việc giải mã CABAC đối với thông tin hình học được mã hóa
S5214
Không khởi tạo cờ khởi tạo CABAC đối với thông tin hình học được mã hóa
S5215
Cờ khởi tạo CABAC đối với thông tin thuộc tính = 1?
Không (=0)
Có
S5216
Khởi tạo việc giải mã CABAC đối với thông tin thuộc tính được mã hóa
S5217
Không khởi tạo cờ khởi tạo CABAC đối với thông tin thuộc tinh được mã hóa
Kết thúc

FIG. 86

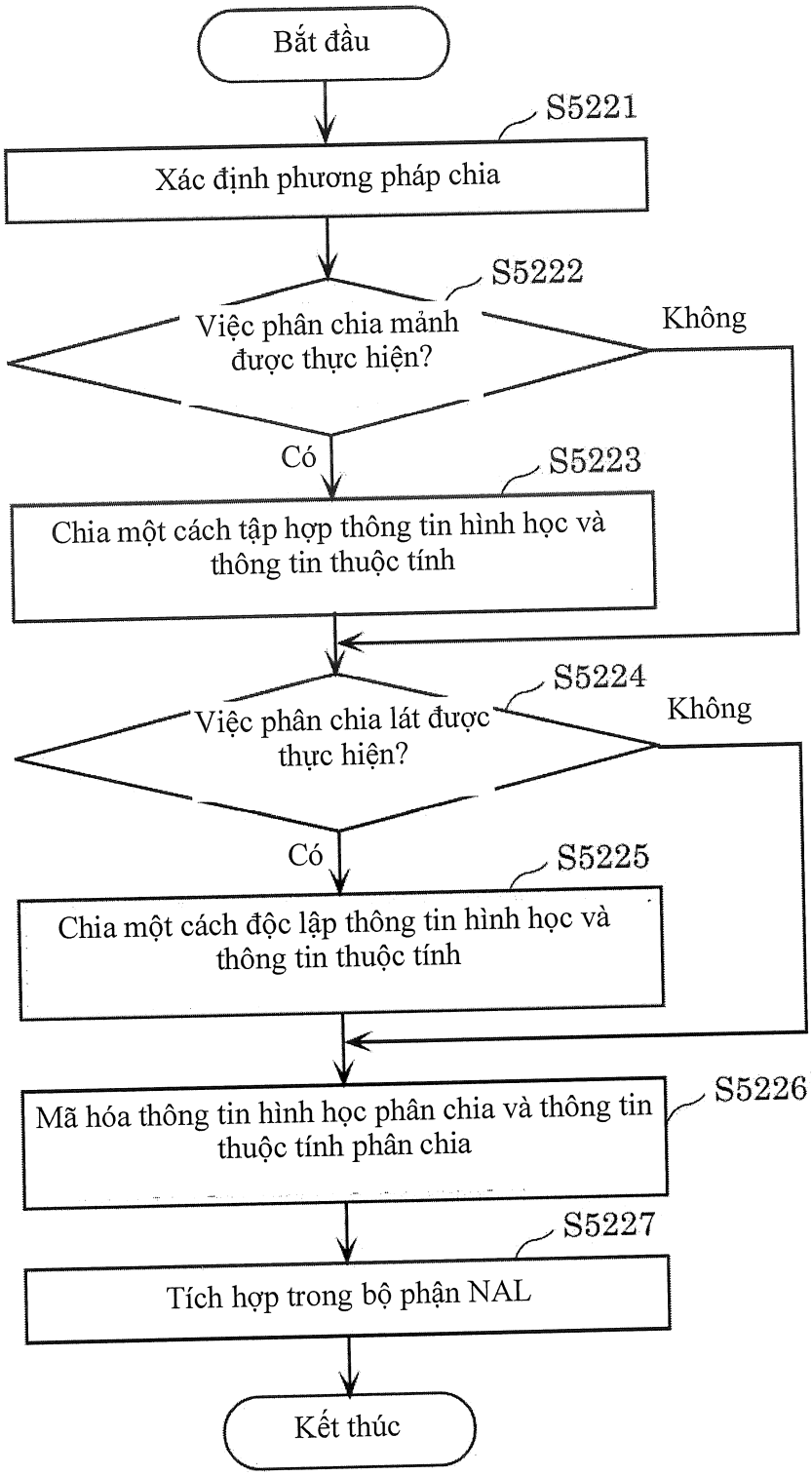

Bắt đầu
S5221
Xác định phương pháp chia
S5222
Việc phân chia mảnh được thực hiện?
Không
Có
S5223
Chia một cách tập hợp thông tin hình học và thông tin thuộc tính
S5224
Việc phân chia lát được thực hiện?
Không
Có
S5225
Chia một cách độc lập thông tin hình học và thông tin thuộc tính
Mã hóa thông tin hình học phân chia và thông tin thuộc tính phân chia
S5226
S5227
Tích hợp trong bộ phận NAL
Kết thúc

FIG. 87

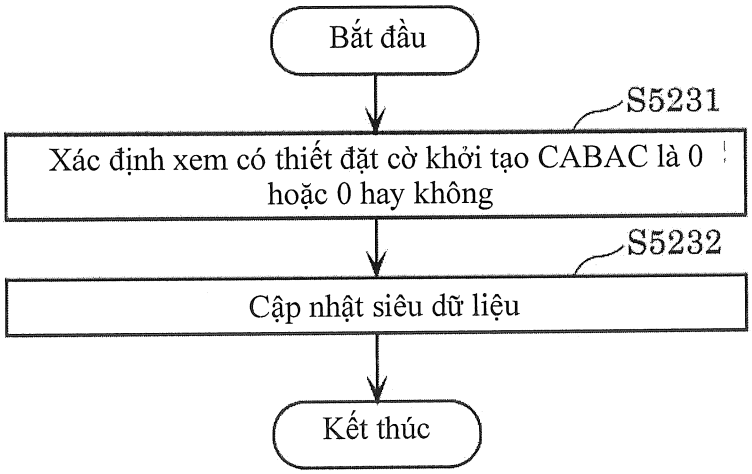

Bắt đầu
S5231
Xác định xem có thiết đặt cờ khởi tạo CABAC là 0 hoặc 0 hay không
S5232
Cập nhật siêu dữ liệu
Kết thúc

FIG. 88

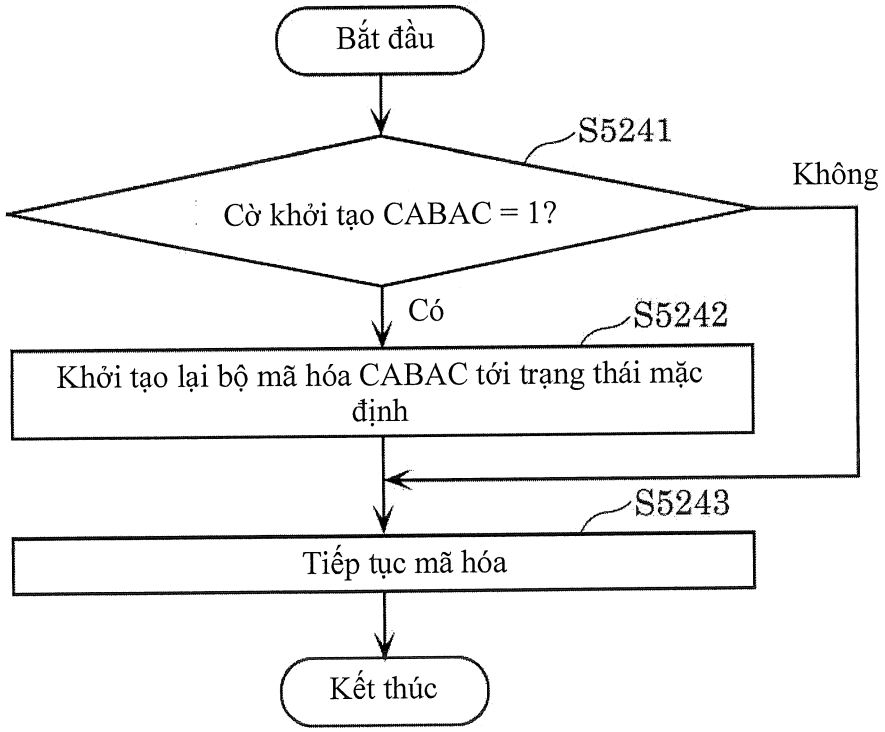

Bắt đầu
S5241
Cờ khởi tạo CABAC = 1?
Không
Có
S5242
Khởi tạo lại bộ mã hóa CABAC tới trạng thái mặc định
S5243
Tiếp tục mã hóa
Kết thúc

FIG. 89

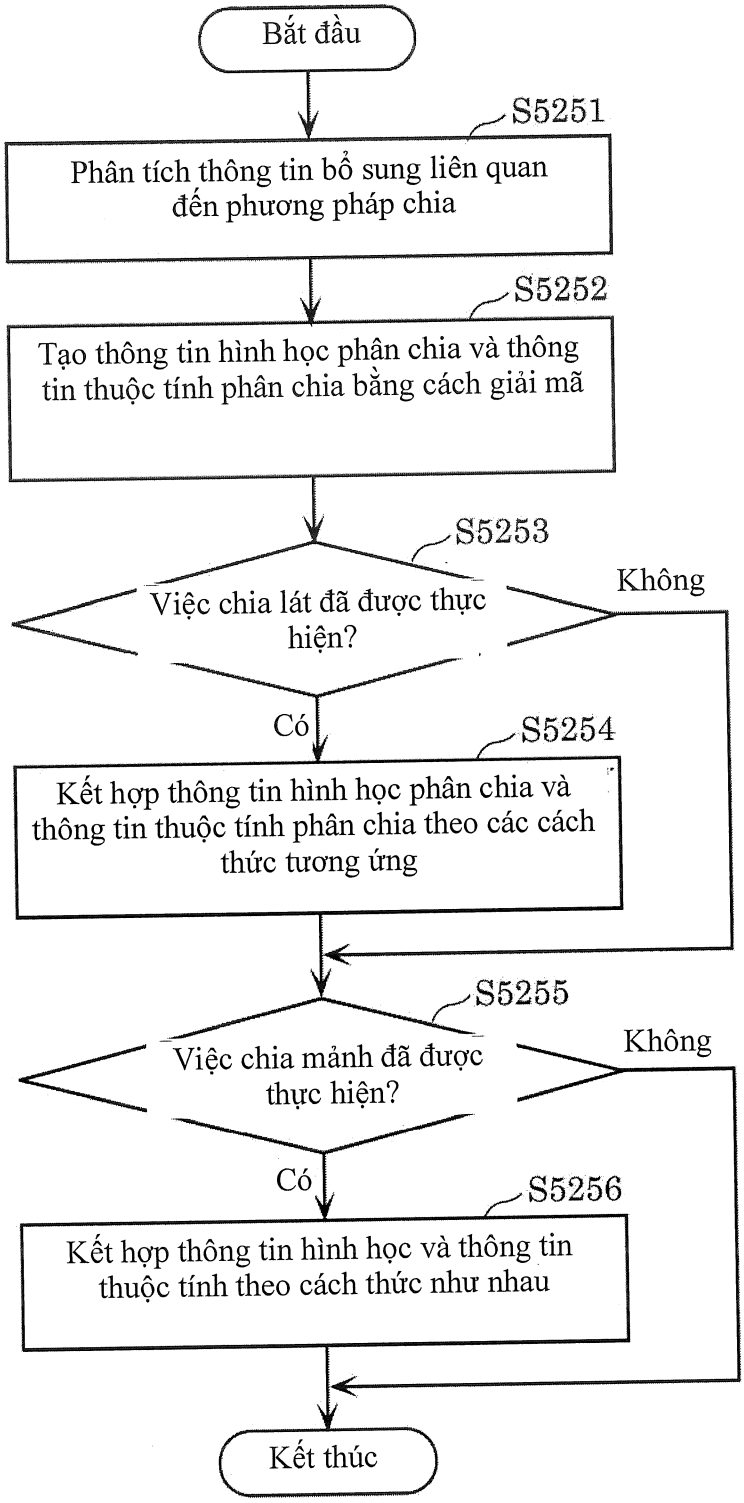

Bắt đầu
S5251
Phân tích thông tin bổ sung liên quan đến phương pháp chia
S5252
Tạo thông tin hình học phân chia và thông tin thuộc tính phân chia bằng cách giải mã
S5253
Việc chia lát đã được thực hiện?
Không
Có
S5254
Kết hợp thông tin hình học phân chia và thông tin thuộc tính phân chia theo các cách thức tương ứng
S5255
Việc chia mảnh đã được thực hiện?
Không
Có
S5256
Kết hợp thông tin hình học và thông tin thuộc tính theo cách thức như nhau
Kết thúc

FIG. 90

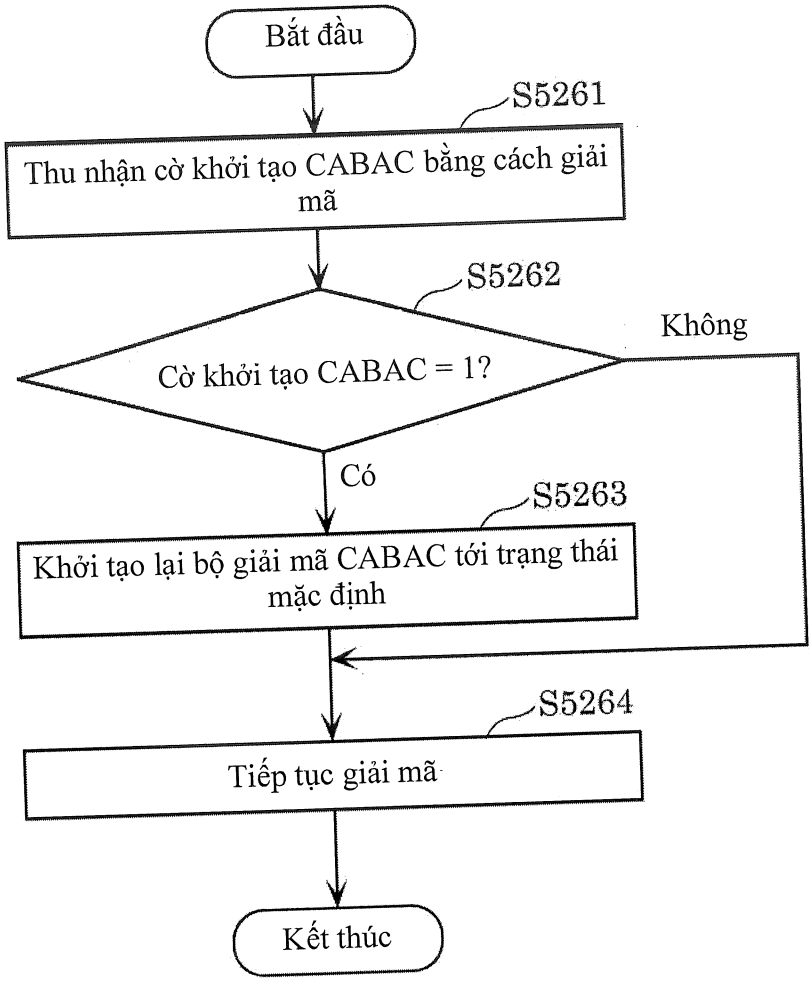

Bắt đầu
S5261
Thu nhận cờ khởi tạo CABAC bằng cách giải mã
S5262
Cờ khởi tạo CABAC = 1?
Không
Có
S5263
Khởi tạo lại bộ giải mã CABAC tới trạng thái mặc định
S5264
Tiếp tục giải mã
Kết thúc

FIG. 91

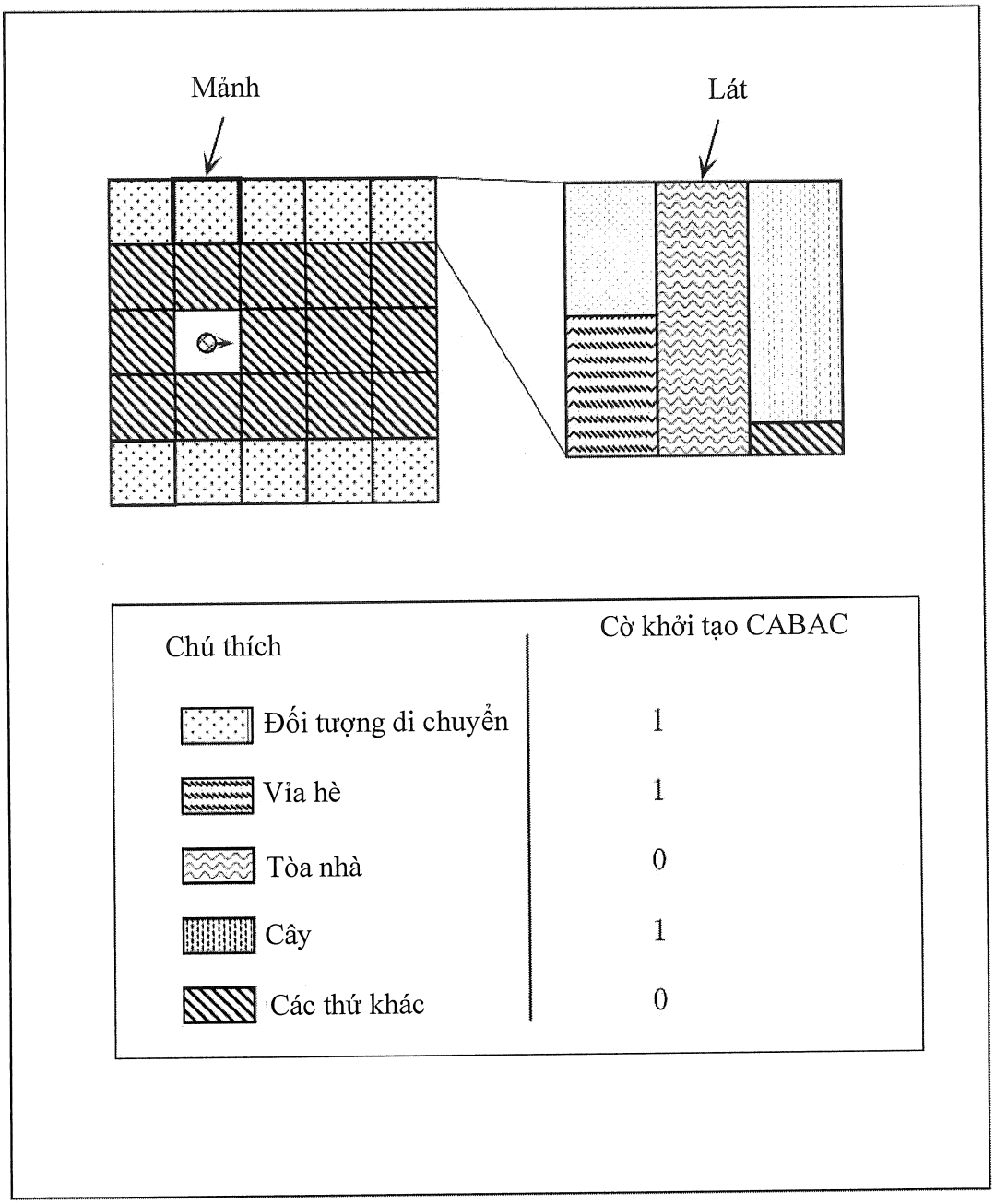

Mảnh
Lát
Chú thích
Cờ khởi tạo CABAC
Đối tượng di chuyển
1
Vỉa hè
1
Tòa nhà
0
Cây
1
Các thứ khác
0

FIG. 92

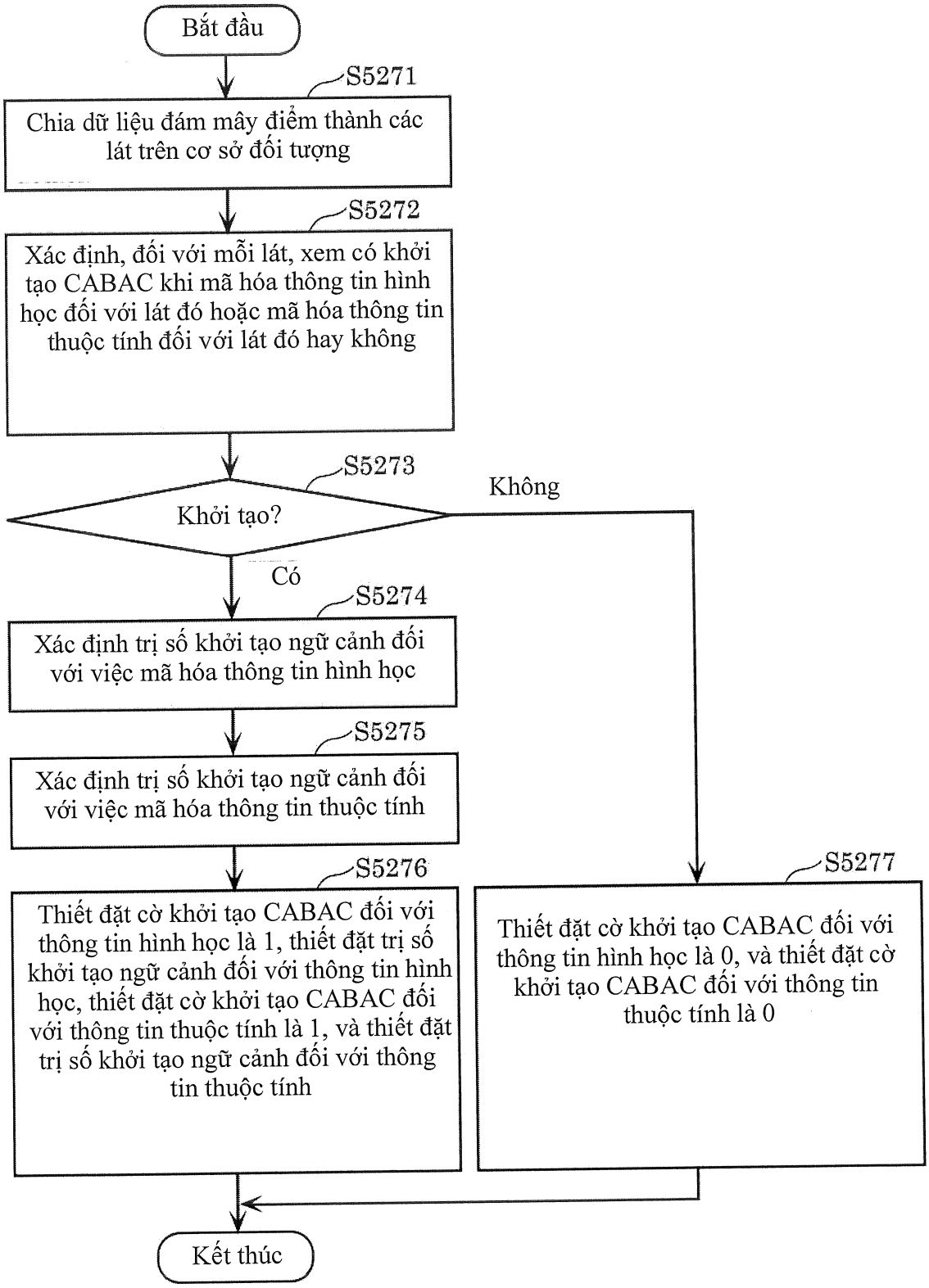

Bắt đầu
S5271
Chia dữ liệu đám mây điểm thành các lát trên cơ sở đối tượng
S5272
Xác định, đối với mỗi lát, xem có khởi tạo CABAC khi mã hóa thông tin hình học đối với lát đó hoặc mã hóa thông tin thuộc tính đối với lát đó hay không
S5273
Khởi tạo?
Không
Có
S5274
Xác định trị số khởi tạo ngữ cảnh đối với việc mã hóa thông tin hình học
S5275
Xác định trị số khởi tạo ngữ cảnh đối với việc mã hóa thông tin thuộc tính
S5276
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1, thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin hình học, thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1, và thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin thuộc tính
S5277
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0, và thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0
Kết thúc

FIG.93

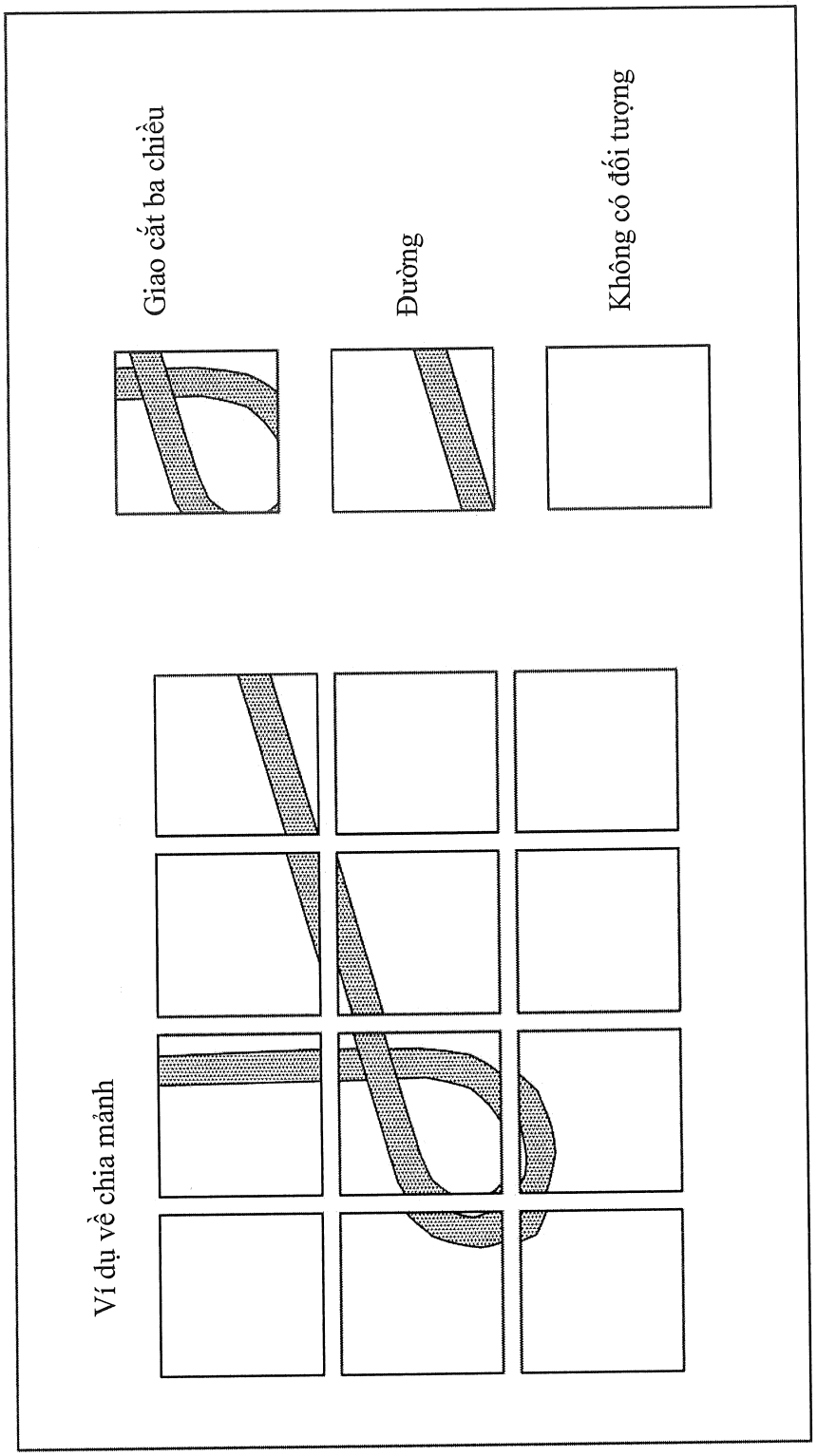


FIG. 94

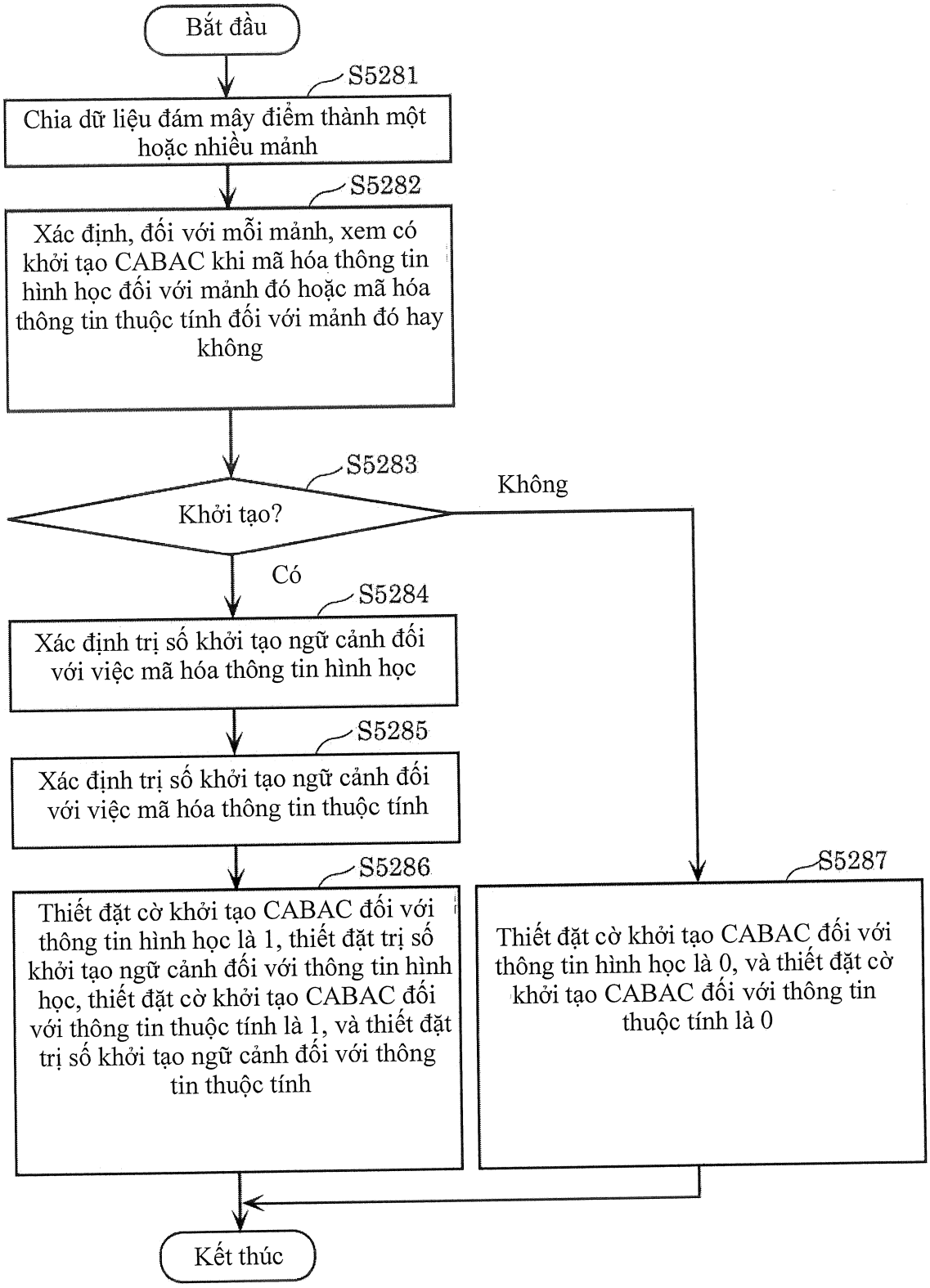

Bắt đầu
S5281
Chia dữ liệu đám mây điểm thành một hoặc nhiều mảnh
S5282
Xác định, đối với mỗi mảnh, xem có khởi tạo CABAC khi mã hóa thông tin hình học đối với mảnh đó hoặc mã hóa thông tin thuộc tính đối với mảnh đó hay không
S5283
Khởi tạo?
Không
Có
S5284
Xác định trị số khởi tạo ngữ cảnh đối với việc mã hóa thông tin hình học
S5285
Xác định trị số khởi tạo ngữ cảnh đối với việc mã hóa thông tin thuộc tính
S5286
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 1, thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin hình học, thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 1, và thiết đặt trị số khởi tạo ngữ cảnh đối với thông tin thuộc tính
S5287
Thiết đặt cờ khởi tạo CABAC đối với thông tin hình học là 0, và thiết đặt cờ khởi tạo CABAC đối với thông tin thuộc tính là 0
Kết thúc

FIG. 95

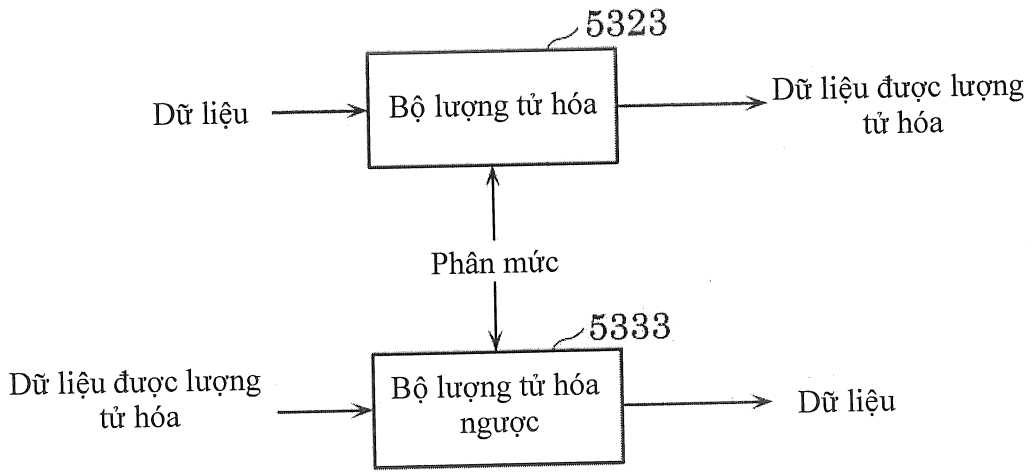

5323
Dữ liệu
Bộ lượng tử hóa
Dữ liệu được lượng tử hóa
Phân mức
5333
Dữ liệu được lượng tử hóa
Bộ lượng tử hóa ngược
Dữ liệu

FIG. 96

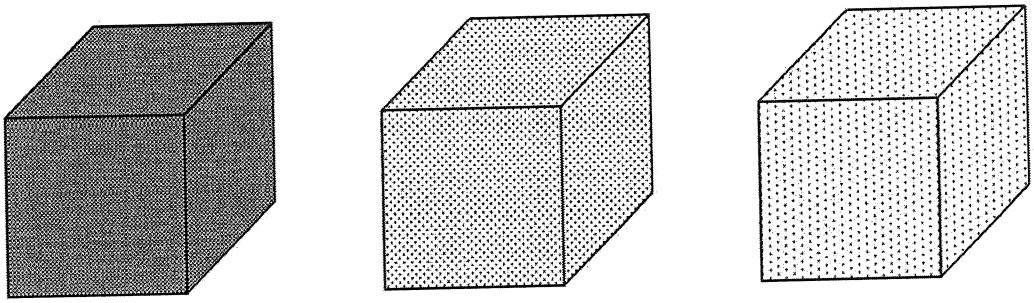

Trị số ngầm định

FIG. 97

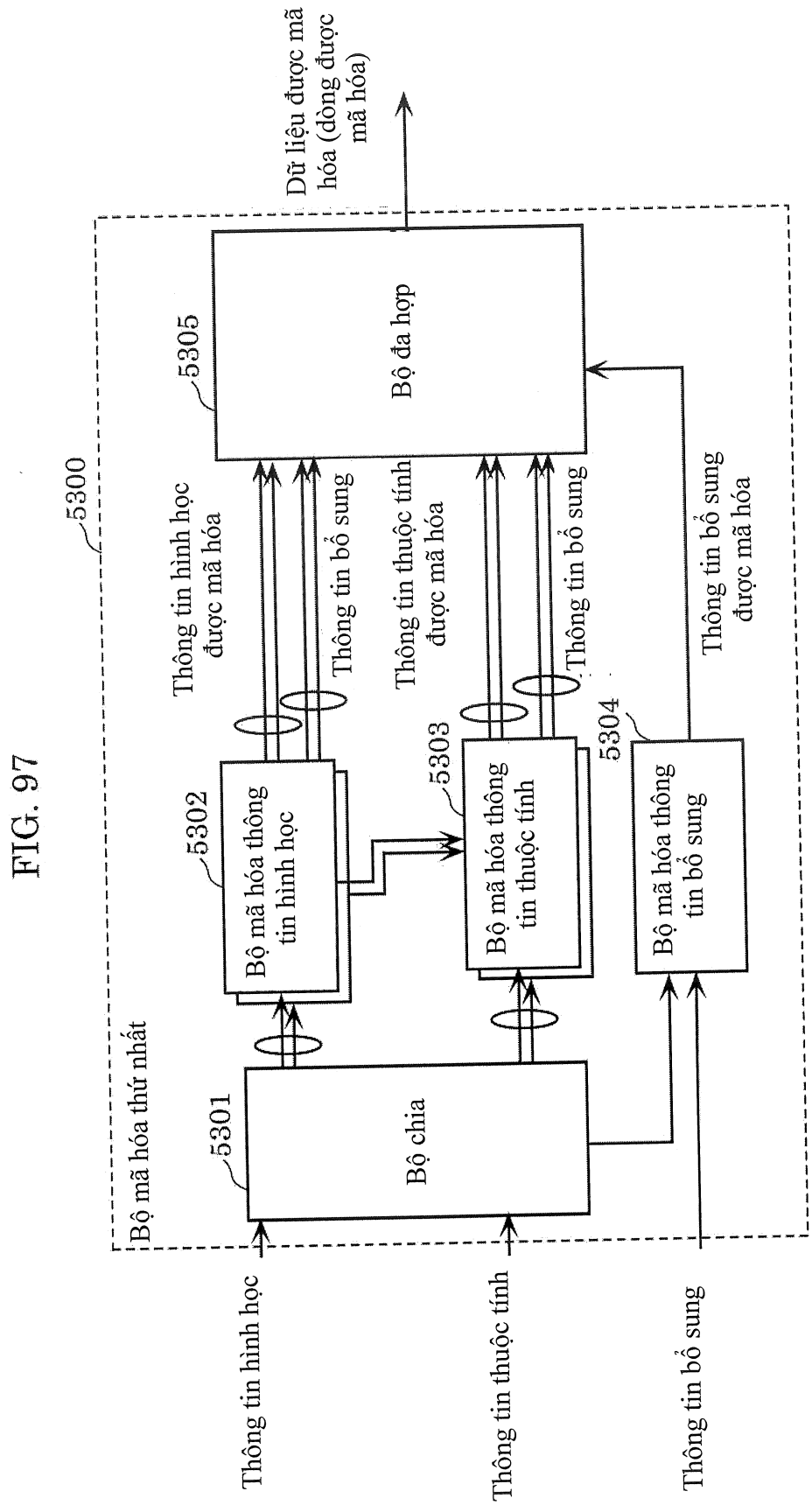

Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung
Bộ mã hóa thứ nhất
5300
Bộ chia
5301
Bộ mã hóa thông tin hình học
5302
Bộ mã hóa thông tin thuộc tính
5303
Bộ mã hóa thông tin bổ sung
5304
Bộ đa hợp
5305
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
Dữ liệu được mã hóa (dòng được mã hóa)

FIG. 98

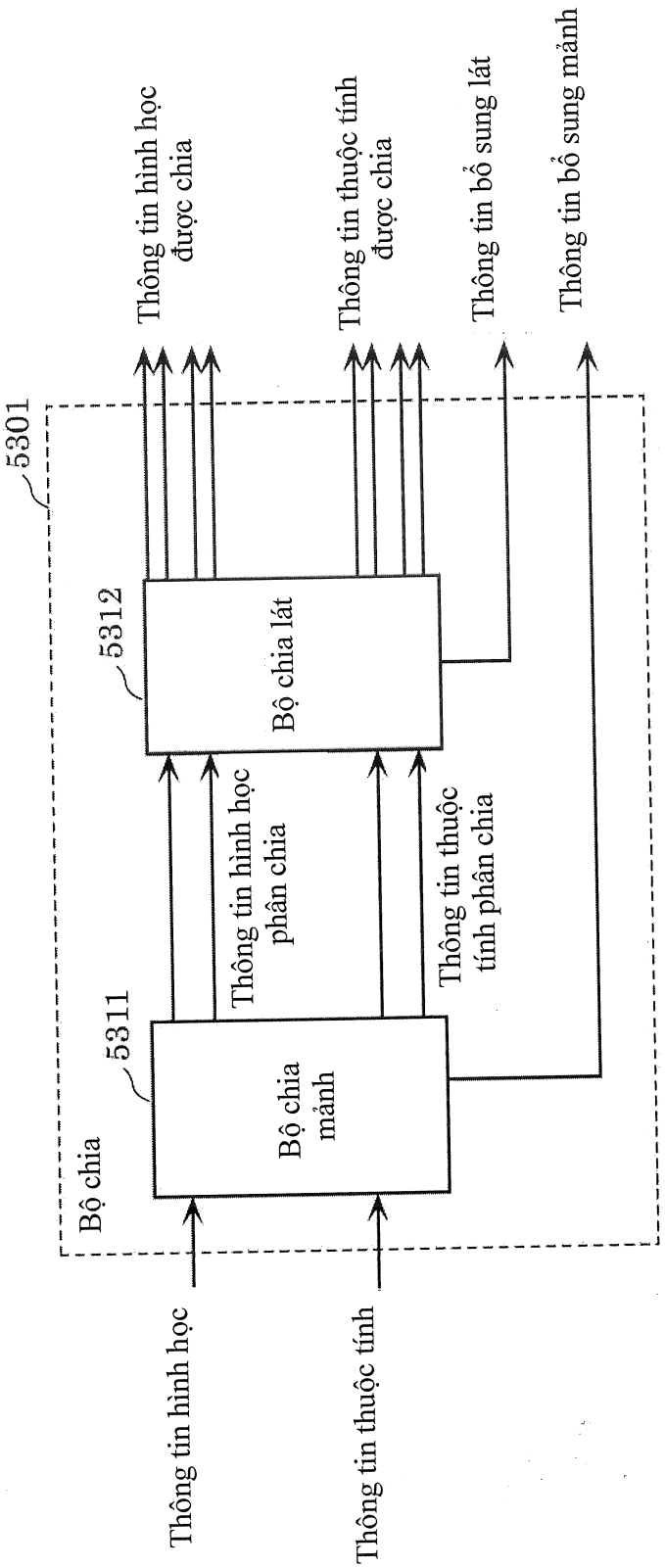

Thông tin hình học
Thông tin thuộc tính
Bộ chia
5301
5311
Bộ chia mảnh
5312
Bộ chia lát
Thông tin hình học phân chia
Thông tin thuộc tính phân chia
Thông tin hình học được chia
Thông tin thuộc tính được chia
Thông tin bổ sung lát
Thông tin bổ sung mảnh

FIG. 99

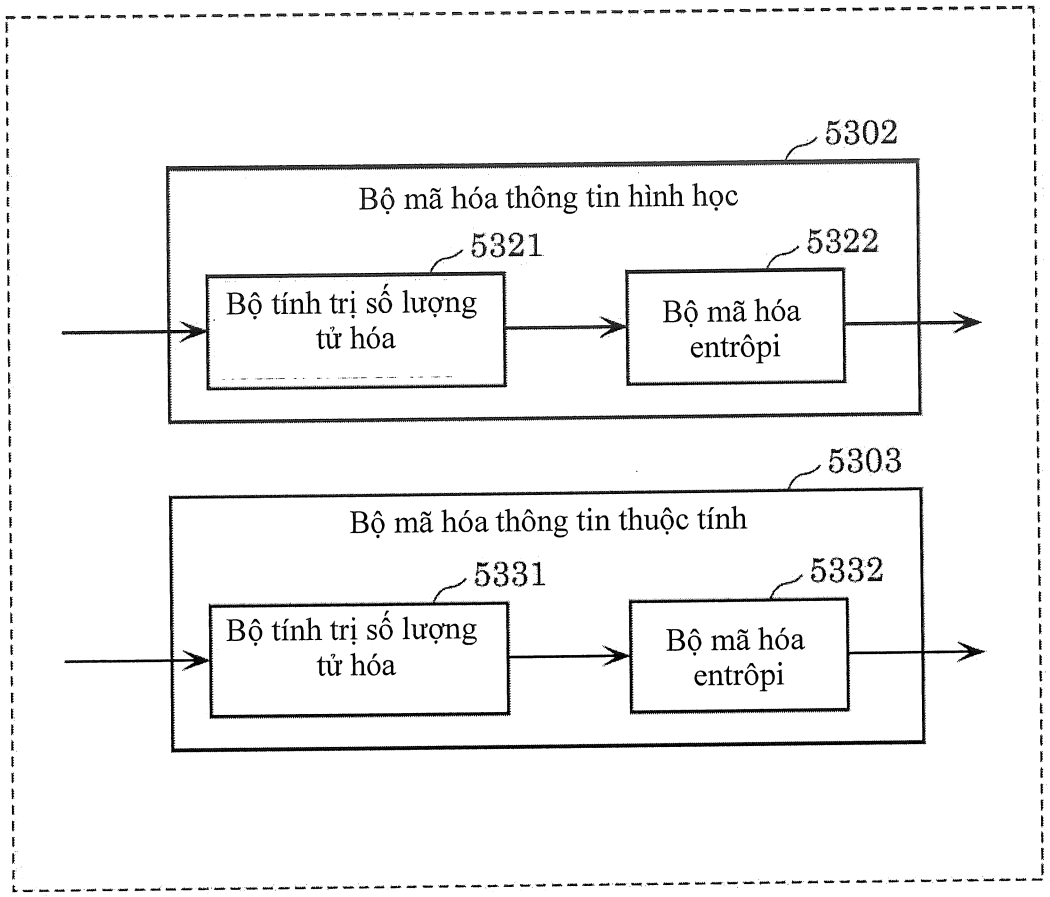

5302
Bộ mã hóa thông tin hình học
5321
Bộ tính trị số lượng tử hóa
5322
Bộ mã hóa entrôpi
5303
Bộ mã hóa thông tin thuộc tính
5331
Bộ tính trị số lượng tử hóa
5332
Bộ mã hóa entrôpi

FIG. 100

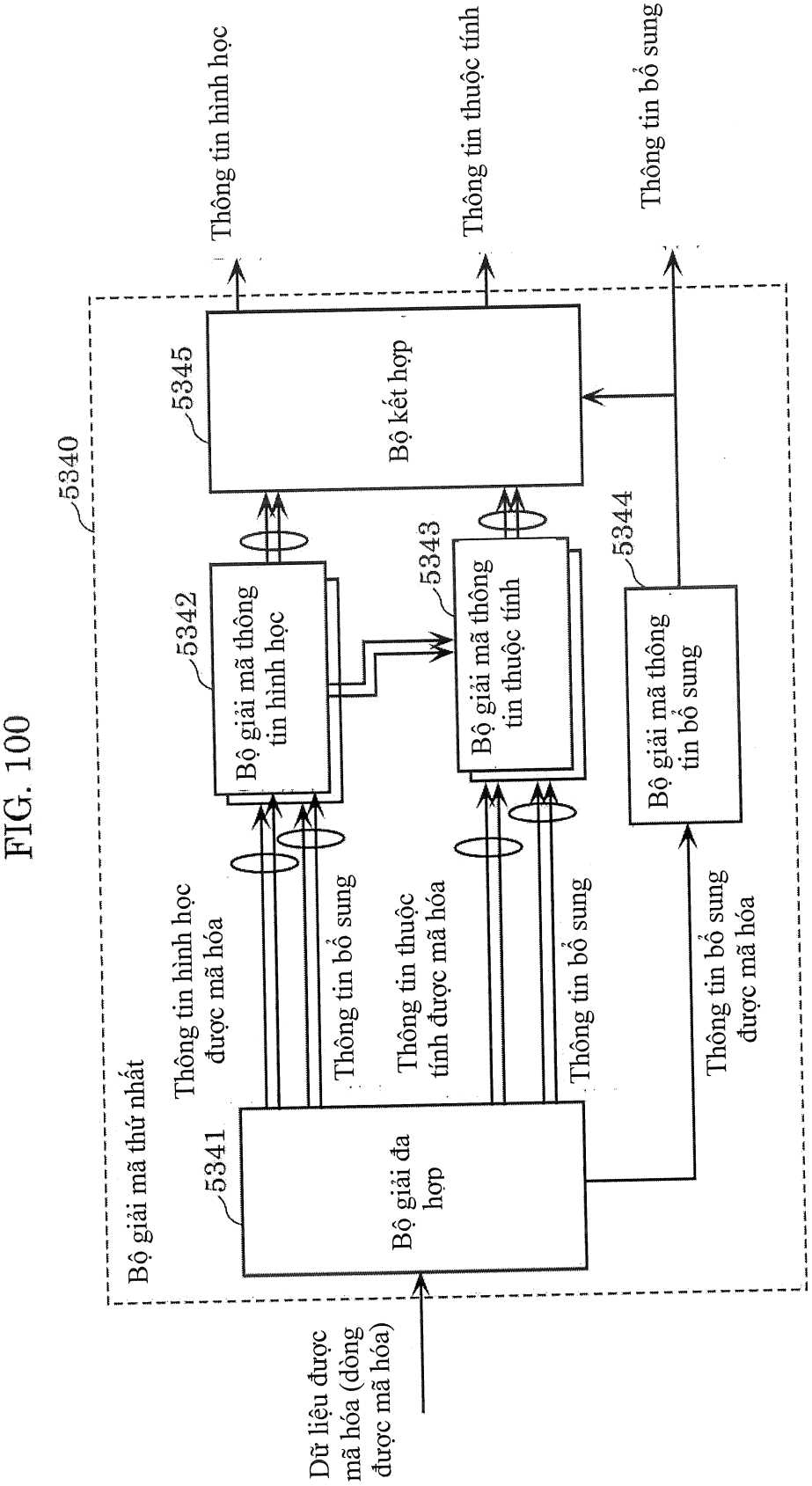

Dữ liệu được mã hóa (dòng được mã hóa)
Bộ giải mã thứ nhất
5340
5341
Bộ giải đa hợp
Thông tin hình học được mã hóa
Thông tin bổ sung
Thông tin thuộc tính được mã hóa
Thông tin bổ sung
Thông tin bổ sung được mã hóa
5342
Bộ giải mã thông tin hình học
5343
Bộ giải mã thông tin thuộc tính
5344
Bộ giải mã thông tin bổ sung
5345
Bộ kết hợp
Thông tin hình học
Thông tin thuộc tính
Thông tin bổ sung

FIG. 101

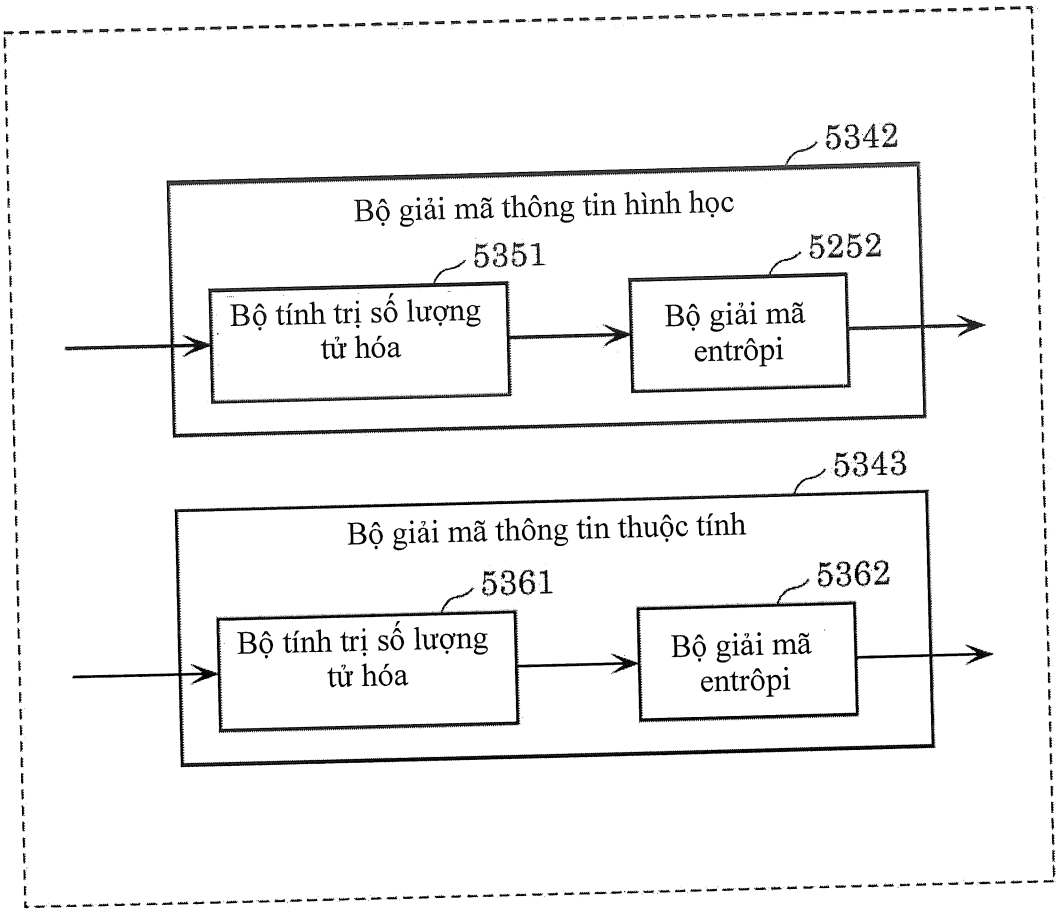

5342
Bộ giải mã thông tin hình học
5351
Bộ tính trị số lượng tử hóa
5252
Bộ giải mã entrôpi
5343
Bộ giải mã thông tin thuộc tính
5361
Bộ tính trị số lượng tử hóa
5362
Bộ giải mã entrôpi

FIG. 102

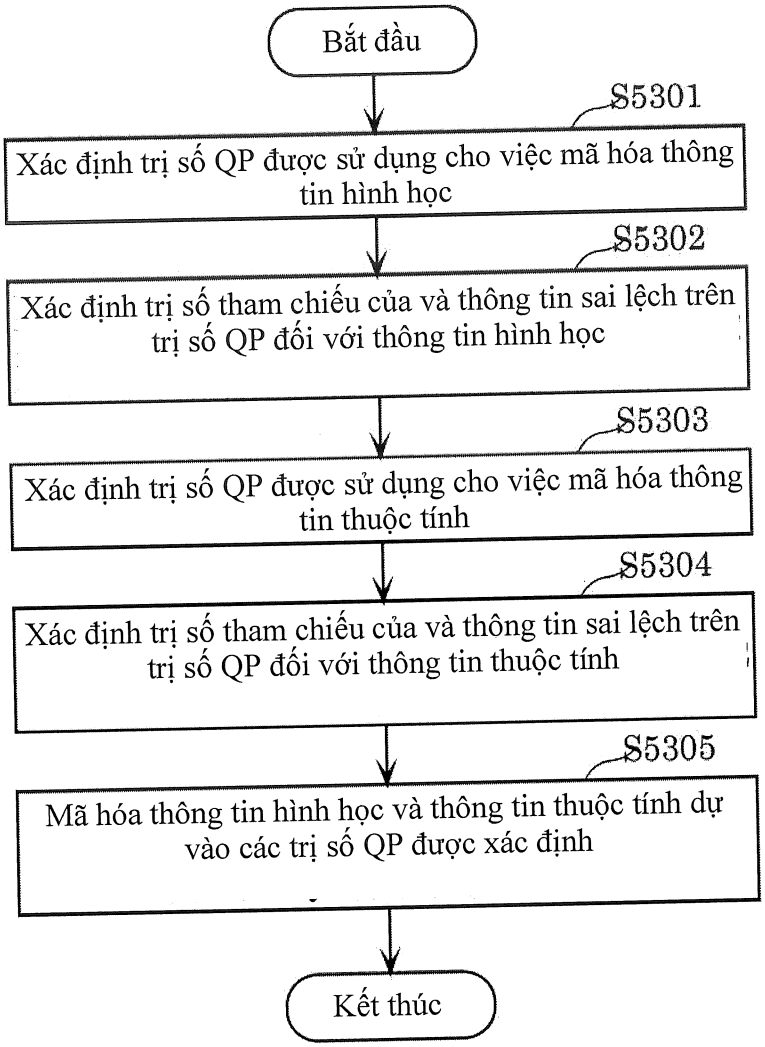

Bắt đầu
S5301
Xác định trị số QP được sử dụng cho việc mã hóa thông tin hình học
S5302
Xác định trị số tham chiếu của và thông tin sai lệch trên trị số QP đối với thông tin hình học
S5303
Xác định trị số QP được sử dụng cho việc mã hóa thông tin thuộc tính
S5304
Xác định trị số tham chiếu của và thông tin sai lệch trên trị số QP đối với thông tin thuộc tính
S5305
Mã hóa thông tin hình học và thông tin thuộc tính dự vào các trị số QP được xác định
Kết thúc

FIG. 103

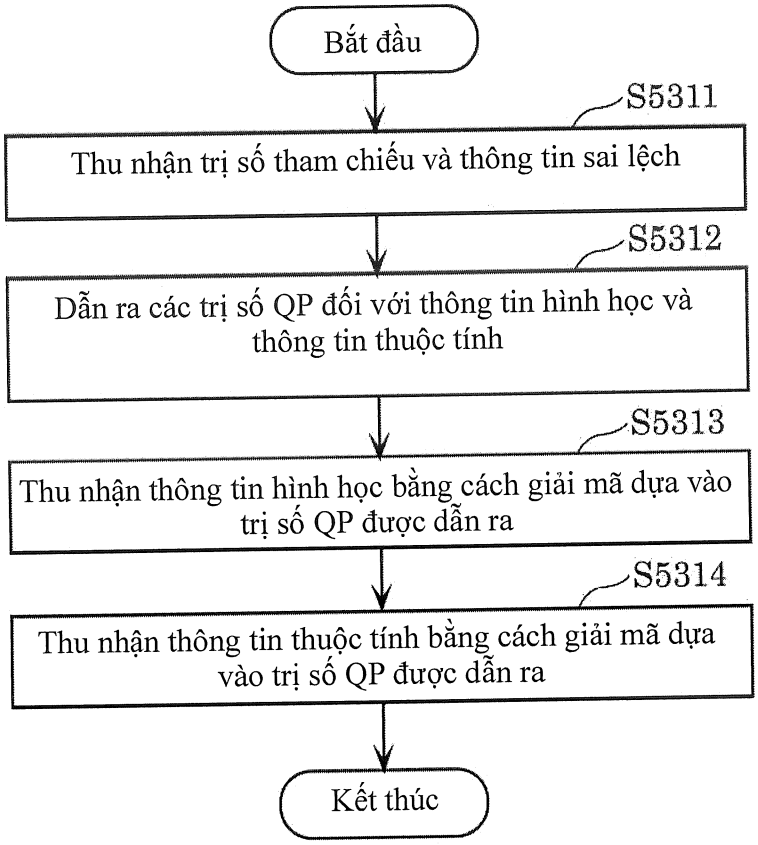

Bắt đầu
S5311
Thu nhận trị số tham chiếu và thông tin sai lệch
S5312
Dẫn ra các trị số QP đối với thông tin hình học và thông tin thuộc tính
S5313
Thu nhận thông tin hình học bằng cách giải mã dựa vào trị số QP được dẫn ra
S5314
Thu nhận thông tin thuộc tính bằng cách giải mã dựa vào trị số QP được dẫn ra
Kết thúc

FIG. 104

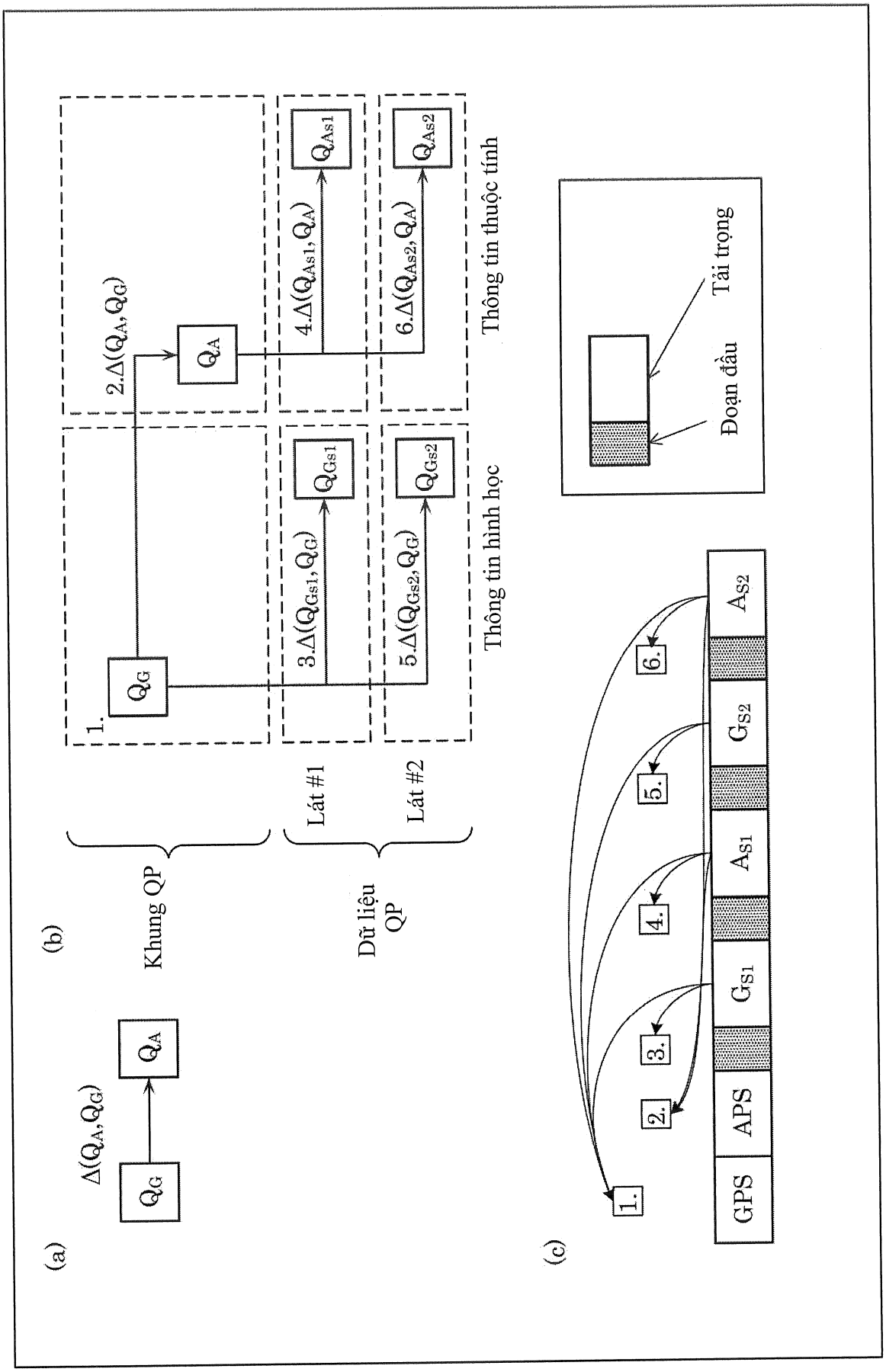

(a)
$\Delta(Q_A, Q_G)$
Q_G
Q_A
(b)
Khung QP
Dữ liệu QP
Lát #1
Lát #2
1.
Q_G
$2.\Delta(Q_A, Q_G)$
Q_A
$3.\Delta(Q_{Gs1}, Q_G)$
Q_{Gs1}
$4.\Delta(Q_{As1}, Q_A)$
Q_{As1}
$5.\Delta(Q_{Gs2}, Q_G)$
Q_{Gs2}
$6.\Delta(Q_{As2}, Q_A)$
Q_{As2}
Thông tin hình học
Thông tin thuộc tính
(c)
Tài trọng
Đoạn đầu
GPS
APS
G_{S1}
A_{S1}
G_{S2}
A_{S2}
1.
2.
3.
4.
5.
6.

FIG. 105

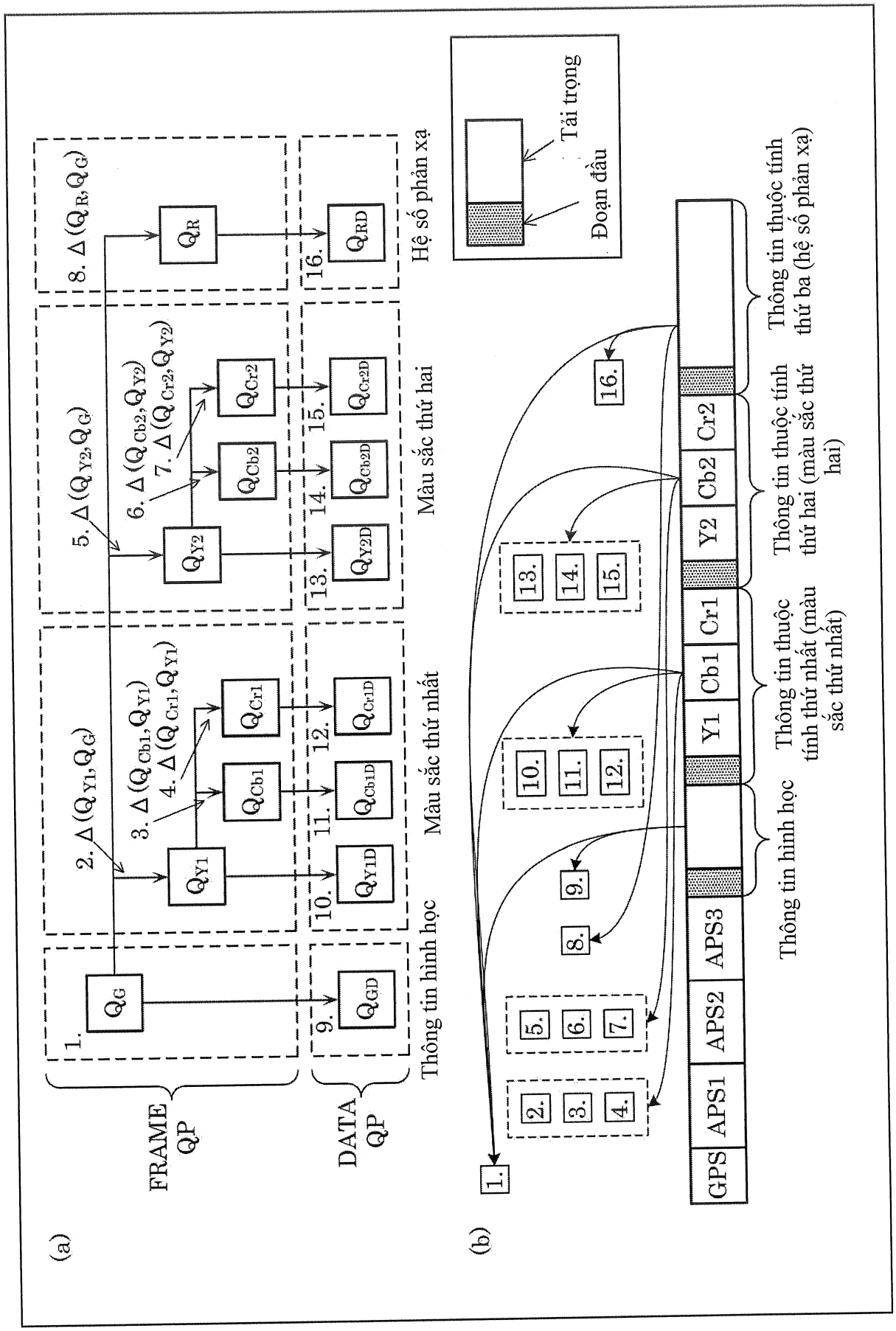

(a)
FRAME QP
DATA QP
Thông tin hình học
Màu sắc thứ nhất
Màu sắc thứ hai
Hệ số phân xạ
(b)
Tải trọng
Đoạn đầu
GPS
APS1
APS2
APS3
Y1
Cb1
Cr1
Y2
Cb2
Cr2
Thông tin hình học
Thông tin thuộc tính thứ nhất (màu sắc thứ nhất)
Thông tin thuộc tính thứ hai (màu sắc thứ hai)
Thông tin thuộc tính thứ ba (hệ số phân xạ)

FIG. 106

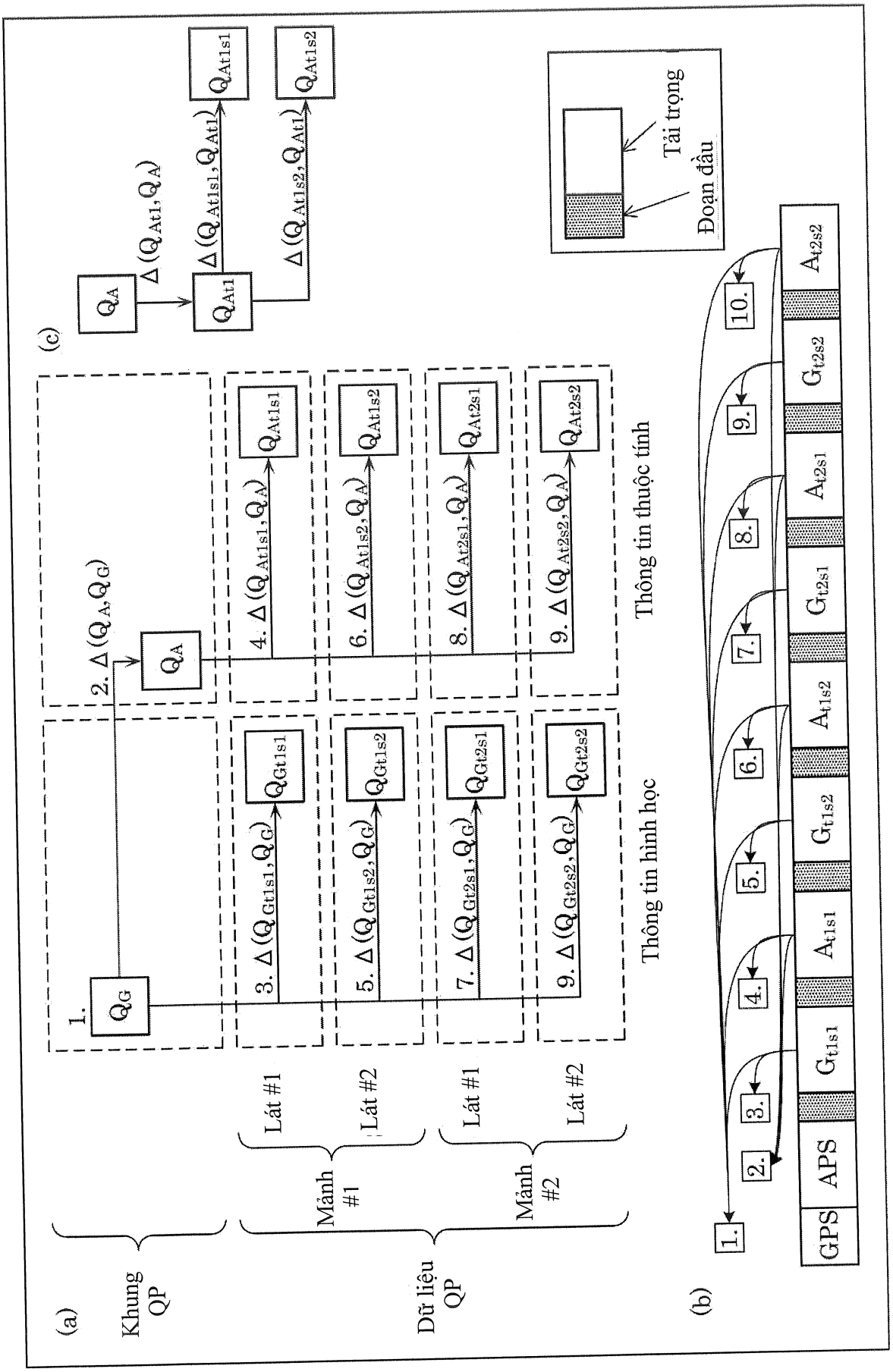

(a)
Khung QP
Dữ liệu QP
Mảnh #1
Mảnh #2
Lát #1
Lát #2
Lát #1
Lát #2
1.
Q_G
2. $\Delta(Q_A, Q_G)$
Q_A
3. $\Delta(Q_{Gt1s1}, Q_G)$
Q_{Gt1s1}
4. $\Delta(Q_{At1s1}, Q_A)$
Q_{At1s1}
5. $\Delta(Q_{Gt1s2}, Q_G)$
Q_{Gt1s2}
6. $\Delta(Q_{At1s2}, Q_A)$
Q_{At1s2}
7. $\Delta(Q_{Gt2s1}, Q_G)$
Q_{Gt2s1}
8. $\Delta(Q_{At2s1}, Q_A)$
Q_{At2s1}
9. $\Delta(Q_{Gt2s2}, Q_G)$
Q_{Gt2s2}
9. $\Delta(Q_{At2s2}, Q_A)$
Q_{At2s2}
Thông tin hình học
Thông tin thuộc tính
(c)
Q_A
$\Delta(Q_{At1}, Q_A)$
Q_{At1}
$\Delta(Q_{At1s1}, Q_{At1})$
Q_{At1s1}
$\Delta(Q_{At1s2}, Q_{At1})$
Q_{At1s2}
Tải trọng
Đoạn đầu
(b)
GPS
APS
G_{t1s1}
A_{t1s1}
G_{t1s2}
A_{t1s2}
G_{t2s1}
A_{t2s1}
G_{t2s2}
A_{t2s2}
1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

FIG. 107

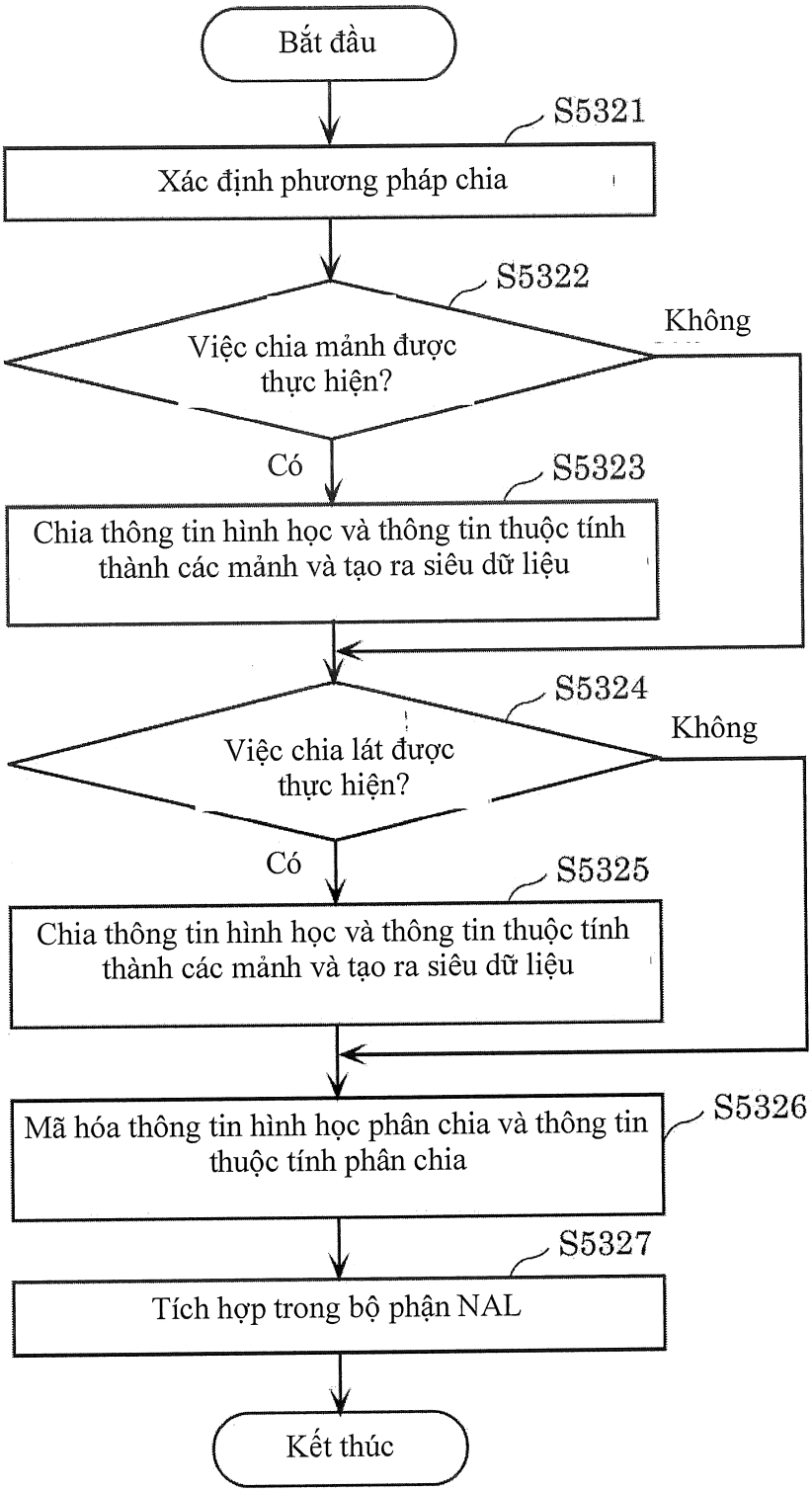

Bắt đầu
S5321
Xác định phương pháp chia
S5322
Việc chia mảnh được thực hiện?
Không
Có
S5323
Chia thông tin hình học và thông tin thuộc tính thành các mảnh và tạo ra siêu dữ liệu
S5324
Việc chia lát được thực hiện?
Không
Có
S5325
Chia thông tin hình học và thông tin thuộc tính thành các mảnh và tạo ra siêu dữ liệu
Mã hóa thông tin hình học phân chia và thông tin thuộc tính phân chia
S5326
S5327
Tích hợp trong bộ phận NAL
Kết thúc

FIG. 108

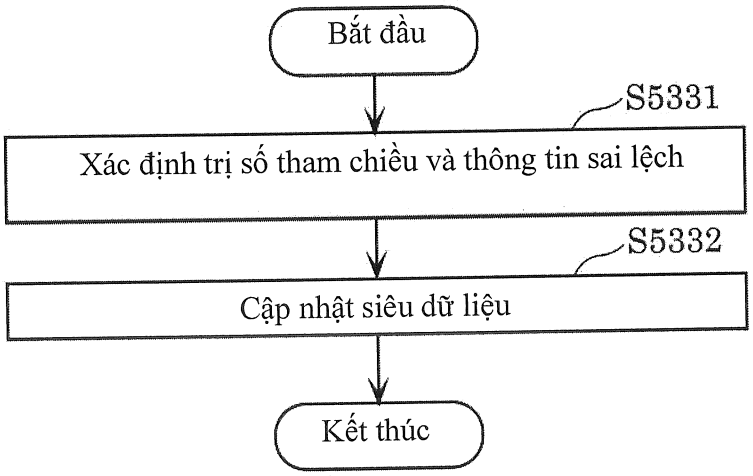


FIG. 109

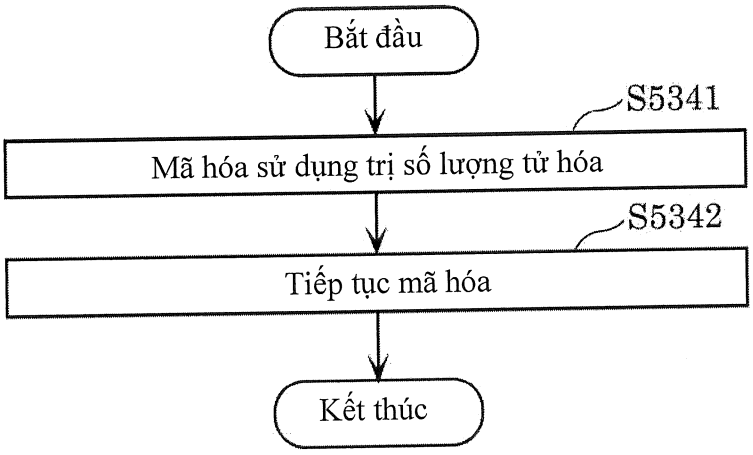


FIG. 110

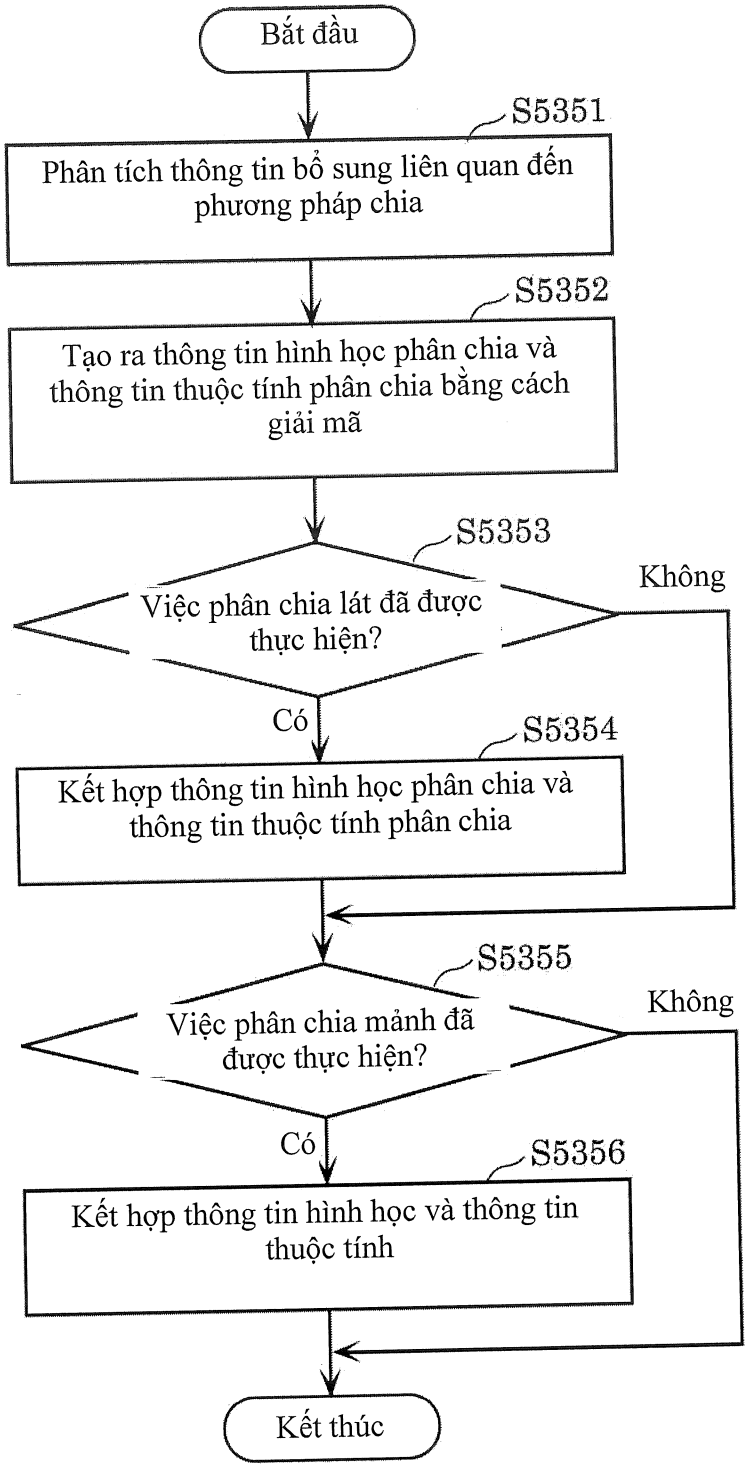

Bắt đầu
S5351
Phân tích thông tin bổ sung liên quan đến phương pháp chia
S5352
Tạo ra thông tin hình học phân chia và thông tin thuộc tính phân chia bằng cách giải mã
S5353
Việc phân chia lát đã được thực hiện?
Không
Có
S5354
Kết hợp thông tin hình học phân chia và thông tin thuộc tính phân chia
S5355
Việc phân chia mảnh đã được thực hiện?
Không
Có
S5356
Kết hợp thông tin hình học và thông tin thuộc tính
Kết thúc

FIG. 111

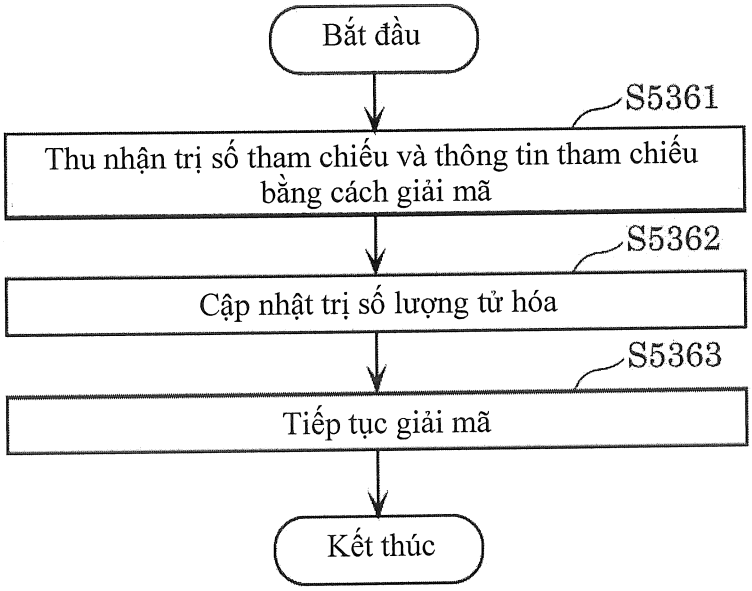


FIG. 112

```
GPS0{
  gps_idx ..PCC frame number
  sps_idx

  gps_information()
  QP_value //defalt_value

   tile_slice_num //0..NO
   divide for(tile_slice_num){
     tile_idx
     slice_idx
     gps_slice_tile_information()
   }
}
```

FIG. 113

```
APS0{
  aps_idx ..PCC frame Number
  gps_idx ..PCC frame Number

  num_of_attribute
  for(num_of_attibute){
    attribute_type  //color, reflectance
    aps_information()

    tile_slice_num  //0..no divide
    for(tile_slice_num){
      tile_idx
      slice_idx
      QP_delta_Attribute_to_Geometry
      if(attribute_type==color){
        QP_delta_Cb_to_Y
        QP_delta_Cr_to_Y
      }
      aps_slice_tile_information()
    }
  }
}
```

FIG. 114

```
Geometry_header0{
  gps_idx
  offset
  QP_delta_data_to_frame
  if(tile){
    tile_type
    tile_idx
  }
  if(slice){
    slice_type
    slice_idx
  }
}
```

FIG. 115

```
Attribute_header0{
   aps_idx
   offset
    attribute_type
   QP_delta_data_to_frame
   if(attribute_type==color){
      QP_delta_Cb_to_frame
      QP_delta_Cr_to_frame
   }
   if(tile){
      tile_type
      tile_idx
   }
   if(slice){
      slice_type
      slice_idx
   }
}
```

FIG. 116

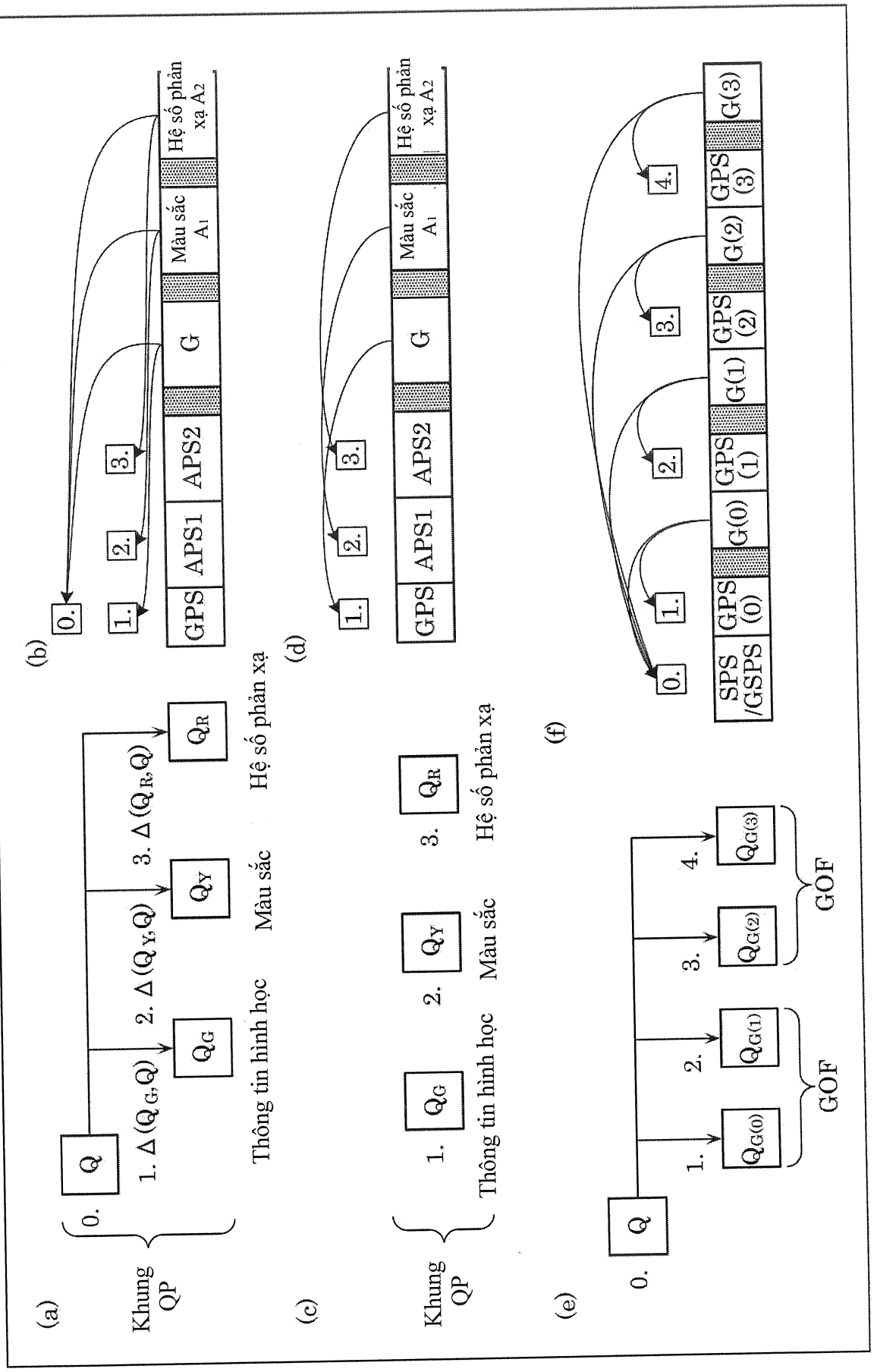

(a)
Khung QP
0.
Q
1. Δ (Q G,Q)
2. Δ (Q Y,Q)
3. Δ (Q R,Q)
Q G
Q Y
Q R
Thông tin hình học
Màu sắc
Hệ số phản xạ
(b)
0.
1.
2.
3.
GPS
APS1
APS2
G
Màu sắc A1
Hệ số phản xạ A2
(c)
Khung QP
1.
Q G
2.
Q Y
3.
Q R
Thông tin hình học
Màu sắc
Hệ số phản xạ
(d)
1.
2.
3.
GPS
APS1
APS2
G
Màu sắc A1
Hệ số phản xạ A2
(e)
0.
Q
1.
2.
3.
4.
Q G(0)
Q G(1)
Q G(2)
Q G(3)
GOF
GOF
(f)
0.
1.
2.
3.
4.
SPS /GSPS
GPS (0)
G(0)
GPS (1)
G(1)
GPS (2)
G(2)
GPS (3)
G(3)

FIG. 117

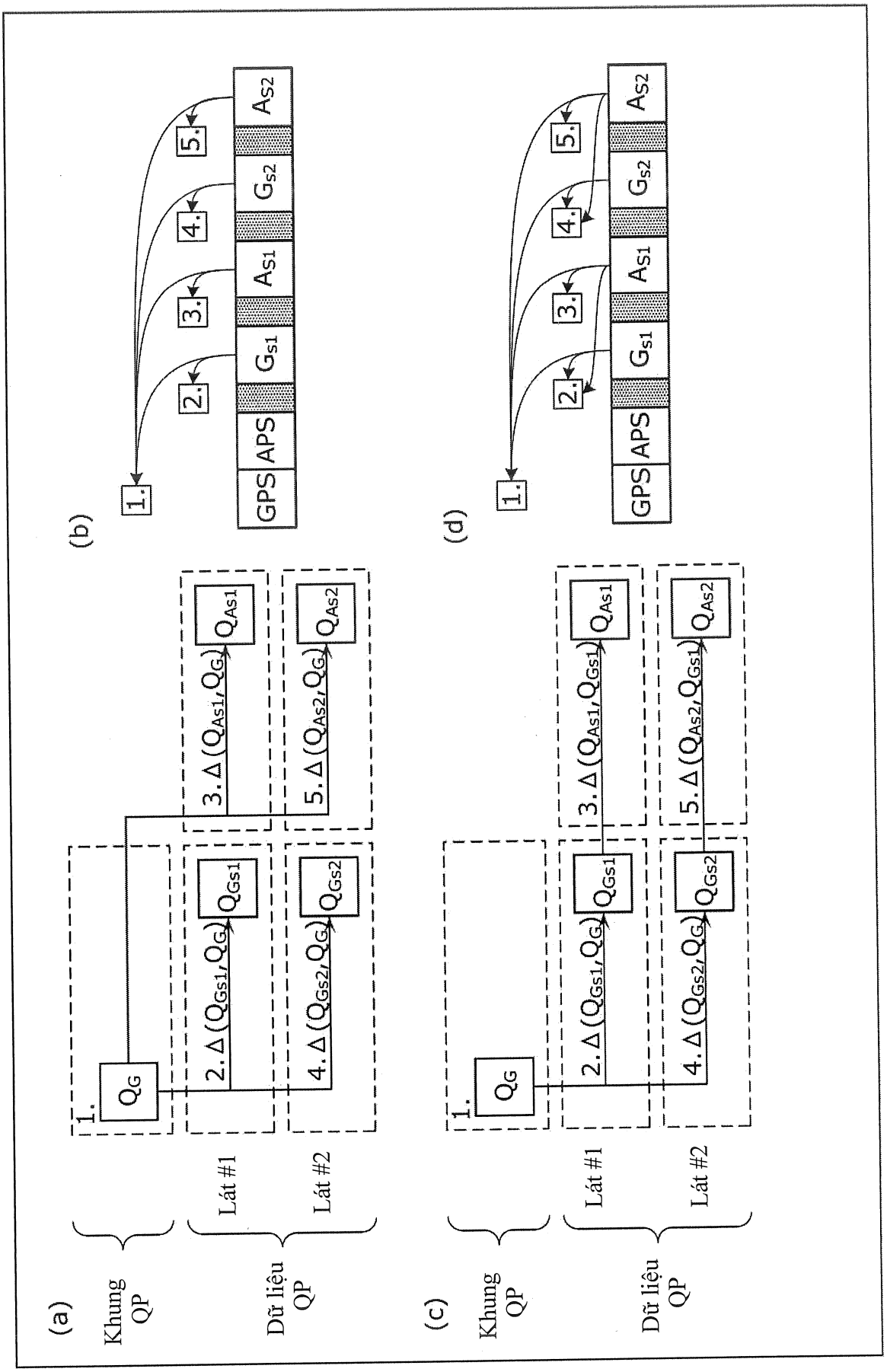

(a)
Khung QP
Dữ liệu QP
Lát #1
Lát #2
1.
Q_G
2. $\Delta(Q_{GS1}, Q_G)$
Q_{GS1}
3. $\Delta(Q_{AS1}, Q_G)$
Q_{AS1}
4. $\Delta(Q_{GS2}, Q_G)$
Q_{GS2}
5. $\Delta(Q_{AS2}, Q_G)$
Q_{AS2}
(b)
1.
GPS
APS
2.
G_{S1}
3.
A_{S1}
4.
G_{S2}
5.
A_{S2}
(c)
Khung QP
Dữ liệu QP
Lát #1
Lát #2
1.
Q_G
2. $\Delta(Q_{GS1}, Q_G)$
Q_{GS1}
3. $\Delta(Q_{AS1}, Q_{GS1})$
Q_{AS1}
4. $\Delta(Q_{GS2}, Q_G)$
Q_{GS2}
5. $\Delta(Q_{AS2}, Q_{GS1})$
Q_{AS2}
(d)
1.
GPS
APS
2.
G_{S1}
3.
A_{S1}
4.
G_{S2}
5.
A_{S2}

FIG. 118

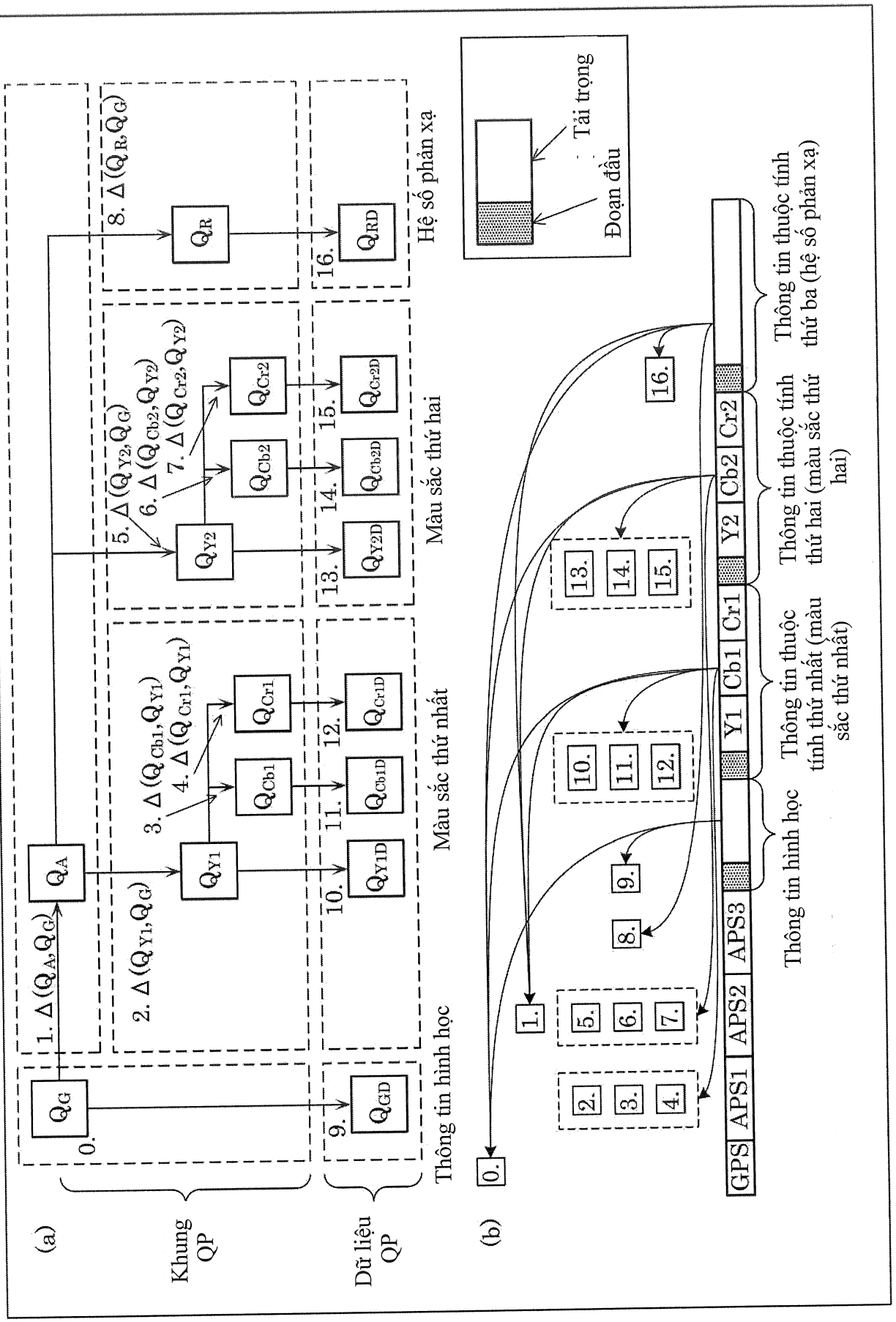

(a)
Khung QP
Dữ liệu QP
Thông tin hình học
Màu sắc thứ nhất
Màu sắc thứ hai
Hệ số phản xạ
(b)
Tải trọng
Đoạn đầu
Thông tin hình học
Thông tin thuộc tính tính thứ nhất (màu sắc thứ nhất)
Thông tin thuộc tính thứ hai (màu sắc thứ hai)
Thông tin thuộc tính thứ ba (hệ số phản xạ)
GPS
APS1
APS2
APS3
Y1
Cb1
Cr1
Y2
Cb2
Cr2

FIG. 119

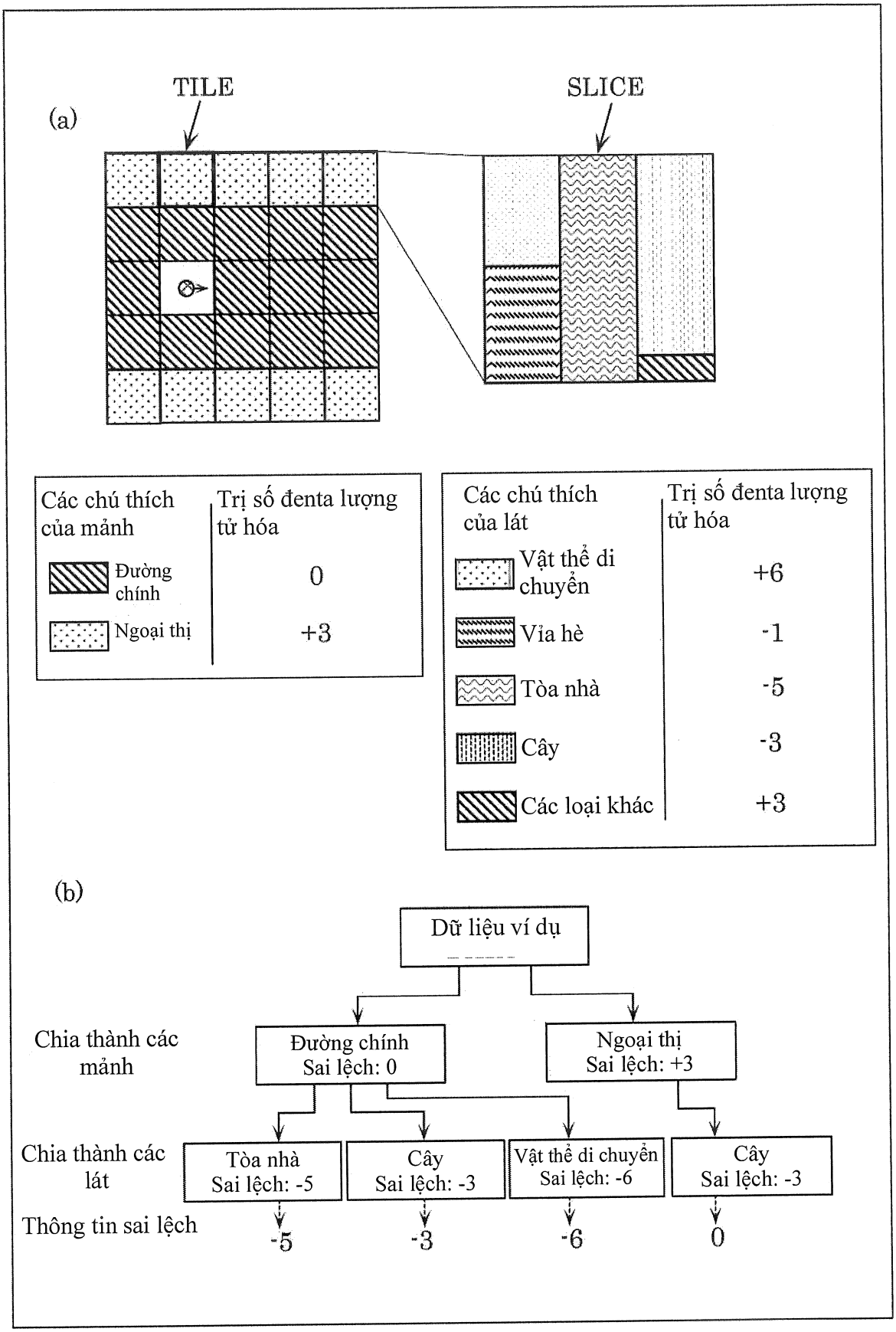

(a)
TILE
SLICE
Các chú thích của mảnh
Trị số đenta lượng tử hóa
Đường chính
0
Ngoại thị
+3
Các chú thích của lát
Trị số đenta lượng tử hóa
Vật thể di chuyển
+6
Vỉa hè
-1
Tòa nhà
-5
Cây
-3
Các loại khác
+3
(b)
Dữ liệu ví dụ
Chia thành các mảnh
Đường chính
Sai lệch: 0
Ngoại thị
Sai lệch: +3
Chia thành các lát
Tòa nhà
Sai lệch: -5
Cây
Sai lệch: -3
Vật thể di chuyển
Sai lệch: -6
Cây
Sai lệch: -3
Thông tin sai lệch
-5
-3
-6
0

FIG. 120

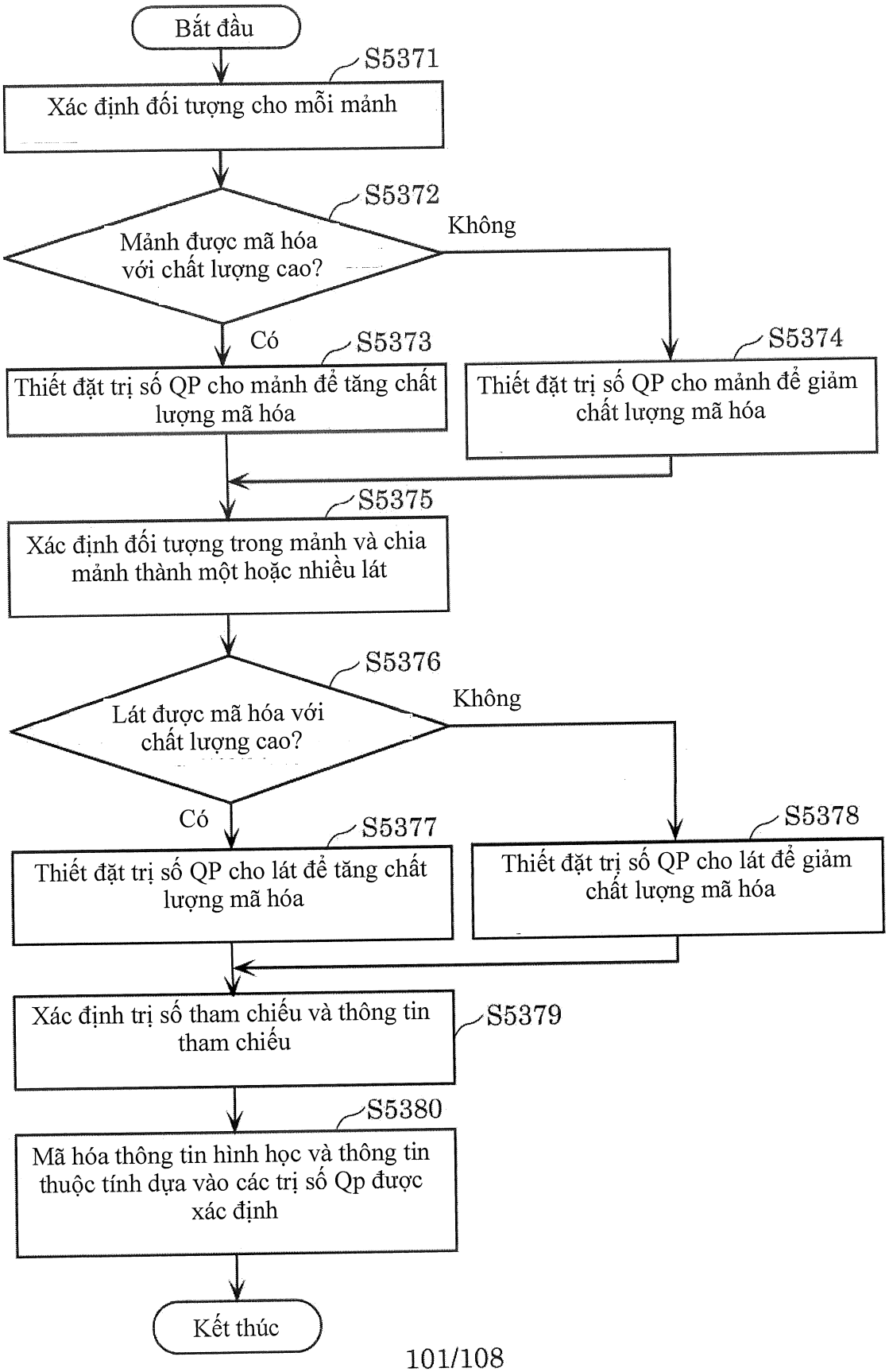

Bắt đầu
S5371
Xác định đối tượng cho mỗi mảnh
S5372
Mảnh được mã hóa với chất lượng cao?
Không
Có
S5373
Thiết đặt trị số QP cho mảnh để tăng chất lượng mã hóa
S5374
Thiết đặt trị số QP cho mảnh để giảm chất lượng mã hóa
S5375
Xác định đối tượng trong mảnh và chia mảnh thành một hoặc nhiều lát
S5376
Lát được mã hóa với chất lượng cao?
Không
Có
S5377
Thiết đặt trị số QP cho lát để tăng chất lượng mã hóa
S5378
Thiết đặt trị số QP cho lát để giảm chất lượng mã hóa
Xác định trị số tham chiếu và thông tin tham chiếu
S5379
S5380
Mã hóa thông tin hình học và thông tin thuộc tính dựa vào các trị số Qp được xác định
Kết thúc

FIG. 121

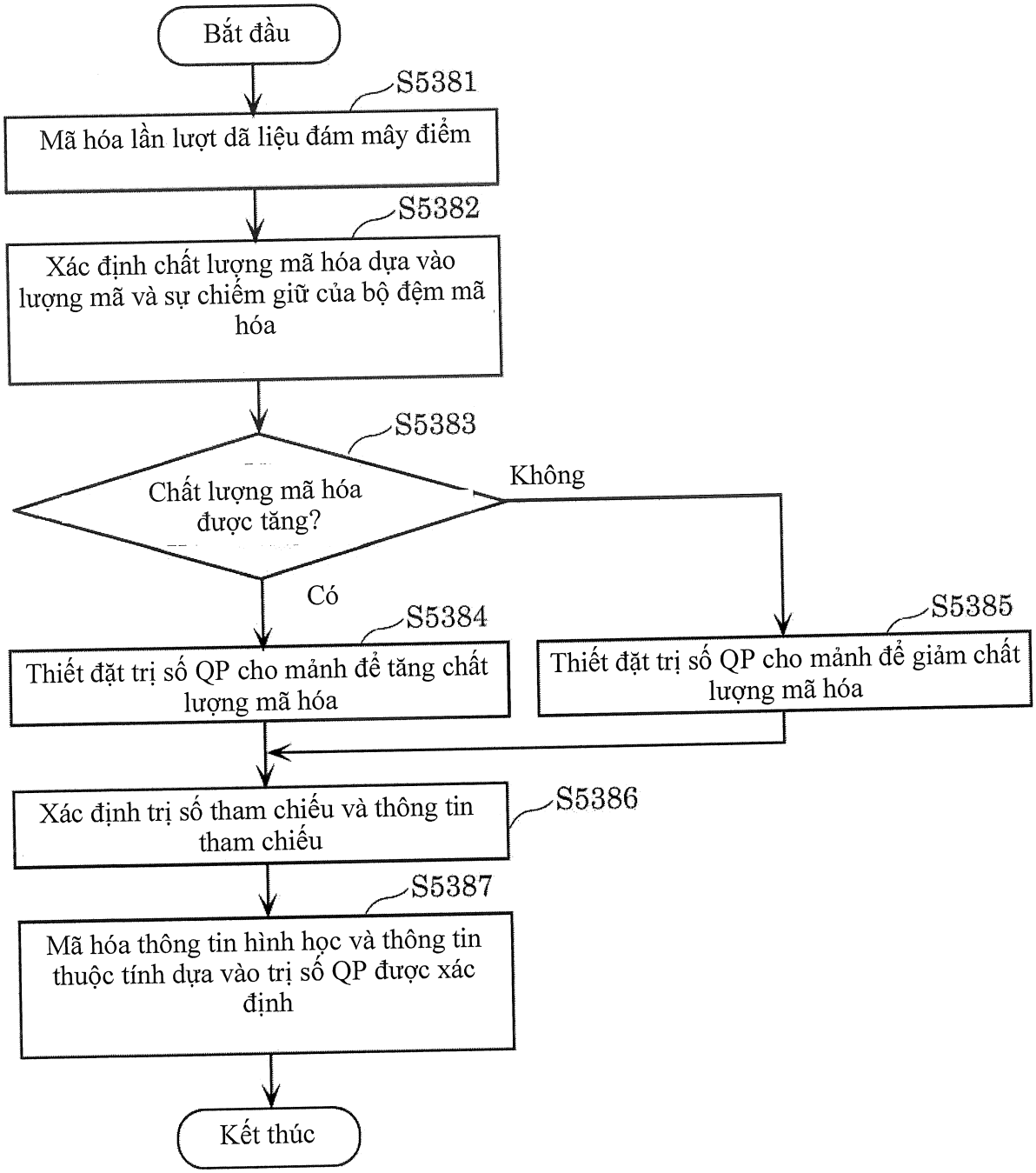

Bắt đầu
S5381
Mã hóa lần lượt dã liệu đám mây điểm
S5382
Xác định chất lượng mã hóa dựa vào lượng mã và sự chiếm giữ của bộ đệm mã hóa
S5383
Chất lượng mã hóa được tăng?
Không
Có
S5384
Thiết đặt trị số QP cho mảnh để tăng chất lượng mã hóa
S5385
Thiết đặt trị số QP cho mảnh để giảm chất lượng mã hóa
Xác định trị số tham chiếu và thông tin tham chiếu
S5386
S5387
Mã hóa thông tin hình học và thông tin thuộc tính dựa vào trị số QP được xác định
Kết thúc

FIG. 122

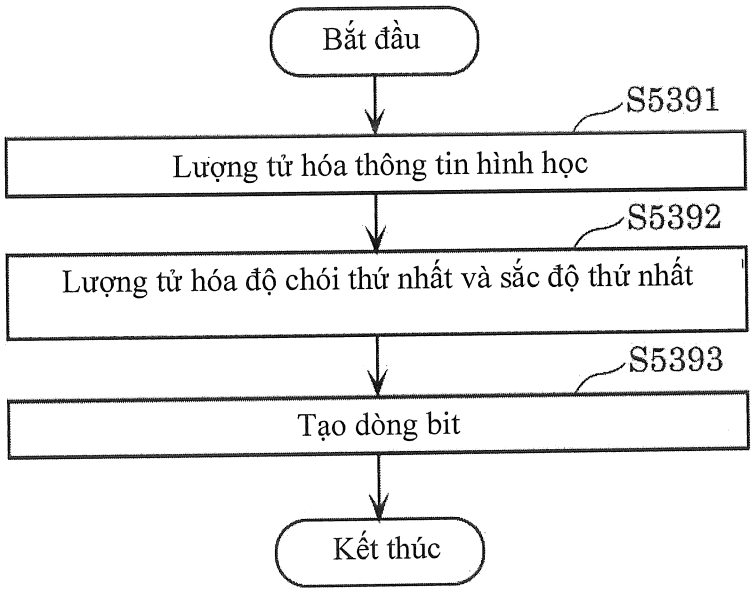


FIG. 123

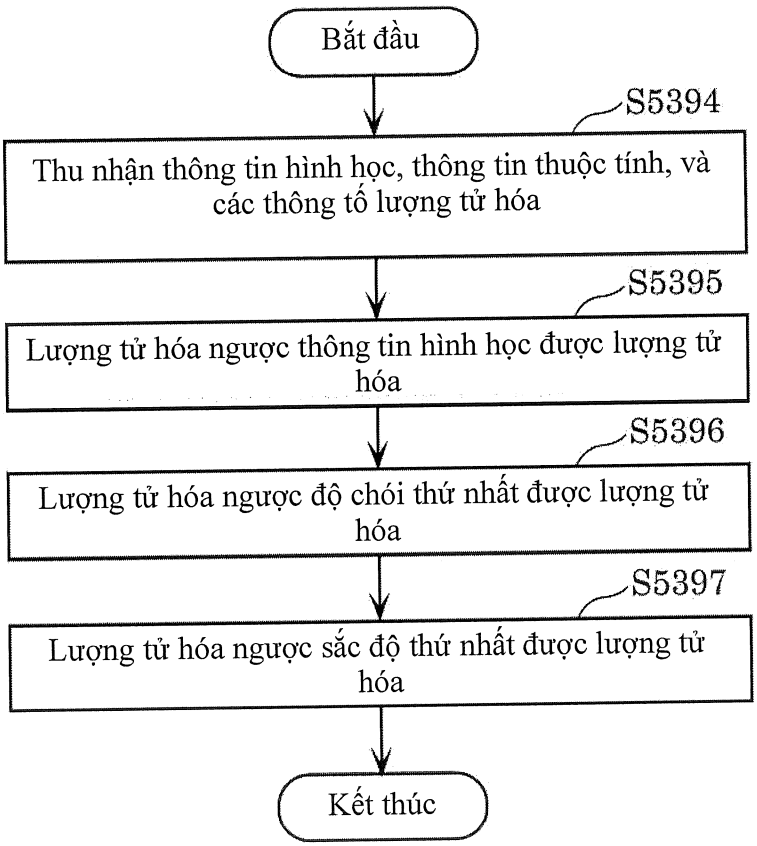


FIG. 124

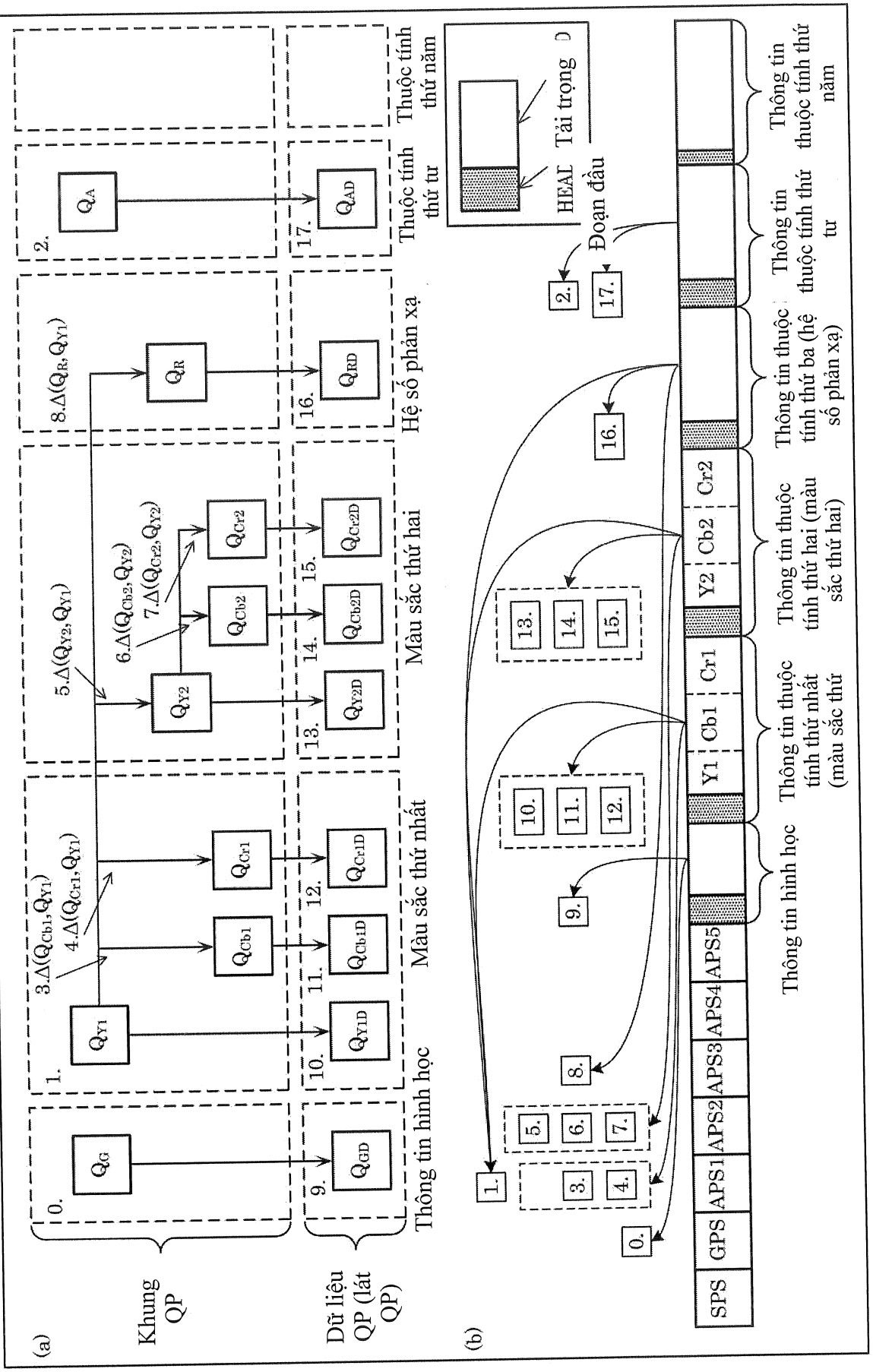

(a)
Khung QP
Dữ liệu QP (lát QP)
Thông tin hình học
Màu sắc thứ nhất
Màu sắc thứ hai
Hệ số phản xạ
Thuộc tính thứ tư
Thuộc tính thứ năm
(b)
Đoạn đầu
HEAL
Tải trọng
SPS
GPS
APS1
APS2
APS3
APS4
APS5
Y1
Cb1
Cr1
Y2
Cb2
Cr2
Thông tin hình học
Thông tin thuộc tính thứ nhất (màu sắc thứ
Thông tin thuộc tính thứ hai (màu sắc thứ hai)
Thông tin thuộc tính thứ ba (hệ số phản xạ)
Thông tin thuộc tính thứ tư
Thông tin thuộc tính thứ năm

FIG. 125

```
APS0{
  aps_idx
  sps_idx

  num_of_attribute
  for(int i=0;i>num_of_attribute ; i++){
    attribute_type  //color, reflectance
    aps_information()                          6701
    if(attribute_type==color){
      QP_value_Luma
      QP_delta_Cb_to_Luma
      QP_delta_Cr_to_Luma
      data_QP_delta_present_flag
    }else if(attribute_type==refrectance){
      QP_value
      data_QP_delta_present_flag
    }else{
      //No QP value
    }
  }
}
```

```
Attribute_header0{
  aps_idx
  sps_idx
  geom_data_id
  offset
  attribute_type                               6702
  if(data_QP_delta_present_flag){
    if(attribute_type==color){
      QP_delta_data_to_frame
      QP_delta_Cb_to_frame
      QP_delta_Cr_to_frame
    }elseif(attribute_type==refrectance){
      QP_delta_data_to_frame
    }else{
      //No QP value
    }
  }
}
```

FIG. 126

```
APS(){
  aps_idx
  sps_idx

  aps_information()
     QP_value
     QP_delta1
     QP_delta2
     data_QP_delta_present_flag
   }
}
```

FIG. 127

```
Attribute_header(){
  aps_idx
  sps_idx
  geom_data_id
  offset
  if(data_QP_delta_present_flag){
     QP_delta_data_to_frame
     QP_delta1_to_frame
     QP_delta2_to_frame
   }
}
```

FIG. 128

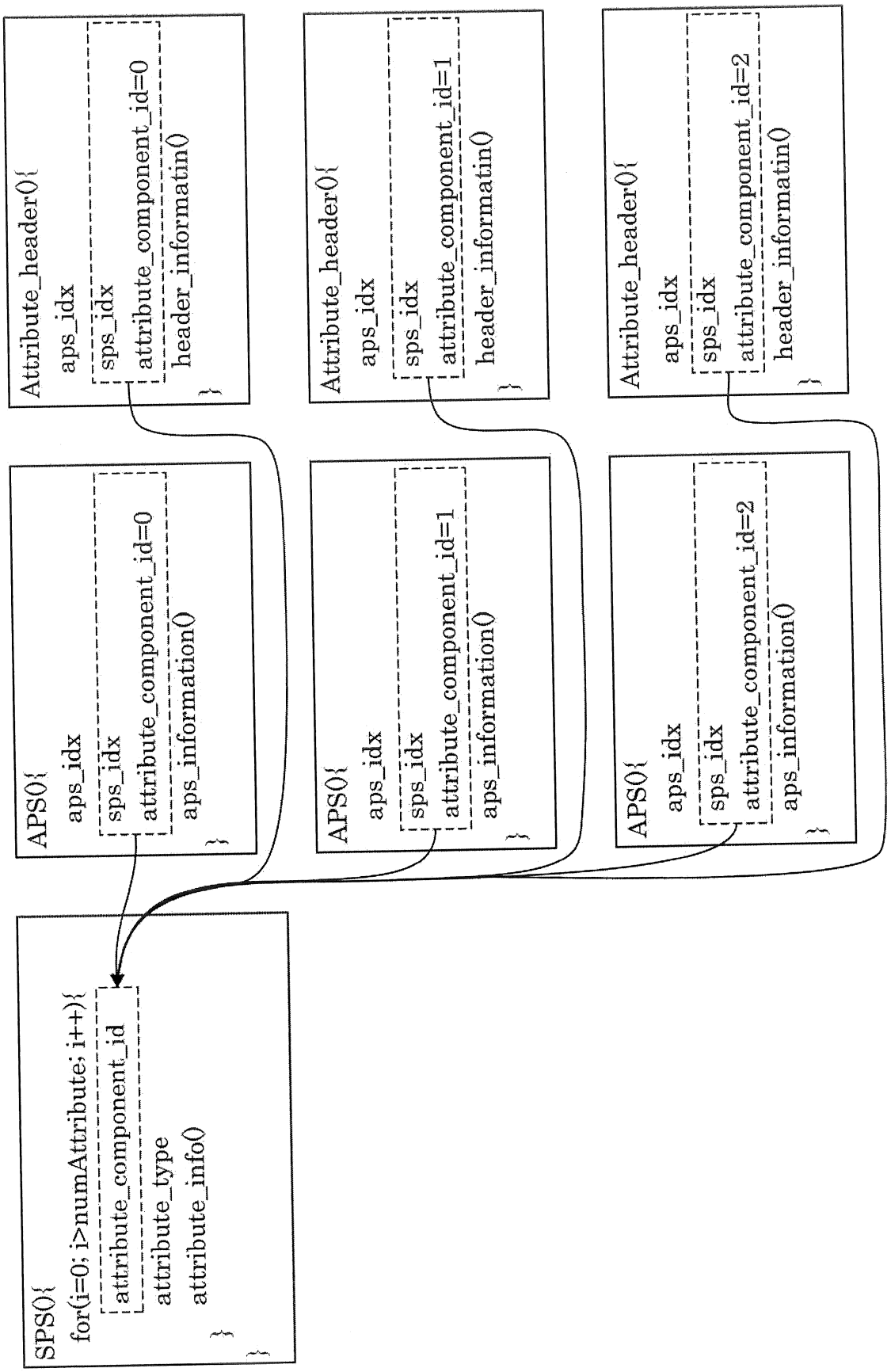

SPS0{
for(i=0; i>numAttribute; i++){
attribute_component_id
attribute_type
attribute_info()
}
}
APS0{
aps_idx
sps_idx
attribute_component_id=0
aps_information()
}
APS0{
aps_idx
sps_idx
attribute_component_id=1
aps_information()
}
APS0{
aps_idx
sps_idx
attribute_component_id=2
aps_information()
}
Attribute_header0{
aps_idx
sps_idx
attribute_component_id=0
header_informatin()
}
Attribute_header0{
aps_idx
sps_idx
attribute_component_id=1
header_informatin()
}
Attribute_header0{
aps_idx
sps_idx
attribute_component_id=2
header_informatin()
}

FIG. 129

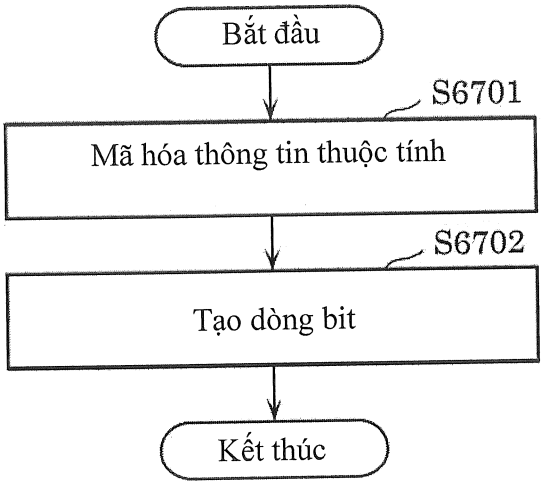


FIG. 130