



(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)



1-0044961

H04N 19/43; H04N 19/107; H04N (13) B

19/157; H04N 19/159; H04N 19/20;

(51)^{2020.01} H04N 19/103; H04N 19/44; H04N
19/50; H04N 19/51; H04N 19/52; H04N
19/60

(21) 1-2021-06245

(22) 12/03/2020

(86) PCT/US2020/022395 12/03/2020

(87) WO2020/186060 A1 17/09/2020

(30) 62/817,391 12/03/2019 US

(45) 25/04/2025 445

(43) 27/12/2021 405A

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

Huawei Administration Building, Bantian, Longgang District, Shenzhen, Guangdong
518129, P. R. China

(72) ZAKHARCHENKO, Vladyslav (UA); CHEN, Jianle (CN); CAI, Kangying (CN).

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

(54) PHƯƠNG PHÁP MÃ HÓA ĐÁM MÂY ĐIỂM ĐƯỢC THỰC HIỆN BỞI BỘ GIẢI
MÃ, PHƯƠNG PHÁP MÃ HÓA ĐÁM MÂY ĐIỂM ĐƯỢC THỰC HIỆN BỞI BỘ
MÃ HÓA, THIẾT BỊ GIẢI MÃ VÀ THIẾT BỊ MÃ HÓA

(21) 1-2021-06245

(57) Sáng chế đề cập đến phương pháp mã hóa đám mây điểm được thực hiện bởi bộ giải mã, phương pháp mã hóa đám mây điểm được thực hiện bởi bộ mã hóa, thiết bị giải mã và thiết bị mã hóa. Phương pháp mã hóa đám mây điểm (PCC) bao gồm thu dữ liệu thông tin phần vá được mã hóa; thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá có kiểu phần vá; xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá cuối; và kết thúc xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối. Một phương pháp khác bao gồm các bước thu và thu nhận giống nhau, nhưng xác định xem kiểu phần vá dùng cho phần vá có là kiểu phần vá không liên tục; giải mã chỉ số phần vá tham chiếu tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục; xác định chỉ số tham chiếu dùng cho phần vá dựa trên chỉ số khung tham chiếu tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; và khôi phục sự biểu diễn thể tích của phần vá.

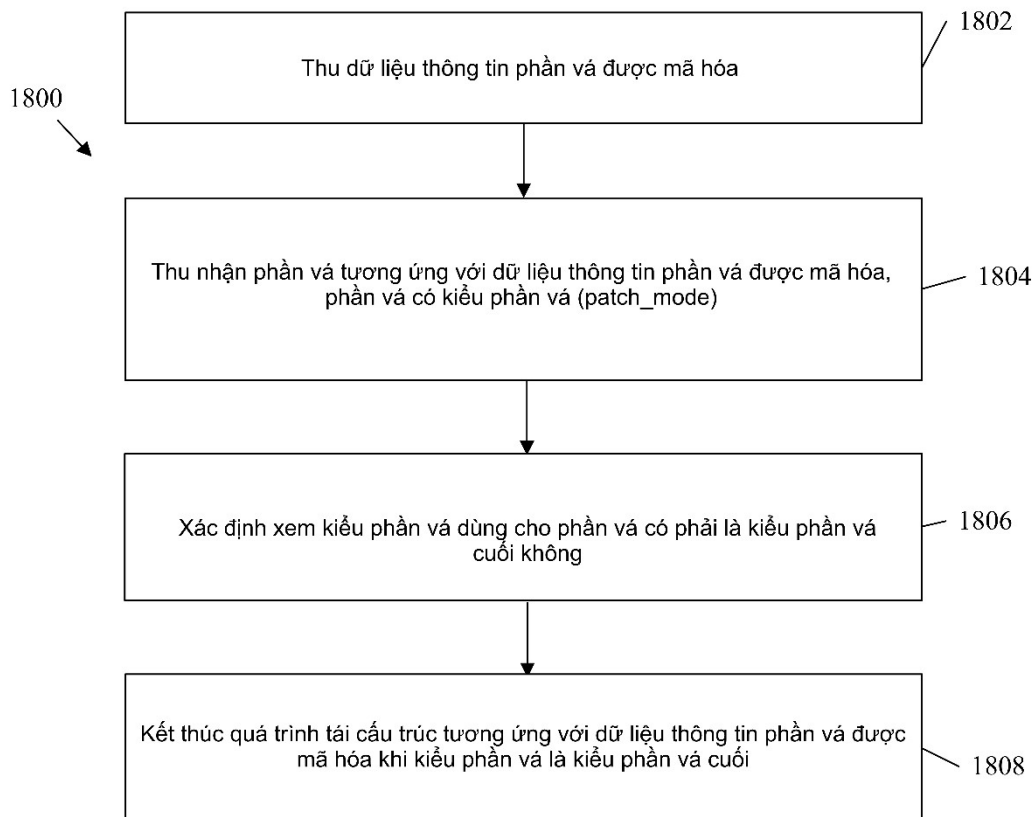


FIG. 18

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến mã hóa đám mây điểm, và cụ thể là đến cú pháp bậc cao cho việc mã hóa đám mây điểm.

Tình trạng kỹ thuật của sáng chế

Đám mây điểm được sử dụng trong rất nhiều ứng dụng bao gồm nền công nghiệp giải trí, định vị ô tô thông minh, theo dõi không gian vật lý, lập mô hình ba chiều (3D) của các đối tượng thế giới, sự biểu diễn trực quan v.v..Xem xét rằng vị trí mà lấy mẫu không thống nhất của đám mây điểm, sự thể hiện nén để lưu trữ và truyền của dữ liệu này là hữu ích. So sánh với các thể hiện 3D khác, đám mây điểm bất thường là phổ biến và áp dụng được cho phạm vi rộng hơn của các cảm biến và cách thu nhận dữ liệu. Ví dụ, khi thực hiện thể hiện 3D trong thế giới thực ảo hoặc việc kết xuất từ xa trong môi trường hiện diện từ xa, việc kết xuất của các hình vẽ ảo và các lệnh máy thời gian thực được xử lý như tập dữ liệu đám mây có mật độ cao.

Bản chất kỹ thuật của sáng chế

Khía cạnh thứ nhất đề cập đến phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ giải mã. Phương pháp bao gồm thu, bởi bộ phận thu của bộ giải mã, dữ liệu thông tin phần vá được mã hóa; thu nhận, bởi bộ phận xử lý của bộ giải mã, phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá có kiểu phần vá (patch_mode); xác định, bởi bộ phận xử lý, xem kiểu phần vá dùng cho phần vá có là kiểu phần vá cuối không; và kết thúc, bởi bộ phận xử lý, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn

vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Khía cạnh thứ hai đề cập đến phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ giải mã. Phương pháp bao gồm thu, bởi bộ phận thu của bộ giải mã, dữ liệu thông tin phần vá được mã hóa; thu nhận, bởi bộ phận xử lý của bộ giải mã, phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá mà có kiểu phần vá (patch_mode); xác định, bởi bộ phận xử lý, để xem kiểu phần vá dùng cho phần vá là kiểu phần vá không liên tục; giải mã, bởi bộ phận xử lý, chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục; xác định, bởi bộ phận xử lý, chỉ số tham chiếu (refIdx) cho phần vá

dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; và khôi phục, bởi bộ phận xử lý, biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của phương pháp theo khía cạnh thứ hai, phương pháp còn bao gồm xác định rằng kiểu phần vá cho phần vá là kiểu phần vá nội bộ; xác định chỉ số tham chiếu (`refIdx`) cho phần vá dựa trên chỉ số khung (`[frmIdx]`) tương ứng với phần vá và chỉ số phần vá giảm dần; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ hai của phương pháp theo khía cạnh thứ hai hoặc dạng thực hiện bất kỳ nêu trên của khía cạnh thứ hai, phương pháp còn bao gồm xác định rằng kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới; giải mã chỉ số phần vá tham chiếu thứ hai (`dpdu_patch_index`) tương ứng với phần vá; xác định chỉ số tham chiếu (`refIdx`) dùng cho phần vá dựa trên chỉ số khung tham chiếu (`[refFrmIdx]`) tương ứng với phần vá và chỉ số phần vá tham chiếu thứ hai như được giải mã; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ ba của phương pháp theo khía cạnh thứ hai hoặc dạng thực hiện bất kỳ nêu trên của khía cạnh thứ hai, phương pháp còn bao gồm xác định rằng kiểu phần vá dùng cho phần vá là kiểu phần vá điều chế mã xung (PCM); xác định chỉ số tham chiếu (`refIdx`) cho phần vá dựa trên chỉ số khung (`[frmIdx]`) tương ứng với phần vá và chỉ số phần vá giảm dần; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D) cho các điểm độc lập tương ứng với phần vá và các thành phần ba chiều (3D) cho các điểm độc lập tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ tư của phương pháp theo khía cạnh thứ hai hoặc dạng thực hiện bất kỳ nêu trên của khía cạnh thứ hai, phương pháp còn bao gồm thu đầu vào tương ứng với dữ liệu thông tin phần vá được mã hóa, trong đó đầu vào bao gồm một hoặc nhiều trong số `patch_mode`, chỉ số phần vá, chỉ số tham chiếu, chỉ số khung, và chỉ số khung tham chiếu.

Trong dạng thực hiện thứ tư của phương pháp theo khía cạnh thứ hai hoặc dạng thực hiện bất kỳ nêu trên của khía cạnh thứ hai, phương pháp còn bao gồm hiển thị, trên màn hiển thị của thiết bị điện tử, hình ảnh được tạo ra dựa trên sự biểu diễn thể tích khi được khôi phục.

Khía cạnh thứ ba đề cập đến phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ giải mã. Phương pháp bao gồm thu, bởi bộ phận thu của bộ giải mã, dữ liệu thông tin phần vá được mã hóa; thu nhận, bởi bộ phận xử lý của bộ giải mã, phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá có kiểu phần vá (`patch_mode`); xác định, bởi bộ phận xử lý, để xem kiểu phần vá dùng cho phần vá có là kiểu phần vá không liên tục hay không; giải mã, bởi bộ phận xử lý, chỉ số phần vá tham chiếu (`spdu_patch_index`) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục; xác định, bởi bộ phận xử lý, chỉ số tham chiếu (`refIdx`) dùng cho phần vá dựa trên chỉ số khung tham chiếu (`[refFrmIdx]`) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; khôi phục, bởi bộ phận xử lý, sự biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục; xác định, bởi bộ phận xử lý, để xem cờ khả dụng của nhiều phần vá có giá trị thứ nhất hoặc giá trị thứ hai; lưu trữ, trong bộ nhớ của bộ giải mã, sự biểu diễn thể tích khi được khôi phục khi cờ khả dụng của nhiều phần vá có giá trị thứ nhất; và kết thúc, bởi bộ phận xử lý, xử lý khôi phục của dữ liệu thông tin phần vá được mã hóa khi cờ khả dụng của nhiều phần vá có giá trị thứ hai.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (`spdu`) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham

chiều khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của phương pháp theo khía cạnh thứ ba, phương pháp còn bao gồm xác định rằng kiểu phần vá cho phần vá là kiểu phần vá nội bộ; xác định chỉ số tham chiếu (`refIdx`) cho phần vá dựa trên chỉ số khung (`[frmIdx]`) tương ứng với phần vá và chỉ số phần vá giảm dần; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ hai của phương pháp theo khía cạnh thứ ba, phương pháp còn bao gồm xác định rằng kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới; giải mã chỉ số phần vá tham chiếu thứ hai (`dpdu_patch_index`) tương ứng với phần vá; xác định chỉ số tham chiếu (`refIdx`) dùng cho phần vá dựa trên chỉ số khung tham chiếu (`[refFrmIdx]`) tương ứng với phần vá và chỉ số phần vá tham chiếu thứ hai như được giải mã; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D)

tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ ba của phương pháp theo khía cạnh thứ ba, phương pháp còn bao gồm xác định rằng kiểu phần vá dùng cho phần vá là kiểu phần vá điều chế mã xung (PCM); xác định chỉ số tham chiếu (`refIdx`) cho phần vá dựa trên chỉ số khung (`[frmIdx]`) tương ứng với phần vá và chỉ số phần vá giảm dần; giải mã phần vá nhờ sử dụng các thành phần hai chiều (2D) cho các điểm độc lập tương ứng với phần vá và các thành phần ba chiều (3D) cho các điểm độc lập tương ứng với phần vá; và khôi phục sự biểu diễn thể tích dựa trên phần vá như được giải mã.

Trong dạng thực hiện thứ tư của phương pháp theo khía cạnh thứ ba, phương pháp còn bao gồm thu đầu vào tương ứng với dữ liệu thông tin phần vá được mã hóa, trong đó đầu vào bao gồm một hoặc nhiều trong số `patch_mode`, chỉ số phần vá, chỉ số tham chiếu, chỉ số khung, và chỉ số khung tham chiếu.

Trong dạng thực hiện thứ năm của phương pháp theo khía cạnh thứ năm, phương pháp còn bao gồm hiển thị, trên màn hiển thị của thiết bị điện tử, hình ảnh được tạo ra dựa trên sự biểu diễn thể tích khi được khôi phục.

Khía cạnh thứ tư đề cập đến phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ mã hóa. Phương pháp bao gồm thu nhận, bởi bộ phận thu của bộ mã hóa, đơn vị dữ liệu khung phần vá (`pdfu`) mà nhận dạng kiểu phần vá (`pdfu_patch_mode`) cho mỗi trong số nhiều phần vá; xác định, bởi bộ phận xử lý của bộ mã hóa, xem kiểu phần vá dùng cho phần vá từ nhiều phần vá có là kiểu phần vá cuối hay không; mã hóa, bởi bộ phận xử lý của bộ mã hóa, dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá không là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá dùng cho phần vá; và mã hóa, bởi bộ phận xử lý của bộ mã hóa, dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá được thiết đặt cho kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá cuối dùng cho phần vá.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau

(ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của phương pháp theo khía cạnh thứ tư, kiểu phần vá là một trong các kiểu phần vá không liên tục, kiểu phần vá liên đới, kiểu phần vá nội bộ, và kiểu phần vá điều chế mã xung (PCM).

Trong dạng thực hiện thứ hai của phương pháp theo khía cạnh thứ tư, đơn vị dữ liệu khung phần vá bao gồm chỉ số khung (frmIdx), các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

Trong dạng thực hiện thứ ba của phương pháp theo khía cạnh thứ tư, dữ liệu thông tin phần vá bao gồm chỉ số khung, các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

Trong dạng thực hiện thứ tư của phương pháp theo khía cạnh thứ tư, phương pháp còn bao gồm lặp lại các bước của việc xác định xem kiểu phần vá có là kiểu phần vá cuối hay không và mã hóa dữ liệu thông tin phần vá dùng cho các phần vá kế tiếp từ nhiều phần vá cho đến khi một trong các phần vá từ nhiều phần vá được xác định có kiểu phần vá cuối.

Trong dạng thực hiện thứ năm của phương pháp theo khía cạnh thứ tư, phương pháp còn bao gồm thực hiện căn chỉnh byte sau khi tất cả dữ liệu thông tin phần vá được mã hóa.

Trong dạng thực hiện thứ sáu của phương pháp theo khía cạnh thứ tư, phương pháp còn bao gồm tạo đơn vị dữ liệu khung phần vá được nén sau khi tất cả dữ liệu thông tin phần vá được mã hóa.

Trong dạng thực hiện thứ bảy của phương pháp theo khía cạnh thứ tư, phương pháp còn bao gồm lưu trữ, trong bộ nhớ của bộ mã hóa, đơn vị dữ liệu khung phần vá được nén cho việc truyền tới bộ giải mã.

Khía cạnh thứ năm đề cập đến phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ mã hóa. Phương pháp bao gồm thu nhận, bởi bộ phận thu của bộ mã hóa, đơn vị dữ liệu khung phần vá (pfdu) dùng cho mỗi trong số nhiều phần vá; thêm, bởi bộ phận xử lý của bộ mã hóa, cờ phần vá cuối cho mỗi trong số nhiều phần vá; xác định, bởi bộ phận xử lý của bộ mã hóa, để xem kiểu phần vá dùng cho phần vá từ nhiều phần vá có là kiểu phần vá cuối hay không dựa trên giá trị của cờ phần vá cuối; và mã hóa, bởi bộ phận xử lý của bộ mã hóa, dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá không là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá và cờ phần vá cuối dùng cho phần vá.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn

vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của phương pháp theo khía cạnh thứ năm, kiểu phần vá là một trong các kiểu phần vá không liên tục, kiểu phần vá liên đới, kiểu phần vá nội bộ, và kiểu phần vá điều chế mã xung (PCM).

Trong dạng thực hiện thứ hai của phương pháp theo khía cạnh thứ năm, đơn vị dữ liệu khung phần vá bao gồm chỉ số khung (frmIdx), các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

Trong dạng thực hiện thứ ba của phương pháp theo khía cạnh thứ năm, dữ liệu thông tin phần vá bao gồm chỉ số khung, các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

Trong dạng thực hiện thứ tư của phương pháp theo khía cạnh thứ năm, phương pháp còn bao gồm lặp lại các bước xác định xem kiểu phần vá có là kiểu phần vá cuối hay không dựa trên giá trị của cờ phần vá cuối và mã hóa dữ liệu thông tin phần vá cho các phần vá kế tiếp từ nhiều phần vá cho đến khi một trong các phần vá từ nhiều phần vá được xác định có kiểu phần vá cuối dựa trên giá trị của cờ phần vá cuối.

Trong dạng thực hiện thứ năm của phương pháp theo khía cạnh thứ năm, phương pháp còn bao gồm thực hiện căn chỉnh byte sau khi một trong các phần vá từ nhiều phần vá được xác định có kiểu phần vá cuối dựa trên giá trị của cờ phần vá cuối.

Trong dạng thực hiện thứ sáu của phương pháp theo khía cạnh thứ năm, phương pháp còn bao gồm tạo đơn vị dữ liệu khung phần vá được nén sau khi tất cả dữ liệu thông tin phần vá được mã hóa.

Trong dạng thực hiện thứ bảy của phương pháp theo khía cạnh thứ năm, phương pháp còn bao gồm lưu trữ, trong bộ nhớ của bộ mã hóa, đơn vị dữ liệu khung phần vá được nén cho việc truyền tới bộ giải mã.

Khía cạnh thứ sáu đề cập đến thiết bị giải mã bao gồm bộ phận thu được tạo cấu hình để thu thông tin phần vá được mã hóa; bộ nhớ được ghép nối với bộ phận thu, bộ nhớ lưu trữ các lệnh máy; và bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực thi các lệnh máy khiến cho thiết bị giải mã: thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá mà có kiểu phần vá (patch_mode); xác định xem kiểu phần vá dùng cho phần vá có là kiểu phần vá cuối hay không; kết thúc xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn

vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Khía cạnh thứ bảy đề cập đến thiết bị giải mã bao gồm bộ phận thu được tạo cấu hình để thu thông tin phần vá được mã hóa; bộ nhớ được ghép nối với bộ phận thu, bộ nhớ lưu trữ các lệnh máy; và bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực thi các lệnh máy khiến cho thiết bị giải mã: thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá có kiểu phần vá (patch_mode); xác định xem kiểu phần vá dùng cho phần vá có là kiểu phần vá không liên tục hay không; giải mã chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục; xác định chỉ

số tham chiếu (refIdx) dùng cho phần vá dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; và khôi phục sự biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục.

Sử dụng thiết bị giải mã này cho phép việc xếp thứ tự linh hoạt của các phần vá trong bộ tạo đệm phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong bộ tạo đệm phần vá theo thứ tự bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc được khớp hoàn hảo. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo danh sách các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong bộ tạo đệm phần vá. Thêm vào đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong bộ tạo đệm phần vá (ví dụ như, kích cỡ của bộ tạo đệm phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của bộ tạo đệm phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa video được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa video được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các video được gửi, được thu, và/hoặc được quan sát.

Khía cạnh thứ tám đề cập đến thiết bị giải mã bao gồm bộ phận thu được tạo cấu hình để thu thông tin phần vá được mã hóa; bộ nhớ được ghép nối với bộ phận thu, bộ nhớ lưu trữ các lệnh máy; và bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực thi các lệnh máy khiến cho thiết bị giải mã: thu nhận

phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá có kiểu phần vá (patch_mode); xác định xem kiểu phần vá dùng cho phần vá có là kiểu phần vá không liên tục hay không; giải mã chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục; xác định chỉ số tham chiếu (refIdx) dùng cho phần vá dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; khôi phục sự biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục; xác định xem cờ khả dụng của nhiều phần vá có giá trị thứ nhất hay giá trị thứ hai; lưu trữ sự biểu diễn thể tích khi được khôi phục trong bộ nhớ khi cờ khả dụng của nhiều phần vá có giá trị thứ nhất; và kết thúc xử lý khôi phục của dữ liệu thông tin phần vá được mã hóa khi cờ khả dụng của nhiều phần vá có giá trị thứ hai.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần

cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của thiết bị giải mã theo các khía cạnh thứ sáu, thứ bảy, và thứ tám, thiết bị giải mã còn bao gồm màn hiển thị được cấu hình để hiển thị hình ảnh được tạo ra dựa trên sự biểu diễn thể tích khi được khôi phục.

Khía cạnh thứ chín đề cập đến thiết bị mã hóa, bao gồm: bộ phận thu được tạo cấu hình để thu hình ảnh ba chiều (3D); bộ nhớ được ghép nối với bộ phận thu, bộ nhớ chứa các lệnh máy; bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực hiện các lệnh máy khiến cho thiết bị mã hóa: thu nhận đơn vị dữ liệu khung phần vá (pdfu) mà nhận dạng kiểu phần vá (pdfu_patch_mode) cho mỗi trong số nhiều phần vá; xác định xem kiểu phần vá dùng cho phần vá từ nhiều phần vá có là kiểu phần vá cuối hay không; mã hóa dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá không là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá dùng cho phần vá; và mã hóa dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá được thiết đặt là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá cuối dùng cho phần vá.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo

rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Khía cạnh thứ mười đề cập đến thiết bị mã hóa, bao gồm: bộ phận thu được tạo cấu hình để thu hình ảnh ba chiều (3D); bộ nhớ được ghép nối với bộ phận thu, bộ nhớ chứa các lệnh máy; bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực hiện các lệnh máy khiến cho thiết bị mã hóa: thu nhận đơn vị dữ liệu khung phần vá (pfdu) cho mỗi trong số nhiều phần vá; thêm cờ phần vá cuối cho mỗi trong số nhiều phần vá; xác định xem kiểu phần vá dùng cho phần vá từ nhiều phần vá có là kiểu phần vá cuối hay không dựa trên giá trị của cờ phần vá cuối; và mã hóa dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá không là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá và cờ phần vá cuối dùng cho phần vá.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn

vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của thiết bị mã hóa theo các khía cạnh thứ chín và thứ mười, thiết bị mã hóa còn bao gồm bộ phận truyền được ghép nối với bộ phận xử lý, bộ phận truyền được tạo cấu hình để truyền dòng bit có dữ liệu thông tin phần vá khi được mã hóa tới bộ giải mã.

Khía cạnh thứ mười một đề cập đến thiết bị mã hóa, bao gồm: bộ phận thu được tạo cấu hình để thu hình ảnh thể tích để mã hóa hoặc thu dòng bit để giải mã; bộ phận truyền được ghép nối với bộ phận thu, bộ phận truyền được tạo cấu hình để truyền dòng bit đến bộ giải mã hoặc để truyền hình ảnh thể tích được giải mã đến thiết

bị khôi phục được tạo cấu hình để khôi phục hình ảnh thể tích được giải mã; bộ nhớ được ghép nối với ít nhất một của bộ phận thu hoặc bộ phận truyền, bộ nhớ được tạo cấu hình để lưu trữ các lệnh máy; và bộ phận xử lý được ghép nối với bộ nhớ, bộ phận xử lý được tạo cấu hình để thực thi các lệnh máy được lưu trữ trong bộ nhớ để thực hiện bất kỳ phương pháp nào được bộc lộ ở đây.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Trong dạng thực hiện thứ nhất của thiết bị mã hóa theo khía cạnh thứ năm, thiết bị mã hóa còn bao gồm màn hiển thị được cấu hình để hiển thị hình ảnh dựa trên phân vá khi được khôi phục.

Khía cạnh thứ mười hai đề cập đến hệ thống, bao gồm: bộ mã hóa; và bộ giải mã truyền thông với bộ mã hóa, trong đó bộ mã hóa hoặc bộ giải mã bao gồm thiết bị mã hóa, thiết bị giải mã, hoặc thiết bị mã hóa được mô tả ở đây.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể

nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Khía cạnh thứ mười ba đề cập đến phương tiện mã hóa, bao gồm phương tiện thu được tạo cấu hình để thu hình ảnh thể tích để mã hóa hoặc để thu dòng bit để giải mã, khôi phục, và chiếu; phương tiện truyền được ghép nối với phương tiện thu, phương tiện truyền được tạo cấu hình để truyền dòng bit đến bộ giải mã hoặc để truyền hình ảnh được giải mã đến các phương tiện hiển thị; phương tiện lưu trữ được ghép nối với ít nhất một trong số phương tiện thu hoặc phương tiện truyền, the phương tiện lưu trữ được tạo cấu hình để lưu trữ các lệnh máy; và phương tiện xử lý được ghép nối với phương tiện lưu trữ, các phương tiện xử lý được tạo cấu hình để thực thi các lệnh máy được lưu trữ trong phương tiện lưu trữ để thực hiện phương pháp bất kỳ được bộc lộ ở đây.

Sử dụng phương pháp mã hóa này dùng cho thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Ngoài ra, các phần vá bổ sung có thể được thêm vào đơn vị dữ liệu khung phần vá. Các phần vá được thêm vào có thể được thêm vào hoặc ở phần cuối của đơn vị dữ liệu khung phần vá, hoặc ở vị trí chỉ số phần vá ngẫu nhiên bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc trong khoảng dung sai cho phép. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Bên cạnh đó, các kỹ thuật mã hóa loại trừ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay thế xử lý đó bằng kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần

cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Nhiều phần vá PCM cũng được cho phép. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa đám mây điểm được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa đám mây điểm được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

Nhằm mục đích làm rõ, bất kỳ một trong số các phương án nêu trên có thể được kết hợp với bất kỳ một hoặc nhiều phương án trong số các phương án nêu trên khác để tạo ra phương án mới nằm trong phạm vi của sáng chế.

Các đặc điểm này và các đặc điểm khác sẽ được hiểu rõ ràng hơn từ phần mô tả chi tiết sau đây đi kèm với các hình vẽ kèm theo và bộ yêu cầu bảo hộ.

Mô tả vắn tắt các hình vẽ

Để hiểu rõ hơn về sáng chế này, sự tham chiếu được thực hiện cho phần mô tả vắn tắt sau đây, đi kèm với các hình vẽ kèm theo và phần mô tả chi tiết, trong đó các số tham chiếu tương tự thể hiện các phần tương tự.

FIG.1 là sơ đồ khối minh họa hệ thống mã hóa ví dụ mà có thể sử dụng các kỹ thuật mô hình hóa bối cảnh.

FIG.2 sơ đồ khối minh họa bộ mã hóa ví dụ mà có thể thực hiện các kỹ thuật mô hình hóa bối cảnh.

FIG.3 sơ đồ khối minh họa bộ giải mã ví dụ mà có thể thực hiện các kỹ thuật mô hình hóa bối cảnh.

FIG.4 là sự thể hiện của chuỗi của các khung đám mây điểm mà từng khung chứa đám mây điểm ba chiều (3D).

FIG.5 là sự thể hiện của một trong đám mây điểm 3D của FIG.5 được chiếu lên hộp giới hạn để tạo ra các phép chiếu hai chiều (2D).

Các FIG.6A-6C là sự thể hiện của ánh xạ chiếm giữ, ánh xạ hình học, và ánh xạ thuộc tính tương ứng với các phép chiếu 2D từ hộp giới hạn của FIG.5.

FIG.7 là sự thể hiện của dòng bit mã hóa đám mây điểm dựa trên video (V-PCC).

FIG.8 là sự thể hiện của sự phân giải được mã hóa của chuỗi đám mây điểm cho các khung.

FIG.9 là sự thể hiện của đơn vị dữ liệu phần vá đơn nhất được trích xuất từ một trong các khung đám mây điểm trong chuỗi đám mây điểm của FIG.8.

Các FIG.10A-10C là sự thể hiện của ánh xạ chiếm giữ, ánh xạ hình học, và ánh xạ thuộc tính tương ứng với đơn vị dữ liệu phần vá đơn nhất của FIG.9.

FIG.11 là sự thể hiện của các thành phần 2D dùng cho đơn vị dữ liệu phần vá trong hộp giới hạn 2D.

Các FIG.12A-12B là sự thể hiện giản lược và sự thể hiện tập dữ liệu của các thành phần 3D dùng cho đơn vị dữ liệu phần vá của FIG.11.

FIG.13 là phương án của xử lý mã hóa đơn vị dữ liệu khung phần vá.

FIG.14 là phương án của xử lý mã hóa đơn vị dữ liệu khung phần vá bằng cách sử dụng cờ phần vá cuối.

FIG.15 là phương án của khung dữ liệu thông tin phần vá với sự mô tả của các loại đơn vị phần vá.

FIG.16 là phương án của xử lý giải mã đơn vị dữ liệu khung phần vá.

FIG.17 là phương án của xử lý giải mã đơn vị dữ liệu khung phần vá bằng cách sử dụng cờ phần vá cuối.

FIG.18 là phương án của phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ giải mã.

FIG.19 là phương án của phương pháp PCC được thực hiện bởi bộ giải mã.

FIG.20 là phương án của phương pháp PCC được thực hiện bởi bộ giải mã.

FIG.21 là phương án của phương pháp PCC được thực hiện bởi bộ mã hóa.

FIG.22 là phương án của phương pháp PCC được thực hiện bởi bộ mã hóa.

FIG.23 là sơ đồ giản lược của thiết bị mã hóa.

FIG.24 là sơ đồ giản lược của phương án của các phương tiện mã hóa.

Mô tả chi tiết sáng chế

Cần hiểu ngay từ đầu rằng mặc dù cách thực hiện mang tính minh họa của một hoặc nhiều phương án được đưa ra dưới đây, các hệ thống và/hoặc các phương pháp được bộc lộ có thể được thực hiện bằng cách sử dụng bất kỳ số lượng kỹ thuật nào, để xem đã biết hay đang có. Phần bộc lộ không nên được giới hạn ở các cách thực hiện mang tính minh họa, các hình vẽ, và các kỹ thuật được minh họa dưới đây, bao gồm các phương án ví dụ và các cách thực hiện được minh họa và được mô tả ở đây, nhưng có thể được chỉnh sửa trong phạm vi của các yêu cầu bảo hộ kèm theo cùng với phạm vi tương đương đầy đủ của chúng.

Trong suốt bản mô tả và yêu cầu bảo hộ, các thuật ngữ sau đây có ý nghĩa được liên kết rõ ràng ở đây, trừ khi bối cảnh được đưa ra rõ ràng. Chỉ số phần vá tham chiếu (ví dụ như, `spdu_patch_index`) chỉ báo độ chênh lệch giữa chỉ số của đơn vị dữ liệu phần vá hiện tại `p` [trong đơn vị dữ liệu khung phần vá hiện tại] và chỉ số của đơn vị dữ liệu phần vá tham chiếu [trong đơn vị dữ liệu khung phần vá tham chiếu]. Chỉ số tham chiếu (ví dụ như, `refIdx`) chỉ báo chỉ số thực tế của đơn vị dữ liệu phần vá tham chiếu trong đơn vị dữ liệu phần vá được giải mã trước đó [trong bộ đệm đơn vị dữ liệu phần vá được giải mã]. Chỉ số khung tham chiếu (ví dụ như, `refFrmIdx`) chỉ báo chỉ số của đơn vị dữ liệu phần vá được giải mã trước đó [trong bộ đệm đơn vị dữ liệu phần vá được giải mã].

Các tiêu chuẩn mã hóa video bao gồm Lĩnh vực tiêu chuẩn hóa viễn thông của Liên minh Viễn thông Quốc tế (ITU-T) H.261, Tổ chức Quốc tế về Tiêu chuẩn hoá (ISO) / Nhóm chuyên gia ảnh động (MPEG) của Ủy ban kỹ thuật điện quốc tế (IEC) -1 Phần 2, ITU-T H.262 hoặc ISO/IEC MPEG-2 Phần 2, ITU-T H.263, ISO/IEC MPEG-4 Phần 2, Mã hóa video nâng cao (AVC), cũng được biết đến là ITU-T H.264 hoặc ISO/IEC MPEG-4 Part 10, và Mã hóa video hiệu quả cao (HEVC), cũng được biết đến là ITU-T H.265 hoặc MPEG-H Phần 2. AVC bao gồm các phần mở rộng như là Mã hóa video tính tỷ lệ được (SVC), Mã hóa video đa góc nhìn (MVC), và Mã hóa

video đa góc nhìn cộng với độ sâu (MVC+D), và 3D AVC (3D-AVC). HEVC bao gồm các phần mở rộng như là HEVC lấy tỷ lệ được (SHVC), HEVC đa góc nhìn (MV-HEVC), và 3D HEVC (3D-HEVC).

Đám mây điểm là tập hợp các điểm dữ liệu trong không gian 3D. Mỗi điểm dữ liệu bao gồm các tham số mà xác định vị trí (ví dụ như, X, Y, Z), màu sắc (ví dụ như, R, G, B hoặc Y, U, V), và có thể là các đặc tính khác như độ trong suốt, độ phản chiếu, thời gian thu nhận, v.v. Thông thường, mỗi điểm trong đám mây có cùng số lượng thuộc tính được đính kèm. Các đám mây điểm có thể được sử dụng trong các ứng dụng khác nhau như là dịch chuyển tức thời 3D trong thời gian thực, chế độ xem nội dung thực tế ảo (VR) bằng thị sai tương tác, phát rộng phát lại thể thao góc nhìn miễn phí 3D, các hệ thống thông tin địa lý, di sản văn hóa, điều hướng tự động dựa trên các ảnh xạ động 3D tỷ lệ lớn, và các ứng dụng tự động.

Nhóm chuyên gia hình ảnh chuyển động ISO / IEC (MPEG) bắt đầu nghiên cứu tiêu chuẩn mã hóa-giải mã mới trên Mã hóa đám mây điểm cho dữ liệu đám mây điểm được nén tổn hao và không tổn hao với hiệu quả mã hóa cơ bản và tính bền vững cho các môi trường mạng vào năm 2016. Việc sử dụng tiêu chuẩn mã hóa-giải mã này cho phép các đám mây điểm được điều khiển như dạng dữ liệu máy tính và được lưu trữ trên phương tiện lưu trữ khác, được truyền và thu qua các mạng đang tồn tại và mạng tương lai và được phân phối trên các kênh phát rộng đang tồn tại và trong tương lai.

Hiện nay, công việc mã hóa đám mây điểm (PCC) đã được phân chia thành ba danh mục, danh mục PCC 1, danh mục PCC 2, và danh mục PCC 3, trong đó hai bản thảo thực hiện riêng biệt được phát triển, một cho danh mục PCC 2 (PCC Cat2), và còn lại cho các danh mục PCC 1 và 3 (PCC Cat13). Bản thảo thực hiện (WD) cuối cùng cho PCC Cat2 được chứa trong tài liệu đưa ra MPEG N17534, và WD cuối cùng cho PCC Cat13 được chứa trong tài liệu đưa ra MPEG N17533.

Ý chính sau phương án của bộ mã hóa-giải mã PCC Cat2 trong PCC Cat2 WD là tận dụng bộ mã hóa-giải mã video hiện có để nén thông tin địa lý và thông tin kết cấu của đám mây điểm động, bằng cách nén dữ liệu đám mây điểm thành tập hợp

các chuỗi video khác nhau. Cụ thể là, hai chuỗi video, một biểu diễn thông tin địa lý của dữ liệu đám mây điểm và chuỗi còn lại biểu diễn thông tin kết cấu, được tạo ra và nén bằng cách sử dụng các bộ mã hóa-giải mã video. Siêu dữ liệu bổ sung để diễn giải hai chuỗi video, tức là, thông tin phân vá ánh xạ và phụ trợ chiếm giữ, cũng được tạo ra và được nén một cách riêng biệt.

Không may là, các phương án đã có của PCC có các nhược điểm. Ví dụ, các đơn vị dữ liệu mà liên quan đến một yêu cầu thời gian, tức là, một đơn vị truy cập (AU), không liên kết trong thứ tự giải mã. Trong PCC Cat 2 WD, các đơn vị dữ liệu của thông tin kết cấu, địa lý, phụ trợ, và ánh xạ chiếm giữ cho mỗi AU được đặt xen kẽ vào các đơn vị của nhóm khung. Tức là, dữ liệu địa lý cho tất cả các khung trong nhóm là cùng nhau. Điều tương tự thường đúng cho dữ liệu kết cấu, và v.v.. Trong PCC Cat13 WD, các đơn vị dữ liệu của địa lý và các thuộc tính thông thường cho mỗi AU được đặt xen kẽ trên cấp độ của toàn bộ dòng bit PCC (ví dụ như, tương tự như trong PCC Cat2 WD khi chỉ một nhóm các khung mà có cùng độ dài như toàn bộ dòng bit PCC). Việc đặt xen kẽ của các đơn vị dữ liệu thuộc về một AU vốn gây ra độ trễ đầu cuối lớn mà ít nhất là bằng độ dài của nhóm các khung trong khoảng thời gian trình diễn của các hệ thống ứng dụng.

Một nhược điểm khác đề cập đến định dạng dòng bit. Định dạng dòng bit cho phép sự mô phỏng của mẫu mã hóa khởi động như 0x0003 và do đó không thực hiện cho việc truyền qua dòng vận chuyển (TS) MPEG-2 trong đó việc ngăn chặn mô phỏng mã hóa khởi động là cần có. Đối với PCC Cat2, hiện tại chỉ `group_of_frames_geometry_video_payload()` và `group_of_frames_texture_video_payload()` có sự ngăn chặn mô phỏng mã hóa khởi động trong vị trí khi hoặc HEVC hoặc AVC được sử dụng để mã hóa của các thành phần địa lý và kết cấu. Đối với PCC Cat13, sự ngăn chặn mô phỏng mã hóa khởi động không nằm ở vị trí bất kỳ trong dòng bit.

Trong PCC Cat 2 WD, một số trong thông tin mã hóa-giải mã (ví dụ như, mã hóa-giải mã, hồ sơ, cấp độ nào, v.v., của bộ mã hóa-giải mã) dùng cho các dòng bit địa lý và kết cấu nằm sâu trong các thực thể (instance) của các cấu trúc

group_of_frames_geometry_video_payload() và group_of_frames_texture_video_payload(). Hơn nữa, một vài trong số các thông tin như hồ sơ và cấp độ mà chỉ báo các khả năng giải mã của các thành phần ánh xạ chiếm giữ và thông tin phụ trợ, cũng như đối với sự khôi phục đám mây điểm, đang thiếu.

Các phương án về cú pháp bậc cao mà giải quyết một hoặc nhiều vấn đề nêu trên được kết hợp với mã hóa đám mây điểm được đề xuất. Như sẽ được giải thích rõ hơn dưới đây, sáng chế sử dụng ký hiệu chỉ báo loại trong tiêu đề đơn vị dữ liệu (còn được gọi là, tiêu đề lớp truy nhập mạng (NAL) PCC) để chỉ định cụ thể loại nội dung trong tải trọng của đơn vị PCC NAL. Ngoài ra, sáng chế sử dụng nhóm đơn vị NAL tiêu đề khung để mang nhóm các tham số tiêu đề khung. Nhóm đơn vị NAL tiêu đề các khung có thể cũng được sử dụng để báo hiệu hồ sơ và cấp độ của mỗi dòng bit địa lý hoặc kết cấu.

FIG.1 là sơ đồ khối minh họa ví dụ về hệ thống mã hóa 10 mà có thể sử dụng các kỹ thuật mã hóa video PCC. Như được thể hiện trên FIG.1, hệ thống mã hóa 10 bao gồm thiết bị nguồn 12 mà cung cấp dữ liệu video được mã hóa cần được giải mã ở thời điểm sau bởi thiết bị đích 14. Cụ thể là, thiết bị nguồn 12 có thể cung cấp dữ liệu video cho thiết bị đích 14 qua phương tiện đọc được bằng máy tính 16. Thiết bị nguồn 12 và thiết bị đích 14 có thể bao gồm bất kỳ thiết bị nào, bao gồm các máy tính để bàn, máy tính xách tay (ví dụ như, laptop), máy tính bảng, thiết bị giải mã tín hiệu, điện thoại cầm tay như là điện thoại “thông minh”, bàn phím “thông minh”, tivi, máy ảnh, thiết bị hiển thị, trình phát truyền thông số, máy chơi trò chơi video, thiết bị phát trực tuyến video, hoặc loại tương tự. Trong một số trường hợp, thiết bị nguồn 12 và thiết bị đích 14 có thể được trang bị cho truyền thông không dây.

Thiết bị đích 14 có thể thu dữ liệu video được mã hóa cần được giải mã qua phương tiện đọc được bằng máy tính 16. Phương tiện đọc được bằng máy tính 16 có thể bao gồm bất kỳ loại phương tiện hoặc thiết bị nào có khả năng chuyển dữ liệu video được mã hóa từ thiết bị nguồn 12 tới thiết bị đích 14. Trong một ví dụ, phương tiện đọc được bằng máy tính 16 có thể bao gồm phương tiện truyền thông để cho phép thiết bị nguồn 12 truyền dữ liệu video được mã hóa trực tiếp đến thiết bị đích 14 thời

gian thực. Dữ liệu video được mã hóa có thể được điều chế theo tiêu chuẩn truyền thông, như là giao thức truyền thông không dây, và được truyền đến thiết bị đích 14. Phương tiện truyền thông có thể bao gồm bất kỳ phương tiện truyền thông không dây hoặc có dây, như là phổ tần số vô tuyến (RF) hoặc một hoặc nhiều đường truyền vật lý. Phương tiện truyền thông có thể tạo thành một phần của mạng dựa trên gói tin, như là mạng vùng cục bộ, mạng vùng rộng, hoặc mạng toàn cầu như là mạng Internet. Phương tiện truyền thông có thể bao gồm bộ định tuyến, bộ chuyển mạch, trạm gốc, hoặc thiết bị bất kỳ khác mà có thể hữu ích để thuận tiện cho việc truyền thông từ thiết bị nguồn 12 đến thiết bị đích 14.

Trong một số ví dụ, dữ liệu được mã hóa có thể xuất ra từ giao diện đầu ra 24 cho thiết bị lưu trữ. Một cách tương tự, dữ liệu được mã hóa có thể được truy cập từ thiết bị lưu trữ bởi giao diện đầu vào. Thiết bị lưu trữ có thể bao gồm bất kỳ trong số phương tiện lưu trữ dữ liệu được phân phối hoặc được truy cập cục bộ như là ổ đĩa cứng, đĩa Blu-ray, đĩa video kỹ thuật số (DVD), Bộ nhớ chỉ đọc đĩa compact (CD-ROM), bộ nhớ chớp, bộ nhớ bay hơi hoặc không bay hơi, hoặc phương tiện lưu trữ kỹ thuật số phù hợp bất kỳ khác để lưu trữ dữ liệu video được mã hóa. Trong ví dụ khác, thiết bị lưu trữ có thể tương ứng với máy chủ tệp tin hoặc một thiết bị lưu trữ trung gian mà lưu trữ dữ liệu video được mã hóa được tạo ra bởi thiết bị nguồn 12. Thiết bị đích 14 có thể truy cập dữ liệu video được lưu trữ từ thiết bị lưu trữ qua việc phát trực tuyến hoặc được tải xuống. Máy chủ tệp tin có thể là kiểu máy chủ bất kỳ có khả năng lưu trữ dữ liệu video được mã hóa và truyền dữ liệu video được mã hóa đến thiết bị đích 14. Máy chủ tệp tin trong ví dụ bao gồm máy chủ mạng (ví dụ, dùng cho trang mạng), máy chủ Giao thức truyền tải tệp tin (FTP), thiết bị lưu trữ được gắn vào mạng (NAS), hoặc ổ đĩa cục bộ. Thiết bị đích 14 có thể truy nhập dữ liệu video được mã hóa qua kết nối dữ liệu tiêu chuẩn bất kỳ, bao gồm kết nối internet. Việc này có thể bao gồm kênh không dây (ví dụ như, kết nối Wi-Fi), kết nối có dây (ví dụ như, đường dây thuê bao số (DSL), môdem cáp, v.v.), hoặc kết hợp của cả hai mà phù hợp để truy cập dữ liệu video được mã hóa được lưu trữ trên máy chủ tệp tin. Việc truyền của dữ liệu video được mã hóa từ thiết bị lưu trữ có thể là việc truyền phát trực tiếp, truyền tải xuống, hoặc kết hợp của nó.

Các kỹ thuật của sáng chế này không cần được giới hạn ở các ứng dụng hoặc thiết lập không dây. Các kỹ thuật có thể được áp dụng cho việc mã hóa video với sự hỗ trợ của nhiều ứng dụng đa phương tiện, như là phát rộng cho tivi qua không gian, truyền qua tivi bằng cáp, truyền qua tivi bằng vệ tinh, truyền video trực tuyến trên Internet, như là trực tuyến thích ứng động qua HTTP (DASH), video kỹ thuật số mà được mã hóa lên phương tiện lưu trữ dữ liệu, giải mã của video kỹ thuật số được lưu trữ trên phương tiện lưu trữ dữ liệu, hoặc các ứng dụng khác. Trong một số ví dụ, hệ thống mã hóa 10 có thể được cấu hình để hỗ trợ việc truyền video đơn hướng hoặc song hướng để hỗ trợ các ứng dụng như là trực tuyến video, trình phát lại video, phát rộng video, và/hoặc điện thoại truyền hình.

Trong ví dụ của FIG.1, thiết bị nguồn 12 bao gồm nguồn video 18 được tạo cấu hình để cung cấp hình ảnh thể tích, thiết bị trình chiếu 20, bộ mã hóa video 22, và giao diện đầu ra 24. Thiết bị đích 14 bao gồm giao diện đầu vào 26, bộ giải mã video 28, thiết bị khôi phục 30, và thiết bị hiển thị 32. Theo sáng chế này, bộ mã hóa 22 của thiết bị nguồn 12 và/hoặc bộ giải mã 28 của thiết bị đích 14 có thể được tạo cấu hình để áp dụng các kỹ thuật để mã hóa video. Trong các ví dụ khác, thiết bị nguồn và thiết bị đích có thể bao gồm các thành phần hoặc cơ cấu khác. Ví dụ, thiết bị nguồn 12 có thể thu dữ liệu video từ nguồn video ngoại vi, như là máy ảnh ngoại vi. Tương tự, thiết bị đích 14 có thể giao diện với thiết bị hiển thị ngoại vi, ngoài bao gồm thiết bị hiển thị được tích hợp.

Hệ thống mã hóa được minh họa của FIG.1 chỉ là một ví dụ. Các kỹ thuật để mã hóa video có thể được thực hiện bởi bất kỳ thiết bị mã hóa và/hoặc giải mã video kỹ thuật số nào. Mặc dù các kỹ thuật của sáng chế này thường được thực hiện bởi thiết bị mã hóa, các kỹ thuật có thể cũng được thực hiện bởi bộ mã hóa/bộ giải mã, được gọi chung là “CODEC.” Ngoài ra, các kỹ thuật của sáng chế này có thể cũng được thực hiện bởi bộ tiền xử lý video. Bộ mã hóa và/hoặc bộ giải mã có thể là bộ phận xử lý đồ họa (GPU) hoặc thiết bị tương tự.

Thiết bị nguồn 12 và thiết bị đích 14 chỉ là ví dụ về các thiết bị mã hóa mà trong đó thiết bị nguồn 12 tạo ra dữ liệu video được mã hóa cho việc truyền đến thiết

bị đích 14. Trong một số ví dụ, thiết bị nguồn 12 và thiết bị đích 14 có thể thao tác theo cách thức cơ bản là đối xứng để mỗi trong số các thiết bị nguồn và thiết bị đích 12, 14 bao gồm các thành phần mã hóa và giải mã video. Do đó, hệ thống mã hóa 10 có thể hỗ trợ việc truyền dữ liệu đơn hướng hoặc song hướng giữa các thiết bị video 12, 14, ví dụ như, để phát trực tuyến video, phát lại video, phát rộng video, hoặc phát qua điện thoại video.

Nguồn video 18 của thiết bị nguồn 12 có thể bao gồm thiết bị chụp hình video, như là máy ảnh video, lưu trữ video chứa video được chụp trước đó, và/hoặc giao diện cấp video để thu hình ảnh thể tích hoặc video từ thiết bị cung cấp nội dung video. Như một sự thay thế khác, nguồn video 18 có thể tạo ra hình ảnh thể tích hoặc dữ liệu dựa trên đồ họa máy tính như video nguồn, hoặc kết hợp của video trực tiếp, video được lưu trữ, và video được tạo bằng máy tính.

Trong một số trường hợp, khi nguồn video 18 là máy ảnh video, thiết bị nguồn 12 và thiết bị đích 14 có thể tạo ra cuộc gọi qua máy ảnh hoặc cuộc gọi video. Như được đề cập nêu trên, tuy nhiên, các kỹ thuật được mô tả trong sáng chế này có thể áp dụng được cho việc mã hóa video nói chung, và có thể được áp dụng cho các ứng dụng không dây và/hoặc có dây.

Thiết bị trình chiếu 20 được tạo cấu hình để chiếu hình ảnh thể tích lên bề mặt phẳng (ví dụ như, hộp giới hạn) như được giải thích đầy đủ hơn dưới đây. Tức là, thiết bị trình chiếu 20 được tạo cấu hình để chuyển đổi hình ảnh ba chiều (3D) thành hình ảnh hai chiều (2D) hoặc các hình ảnh.

Trong trường hợp bất kỳ, hình ảnh thể tích, video được chụp, video trước khi chụp, hoặc video được tạo bằng máy tính có thể được mã hóa bởi bộ mã hóa 22. Thông tin video được mã hóa sau đó có thể được xuất ra bởi giao diện đầu ra 24 trên phương tiện đọc được bằng máy tính 16.

Phương tiện đọc được bằng máy tính 16 có thể bao gồm phương tiện tạm thời, như là phát rộng không dây hoặc truyền qua mạng có dây, hoặc phương tiện lưu trữ (tức là, phương tiện lưu trữ bất biến), như là đĩa cứng, đĩa chớp, đĩa compact, đĩa video kỹ thuật số, đĩa Blu-ray, hoặc phương tiện đọc được bằng máy tính khác. Trong

một số ví dụ, máy chủ mạng (không được thể hiện) có thể thu dữ liệu video được mã hóa từ thiết bị nguồn 12 và cung cấp dữ liệu video được mã hóa cho thiết bị đích 14, ví dụ như, qua việc truyền mạng. Một cách tương tự, thiết bị tính toán của hạ tầng sản phẩm phương tiện, như là hạ tầng dập đĩa, có thể thu dữ liệu video được mã hóa từ thiết bị nguồn 12 và tạo ra đĩa chứa dữ liệu video được mã hóa. Do đó, phương tiện đọc được bằng máy tính 16 có thể được hiểu là bao gồm một hoặc nhiều phương tiện đọc được bằng máy tính theo các dạng khác nhau, trong các ví dụ khác nhau.

Giao diện đầu vào 26 của thiết bị đích 14 thu thông tin từ phương tiện đọc được bằng máy tính 16. Thông tin của phương tiện đọc được bằng máy tính 16 có thể bao gồm thông tin cú pháp được xác định bởi bộ mã hóa 22, mà cũng được sử dụng bởi bộ giải mã 28, rằng bao gồm các phần tử cú pháp mà mô tả các đặc tính và/hoặc xử lý của các khối và các đơn vị được mã hóa khác, ví dụ như, nhóm các hình ảnh (GOP).

Thiết bị khôi phục 30 được tạo cấu hình để chuyển đổi hình ảnh phẳng hoặc các hình ảnh trở lại hình ảnh thể tích như được giải thích đầy đủ hơn dưới đây. Tức là, thiết bị khôi phục 30 được tạo cấu hình để chuyển đổi hình ảnh 2D hoặc các hình ảnh trở lại hình ảnh 3D.

Thiết bị hiển thị 32 hiển thị hình ảnh thể tích hoặc dữ liệu video được giải mã cho người dùng, và có thể bao gồm bất kỳ thiết bị hiển thị nào như là ống tia âm cực (CRT), màn hiển thị tinh thể lỏng (LCD), màn hiển thị plasma, màn điôt phát quang hữu cơ (OLED), hoặc một loại khác của thiết bị hiển thị.

Bộ mã hóa 22 và bộ giải mã 28 có thể thao tác theo tiêu chuẩn mã hóa video, như là tiêu chuẩn mã hóa video hiệu quả cao (HEVC) hiện đang được phát triển, và có thể phù hợp với Mô hình thử nghiệm HEVC (HM). Ngoài ra, bộ mã hóa 22 và bộ giải mã 28 có thể thao tác theo các tiêu chuẩn phù hợp hoặc công nghiệp khác, như là tiêu chuẩn Lĩnh vực Tiêu chuẩn hóa Viễn thông Liên minh Viễn thông Quốc tế (ITU-T) H.264, còn được gọi là Nhóm chuyên gia hình ảnh chuyển động (MPEG)-4, Phần 10, Mã hóa video nâng cao (AVC), H.265/HEVC, hoặc sự mở rộng của các tiêu chuẩn này. Các kỹ thuật của sáng chế này, tuy nhiên, không được giới hạn ở bất kỳ tiêu chuẩn

mã hóa cụ thể nào. Ví dụ khác về các tiêu chuẩn mã hóa video bao gồm MPEG-2 và ITU-T H.263. Mặc dù không được thể hiện trên FIG.1, trong một số khía cạnh, bộ mã hóa 22 và bộ giải mã 28 có thể đều được tích hợp với bộ mã hóa và bộ giải mã audio, và có thể bao gồm các đơn vị ghép kênh-giải ghép kênh (MUX-DEMUX) thích hợp, hoặc phần cứng và phần mềm khác, để xử lý mã hóa của cả audio và video trong dòng dữ liệu chung hoặc các dòng dữ liệu riêng biệt. Nếu áp dụng được, các đơn vị MUX-DEMUX có thể phù hợp với giao thức ghép kênh ITU H.223, hoặc các giao thức khác như là giao thức biểu đồ dữ liệu người dùng (UDP).

Bộ mã hóa 22 và bộ giải mã 28 có thể được thực hiện như mạch bộ mã hóa phù hợp bất kỳ, như là một hoặc nhiều bộ vi xử lý, các bộ phận xử lý tín hiệu số (DSP), các mạch tích hợp ứng dụng cụ thể (ASIC), các mảng cổng khả trình trường (FPGA), logic rời rạc, phần mềm, phần cứng, phần sụn hoặc các kết hợp bất kỳ của nó. Khi các kỹ thuật được thực hiện theo phần trong phần mềm, thiết bị có thể lưu trữ các lệnh cho phần mềm trong phương tiện bất biến đọc được bằng máy tính phù hợp và thực thi các lệnh máy trong phần cứng bằng cách sử dụng một hoặc nhiều bộ phận xử lý để thực hiện các kỹ thuật của phần bộc lộ này. Bộ mã hóa 22 và bộ giải mã 28 có thể được chứa trong một hoặc nhiều bộ mã hóa hoặc bộ giải mã, mỗi trong số hai bộ này có thể được tích hợp làm một phần của bộ mã hóa/bộ giải mã được kết hợp (CODEC) trong thiết bị tương ứng. Thiết bị bao gồm bộ mã hóa 22 và/hoặc bộ giải mã 28 có thể bao gồm mạch tích hợp, bộ vi xử lý, và/hoặc thiết bị truyền thông không dây, như là điện thoại di động.

FIG.2 sơ đồ khối minh họa bộ mã hóa 22 ví dụ mà có thể thực hiện các kỹ thuật mã hóa video. Bộ mã hóa 22 có thể thực hiện mã hóa nội bộ và mã hóa liên đới của các khối video trong các lát video. Mã hóa nội bộ dựa trên dự đoán không gian để làm giảm hoặc loại bỏ dư thừa không gian trong video nằm trong khung hoặc hình ảnh video đã cho. Mã hóa liên đới dựa trên dự đoán thời gian để làm giảm hoặc loại bỏ dư thừa thời gian trong video nằm trong các khung hoặc các hình ảnh liên tiếp của chuỗi video. Chế độ nội bộ (chế độ I) có thể dựa vào bất kỳ các chế độ mã hóa dựa trên không gian. Các chế độ liên đới, như là dự đoán đơn hướng (cũng được gọi là, dự đoán

đơn nhất) (chế độ P) hoặc dự đoán song hướng (cũng được gọi là, dự đoán đôi) (chế độ B), có thể dựa vào bất kỳ chế độ mã hóa dựa trên thời gian.

Như được thể hiện trên FIG.2, bộ mã hóa 22 thu khối video hiện thời nằm trong khung video cần được mã hóa. Trong ví dụ của FIG.2, bộ mã hóa 22 bao gồm bộ phận lựa chọn chế độ 40, bộ nhớ khung tham chiếu 64, bộ phận tính tổng 50, bộ phận xử lý biến đổi 52, bộ phận lượng tử hóa 54, và bộ phận mã hóa entropy 56. Bộ phận lựa chọn chế độ 40, lần lượt, bao gồm bộ phận bù chuyển động 44, bộ phận ước lượng chuyển động 42, bộ phận dự đoán nội bộ (cũng được gọi là, dự đoán nội bộ) 46, và bộ phận phân vùng 48. Đối với việc khôi phục khối video, bộ mã hóa 22 cũng bao gồm bộ phận lượng tử hóa ngược 58, bộ phận biến đổi ngược 60, và bộ phận tính tổng 62. Bộ lọc giải khối (không được thể hiện trên FIG.2) có thể cũng nằm trong đó để lọc các biên khối để loại bỏ các tạo tác khối từ video được khôi phục. Nếu được như mong muốn, bộ lọc giải khối cần lọc đầu ra của bộ phận tính tổng 62. Các bộ lọc bổ sung (trong vòng hoặc sau vòng) có thể cũng được sử dụng ngoài bộ lọc giải khối. Các bộ lọc này không được thể hiện ngắn gọn, nhưng nếu được mong muốn, có thể lọc đầu ra của bộ phận tính tổng 50 (như bộ lọc trong vòng).

Trong xử lý mã hóa, bộ mã hóa 22 thu khung video hoặc lát video cần được mã hóa. Khung hoặc lát có thể được chia thành nhiều khối video. Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 thực hiện mã hóa dự đoán liên đới của khối video thu được tương ứng với một hoặc nhiều khối trong một hoặc nhiều khung tham chiếu để cung cấp dự đoán thời gian. Bộ phận dự đoán nội bộ 46 có thể còn thực hiện mã hóa dự đoán nội bộ của khối video thu được tương ứng với một hoặc nhiều khối lân cận trong cùng một khung hoặc lát như khối cần được mã hóa để cung cấp dự đoán không gian. Bộ mã hóa 22 có thể thực hiện nhiều lần mã hóa, ví dụ như, để lựa chọn chế độ mã hóa phù hợp cho mỗi khối của dữ liệu video.

Ngoài ra, bộ phận phân vùng 48 có thể phân vùng các khối của dữ liệu video thành các khối con, dựa trên sự đánh giá của các cơ chế phân vùng trước đó trong các lần mã hóa trước đó. Ví dụ, bộ phận phân vùng 48 ban đầu có thể phân vùng khung hoặc lát thành các đơn vị mã hóa lớn nhất (LCU), và chia phần mỗi LCU thành các

đơn vị mã hóa con (sub-CU) dựa trên phân tích tỷ lệ biến dạng (ví dụ như, tối ưu hóa tỷ lệ biến dạng). Bộ phận lựa chọn chế độ 40 có thể còn đưa ra cấu trúc dữ liệu cây tứ phân chỉ báo sự phân vùng của LCU thành các sub-CU. Các CU nút lá của cây tứ phân có thể bao gồm một hoặc nhiều bộ dự đoán (PU) và một hoặc nhiều bộ biến đổi (TU).

Sáng chế sử dụng thuật ngữ “khối” đề cập đến bất kỳ loại nào trong số CU, PU, hoặc TU, trong bối cảnh của HEVC, hoặc các cấu trúc dữ liệu tương tự trong bối cảnh của các tiêu chuẩn khác (ví dụ như, các khối macro và các khối con của nó trong H.264/AVC). CU bao gồm nút mã hóa, các PU, và các TU được kết hợp với nút mã hóa. Kích cỡ của CU tương ứng với kích cỡ của nút mã hóa và có hình dạng vuông. Kích cỡ của CU có thể nằm trong khoảng từ 8×8 điểm ảnh lên tới kích cỡ của khối cây tối đa là 64×64 điểm ảnh hoặc hơn. Mỗi CU có thể chứa một hoặc nhiều PU và một hoặc nhiều TU. Dữ liệu cú pháp được kết hợp với CU có thể mô tả, ví dụ, sự phân vùng của CU thành một hoặc nhiều PU. Các chế độ phân vùng có thể khác nhau giữa việc CU được bỏ qua hoặc được mã hóa chế độ trực tiếp, được mã hóa chế độ dự đoán nội bộ, hoặc được mã hóa chế độ dự đoán liên đới (còn được gọi là, dự đoán liên đới). Các PU có thể được phân vùng theo hình dạng không vuông. Dữ liệu cú pháp được kết hợp với CU có thể cũng mô tả, ví dụ, sự phân vùng của CU thành một hoặc nhiều TU theo cây tứ phân. TU có thể có hình dạng vuông hoặc không vuông (ví dụ như, hình chữ nhật).

Bộ phận lựa chọn chế độ 40 có thể lựa chọn một trong các chế độ mã hóa, dự đoán nội bộ hoặc dự đoán liên đới, ví dụ như, dựa trên các kết quả sai số, và cung cấp khối có kết quả dự đoán nội bộ hoặc dự đoán liên đới cho bộ phận tính tổng 50 để tạo ra dữ liệu khối dư và cho bộ phận tính tổng 62 để khôi phục khối được mã hóa cho việc sử dụng làm khung tham chiếu. Bộ phận lựa chọn chế độ 40 cũng cung cấp các phần tử cú pháp, như là các vectơ chuyển động, bộ chỉ báo chế độ nội bộ, thông tin phân vùng, và thông tin cú pháp khác, cho bộ phận mã hóa entropy 56.

Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 có thể được tích hợp, nhưng được minh họa một cách riêng biệt nhằm mục đích xác định khái niệm. Ước lượng chuyển động, được thực hiện bởi bộ phận ước lượng chuyển

động 42, là xử lý tạo ra các vector chuyển động, mà ước lượng chuyển động của các khối video. Vector chuyển động, ví dụ, có thể chỉ báo sự dời chỗ của PU của khối video nằm trong khung hoặc hình ảnh video hiện thời hoặc hình ảnh tương ứng với khối dự đoán nằm trong khung tham chiếu (hoặc đơn vị được mã hóa khác) tương ứng với khối hiện thời được mã hóa trong khung hiện thời (hoặc đơn vị được mã hóa khác). Khối dự đoán là khối mà được xác định để khớp gần với khối cần được mã hóa, về độ chênh lệch điểm ảnh, mà có thể được xác định bởi tổng của các độ chênh lệch tuyệt đối (SAD), tổng của bình phương độ chênh lệch (SSD), hoặc các chỉ số độ chênh lệch khác. Trong một số ví dụ, bộ mã hóa 22 có thể tính toán các giá trị cho các vị trí điểm ảnh số nguyên phụ của các hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 64. Ví dụ, bộ mã hóa 22 có thể nội suy các giá trị của các vị trí điểm ảnh một phần tư, các vị trí điểm ảnh một phần tám, hoặc các vị trí điểm ảnh tỷ lệ khác của hình ảnh tham chiếu. Do đó, bộ phận ước lượng chuyển động 42 có thể thực hiện tìm kiếm chuyển động tương ứng với các vị trí điểm ảnh đầy đủ và các vị trí điểm ảnh theo tỷ lệ và xuất ra vector chuyển động với độ chính xác điểm ảnh theo tỷ lệ.

Bộ phận ước lượng chuyển động 42 tính toán vector chuyển động cho PU của khối video trong lát được mã hóa liên đới bằng cách so sánh vị trí của PU với vị trí của khối dự đoán của hình ảnh tham chiếu. Hình ảnh tham chiếu có thể được lựa chọn từ danh sách hình ảnh tham chiếu thứ nhất (List 0) hoặc danh sách hình ảnh tham chiếu thứ hai (List 1), mỗi trong số đó nhận dạng một hoặc nhiều hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 64. Bộ phận ước lượng chuyển động 42 gửi vector chuyển động được tính toán đến bộ phận mã hóa entropy 56 và bộ phận bù chuyển động 44.

Việc bù chuyển động, được thực hiện bởi bộ phận bù chuyển động 44, có thể liên quan đến tìm nạp hoặc tạo ra khối dự đoán dựa trên the vector chuyển động được xác định bởi bộ phận ước lượng chuyển động 42. Một cách lặp lại, bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 có thể được tích hợp theo chức năng, trong một số ví dụ. Khi thu vector chuyển động cho PU của khối video hiện thời, bộ phận bù chuyển động 44 có thể định vị khối dự đoán mà vector chuyển động chỉ đến trong một trong các danh sách hình ảnh tham chiếu. Bộ tính tổng 50 tạo ra khối

video dư bằng cách trừ đi các giá trị điểm ảnh của khối dự đoán từ các giá trị điểm ảnh của khối video hiện tại được mã hóa, tạo ra các giá trị chênh lệch điểm ảnh, như được thảo luận dưới đây. Nói chung, bộ phận ước lượng chuyển động 42 thực hiện ước lượng chuyển động tương ứng với các thành phần luma, và bộ phận bù chuyển động 44 sử dụng các vector chuyển động được tính toán dựa trên các thành phần luma cho cả các thành phần chroma và các thành phần luma. Bộ phận lựa chọn chế độ 40 cũng có thể tạo ra các phần tử cú pháp được kết hợp với các khối video và lát video để sử dụng bởi bộ giải mã 28 khi giải mã các khối video của lát video.

Bộ phận dự đoán nội bộ 46 có thể dự đoán nội bộ khối hiện tại, như cách thay thế cho dự đoán liên đới được thực hiện bởi bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44, như được mô tả ở trên. Cụ thể là, bộ phận dự đoán nội bộ 46 có thể xác định chế độ dự đoán nội bộ để sử dụng để mã hóa khối hiện tại. Trong một số ví dụ, bộ phận dự đoán nội bộ 46 có thể mã hóa khối hiện tại bằng cách sử dụng các chế độ dự đoán nội bộ khác nhau, ví dụ như, trong các lần mã hóa riêng biệt, và bộ phận dự đoán nội bộ 46 (hoặc bộ phận lựa chọn chế độ 40, trong một số ví dụ) có thể lựa chọn chế độ dự đoán nội bộ phù hợp để sử dụng từ các chế độ được thử nghiệm.

Ví dụ, bộ phận dự đoán nội bộ 46 có thể tính toán các giá trị biến dạng tỷ lệ bằng cách sử dụng phân tích biến dạng tỷ lệ cho các chế độ dự đoán nội bộ khác nhau được thử nghiệm, và lựa chọn chế độ dự đoán nội bộ có các đặc tính biến dạng tỷ lệ tốt nhất trong số các chế độ được thử nghiệm. Phân tích biến dạng tỷ lệ thường xác định lượng biến dạng (hoặc sai số) giữa khối được mã hóa và khối gốc, chưa được mã hóa mà đã được mã hóa để đưa ra khối được mã hóa, cũng như tỷ lệ bit (tức là, số lượng bit) được sử dụng để đưa ra khối được mã hóa. Bộ phận dự đoán nội bộ 46 có thể tính toán các tỷ lệ từ các biến dạng tỷ lệ và các tỷ lệ cho các khối được mã hóa khác nhau để xác định chế độ dự đoán nội bộ nào đưa ra giá trị biến dạng tỷ lệ tốt nhất cho khối.

Ngoài ra, bộ phận dự đoán nội bộ 46 có thể được tạo cấu hình để mã hóa các khối độ sâu của ánh xạ độ sâu bằng cách sử dụng chế độ tạo mẫu độ sâu (DMM). Bộ

phần lựa chọn chế độ 40 có thể xác định xem chế độ DMM khả dụng đưa ra các kết quả mã hóa tốt hơn so với chế độ dự đoán nội bộ và các chế độ DMM khác, ví dụ như, bằng cách sử dụng tối ưu hóa biến dạng tỷ lệ (RDO). Dữ liệu dùng cho hình ảnh kết cấu tương ứng với ánh xạ độ sâu có thể được lưu trữ trong bộ nhớ khung tham chiếu 64. Bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 có thể cũng được tạo cấu hình cho các khối độ sâu dự đoán liên đới của ánh xạ độ sâu.

Sau khi lựa chọn chế độ dự đoán nội bộ cho khối (ví dụ như, chế độ dự đoán nội bộ thông thường hoặc một trong các chế độ DMM), bộ phận dự đoán nội bộ 46 có thể cung cấp thông tin chỉ báo chế độ dự đoán nội bộ được lựa chọn cho khối cho bộ phận mã hóa entropy 56. Bộ mã hóa entropy 56 có thể mã hóa thông tin mà chỉ báo chế độ dự đoán nội bộ được lựa chọn. Bộ mã hóa 22 có thể bao gồm trong dữ liệu cấu trúc dòng bit được truyền, mà có thể bao gồm các bảng chỉ số chế độ dự đoán nội bộ và các bảng chỉ số chế độ dự đoán nội bộ được hiệu chỉnh (cũng được gọi là các bảng ánh xạ từ mã), các định nghĩa của các ngữ cảnh mã hóa cho các khối khác nhau, và sự chỉ báo của chế độ dự đoán nội bộ có thể xảy ra nhất, bảng chỉ số chế độ dự đoán nội bộ, và bảng chỉ số chế độ dự đoán nội bộ được hiệu chỉnh để sử dụng cho mỗi trong số các ngữ cảnh.

Bộ mã hóa 22 tạo ra khối video dư bằng cách trừ đi dữ liệu dự đoán từ bộ phận lựa chọn chế độ 40 từ khối video gốc được mã hóa. Bộ tính tổng 50 thể hiện thành phần hoặc các thành phần mà thực hiện thao tác trừ này.

Bộ phận xử lý biến đổi 52 áp dụng biến đổi, như là biến đổi cosin rời rạc (DCT) hoặc biến đổi tương tự về mặt khái niệm, cho khối dư, đưa ra khối video bao gồm các giá trị hệ số biến đổi dư. Bộ phận xử lý biến đổi 52 có thể thực hiện các biến đổi khác mà tương tự về mặt khái niệm với DCT. Các biến đổi sóng cầu thứ cấp, các biến đổi số nguyên, các biến đổi băng con hoặc các loại biến đổi khác có thể cũng được sử dụng.

Bộ phận xử lý biến đổi 52 áp dụng biến đổi cho khối dư, sản xuất khối của các hệ số biến đổi phần dư. Bộ phận biến đổi có thể chuyển đổi thông tin phần dư từ miền giá trị điểm ảnh thành miền biến đổi, ví dụ, như là miền tần số. Bộ phận xử lý

biến đổi 52 có thể gửi các hệ số biến đổi cuối cùng đến bộ phận lượng tử hóa 54. Bộ lượng tử hóa 54 lượng tử hóa các hệ số biến đổi để giảm hơn nữa tốc độ bit. Xử lý lượng tử hóa có thể làm giảm chiều sâu bit được kết hợp với một số hoặc tất cả các hệ số biến đổi. Mức độ lượng tử hóa có thể được hiệu chỉnh bằng cách điều chỉnh tham số lượng tử. Trong một số ví dụ, bộ phận lượng tử hóa 54 sau đó có thể thực hiện quét của ma trận bao gồm các hệ số biến đổi được lượng tử hóa. Ngoài ra, bộ phận mã hóa entropy 56 có thể thực hiện quét.

Việc lượng tử hóa sau đây, bộ phận mã hóa entropy 56 mã hóa entropy các hệ số biến đổi được lượng tử hóa. Ví dụ, bộ phận mã hóa entropy 56 có thể thực hiện mã hóa độ dài biến thích ứng ngữ cảnh (CAVLC), mã hóa số học nhị phân thích ứng ngữ cảnh (CABAC), mã hóa số học nhị phân thích ứng ngữ cảnh dựa trên cú pháp (SBAC), mã hóa entropy phân vùng khoảng khả thi (PIPE) hoặc một kỹ thuật mã hóa entropy khác. Trong trường hợp của mã hóa entropy dựa trên ngữ cảnh, ngữ cảnh có thể dựa trên các khối lân cận. Mã hóa entropy dưới đây bởi bộ phận mã hóa entropy 56, dòng bit được mã hóa có thể được truyền đến một thiết bị khác (ví dụ như, bộ giải mã 28) hoặc được lưu trữ cho việc truyền hoặc tìm kiếm sau.

Bộ lượng tử hóa ngược 58 và bộ phận biến đổi ngược 60 áp dụng lượng tử hóa ngược và biến đổi ngược, theo cách tương ứng, để khôi phục khối dư trong miền điểm ảnh, ví dụ như, cho việc sử dụng sau như khối tham chiếu. Bộ phận bù chuyển động 44 có thể tính toán khối tham chiếu bằng cách thêm khối dư vào khối dự đoán của một trong các khung của bộ nhớ khung tham chiếu 64. Bộ phận bù chuyển động 44 có thể cũng áp dụng một hoặc nhiều bộ lọc nội suy cho khối dư được khôi phục để tính toán các giá trị điểm ảnh số nguyên phụ cho việc sử dụng trong ước lượng chuyển động. Bộ tính tổng 62 thêm khối dư được khôi phục vào khối dư được bù chuyển động được đưa ra bởi bộ phận bù chuyển động 44 để đưa ra khối video được khôi phục cho việc lưu trữ trong bộ nhớ khung tham chiếu 64. Khối video được khôi phục có thể được sử dụng bởi bộ phận ước lượng chuyển động 42 và bộ phận bù chuyển động 44 làm khối tham chiếu để mã hóa liên đới khối trong khung video kế tiếp.

FIG.3 sơ đồ khối minh họa bộ giải mã 28 ví dụ mà có thể thực hiện các kỹ thuật mã hóa video. Trong ví dụ của FIG.3, bộ giải mã 28 bao gồm bộ giải mã entropy 70, bộ phận bù chuyển động 72, bộ phận dự đoán nội bộ 74, bộ phận lượng tử hóa ngược 76, bộ phận biến đổi ngược 78, bộ nhớ khung tham chiếu 82, và bộ phận tính tổng 80. Bộ giải mã 28 có thể, trong một số ví dụ, thực hiện cuộc giải mã thường đối ứng với cuộc mã hóa được mô tả đối với bộ mã hóa 22 (FIG.2). Bộ phận bù chuyển động 72 có thể tạo ra dữ liệu dự đoán dựa trên các vector chuyển động được thu từ bộ giải mã entropy 70, trong khi đó bộ phận dự đoán nội bộ 74 có thể tạo ra dữ liệu dự đoán dựa trên các bộ chỉ báo chế độ dự đoán nội bộ được thu từ bộ giải mã entropy 70.

Trong xử lý giải mã, bộ giải mã 28 thu dòng bit video được mã hóa mà thể hiện các khối video của lát video được mã hóa và các phần tử cú pháp được kết hợp từ bộ mã hóa 22. Bộ phận giải mã entropy 70 của bộ giải mã 28 giải mã entropy dòng bit để tạo ra các hệ số được lượng tử hóa, các vector chuyển động hoặc các bộ chỉ báo chế độ dự đoán nội bộ, và các phần tử cú pháp khác. Bộ phận giải mã entropy 70 chuyển tiếp các vector chuyển động và các phần tử cú pháp khác tới bộ phận bù chuyển động 72. Bộ giải mã 28 có thể thu các phần tử cú pháp ở cấp độ lát video và/hoặc cấp độ khối video.

Khi phần video được mã hóa là lát được mã hóa nội bộ (I), bộ phận dự đoán nội bộ 74 có thể tạo ra dữ liệu dự đoán được sử dụng cho khối video của phần video hiện thời dựa trên chế độ nội dự đoán được báo hiệu và dữ liệu từ các khối được giải mã trước đó của khung hoặc hình ảnh hiện thời. Khi khung video được mã hóa như lát được mã hóa liên đới (ví dụ như, B, P, hoặc GPB), bộ phận bù chuyển động 72 đưa ra các khối dự đoán cho khối video của lát video hiện thời dựa trên các vector chuyển động và các phần tử cú pháp khác được thu từ bộ giải mã entropy 70. Các khối dự đoán có thể được tạo ra từ một trong các hình ảnh tham chiếu trong một trong các danh sách hình ảnh tham chiếu. Bộ giải mã 28 có thể xây dựng cả danh sách hình ảnh tham chiếu, Danh sách 0 và Danh sách 1, bằng cách sử dụng các kỹ thuật xây dựng mặc định dựa trên các hình ảnh tham chiếu được lưu trữ trong bộ nhớ khung tham chiếu 82.

Bộ phận bù chuyển động 72 xác định sự dự đoán thông tin cho khối video của lát video hiện thời bằng cách phân tách các vector chuyển động và các phần tử cú pháp khác, và sử dụng sự dự đoán thông tin để tạo ra các khối dự đoán cho khối video hiện thời được giải mã. Ví dụ, bộ phận bù chuyển động 72 sử dụng một số phần tử cú pháp trong các phần tử cú pháp thu được để xác định chế độ dự đoán (ví dụ như, dự đoán nội bộ hoặc dự đoán liên đới) được sử dụng để mã hóa các khối video của lát video, kiểu lát dự đoán liên đới (ví dụ như, lát B, lát P, hoặc lát GPB), thông tin cấu trúc cho một hoặc nhiều trong số các danh sách hình ảnh tham chiếu cho lát, các vector chuyển động cho mỗi khối video được mã hóa liên đới của lát, trạng thái dự đoán liên đới cho mỗi khối video được mã hóa liên đới của lát, và thông tin khác để giải mã các khối video trong lát video hiện thời.

Bộ phận bù chuyển động 72 có thể cũng thực hiện phép nội suy dựa trên các bộ lọc nội suy. Bộ phận bù chuyển động 72 có thể sử dụng các bộ lọc nội suy như được sử dụng bởi bộ mã hóa 22 trong khi mã hóa của các khối video để tính toán các giá trị được nội suy cho các điểm ảnh số nguyên phụ của các khối tham chiếu. Trong trường hợp này, bộ phận bù chuyển động 72 có thể xác định các bộ lọc nội suy được sử dụng bởi bộ mã hóa 22 từ các phần tử cú pháp thu được và sử dụng các bộ lọc nội suy để đưa ra các khối dự đoán.

Dữ liệu dùng cho hình ảnh kết cấu tương ứng với ánh xạ độ sâu có thể được lưu trữ trong bộ nhớ khung tham chiếu 82. Bộ phận bù chuyển động 72 có thể cũng được tạo cấu hình cho các khối độ sâu dự đoán liên đới của ánh xạ độ sâu.

Các kỹ thuật mã hóa thông thường yêu cầu các phần vá được sắp xếp theo chế độ vá của chúng trong đơn vị dữ liệu khung phần vá (trước đó được gọi là, bộ tạo đệm cho phần vá). Tức là, sự thể hiện của các loại đơn vị dữ liệu phần vá trong đơn vị dữ liệu phần vá thông tin phụ trợ trong mã hóa đám mây điểm dựa trên video (V-PCC) yêu cầu tất cả thông tin liên quan đến phần vá cần được sắp xếp theo thứ tự nhất định trong đơn vị dữ liệu khung phần vá. Thứ nhất, các đơn vị dữ liệu phần vá đã khớp được thêm vào danh sách đơn vị dữ liệu phần vá, tiếp sau đó là các đơn vị dữ liệu phần vá không khớp. Các đơn vị dữ liệu phần vá đã khớp có thể được mã hóa bằng cách sử

dùng mã hóa vi sai (dự đoán liên đới - dpdu) và các đơn vị dữ liệu phần vá không khớp được mã hóa bằng cách sử dụng các giá trị tuyệt đối (dự đoán nội bộ). Tức là, tất cả các phần vá được mã hóa liên đới phải được liệt kê trước tiên trong đơn vị dữ liệu khung phần vá, tiếp theo sau là tất cả các phần vá được mã hóa nội bộ, và cứ như vậy. Do đó, để thêm phần vá được mã hóa liên đới mới hoặc phần vá được mã hóa nội bộ mới vào đơn vị dữ liệu khung phần vá, toàn bộ đơn vị dữ liệu khung phần vá phải được dựng lại bằng cách sử dụng thứ tự được yêu cầu, mà làm cho việc mã hóa không hiệu quả.

Phần bộc lộ ở đây là các kỹ thuật mã hóa mà cho phép thứ tự linh hoạt của các phần vá trong đơn vị dữ liệu khung phần vá. Tức là, các phần vá mà có chế độ vá khác nhau (ví dụ như, dự đoán nội bộ, dự đoán liên đới, PCM, v.v.) có thể được chứa trong đơn vị dữ liệu khung phần vá theo thứ tự bất kỳ. Các kỹ thuật mã hóa cũng cho phép loại đơn vị dữ liệu phần vá không liên tục (spdu) cần được sử dụng. Đơn vị dữ liệu phần vá không liên tục chỉ báo rằng tất cả các tham số dùng cho phần vá hiện tại có thể được kế thừa từ phần vá tham chiếu khi các phần tử đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại và đơn vị dữ liệu phần vá tham chiếu là như nhau hoặc được khớp hoàn hảo. Các kỹ thuật mã hóa cũng cho phép phần tử cú pháp, mà chỉ báo số lượng các phần vá đã khớp, cần được loại bỏ khỏi báo hiệu vì báo hiệu kiểu phần vá được mô tả ở đây là đủ để thể hiện toàn bộ danh sách các phần vá trong đơn vị dữ liệu khung phần vá. Ngoài ra, các kỹ thuật mã hóa loại bỏ nhu cầu báo hiệu tổng số phần vá trong đơn vị dữ liệu khung phần vá (ví dụ như, kích cỡ của đơn vị dữ liệu khung phần vá) và thay bằng xử lý với kiểu phần vá kết thúc đặc biệt. Hơn nữa, các kỹ thuật mã hóa cho phép phần vá điều chế được mã hóa xung (PCM), mà thường được đặt ở phần cuối của đơn vị dữ liệu khung phần vá, để được thay thế bằng chỉ báo kiểu phần vá PCM. Do đó, bộ mã hóa / giải mã (cũng được gọi là, “codec”) trong mã hóa video được cải thiện (ví dụ như, xử lý mã hóa và giải mã các phần vá trong V-PCC) tương ứng với các xử lý mã hóa-giải mã hiện tại. Đối với vấn đề thực tiễn, xử lý mã hóa video được cải thiện có thể nâng cao hiệu quả mã hóa, mà cho người dùng trải nghiệm tốt hơn khi các đám mây điểm được gửi, được thu, và/hoặc được quan sát.

FIG.4 là sự thể hiện của chuỗi 400 của các khung đám mây điểm 402, 404, 406 (ví dụ như, khung PCC_0, khung PCC_1, khung PCC_4) đều chứa đám mây điểm 408. Đám mây điểm 408 là sự biểu diễn thể tích của không gian trên lưới 3D thông thường. Tức là, đám mây điểm 408 là ba chiều (3D). Như được thể hiện trên FIG.4, đám mây điểm 408 chứa nội dung đám mây điểm 410 trong không gian 3D 412. Nội dung đám mây điểm 410 được biểu diễn bởi tập hợp các điểm (ví dụ như, các điểm ảnh ba chiều) trong không gian 3D 412. Điểm ảnh ba chiều là phần tử thể tích biểu diễn một số lượng số học, như là màu sắc của điểm trong không gian ba chiều, được sử dụng trong việc trực quan hóa và phân tích của dữ liệu ba chiều. Do đó, điểm ảnh ba chiều có thể được coi là tương đương ba chiều của điểm ảnh trong hình ảnh 2D.

Mỗi điểm ảnh ba chiều trong đám mây điểm 408 của FIG.4 có các tọa độ (ví dụ như, tọa độ $x y z$) và một hoặc nhiều thuộc tính (ví dụ như, thành phần màu sắc đỏ/xanh lá/xanh da trời (RGB), độ phản chiếu, v.v.). Trong khi đó nội dung đám mây điểm 410 trên FIG.4 mô tả người, nội dung đám mây điểm 410 có thể là đối tượng thể tích bất kỳ hoặc hình ảnh trong các ứng dụng thực tiễn.

FIG.5 là sự thể hiện của đám mây điểm 400 của FIG.4 được chiếu lên hộp giới hạn 500. Như được thể hiện trên FIG.5, hộp giới hạn 500 chứa các phần vá 502 được chiếu lên các bề mặt hai chiều (2D) hoặc các mặt phẳng 504 của nó. Do đó, các phần vá 502 là sự thể hiện 2D của các phần của hình ảnh 3D. Các phần vá 502 nhìn chung là tương ứng với nội dung đám mây điểm 410 của FIG.4. Sự biểu diễn dữ liệu trong mã hóa đám mây điểm dựa trên video (V-PCC), mà có thể cũng được gọi là nén đám mây điểm, dựa trên việc chuyển đổi từ 3D sang 2D.

Sự biểu diễn dữ liệu trong V-PCC được mô tả là tập hợp các hình ảnh 2D phẳng (ví dụ như, các phần vá 502) bằng cách sử dụng, ví dụ, ánh xạ chiếm giữ 610 của FIG.6A, ánh xạ địa lý 612 của FIG.6B, và ánh xạ thuộc tính 614 của FIG.6C.

FIG.6A là sự thể hiện của ánh xạ chiếm giữ 610 tương ứng với các phép chiếu 2D (ví dụ như, các phần vá 502) từ hộp giới hạn 500 của FIG.5. Ánh xạ chiếm giữ 610 được mã hóa theo dạng nhị phân. Ví dụ, số không biểu diễn rằng phần của hộp giới hạn 600 không được chiếm giữ bởi một trong các phần vá 602. Các phần đó

của hộp giới hạn 600 được biểu diễn bởi số không không tham gia vào việc khôi phục của sự biểu diễn thể tích (ví dụ như, nội dung đám mây điểm 410). Ngược lại, số một biểu diễn rằng phần của hộp giới hạn 600 được chiếm giữ bởi một trong các phần vá 602. Các phần đó của hộp giới hạn 600 được biểu diễn bởi số một tham gia vào việc khôi phục của sự biểu diễn thể tích (ví dụ như, nội dung đám mây điểm 410).

FIG.6B là sự thể hiện của ánh xạ địa lý 612 tương ứng với các phép chiếu 2D (ví dụ như, các phần vá 502) từ hộp giới hạn 500 của FIG.5. Ánh xạ hình học 612 cung cấp hoặc mô tả đường bao hoặc địa hình của mỗi trong số các phần vá 602. Tức là, ánh xạ hình học 612 chỉ báo khoảng cách của mỗi điểm trong các phần vá 602 cách xa khỏi bề mặt phẳng (ví dụ như, các mặt phẳng 504) của hộp giới hạn 600.

FIG.6C là sự thể hiện của ánh xạ thuộc tính 614 tương ứng với các phép chiếu 2D (ví dụ như, các phần vá 502) từ hộp giới hạn 500 của FIG.5. Ánh xạ thuộc tính 614 cung cấp hoặc mô tả thuộc tính của mỗi điểm trong các phần vá 602 trong hộp giới hạn 600. Thuộc tính trong ánh xạ thuộc tính 614 có thể là, ví dụ, thành phần màu sắc của điểm. Thành phần màu sắc có thể dựa trên mô hình màu sắc RGB, mô hình màu sắc YUV, hoặc dựa trên một mô hình màu sắc đã biết khác.

FIG.7 là sự thể hiện của dòng bit V-PCC 700. Dòng bit V-PCC 700 có thể được sử dụng để mang thông tin được mã hóa cần dùng để khôi phục đám mây điểm (ví dụ như, đám mây điểm 408 của FIG.4). Như được thể hiện, dòng bit V-PCC 700 bao gồm các đơn vị V-PCC 702. Mỗi đơn vị V-PCC 702 chứa tiêu đề đơn vị V-PCC 704 và tải đơn vị V-PCC 706. Tiêu đề đơn vị V-PCC 704 mô tả tải đơn vị V-PCC 706 được mang bởi đơn vị V-PCC 702.

Tải đơn vị V-PCC 706 bao gồm tập tham số chuỗi 708, dữ liệu chuỗi phần vá 710, dữ liệu video chiếm giữ 712, dữ liệu video hình học 714, và dữ liệu video thuộc tính 716 (ví dụ như, dữ liệu video kết cấu). Tập tham số chuỗi 708 bao gồm các tham số và thông tin tương ứng với chuỗi của các đám mây điểm (ví dụ như, chuỗi 400 của các khung đám mây điểm 402, 404, 406 đều chứa đám mây điểm 408 trên FIG.4). Dữ liệu video chiếm giữ 712, dữ liệu video hình học 714, và dữ liệu video

thuộc tính 716 mang dữ liệu chiếm giữ, dữ liệu hình học, và dữ liệu thuộc tính hoặc kết cấu tương ứng với đám mây điểm hoặc hình ảnh 3D.

Dữ liệu chuỗi phần vá 710 bao gồm các tham số và thông tin tương ứng với chuỗi của các phần vá (ví dụ như, các phần vá 502 trên FIG.5). Dữ liệu chuỗi phần vá 710 bao gồm tập tham số chuỗi 720, tập tham số khung 722, tập tham số hình học 724, tập tham số thuộc tính 726, tập tham số phần vá hình học 728, tập tham số phần vá thuộc tính 730, và khung phần vá 732.

Tập tham số chuỗi 720 bao gồm các tham số và thông tin tương ứng với chuỗi của các phần vá 2D (ví dụ như, các phần vá 502 của FIG.5) mà vẫn dùng cho toàn bộ khoảng thời gian của chuỗi đám mây điểm hoặc cho đến khi tập tham số chuỗi mới được báo hiệu trong dòng bit. Tập tham số khung 722 chứa thông tin tương ứng với chuỗi của các phần vá 2D vẫn dùng cho khoảng thời gian của khung đơn của chuỗi đám mây điểm. Tập tham số hình học 724 chứa thông tin mà tương ứng với xử lý khôi phục của các đặc tính hình học của dữ liệu chuỗi phần vá. Tập tham số thuộc tính 726 chứa thông tin mà tương ứng với xử lý khôi phục của các đặc tính thuộc tính của dữ liệu chuỗi phần vá. Tập tham số phần vá hình học 728 chứa thông tin mà tương ứng với xử lý giải mã của các đặc tính hình học của dữ liệu chuỗi phần vá. Thuộc tính 730 chứa thông tin mà tương ứng với xử lý giải mã của các đặc tính thuộc tính của dữ liệu chuỗi phần vá.

Đơn vị dữ liệu khung phần vá 732 (được gọi là khung phần vá) bao gồm tập các đơn vị dữ liệu phần vá 734. Các đơn vị dữ liệu phần vá 734 nhìn chung chứa dữ liệu phần vá (ví dụ như, dữ liệu phần vá $F_0, \dots, F_k$) dùng cho phần vá (ví dụ như, phần vá 502). Mỗi trong số các đơn vị dữ liệu phần vá 734 có kiểu phần vá được đề cập đến ở đây như chế độ phần vá. Như sẽ được giải thích đầy đủ hơn dưới đây, chế độ phần vá (ví dụ như, `patch_mode`) chỉ báo cách thức phần vá cụ thể cần được xử lý trong các xử lý mã hóa.

FIG.8 là sự thể hiện của sự phân giải được mã hóa của chuỗi đám mây điểm 800 cho các khung 802. Các khung 802 trên FIG.8 tương tự như các khung đám mây điểm 402, 404, 406 trên FIG.4. Do đó, mỗi trong số các khung 802 chứa nội dung đám

mây điểm 810 dùng cho đám mây điểm 3D. Nội dung đám mây điểm 810 trong mỗi khung 802 của FIG.8 tương ứng với ánh xạ chiếm giữ 812, ánh xạ hình học 814, và ánh xạ thuộc tính 816 trong phép chiếu 2D mà được sử dụng để khôi phục nội dung đám mây điểm 810. Như được thể hiện, phép chiếu 2D có thể chứa nhiều hơn một ánh xạ thuộc tính 816.

Như được thể hiện, chuỗi đám mây điểm 800 cũng chứa các đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 (AUX_INF). Trong phương án, các đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 chứa thông tin hộp giới hạn 2D và 3D (ví dụ như, hộp giới hạn 3D phần vá 1204 và hộp giới hạn phép chiếu 2D phần vá 1208 trên FIG.12). Sự tập hợp của các đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 tương ứng với dữ liệu chuỗi phần vá 710 trên FIG.7. Mỗi đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 chứa tập tham số chuỗi 720, tập tham số khung 722, tập tham số hình học 724, tập tham số thuộc tính 726, tập tham số phần vá hình học 728, tập tham số phần vá thuộc tính 730, và khung phần vá 732. Trong phương án, khung phần vá 732 được chứa trong các đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 và phần còn lại của các thành phần trong dữ liệu chuỗi phần vá 710 là tùy chọn. Trong phương án, tập hợp của các ánh xạ chiếm giữ 812 tương ứng với dữ liệu video chiếm giữ 712, tập hợp của các ánh xạ hình học 814 tương ứng với dữ liệu video hình học 714, tập hợp của các hình thuộc tính 816 tương ứng với dữ liệu video thuộc tính 716. Đáng chú ý, ánh xạ chiếm giữ không thể hiện trên hình ảnh này. Các đơn vị dữ liệu khung phần vá thông tin phụ trợ 850 đều tương ứng với nội dung đám mây điểm 810 trong một trong các khung 802. Đơn vị dữ liệu phần vá (ví dụ như, 850 chứa thông tin được sử dụng hoặc cần dùng để khôi phục nội dung đám mây điểm 810 trong đám mây điểm 3D dựa trên ánh xạ chiếm giữ 812, ánh xạ hình học 814, và ánh xạ thuộc tính 816 trong phép chiếu 2D.

FIG.9 là sự thể hiện của đơn vị dữ liệu phần vá đơn nhất 900 được trích xuất từ một trong các khung đám mây điểm 802 trong chuỗi đám mây điểm 800 của FIG.8. Đơn vị dữ liệu phần vá đơn nhất 900 tương tự với các đơn vị dữ liệu phần vá 734 trong khung phần vá 732 của FIG.7 hoặc các đơn vị dữ liệu khung phần vá thông tin phụ trợ

850. Đơn vị dữ liệu phần vá đơn nhất 900 được mô tả tương ứng với không gian 3D xung quanh 902.

Các FIG.10A-10C là sự thể hiện của ánh xạ chiếm giữ 1000, ánh xạ hình học 1002, và ánh xạ thuộc tính 1004 tương ứng với đơn vị dữ liệu phần vá đơn nhất 900 của FIG.9. Ánh xạ chiếm giữ 1000, ánh xạ hình học 1002, và ánh xạ thuộc tính 1004 là tương tự với ánh xạ chiếm giữ 610, ánh xạ hình học 612, và ánh xạ thuộc tính 614 trên các FIG.6A-6C. Tuy nhiên, ánh xạ chiếm giữ 1000, ánh xạ hình học 1002, và ánh xạ thuộc tính 1004 trên các FIG.10A-10C chỉ thể hiện đơn vị dữ liệu phần vá đơn nhất 900 thay vì tập hợp của các phần vá 502 từ toàn bộ hộp giới hạn 500. Do đó, ánh xạ chiếm giữ 1000, ánh xạ hình học 1002, và ánh xạ thuộc tính 1004 trên các FIG.10A-10C chỉ chứa phần vá đơn nhất 1006 tương ứng với đơn vị dữ liệu phần vá đơn nhất 900 của FIG.9.

FIG.11 là sự thể hiện của các thành phần 2D dùng cho đơn vị dữ liệu phần vá 1152 trong hộp giới hạn 2D 1154. Đơn vị dữ liệu phần vá 1152 trên FIG.11 tương tự với đơn vị dữ liệu phần vá đơn nhất 900 của FIG.9. Như được thể hiện trên FIG.11, các thành phần 2D thể hiện tọa độ phần vá (ví dụ như, $[u_0, v_0]$) và các kích thước (ví dụ như, $Size_u_0$, $Size_v_0$) dùng cho đơn vị dữ liệu phần vá 1152. Các thành phần 2D này được chứa trong đơn vị dữ liệu phần vá (ví dụ như, đơn vị dữ liệu phần vá 734 của FIG.7 hoặc đơn vị dữ liệu phần vá 850 của FIG.8).

Các FIG.12A-12B là sự thể hiện giản lược 1200 và sự thể hiện tập dữ liệu 1252 của các thành phần 3D dùng cho đơn vị dữ liệu phần vá 1152 của FIG.11. Như được thể hiện, sự thể hiện giản lược 1200 bao gồm hộp giới hạn 3D đám mây điểm 1202, hộp giới hạn 3D phần vá 1204, mặt phẳng chiếu 1206, và hộp giới hạn phép chiếu 2D phần vá 1208. Các tọa độ khác nhau (ví dụ như, $[0,0,0]$) và các chiều (ví dụ như, $x, y, z, u, v, d1, u1, v1$) và các vectơ (b, t, bt) được sử dụng để mô tả quá thể hiện nội dung của hộp giới hạn 3D đám mây điểm 1202 lên hộp giới hạn phép chiếu 2D phần vá 1208. Sự thể hiện tập dữ liệu 1252 mô tả cách thức mà trong đó đơn vị dữ liệu phần vá đơn nhất 1254 (mà tương tự với đơn vị dữ liệu phần vá đơn nhất 900) có thể được chiếu lên mặt phẳng chiếu 1206 dựa trên các tọa độ (ví dụ như, $[0,0,0]$, $[u1, v1,$

d1]) và các vector (ví dụ như, n, bt). Các thành phần 3D này được chứa trong đơn vị dữ liệu phần vá (ví dụ như, đơn vị dữ liệu phần vá 734 của FIG.7 hoặc đơn vị dữ liệu phần vá 850 của FIG.8).

Đơn vị dữ liệu phần vá (ví dụ như, đơn vị dữ liệu phần vá 734 của FIG.7 hoặc đơn vị dữ liệu phần vá 850 của FIG.8) chứa thông tin cần dùng cho xử lý khôi phục 2D đến xử lý khôi phục 3D, và được thể hiện trong phần dưới đây (xem FIG.11):

các thành phần liên quan đến 2d

U0 (Patch2dShiftU)

V0 (Patch2dShiftV)

Size_u0 (Patch2dSizeU)

Size_v0 (Patch2dSizeV)

và trong phần dưới đây (xem các FIG.12A-12B):

các thành phần liên quan đến 3d

- u1 (Patch3dShiftTangentAxis) - khoảng cách từ hộp giới hạn phần vá 3d theo trục tiếp tuyến (t) đến hộp giới hạn đám mây điểm
- v1 (Patch3dShiftBiTangentAxis) - khoảng cách từ hộp giới hạn phần vá 3d theo trục song tiếp tuyến (bt) đến hộp giới hạn đám mây điểm
- d1 (Patch3dShiftNormalAxis) - khoảng cách từ hộp giới hạn phần vá 3d theo trục vuông góc (n) đến hộp giới hạn đám mây điểm
- n (PatchNormalAxis) - trục mà vuông góc (vuông góc) với mặt phẳng chiếu phần vá (cạnh bất kỳ của hộp giới hạn có thể là mặt phẳng chiếu)
- t (Patch3dShiftTangentAxis) - trục mà tiếp tuyến với phần vá bề mặt và vuông góc với trục vuông góc
- bt (Patch3dShiftBiTangentAxis) - trục mà vuông góc với các trục vuông góc và tiếp tuyến

Như đã lưu ý ở trên, các kỹ thuật mã hóa thông thường yêu cầu các phần vá (các đơn vị dữ liệu phần vá 734 của FIG.7) được xếp thứ tự theo chế độ phần vá của chúng trong đơn vị dữ liệu khung phần vá. Tức là, sự thể hiện của các loại đơn vị dữ liệu phần vá trong đơn vị dữ liệu khung phần vá trong V-PCC yêu cầu tất cả thông tin liên quan đến phần vá cần được sắp xếp theo thứ tự nhất định trong đơn vị dữ liệu khung phần vá. Thứ nhất, các đơn vị dữ liệu phần vá đã khớp được thêm vào khung dữ liệu phần vá, tiếp sau đó là các đơn vị dữ liệu phần vá không khớp. Các đơn vị dữ liệu phần vá đã khớp có thể được mã hóa bằng cách sử dụng mã hóa vi sai (dự đoán liên đới - dpdu) và các đơn vị dữ liệu phần vá không khớp được mã hóa bằng cách sử dụng các giá trị tuyệt đối (dự đoán nội bộ). Tức là, tất cả các phần vá được mã hóa liên đới phải được liệt kê trước tiên trong đơn vị dữ liệu khung phần vá, tiếp theo sau là tất cả các phần vá được mã hóa nội bộ, và cứ như vậy. Do đó, để thêm phần vá được mã hóa liên đới mới hoặc phần vá được mã hóa nội bộ mới vào đơn vị dữ liệu khung phần vá, toàn bộ đơn vị dữ liệu khung phần vá phải được dựng lại bằng cách sử dụng thứ tự được yêu cầu, mà làm cho việc mã hóa không hiệu quả. Các phương pháp được mô tả dưới đây khắc phục việc mã hóa không hiệu quả.

FIG.13 là phương án của xử lý mã hóa đơn vị dữ liệu khung phần vá 1300. Xử lý mã hóa đơn vị dữ liệu khung phần vá 1300 có thể được sử dụng để mã hóa (cũng được gọi là, nén) các đơn vị dữ liệu phần vá như là đơn vị dữ liệu phần vá 734 của FIG.7. Xử lý mã hóa đơn vị dữ liệu khung phần vá 1300 có thể được thực hiện bởi bộ mã hóa. Ở đây, đơn vị dữ liệu khung phần vá, mà tương ứng với hộp giới hạn 2D 1154, là tập hợp của các đơn vị dữ liệu phần vá, mà tương ứng với đơn vị dữ liệu phần vá 1152. Trong phương án, đầu vào cho xử lý mã hóa 1300 là tập hợp của các đơn vị dữ liệu phần vá mà tương ứng với đơn vị dữ liệu khung phần vá.

Trong khối 1302, đơn vị dữ liệu khung phần vá (pdfu) được thu nhận. Đơn vị dữ liệu khung phần vá có thể được thu bởi bộ phận thu của bộ mã hóa. Đơn vị dữ liệu khung phần vá bao gồm tập hợp của các đơn vị dữ liệu phần vá. Đơn vị dữ liệu phần vá nhận dạng hoặc bao gồm kiểu phần vá (pdfu_patch_mode) cho mỗi đơn vị dữ liệu phần vá (ví dụ như, các phần vá 502). Trong phương án, kiểu phần vá là một trong các kiểu không liên tục, nội bộ, liên đới, PCM, hoặc cuối. Kiểu vá không liên tục chỉ

báo rằng phần vá hiện thời giống với phần vá tham chiếu. Do đó, các tham số từ phần vá tham chiếu được sử dụng khi mã hóa phần vá hiện thời. Kiểu phần vá nội bộ chỉ báo rằng phần vá hiện thời cần được mã hóa bằng cách sử dụng dự đoán nội bộ (cũng được gọi là, mã hóa nội bộ). Kiểu phần vá liên đới chỉ báo rằng phần vá hiện thời cần được mã hóa bằng cách sử dụng dự đoán liên đới (cũng được gọi là, mã hóa liên đới). Kiểu phần vá PCM chỉ báo rằng phần vá hiện thời thể hiện các điểm phân tán độc lập từ đám mây điểm 3D. Kiểu vá cuối chỉ báo rằng phần vá hiện thời là phần vá cuối cần được mã hóa và, do đó, xử lý nén của đơn vị dữ liệu khung phần vá cần được hoàn tất hoặc chấm dứt.

Trong khối 1304, chỉ số đơn vị dữ liệu phần vá (p) là giảm dần. Bằng cách giảm dần chỉ số đơn vị dữ liệu phần vá, chỉ số đơn vị dữ liệu phần vá có thể bắt đầu với giá trị âm (ví dụ như, -1). Do đó, khi chỉ số đơn vị dữ liệu phần vá là tăng dần trong khối 1306, chỉ số đơn vị dữ liệu phần vá được khởi tạo hiệu quả để bắt đầu ở mức không. Trong phương án, xử lý 1300 có thể cũng thiết lập chỉ số đơn vị dữ liệu phần vá bằng không lúc ban đầu, và sau đó tăng dần chỉ số đơn vị dữ liệu phần vá sau vòng hoặc phần lặp của xử lý được thực hiện (như được giải thích dưới đây).

Trong khối 1308, việc xác định xem loại đơn vị dữ liệu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại từ các đơn vị dữ liệu phần vá trong đơn vị dữ liệu khung phần vá là kiểu phần vá cuối được thực hiện. Khi đơn vị dữ liệu phần vá hiện tại không là phần vá cuối, xử lý 1300 tiến hành đến khối 1310. Trong khối 1310, dữ liệu thông tin phần vá được mã hóa cho phần vá hiện thời. Trong phương án, dữ liệu thông tin phần vá bao gồm chỉ số khung (frmIdx), chỉ số phần vá (p), và kiểu phần vá (pdfu_patch_mode) cũng như thông tin 2D và 3D tương ứng. Sau khi mã hóa trong khối 1310, xử lý 1300 lặp lại đến khối 1306. Ở đó, chỉ số đơn vị dữ liệu phần vá tăng dần. Bằng cách tăng dần chỉ số phần vá, đơn vị dữ liệu phần vá hiện thời mới được xét đến. Các bước này được lặp lại cho đến khi đơn vị dữ liệu phần vá hiện tại được xác định là phần vá cuối trong khối 1308.

Khi đơn vị dữ liệu phần vá hiện tại được xác định là đơn vị dữ liệu phần vá cuối, xử lý 1300 tiến hành đến khối 1312. Trong khối 1312, dữ liệu thông tin phần vá

được mã hóa cho đơn vị dữ liệu phần vá hiện tại. Trong phương án, dữ liệu thông tin phần vá bao gồm chỉ số khung (frmIdx), chỉ số phần vá (p), và kiểu phần vá (pdfu_patch_mode), cần được đề cập đến rằng đơn vị dữ liệu phần vá cuối không có thông tin 2D hoặc 3D được liên kết với đơn vị dữ liệu phần vá. Không như trong khối 1310, kiểu phần vá trong khối 1312 được mã hóa là kiểu phần vá cuối.

Trong khối 1314, xử lý căn chỉnh byte được thực hiện. Trong phương án, căn chỉnh byte sắp xếp hoặc bố trí các byte của dữ liệu thành định hướng hoặc kích cỡ mong muốn. Sau đó, trong khối 1316, có được đơn vị dữ liệu khung phần vá được nén. Đơn vị dữ liệu khung phần vá được nén có thể được tích hợp vào dòng bit để truyền về phía bộ giải mã.

FIG.14 là phương án của xử lý mã hóa đơn vị dữ liệu khung phần vá 1400 bằng cách sử dụng cờ phần vá cuối. Xử lý mã hóa đơn vị dữ liệu khung phần vá 1400 có thể được sử dụng để mã hóa (cũng được gọi là, nén) các đơn vị dữ liệu phần vá như là đơn vị dữ liệu phần vá 734 của FIG.7. Xử lý mã hóa đơn vị dữ liệu khung phần vá 1400 có thể được thực hiện bởi bộ mã hóa.

Trong khối 1402, đơn vị dữ liệu khung phần vá (pdfu) được thu nhận. Đơn vị dữ liệu khung phần vá có thể được thu bởi bộ phận thu của bộ mã hóa. Đơn vị dữ liệu khung phần vá bao gồm các đơn vị dữ liệu phần vá (ví dụ như, các phần vá 502).

Trong khối 1404, chỉ số phần vá (p) giảm dần. Bằng cách giảm dần chỉ số phần vá, chỉ số phần vá có thể bắt đầu với giá trị âm (ví dụ như, -1). Trong khối 1406, cờ phần vá cuối được thêm vào mỗi trong số nhiều phần vá. Trong phương án, cờ phần vá cuối được gán pdfu_is_last_patch_flag. Cờ phần vá cuối có thể được thiết lập hoặc được gán giá trị theo kiểu phần vá của mỗi phần vá. Ví dụ, cờ phần vá có thể chỉ báo rằng kiểu phần vá dùng cho phần vá là một trong các kiểu không liên tục, nội bộ, liên đới, PCM, hoặc cuối.

Trong khối 1408, chỉ số phần vá tăng dần để khởi tạo hiệu quả chỉ số phần vá để bắt đầu ở mức không. Trong khối 1410, việc xác định xem kiểu phần vá dùng cho đơn vị dữ liệu phần vá hiện tại từ các đơn vị dữ liệu phần vá mà tương ứng với

đơn vị dữ liệu khung phần vá là kiểu phần vá cuối được thực hiện dựa trên giá trị của cờ phần vá cuối. Khi đơn vị dữ liệu phần vá hiện tại không là phần vá cuối, xử lý 1400 tiến hành đến khối 1412. Trong khối 1412, dữ liệu thông tin phần vá được mã hóa cho đơn vị dữ liệu phần vá hiện tại. Trong phương án, dữ liệu thông tin phần vá bao gồm chỉ số khung (*frmIdx*), chỉ số phần vá (*p*), và kiểu phần vá (*pdfu_patch_mode*) cũng như thông tin 2D và 3D tương ứng. Trong khối 1412 của xử lý 1400, kiểu phần vá có thể là kiểu không liên tục, liên đới, nội bộ, hoặc PCM, nhưng không phải kiểu cuối. Trong khối 1414, cờ phần vá cuối được mã hóa.

Sau khi mã hóa trong khối 1414, xử lý 1400 lặp lại đến khối 1408. Ở đó, chỉ số phần vá tăng dần. Bằng cách tăng dần chỉ số phần vá, phần vá hiện thời mới được xét đến. Các bước này được lặp lại cho đến khi phần vá hiện thời được xác định là phần vá cuối trong khối 1410 dựa trên giá trị của cờ phần vá cuối.

Khi phần vá hiện thời được xác định là phần vá cuối dựa trên cờ phần vá cuối, xử lý 1400 tiến hành đến khối 1416. Trong khối 1416, xử lý căn chỉnh byte được thực hiện. Sau đó, trong khối 1418, có được đơn vị dữ liệu khung phần vá được nén. Đơn vị dữ liệu khung phần vá được nén có thể được tích hợp vào dòng bit để truyền về phía bộ giải mã.

FIG.15 là phương án của khung dữ liệu thông tin phần vá 1500 với sự mô tả của các loại đơn vị phần vá. Khung dữ liệu thông tin phần vá (đơn vị dữ liệu khung vá) 1500 tương tự với khung phần vá (đơn vị dữ liệu khung vá) 732 trên FIG.7. Như được thể hiện, dữ liệu thông tin khung phần vá 1500 bao gồm đơn vị dữ liệu khung phần vá tham chiếu (được chỉ báo bởi *refFrmIdx*) 1502, đơn vị dữ liệu khung phần vá hiện tại (*frmIdx*) 1504, và đơn vị thông tin phần vá 1506. Đơn vị dữ liệu khung phần vá tham chiếu 1502 bao gồm, ví dụ, khung tham chiếu được sử dụng khi kết nối với dự đoán liên đới hoặc không liên tục. Đơn vị dữ liệu khung phần vá hiện thời 1504 chứa các đơn vị dữ liệu phần vá cần được mã hóa hoặc giải mã. Đơn vị dữ liệu thông tin phần vá 1506 bao gồm các phần vá của các đơn vị dữ liệu phần vá (ví dụ như, phần vá 502) có một trong các kiểu phần vá khả dụng. Trong phương án, đơn vị dữ liệu phần vá có thể là kiểu phần vá nội bộ (*pdu*) 1508, kiểu phần vá PCM (*ppdu*) 1510,

kiểu phân vá liên đới (dpdu) 1512, và kiểu phân vá không liên tục (spdu) 1514. Có thể có số lượng bất kỳ của các phân vá trong đơn vị thông tin phân vá 1506, và các phân vá có thể theo thứ tự bất kỳ.

FIG.16 là phương án của xử lý giải mã đơn vị dữ liệu khung phân vá 1600. Xử lý giải mã đơn vị dữ liệu khung phân vá 1600 có thể được sử dụng để giải mã (cũng được gọi là, giải nén) hình ảnh thể tích mà đã được mã hóa bởi bộ mã hóa. Xử lý giải mã đơn vị dữ liệu khung phân vá 1600 có thể được thực hiện bởi bộ mã hóa (ví dụ như, bộ giải mã entropi 70).

Trong khối 1602, dữ liệu thông tin phân vá được mã hóa được thu và được đọc. Dữ liệu thông tin phân vá được mã hóa có thể được thu bởi bộ phận thu của bộ giải mã làm một phần của dòng bit. Dữ liệu thông tin phân vá được mã hóa chứa dữ liệu và thông tin cần để giải mã hình ảnh thể tích mà đã được mã hóa bởi bộ mã hóa. Ví dụ, dữ liệu thông tin phân vá được mã hóa bao gồm các đơn vị dữ liệu phân vá (ví dụ như, các phân vá 502 hoặc các đơn vị dữ liệu phân vá 850) mà được nén và cần dùng để khôi phục hình ảnh thể tích (ví dụ như, đám mây điểm 3D) từ các phân vá. Như một phần của khối 1602, phân vá (ví dụ như, đơn vị dữ liệu phân vá ban đầu hoặc đơn vị dữ liệu phân vá hiện tại) từ nhiều phân vá có thể được thu nhận một lần xử lý 1600 được khởi tạo.

Trong khối 1604, đầu vào tương ứng với dữ liệu thông tin phân vá được mã hóa được thu. Trong phương án, đầu vào tương ứng với một hoặc nhiều của các phân vá được nén. Trong phương án, đầu vào bao gồm một hoặc nhiều của patch_mode, chỉ số phân vá, chỉ số tham chiếu, chỉ số khung, và chỉ số khung tham chiếu tương ứng với phân vá.

Trong khối 1606, sự xác định xem kiểu phân vá dùng cho phân vá là kiểu phân vá không liên tục được thực hiện. Khi kiểu phân vá dùng cho phân vá là kiểu phân vá không liên tục, xử lý 1600 tiến hành đến khối 1608. Trong khối 1608, chỉ số phân vá tham chiếu (spdu_patch_index) tương ứng với phân vá được giải mã. Tức là, giá trị của chỉ số phân vá tham chiếu được giải mã. Trong phương án,

spdu_patch_index là chỉ số của phần vá trong khung tham chiếu, được chỉ báo bởi refFrmIdx.

Trong khối 1610, chỉ số tham chiếu (refIdx) dùng cho phần vá được xác định dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục. Trong phương án, chỉ số tham chiếu, mà có thể cũng được gọi là chỉ số phần vá tham chiếu (RefPatchIdx), được xác định dựa trên phần sau đây: $\text{RefPatchIdx} = \text{predictorIdx} + \text{spdu_patch_index}[\text{frmIdx}][p]$, trong đó predictorIdx là chỉ số dự đoán, spdu_patch_index là chỉ số đơn vị dữ liệu phần vá kiểu không liên tục, frmIdx là chỉ số khung, và p là chỉ số phần vá. Trong phương án, chỉ số đơn vị dữ liệu khung phần vá hiện thời (frmIdx) và bộ chỉ điểm cho đơn vị dữ liệu phần vá hiệu chỉnh trong chỉ số phần vá (p) chỉ tới điểm thích hợp trong mảng đa chiều để tìm kiếm chỉ số tham chiếu thực tế dùng cho phần vá.

Trong khối 1612, phần vá được khôi phục dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

$$\text{Patch2dShiftU}[\text{frmIdx}][p] = \text{Patch2dShiftU}[\text{refIdx}][\text{RefPatchIdx}] * \text{ops_occupancy_packing_block_size}$$

$$\text{Patch2dShiftV}[\text{frmIdx}][p] = \text{Patch2dShiftV}[\text{refIdx}][\text{RefPatchIdx}] * \text{ops_occupancy_packing_block_size}$$

$$\text{Patch2dSizeU}[\text{frmIdx}][p] = \text{Patch2dSizeU}[\text{refIdx}][\text{RefPatchIdx}] * \text{ops_occupancy_packing_block_size}$$

$$\text{Patch2dSizeV}[\text{frmIdx}][p] = \text{Patch2dSizeV}[\text{refIdx}][\text{RefPatchIdx}] * \text{ops_occupancy_packing_block_size}$$

$$\text{Patch3dShiftTangentAxis}[\text{frmIdx}][p] = \text{Patch3dShiftTangentAxis}[\text{refIdx}][\text{RefPatchIdx}]$$

```

Patch3dShiftBiTangentAxis[ frmIdx ][ p ] = Patch3dShiftBiTangentAxis[
refIdx ][ RefPatchIdx ]

Patch3dShiftNormalAxis[ frmIdx ][ p ] = Patch3dShiftNormalAxis[ refIdx ][
RefPatchIdx ]

PatchNormalAxis[ frmIdx ][ p ] = PatchNormalAxis[ refIdx ][ RefPatchIdx ]

PatchTangentAxis[ frmIdx ][ p ] = PatchTangentAxis[ refIdx ][ RefPatchIdx ]

PatchBiTangentAxis[ frmIdx ][ p ] = PatchBiTangentAxis[ refIdx ][
RefPatchIdx ]

PatchOrientationSwapFlag[ frmIdx ][ p ] = PatchOrientationSwapFlag[ refIdx
][ RefPatchIdx ]

PatchLod[ frmIdx ][ p ] = PatchLod[ refIdx ][ RefPatchIdx ]

PatchProjectionMode[ frmIdx ][ p ] = dpdu_projection_mode[ frmIdx ][ p ]

```

Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

```

Patch2dShiftU[p]=pdu_2d_shift_u[refIdx]

Patch2dShiftV[p]=pdu_2d_shift_v[refIdx]

Patch2dSizeU[p]=Patch2dSizeU[refIdx]

Patch2dSizeV[p]=Patch2dSizeV[refIdx]

Patch3dShiftT[p]=Patch3dShiftT[refIdx]

Patch3dShiftBT[p]=Patch3dShiftBT[refIdx]

Patch3dShiftN[p]=Patch3dShiftN[refIdx]

PatchNormalAxis[p]=PatchNormalAxis[refIdx]

Orientation[p]=Orientation[refIdx]

PatchLod[p]=PatchLod[refIdx]

```

Trong khối 1650, dữ liệu thông tin phần vá được khôi phục được tập hợp (ví dụ như, được lưu trữ trong bộ nhớ) và xử lý 1600 lặp vòng lại đến khối 1602 sao cho phần vá mới có thể được giải mã.

Trong khối 1614, việc xác định xem kiểu phần vá dùng cho đơn vị dữ liệu phần vá có là kiểu phần vá nội bộ được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá nội bộ, xử lý 1600 tiến hành đến khối 1616. Trong khối 1616, chỉ số đơn vị dữ liệu phần vá tham chiếu (refIdx) dùng cho đơn vị dữ liệu phần vá được xác định dựa trên đơn vị dữ liệu phần vá trước đó (tương ứng với phần vá và chỉ số phần vá giảm dần) trong chỉ số khung hiện thời ([frmIdx]). Trong phương án, khung hiện thời (ví dụ như, đơn vị dữ liệu khung phần vá) được sử dụng cho việc tham chiếu.

Trong khối 1618, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, phần vá được giải mã theo phần sau đây:

Thuật toán giải mã:

u0 (pdu_2d_shift_u)

u1 (pdu_2d_shift_v)

d_size_u0 (pdu_2d_delta_size_u)

d_size_v0 (pdu_2d_delta_size_v)

u1 (pdu_3d_shift_tangent_axis)

v1 (pdu_3d_shift_bitangent_axis)

d1 (pdu_3d_shift_normal_axis)

n (pdu_normI_axis)

swap (pdu_orientation_swap_flag)

L0D (pdu_Iod)

Trong khối 1620, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

Patch2dShiftU[p] = pdu_2d_shift_u[p]

Patch2dShiftV[p] = pdu_2d_shift_v[p]

Patch2dSizeU[p] = pdu_2d_delta_size_u[p] +
+ Patch2dSizeU[refIdx]

Patch2dSizeV[p] = pdu_2d_delta_size_v[p] +
+ Patch2dsizeV[refIdx]

Patch3dShiftT[p] = pdu_3d_shift_tan[p]

Patch3dShiftBT[p] = pdu_3d_shift_bitan[p]

Patch3dShiftN[p] = pdu_shift_norm[p]

PatchNormalAxis[p] = pdu_norm_axis[p]

Orientation[p] = pdu_orientation_swap_flag[p]

PatchLod[p] = pdu_lod[p]

Trong khối 1650, dữ liệu thông tin phần vá được khôi phục được tập hợp (ví dụ như, được lưu trữ trong bộ nhớ) và xử lý 1600 lặp vòng lại đến khối 1602 sao cho phần vá mới có thể được giải mã.

Trong khối 1622, sự xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới, xử lý 1600 tiến hành đến khối 1624. Trong khối 1624, chỉ số phần vá tham chiếu (dpdu_patch_index) tương ứng với phần vá được giải mã. Tức là, giá trị của chỉ số phần vá tham chiếu được giải mã. Trong khối 1626, chỉ số tham chiếu (refIdx) dùng cho phần vá được xác định dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã.

Trong khối 1628, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, phần vá được giải mã theo phần sau đây:

Thuật toán giải mã:

d_u0 (pdu_2d_shift_u)

d_u1 (pdu_2d_shift_v)

d_size_u0 (pdu_2d_delta_size_u)

d_size_v0 (pdu_2d_delta_size_v)

d_u1 (pdu_3d_shift_tangent_axis)

d_v1 (pdu_3d_shift_bitangent_axis)

d_d1 (pdu_3d_shift-normal_axis)

Trong khối 1630, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

Patch2dShiftU[p]=pdu_2d_shift_u[p]+

+Patch2dShiftU[refldx]

Patch2dShiftV[p]=pdu_2d_shift_v[p]+

+Patch2dshiftV[refldx]

Patch2dsizeU[p]=pdu_2d_delta_size_u[p]+

+Patch2dSizeU[refldx]

Patch2dSizeV[p]=pdu_2d_delta_size_v[p]+

+Patch2dSizeV[refldx]

Patch3dShiftT[p]=pdu_3d_shift_tan[p]+

+Patch3dshiftT[refldx]

Patch3dShiftBT[p]=pdu_3d_shift_bitan[p]+

+patch3dshiftBT[refldx]

Patch3dShiftN[p]=pdu_shift_norm[p]+

+Patch3dShiftN[refIdx]

PatchNormalAxis[p]=PatchnormalAxis[refIdx]

Orientation[p]=Orientation[refIdx]

PatchLod[p]=PatchLod[refIdx]

Trong khối 1650, dữ liệu thông tin phần vá được khôi phục được tập hợp (ví dụ như, được lưu trữ trong bộ nhớ) và xử lý 1600 lặp vòng lại đến khối 1602 sao cho phần vá mới có thể được giải mã.

Trong khối 1632, việc xác định xem kiểu phần vá dùng cho phần vá là loại phần vá PCM (hoặc thô) được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá PCM, xử lý 1600 tiến hành đến khối 1634. Trong khối 1634, chỉ số tham chiếu (refIdx) dùng cho phần vá được xác định dựa trên chỉ số khung ([frmIdx]) tương ứng với phần vá và chỉ số phần vá giảm dần.

Trong khối 1636, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, các thành phần 2D và 3D bao gồm phần sau đây:

Thuật toán giải mã:

separate_video_flag (ppdu_patch...)

u0 (ppdu_2d_shift_u)

u1 (ppdu_2d_shift_y)

d_size_u0 (ppdu_2d_delta_size_u)

d_size_v0 (ppdu_2d_delta_size_v)

Các điểm PCM (ppdu_pcm_points)

Trong khối 1638, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

Patch2dShiftU[p] = pdu_2d_shift_u[p]

$$\text{Patch2dShiftV}[p] = \text{pdu_2d_shifty}[p]$$

$$\begin{aligned} \text{Patch2dSizeU}[p] &= \text{pdu_2d_deltasize_u}[p] + \\ &+ \text{Patch2dSizeU}[\text{refIdx}] \end{aligned}$$

$$\begin{aligned} \text{Patch2dSizeV}[p] &= \text{pdu_2d_delta_sizev}[p] + \\ &+ \text{Patch2dSizeV}[\text{refIdx}] \end{aligned}$$

$$\text{PatchPcmPoints}[p] = \text{ppdu_pcm_points}[p]$$

Trong khối 1650, dữ liệu thông tin phần vá được khôi phục được tập hợp (ví dụ như, được lưu trữ trong bộ nhớ) và xử lý 1600 lặp vòng lại đến khối 1602 sao cho phần vá mới có thể được giải mã.

Trong khối 1640, sự xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá cuối được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá cuối, xử lý 1600 tiến hành đến khối 1642. Trong khối 1642, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa được kết thúc khi kiểu phần vá là kiểu phần vá cuối. Tức là, xử lý khôi phục được hoàn tất. Trong phương án, dữ liệu bất kỳ được chứa trong phần vá có kiểu phần vá cuối có thể được giải mã trước khi xử lý giải mã đơn vị dữ liệu khung phần vá 1600 được kết thúc.

FIG.17 là phương án của xử lý giải mã đơn vị dữ liệu khung phần vá 1700. Xử lý giải mã đơn vị dữ liệu khung phần vá 1700 có thể được sử dụng để giải mã (cũng được gọi là, giải nén) hình ảnh thể tích mà đã được mã hóa bởi bộ mã hóa. Xử lý giải mã đơn vị dữ liệu khung phần vá 1700 có thể được thực hiện bởi bộ mã hóa (ví dụ như, bộ giải mã entropy 70).

Trong khối 1702, dữ liệu thông tin phần vá được mã hóa được thu và được đọc. Dữ liệu thông tin phần vá được mã hóa có thể được thu bởi bộ phận thu của bộ giải mã làm một phần của dòng bit. Dữ liệu thông tin phần vá được mã hóa chứa dữ liệu và thông tin cần để giải mã hình ảnh thể tích mà đã được mã hóa bởi bộ mã hóa. Ví dụ, dữ liệu thông tin phần vá được mã hóa bao gồm các đơn vị dữ liệu phần vá (ví dụ như, các phần vá 502 hoặc các đơn vị dữ liệu phần vá 850) mà được nén cần dùng để khôi phục hình ảnh thể tích (ví dụ như, đám mây điểm 3D) từ các phần vá. Như

một phần của khối 1702, phần vá (ví dụ như, phần vá ban đầu hoặc phần vá hiện tại) từ nhiều phần vá có thể được thu nhận một lần xử lý 1700 được khởi tạo.

Trong khối 1704, đầu vào tương ứng với dữ liệu thông tin phần vá được mã hóa được thu. Trong phương án, đầu vào tương ứng với một hoặc nhiều của các phần vá được nén. Trong phương án, đầu vào bao gồm một hoặc nhiều của patch_mode, chỉ số phần vá, chỉ số tham chiếu, chỉ số khung, và chỉ số khung tham chiếu tương ứng với phần vá.

Trong khối 1706, sự xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá không liên tục được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá không liên tục, xử lý 1700 tiến hành đến khối 1708. Trong khối 1708, chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá được giải mã (hoặc được kết thúc). Tức là, giá trị của chỉ số phần vá tham chiếu được giải mã hoặc được xác định.

Trong khối 1710, chỉ số tham chiếu (refIdx) dùng cho phần vá được xác định dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục. Trong phương án, chỉ số tham chiếu, mà có thể cũng được gọi là chỉ số phần vá tham chiếu (RefPatchIdx), được xác định dựa trên phần sau đây: $\text{RefPatchIdx} = \text{predictorIdx} + \text{spdu_patch_index}[\text{frmIdx}][p]$, trong đó predictorIdx là chỉ số dự đoán, spdu_patch_index là chỉ số đơn vị dữ liệu phần vá kiểu không liên tục, frmIdx là chỉ số khung, và p là chỉ số phần vá.

Trong khối 1712, phần vá được khôi phục dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

$$\text{Patch2dShiftU}[\text{frmIdx}][p] = \text{Patch2dShiftU}[\text{refIdx}][\text{RefPatchIdx}] * \text{ops_occupancy_packing_block_size}$$

```

Patch2dShiftV[ frmIdx ][ p ] = Patch2dShiftV[ refIdx ][ RefPatchIdx ] *
ops_occupancy_packing_block_size

Patch2dSizeU[ frmIdx ][ p ] = Patch2dSizeU[ refIdx ][ RefPatchIdx ] *
ops_occupancy_packing_block_size

Patch2dSizeV[ frmIdx ][ p ] = Patch2dSizeV[ refIdx ][ RefPatchIdx ] *
ops_occupancy_packing_block_size

Patch3dShiftTangentAxis[ frmIdx ][ p ] = Patch3dShiftTangentAxis[ refIdx ][
RefPatchIdx ]

Patch3dShiftBiTangentAxis[ frmIdx ][ p ] = Patch3dShiftBiTangentAxis[
refIdx ][ RefPatchIdx ]

Patch3dShiftNormalAxis[ frmIdx ][ p ] = Patch3dShiftNormalAxis[ refIdx ][
RefPatchIdx ]

PatchNormalAxis[ frmIdx ][ p ] = PatchNormalAxis[ refIdx ][ RefPatchIdx ]

PatchTangentAxis[ frmIdx ][ p ] = PatchTangentAxis[ refIdx ][ RefPatchIdx ]

PatchBiTangentAxis[ frmIdx ][ p ] = PatchBiTangentAxis[ refIdx ][
RefPatchIdx ]

PatchOrientationSwapFlag[ frmIdx ][ p ] = PatchOrientationSwapFlag[ refIdx
][ RefPatchIdx ]

PatchLod[ frmIdx ][ p ] = PatchLod[ refIdx ][ RefPatchIdx ]

PatchProjectionMode[ frmIdx ][ p ] = dpdu_projection_mode[ frmIdx ][ p ]

```

Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

```

Patch2dShiftU[p]=pdu_2d_shift_u[refIdx]

Patch2dShiftV[p]=pdu_2d_shift_v[refIdx]

Patch2dSizeU[p]=Patch2dSizeU[refIdx]

Patch2dSizeV[p]=Patch2dSizeV[refIdx]

```

Patch3dShiftT[p]=Patch3dShiftT[refIdx]

Patch3dShiftBT[p]=Patch3dShiftBT[refIdx]

Patch3dShiftN[p]=Patch3dShiftN[refIdx]

PatchNormalAxis[p]=PatchNormalAxis[refIdx]

Orientation[p]=Orientation[refIdx]

PatchLod[p]=PatchLod[refIdx]

Trong khối 1760, cờ được giải mã để xác định xem có nhiều các phần vá để xét đến hay không. Trong phương án, cờ được gán `more_patches_available_flag`. Trong khối 1762, giá trị của cờ được xác định. Khi cờ có giá trị thứ nhất (ví dụ như, một), xử lý 1700 tiến hành đến khối 1764. Trong khối 1764, dữ liệu thông tin phần vá được khôi phục được lưu trữ và xử lý lặp vòng lại đến khối 1702 sao cho phần vá mới có thể được giải mã. Khi cờ có giá trị thứ hai (ví dụ như, không), xử lý 1700 tiến hành đến khối 1742. Trong khối 1742, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa được kết thúc khi kiểu phần vá là kiểu phần vá cuối. Tức là, xử lý khôi phục được hoàn tất.

Trong khối 1714, việc xác định xem kiểu phần vá dùng cho đơn vị dữ liệu phần vá có là kiểu phần vá nội bộ được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá nội bộ, xử lý 1700 tiến hành đến khối 1716. Trong khối 1716, chỉ số đơn vị dữ liệu phần vá tham chiếu (`refIdx`) dùng cho đơn vị dữ liệu phần vá được xác định dựa trên đơn vị dữ liệu phần vá trước đó (tương ứng với phần vá và chỉ số phần vá giảm dần) trong chỉ số khung hiện thời (`[frmIdx]`).

Trong khối 1718, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, phần vá được giải mã theo phần sau đây:

Thuật toán giải mã:

`u0 (pdu_2d_shift_u)`

`u1 (pdu_2d_shift_v)`

d_size_u0 (pdu_2d_delta_size_u)

d_size_v0 (pdu_2d_delta_size_v)

u1 (pdu_3d_shift_tangent_axis)

v1 (pdu_3d_shift_bitangent_axis)

d1 (pdu_3d_shift_normal_axis)

n (pdu_normI_axis)

swap (pdu_orientation_swap_flag)

L0D (pdu_lod)

Trong khối 1720, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

Patch2dShiftU[p] = pdu_2d_shift_u[p]

Patch2dShiftV[p] = pdu_2d_shift_v[p]

Patch2dSizeU[p] = pdu_2d_delta_size_u[p] +

+ Patch2dSizeU[refldx]

Patch2dSizeV[p] = pdu_2d_delta_size_v[p] +

+ Patch2dsizeV[refldx]

Patch3dShiftT[p] = pdu_3d_shift_tan[p]

Patch3dShiftBT[p] = pdu_3d_shift_bitan[p]

Patch3dShiftN[p] = pdu_shift_norm[p]

PatchNormalAxis[p] = pdu_norm_axis[p]

Orientation[p] = pdu_orientation_swap_flag[p]

PatchLod[p] = pdu_lod[p]

Trong khối 1760, cờ được giải mã để xác định xem có nhiều các phần vá để xét đến hay không. Trong phương án, cờ được gán more_patches_available_flag.

Trong khối 1762, giá trị của cờ được xác định. Khi cờ có giá trị thứ nhất (ví dụ như, một), xử lý 1700 tiến hành đến khối 1764. Trong khối 1764, dữ liệu thông tin phần vá được khôi phục được lưu trữ và xử lý 1700 lặp vòng lại đến khối 1702 sao cho phần vá mới có thể được giải mã. Khi cờ có giá trị thứ hai (ví dụ như, không), xử lý 1700 tiến hành đến khối 1742. Trong khối 1742, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa được kết thúc khi kiểu phần vá là kiểu phần vá cuối. Tức là, xử lý khôi phục được hoàn tất.

Trong khối 1722, sự xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá liên đới, xử lý 1700 tiến hành đến khối 1724. Trong khối 1724, chỉ số phần vá tham chiếu (`pdu_patch_index`) tương ứng với phần vá được giải mã. Tức là, giá trị của chỉ số phần vá tham chiếu được giải mã. Trong khối 1726, chỉ số tham chiếu (`refIdx`) dùng cho phần vá được xác định dựa trên chỉ số khung tham chiếu (`[refFrmIdx]`) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã.

Trong khối 1728, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, phần vá được giải mã theo phần sau đây:

Thuật toán giải mã:

`d_u0 (pdu_2d_shift_u)`

`d_u1 (pdu_2d_shift_v)`

`d_size_u0 (pdu_2d_delta_size_u)`

`d_size_v0 (pdu_2d_delta_size_v)`

`d_u1 (pdu_3d_shift_tangent_axis)`

`d_v1 (pdu_3d_shift_bitangent_axis)`

`d_d1 (pdu_3d_shift-normal_axis)`

Trong khối 1730, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

```

Patch2dShiftU[p]=pdu_2d_shift_u[p]+
+Patch2dShiftU[refldx]
Patch2dShiftVf[p]=pdu_2d_shift_v[p]+
+Patch2dshiftV[refldx]
Patch2dsizeU[p]=pdu_2d_delta_size_u[p]+
+Patch2dSizeU[refldx]
Patch2dSizeV[p]=pdu_2d_delta_size_v[p]+
+Patch2dSizeV[refldx]
Patch3dShiftT[p]=pdu_3d_shift_tan[p]+
+Patch3dshiftT[refldx]
Patch3dShiftBT[p]=pdu_3d_shift_bitan[p]+
+patch3dshiftBT[refldx]
Patch3dShiftN[p]=pdu_shift_norm[p]+
+Patch3dShiftN[refldx]
PatchNormalAxis[p]=PatchnormalAxis[refldx]
Orientation[p]=Orientation[refldx]
PatchLod[p]=PatchLod[refldx]

```

Trong khối 1760, cờ được giải mã để xác định xem có nhiều các phân vá để xét đến hay không. Trong phương án, cờ được gán `more_patches_available_flag`. Trong khối 1762, giá trị của cờ được xác định. Khi cờ có giá trị thứ nhất (ví dụ như, một), xử lý 1700 tiến hành đến khối 1764. Trong khối 1764, dữ liệu thông tin phân vá được khôi phục được lưu trữ và xử lý 1700 lặp vòng lại đến khối 1702 sao cho phân vá mới có thể được giải mã. Khi cờ có giá trị thứ hai (ví dụ như, không), xử lý 1700 tiến hành đến khối 1742. Trong khối 1742, xử lý khôi phục tương ứng với dữ liệu

thông tin phần vá được mã hóa được kết thúc khi kiểu phần vá là kiểu phần vá cuối. Tức là, xử lý khôi phục được hoàn tất.

Trong khối 1732, việc xác định xem kiểu phần vá dùng cho phần vá là loại phần vá PCM (hoặc thô) được thực hiện. Khi kiểu phần vá dùng cho phần vá là kiểu phần vá PCM, xử lý 1700 tiến hành đến khối 1734. Trong khối 1734, chỉ số tham chiếu (refIdx) dùng cho phần vá được xác định dựa trên chỉ số khung ([frmIdx]) tương ứng với phần vá và chỉ số phần vá giảm dần.

Trong khối 1736, phần vá được giải mã bằng cách sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá. Trong phương án, các thành phần 2D và 3D bao gồm phần sau đây:

Thuật toán giải mã:

separate_video_flag (ppdu_patch...)

u0 (ppdu_2d_shift_u)

u1 (ppdu_2d_shift_y)

d_size_u0 (ppdu_2d_delta_size_u)

d_size_v0 (ppdu_2d_delta_size_v)

Các điểm PCM (ppdu_pcm_points)

Trong khối 1738, phần vá được khôi phục dựa trên phần vá như được giải mã. Trong phương án, phần vá được khôi phục theo phần sau đây:

Khôi phục:

Patch2dShiftU[p] = pdu_2d_shift_u[p]

Patch2dShiftV[p] = pdu_2d_shifty[p]

Patch2dSizeU[p] = pdu_2d_deltasize_u[p]+

+ Patch2dSizeU[refIdx]

Patch2dSizeV[p] = pdu_2d_delta_sizev[p]+

+Patch2dSizeV[refIdx]

PatchPcmPoints[p] = ppdu_pcm_points[p]

Trong khối 1760, cờ được giải mã để xác định xem có nhiều các phần vá để xét đến hay không. Trong phương án, cờ được gán `more_patches_available_flag`. Trong khối 1762, giá trị của cờ được xác định. Khi cờ có giá trị thứ nhất (ví dụ như, một), xử lý 1700 tiến hành đến khối 1764. Trong khối 1764, dữ liệu thông tin phần vá được khôi phục được lưu trữ và xử lý 1700 lặp vòng lại đến khối 1702 sao cho phần vá mới có thể được giải mã. Khi cờ có giá trị thứ hai (ví dụ như, không), xử lý 1700 tiến hành đến khối 1742. Trong khối 1742, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa được kết thúc khi kiểu phần vá là kiểu phần vá cuối. Tức là, xử lý khôi phục được hoàn tất.

FIG.18 là phương án của phương pháp 1800 của PCC được thực hiện bởi bộ giải mã (ví dụ như, bộ giải mã entropi 70). Phương pháp 1800 có thể được sử dụng để giải mã dòng bit được mã hóa để khôi phục hình ảnh thể tích. Trong khối 1802, bộ phận thu của bộ giải mã thu dữ liệu thông tin phần vá được mã hóa. Thông tin phần vá được mã hóa có thể bao gồm thông tin tương ứng với các các phần vá (ví dụ như, các phần vá 502). Trong khối 1804, bộ phận xử lý của bộ giải mã thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa. Phần vá có kiểu phần vá (`patch_mode`). Kiểu vá có thể là, ví dụ, kiểu không liên tục, nội bộ, liên đới, PCM, hoặc cuối.

Trong khối 1806, bộ phận xử lý của bộ giải mã xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá cuối hay không. Trong khối 1808, bộ phận xử lý của bộ giải mã kết thúc xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối.

FIG.19 là phương án của phương pháp 1900 của PCC được thực hiện bởi bộ giải mã (ví dụ như, bộ giải mã entropi 70). Phương pháp 1900 có thể được sử dụng để giải mã dòng bit được mã hóa để khôi phục hình ảnh thể tích. Trong khối 1902, bộ phận thu của bộ giải mã thu dữ liệu thông tin phần vá được mã hóa. Thông tin phần vá được mã hóa có thể bao gồm thông tin tương ứng với các các phần vá (ví dụ như, các phần vá 502). Trong khối 1904, bộ phận xử lý của bộ giải mã thu nhận phần vá

tương ứng với dữ liệu thông tin phần vá được mã hóa. Phần vá có kiểu phần vá (patch_mode). Kiểu vá có thể là, ví dụ, kiểu không liên tục, nội bộ, liên đới, PCM, hoặc cuối.

Trong khối 1906, bộ phận xử lý của bộ giải mã xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá không liên tục hay không. Trong khối 1908, bộ phận xử lý của bộ giải mã giải mã chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục. Trong khối 1910, bộ phận xử lý của bộ giải mã xác định chỉ số tham chiếu (refIdx) dùng cho phần vá dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục. Trong khối 1912, bộ phận xử lý của bộ giải mã khôi phục sự biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục. Khi được khôi phục, hình ảnh 3D có thể được hiển thị cho người dùng trên màn hiển thị của thiết bị điện tử (ví dụ như, điện thoại thông minh, máy tính bảng, máy tính xách tay, v.v.).

FIG.20 là phương án của phương pháp 2000 của PCC được thực hiện bởi bộ giải mã (ví dụ như, bộ giải mã entropy 70). Phương pháp 2000 có thể được sử dụng để giải mã dòng bit được mã hóa để khôi phục hình ảnh thể tích. Trong khối 2002, bộ phận thu của bộ giải mã thu dữ liệu thông tin phần vá được mã hóa. Thông tin phần vá được mã hóa có thể bao gồm thông tin tương ứng với các các phần vá (ví dụ như, các phần vá 502). Trong khối 2004, bộ phận xử lý của bộ giải mã thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa. Phần vá có kiểu phần vá (patch_mode). Kiểu vá có thể là, ví dụ, kiểu không liên tục, nội bộ, liên đới, PCM, hoặc cuối.

Trong khối 2006, bộ phận xử lý của bộ giải mã xác định xem kiểu phần vá dùng cho phần vá là kiểu phần vá không liên tục hay không. Trong khối 2008, bộ phận xử lý của bộ giải mã giải mã chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục. Trong khối 2010, bộ phận xử lý của bộ giải mã xác định chỉ số tham chiếu (refIdx) dùng cho phần vá dựa trên

chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục. Trong khối 2012, bộ phận xử lý của bộ giải mã khôi phục sự biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà đã được xác định khi kiểu phần vá là kiểu phần vá không liên tục.

Trong khối 2014, bộ phận xử lý của bộ giải mã xác định xem cờ khả dụng của nhiều phần vá có giá trị thứ nhất hay giá trị thứ hai. Trong khối 2016, sự biểu diễn thể tích khi được khôi phục được lưu trữ trong bộ nhớ của bộ giải mã khi cờ khả dụng của nhiều phần vá có giá trị thứ nhất. Khi tất cả các phần vá được khôi phục, the hình ảnh 3D có thể được hiển thị cho người dùng trên màn hiển thị của thiết bị điện tử (ví dụ như, điện thoại thông minh, máy tính bảng, máy tính xách tay, v.v.).

Trong khối 2018, bộ phận xử lý của bộ giải mã kết thúc xử lý khôi phục của dữ liệu thông tin phần vá được mã hóa khi cờ khả dụng của nhiều phần vá có giá trị thứ hai.

FIG.21 là phương án của phương pháp 2100 của PCC được thực hiện bởi bộ mã hóa (ví dụ như, bộ phận mã hóa entropi 56). Phương pháp 2100 có thể được thực hiện để mã hóa hình ảnh thể tích vào dòng bit để truyền về phía bộ giải mã. Trong khối 2102, bộ phận thu của bộ mã hóa thu nhận đơn vị dữ liệu khung phần vá (pdfu) mà nhận dạng kiểu phần vá (pdfu_patch_mode) cho mỗi trong số nhiều phần vá. Trong khối 2104, bộ phận xử lý của bộ mã hóa xác định xem kiểu phần vá dùng cho phần vá từ nhiều phần vá có là kiểu phần vá cuối.

Trong khối 2106, bộ phận xử lý của bộ mã hóa mã hóa dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá không phải là kiểu phần vá cuối, dữ liệu thông tin phần vá bao gồm kiểu phần vá dùng cho phần vá. Trong khối 2108, bộ phận xử lý của bộ mã hóa mã hóa dữ liệu thông tin phần vá dùng cho phần vá khi kiểu phần vá được thiết đặt là kiểu phần vá cuối. Dữ liệu thông tin phần vá bao gồm kiểu phần vá cuối dùng cho phần vá.

Trong phương án, dữ liệu thông tin phần vá và kiểu phần vá cuối được mã hóa vào dòng bit mà được truyền về phía bộ giải mã.

FIG.22 là phương án của phương pháp 2200 của PCC được thực hiện bởi bộ mã hóa (ví dụ như, bộ phận mã hóa entropi 56). Phương pháp 2200 có thể được thực hiện để mã hóa hình ảnh thể tích vào dòng bit để truyền về phía bộ giải mã. Trong khối 2202, bộ phận thu của bộ mã hóa thu nhận đơn vị dữ liệu khung phân vá (pfdu) cho mỗi trong số nhiều phân vá. Trong khối 2204, bộ phận xử lý của bộ mã hóa thêm cờ phân vá cuối vào mỗi trong số nhiều phân vá.

Trong khối 2206, bộ phận xử lý của bộ mã hóa xác định xem kiểu phân vá dùng cho phân vá từ nhiều phân vá có là kiểu phân vá cuối dựa trên giá trị của cờ phân vá cuối. Trong khối 2208, bộ phận xử lý của bộ mã hóa mã hóa dữ liệu thông tin phân vá dùng cho phân vá khi kiểu phân vá không phải là kiểu phân vá cuối, dữ liệu thông tin phân vá bao gồm kiểu phân vá và cờ phân vá cuối dùng cho phân vá.

Trong phương án, dữ liệu thông tin phân vá bao gồm kiểu phân vá và cờ phân vá cuối được mã hóa vào dòng bit mà được truyền về phía bộ giải mã.

FIG.23 là sơ đồ giản lược của thiết bị mã hóa 2300 (ví dụ như, bộ mã hóa 22, bộ giải mã 28, v.v.) theo phương án của sáng chế. Thiết bị mã hóa 2300 phù hợp để thực hiện các phương pháp và các quy trình được bộc lộ ở đây. Thiết bị mã hóa 2300 bao gồm các cổng xâm nhập 2310 và các đơn vị bộ phận thu (Rx) 2320 để thu dữ liệu; bộ phận xử lý, bộ logic, hoặc bộ phận xử lý trung tâm (CPU) 2330 để xử lý dữ liệu; các đơn vị bộ phận truyền (Tx) 2340 và các cổng xuất 2350 để truyền dữ liệu; và bộ nhớ 2360 để lưu trữ dữ liệu. Thiết bị mã hóa 2300 có thể cũng bao gồm các thành phần quang-điện (OE) và các thành phần điện-quang (EO) được ghép nối với các cổng vào 2310, các đơn vị bộ phận thu 2320, các đơn vị bộ phận truyền 2340, và các cổng ra 2350 để đưa ra hoặc đưa vào các tín hiệu quang hoặc các tín hiệu điện.

Bộ phận xử lý 2330 được thực hiện bởi phần cứng và phần mềm. Bộ phận xử lý 2330 có thể được thực hiện như một hoặc nhiều chip CPU, lõi (ví dụ như, như bộ phận xử lý đa lõi), các mảng cổng khả trình trường (FPGA), các mạch tích hợp ứng dụng cụ thể (ASIC), và các bộ phận xử lý tín hiệu số (DSP). Bộ phận xử lý 2330 truyền thông với các cổng vào 2310, các đơn vị bộ phận thu 2320, các đơn vị bộ phận truyền 2340, các cổng ra 2350, và bộ nhớ 2360. Bộ phận xử lý 2330 bao gồm môđun mã hóa

2370. Môđun mã hóa 2370 thực hiện các phương án được bộc lộ được mô tả ở trên. Trong phương án, môđun mã hóa 2370 là môđun khôi phục được tạo cấu hình để chiếu hình ảnh thể tích được khôi phục. Sự bao gồm của môđun mã hóa 2370 do đó đưa ra cải thiện căn bản cho chức năng của thiết bị mã hóa 2300 và tác động tới sự biến đổi của thiết bị mã hóa 2300 đến trạng thái khác. Ngoài ra, môđun mã hóa 2370 được thực hiện như các lệnh máy được lưu trữ trong bộ nhớ 2360 và được thực hiện bởi bộ phận xử lý 2330.

Thiết bị mã hóa 2300 có thể cũng bao gồm các thiết bị đầu vào và/hoặc đầu ra (I/O) 2380 để truyền thông dữ liệu đến và từ người dùng. Thiết bị I/O 2380 có thể bao gồm các thiết bị đầu ra như là màn hiển thị để hiển thị dữ liệu video, các loa để đưa ra dữ liệu audio, v.v. Các thiết bị I/O 2380 có thể cũng bao gồm các thiết bị đầu vào, như là bàn phím, chuột, con lăn, v.v., và/hoặc các giao diện tương ứng để tương tác với các thiết bị đầu ra này.

Bộ nhớ 2360 bao gồm một hoặc nhiều ổ, ổ băng, và ổ trạng thái cứng, và có thể được sử dụng làm thiết bị lưu trữ dữ liệu dư thừa, để lưu trữ các chương trình khi các chương trình này được lựa chọn cho việc thực thi, và lưu trữ các lệnh máy và dữ liệu mà được đọc trong xử lý thực thi chương trình. Bộ nhớ 2360 có thể là khả biến và bất khả biến, và có thể là bộ nhớ chỉ đọc (ROM), bộ nhớ truy nhập ngẫu nhiên (RAM), bộ nhớ lưu địa chỉ nội dung bậc ba (ternary content-addressable memory, TCAM), và bộ nhớ truy nhập ngẫu nhiên tĩnh (SRAM).

FIG.24 là sơ đồ giản lược của phương án của các phương tiện mã hóa 2400. Trong phương án, phương tiện để mã hóa 2400 được thực hiện trong thiết bị mã hóa 2402 (ví dụ như, bộ mã hóa 22 hoặc bộ giải mã 28). Thiết bị mã hóa 2402 bao gồm phương tiện thu 2401. Phương tiện thu 2401 được tạo cấu hình để thu hình ảnh để mã hóa hoặc để thu dòng bit để giải mã. Thiết bị mã hóa 2402 bao gồm phương tiện truyền 2407 được ghép nối với phương tiện thu 2401. Phương tiện truyền 2407 được tạo cấu hình để truyền dòng bit đến bộ giải mã hoặc để truyền hình ảnh được giải mã đến các phương tiện hiển thị (ví dụ như, một trong số các thiết bị I/O 2380).

Thiết bị mã hóa 2402 bao gồm phương tiện lưu trữ 2403. Phương tiện lưu trữ 2403 được ghép nối với ít nhất một trong phương tiện thu 2401 hoặc phương tiện truyền 2407. Phương tiện lưu trữ 2403 được tạo cấu hình để lưu trữ các lệnh máy. Thiết bị mã hóa 2402 cũng bao gồm phương tiện xử lý 2405. Các phương tiện xử lý 2405 được ghép nối với phương tiện lưu trữ 2403. Các phương tiện xử lý 2405 được tạo cấu hình để thực thi các lệnh máy được lưu trữ trong phương tiện lưu trữ 2403 để thực hiện các phương pháp được bộc lộ ở đây.

Trong phương án, cú pháp phù hợp để thực hiện các khái niệm được bộc lộ ở đây được đề xuất.

Đơn vị dữ liệu khung phần vá (pfdu). Trong phương án, loại đơn vị dữ liệu phần vá kết thúc đặc biệt được đưa ra để chỉ báo việc kết thúc đơn vị dữ liệu khung phần vá.

patch_frame_data_unit(frmIdx) {	Nhãn mô tả
p = -1	
while (pfdu_patch_mode[frmIdx][p] != LAST) {	
p ++	
pfdu_patch_mode [frmIdx][p]	ae(v)
patch_information_data(frmIdx, p, pfdu_patch_mode[frmIdx][p])	
}	
PfdTotalNumberOfPatches[frmIdx] = p	
if(sps_point_local_reconstruction_enabled_flag)	
point_local_reconstruction()	
byte_alignment()	
}	

Trong phương án, cờ một bit (1-bit) cụ thể được thêm vào mỗi đơn vị dữ liệu phần vá để chỉ báo sự chấm dứt đơn vị dữ liệu khung phần vá.

patch_frame_data_unit(frmIdx) {	Nhãn mô tả
-----------------------------------	-------------------

p = -1	
pfd_u_more_patches_available_flag	ae(v)
while (more_patches_available_flag) {	
p ++	
pfd_u_patch_mode [frmIdx][p]	ae(v)
patch_information_data(frmIdx, p, pfd_u_patch_mode[frmIdx][p])	
pfd_u_more_patches_available_flag	ae(v)
}	
Pfd_uTotalNumberOfPatches[frmIdx] = p + 1	
if(sps_point_local_reconstruction_enabled_flag)	
point_local_reconstruction()	
byte_alignment()	
}	

Trong phương án, dữ liệu thông tin phân vá như sau:

patch_information_data (frmIdx, p, patch_mode) () {	Nhãn mô tả
if(patch_mode == SKIP) {	
skip_patch_data_unit(frmIdx, p)	
}	
else if(patch_mode == INTRA) {	
if(pfps_local_override_geometry_patch_enable_flag)	
pid_override_geometry_patch_flag [frmIdx][p]	ae(v)
if(pid_override_geometry_patch_flag[frmIdx][p])	
pid_geometry_patch_parameter_set_id [frmIdx][p]	ae(v)
for(i = 0; i < sps_attribute_count; i++)	

if(pfps_local_override_attribute_patch_enable_flag[i]) {	
pid_override_attribute_patch_flag[p][i]	ae(v)
if(pid_override_attribute_patch_flag[p][i])	
pid_attribute_patch_parameter_set_id[p][i]	ae(v)
}	
patch_data_unit(frmIdx, p)	
}	
else if(patch_mode == P_INTER)	
delta_patch_data_unit(frmIdx, p)	
else if(patch_mode == I_PCM patch_mode == P_PCM)	
pcm_patch_data_unit(frmIdx, p)	
}	

Trong phương án, phần tử cú pháp delta_patch_data_unit gốc là như sau:

delta_patch_data_unit(frmIdx, patchIndex) {	Nhãn mô tả
dpdu_patch_index[frmIdx][patchIndex]	ae(v)
dpdu_2d_shift_u[frmIdx][patchIndex]	ae(v)
dpdu_2d_shift_v[frmIdx][patchIndex]	ae(v)
dpdu_2d_delta_size_u[frmIdx][patchIndex]	ae(v)
dpdu_2d_delta_size_v[frmIdx][patchIndex]	ae(v)
dpdu_3d_shift_tangent_axis[frmIdx][patchIndex]	ae(v)
dpdu_3d_shift_bitangent_axis[frmIdx][patchIndex]	ae(v)
dpdu_3d_shift_normal_axis[frmIdx][patchIndex]	ae(v)
projectionFlag = 0	
i = 0	
while (i < sps_layer_count_minus1 + 1 && projectionFlag == 0) {	

projectionFlag	
= projectionFlag sps_layer_absolute_coding_enabled_flag[i]	
i++	
}	
if (projectionFlag)	
dpdu_projection_mode [frmIdx][patchIndex]	ae(v)
}	

Trong phương án, phần tử cú pháp added skip_patch_data_unit là như sau:

skip_patch_data_unit(frmIdx, patchIndex) {	Nhãn mô tả
spdu_patch_index [frmIdx][patchIndex]	ae(v)
}	

Vì loại đơn vị dữ liệu phần vá không liên tục được đưa ra, lượng thông tin tiêu đề trong việc biểu diễn dòng bit được mã hóa được đưa ra.

Ngữ nghĩa của các thành phần mới được thêm vào là như sau.

spdu_patch_index[frmIdx][p] cộng p chỉ định chỉ số, PredIdx, của phần vá trong khung phần vá với chỉ số RefIdx mà tương ứng với khung phần vá thứ nhất trong danh sách khung phần vá tham chiếu.

Xử lý giải mã cho các đơn vị dữ liệu phần vá được mã hóa trong chế độ dự đoán không liên tục là như sau. Khi pfdpu_patch_mode[frmIdx][p] bằng P_SKIP, sau đó xử lý để giải mã các đơn vị dữ liệu phần vá được mã hóa không liên tục được sử dụng, với frmIdx và p là các đầu vào cho xử lý và là các đầu ra của xử lý (Patch2dShiftU, Patch2dShiftV, Patch2dSizeU, Patch2dSizeV, Patch3dShiftT, Patch3dShiftBT, Patch3dShiftN, PatchNormalAxis, Orientation, PatchLod) được sử dụng làm đầu ra.

Từ phần nêu trên, cần được biết rằng định hướng của phần vá có thể là khác nhau đối với quá thể hiện mặc định. Việc định hướng của phần vá có thể được báo hiệu theo cách thức đơn giản bằng cách sử dụng cờ 1-bit. Cơ chế để chuyển đổi giữa định hướng mặc định và được ưu tiên được đưa ra.

Trong khi đó các phương án được đề xuất trong sáng chế, có thể được hiểu rằng các hệ thống và phương pháp được bộc lộ có thể được thực hiện theo các dạng cụ thể khác mà không nằm ngoài bản chất hoặc phạm vi của sáng chế. Các ví dụ hiện tại được xem xét mang tính minh họa và không giới hạn, và sáng chế không cần được giới hạn ở chi tiết được cho ở đây. Ví dụ, các thành phần hoặc phần tử khác nhau có thể được kết hợp hoặc được tích hợp trong một hệ thống khác hoặc các đặc trưng nhất định có thể được bỏ qua, hoặc không được thực hiện.

Ngoài ra, các kỹ thuật, hệ thống, hệ thống con, và phương pháp được mô tả và được minh họa trong các phương án khác nhau là rời rạc hoặc riêng biệt có thể được kết hợp hoặc được tích hợp với các hệ thống, thành phần, kỹ thuật, hoặc phương pháp khác mà không nằm ngoài phạm vi của sáng chế. Ví dụ khác về sự thay đổi, thay thế có thể xác định được bởi người có hiểu biết trung bình trong lĩnh vực kỹ thuật và có thể được thực hiện mà không nằm ngoài bản chất và phạm vi được bộc lộ ở đây.

Sáng chế này yêu cầu hưởng quyền ưu tiên của Đơn xin cấp Patent tạm thời Mỹ số 62/817,391, được nộp ngày 12/03/2019, bởi Vladyslav Zakharchenko, (và các tác giả khác)., và có tên “Mã hóa và Giải mã đơn vị dữ liệu phần vá dùng cho Mã hóa đám mây điểm” mà được kết hợp ở đây nhằm mục đích tham chiếu.

YÊU CẦU BẢO HỘ

1. Phương pháp mã hóa đám mây điểm (PCC - point cloud coding) được thực hiện bởi bộ giải mã, bao gồm:

thu, bởi bộ giải mã, dữ liệu thông tin phần vá được mã hóa;

thu nhận, bởi bộ giải mã, phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá mà có kiểu phần vá (patch_mode);

xác định, theo cờ phần vá bởi bộ giải mã, xem liệu kiểu phần vá đối với phần vá có phải là kiểu phần vá cuối cùng hay không, trong đó cờ phần vá chỉ báo xem liệu kiểu phần vá cho phần vá là kiểu phần vá không liên tục, kiểu phần vá liên đới, kiểu phần vá nội bộ hay kiểu phần vá cuối cùng; và

kết thúc, bởi bộ giải mã, xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối cùng.

2. Phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ giải mã, bao gồm:

thu, bởi bộ giải mã, dữ liệu thông tin phần vá được mã hóa;

thu nhận, bởi bộ giải mã, phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá mà có kiểu phần vá (patch_mode);

xác định, bởi bộ giải mã, mà kiểu phần vá đối với phần vá là kiểu phần vá liên đới;

giải mã, bởi bộ giải mã, chỉ số phần vá tham chiếu thứ hai (dpdu_patch_index) tương ứng với phần vá;

xác định, bởi bộ giải mã, chỉ số tham chiếu (refIdx) đối với phần vá dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu thứ hai như được giải mã;

giải mã, bởi bộ giải mã, phần vá mà sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá, trong đó các thành phần hai chiều (2D) và các thành phần ba chiều (3D) thu nhận được dựa trên chỉ số tham chiếu (refIdx) đối với phần vá; và

khôi phục, bởi bộ giải mã, biểu diễn thể tích dựa trên phần vá như được giải mã.

3. Phương pháp theo điểm 2, còn bao gồm:

xác định, mà kiểu phần vá đối với phần vá là kiểu phần vá nội bộ;

xác định chỉ số tham chiếu (refIdx) đối với phần vá dựa trên chỉ số khung ([frmIdx]) tương ứng với phần vá và chỉ số phần vá chỉ số phần vá giảm dần;

giải mã phần vá mà sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và

khôi phục biểu diễn thể tích dựa trên phần vá như được giải mã.

4. Phương pháp theo điểm bất kỳ trong số các điểm 2 đến 3, còn bao gồm:

xác định, bởi bộ giải mã, mà kiểu phần vá đối với phần vá là kiểu phần vá không liên tục;

giải mã, bởi bộ giải mã, chỉ số phần vá tham chiếu (spdu_patch_index) tương ứng với phần vá khi kiểu phần vá là kiểu phần vá không liên tục;

xác định, bởi bộ giải mã, chỉ số tham chiếu (refIdx) đối với phần vá dựa trên chỉ số khung tham chiếu ([refFrmIdx]) tương ứng với phần vá và chỉ số phần vá tham chiếu như được giải mã khi kiểu phần vá là kiểu phần vá không liên tục; và

khôi phục, bởi bộ giải mã, biểu diễn thể tích của phần vá dựa trên chỉ số tham chiếu mà được xác định khi kiểu phần vá là kiểu phần vá không liên tục.

5. Phương pháp theo điểm bất kỳ trong số các điểm 2 đến 4, còn bao gồm:

xác định rằng kiểu phần vá đối với phần vá là kiểu phần vá điều chế mã xung (PCM);

xác định chỉ số tham chiếu (refIdx) đối với phần vá dựa trên chỉ số khung ([frmIdx]) tương ứng với phần vá và chỉ số phần vá chỉ số phần vá giảm dần;

giải mã phần vá mà sử dụng các thành phần hai chiều (2D) đối với các điểm độc lập tương ứng với phần vá và các thành phần ba chiều (3D) đối với các điểm độc lập tương ứng với phần vá; và

khôi phục biểu diễn thể tích dựa trên phần vá như được giải mã.

6. Phương pháp theo điểm bất kỳ trong số các điểm 2 đến 5, còn bao gồm việc thu đầu vào tương ứng với dữ liệu thông tin phần vá được mã hóa, trong đó đầu vào bao gồm một hoặc nhiều `patch_mode`, chỉ số phần vá, chỉ số tham chiếu, chỉ số khung, và chỉ số khung tham chiếu.

7. Phương pháp theo điểm bất kỳ trong số các điểm 2 đến 6, còn bao gồm: việc hiển thị, trên bộ phận hiển thị của thiết bị điện tử, hình ảnh được khởi tạo dựa trên biểu diễn thể tích như được khôi phục.

8. Phương pháp theo điểm bất kỳ trong số các điểm 2 đến 6, còn bao gồm:

xác định, bởi bộ giải mã, xem liệu cờ khả dụng của nhiều phần vá có giá trị thứ nhất hay là giá trị thứ hai;

lưu trữ, trong bộ nhớ của bộ giải mã, biểu diễn thể tích như được khôi phục khi cờ khả dụng của nhiều phần vá có giá trị thứ nhất; và

kết thúc, bởi bộ giải mã, xử lý khôi phục của dữ liệu thông tin phần vá được mã hóa khi cờ khả dụng của nhiều phần vá có giá trị thứ hai.

9. Phương pháp theo điểm 8, còn bao gồm:

xác định, mà kiểu phần vá đối với phần vá là kiểu phần vá nội bộ;

xác định chỉ số tham chiếu (`refIdx`) đối với phần vá dựa trên chỉ số khung (`[frmIdx]`) tương ứng với phần vá và chỉ số phần vá chỉ số phần vá giảm dần;

giải mã phần vá mà sử dụng các thành phần hai chiều (2D) tương ứng với phần vá và các thành phần ba chiều (3D) tương ứng với phần vá; và

khôi phục biểu diễn thể tích dựa trên phần vá như được giải mã.

10. Phương pháp mã hóa đám mây điểm (PCC) được thực hiện bởi bộ mã hóa, bao gồm:

thu nhận, bởi bộ mã hóa, đơn vị dữ liệu khung phần vá (`pfdv`) mà nhận dạng kiểu phần vá (`pfdv_patch_mode`) đối với mỗi trong số nhiều phần vá;

xác định, bởi bộ mã hóa, liệu kiểu phần vá đối với mỗi phần vá từ nhiều phần vá là kiểu phần vá cuối cùng; và

mã hóa, bởi bộ mã hóa, dữ liệu thông tin phần vá đối với phần vá, dữ liệu thông tin phần vá bao gồm cờ phần vá, trong đó cờ phần vá chỉ báo xem liệu kiểu phần vá đối với phần vá là kiểu phần vá không liên tục, kiểu phần vá liên đới, kiểu phần vá nội bộ hay kiểu phần vá cuối cùng.

11. Phương pháp theo điểm 10, trong đó đơn bị dữ liệu khung phần vá bao gồm chỉ số khung (`frmIdx`), các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

12. Phương pháp theo điểm 10 hoặc 11, trong đó dữ liệu thông tin phần vá bao gồm chỉ số khung, các thành phần hai chiều (2D) tương ứng với phần vá, và các thành phần ba chiều (3D) tương ứng với phần vá.

13. Phương pháp theo điểm bất kỳ trong số các điểm 10 đến 12, còn bao gồm việc lặp lại các bước xác định xem liệu kiểu phần vá là kiểu phần vá cuối cùng và việc mã hóa dữ liệu thông tin phần vá đối với các phần vá tiếp theo từ nhiều phần vá đến khi một trong các phần vá từ nhiều phần vá được xác định để có kiểu phần vá cuối cùng.

14. Thiết bị giải mã, bao gồm:

bộ thu có cấu trúc để thu dữ liệu thông tin phần vá được mã hóa, trong đó dữ liệu thông tin phần vá bao gồm cờ phần vá;

bộ nhớ được ghép nối tới bộ thu, bộ nhớ mà lưu trữ các lệnh; và

bộ xử lý được ghép nối tới bộ nhớ, bộ xử lý có cấu trúc để thực thi các lệnh để làm cho thiết bị giải mã:

thu nhận phần vá tương ứng với dữ liệu thông tin phần vá được mã hóa, phần vá mà có kiểu phần vá (`patch_mode`);

xác định, theo cờ phần vá, xem liệu kiểu phần vá đối với phần vá là kiểu phần vá cuối cùng, mà trong đó cờ phần vá chỉ báo xem liệu kiểu phần vá đối với phần vá là kiểu phần vá không liên tục, kiểu phần vá liên đới, kiểu phần vá nội bộ hay kiểu phần vá cuối cùng ; và

kết thúc xử lý khôi phục tương ứng với dữ liệu thông tin phần vá được mã hóa khi kiểu phần vá là kiểu phần vá cuối cùng.

15. Thiết bị giải mã, bao gồm:

Bộ thu có cấu trúc để thu dữ liệu thông tin phân vá được mã hóa;

bộ nhớ được ghép nối tới bộ thu, bộ nhớ mà lưu trữ các lệnh; và

bộ xử lý được ghép nối với bộ nhớ, bộ xử lý có cấu trúc để thực thi các lệnh để làm cho thiết bị giải mã thực hiện phương pháp theo các điểm 2 đến 9.

16. Thiết bị mã hóa, bao gồm:

bộ thu có cấu trúc để thu hình ảnh ba chiều (3D);

bộ nhớ được ghép nối tới bộ thu, bộ nhớ mà chứa các lệnh;

bộ xử lý được ghép nối với bộ nhớ, bộ xử lý có cấu trúc để thực thi các lệnh để làm cho thiết bị mã hóa thực hiện phương pháp theo các điểm 10 đến 13.

17. Phương tiện lưu trữ đọc được bởi máy tính, lưu trữ dòng bit mã hóa thu nhận được bằng cách thực hiện phương pháp theo điểm bất kỳ trong số các điểm 10 đến 13 bởi một hoặc nhiều bộ xử lý, trong đó dòng bit được mã hóa bao gồm dữ liệu thông tin phân vá được mã hóa.

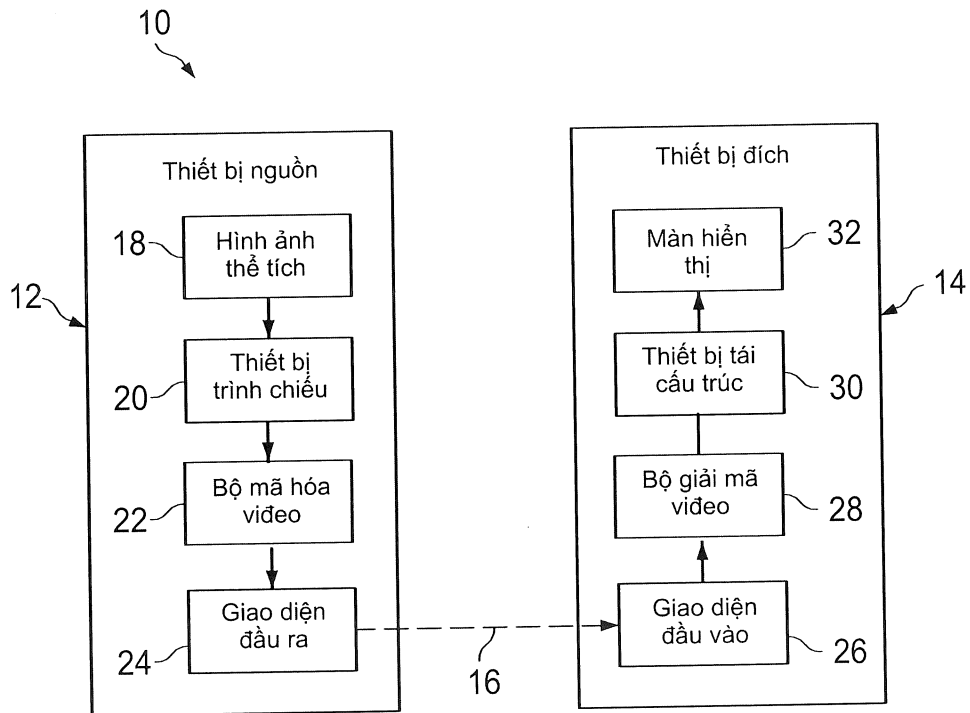


FIG. 1

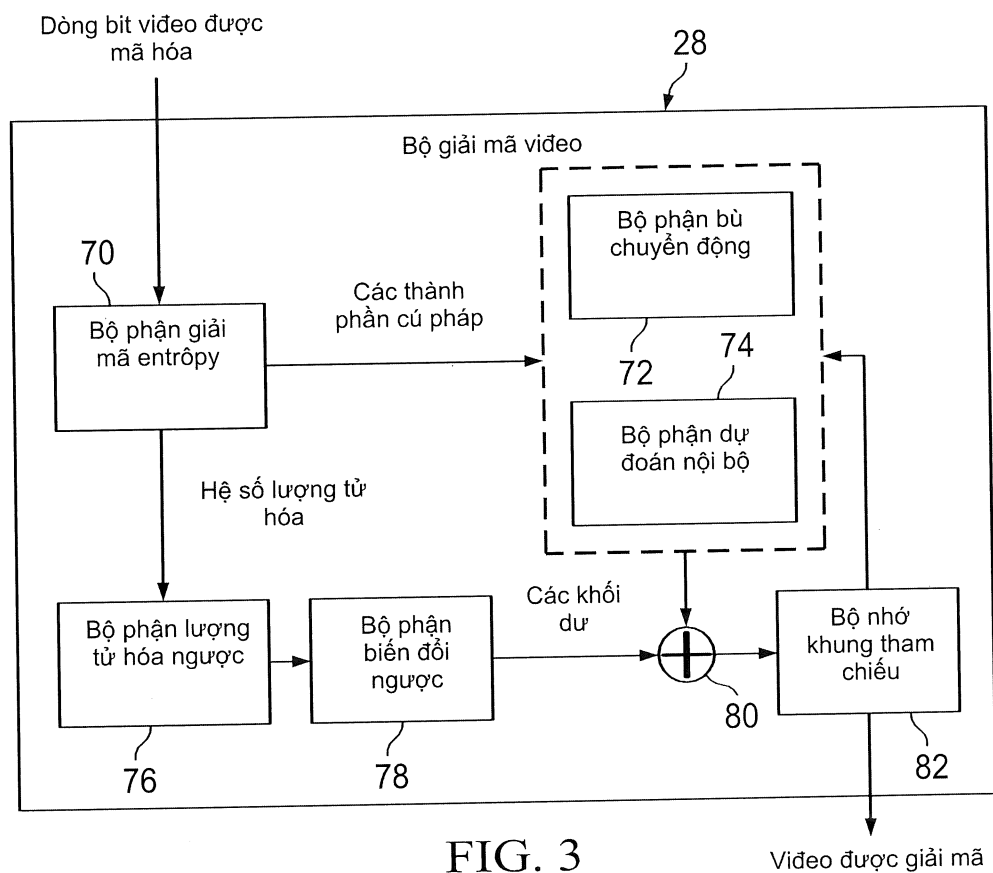


FIG. 3

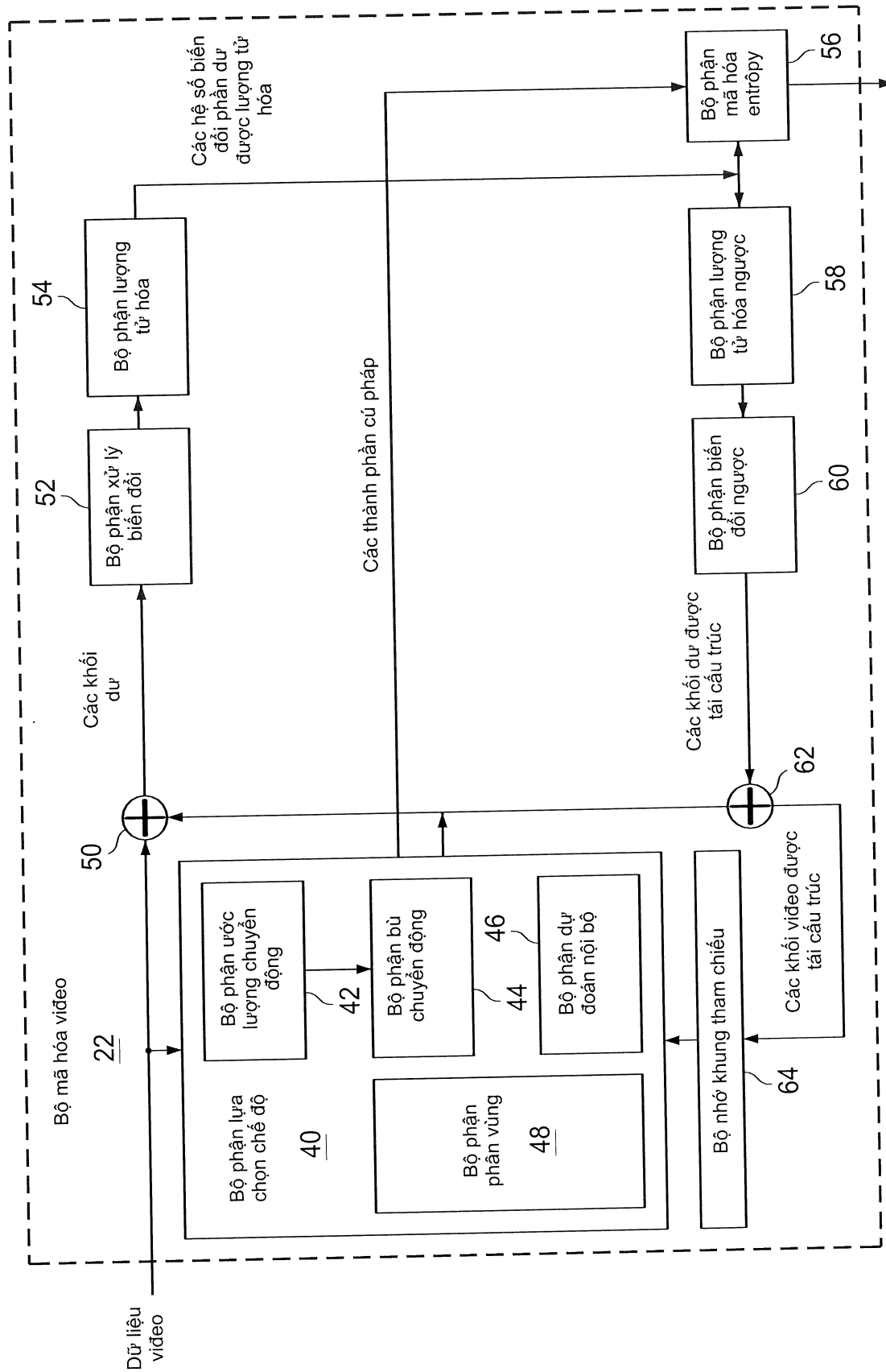


FIG. 2

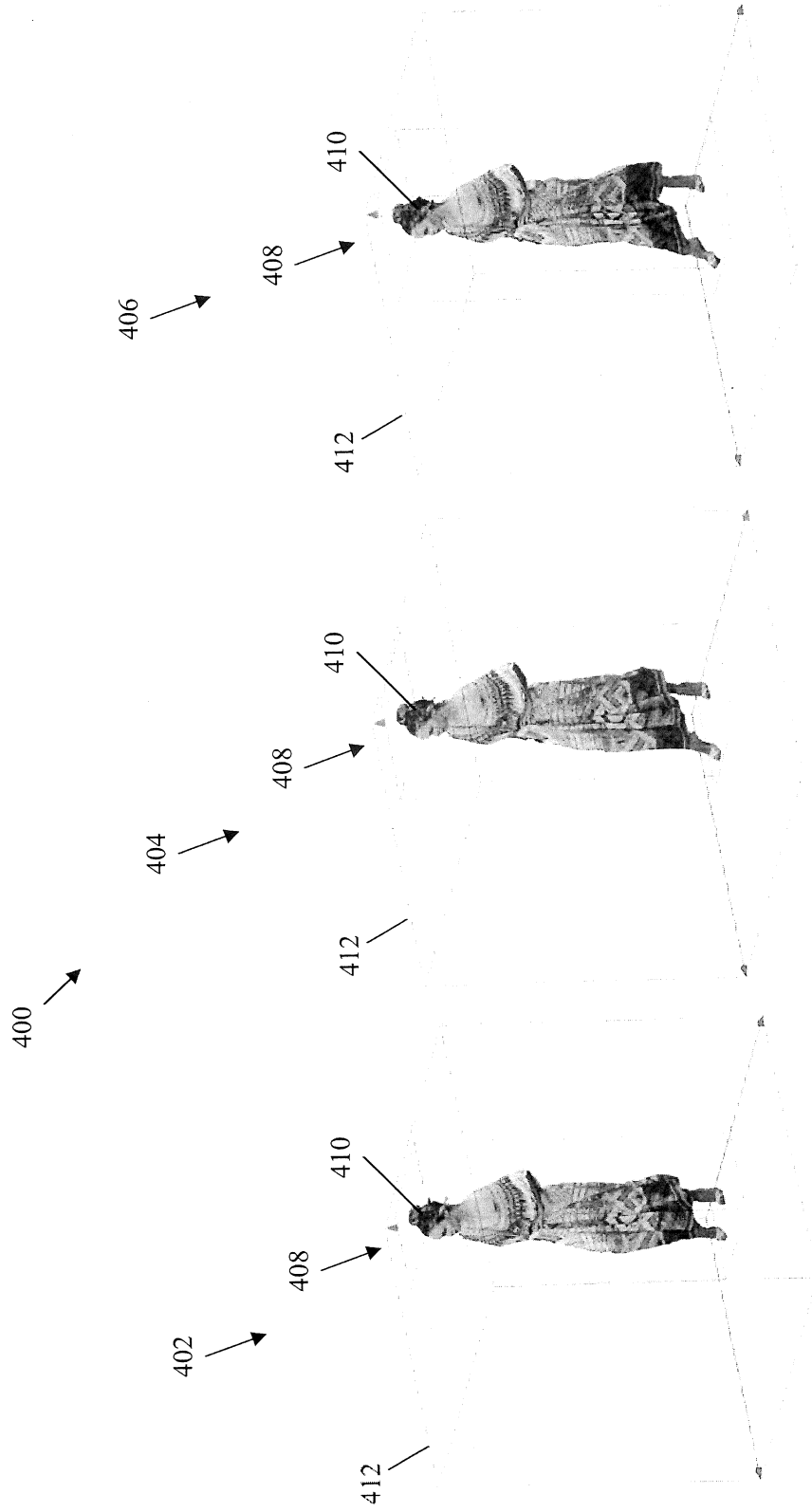


FIG. 4

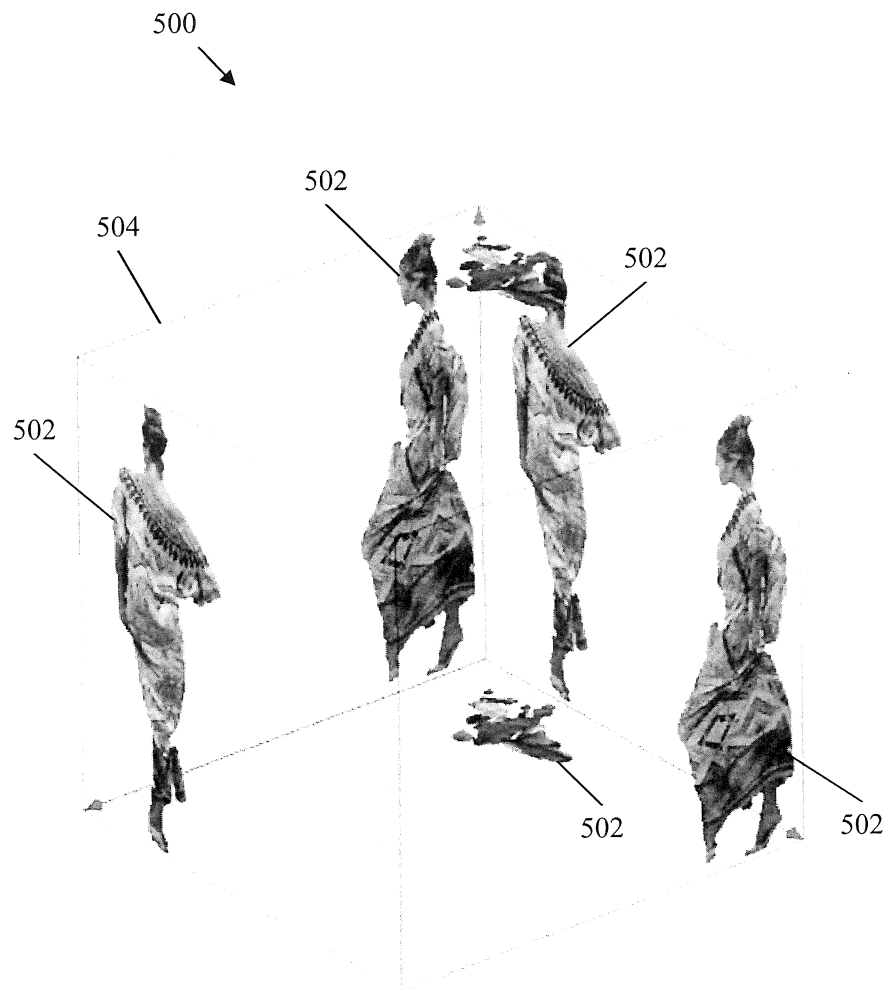


FIG. 5

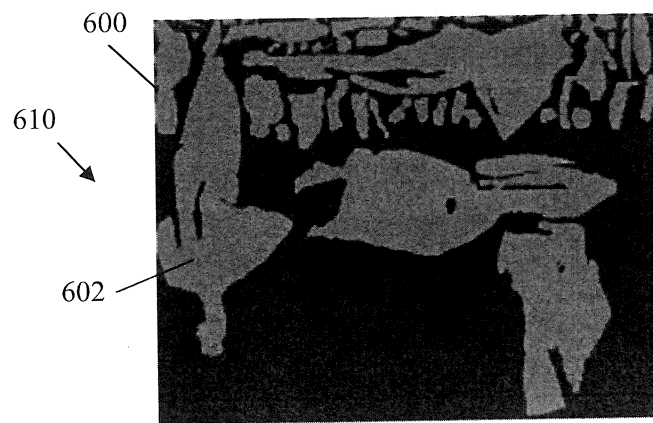


FIG. 6A

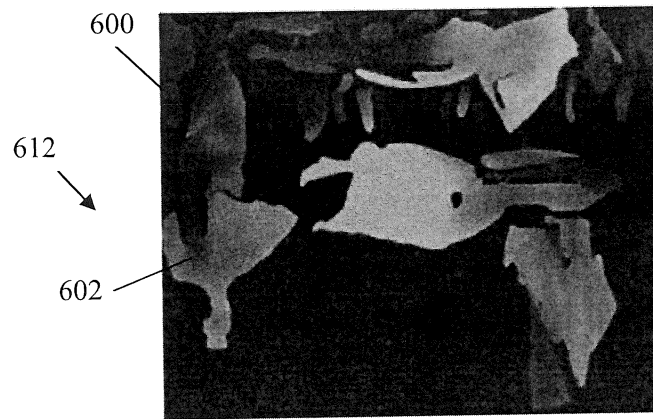


FIG. 6B

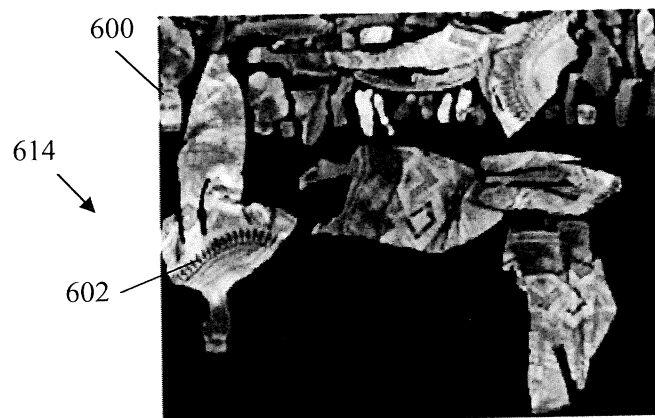


FIG. 6C

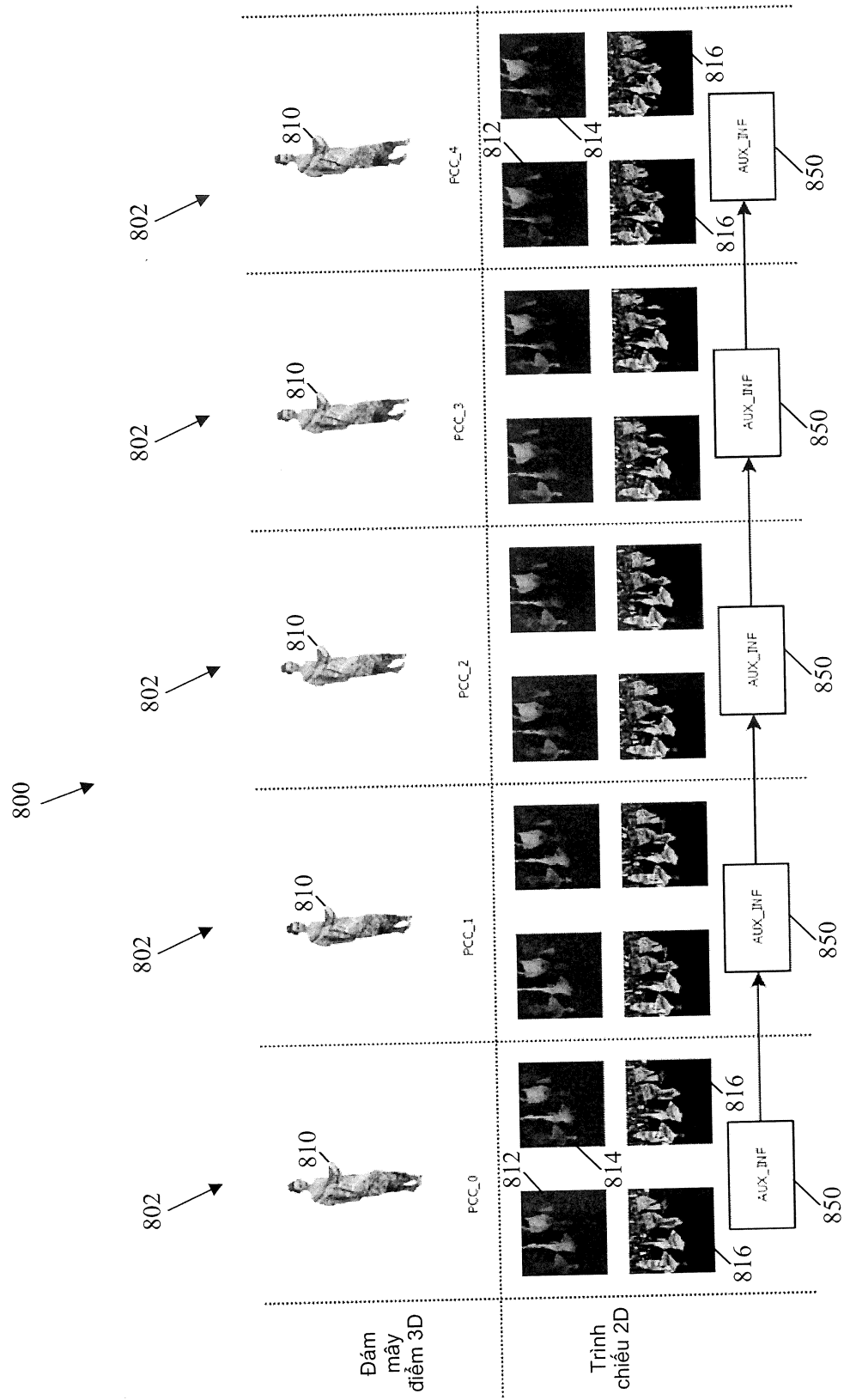


FIG. 8

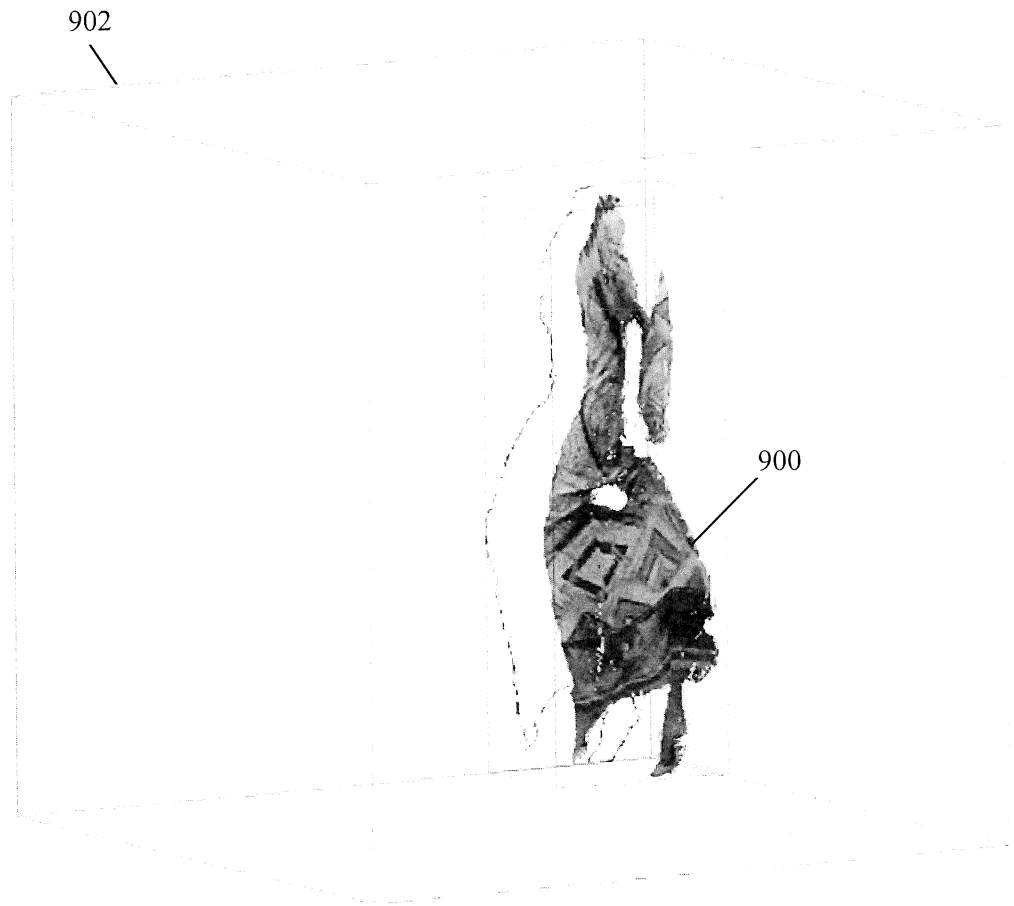


FIG. 9

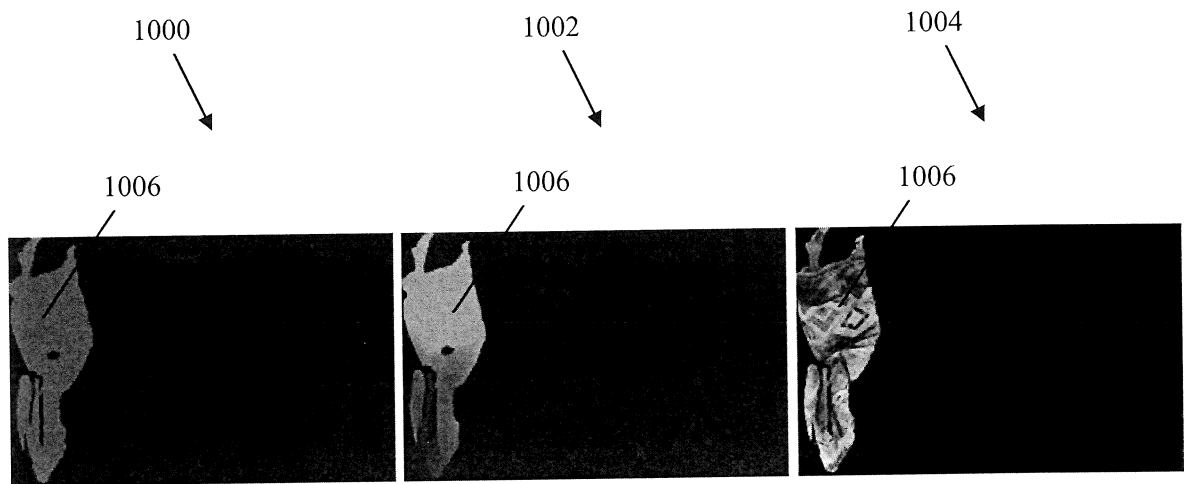


FIG. 10A

FIG. 10B

FIG. 10C

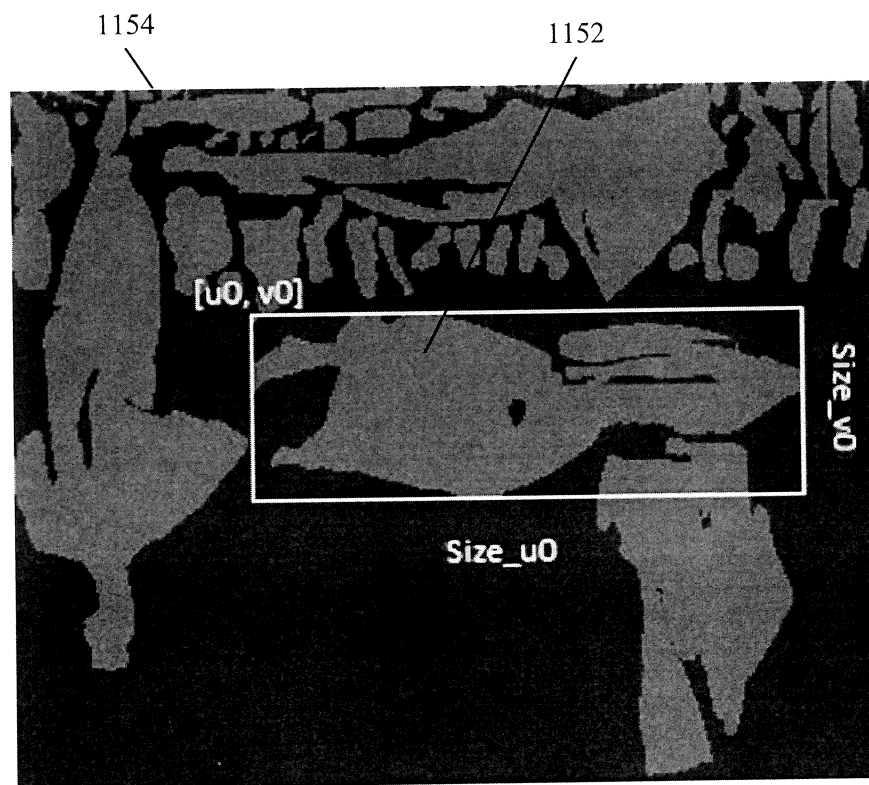
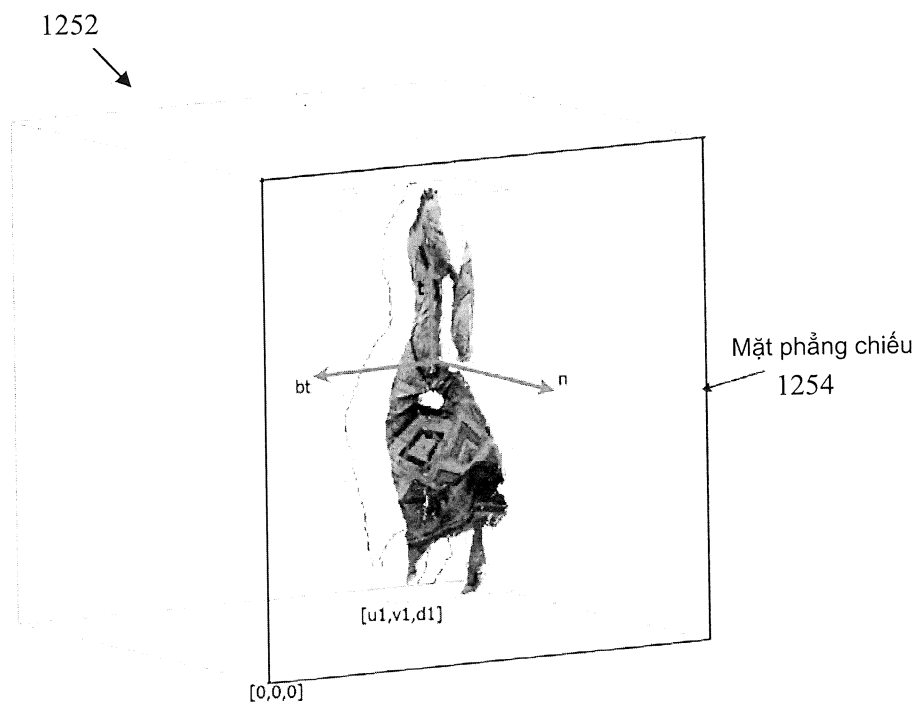
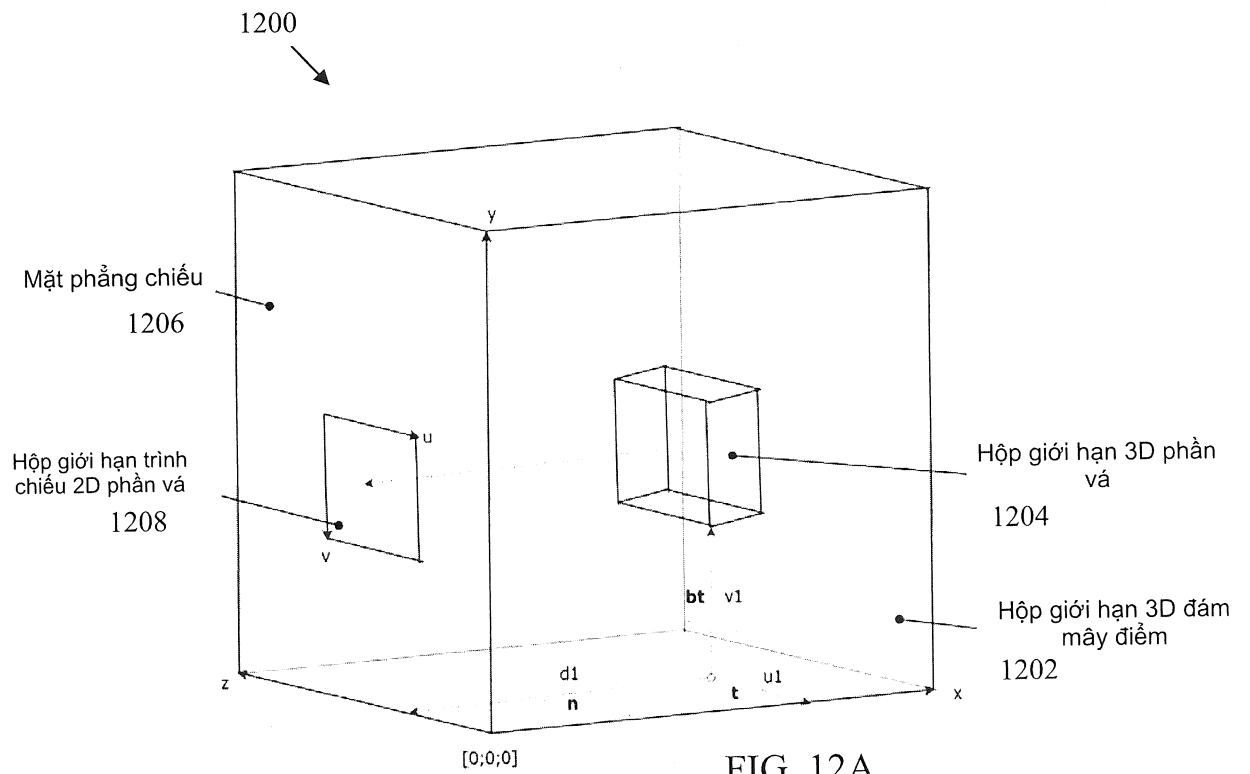


FIG. 11



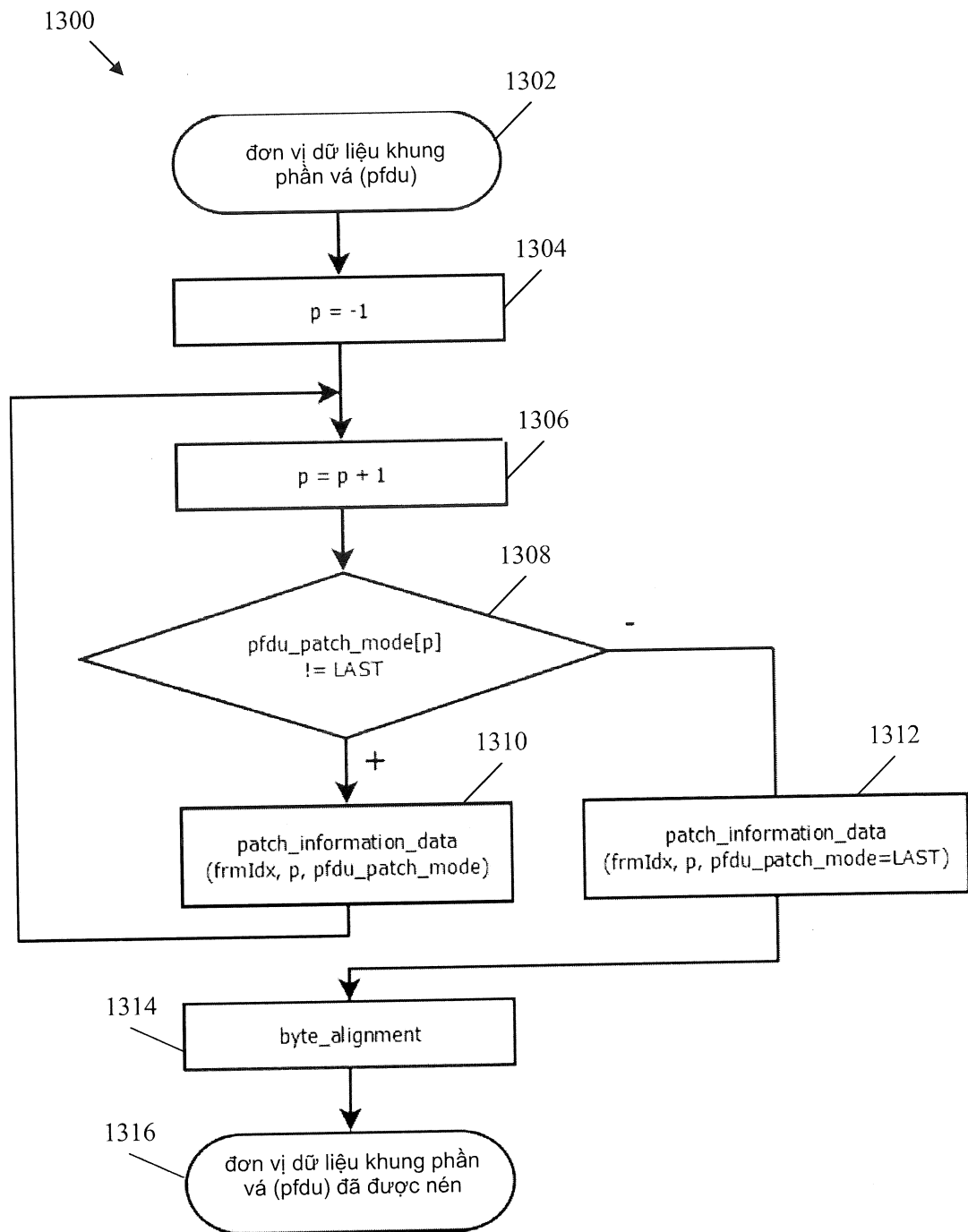


FIG. 13

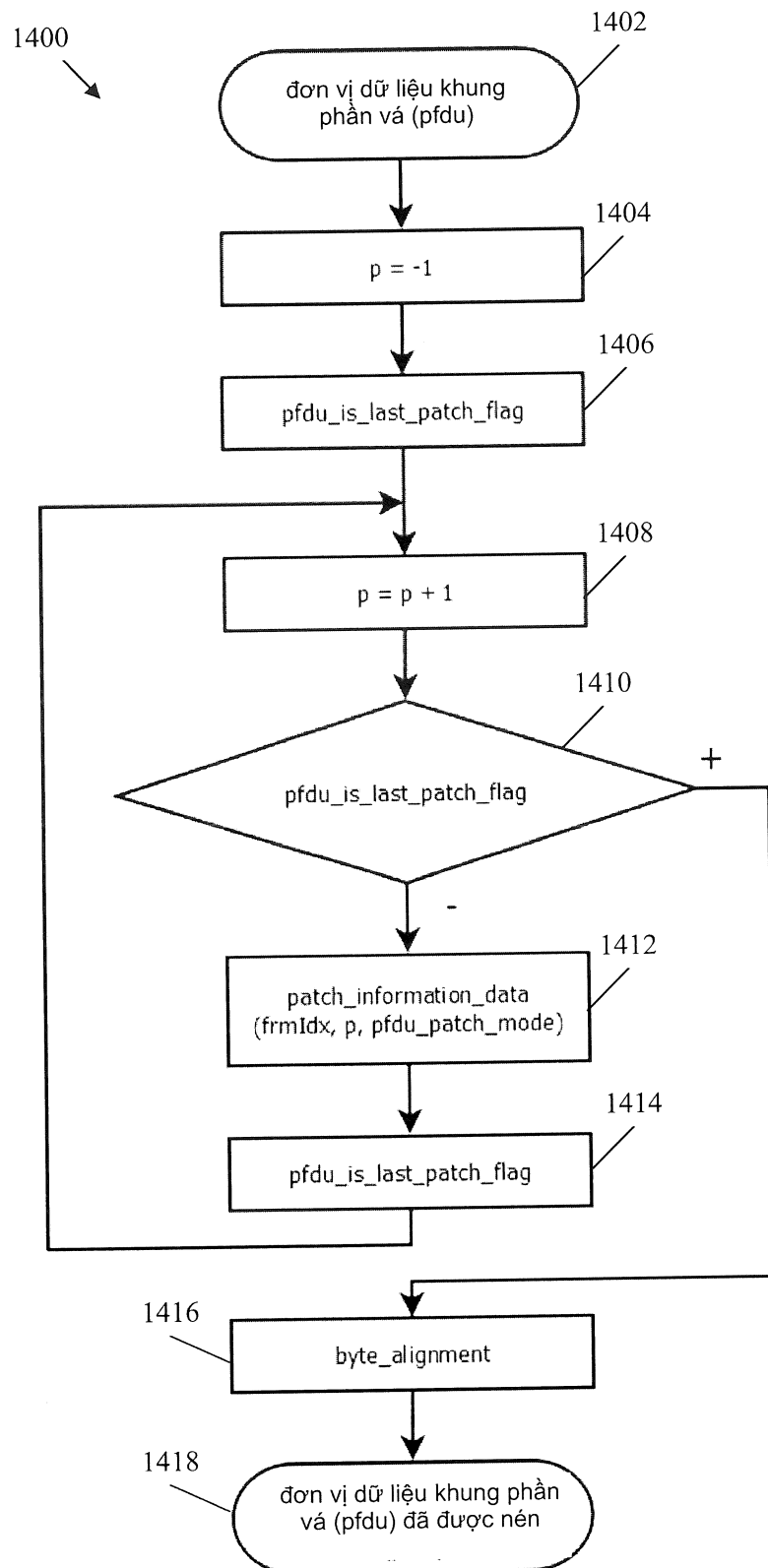


FIG. 14

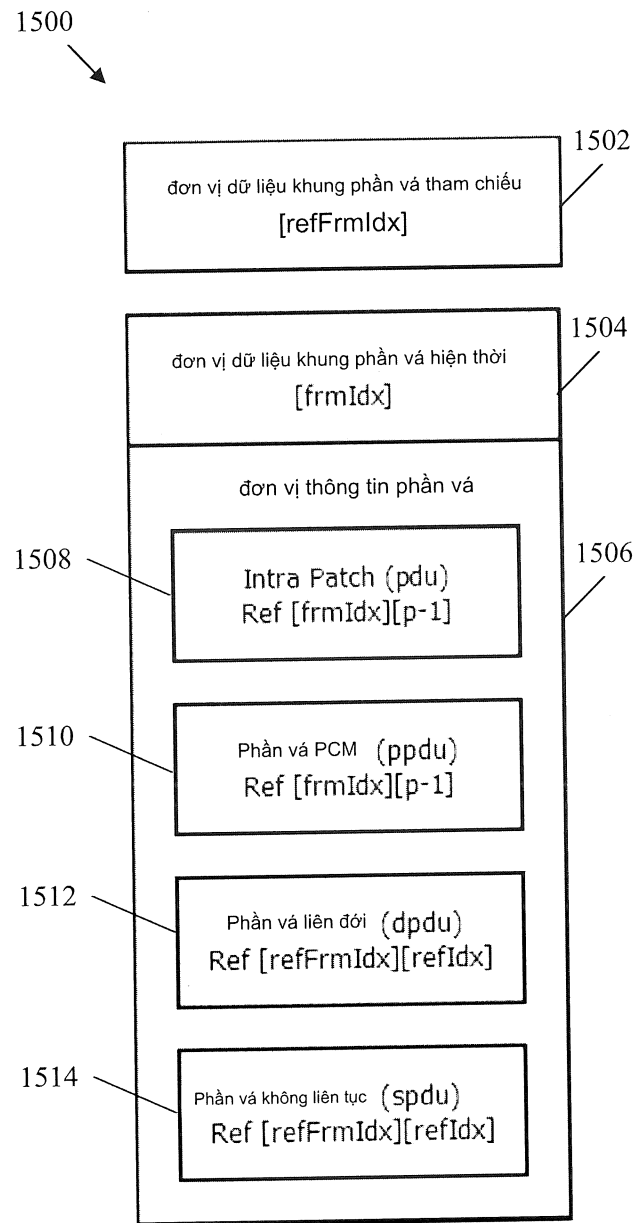
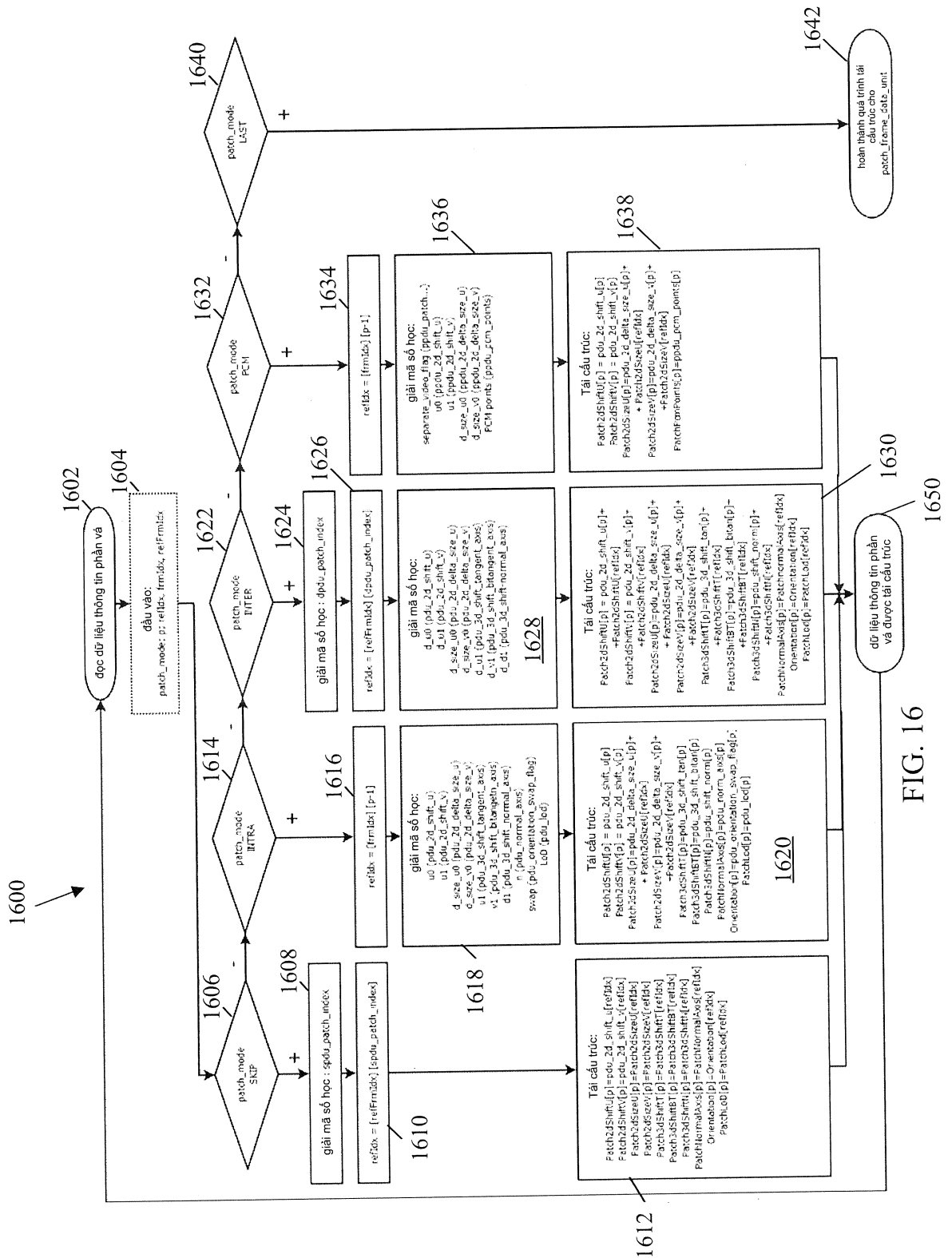


FIG. 15



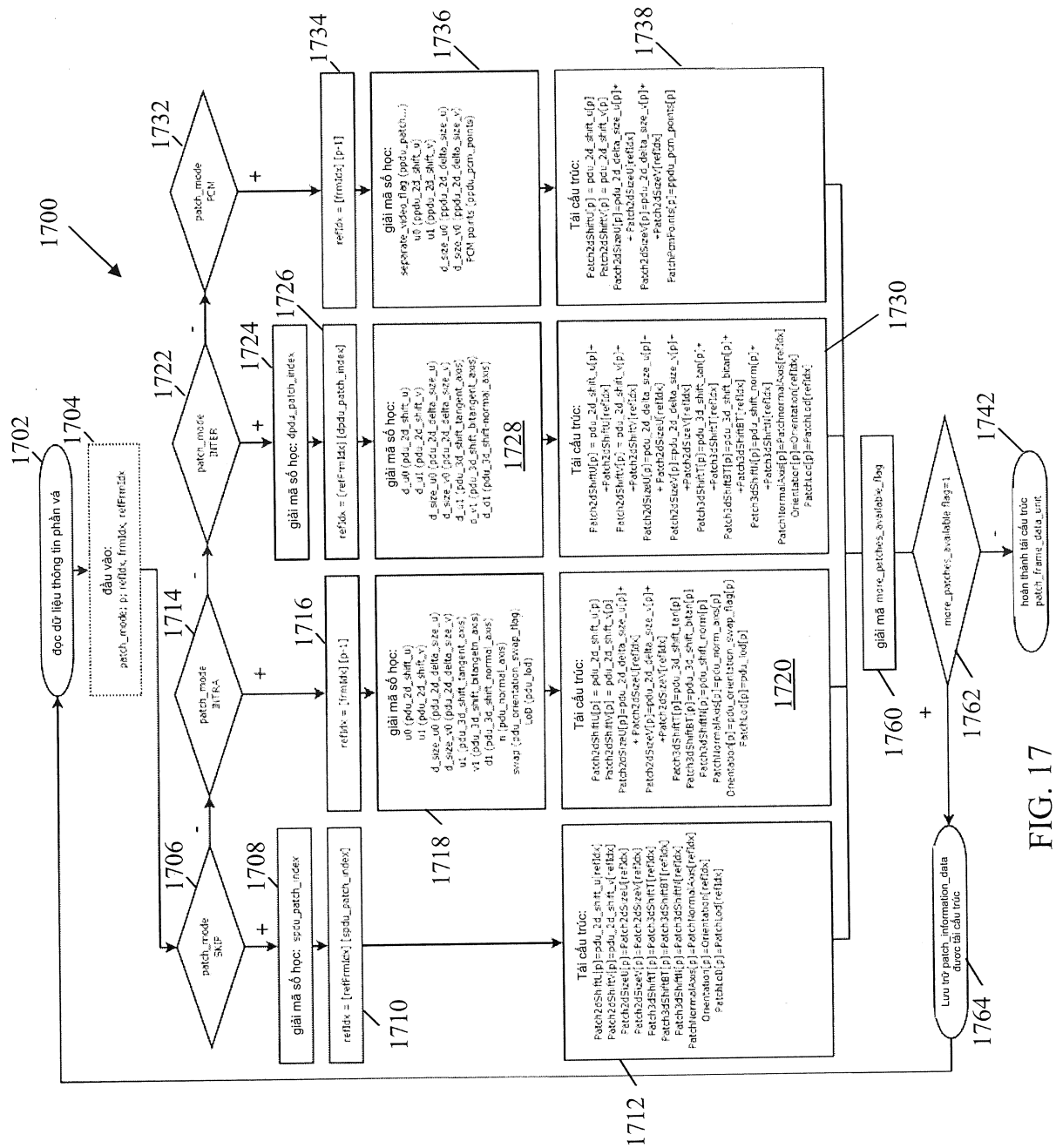


FIG. 17

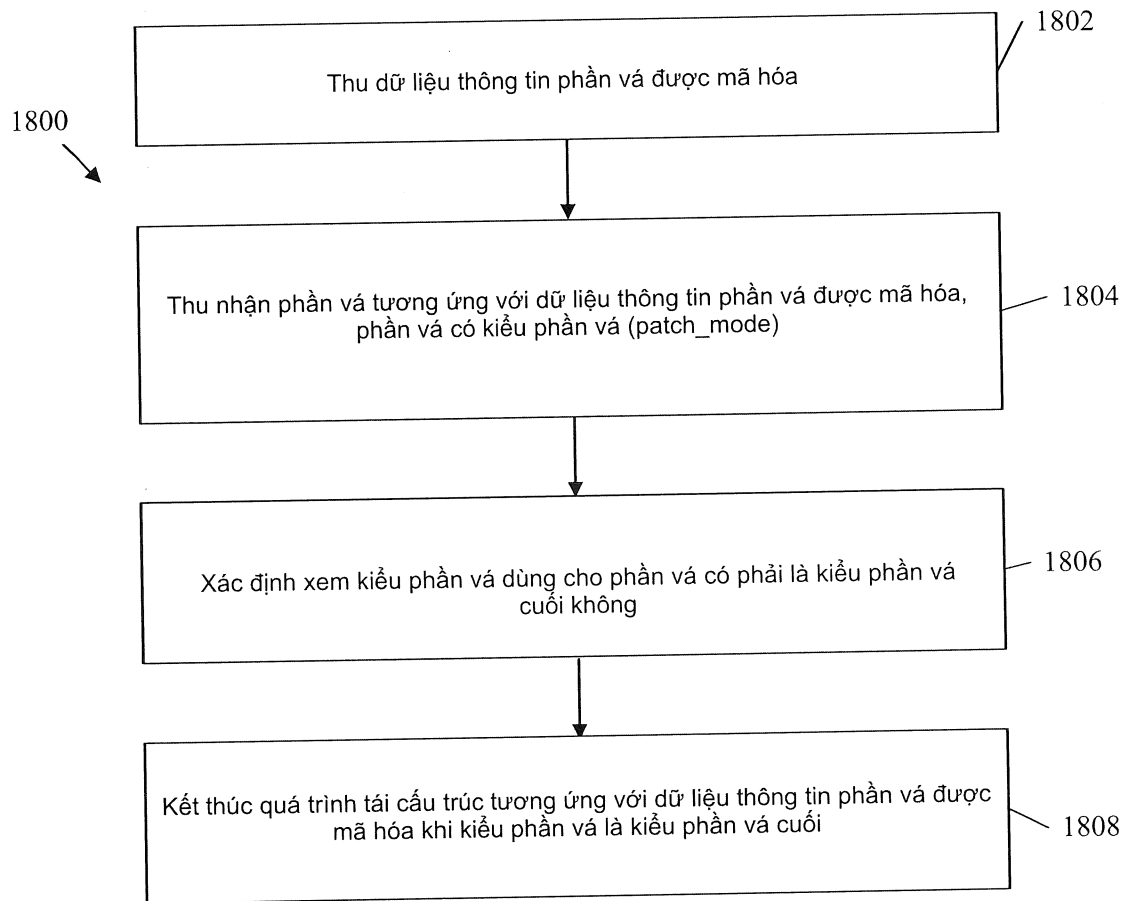


FIG. 18

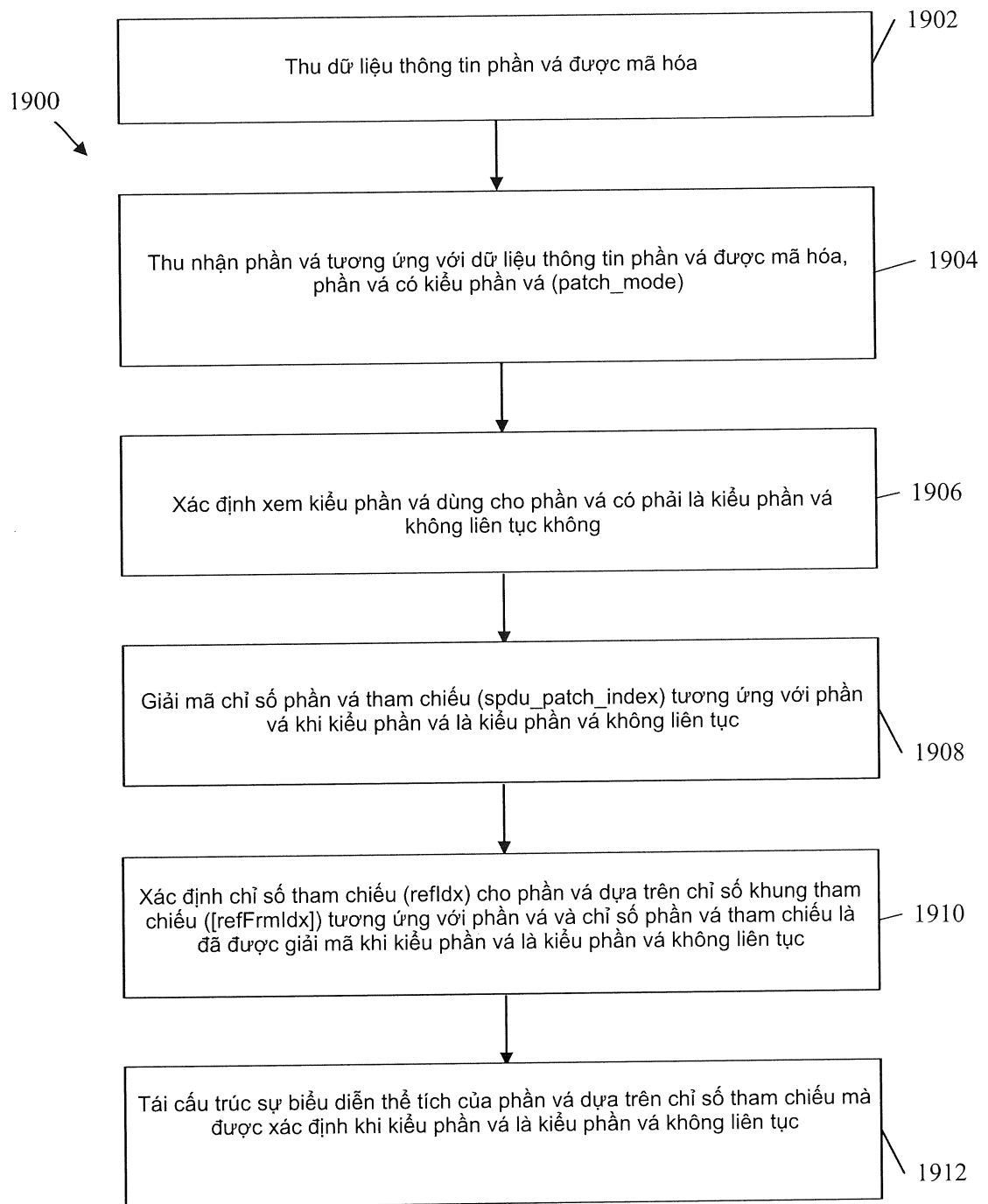


FIG. 19

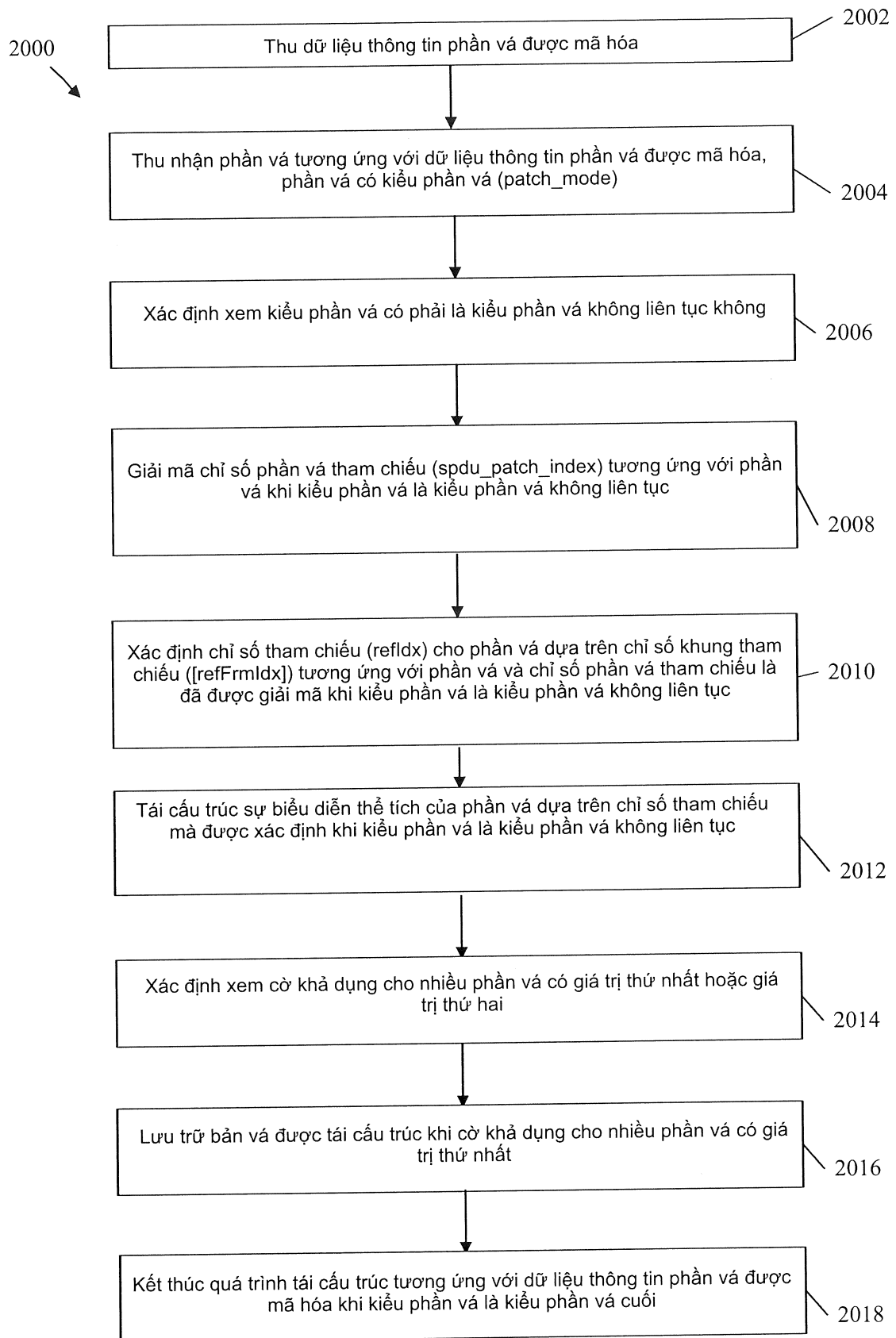


FIG. 20

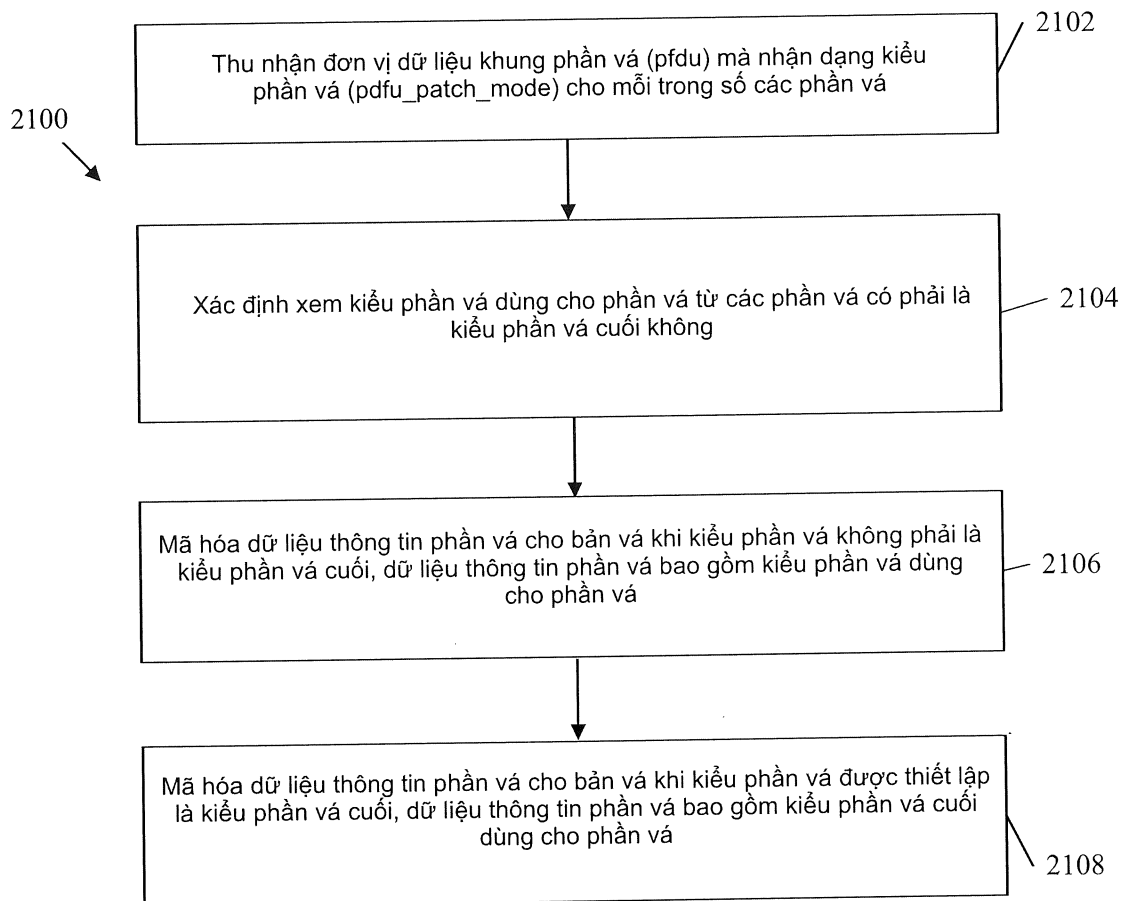


FIG. 21

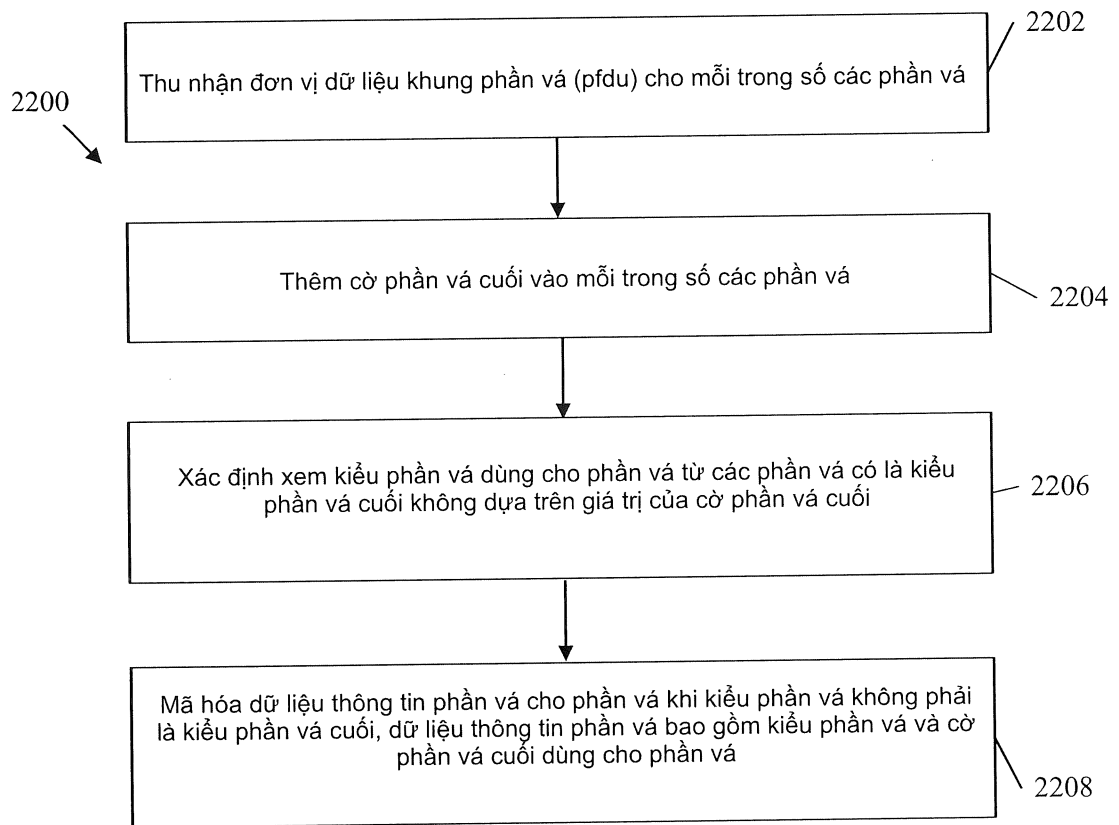


FIG. 22

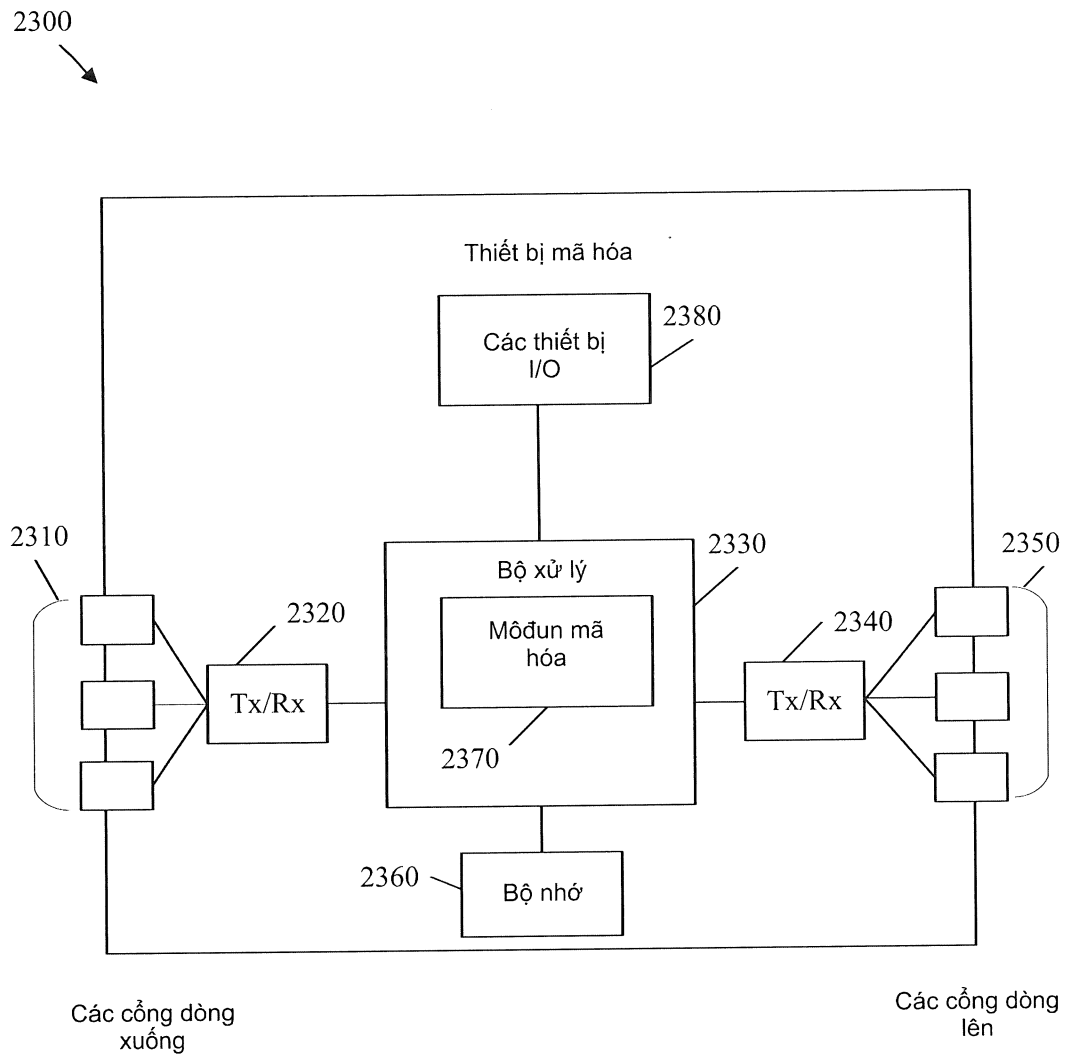


FIG. 23

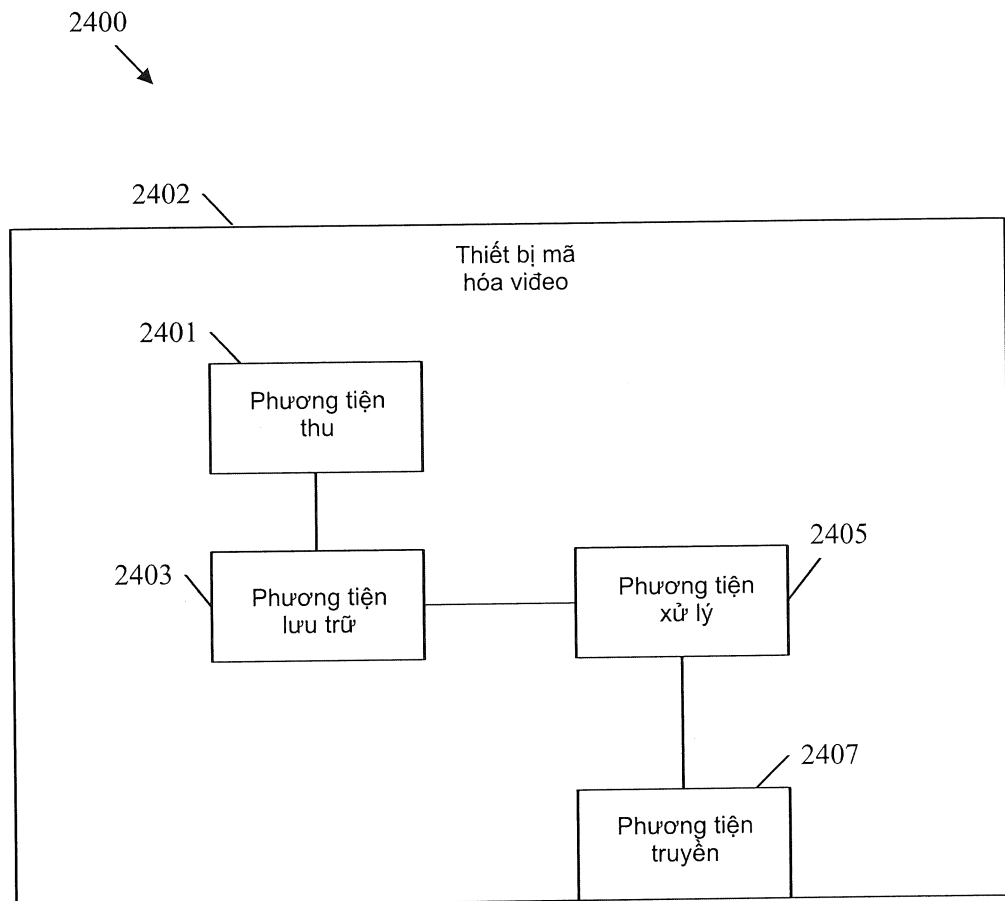


FIG. 24