



(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)



1-0053969

(51)^{2020.01} H04N 19/60; H04N 19/124; H04N
19/132; H04N 19/70; H04N 19/186;
H04N 19/122; H04N 19/137

(13) B

(21) 1-2022-02587

(22) 05/10/2020

(86) PCT/KR2020/013495 05/10/2020

(87) WO2021/066618 08/04/2021

(30) 62/911,221 05/10/2019 US

(45) 25/11/2025 452

(43) 25/07/2022 412A

(73) LG ELECTRONICS INC. (KR)

128, Yeoui-daero, Yeongdeungpo-gu Seoul 07336, Korea

(72) YOO, Sunmi (KR); CHOI, Jungah (KR); CHOI, Jangwon (KR).

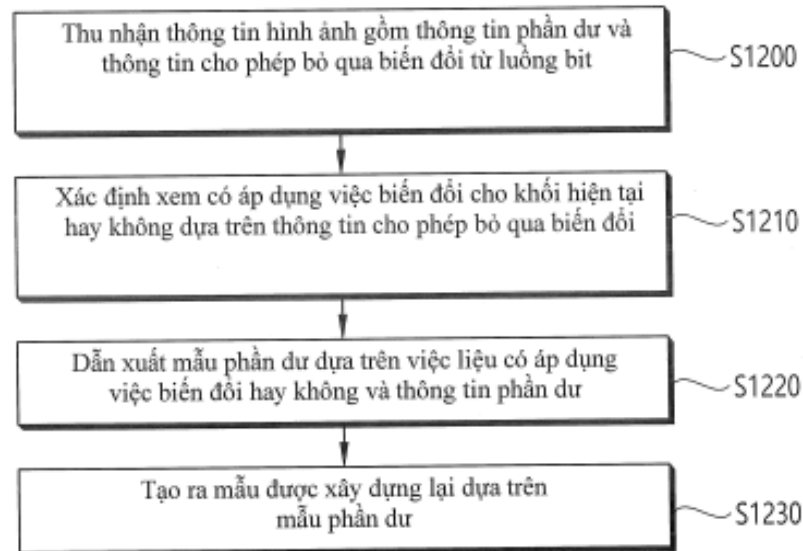
(74) Công ty Luật TNHH T&G (TGVN)

(54) PHƯƠNG PHÁP GIẢI MÃ HÌNH ẢNH, PHƯƠNG PHÁP MÃ HÓA HÌNH ẢNH,
PHƯƠNG TIỆN LƯU TRỮ ĐỌC ĐƯỢC BỞI MÁY TÍNH KHÔNG CHUYỂN
TIẾP VÀ PHƯƠNG PHÁP TRUYỀN DỮ LIỆU CHO HÌNH ẢNH

(21) 1-2022-02587

(57) Sáng chế đề cập đến phương pháp giải mã hình ảnh, phương pháp mã hóa hình ảnh, phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp và phương pháp truyền dữ liệu cho hình ảnh. Theo phần bộc lộ của sáng chế, thông tin tính khả dụng bỏ qua biến đổi và thông tin tính khả dụng bảng màu được báo hiệu thông qua tập thông số trình tự (sequence parameter set, SPS), và thông tin liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được phân tích cú pháp/được báo hiệu một cách hiệu quả trên cơ sở của ít nhất một trong số thông tin tính khả dụng bỏ qua biến đổi và thông tin tính khả dụng bảng màu. Theo đó, các bit mà cần được báo hiệu cho việc lập mã video/hình ảnh có thể được giảm, và hiệu quả lập mã có thể được cải thiện.

FIG. 12



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến việc lập mã hình ảnh hoặc video, và ví dụ, đến công nghệ lập mã hình ảnh hoặc video dựa trên việc báo hiệu của thông tin liên quan đến việc lập mã bảng màu và việc bỏ qua biến đổi.

Tình trạng kỹ thuật của sáng chế

Hiện tại, nhu cầu về hình ảnh/video chất lượng cao, độ phân giải cao chẳng hạn như hình ảnh 4K, 8K hoặc độ phân giải cực cao (Ultra High Definition, UHD) khác đang tăng trong các lĩnh vực khác nhau. Khi độ phân giải hoặc chất lượng hình ảnh/video trở nên cao hơn, lượng tương đối nhiều hơn của thông tin hoặc các bit được truyền so với dữ liệu hình ảnh/video thông thường. Do đó, nếu dữ liệu hình ảnh/video được truyền thông qua phương tiện chẳng hạn như đường băng rộng có dây/không dây hiện có hoặc được lưu trữ trong phương tiện lưu trữ sẵn có, các chi phí cho việc truyền và lưu trữ tăng đáng kể.

Hơn thế nữa, sự quan tâm và nhu cầu đang tăng đối với các nội dung thực tế ảo (virtual reality, VR) và thực tế nhân tạo (artificial reality, AR), và phương tiện nhập vai chẳng hạn như ảnh ba chiều; và việc phát rộng của các hình ảnh/các video thể hiện các đặc tính hình ảnh/video khác với các đặc tính của hình ảnh/video thực, chẳng hạn như các hình ảnh/các video trò chơi, cũng đang tăng.

Do đó, kỹ thuật nén hình ảnh/video hiệu quả cao được yêu cầu phải nén và truyền, lưu trữ, hoặc chơi một cách hiệu quả các hình ảnh/các video chất lượng cao, độ phân giải cao thể hiện các đặc tính khác nhau như được mô tả ở trên.

Ngoài ra, có đòi hỏi về sơ đồ cải thiện hiệu quả lập mã hình ảnh/video tổng thể

bằng việc phân loại một cách hiệu quả xem có lập mã thông tin có liên quan theo sự phụ thuộc và sự không phụ thuộc của thông tin mà được sử dụng hoàn toàn cần thiết hoặc bổ sung trong việc thực hiện việc bỏ qua biến đổi và việc lập mã bảng màu hay không.

Bản chất kỹ thuật của sáng chế

Mục đích kỹ thuật của sáng chế là nhằm đề xuất phương pháp và thiết bị để nâng cao hiệu quả lập mã video/hình ảnh.

Mục đích kỹ thuật khác của sáng chế là nhằm đề xuất phương pháp và thiết bị để phân tích cú pháp/báo hiệu một cách hiệu quả thông tin liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu.

Mục đích kỹ thuật khác nữa của sáng chế là nhằm đề xuất phương pháp và thiết bị để xác định một cách hiệu quả xem có thực hiện việc lập mã theo sự phụ thuộc và/hoặc sự không phụ thuộc của thông tin được sử dụng trong suốt việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu hay không.

Mục đích kỹ thuật khác nữa của sáng chế là nhằm đề xuất phương pháp và thiết bị để xác định điều kiện phụ thuộc để phân tích cú pháp một cách hiệu quả phần tử cú pháp có sự phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu, và xác định xem có thực hiện việc phân tích cú pháp dựa trên điều kiện phụ thuộc hay không.

Theo một phương án của sáng chế, thông tin cho phép bỏ qua biến đổi và thông tin cho phép công có thể được báo hiệu thông qua tập thông số trình tự (sequence parameter set, SPS), và việc liệu có phân tích cú pháp/báo hiệu thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi hay không có thể được xác định dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi và thông tin được cho phép bảng màu. Ví dụ, thông tin thông số lượng tử hóa tối thiểu có thể được phân tích cú pháp/được báo hiệu thông qua SPS dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi là 1 hoặc giá trị

của thông tin cho phép lập mã bảng màu là 1.

Theo một phương án của sáng chế, phương pháp giải mã video/hình ảnh được thực hiện bởi thiết bị giải mã được đề xuất. Phương pháp giải mã video/hình ảnh có thể gồm phương pháp được bộc lộ trong các phương án của sáng chế.

Theo một phương án của sáng chế, thiết bị giải mã thực hiện việc giải mã video/hình ảnh được đề xuất. Thiết bị giải mã có thể thực hiện phương pháp được bộc lộ trong các phương án của sáng chế.

Theo một phương án của sáng chế, phương pháp mã hóa video/hình ảnh được thực hiện bởi thiết bị mã hóa được đề xuất. Phương pháp mã hóa video/hình ảnh có thể gồm phương pháp được bộc lộ trong các phương án của sáng chế.

Theo một phương án của sáng chế, thiết bị mã hóa thực hiện việc mã hóa video/hình ảnh được đề xuất. Thiết bị mã hóa có thể thực hiện phương pháp được bộc lộ trong các phương án của sáng chế.

Theo một phương án của sáng chế, phương tiện lưu trữ kỹ thuật số đọc được bởi máy tính lưu trữ thông tin video/hình ảnh được mã hóa được tạo ra theo phương pháp mã hóa video/hình ảnh được bộc lộ trong ít nhất một trong số các phương án của sáng chế được đề xuất.

Theo một phương án của sáng chế, phương tiện lưu trữ kỹ thuật số đọc được bởi máy tính lưu trữ thông tin được mã hóa hoặc thông tin video/hình ảnh được mã hóa khiến thiết bị giải mã thực hiện phương pháp giải mã video/hình ảnh được bộc lộ trong ít nhất một trong số các phương án của sáng chế được đề xuất.

Sáng chế có thể có các hiệu quả khác nhau. Ví dụ, theo một phương án của sáng chế, hiệu quả nén hình ảnh/video tổng cộng có thể được tăng cường. Ngoài ra, theo một phương án của sáng chế, thông tin liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được phân tích cú pháp/được báo hiệu một cách hiệu quả. Ngoài ra, theo một phương án của sáng chế, việc liệu có thực hiện việc lập mã hay không có thể

được xác định một cách hiệu quả theo sự phụ thuộc và/hoặc sự không phụ thuộc của thông tin được sử dụng trong suốt việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ngoài ra, theo một phương án của sáng chế, việc lập mã hiệu quả là có thể bằng việc xác định điều kiện phụ thuộc để phân tích cú pháp một cách hiệu quả phần tử cú pháp có sự phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu, và xác định xem có thực hiện việc phân tích cú pháp theo điều kiện phụ thuộc hay không. Ngoài ra, theo một phương án của sáng chế, các bit được truyền có thể được lưu bằng việc xác định xem có thực hiện việc phân tích cú pháp theo điều kiện phụ thuộc hay không đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu.

Các hiệu quả mà có thể thu nhận được thông qua các phương án chi tiết trong sáng chế không bị giới hạn ở các hiệu quả được đề cập ở trên. Ví dụ, có thể có các hiệu quả kỹ thuật khác nhau mà có thể được hiểu hoặc được tạo ra từ sáng chế bởi người có trình độ trung bình trong lĩnh vực kỹ thuật liên quan. Theo đó, các hiệu quả chi tiết của sáng chế không bị giới hạn ở các hiệu quả được mô tả rõ ràng trong sáng chế, và có thể gồm các hiệu quả khác nhau mà có thể được hiểu hoặc được tạo ra từ các dấu hiệu kỹ thuật của sáng chế.

Mô tả vắn tắt các hình vẽ

Fig.1 minh họa sơ lược ví dụ của hệ thống lập mã video/hình ảnh với nó các phương án của sáng chế có thể ứng dụng.

Fig.2 là sơ đồ minh họa sơ lược cấu hình của thiết bị mã hóa video/hình ảnh với nó các phương án của sáng chế có thể ứng dụng.

Fig.3 là sơ đồ giải thích sơ lược cấu hình của thiết bị giải mã video/hình ảnh với nó các phương án của sáng chế có thể ứng dụng.

Fig.4 minh họa ví dụ của phương pháp mã hóa hình ảnh/video sơ lược với nó các phương án của sáng chế có thể ứng dụng.

Fig.5 minh họa ví dụ của phương pháp giải mã hình ảnh/video sơ lược với nó các phương án của sáng chế có thể ứng dụng.

Fig.6 minh họa sơ lược ví dụ của phương pháp mã hóa entropi với nó các phương án của sáng chế có thể ứng dụng, và Fig.7 minh họa sơ lược bộ mã hóa entropi trong thiết bị mã hóa.

Fig.8 minh họa sơ lược ví dụ của phương pháp giải mã entropi với nó các phương án của sáng chế có thể ứng dụng, và Fig.9 minh họa sơ lược bộ giải mã entropi trong thiết bị giải mã.

Fig.10 và Fig.11 minh họa một cách sơ lược phương pháp mã hóa video/hình ảnh và ví dụ của các thành phần liên quan theo (các) phương án của sáng chế.

Fig.12 và Fig.13 minh họa một cách sơ lược phương pháp giải mã video/hình ảnh và ví dụ của các thành phần liên quan theo (các) phương án của sáng chế.

Fig.14 minh họa ví dụ của hệ thống phát luồng nội dung mà cho nó các phương án được bộc lộ trong sáng chế có thể ứng dụng.

Mô tả chi tiết sáng chế

Sáng chế có thể được cải biến theo nhiều cách khác nhau và có thể có các phương án khác nhau, và các phương án cụ thể sẽ được minh họa trên các hình vẽ và được mô tả chi tiết. Tuy nhiên, điều này không nhằm giới hạn sáng chế ở các phương án cụ thể. Các thuật ngữ được sử dụng thông thường trong bản mô tả này được sử dụng để mô tả phương án cụ thể và không được sử dụng để giới hạn bản chất kỹ thuật của sáng chế. Sự diễn đạt số ít bao gồm các sự diễn đạt số nhiều trừ khi được diễn đạt rõ ràng khác đi trong bối cảnh. Thuật ngữ, chẳng hạn như “gồm” hoặc “có” trong bản mô tả này, cần được hiểu là để chỉ sự tồn tại của đặc tính, số lượng, bước, thao tác, phần tử, phần, hoặc sự kết hợp của chúng được mô tả trong bản mô tả và không loại trừ sự tồn tại hoặc khả năng bổ sung một hoặc nhiều đặc tính, các số lượng, các bước, các thao tác, các phần tử, các phần khác hoặc sự kết hợp của chúng.

Trong khi đó, các thành phần trong các hình vẽ được mô tả trong sáng chế được minh họa một cách độc lập để thuận tiện cho việc mô tả liên quan tới các chức năng đặc tính khác nhau. Điều này không có nghĩa rằng mỗi trong số các thành phần được thực hiện như phần cứng tách biệt hoặc phần mềm tách biệt. Ví dụ, ít nhất hai trong số các thành phần có thể được kết hợp để tạo thành thành phần đơn, hoặc thành phần đơn có thể được chia thành nhiều thành phần. Phương án trong đó các thành phần được kết hợp và/hoặc được phân tách cũng được gồm trong phạm vi bảo hộ của sáng chế trừ khi nó trệtch khỏi bản chất của sáng chế.

Theo sáng chế, thuật ngữ “A hoặc B” có thể nghĩa là “chỉ A”, “chỉ B”, hoặc “cả A và B”. Nói cách khác, trong sáng chế, thuật ngữ “A hoặc B” có thể được diễn dịch là để chỉ “A và/hoặc B”. Ví dụ, trong sáng chế, thuật ngữ “A, B hoặc C” có thể nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “kết hợp bất kỳ của A, B và C”.

A gạch chéo “/” hoặc dấu phẩy được sử dụng trong sáng chế có thể nghĩa là “và/hoặc”. Ví dụ, “A/B” có nghĩa là “A và/hoặc B”. Theo đó, “A/B” có nghĩa là “chỉ A”, “chỉ B”, hoặc “cả A và B”. Ví dụ, “A, B, C” có thể nghĩa là “A, B hoặc C”.

Theo sáng chế, “ít nhất một trong số A và B” có thể nghĩa là “chỉ A”, “chỉ B”, hoặc “cả A và B”. Ngoài ra, trong sáng chế, việc diễn đạt “ít nhất một trong số A hoặc B” hoặc “ít nhất một trong số A và/hoặc B” có thể được diễn giải giống như “ít nhất một trong số A và B”.

Ngoài ra, trong sáng chế, “ít nhất một trong số A, B và C” có thể nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “kết hợp bất kỳ của A, B và C”. Ngoài ra, “ít nhất một trong số A, B hoặc C” hoặc “ít nhất một trong số A, B và/hoặc C” có thể nghĩa là “ít nhất một trong số A, B và C”.

Ngoài ra, dấu ngoặc đơn được sử dụng trong sáng chế có thể nghĩa là “ví dụ”. Cụ thể là, trong trường hợp mà “việc dự đoán (việc dự đoán nội)” được diễn đạt, nó có thể được chỉ ra rằng “việc dự đoán nội” được đề xuất như ví dụ của “việc dự đoán”. Nói

cách khác, thuật ngữ “việc dự đoán” trong sáng chế không bị giới hạn ở “việc dự đoán nội”, và nó có thể được chỉ ra rằng “việc dự đoán nội” được đề xuất như ví dụ của “việc dự đoán”. Ngoài ra, ngay cả trong trường hợp mà “việc dự đoán (nghĩa là, việc dự đoán nội)” được diễn đạt, nó có thể được chỉ ra rằng “việc dự đoán nội” được đề xuất như ví dụ của “việc dự đoán”.

Sáng chế liên quan đến việc lập mã video/hình ảnh. Ví dụ, các phương pháp/các phương án được bộc lộ trong sáng chế có thể được áp dụng cho phương pháp được bộc lộ trong tiêu chuẩn lập mã video vạn năng (versatile video coding, VVC). Ngoài ra, các phương pháp/các phương án được bộc lộ trong sáng chế có thể được áp dụng cho phương pháp được bộc lộ trong tiêu chuẩn lập mã video thiết yếu (essential video coding, EVC), tiêu chuẩn AOMedia video 1 (AOMedia Video 1, AV1), thế hệ thứ hai của tiêu chuẩn lập mã video audio (2nd generation of audio video coding standard, AVS2), hoặc tiêu chuẩn lập mã video/hình ảnh thế hệ tiếp theo (ví dụ H.267 hoặc H.268, v.v.).

Sáng chế đề xuất các phương án khác nhau của việc lập mã video/hình ảnh, và các phương án có thể được thực hiện kết hợp với nhau trừ khi được đề cập khác đi.

Trong sáng chế, video có thể nghĩa là tập các chuỗi hình ảnh theo dòng thời gian. Ảnh có nghĩa chung là đơn vị biểu diễn một hình ảnh trong khoảng thời gian cụ thể, và lát/phiến là đơn vị cấu thành một phần của ảnh trong việc lập mã. Lát/phiến có thể gồm một hoặc nhiều đơn vị cây lập mã (coding tree unit, CTU). Một ảnh có thể gồm một hoặc nhiều lát/phiến. Phiến là vùng hình chữ nhật của các CTU nằm trong cột phiến cụ thể và hàng phiến cụ thể trong ảnh. Cột phiến là vùng hình chữ nhật của các CTU có chiều cao bằng với chiều cao của ảnh và chiều rộng được chỉ rõ bởi các phân tử cú pháp trong tập thông số ảnh. Hàng phiến là vùng hình chữ nhật của các CTU có chiều rộng được chỉ rõ bởi các phân tử cú pháp trong tập thông số ảnh và chiều cao bằng với chiều cao của ảnh. Việc quét phiến là việc sắp thứ tự theo trình tự cụ thể của các CTU phân vùng ảnh mà trong đó các CTU được sắp thứ tự liên tiếp trong việc quét mảnh CTU trong phiến, trong khi các phiến trong ảnh được sắp thứ tự liên tiếp trong việc quét mảnh

của các phiên của ảnh. Lát gồm số nguyên lần của các phiên hoàn chỉnh hoặc số nguyên lần của các hàng CTU hoàn chỉnh liên tiếp trong phiên của ảnh mà có thể được chứa theo cách dành riêng trong đơn vị NAL đơn.

Trong lúc đó, một ảnh có thể được phân chia thành hai hoặc nhiều ảnh con. Ảnh con có thể là vùng hình chữ nhật của một hoặc nhiều lát trong ảnh.

Điểm ảnh (pixel) hoặc pel có thể có nghĩa là đơn vị nhỏ nhất cấu thành một ảnh (hoặc hình ảnh). Ngoài ra, ‘mẫu’ có thể được sử dụng như là thuật ngữ tương ứng với điểm ảnh. Mẫu có thể biểu diễn chung cho điểm ảnh hoặc giá trị của điểm ảnh, và có thể chỉ biểu diễn điểm ảnh/giá trị điểm ảnh của thành phần độ chói hoặc chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ. Mặt khác, mẫu có thể nghĩa là giá trị điểm ảnh trong miền không gian, hoặc có thể nghĩa là hệ số biến đổi trong miền tần số khi giá trị điểm ảnh được biến đổi vào miền tần số.

Đơn vị có thể biểu diễn đơn vị cơ bản của việc xử lý hình ảnh. Đơn vị có thể gồm ít nhất một thành phần trong số vùng cụ thể của ảnh và thông tin liên quan đến vùng. Một đơn vị có thể gồm một khối độ chói và hai khối sắc độ (ví dụ cb, cr). Đơn vị có thể được sử dụng theo kiểu thay thế cho nhau theo nghĩa như là khối hoặc khu vực trong một số trường hợp. Trong trường hợp chung, các khối $M \times N$ có thể gồm các mẫu (hoặc các mảng mẫu) hoặc tập hợp (hoặc các mảng) của các hệ số biến đổi của M cột và N hàng.

Ngoài ra, trong sáng chế, ít nhất một trong số việc lượng tử hóa/việc khử lượng tử hóa và/hoặc việc biến đổi/việc biến đổi ngược có thể được bỏ qua. Khi việc lượng tử hóa/việc khử lượng tử hóa được lược bỏ, thì hệ số biến đổi được lượng tử hóa có thể được gọi là hệ số biến đổi. Khi việc biến đổi/việc biến đổi ngược được bỏ qua, các hệ số biến đổi có thể được gọi là các hệ số hoặc các hệ số phần dư, hoặc có thể vẫn được gọi là các hệ số biến đổi để trình bày đồng nhất.

Trong sáng chế, hệ số biến đổi được lượng tử hóa và hệ số biến đổi có thể được

gọi là hệ số biến đổi và hệ số biến đổi được định cỡ, một cách tương ứng. Trong trường hợp này, thông tin phần dư có thể gồm thông tin về (các) hệ số biến đổi, và thông tin về (các) hệ số biến đổi có thể được báo hiệu thông qua cú pháp lập mã phần dư. Các hệ số biến đổi có thể được dẫn xuất dựa trên thông tin phần dư (hoặc thông tin về (các) hệ số biến đổi), và các hệ số biến đổi được định cỡ có thể được dẫn xuất thông qua việc biến đổi ngược (định cỡ) trên các hệ số biến đổi. Các mẫu phần dư có thể được dẫn xuất dựa trên việc biến đổi ngược (biến đổi) cho các hệ số biến đổi được định cỡ. Điều này cũng có thể được áp dụng/được trình bày trong các phần khác của sáng chế.

Theo sáng chế, các dấu hiệu kỹ thuật được mô tả riêng lẻ trong một hình vẽ có thể được thực hiện một cách riêng lẻ, hoặc có thể được thực hiện đồng thời.

Sau đây, các phương án ưu tiên của sáng chế được mô tả cụ thể hơn dựa vào các hình vẽ kèm theo. Sau đây, trong các hình vẽ, cùng số tham chiếu được sử dụng trong cùng thành phần, và phần mô tả thừa của cùng thành phần có thể được bỏ qua.

Fig.1 minh họa ví dụ của hệ thống lập mã video/hình ảnh mà cho nó các phương án của sáng chế có thể được áp dụng.

Tham khảo đến Fig.1, hệ thống lập mã video/hình ảnh có thể gồm thiết bị nguồn và thiết bị nhận. Thiết bị nguồn có thể truyền thông tin video/hình ảnh đã được lập mã hoặc dữ liệu đến thiết bị nhận thông qua phương tiện lưu trữ kỹ thuật số hoặc mạng trong dạng tệp hoặc phát luồng.

Thiết bị nguồn có thể gồm nguồn video, thiết bị mã hóa, và bộ truyền. Thiết bị nhận có thể gồm bộ nhận, thiết bị giải mã, và bộ kết xuất. Thiết bị mã hóa có thể được gọi là thiết bị mã hóa video/hình ảnh, và thiết bị giải mã có thể được gọi là thiết bị giải mã video/hình ảnh. Bộ truyền có thể được gồm trong thiết bị mã hóa. Bộ nhận có thể được gồm trong thiết bị giải mã. Bộ kết xuất có thể gồm bộ phận hiển thị, bộ phận hiển thị có thể được tạo cấu hình là thiết bị riêng biệt hoặc thành phần ngoài.

Nguồn video có thể giành được video/hình ảnh thông qua quy trình chụp, tổng

hợp, hoặc tạo ra video/hình ảnh. Nguồn video có thể gồm thiết bị chụp video/hình ảnh và/hoặc thiết bị tạo ra video/hình ảnh. Thiết bị chụp video/hình ảnh có thể gồm, ví dụ, một hoặc nhiều camera, kho lưu trữ video/hình ảnh gồm các video/hình ảnh đã được chụp trước đó, và tương tự. Thiết bị tạo ra video/hình ảnh có thể gồm, ví dụ, máy tính, máy tính bảng, và điện thoại thông minh, và có thể tạo ra các video/hình ảnh (theo cách điện). Ví dụ, video/hình ảnh ảo có thể được tạo ra thông qua máy tính hoặc tương tự. Trong trường hợp này, quy trình chụp video/hình ảnh có thể được thay thế bởi quy trình tạo ra dữ liệu liên quan.

Thiết bị mã hóa có thể mã hóa video/hình ảnh đầu vào. Thiết bị mã hóa có thể thực hiện chuỗi các thủ tục chẳng hạn như dự đoán, biến đổi, và lượng tử hóa cho hiệu quả nén và lập mã. Dữ liệu được mã hóa (thông tin video/hình ảnh được mã hóa) có thể được xuất ra trong dạng luồng bit.

Bộ truyền có thể truyền hình ảnh/thông tin hình ảnh được mã hóa hoặc dữ liệu được xuất trong dạng luồng bit đến bộ nhận của thiết bị nhận thông qua phương tiện lưu trữ kỹ thuật số hoặc mạng trong dạng tệp hoặc phát luồng. Phương tiện lưu trữ kỹ thuật số có thể gồm các phương tiện lưu trữ khác nhau như là USB, SD, CD, DVD, Blu-ray, HDD, SSD, và tương tự. Bộ truyền có thể gồm phần tử để tạo ra tệp phương tiện thông qua định dạng tệp được xác định trước và có thể gồm phần tử để truyền thông qua mạng phát rộng/truyền thông. Bộ nhận có thể nhận/trích xuất luồng bit và truyền luồng bit nhận được đến thiết bị giải mã.

Thiết bị giải mã có thể giải mã video/hình ảnh bằng cách thực hiện một loạt các thủ tục như là khử lượng tử hóa, biến đổi ngược, và dự đoán tương ứng với hoạt động của thiết bị mã hóa.

Bộ kết xuất có thể kết xuất video/hình ảnh được giải mã. Video/hình ảnh được kết xuất có thể được hiển thị qua bộ phận hiển thị.

Fig.2 là sơ đồ minh họa sơ lược cấu hình của thiết bị mã hóa video/hình ảnh mà

cho nó các phương án của sáng chế có thể được áp dụng. Sau đây, phần được gọi là thiết bị mã hóa có thể gồm thiết bị mã hóa hình ảnh và/hoặc thiết bị mã hóa video.

Tham khảo đến Fig.2, thiết bị mã hóa 200 có thể gồm và được tạo cấu hình với bộ phân vùng hình ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, bộ mã hóa entropi 240, bộ cộng 250, bộ lọc 260, và bộ nhớ 270. Bộ dự đoán 220 có thể gồm bộ dự đoán liên 221 và bộ dự đoán nội 222. Bộ xử lý phần dư 230 có thể gồm bộ biến đổi 232, bộ lượng tử hóa 233, bộ khử lượng tử hóa 234, và bộ biến đổi ngược 235. Bộ xử lý phần dư 230 có thể còn gồm bộ trừ 231. Bộ cộng 250 có thể được gọi là bộ xây dựng lại hoặc bộ tạo khối được xây dựng lại. Bộ phân vùng hình ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, bộ mã hóa entropi 240, bộ cộng 250, và bộ lọc 260, mà đã được mô tả ở trên, có thể được tạo cấu hình bởi một hoặc nhiều thành phần phần cứng (ví dụ, các bộ chip bộ mã hóa hoặc các bộ xử lý) theo một phương án. Ngoài ra, bộ nhớ 270 có thể gồm bộ đệm ảnh được giải mã (decoded picture buffer, DPB), và cũng có thể được tạo cấu hình bởi phương tiện lưu trữ kỹ thuật số. Thành phần phần cứng có thể còn gồm bộ nhớ 270 như là thành phần trong/ngoài.

Bộ phân vùng hình ảnh 210 có thể chia hình ảnh đầu vào (hoặc, ảnh, khung) được nhập vào thiết bị mã hóa 200 thành một hoặc nhiều đơn vị xử lý. Như một ví dụ, đơn vị xử lý có thể được gọi là đơn vị lập mã (coding unit, CU). Trong trường hợp này, đơn vị lập mã có thể được chia theo cách đệ quy theo cấu trúc cây tứ phân cây nhị phân cây tam phân (Quad-tree binary-tree ternary-tree, QTBT TT) từ đơn vị cây lập mã (coding tree unit, CTU) hoặc đơn vị lập mã lớn nhất (largest coding unit, LCU). Ví dụ, một đơn vị lập mã có thể được chia thành nhiều đơn vị lập mã có độ sâu sâu hơn dựa trên cấu trúc cây tứ phân, cấu trúc cây nhị phân, và/hoặc cấu trúc cây tam phân. Trong trường hợp này, ví dụ, cấu trúc cây tứ phân được áp dụng đầu tiên và cấu trúc cây nhị phân và/hoặc cấu trúc cây tam phân có thể được áp dụng sau đó. Mặt khác, cấu trúc cây nhị phân cũng có thể được áp dụng trước tiên. Thủ tục lập mã theo sáng chế có thể được thực hiện dựa trên đơn vị lập mã cuối cùng mà không được chia tiếp nữa. Trong trường

hợp này, dựa trên hiệu quả lập mã theo các đặc tính hình ảnh hoặc tương tự, đơn vị lập mã tối đa có thể được sử dụng trực tiếp như đơn vị lập mã cuối cùng, hoặc nếu cần thiết, đơn vị lập mã có thể được chia theo cách đệ quy thành các đơn vị lập mã có độ sâu sâu hơn, sao cho đơn vị lập mã có kích thước tối ưu có thể được sử dụng làm đơn vị lập mã cuối cùng. Ở đây, thủ tục lập mã có thể gồm thủ tục chẳng hạn như dự đoán, biến đổi, và xây dựng lại sẽ được mô tả sau. Theo một ví dụ khác, đơn vị xử lý có thể còn gồm đơn vị dự đoán (prediction unit, PU) hoặc đơn vị biến đổi (transform unit, TU). Trong trường hợp này, mỗi trong số đơn vị dự đoán và đơn vị biến đổi có thể được chia tách hoặc được phân vùng từ đơn vị lập mã cuối cùng được đề cập ở trên. Đơn vị dự đoán có thể là đơn vị của việc dự đoán mẫu, và đơn vị biến đổi có thể là đơn vị để tạo ra hệ số biến đổi và/hoặc đơn vị để tạo ra tín hiệu phần dư từ hệ số biến đổi.

Đơn vị có thể được sử dụng thay thế lẫn nhau với thuật ngữ chẳng hạn như khối hoặc khu vực trong một số trường hợp. Thông thường, khối $M \times N$ có thể biểu diễn các mẫu được cấu thành từ M cột và N hàng hoặc nhóm các hệ số biến đổi. Mẫu có thể biểu diễn chung điểm ảnh hoặc giá trị của điểm ảnh, và có thể cũng biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần độ chói, và cũng biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ. Mẫu có thể được sử dụng làm thuật ngữ tương ứng với điểm ảnh hoặc pel cấu hình một ảnh (hoặc hình ảnh).

Thiết bị mã hóa 200 có thể tạo ra tín hiệu phần dư (khối phần dư, mảng mẫu phần dư) bằng việc trừ tín hiệu được dự đoán (khối được dự đoán, mảng mẫu dự đoán) được xuất từ bộ dự đoán liên 221 hoặc bộ dự đoán nội 222 từ tín hiệu hình ảnh đầu vào (khối gốc, mảng mẫu gốc), và tín hiệu phần dư được tạo ra được truyền tới bộ biến đổi 232. Trong trường hợp này, như được minh họa, đơn vị để trừ tín hiệu được dự đoán (khối được dự đoán, mảng mẫu dự đoán) từ tín hiệu hình ảnh đầu vào (khối gốc, mảng mẫu gốc) trong bộ mã hóa 200 có thể được gọi là bộ trừ 231. Bộ dự đoán có thể thực hiện việc dự đoán cho khối cần được xử lý (sau đây, được gọi là khối hiện tại), và tạo ra khối được dự đoán gồm các mẫu dự đoán của khối hiện tại. Bộ dự đoán có thể xác định

xem việc dự đoán nội được áp dụng hay việc dự đoán liên được áp dụng trong các đơn vị của khối hiện tại hoặc CU. Bộ dự đoán có thể tạo ra thông tin khác nhau về việc dự đoán, chẳng hạn như thông tin chế độ dự đoán, để chuyển thông tin được tạo ra tới bộ mã hóa entropi 240 như được mô tả sau đây trong phần mô tả của mỗi chế độ dự đoán. Thông tin về việc dự đoán có thể được mã hóa bởi bộ mã hóa entropi 240 để được xuất ra dưới dạng luồng bit.

Bộ dự đoán nội 222 có thể dự đoán khối hiện tại với tham chiếu tới các mẫu trong ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt lân cận với khối hiện tại, hoặc cũng có thể được đặt xa khỏi khối hiện tại theo chế độ dự đoán. Các chế độ dự đoán trong việc dự đoán nội có thể gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Chế độ không có hướng có thể gồm, ví dụ, chế độ DC hoặc chế độ phẳng. Chế độ có hướng có thể gồm, ví dụ, 33 chế độ dự đoán có hướng hoặc 65 chế độ dự đoán có hướng theo độ chi tiết của hướng dự đoán. Tuy nhiên, đây là minh họa và các chế độ dự đoán có hướng mà lớn hơn hoặc nhỏ hơn số lượng nêu trên có thể được sử dụng theo việc thiết đặt. Bộ dự đoán nội 222 cũng có thể xác định chế độ dự đoán được áp dụng cho khối hiện tại sử dụng chế độ dự đoán được áp dụng cho khối lân cận.

Bộ dự đoán liên 221 có thể tạo ra khối được dự đoán của khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bởi vector chuyển động trên ảnh tham chiếu. Tại thời điểm này, để làm giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên, thông tin chuyển động có thể được dự đoán trong các đơn vị của khối, khối con, hoặc mẫu dựa trên sự tương liên của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể gồm vector chuyển động và chỉ số ảnh tham chiếu. Thông tin chuyển động có thể còn gồm thông tin hướng dự đoán liên (việc dự đoán L0, việc dự đoán L1, việc dự đoán Bi, hoặc tương tự). Trong trường hợp việc dự đoán liên, khối lân cận có thể gồm khối lân cận theo không gian có mặt trong ảnh hiện tại và khối lân cận theo thời gian có mặt trong ảnh tham chiếu. Ảnh tham chiếu gồm khối tham chiếu và ảnh tham chiếu gồm khối lân cận theo thời gian cũng có

thể là giống với nhau, và cũng có thể là khác nhau. Khối lân cận theo thời gian có thể được gọi tên chẳng hạn như khối tham chiếu được đặt cùng một chỗ, CU được đặt cùng một chỗ (collocated CU, colCU), hoặc tương tự, và ảnh tham chiếu gồm khối lân cận theo thời gian cũng có thể được gọi là ảnh được đặt cùng một chỗ (collocated picture, colPic). Ví dụ, bộ dự đoán liên 221 có thể tạo cấu hình danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận, và tạo ra thông tin chỉ ra ứng viên nào được sử dụng để dẫn xuất vector chuyển động và/hoặc chỉ số ảnh tham chiếu của khối hiện tại. Việc dự đoán liên có thể được thực hiện dựa trên các chế độ dự đoán khác nhau, và ví dụ, trong trường hợp chế độ bỏ qua và chế độ hợp nhất, bộ dự đoán liên 221 có thể sử dụng thông tin chuyển động của khối lân cận như thông tin chuyển động của khối hiện tại. Trong trường hợp chế độ bỏ qua, tín hiệu phần dư có thể không được truyền không giống chế độ hợp nhất. Chế độ dự đoán vector chuyển động (motion vector prediction, MVP) có thể chỉ ra vector chuyển động của khối hiện tại bằng việc sử dụng vector chuyển động của khối lân cận làm bộ dự đoán vector chuyển động, và báo hiệu sự chênh lệch vector chuyển động.

Bộ dự đoán 200 có thể tạo ra tín hiệu được dự đoán dựa trên các phương pháp dự đoán khác nhau sẽ được mô tả sau. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc dự đoán nội hoặc việc dự đoán liên để dự đoán một khối, mà cả áp dụng đồng thời việc dự đoán nội và việc dự đoán liên. Việc này có thể được gọi là việc dự đoán nội và liên được kết hợp (CIIP). Ngoài ra, bộ dự đoán có thể dựa trên chế độ dự đoán sao chép khối nội (IBC), hoặc chế độ bảng màu để thực hiện việc dự đoán trên khối. Chế độ dự đoán IBC hoặc chế độ bảng màu có thể được sử dụng cho việc lập mã hình ảnh/video nội dung của trò chơi hoặc tương tự, chẳng hạn như lập mã nội dung màn (SCC). IBC về cơ bản thực hiện việc dự đoán trong khối hiện tại, nhưng nó có thể được thực hiện tương tự với việc dự đoán liên mà trong đó nó dẫn xuất khối tham chiếu trong ảnh hiện tại. Nghĩa là, IBC có thể sử dụng ít nhất một trong số các kỹ thuật dự đoán liên được mô tả trong sáng chế. Chế độ bảng màu có thể được coi như là ví dụ của việc lập mã nội hoặc

việc dự đoán nội. Khi chế độ bảng màu được áp dụng, giá trị mẫu trong ảnh có thể được báo hiệu dựa trên thông tin về chỉ số bảng màu và bảng của bảng màu.

Tín hiệu được dự đoán được tạo ra thông qua bộ dự đoán (gồm bộ dự đoán liên 221 và/hoặc bộ dự đoán nội 222) có thể được sử dụng để tạo ra tín hiệu được xây dựng lại hoặc được sử dụng để tạo ra tín hiệu phần dư. Bộ biến đổi 232 có thể tạo ra các hệ số biến đổi bằng việc áp dụng kỹ thuật biến đổi cho tín hiệu phần dư. Ví dụ, kỹ thuật biến đổi có thể gồm ít nhất một kỹ thuật trong số biến đổi cosin rời rạc (Discrete Cosine Transform, DCT), biến đổi sin rời rạc (Discrete Sine Transform, DST), biến đổi Karhunen–Loève (Karhunen-Loève Transform, KLT), biến đổi dựa trên đồ thị (Graph-Based Transform, GBT), hoặc biến đổi phi tuyến tùy theo điều kiện (Conditionally Non-Linear Transform, CNT). Ở đây, khi thông tin mối quan hệ giữa các điểm ảnh được minh họa dưới dạng đồ thị, GBT có nghĩa là việc biến đổi được thu nhận từ đồ thị. CNT có nghĩa là việc biến đổi mà được thu nhận dựa trên tín hiệu được dự đoán được tạo ra bằng sử dụng tất cả các điểm ảnh được xây dựng lại trước đó. Ngoài ra, quy trình biến đổi cũng có thể được áp dụng cho khối điểm ảnh có cùng kích thước của hình vuông, và cũng có thể được áp dụng cho khối có kích thước thay đổi khác với hình vuông.

Bộ lượng tử hóa 233 có thể lượng tử hóa các hệ số biến đổi để truyền các hệ số biến đổi được lượng tử hóa đến bộ mã hóa entropi 240, và bộ mã hóa entropi 240 có thể mã hóa tín hiệu được lượng tử hóa (thông tin về các hệ số biến đổi được lượng tử hóa) tới tín hiệu được lượng tử hóa được mã hóa tới luồng bit. Thông tin về các hệ số biến đổi được lượng tử hóa có thể được gọi là thông tin phần dư. Bộ lượng tử hóa 233 có thể sắp xếp lại các hệ số biến đổi được lượng tử hóa có dạng khối trong dạng vector một chiều dựa trên thứ tự quét hệ số, và cũng tạo ra thông tin về các hệ số biến đổi được lượng tử hóa dựa trên các hệ số biến đổi được lượng tử hóa của dạng vector một chiều. Bộ mã hóa entropi 240 có thể thực hiện các phương pháp mã hóa khác nhau, ví dụ, chẳng hạn như lập mã Golomb hàm số mũ, lập mã chiều dài biến đổi thích ứng ngữ cảnh (context-adaptive variable length coding, CAVLC), và lập mã số học nhị phân thích ứng

ngữ cảnh (context-adaptive binary arithmetic coding, CABAC). Bộ mã hóa entropi 240 cũng có thể mã hóa thông tin (ví dụ, các giá trị của các phần tử cú pháp và tương tự) cần thiết để xây dựng lại video/hình ảnh ngoài các hệ số biến đổi được lượng tử hóa cùng nhau hoặc tách biệt. Thông tin được mã hóa (ví dụ, thông tin video/hình ảnh được mã hóa) có thể được truyền hoặc được lưu trữ trong các đơn vị của đơn vị lớp trừu tượng mạng (network abstraction layer, NAL) dưới dạng luồng bit. Thông tin video/hình ảnh có thể còn gồm thông tin về các tập thông số khác nhau chẳng hạn như tập thông số thích ứng (APS), tập thông số ảnh (PPS), tập thông số trình tự (SPS), hoặc tập thông số video (VPS). Bên cạnh đó, thông tin video/hình ảnh có thể còn gồm thông tin ràng buộc chung. Thông tin được báo hiệu/được truyền và/hoặc các phần tử cú pháp sẽ được mô tả sau trong bản mô tả này có thể được mã hóa thông qua thủ tục mã hóa được đề cập ở trên và do đó được gồm trong luồng bit. Luồng bit có thể được truyền qua mạng, hoặc được lưu trữ trong phương tiện lưu trữ kỹ thuật số. Ở đây, mạng có thể gồm mạng phát rộng và/hoặc mạng truyền thông, hoặc tương tự, và phương tiện lưu trữ kỹ thuật số có thể gồm phương tiện lưu trữ khác nhau chẳng hạn như USB, SD, CD, DVD, Blue-ray, HDD, và SSD. Bộ truyền (không được minh họa) để truyền tín hiệu được xuất từ bộ mã hóa entropi 240 và/hoặc bộ lưu trữ (không được minh họa) để lưu trữ tín hiệu có thể được tạo cấu hình dưới dạng các thành phần trong/ngoài của thiết bị mã hóa 200, hoặc bộ truyền cũng có thể được gồm trong bộ mã hóa entropi 240.

Các hệ số biến đổi được lượng tử hóa được xuất từ bộ lượng tử hóa 233 có thể được sử dụng để tạo ra tín hiệu được dự đoán. Ví dụ, bộ khử lượng tử hóa 234 và bộ biến đổi ngược 235 áp dụng việc khử lượng tử hóa và việc biến đổi ngược cho các hệ số biến đổi được lượng tử hóa, sao cho tín hiệu phần dư (khối phần dư hoặc các mẫu phần dư) có thể được xây dựng lại. Bộ cộng 250 cộng tín hiệu phần dư được xây dựng lại vào tín hiệu được dự đoán được xuất từ bộ dự đoán liên 221 hoặc bộ dự đoán nội 222, sao cho tín hiệu được xây dựng lại (ảnh được xây dựng lại, khối được xây dựng lại, mảng mẫu được xây dựng lại) có thể được tạo ra. Như trong trường hợp trong đó chế độ bỏ

qua được áp dụng, nếu không có phần dư cho khối cần được xử lý, khối được dự đoán có thể được sử dụng làm khối được xây dựng lại. Bộ cộng 250 có thể được gọi là bộ xây dựng lại hoặc bộ tạo khối được xây dựng lại. Tín hiệu được xây dựng lại được tạo ra có thể được sử dụng cho việc dự đoán nội của khối tiếp theo cần được xử lý trong ảnh hiện tại, và như được mô tả sau đây, cũng được sử dụng cho việc dự đoán liên của ảnh tiếp theo qua việc lọc.

Trong khi đó, việc ánh xạ độ chói với việc định cỡ sắc độ (luma mapping with chroma scaling, LMCS) cũng có thể được áp dụng trong quy trình xây dựng lại và/hoặc mã hóa ảnh.

Bộ lọc 260 có thể áp dụng việc lọc cho tín hiệu được xây dựng lại, nhờ đó cải thiện các chất lượng hình ảnh chủ quan/khách quan. Ví dụ, bộ lọc 260 có thể áp dụng các phương pháp lọc khác nhau cho ảnh được xây dựng lại để tạo ra ảnh được xây dựng lại được cải biến, và lưu trữ ảnh được xây dựng lại được cải biến trong bộ nhớ 270, cụ thể là, DPB của bộ nhớ 270. Các phương pháp lọc khác nhau có thể gồm, ví dụ, lọc khử khối, dịch chuyển thích ứng mẫu, bộ lọc vòng lặp thích ứng, bộ lọc song phương, và tương tự. Bộ lọc 260 có thể tạo ra thông tin liên quan đến việc lọc khác nhau để chuyển thông tin được tạo ra tới bộ mã hóa entropi 240, như được mô tả sau đây trong phần mô tả của mỗi phương pháp lọc. Thông tin liên quan đến việc lọc có thể được mã hóa bởi bộ mã hóa entropi 240 để được xuất ra dưới dạng luồng bit.

Ảnh được xây dựng lại mà sửa được đổi sẽ được truyền đến bộ nhớ 270 có thể được sử dụng làm ảnh tham chiếu trong bộ dự đoán liên 221. Nếu việc dự đoán liên được áp dụng bởi bộ dự đoán liên, thiết bị mã hóa có thể tránh sự không khớp của việc dự đoán giữa thiết bị mã hóa 200 và thiết bị giải mã, và cũng cải thiện hiệu quả lập mã.

DPB của bộ nhớ 270 có thể lưu trữ ảnh được xây dựng lại được cải biến để được sử dụng làm ảnh tham chiếu trong bộ dự đoán liên 221. Bộ nhớ 270 có thể lưu trữ thông tin chuyển động của khối trong đó thông tin chuyển động trong ảnh hiện tại được dẫn xuất (hoặc được mã hóa) và/hoặc thông tin chuyển động của các khối trong ảnh được

xây dựng lại trước đó. Thông tin chuyển động được lưu trữ có thể được truyền đến bộ dự đoán liên 221 để được sử dụng như là thông tin chuyển động của khối lân cận theo không gian hoặc thông tin chuyển động của khối lân cận theo thời gian. Bộ nhớ 270 có thể lưu trữ các mẫu được xây dựng lại của các khối được xây dựng lại trong ảnh hiện tại, và truyền các mẫu được xây dựng lại tới bộ dự đoán nội 222.

Fig.3 là sơ đồ giải thích sơ lược cấu hình của thiết bị giải mã video/hình ảnh mà cho nó sáng chế có thể ứng dụng. Sau đây, phần được gọi là thiết bị giải mã có thể gồm thiết bị giải mã hình ảnh và/hoặc thiết bị giải mã video.

Tham khảo đến Fig.3, thiết bị giải mã 300 có thể gồm và được tạo cấu hình với bộ giải mã entropi 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, bộ lọc 350, và bộ nhớ 360. Bộ dự đoán 330 có thể gồm bộ dự đoán liên 331 và bộ dự đoán nội 332. Bộ xử lý phần dư 320 có thể gồm bộ khử lượng tử hóa 321 và bộ biến đổi ngược 322. Bộ giải mã entropi 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, và bộ lọc 350, mà đã được mô tả ở trên, có thể được tạo cấu hình bởi một hoặc nhiều thành phần phần cứng (ví dụ, các bộ chip bộ giải mã hoặc các bộ xử lý) theo một phương án. Ngoài ra, bộ nhớ 360 có thể gồm bộ đệm ảnh được giải mã (DPB), và có thể được tạo cấu hình bởi phương tiện lưu trữ kỹ thuật số. Thành phần phần cứng có thể còn gồm bộ nhớ 360 như là thành phần trong/ngoài.

Khi luồng bit gồm thông tin video/hình ảnh được nhập, thiết bị giải mã 300 có thể xây dựng lại hình ảnh tương ứng với quy trình trong đó thông tin video/hình ảnh được xử lý trong thiết bị mã hóa được minh họa trên Fig.2. Ví dụ, thiết bị giải mã 300 có thể dẫn xuất các đơn vị/các khối dựa trên thông tin liên quan đến việc phân tách khối thu nhận được từ luồng bit. Thiết bị giải mã 300 có thể thực hiện việc giải mã sử dụng đơn vị xử lý được áp dụng cho thiết bị mã hóa. Do đó, đơn vị xử lý để giải mã có thể là, ví dụ, đơn vị lập mã, và đơn vị lập mã có thể được chia tách theo cấu trúc cây tứ phân, cấu trúc cây nhị phân, và/hoặc cấu trúc cây tam phân từ đơn vị cây lập mã hoặc đơn vị lập mã tối đa. Một hoặc nhiều đơn vị biến đổi có thể được dẫn xuất từ đơn vị lập mã.

Ngoài ra, tín hiệu hình ảnh được xây dựng lại được giải mã và được xuất ra thông qua thiết bị giải mã 300 có thể được tái tạo lại thông qua thiết bị tái tạo.

Thiết bị giải mã 300 có thể nhận tín hiệu được xuất từ thiết bị mã hóa được minh họa trên Fig.2 dưới dạng luồng bit, và tín hiệu nhận được có thể được giải mã thông qua bộ giải mã entropi 310. Ví dụ, bộ giải mã entropi 310 có thể dẫn xuất thông tin (ví dụ, thông tin video/hình ảnh) cần thiết cho việc xây dựng lại hình ảnh (hoặc việc xây dựng lại ảnh) bằng việc phân tích cú pháp luồng bit. Thông tin video/hình ảnh có thể còn gồm thông tin về các tập thông số khác nhau chẳng hạn như tập thông số thích ứng (adaptation parameter set, APS), tập thông số ảnh (picture parameter set, PPS), tập thông số trình tự (sequence parameter set, SPS), và tập thông số video (video parameter set, VPS). Bên cạnh đó, thông tin video/hình ảnh có thể còn gồm thông tin ràng buộc chung. Thiết bị giải mã có thể giải mã ảnh thêm dựa trên thông tin về tập thông số và/hoặc thông tin ràng buộc chung. Thông tin được báo hiệu/được nhận và/hoặc các phần tử cú pháp sẽ được mô tả sau trong bản mô tả này có thể được giải mã thông qua thủ tục giải mã và thu nhận được từ luồng bit. Ví dụ, bộ giải mã entropi 310 có thể giải mã thông tin trong luồng bit dựa trên phương pháp lập mã chẳng hạn như lập mã Golomb hàm số mũ, CAVLC, hoặc CABAC, và xuất ra giá trị của phần tử cú pháp cần thiết cho việc xây dựng lại hình ảnh, và các giá trị được lượng tử hóa của hệ số biến đổi liên quan đến phần dư. Cụ thể hơn là, phương pháp giải mã entropi CABAC có thể nhận ngăn tương ứng với mỗi phần tử cú pháp từ luồng bit, xác định mô hình ngữ cảnh sử dụng thông tin phần tử cú pháp cần được giải mã và không tin giải mã của khối lân cận và khối cần được giải mã hoặc thông tin của ký hiệu/ngăn được giải mã trong giai đoạn trước, và tạo ra ký hiệu tương ứng với giá trị của mỗi phần tử cú pháp bằng việc dự đoán khả năng tạo ra của ngăn theo mô hình ngữ cảnh được xác định để thực hiện việc giải mã số học của ngăn. Tại thời điểm này, phương pháp giải mã entropi CABAC có thể xác định mô hình ngữ cảnh và sau đó cập nhật mô hình ngữ cảnh sử dụng thông tin của ký hiệu/ngăn được giải mã cho mô hình ngữ cảnh của ký hiệu/ngăn tiếp theo. Thông tin về việc dự đoán

trong số thông tin được giải mã bởi bộ giải mã entropi 310 có thể được cung cấp cho bộ dự đoán (bộ dự đoán liên 332 và bộ dự đoán nội 331), và giá trị phần dư tại đó việc giải mã entropi được thực hiện bởi bộ giải mã entropi 310, nghĩa là, các hệ số biến đổi được lượng tử hóa và thông tin thông số liên quan có thể được nhập vào bộ xử lý phần dư 320. Bộ xử lý phần dư 320 có thể dẫn xuất tín hiệu phần dư (khối phần dư, các mẫu phần dư, mảng mẫu phần dư). Ngoài ra, thông tin về việc lọc trong số thông tin được giải mã bởi bộ giải mã entropi 310 có thể được cung cấp cho bộ lọc 350. Trong khi đó, bộ nhận (không được minh họa) để nhận tín hiệu được xuất từ thiết bị mã hóa có thể còn được tạo cấu hình dưới dạng thành phần trong/ngoài của thiết bị giải mã 300, hoặc bộ nhận có thể cũng là thành phần của bộ giải mã entropi 310. Trong khi đó, thiết bị giải mã theo bản mô tả này có thể được gọi là thiết bị giải mã video/hình ảnh/ảnh, và thiết bị giải mã cũng có thể được phân loại thành bộ giải mã thông tin (bộ giải mã thông tin video/hình ảnh/ảnh) và bộ giải mã mẫu (bộ giải mã mẫu video/hình ảnh/ảnh). Bộ giải mã thông tin có thể gồm bộ giải mã entropi 310, và bộ giải mã mẫu có thể gồm ít nhất một thành phần trong số bộ khử lượng tử hóa 321, bộ biến đổi ngược 322, bộ cộng 340, bộ lọc 350, bộ nhớ 360, bộ dự đoán liên 332, và bộ dự đoán nội 331.

Bộ khử lượng tử hóa 321 có thể khử lượng tử hóa các hệ số biến đổi được lượng tử hóa để xuất ra các hệ số biến đổi. Bộ khử lượng tử hóa 321 có thể sắp xếp lại các hệ số biến đổi được lượng tử hóa trong dạng khối hai chiều. Trong trường hợp này, việc sắp xếp lại có thể được thực hiện dựa trên thứ tự quét hệ số được thực hiện bởi thiết bị mã hóa. Bộ khử lượng tử hóa 321 có thể thực hiện việc khử lượng tử hóa cho các hệ số biến đổi được lượng tử hóa sử dụng thông số lượng tử hóa (ví dụ, thông tin kích thước bước lượng tử hóa), và thu nhận các hệ số biến đổi.

Bộ biến đổi ngược 322 biến đổi ngược các hệ số biến đổi để thu nhận tín hiệu phần dư (khối phần dư, mảng mẫu phần dư).

Bộ dự đoán 330 có thể thực hiện việc dự đoán của khối hiện tại, và tạo ra khối được dự đoán gồm các mẫu dự đoán của khối hiện tại. Bộ dự đoán có thể xác định xem

việc dự đoán nội được áp dụng hay việc dự đoán liên được áp dụng cho khối hiện tại dựa trên thông tin về việc dự đoán được xuất từ bộ giải mã entropi 310, và xác định chế độ dự đoán nội/liên cụ thể.

Bộ dự đoán có thể tạo ra tín hiệu được dự đoán dựa trên các phương pháp dự đoán khác nhau sẽ được mô tả sau. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc dự đoán nội hoặc việc dự đoán liên cho việc dự đoán của một khối, mà cả áp dụng việc dự đoán nội và việc dự đoán liên cùng lúc. Việc này có thể được gọi là việc dự đoán nội và liên được kết hợp (combined inter and intra prediction, CIIP). Ngoài ra, bộ dự đoán có thể dựa trên chế độ dự đoán sao chép khối nội (intra block copy, IBC), hoặc chế độ bảng màu để thực hiện việc dự đoán trên khối. Chế độ dự đoán IBC hoặc chế độ bảng màu có thể được sử dụng cho việc lập mã hình ảnh/video nội dung của trò chơi hoặc tương tự, chẳng hạn như lập mã nội dung màn (screen content coding, SCC). IBC về cơ bản thực hiện việc dự đoán trong khối hiện tại, nhưng nó có thể được thực hiện tương tự với việc dự đoán liên mà trong đó nó dẫn xuất khối tham chiếu trong ảnh hiện tại. Nghĩa là, IBC có thể sử dụng ít nhất một trong số các kỹ thuật dự đoán liên được mô tả trong sáng chế. Chế độ bảng màu có thể được coi như là ví dụ của việc lập mã nội hoặc việc dự đoán nội. Khi chế độ bảng màu được áp dụng, thông tin về bảng của bảng màu và chỉ số bảng màu có thể được gồm trong thông tin video/hình ảnh và được báo hiệu.

Bộ dự đoán nội 331 có thể dự đoán khối hiện tại tham chiếu tới các mẫu trong ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt lân cận với khối hiện tại theo chế độ dự đoán, hoặc cũng có thể được đặt xa khỏi khối hiện tại. Các chế độ dự đoán trong việc dự đoán nội có thể gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Bộ dự đoán nội 331 cũng có thể xác định chế độ dự đoán được áp dụng cho khối hiện tại sử dụng chế độ dự đoán được áp dụng cho khối lân cận.

Bộ dự đoán liên 332 có thể tạo ra khối được dự đoán của khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bởi vectơ chuyển động trên ảnh tham chiếu. Tại thời điểm này, để làm giảm lượng thông tin chuyển động được truyền

trong chế độ dự đoán liên, thông tin chuyển động có thể được dự đoán trong các đơn vị của khối, khối con, hoặc mẫu dựa trên sự tương liên của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể gồm vectơ chuyển động và chỉ số ảnh tham chiếu. Thông tin chuyển động có thể còn gồm thông tin hướng dự đoán liên (việc dự đoán L0, việc dự đoán L1, việc dự đoán Bi, hoặc tương tự). Trong trường hợp việc dự đoán liên, khối lân cận có thể gồm khối lân cận theo không gian có mặt trong ảnh hiện tại và khối lân cận theo thời gian có mặt trong ảnh tham chiếu. Ví dụ, bộ dự đoán liên 332 có thể tạo cấu hình danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận, và dẫn xuất vectơ chuyển động và/hoặc chỉ số ảnh tham chiếu của khối hiện tại dựa trên thông tin lựa chọn ứng viên được nhận. Việc dự đoán liên có thể được thực hiện dựa trên các chế độ dự đoán khác nhau, và thông tin về việc dự đoán có thể gồm thông tin chỉ ra chế độ của việc dự đoán liên của khối hiện tại.

Bộ cộng 340 có thể cộng tín hiệu phần dư được thu nhận với tín hiệu được dự đoán (khối được dự đoán, mảng mẫu dự đoán) được xuất từ bộ dự đoán (gồm bộ dự đoán liên 332 và/hoặc bộ dự đoán nội 331) để tạo ra tín hiệu được xây dựng lại (ảnh được xây dựng lại, khối được xây dựng lại, mảng mẫu được xây dựng lại). Như trong trường hợp trong đó chế độ bỏ qua được áp dụng, nếu không có phần dư cho khối cần được xử lý, khối được dự đoán có thể được sử dụng làm khối được xây dựng lại.

Bộ cộng 340 có thể được gọi là bộ xây dựng lại hoặc bộ tạo khối được xây dựng lại. Tín hiệu được xây dựng lại được tạo ra có thể được sử dụng cho việc dự đoán nội của khối tiếp theo cần được xử lý trong ảnh hiện tại, và như được mô tả sau đây, có thể cũng được xuất ra qua việc lọc hoặc cũng có thể được sử dụng cho việc dự đoán liên của ảnh tiếp theo.

Trong khi đó, việc ánh xạ độ chói với việc định cỡ sắc độ (LMCS) cũng có thể được áp dụng trong quy trình giải mã ảnh.

Bộ lọc 350 có thể áp dụng việc lọc cho tín hiệu được xây dựng lại, nhờ đó cải thiện các chất lượng hình ảnh chủ quan/khách quan. Ví dụ, bộ lọc 350 có thể áp dụng

các phương pháp lọc khác nhau cho ảnh được xây dựng lại để tạo ra ảnh được xây dựng lại được cải biến, và truyền ảnh được xây dựng lại được cải biến tới bộ nhớ 360, cụ thể là, DPB của bộ nhớ 360. Các phương pháp lọc khác nhau có thể gồm, ví dụ, lọc khử khối, dịch chuyển thích ứng mẫu, bộ lọc vòng lặp thích ứng, bộ lọc song phương, và tương tự.

Ảnh được xây dựng lại (được cải biến) được lưu trữ trong DPB của bộ nhớ 360 có thể được sử dụng làm ảnh tham chiếu trong bộ dự đoán liên 332. Bộ nhớ 360 có thể lưu trữ thông tin chuyển động của khối trong đó thông tin chuyển động trong ảnh hiện tại được dẫn xuất (được giải mã) và/hoặc thông tin chuyển động của các khối trong ảnh được xây dựng lại trước đó. Thông tin chuyển động được lưu trữ có thể được truyền đến bộ dự đoán liên 260 để được sử dụng như là thông tin chuyển động của khối lân cận theo không gian hoặc thông tin chuyển động của khối lân cận theo thời gian. Bộ nhớ 360 có thể lưu trữ các mẫu được xây dựng lại của các khối được xây dựng lại trong ảnh hiện tại, và truyền các mẫu được xây dựng lại được lưu trữ tới bộ dự đoán nội 331.

Theo sáng chế, các phương án ví dụ được mô tả trong bộ lọc 260, bộ dự đoán liên 221, và bộ dự đoán nội 222 của thiết bị mã hóa 200 có thể được áp dụng đồng đều hoặc tương ứng với bộ lọc 350, bộ dự đoán liên 332, và bộ dự đoán nội 331 của thiết bị giải mã 300, một cách tương ứng.

Trong khi đó, như được mô tả ở trên, trong khi thực hiện lập mã video, việc dự đoán được thực hiện để cải thiện hiệu quả nén. Thông qua việc này, khối được dự đoán gồm các mẫu dự đoán cho khối hiện tại như khối cần được lập mã (tức là, khối mục tiêu lập mã) có thể được tạo ra. Ở đây, khối được dự đoán gồm các mẫu dự đoán trong miền không gian (hoặc miền điểm ảnh). Khối được dự đoán được dẫn xuất theo cách giống như trong thiết bị mã hóa và thiết bị giải mã, và thiết bị mã hóa có thể báo hiệu thông tin (thông tin phân dư) về phân dư giữa khối gốc và khối được dự đoán, thay vì giá trị mẫu gốc của khối gốc, đến thiết bị giải mã, nhờ đó làm tăng hiệu quả lập mã hình ảnh. Thiết bị giải mã có thể dẫn xuất khối phân dư gồm các mẫu phân dư dựa trên thông tin

phần dư, bổ sung khối phần dư và khối được dự đoán để tạo ra các khối được xây dựng lại gồm các mẫu được xây dựng lại, và tạo ra ảnh được xây dựng lại gồm các khối được xây dựng lại.

Thông tin phần dư có thể được tạo ra thông qua thủ tục biến đổi và lượng tử hóa. Ví dụ, thiết bị mã hóa có thể dẫn xuất khối phần dư giữa khối gốc và khối được dự đoán, thực hiện thủ tục biến đổi trên các mẫu phần dư (mảng mẫu phần dư) được gồm trong khối phần dư để dẫn xuất các hệ số biến đổi, thực hiện thủ tục lượng tử hóa trên các hệ số biến đổi để dẫn xuất các hệ số biến đổi được lượng tử hóa, và báo hiệu thông tin phần dư liên quan đến thiết bị giải mã (thông qua luồng bit). Ở đây, thông tin phần dư có thể gồm thông tin giá trị của các hệ số biến đổi được lượng tử hóa, thông tin vị trí, kỹ thuật biến đổi, nhân biến đổi, thông số lượng tử hóa, và tương tự. Thiết bị giải mã có thể thực hiện thủ tục khử lượng tử hóa/biến đổi ngược dựa trên thông tin phần dư và dẫn xuất các mẫu phần dư (hoặc các khối phần dư). Thiết bị giải mã có thể tạo ra ảnh được xây dựng lại dựa trên khối được dự đoán và khối phần dư. Ngoài ra, đối với sự tham chiếu cho việc dự đoán liên của ảnh sau đó, thiết bị mã hóa cũng có thể khử lượng tử hóa/biến đổi ngược các hệ số biến đổi được lượng tử hóa để dẫn xuất khối phần dư và tạo ra ảnh được xây dựng lại dựa trên đó.

Hình vẽ sau đây đã được chuẩn bị để giải thích ví dụ chi tiết của sáng chế. Do tên gọi của thiết bị chi tiết hoặc thuật ngữ chi tiết hoặc tên gọi (ví dụ, tên của cú pháp) được mô tả trong hình vẽ được thể hiện ví dụ, các dấu hiệu kỹ thuật của sáng chế không bị giới hạn ở tên gọi chi tiết được sử dụng trong hình vẽ.

Fig.4 minh họa ví dụ của phương pháp mã hóa hình ảnh/video sơ lược với nó các phương án của sáng chế có thể ứng dụng.

Phương pháp được bộc lộ trên Fig.4 có thể được thực hiện bởi thiết bị mã hóa 200 của Fig.2 như được mô tả ở trên. Cụ thể là, S400 có thể được thực hiện bởi bộ dự đoán liên 221 hoặc bộ dự đoán nội 222 của thiết bị mã hóa 200, và S410, S420, S430, và S440 có thể được thực hiện bởi bộ trừ 231, bộ biến đổi 232, bộ lượng tử hóa 233, và

bộ mã hóa entropi 240 của thiết bị mã hóa 200, một cách tương ứng.

Tham khảo đến Fig.4, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán thông qua việc dự đoán cho khối hiện tại (S400). Thiết bị mã hóa có thể xác định xem thực hiện việc dự đoán liên hay việc dự đoán nội đối với khối hiện tại, và có thể xác định chế độ dự đoán liên chi tiết hoặc chế độ dự đoán nội chi tiết dựa trên chi phí RD. Theo chế độ được xác định, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán cho khối hiện tại.

Thiết bị mã hóa có thể dẫn xuất các mẫu phần dư thông qua việc so sánh của các mẫu dự đoán với các mẫu gốc cho khối hiện tại (S410).

Thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi thông qua quy trình biến đổi cho các mẫu phần dư (S420), và có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng việc lượng tử hóa các hệ số biến đổi được dẫn xuất (S430).

Việc lượng tử hóa có thể được thực hiện dựa trên thông số lượng tử hóa. Quy trình biến đổi và/hoặc quy trình lượng tử hóa có thể được bỏ qua. Trong trường hợp mà quy trình biến đổi được bỏ qua, các hệ số (phần dư) (được lượng tử hóa) cho các mẫu phần dư có thể được lập mã theo kỹ thuật lập mã phần dư sẽ được mô tả sau. Để thống nhất thuật ngữ, thậm chí hệ số (phần dư) (được lượng tử hóa) có thể được gọi là hệ số biến đổi (được lượng tử hóa).

Thiết bị mã hóa có thể mã hóa thông tin hình ảnh gồm thông tin dự đoán và thông tin phần dư, và có thể xuất ra thông tin hình ảnh được mã hóa dưới dạng luồng bit (S440). Thông tin dự đoán có thể là thông tin liên quan tới quy trình dự đoán, và có thể gồm thông tin (ví dụ, trong trường hợp mà việc dự đoán liên được áp dụng) về thông tin chế độ dự đoán và thông tin chuyển động. Thông tin phần dư có thể gồm thông tin về các hệ số biến đổi được lượng tử hóa. Thông tin phần dư có thể được lập mã entropi. Mặt khác, thông tin phần dư có thể gồm thông tin về các hệ số (phần dư) (được lượng tử hóa).

Luồng bit đầu ra có thể được truyền đến thiết bị giải mã thông qua phương tiện

lưu trữ hoặc mạng.

Fig.5 minh họa ví dụ của thủ tục giải mã video/hình ảnh sơ lược với nó các phương án của sáng chế có thể ứng dụng.

Phương pháp được bộc lộ trên Fig.5 có thể được thực hiện bởi thiết bị giải mã 300 của Fig.3 như được mô tả ở trên. Cụ thể là, S500 có thể được thực hiện bởi bộ dự đoán liên 332 hoặc bộ dự đoán nội 331 của thiết bị giải mã 300. Trong S500, phương pháp dẫn xuất các giá trị của các phần tử cú pháp liên quan bằng việc giải mã thông tin dự đoán được gồm trong luồng bit có thể được thực hiện bởi bộ giải mã entropi 310 của thiết bị giải mã 300. S510, S520, S530, và S540 có thể được thực hiện bởi bộ giải mã entropi 310, bộ khử lượng tử hóa 321, bộ biến đổi ngược 322, và bộ cộng 340 của thiết bị giải mã 300, một cách tương ứng.

Tham khảo đến Fig.5, thiết bị giải mã có thể thực hiện hoạt động tương ứng với hoạt động được thực hiện bởi thiết bị mã hóa. Thiết bị giải mã có thể thực hiện việc dự đoán liên hoặc việc dự đoán nội đối với khối hiện tại dựa trên thông tin dự đoán nhận được, và có thể dẫn xuất các mẫu dự đoán (S500).

Thiết bị giải mã có thể dẫn xuất các hệ số biến đổi được lượng tử hóa cho khối hiện tại dựa trên thông tin phần dư nhận được (S510). Thiết bị giải mã có thể dẫn xuất các hệ số biến đổi được lượng tử hóa từ thông tin phần dư thông qua việc giải mã entropi.

Thiết bị giải mã có thể dẫn xuất các hệ số biến đổi bằng việc khử lượng tử hóa các hệ số biến đổi được lượng tử hóa (S520). Việc khử lượng tử hóa có thể được thực hiện dựa trên thông số lượng tử hóa.

Thiết bị giải mã có thể dẫn xuất các mẫu phần dư thông qua quy trình khử lượng tử hóa cho các hệ số biến đổi (S530).

Quy trình biến đổi ngược và/hoặc quy trình khử lượng tử hóa có thể được bỏ qua. Trong trường hợp mà quy trình biến đổi ngược được bỏ qua, các hệ số (phần dư) (được lượng tử hóa) có thể được dẫn xuất từ thông tin phần dư, và các mẫu phần dư có

thể được dẫn xuất dựa trên các hệ số (phần dư) (được lượng tử hóa).

Thiết bị giải mã có thể tạo ra các mẫu được xây dựng lại cho khối hiện tại dựa trên các mẫu dự đoán và các mẫu phần dư, và dựa trên điều này, có thể tạo ra ảnh được xây dựng lại (S540). Sau đó, quy trình lọc trong vòng lặp có thể còn được áp dụng cho ảnh được xây dựng lại như được mô tả ở trên.

Trong khi đó, như được mô tả ở trên, thiết bị mã hóa có thể thực hiện việc mã hóa entropi dựa trên các phương pháp mã hóa khác nhau, ví dụ, chẳng hạn như Golomb hàm số mũ, lập mã chiều dài biến đổi thích ứng ngữ cảnh (CAVLC), và lập mã số học nhị phân thích ứng ngữ cảnh (CABAC). Ngoài ra, thiết bị giải mã có thể thực hiện việc giải mã entropi dựa trên phương pháp lập mã, chẳng hạn như lập mã Golomb hàm số mũ, CAVLC, hoặc CABAC. Sau đây, quy trình mã hóa/giải mã entropi sẽ được mô tả.

Fig.6 minh họa sơ lược ví dụ của phương pháp mã hóa entropi với nó các phương án của sáng chế có thể ứng dụng, và Fig.7 minh họa sơ lược bộ mã hóa entropi trong thiết bị mã hóa. Bộ mã hóa entropi trong thiết bị mã hóa của Fig.7 có thể được áp dụng ngang bằng hoặc tương ứng ngay cả với bộ mã hóa entropi 240 của thiết bị mã hóa 200 của Fig.2 như được mô tả ở trên.

Tham khảo đến Fig.6 và Fig.7, thiết bị mã hóa (bộ mã hóa entropi) có thể thực hiện quy trình lập mã entropi cho thông tin hình ảnh/video. Thông tin hình ảnh/video có thể gồm thông tin liên quan đến việc phân vùng, thông tin liên quan đến việc dự đoán (ví dụ, thông tin phân loại việc dự đoán liên/nội, thông tin chế độ dự đoán nội, thông tin chế độ dự đoán liên, và tương tự), thông tin phần dư, và thông tin liên quan đến việc lọc trong vòng lặp, và cũng có thể gồm các phần tử cú pháp khác nhau của chúng. Việc lập mã entropi có thể được thực hiện trong đơn vị của phần tử cú pháp. S600 đến S610 có thể được thực hiện bởi bộ mã hóa entropi 240 của thiết bị mã hóa 200 của Fig.2 như được mô tả ở trên.

Thiết bị mã hóa có thể thực hiện việc nhị phân hóa cho phần tử cú pháp mục tiêu

(S600). Ở đây, việc nhị phân hóa có thể dựa trên các phương pháp nhị phân hóa khác nhau, chẳng hạn như quy trình nhị phân hóa rice được cắt cụt và quy trình nhị phân hóa độ dài cố định, và phương pháp nhị phân hóa cho phần tử cú pháp mục tiêu có thể được định rõ trước. Quy trình nhị phân hóa có thể được thực hiện bởi bộ nhị phân hóa 242 trong bộ mã hóa entropi 240.

Thiết bị mã hóa có thể thực hiện việc mã hóa entropi cho phần tử cú pháp mục tiêu (S610). Thiết bị mã hóa có thể thực hiện việc mã hóa dựa trên việc lập mã thông thường (dựa trên ngữ cảnh) hoặc dựa trên việc lập mã đi vòng lặp qua của chuỗi ngắn của phần tử cú pháp mục tiêu dựa trên kỹ thuật lập mã entropi, chẳng hạn như lập mã số học thích ứng ngữ cảnh (CABAC) hoặc lập mã chiều dài biến đổi thích ứng ngữ cảnh (CAVLC), và đầu ra của nó có thể được gồm trong luồng bit. Quy trình mã hóa entropi có thể được thực hiện bởi bộ xử lý mã hóa entropi 243 trong bộ mã hóa entropi 240. Luồng bit có thể được truyền đến thiết bị giải mã thông qua phương tiện lưu trữ (kỹ thuật số) hoặc mạng như được mô tả ở trên.

Fig.8 minh họa sơ lược ví dụ của phương pháp giải mã entropi với nó các phương án của sáng chế có thể ứng dụng, và Fig.9 minh họa sơ lược bộ giải mã entropi trong thiết bị giải mã. Bộ giải mã entropi trong thiết bị giải mã của Fig.9 có thể được áp dụng ngang bằng hoặc tương ứng ngay cả với bộ giải mã entropi 310 của thiết bị giải mã 300 của Fig.3 như được mô tả ở trên.

Tham khảo đến Fig.8 và Fig.9, thiết bị giải mã (bộ giải mã entropi) có thể giải mã thông tin hình ảnh/video được mã hóa. Thông tin hình ảnh/video có thể gồm thông tin liên quan đến việc phân vùng, thông tin liên quan đến việc dự đoán (ví dụ, thông tin phân loại việc dự đoán liên/nội, thông tin chế độ dự đoán nội, thông tin chế độ dự đoán liên, và tương tự), thông tin phần dư, và thông tin liên quan đến việc lọc trong vòng lặp, và cũng có thể gồm các phần tử cú pháp khác nhau của chúng. Việc lập mã entropi có thể được thực hiện trong đơn vị của phần tử cú pháp. S800 đến S810 có thể được thực hiện bởi bộ giải mã entropi 310 của thiết bị giải mã 300 của Fig.3 như được mô tả ở trên.

Thiết bị giải mã có thể thực hiện việc nhị phân hóa cho phần tử cú pháp mục tiêu (S800). Ở đây, việc nhị phân hóa có thể dựa trên các phương pháp nhị phân hóa khác nhau, chẳng hạn như quy trình nhị phân hóa rice được cắt cụt và quy trình nhị phân hóa độ dài cố định, và phương pháp nhị phân hóa cho phần tử cú pháp mục tiêu có thể được định rõ trước. Thiết bị giải mã có thể dẫn xuất các chuỗi ngăn được cho phép (các ứng viên chuỗi ngăn) cho các giá trị được cho phép của phần tử cú pháp mục tiêu thông qua quy trình nhị phân hóa. Quy trình nhị phân hóa có thể được thực hiện bởi bộ nhị phân hóa 312 trong bộ giải mã entropi 310.

Thiết bị giải mã có thể thực hiện việc giải mã entropi cho phần tử cú pháp mục tiêu (S810). Thiết bị giải mã so sánh chuỗi ngăn được dẫn xuất với các chuỗi ngăn được cho phép cho phần tử cú pháp tương ứng trong khi tuần tự giải mã và phân tích cú pháp các ngăn tương ứng cho phần tử cú pháp mục tiêu từ (các) bit đầu vào trong luồng bit. Nếu chuỗi ngăn được dẫn xuất là bằng với một trong các chuỗi ngăn được cho phép, giá trị tương ứng với chuỗi ngăn tương ứng có thể được dẫn xuất làm giá trị của phần tử cú pháp tương ứng. Nếu không, quy trình được mô tả ở trên có thể được thực hiện lại sau khi phân tích cú pháp thêm bit tiếp theo trong luồng bit. Thông qua quy trình này, thông tin tương ứng có thể được báo hiệu sử dụng bit độ dài thay đổi ngay cả không sử dụng bit bắt đầu hoặc bit kết thúc cho thông tin cụ thể (phần tử cú pháp cụ thể) trong luồng bit. Thông qua việc này, số lượng nhỏ hơn tương đối của các bit có thể được phân phối đối với giá trị thấp, và do đó hiệu quả lập mã tổng có thể được tăng cường.

Thiết bị giải mã có thể thực hiện việc giải mã dựa trên ngữ cảnh hoặc dựa trên việc vòng lặp qua của các ngăn tương ứng trong chuỗi ngăn từ luồng bit dựa trên kỹ thuật lập mã entropi, chẳng hạn như CABAC hoặc CAVLC. Ở đây, luồng bit có thể gồm các loại thông tin khác nhau cho việc giải mã hình ảnh/video như được mô tả ở trên. Luồng bit có thể được truyền đến thiết bị giải mã thông qua phương tiện lưu trữ (kỹ thuật số) hoặc mạng như được mô tả ở trên.

Theo sáng chế, để biểu diễn việc báo hiệu của thông tin từ thiết bị mã hóa đến

thiết bị giải mã, bảng (bảng cú pháp) gồm các phần tử cú pháp có thể được sử dụng. Thứ tự của các phần tử cú pháp trong bảng cú pháp được sử dụng trong sáng chế có thể thể hiện thứ tự phân tích cú pháp của các phần tử cú pháp từ luồng bit. Thiết bị mã hóa có thể tạo cấu hình và mã hóa bảng cú pháp sao cho các phần tử cú pháp có thể được phân tích cú pháp bởi thiết bị giải mã trong thứ tự phân tích cú pháp, và thiết bị giải mã có thể thu nhận các giá trị của các phần tử cú pháp bằng việc phân tích cú pháp và giải mã các phần tử cú pháp trong bảng cú pháp tương ứng trong thứ tự phân tích cú pháp từ luồng bit.

Trong khi đó, như được mô tả ở trên, các mẫu phần dư có thể được dẫn xuất dưới dạng các hệ số biến đổi được lượng tử hóa thông qua các quy trình biến đổi và lượng tử hóa. Các hệ số biến đổi được lượng tử hóa có thể được gọi là các hệ số biến đổi. Trong trường hợp này, các hệ số biến đổi trong khối có thể được báo hiệu dưới dạng thông tin phần dư. Thông tin phần dư có thể gồm cú pháp lập mã phần dư. Nghĩa là, thiết bị mã hóa có thể tạo cấu hình cú pháp lập mã phần dư với thông tin phần dư, và có thể mã hóa cú pháp lập mã phần dư được tạo cấu hình để xuất ra cú pháp lập mã phần dư được mã hóa dưới dạng luồng bit. Thiết bị giải mã có thể dẫn xuất các hệ số biến đổi (được lượng tử hóa) phần dư bằng việc giải mã cú pháp lập mã phần dư từ luồng bit. Như được mô tả dưới đây, cú pháp lập mã phần dư có thể gồm các phần tử cú pháp biểu diễn xem việc biến đổi có được áp dụng cho khối tương ứng hay không, đâu là vị trí của hệ số biến đổi hiệu quả cuối cùng trong khối, xem hệ số biến đổi hiệu quả có mặt trong khối con hay không, và kích thước/dấu của hệ số biến đổi hiệu quả là gì.

Ví dụ, các hệ số biến đổi (được lượng tử hóa) có thể được mã hóa và/hoặc được giải mã dựa trên các phần tử cú pháp, chẳng hạn như `last_sig_coeff_x_prefix`, `last_sig_coeff_y_prefix`, `last_sig_coeff_x_suffix`, `last_sig_coeff_y_suffix`, `coded_sub_block_flag`, `sig_coeff_flag`, `par_level_flag`, `abs_level_gtX_flag`, `abs_remainder`, `coeff_sign_flag`, `dec_abs_level` được gồm trong thông tin phần dư. Điều này có thể được gọi là lập mã (dữ liệu) phần dư hoặc lập mã hệ số (biến đổi). Trong

trường hợp này, quy trình biến đổi/lượng tử hóa có thể được bỏ qua. Trong trường hợp này, các giá trị của các mẫu phần dư có thể được lập mã và được báo hiệu theo phương pháp được xác định. Các phân tử cú pháp liên quan tới việc mã hóa/giải mã dữ liệu phần dư có thể được trình bày như trong Bảng 1 dưới đây.

[Bảng 1]

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	Descriptor
if((tu_mts_idx[x0][y0] > 0 (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx == 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
MaxCbs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)	
if(tu_mts_idx[x0][y0] > 0 (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx == 0 && log2TbHeight > 4)	
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	
if(log2TbWidth > 0)	
last_sig_coeff_x_prefix	ae(v)
if(log2TbHeight > 0)	
last_sig_coeff_y_prefix	ae(v)
if(last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
log2TbWidth = log2ZoTbWidth	
log2TbHeight = log2ZoTbHeight	
remBinsPass1 = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
log2SbW = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3) {	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 - log2SbW	
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 - log2SbH	
}	
}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TbWidth + log2TbHeight - (log2SbW + log2SbH))) - 1	
do {	
if(lastScanPos == 0) {	
lastScanPos = numSbCoeff	
lastSubBlock--	
}	
lastScanPos--	

$xS = \text{DiagScanOrder}[\log_2 \text{TbWidth} - \log_2 \text{SbW}][\log_2 \text{TbHeight} - \log_2 \text{SbH}][\text{lastSubBlock}][0]$	
$yS = \text{DiagScanOrder}[\log_2 \text{TbWidth} - \log_2 \text{SbW}][\log_2 \text{TbHeight} - \log_2 \text{SbH}][\text{lastSubBlock}][1]$	
$xC = (xS \ll \log_2 \text{SbW}) + \text{DiagScanOrder}[\log_2 \text{SbW}][\log_2 \text{SbH}][\text{lastScanPos}][0]$	
$yC = (yS \ll \log_2 \text{SbH}) + \text{DiagScanOrder}[\log_2 \text{SbW}][\log_2 \text{SbH}][\text{lastScanPos}][1]$	
$\} \text{ while}((xC \neq \text{LastSignificantCoeffX}) \ \ (yC \neq \text{LastSignificantCoeffY}))$	
$\text{if}(\text{lastSubBlock} == 0 \ \&\& \ \log_2 \text{TbWidth} \geq 2 \ \&\& \ \log_2 \text{TbHeight} \geq 2 \ \&\& \ !\text{transform_skip_flag}[x0][y0] \ \&\& \ \text{lastScanPos} > 0)$	
$\text{LfstDcOnly} = 0$	
$\text{if}((\text{lastSubBlock} > 0 \ \&\& \ \log_2 \text{TbWidth} \geq 2 \ \&\& \ \log_2 \text{TbHeight} \geq 2) \ \ (\text{lastScanPos} > 7 \ \&\& \ (\log_2 \text{TbWidth} == 2 \ \ \log_2 \text{TbWidth} == 3) \ \&\& \ \log_2 \text{TbWidth} == \log_2 \text{TbHeight}))$	
$\text{LfstZeroOutSigCoeffFlag} = 0$	
$\text{QState} = 0$	
$\text{for}(i = \text{lastSubBlock}; i \geq 0; i--) \{$	
$\text{startQStateSb} = \text{QState}$	
$xS = \text{DiagScanOrder}[\log_2 \text{TbWidth} - \log_2 \text{SbW}][\log_2 \text{TbHeight} - \log_2 \text{SbH}][i][0]$	
$yS = \text{DiagScanOrder}[\log_2 \text{TbWidth} - \log_2 \text{SbW}][\log_2 \text{TbHeight} - \log_2 \text{SbH}][i][1]$	
$\text{inferSbDcSigCoeffFlag} = 0$	
$\text{if}((i < \text{lastSubBlock}) \ \&\& \ (i > 0)) \{$	
$\text{coded_sub_block_flag}[xS][yS]$	$\text{ae}(v)$
$\text{inferSbDcSigCoeffFlag} = 1$	
$\}$	
$\text{firstSigScanPosSb} = \text{numSbCoeff}$	
$\text{lastSigScanPosSb} = -1$	
$\text{firstPosMode0} = (i == \text{lastSubBlock} ? \text{lastScanPos} : \text{numSbCoeff} - 1)$	
$\text{firstPosMode1} = -1$	
$\text{for}(n = \text{firstPosMode0}; n \geq 0 \ \&\& \ \text{remBinsPass1} \geq 4; n--) \{$	
$xC = (xS \ll \log_2 \text{SbW}) + \text{DiagScanOrder}[\log_2 \text{SbW}][\log_2 \text{SbH}][n][0]$	
$yC = (yS \ll \log_2 \text{SbH}) + \text{DiagScanOrder}[\log_2 \text{SbW}][\log_2 \text{SbH}][n][1]$	
$\text{if}(\text{coded_sub_block_flag}[xS][yS] \ \&\& \ (n > 0 \ \ !\text{inferSbDcSigCoeffFlag}) \ \&\& \ (xC \neq \text{LastSignificantCoeffX} \ \ yC \neq \text{LastSignificantCoeffY})) \{$	
$\text{sig_coeff_flag}[xC][yC]$	$\text{ae}(v)$
$\text{remBinsPass1}--$	
$\text{if}(\text{sig_coeff_flag}[xC][yC])$	
$\text{inferSbDcSigCoeffFlag} = 0$	
$\}$	
$\text{if}(\text{sig_coeff_flag}[xC][yC]) \{$	
$\text{abs_level_gtx_flag}[n][0]$	$\text{ae}(v)$
$\text{remBinsPass1}--$	
$\text{if}(\text{abs_level_gtx_flag}[n][0]) \{$	

par_level_flag[n]	ae(v)
remBinsPass1--	
abs_level_gtx_flag[n][1]	ae(v)
remBinsPass1--	
}	
if(lastSigScanPosSb == -1)	
lastSigScanPosSb = n	
firstSigScanPosSb = n	
}	
AbsLevelPass1[xC][yC] = sig_coeff_flag[xC][yC] + par_level_flag[n] + abs_level_gtx_flag[n][0] + 2 * abs_level_gtx_flag[n][1]	
if(dep_quant_enabled_flag)	
QState = QStateTransTable[QState][AbsLevelPass1[xC][yC] & 1]	
if(remBinsPass1 < 4)	
firstPosModel = n - 1	
}	
for(n = numSbCoeff - 1; n >= firstPosModel; n--) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(abs_level_gtx_flag[n][1])	
abs_remainder[n]	ae(v)
AbsLevel[xC][yC] = AbsLevelPass1[xC][yC] + 2 * abs_remainder[n]	
}	
for(n = firstPosModel; n >= 0; n--) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
dec_abs_level[n]	ae(v)
if(AbsLevel[xC][yC] > 0)	
firstSigScanPosSb = n	
if(dep_quant_enabled_flag)	
QState = QStateTransTable[QState][AbsLevel[xC][yC] & 1]	
}	
if(dep_quant_enabled_flag !sign_data_hiding_enabled_flag)	
signHidden = 0	
else	
signHidden = (lastSigScanPosSb - firstSigScanPosSb > 3 ? 1 : 0)	
for(n = numSbCoeff - 1; n >= 0; n--) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if((AbsLevel[xC][yC] > 0) && (!signHidden (n != firstSigScanPosSb)))	
coeff_sign_flag[n]	ae(v)
}	
if(dep_quant_enabled_flag) {	
QState = startQStateSb	
for(n = numSbCoeff - 1; n >= 0; n--) {	

$xC = (xS \ll \log2SbW) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][0]$	
$yC = (yS \ll \log2SbH) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][1]$	
$\text{if}(\text{AbsLevel}[xC][yC] > 0)$	
$\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$ $(2 * \text{AbsLevel}[xC][yC] - (\text{QState} > 1 ? 1 : 0)) *$ $(1 - 2 * \text{coeff_sign_flag}[n])$	
$\text{QState} = \text{QStateTransTable}[\text{QState}][\text{par_level_flag}[n]]$	
$\}$ else {	
$\text{sumAbsLevel} = 0$	
for($n = \text{numSbCoeff} - 1; n \geq 0; n--$) {	
$xC = (xS \ll \log2SbW) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][0]$	
$yC = (yS \ll \log2SbH) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][1]$	
$\text{if}(\text{AbsLevel}[xC][yC] > 0)$ {	
$\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$ $\text{AbsLevel}[xC][yC] * (1 - 2 * \text{coeff_sign_flag}[n])$	
$\text{if}(\text{signHidden})$ {	
$\text{sumAbsLevel} += \text{AbsLevel}[xC][yC]$	
$\text{if}((n == \text{firstSigScanPosSb}) \&\& (\text{sumAbsLevel} \% 2 == 1))$	
$\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$ $-\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC]$	
$\}$	
$\}$	
$\}$	
$\}$	
$\}$	
$\}$	

Tham khảo đến Bảng 1, `last_sig_coeff_x_prefix`, `last_sig_coeff_y_prefix`, `last_sig_coeff_x_suffix`, và `last_sig_coeff_y_suffix` là các phần tử cú pháp để mã hóa (x, y) thông tin vị trí của hệ số cuối cùng mà khác 0 trong khối được kết hợp. Khối được kết hợp có thể là khối lập mã (coding block, CB) hoặc khối biến đổi (transform block, TB). Liên quan tới quy trình biến đổi (và lượng tử hóa) và lập mã phần dư, CB và TB có thể được sử dụng thay thế cho nhau. Ví dụ, các mẫu phần dư có thể được dẫn xuất cho CB, và các hệ số biến đổi (được lượng tử hóa) có thể được dẫn xuất thông qua việc biến đổi và việc lượng tử hóa đối với các mẫu phần dư như được mô tả ở trên, và thông tin (hoặc các phần tử cú pháp) biểu diễn một cách hiệu quả (vị trí, kích thước, và dấu của) các hệ số biến đổi (được lượng tử hóa) có thể được tạo ra và được báo hiệu thông qua quy trình lập mã phần dư. Các hệ số biến đổi được lượng tử hóa có thể được gọi đơn giản là các hệ số biến đổi. Nói chung, nếu CB không lớn hơn TB tối đa, kích thước của CB có thể bằng với kích thước của TB, và trong trường hợp này, khối mục tiêu mà

được biến đổi (và được lượng tử hóa) và được lập mã phần dư có thể được gọi là CB hoặc TB. Ngoài ra, nếu CB là lớn hơn TB tối đa, khối mục tiêu mà được biến đổi (và được lượng tử hóa) và được lập mã phần dư có thể được gọi là TB. Sau đây, mặc dù được giải thích rằng các phần tử cú pháp liên quan tới việc lập mã phần dư được báo hiệu trong đơn vị của khối biến đổi (TB), đây chỉ là ví dụ, và TB có thể được sử dụng thay thế cho nhau với khối lập mã (CB) như được mô tả ở trên.

Trong khi đó, các phương pháp lập mã phần dư khác nhau có thể được áp dụng phụ thuộc vào xem việc bỏ qua biến đổi có được áp dụng cho việc lập mã phần dư hay không. Như một phương án, liệu việc bỏ qua biến đổi có được áp dụng hay không có thể được biểu diễn sử dụng phần tử cú pháp cờ bỏ qua biến đổi, và việc lập mã phần dư có thể được phân nhánh theo giá trị của phần tử cú pháp `transform_skip_flag` của cờ bỏ qua biến đổi. Nghĩa là, các phần tử cú pháp khác nhau có thể được sử dụng cho việc lập mã phần dư dựa trên giá trị của cờ bỏ qua biến đổi (dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không). Việc lập mã phần dư được sử dụng trong trường hợp mà việc bỏ qua biến đổi không được áp dụng (nghĩa là, trong trường hợp mà việc biến đổi được áp dụng) có thể được gọi là lập mã phần dư thông thường (regular residual coding, RRC), và việc lập mã phần dư trong trường hợp mà việc bỏ qua biến đổi được áp dụng (nghĩa là, trong trường hợp mà việc biến đổi không được áp dụng) có thể được gọi là lập mã phần dư bỏ qua biến đổi (transform skip residual coding, TSRC).

Bảng 2 dưới đây biểu diễn quy trình trong đó việc lập mã phần dư được phân nhánh dựa trên phần tử cú pháp của cờ bỏ qua biến đổi.

[Bảng 2]

transform unit(x0, y0, tbWidth, tbHeight, treeType, subTulIndex, chType) {	Descriptor
transform skip flag [x0][y0][0]	ae(v)
if(!transform skip flag[x0][y0][0] slice ts residual coding disabled flag)	
residual coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
else	
residual ts coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
}	
if(tu_cbf_cb[xC][yC] && treeType != DUAL_TREE_LUMA) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][1] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	
transform skip flag [xC][yC][1]	ae(v)
if(!transform skip flag[xC][yC][1] slice ts residual coding disabled flag)	
residual coding(xC, yC, Log2(wC), Log2(hC), 1)	
else	
residual ts coding(xC, yC, Log2(wC), Log2(hC), 1)	
}	
if(tu_cbf_cr[xC][yC] && treeType != DUAL_TREE_LUMA && !(tu_cbf_cb[xC][yC] && tu_joint_cbr_residual_flag[xC][yC])) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][2] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	
transform skip flag [xC][yC][2]	ae(v)
if(!transform skip flag[xC][yC][2] slice ts residual coding disabled flag)	
residual coding(xC, yC, Log2(wC), Log2(hC), 2)	
else	
residual ts coding(xC, yC, Log2(wC), Log2(hC), 2)	
}	
}	

Tham khảo đến Bảng 2 ở trên, trong trường hợp mà việc bỏ qua biến đổi không được áp dụng (ví dụ, trong trường hợp mà giá trị của transform_skip_flag là 0), việc lập mã phần dư thông thường được thực hiện, và điều này có thể được thực hiện dựa trên các phần tử cú pháp được liệt kê trong Bảng 1 được mô tả ở trên. Ngoài ra, trong trường hợp mà việc bỏ qua biến đổi được áp dụng (ví dụ, trong trường hợp mà giá trị của transform_skip_flag là 1), việc lập mã phần dư bỏ qua biến đổi được thực hiện, và điều này có thể được thực hiện dựa trên các phần tử cú pháp được liệt kê trong Bảng 3 dưới đây.

Bảng 3 dưới đây biểu diễn các phần tử cú pháp cho việc lập mã phần dư bỏ qua biến đổi.

[Bảng 3]

	Descriptor
<code>residual_ts_coding(x0, y0, log2TbWidth, log2TbHeight, cldx) {</code>	
<code>log2SbSize = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)</code>	
<code>numSbCoeff = 1 << (log2SbSize << 1)</code>	
<code>lastSubBlock = (1 << (log2TbWidth + log2TbHeight - 2 * log2SbSize)) - 1</code>	
<code>inferSbCbf = 1</code>	
<code>MaxCcbs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)</code>	
<code>for(i = 0; i <= lastSubBlock; i++) {</code>	
<code> xS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][0]</code>	
<code> yS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][1]</code>	
<code> if((i != lastSubBlock !inferSbCbf) {</code>	
<code> coded_sub_block_flag[xS][yS]</code>	ae(v)
<code> }</code>	
<code> if(coded_sub_block_flag[xS][yS] && i < lastSubBlock)</code>	
<code> inferSbCbf = 0</code>	
<code>/* First scan pass */</code>	
<code> inferSbSigCoeffFlag = 1</code>	
<code> for(n = 0; n <= numSbCoeff - 1; n++) {</code>	
<code> xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]</code>	
<code> yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]</code>	
<code> if(coded_sub_block_flag[xS][yS] &&</code>	
<code> (n != numSbCoeff - 1 !inferSbSigCoeffFlag)) {</code>	
<code> sig_coeff_flag[xC][yC]</code>	ae(v)
<code> MaxCcbs--</code>	
<code> if(sig_coeff_flag[xC][yC])</code>	
<code> inferSbSigCoeffFlag = 0</code>	
<code> }</code>	
<code> CoeffSignLevel[xC][yC] = 0</code>	
<code> if(sig_coeff_flag[xC][yC] {</code>	
<code> coeff_sign_flag[n]</code>	ae(v)
<code> MaxCcbs--</code>	
<code> CoeffSignLevel[xC][yC] = (coeff_sign_flag[n] > 0 ? -1 : 1)</code>	
<code> abs_level_gtx_flag[n][0]</code>	ae(v)
<code> MaxCcbs--</code>	
<code> if(abs_level_gtx_flag[n][0]) {</code>	
<code> par_level_flag[n]</code>	ae(v)
<code> MaxCcbs--</code>	
<code> }</code>	
<code> }</code>	
<code> AbsLevelPassX[xC][yC] = sig_coeff_flag[xC][yC] + par_level_flag[n] +</code>	
<code> abs_level_gtx_flag[n][0]</code>	
<code> }</code>	
<code>/* Greater than X scan pass (numGtXFlags=5) */</code>	
<code> for(n = 0; n <= numSbCoeff - 1; n++) {</code>	
<code> xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]</code>	
<code> yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]</code>	
<code> for(j = 1; j < 5; j++) {</code>	

if(abs_level_gtx_flag[n][j - 1])	
abs_level_gtx_flag[n][j]	ae(v)
MaxCcbs--	
AbsLevelPassX[xC][yC] += 2 * abs_level_gtx_flag[n][j]	
}	
}	
/* remainder scan pass */	
for(n = 0; n <= numSbCoeff - 1; n++) {	
xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if(abs_level_gtx_flag[n][4])	
abs_remainder[n]	ae(v)
if(intra_bdpcm_flag == 0) {	
absRightCoeff = abs(TransCoeffLevel[x0][y0][cIdx][xC - 1][yC])	
absBelowCoeff = abs(TransCoeffLevel[x0][y0][cIdx][xC][yC - 1])	
predCoeff = Max(absRightCoeff, absBelowCoeff)	
if(AbsLevelPassX[xC][yC] + abs_remainder[n] == 1 && predCoeff >	
0)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
(1 - 2 * coeff_sign_flag[n]) * predCoeff	
else if(AbsLevelPassX[xC][yC] + abs_remainder[n] <= predCoeff)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n] - 1)	
else	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n])	
} else	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n])	
}	
}	
}	

Ví dụ, cờ bỏ qua biến đổi chỉ ra xem việc bỏ qua biến đổi của khối biến đổi được thực hiện có thể được phân tích cú pháp hay không, và có thể được xác định xem cờ bỏ qua biến đổi có phải là 1 hay không. Trong trường hợp mà giá trị của cờ bỏ qua biến đổi là 1, như được thể hiện trong Bảng 3, các phần tử cú pháp sig_coeff_flag, coeff_sign_flag, abs_level_gtx_flag, par_level_flag, và/hoặc abs_remainder cho hệ số phần dư của khối biến đổi có thể được phân tích cú pháp, và hệ số phần dư có thể được dẫn xuất dựa trên các phần tử cú pháp. Trong trường hợp này, các phần tử cú pháp có thể được phân tích cú pháp tuần tự, và thứ tự phân tích cú pháp có thể được thay đổi. Ở đây, abs_level_gtx_flag có thể thể hiện abs_level_gt1_flag, abs_level_gt3_flag,

`abs_level_gt5_flag`, `abs_level_gt7_flag`, và/hoặc `abs_level_gt9_flag`. Ví dụ, `abs_level_gtx_flag[n][j]` có thể là cờ biểu diễn xem giá trị tuyệt đối của mức hệ số biến đổi (hoặc giá trị thu được bằng việc dịch chuyển mức hệ số biến đổi bởi 1 sang phải) có lớn hơn $(j < 1) + 1$ ở vị trí quét n hay không. $(j < 1) + 1$ có thể được thay thế bởi giá trị ngưỡng cụ thể, chẳng hạn như giá trị ngưỡng thứ nhất hoặc giá trị ngưỡng thứ hai trong một số trường hợp.

Ngoài ra, trong trường hợp mà giá trị của cờ bỏ qua biến đổi là 0, như được thể hiện trong Bảng 1, các phần tử cú pháp `sig_coeff_flag`, `abs_level_gtx_flag`, `par_level_flag`, `abs_remainder`, `dec_abs_level`, và `coeff_sign_flag` cho hệ số phần dư của khối biến đổi có thể được phân tích cú pháp, và hệ số phần dư có thể được dẫn xuất dựa trên các phần tử cú pháp. Trong trường hợp này, các phần tử cú pháp có thể được phân tích cú pháp tuần tự, và thứ tự phân tích cú pháp có thể được thay đổi. Ở đây, `abs_level_gtx_flag` có thể thể hiện `abs_level_gt1_flag` và/hoặc `abs_level_gt3_flag`.

Trong khi đó, như được mô tả ở trên, thiết bị mã hóa có thể dẫn xuất khối phần dư (các mẫu phần dư) dựa trên khối được dự đoán (các mẫu dự đoán) thông qua các việc dự đoán nội/liên/IBC/bảng màu, và có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng việc áp dụng việc biến đổi và việc lượng tử hóa đối với các mẫu phần dư được dẫn xuất. Thông tin (thông tin phần dư) về các hệ số biến đổi được lượng tử hóa có thể được gồm trong cú pháp lập mã phần dư, và có thể được xuất ra dưới dạng luồng bit sau khi được mã hóa. Thiết bị giải mã có thể thu nhận thông tin (thông tin phần dư) về các hệ số biến đổi được lượng tử hóa từ luồng bit, và có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng việc giải mã thông tin được thu nhận. Thiết bị giải mã có thể dẫn xuất các mẫu phần dư thông qua việc khử lượng tử hóa/biến đổi ngược dựa trên các hệ số biến đổi được lượng tử hóa. Như được mô tả ở trên, ít nhất một trong số việc lượng tử hóa/việc khử lượng tử hóa và/hoặc việc biến đổi/việc biến đổi ngược có thể được bỏ qua. Trong trường hợp mà việc biến đổi/việc biến đổi ngược được bỏ qua, hệ số biến đổi có thể được gọi là hệ số hoặc hệ số phần dư, hoặc để trình bày đồng nhất, có thể vẫn được

gọi là hệ số biến đổi. Việc liệu có bỏ qua việc biến đổi/việc biến đổi ngược hay không có thể được báo hiệu dựa trên `transform_skip_flag`. Ví dụ, nếu giá trị của `transform_skip_flag` là 1, nó có thể thể hiện rằng việc biến đổi/việc biến đổi ngược được bỏ qua, và điều này có thể được gọi là chế độ bỏ qua biến đổi.

Nói chung, trong việc lập mã video/hình ảnh, tỷ lệ lượng tử hóa có thể được thay đổi, và tỷ lệ nén có thể được điều chỉnh sử dụng tỷ lệ lượng tử hóa được thay đổi. Từ quan điểm việc thực hiện, thông số lượng tử hóa (quantization parameter, QP) có thể được sử dụng thay cho việc sử dụng trực tiếp tỷ lệ lượng tử hóa xét đến độ phức tạp. Ví dụ, thông số lượng tử hóa có giá trị nguyên trong khoảng từ 0 đến 63 có thể được sử dụng, và mỗi giá trị của thông số lượng tử hóa có thể tương ứng với tỷ lệ lượng tử hóa thực. Ngoài ra, ví dụ, thông số lượng tử hóa QP_Y cho thành phần độ chói (mẫu độ chói) và thông số lượng tử hóa QP_C cho thành phần sắc độ (mẫu sắc độ) có thể được tạo cấu hình khác nhau.

Trong quy trình lượng tử hóa, hệ số biến đổi được lượng tử hóa C' có thể được thu nhận bằng việc chia hệ số biến đổi đầu vào C cho tỷ lệ lượng tử hóa Q_{step} . Trong trường hợp này, xem xét đến độ phức tạp tính toán, tỷ lệ lượng tử hóa được tạo ra dưới dạng số nguyên bằng việc nhân tỷ lệ lượng tử hóa theo một thang, và thao tác dịch chuyển có thể được thực hiện cho tới khi giá trị tương ứng với giá trị thang. Thang lượng tử hóa có thể được dẫn xuất dựa trên tích của tỷ lệ lượng tử hóa và giá trị thang. Nghĩa là, thang lượng tử hóa có thể được dẫn xuất theo QP. Ví dụ, thang lượng tử hóa có thể được áp dụng cho hệ số biến đổi C , và dựa trên điều này, hệ số biến đổi được lượng tử hóa C' có thể được dẫn xuất.

Quy trình khử lượng tử hóa là ngược với quy trình lượng tử hóa, và hệ số biến đổi được lượng tử hóa C' có thể được nhân với tỷ lệ lượng tử hóa Q_{step} , và dựa trên điều này, hệ số biến đổi được xây dựng lại C'' có thể được thu nhận. Trong trường hợp này, thang cấp độ có thể được dẫn xuất theo thông số lượng tử hóa, thang cấp độ có thể được áp dụng cho hệ số biến đổi được lượng tử hóa C' , và dựa trên điều này, hệ số biến đổi

được xây dựng lại C” có thể được dẫn xuất. Hệ số biến đổi được xây dựng lại C” có thể có một số khác biệt với hệ số biến đổi ban đầu C do sự suy hao trong quá trình biến đổi và/hoặc lượng tử hóa. Theo đó, ngay cả thiết bị mã hóa thực hiện việc khử lượng tử hóa theo cùng cách như trong thiết bị giải mã.

Ngoài ra, trong việc thực hiện việc dự đoán, có thể dựa trên việc lập mã bảng màu. Việc lập mã bảng màu là công nghệ hữu ích để biểu diễn các khối gồm số lượng nhỏ các giá trị màu sắc riêng biệt. Thay vì áp dụng việc dự đoán và biến đổi cho khối, trong chế độ bảng màu, chỉ số được báo hiệu để biểu diễn giá trị của mỗi mẫu. Chế độ bảng màu là hữu dụng để tiết kiệm khoảng trống đệm bộ nhớ video. Khối có thể được lập mã sử dụng chế độ bảng màu (ví dụ, `MODE_PLT`). Để giải mã khối được mã hóa như được mô tả ở trên, bộ giải mã nên giải mã mục nhập cổng và chỉ số. Mục nhập bảng màu có thể được biểu diễn bởi bảng của bảng màu, và có thể được mã hóa bởi công cụ lập mã bảng của bảng màu.

Việc lập mã bảng màu có thể được gọi là chế độ bảng màu (nội) hoặc chế độ lập mã bảng màu (nội). Khối hiện tại có thể được xây dựng lại theo việc lập mã bảng màu hoặc chế độ bảng màu. Việc lập mã bảng màu có thể được xem là ví dụ của việc lập mã nội, hoặc có thể được xem là một trong các phương pháp dự đoán nội. Tuy nhiên, theo cách tương tự với chế độ bỏ qua được mô tả ở trên, giá trị phần dư tách biệt cho khối tương ứng có thể không được báo hiệu.

Ví dụ, trong trường hợp mà chế độ bảng màu được lựa chọn, thông tin về bảng của bảng màu có thể được báo hiệu. Bảng của bảng màu có thể gồm chỉ số tương ứng với mỗi điểm ảnh. Bảng của bảng màu có thể tạo cấu hình bảng dự đoán bảng màu từ các giá trị điểm ảnh được sử dụng trong khối trước đó. Ví dụ, các giá trị điểm ảnh được sử dụng trước đó có thể được lưu trữ trong bộ đệm cụ thể (bộ dự đoán bảng màu), và thông tin bộ dự đoán bảng màu `palette_predictor_run` để tạo cấu hình bảng màu hiện tại có thể được nhận từ bộ đệm. Nghĩa là, bộ dự đoán bảng màu có thể gồm dữ liệu biểu diễn chỉ số cho ít nhất một phần của bản đồ chỉ số bảng màu của khối hiện tại. Trong

trường hợp mà mục nhập bảng màu để thể hiện khối hiện tại là không đủ như mục nhập dự đoán bảng màu được tạo cấu hình từ bộ dự đoán bảng màu, thông tin điểm ảnh về mục nhập bảng màu hiện tại có thể được truyền một cách tách biệt.

Chế độ bảng màu có thể được báo hiệu ở mức CU, và có thể thường được sử dụng trong trường hợp mà hầu hết các điểm ảnh trong CU có thể được biểu diễn dưới dạng tập của các giá trị điểm ảnh biểu diễn. Nghĩa là, trong chế độ bảng màu, các mẫu trong CU có thể được thể hiện dưới dạng tập của các giá trị điểm ảnh biểu diễn. Tập như vậy có thể được gọi là bảng màu. Trong trường hợp mẫu có giá trị gần với giá trị điểm ảnh trong bảng màu, chỉ số bảng màu `palette_idx_idc` tương ứng với giá trị điểm ảnh trong bảng màu hoặc thông tin có khả năng chỉ ra chỉ số (`run_copy_flag`, `copy_above_palette_indices_flag`) có thể được báo hiệu. Trong trường hợp mẫu có giá trị điểm ảnh loại trừ mục nhập bảng màu, mẫu có thể được chỉ ra như ký hiệu thoát, và giá trị mẫu được lượng tử hóa có thể được báo hiệu trực tiếp. Theo sáng chế, điểm ảnh hoặc giá trị điểm ảnh có thể được gọi là mẫu hoặc giá trị mẫu.

Để giải mã khối được lập mã trong chế độ bảng màu, bộ giải mã yêu cầu thông tin mục nhập bảng màu và thông tin chỉ số bảng màu. Trong trường hợp mà chỉ số bảng màu tương ứng với ký hiệu thoát, giá trị thoát (được lượng tử hóa) có thể được báo hiệu như thành phần bổ sung. Ngoài ra, bộ mã hóa nên dẫn xuất bảng màu phù hợp cho CU tương ứng, và truyền bảng màu tới bộ giải mã.

Cho việc lập mã hiệu quả của mục nhập công, bộ dự đoán bảng màu có thể được duy trì. Bộ dự đoán bảng màu và kích thước tối đa của bảng màu có thể được báo hiệu thông qua SPS. Ngoài ra, bộ dự đoán bảng màu và kích thước tối đa của bảng màu có thể được định rõ trước. Ví dụ, bộ dự đoán bảng màu và kích thước tối đa của bảng màu có thể được xác định là 31 và 15, một cách tương ứng, phụ thuộc vào việc liệu khối hiện tại là cây đơn hay cây đôi. Trong các tiêu chuẩn VVC, `sps_palette_enabled_flag` biểu diễn xem chế độ bảng màu được cho phép có thể được truyền hay không. Sau đó, cờ `pred_mode_plt_coding` biểu diễn xem đơn vị lập mã hiện tại được lập mã trong chế độ

bảng màu có thể được truyền hay không. Bộ dự đoán bảng màu có thể được bắt đầu tại điểm bắt đầu của mỗi viên gạch hoặc mỗi lát.

Đối với mỗi mục nhập trong bộ dự đoán bảng màu, cờ sử dụng lại có thể được báo hiệu, và có thể thể hiện xem mục nhập có phải là một phần của cổng hiện tại hay không. Cờ sử dụng lại có thể được truyền sử dụng việc lập mã thời gian chạy là 0. Sau đó, số lượng của các mục nhập bảng màu mới có thể được báo hiệu sử dụng việc lập mã Golomb hàm số mũ bậc 0. Cuối cùng, giá trị thành phần cho mục nhập bảng màu với có thể được báo hiệu. Sau khi CU hiện tại được mã hóa, bộ dự đoán bảng màu có thể được cập nhật sử dụng bảng màu hiện tại, và mục nhập của bộ dự đoán bảng màu trước đó mà không được sử dụng lại trong bảng màu hiện tại có thể được bổ sung vào phần cuối của bộ dự đoán bảng màu mới (nhồi bảng màu) cho đến khi nó đạt kích thước tối đa có thể cho phép.

Để lập mã bản đồ chỉ số bảng màu, chỉ số có thể được lập mã sử dụng các việc quét qua ngang và dọc. Thứ tự quét có thể được báo hiệu rõ ràng từ luồng bit sử dụng thông tin cờ (ví dụ, `palette_transpose_flag`).

Trong khi đó, chỉ số bảng màu có thể được lập mã sử dụng hai loại chế độ mẫu bảng màu, và ví dụ, chế độ "INDEX" và chế độ "COPY_ABOVE" có thể được sử dụng. Các chế độ bảng màu như vậy có thể được báo hiệu sử dụng cờ biểu diễn xem chế độ bảng màu là chế độ "INDEX" hay chế độ "COPY_ABOVE". Trong trường hợp này, ký hiệu thoát có thể được báo hiệu trong chế độ "INDEX", và chỉ số có cùng kích thước như kích thước bảng màu hiện tại có thể được phân phối. Ví dụ, nếu được giả định rằng kích thước của bảng màu hiện tại là 10, các chỉ số số 1 đến số 9 có thể nghĩa là các chỉ số mục nhập trong bảng màu, và chỉ số số 10 có thể nghĩa là chỉ số cho ký hiệu thoát. Trong trường hợp mà việc quét ngang được sử dụng, cờ có thể được báo hiệu ngoại trừ dòng trên cùng, và trong trường hợp mà việc quét dọc được sử dụng, hoặc chế độ trước đó là chế độ "COPY_ABOVE", cờ có thể được báo hiệu ngoại trừ cột đầu tiên. Trong chế độ "COPY_ABOVE", chỉ số bảng màu của mẫu của hàng trên cùng có thể được sao

chép. Trong chế độ "INDEX", chỉ số bảng màu có thể được báo hiệu rõ ràng. Trong cả chế độ "INDEX" và chế độ "COPY_ABOVE", giá trị chạy biểu diễn số lượng của các mẫu tiếp theo được lập mã sử dụng cùng chế độ có thể được báo hiệu. Trong trường hợp mà ký hiệu thoát là một phần của việc chạy trong chế độ "INDEX" hoặc chế độ "COPY_ABOVE", các giá trị thành phần thoát có thể được báo hiệu đối với mỗi ký hiệu thoát.

Việc lập mã của chỉ số bảng màu có thể là như dưới đây. Đầu tiên, số lượng của các chỉ số cho CU có thể được báo hiệu. Tiếp theo, các chỉ số thực cho toàn bộ CU có thể được báo hiệu sử dụng việc lập mã độ dài cố định. Số lượng của các chỉ số và chỉ số có thể được lập mã theo chế độ đi vòng lặp qua. Thông qua việc này, các ngăn đi vòng lặp qua liên quan đến chỉ số có thể được nhóm với nhau. Tiếp theo, chế độ mẫu bảng màu `copy_above_palette_indices_flag` và việc chạy có thể được báo hiệu theo cách đan xen. Cuối cùng, các giá trị thoát thành phần tương ứng với các mẫu thoát cho toàn bộ CU có thể được nhóm với nhau, và có thể được lập mã theo chế độ đi vòng lặp qua.

Bảng 4 dưới đây biểu diễn ví dụ của cấu trúc cú pháp gồm các phần tử cú pháp liên quan tới việc lập mã dựa trên chế độ bảng màu cho đơn vị lập mã, và Bảng 5 dưới đây biểu diễn các ngữ nghĩa cho các phần tử cú pháp được gồm trong cú pháp của Bảng 4.

[Bảng 4]

palette_coding(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
startComp = (treeType == DUAL_TREE_CHROMA) ? 1 : 0	
numComps = (treeType == SINGLE_TREE) ? (sps_chroma_format_idc == 0 ? 1 : 3) : (treeType == DUAL_TREE_CHROMA) ? 2 : 1	
maxNumPaletteEntries = (treeType == SINGLE_TREE) ? 31 : 15	
palettePredictionFinished = 0	
NumPredictedPaletteEntries = 0	
for(predictorEntryIdx = 0; predictorEntryIdx < PredictorPaletteSize[startComp] && !palettePredictionFinished && NumPredictedPaletteEntries < maxNumPaletteEntries; predictorEntryIdx++) {	
palette_predictor_run	ae(v)
if(palette_predictor_run != 1) {	
if(palette_predictor_run > 1)	
predictorEntryIdx += palette_predictor_run - 1	
PalettePredictorEntryReuseFlags[predictorEntryIdx] = 1	
NumPredictedPaletteEntries++	
} else	
palettePredictionFinished = 1	
}	
if(NumPredictedPaletteEntries < maxNumPaletteEntries)	
num_signalled_palette_entries	ae(v)
for(cldx = startComp; cldx < (startComp + numComps); cldx++)	
for(i = 0; i < num_signalled_palette_entries; i++)	
new_palette_entries[cldx][i]	ae(v)
if(CurrentPaletteSize[startComp] > 0)	
palette_escape_val_present_flag	ae(v)
if(MaxPaletteIndex > 0) {	
adjust = 0	
palette_transpose_flag	ae(v)
}	
if(treeType != DUAL_TREE_CHROMA && palette_escape_val_present_flag)	
if(pps_cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded) {	
cu_qp_delta_abs	ae(v)
if(cu_qp_delta_abs)	
cu_qp_delta_sign_flag	ae(v)
}	
if(treeType != DUAL_TREE_LUMA && palette_escape_val_present_flag)	
if(sh_cu_chroma_qp_offset_enabled_flag && !IsCuChromaQpOffsetCoded) {	
cu_chroma_qp_offset_flag	ae(v)
if(cu_chroma_qp_offset_flag && pps_chroma_qp_offset_list_len_minus1 > 0)	
cu_chroma_qp_offset_idx	ae(v)
}	
PreviousRunPosition = 0	
PreviousRunType = 0	
for(subSetId = 0; subSetId <= (cbWidth * cbHeight - 1) / 16; subSetId++) {	
minSubPos = subSetId * 16	

if(minSubPos + 16 > cbWidth * cbHeight)	
maxSubPos = cbWidth * cbHeight	
else	
maxSubPos = minSubPos + 16	
RunCopyMap[x0][y0] = 0	
PaletteScanPos = minSubPos	
log2CbWidth = Log2(cbWidth)	
log2CbHeight = Log2(cbHeight)	
while(PaletteScanPos < maxSubPos) {	
xC = x0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][0]	
yC = y0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][1]	
if(PaletteScanPos > 0) {	
xcPrev = x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]	
ycPrev = y0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][1]	
}	
if(MaxPaletteIndex > 0 && PaletteScanPos > 0) {	
run_copy_flag	ae(v)
RunCopyMap[xC][yC] = run_copy_flag	
}	
CopyAboveIndicesFlag[xC][yC] = 0	
if(MaxPaletteIndex > 0 && !RunCopyMap[xC][yC]) {	
if(((!palette_transpose_flag && yC > y0)	
(palette_transpose_flag && xC > x0)) &&	
CopyAboveIndicesFlag[xcPrev][ycPrev] == 0 &&	
PaletteScanPos > 0) {	
copy_above_palette_indices_flag	ae(v)
CopyAboveIndicesFlag[xC][yC] = copy_above_palette_indices_flag	
}	
PreviousRunType = CopyAboveIndicesFlag[xC][yC]	
PreviousRunPosition = PaletteScanPos	
} else if(PaletteScanPos > 0)	
CopyAboveIndicesFlag[xC][yC] = CopyAboveIndicesFlag[xcPrev][ycPrev]	
PaletteScanPos ++	
}	
PaletteScanPos = minSubPos	
while(PaletteScanPos < maxSubPos) {	
xC = x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][0]	
yC = y0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][1]	
if(PaletteScanPos > 0) {	
xcPrev =x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]	

ycPrev = y0 + TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][1]	
}	
if(MaxPaletteIndex > 0 && !RunCopyMap[xC][yC] && CopyAboveIndicesFlag[xC][yC] == 0) {	
if(MaxPaletteIndex - adjust > 0)	
palette_idx_idc	ae(v)
adjust = 1	
}	
if(!RunCopyMap[xC][yC] && CopyAboveIndicesFlag[xC][yC] == 0)	
CurrPaletteIndex = palette_idx_idc	
if(CopyAboveIndicesFlag[xC][yC] == 0)	
PaletteIndexMap[xC][yC] = CurrPaletteIndex	
else if(!palette_transpose_flag)	
PaletteIndexMap[xC][yC] = PaletteIndexMap[xC][yC - 1]	
else	
PaletteIndexMap[xC][yC] = PaletteIndexMap[xC - 1][yC]	
PaletteScanPos ++	
}	
if(palette_escape_val_present_flag) {	
for(cIdx = startComp; cIdx < (startComp + numComps); cIdx++) {	
for(sPos = minSubPos; sPos < maxSubPos; sPos++) {	
xC = x0 + TraverseScanOrder[log2CbWidth][log2CbHeight][sPos][0]	
yC = y0 + TraverseScanOrder[log2CbWidth][log2CbHeight][sPos][1]	
if(!(treeType == SINGLE_TREE && cIdx != 0 && (xC % SubWidthC != 0 yC % SubHeightC != 0))) {	
if(PaletteIndexMap[cIdx][xC][yC] == MaxPaletteIndex) {	
palette_escape_val	ae(v)
PaletteEscapeVal[cIdx][xC][yC] = palette_escape_val	
}	
}	
}	
}	
}	
}	
}	

[Bảng 5]

In the following semantics, the array indices x_0, y_0 specify the location (x_0, y_0) of the top-left luma sample of the considered coding block relative to the top-left luma sample of the picture. The array indices x_C, y_C specify the location (x_C, y_C) of the sample relative to the top-left luma sample of the picture, when `treeType` is equal to `SINGLE_TREE` or `DUAL_TREE_LUMA`; and relative to the top-left chroma sample of the picture, when `treeType` is equal to `DUAL_TREE_CHROMA`. The array index `startComp` specifies the first colour component of the current palette table. `startComp` equal to 0 indicates the Y component; `startComp` equal to 1 indicates the Cb component; `startComp` equal to 2 indicates the Cr component. `numComps` specifies the number of colour components in the current palette table.

The predictor palette consists of palette entries from previous coding units that are used to predict the entries in the current palette.

`PredictorPaletteSize[ startComp ]` specifies the size of the predictor palette for the first colour component of the current palette table `startComp`.

`PredictorPaletteSize[ startComp ]` is derived as specified in subclause 8.4.5.3.

`PalettePredictorEntryReuseFlags[ i ]` equal to 1 specifies that the i -th entry in the predictor palette is reused in the current palette. `PalettePredictorEntryReuseFlags[ i ]` equal to 0 specifies that the i -th entry in the predictor palette is not an entry in the current palette. All elements of the array `PalettePredictorEntryReuseFlags[ i ]` are initialized to 0.

palette_predictor_run is used to determine the number of zeros that precede a non-zero entry in the array `PalettePredictorEntryReuseFlags`.

It is a requirement of bitstream conformance that the value of `palette_predictor_run` shall be in the range of 0 to (`PredictorPaletteSize[ startComp ] - predictorEntryIdx`), inclusive, where `predictorEntryIdx` corresponds to the current position in the array `PalettePredictorEntryReuseFlags`. The variable `NumPredictedPaletteEntries` specifies the number of entries in the current palette that are reused from the predictor palette. The value of `NumPredictedPaletteEntries` shall be in the range of 0 to `maxNumPaletteEntries`, inclusive.

num_signalled_palette_entries specifies the number of entries in the current palette that are explicitly signalled for the first colour component of the current palette table startComp.

When **num_signalled_palette_entries** is not present, it is inferred to be equal to 0.

The variable **CurrentPaletteSize[startComp]** specifies the size of the current palette for the first colour component of the current palette table startComp and is derived as follows:

$$\text{CurrentPaletteSize[startComp]} = \text{NumPredictedPaletteEntries} + \text{num_signalled_palette_entries} \quad (176)$$

The value of **CurrentPaletteSize[startComp]** shall be in the range of 0 to **maxNumPaletteEntries**, inclusive.

new_palette_entries[cIdx][i] specifies the value for the i-th signalled palette entry for the colour component cIdx.

The variable **LocalDualTreeFlag** is derived as follows:

$$\text{LocalDualTreeFlag} = (\text{treeType} \neq \text{SINGLE_TREE} \ \&\& \\ (\text{sh_slice_type} \neq \text{I} \ || \ (\text{sh_slice_type} == \text{I} \ \&\& \\ \text{sps_qtbtt_dual_tree_intra_flag} == 0))) ? 1 : 0 \quad (177)$$

The variable **PredictorPaletteEntries[cIdx][i]** specifies the i-th element in the predictor palette for the colour component cIdx.

The variable **CurrentPaletteEntries[cIdx][i]** specifies the i-th element in the current palette for the colour component cIdx and is derived as follows:

```

numPredictedPaletteEntries = 0
for( i = 0; i < PredictorPaletteSize[ startComp ]; i++ )
    if( PalettePredictorEntryReuseFlags[ i ] ) {
        for( cIdx = LocalDualTreeFlag ? 0 : startComp;
            cIdx < LocalDualTreeFlag ? 3 : ( startComp + numComps ); cIdx++ )
            CurrentPaletteEntries[ cIdx ][ numPredictedPaletteEntries ] =
                PredictorPaletteEntries[ cIdx ][ i ]
    }

```



```

        numPredictedPaletteEntries++
    }
    for( cIdx = startComp; cIdx < (startComp + numComps); cIdx++) (178)
        for( i = 0; i < num_signalled_palette_entries; i++)
            CurrentPaletteEntries[ cIdx ][ numPredictedPaletteEntries + i ] =
                new_palette_entries[ cIdx ][ i ]

```

palette_escape_val_present_flag equal to 1 specifies that the current coding unit contains at least one escape coded sample. **palette_escape_val_present_flag** equal to 0 specifies that there are no escape coded samples in the current coding unit. When not present, the value of **palette_escape_val_present_flag** is inferred to be equal to 1.

The variable **MaxPaletteIndex** specifies the maximum possible value for a palette index for the current coding unit. The value of **MaxPaletteIndex** is set equal to **CurrentPaletteSize[startComp] - 1 + palette_escape_val_present_flag**.

palette_idx_idc is an indication of an index to the palette table, **CurrentPaletteEntries**. The value of **palette_idx_idc** shall be in the range of 0 to **MaxPaletteIndex**, inclusive, for the first index in the block and in the range of 0 to **(MaxPaletteIndex - 1)**, inclusive, for the remaining indices in the block.

When **palette_idx_idc** is not present, it is inferred to be equal to 0.

palette_transpose_flag equal to 1 specifies that vertical traverse scan is applied for scanning the indices for samples in the current coding unit. **palette_transpose_flag** equal to 0 specifies that horizontal traverse scan is applied for scanning the indices for samples in the current coding unit. When not present, the value of **palette_transpose_flag** is inferred to be equal to 0.

The array **TraverseScanOrder** specifies the scan order array for palette coding. If **palette_transpose_flag** is equal to 0, **TraverseScanOrder** is assigned the horizontal scan order **HorTravScanOrder**. Otherwise (**palette_transpose_flag** is equal to 1), **TraverseScanOrder** is assigned the vertical scan order **VerTravScanOrder**.

run_copy_flag equal to 1 specifies that the palette run type is the same as the run type at the previously scanned position and palette index is the same as the index at the previous scanned position if **CopyAboveIndicesFlag[xC][yC]** is equal to 0.

Otherwise, `run_copy_flag` equal to 0 specifies that the palette run type is different from the run type at the previously scanned position.

`copy_above_palette_indices_flag` equal to 1 specifies that the palette index is equal to the palette index at the same location in the row above if horizontal traverse scan is used or the same location in the left column if vertical traverse scan is used. `copy_above_palette_indices_flag` equal to 0 specifies that an indication of the palette index of the sample is coded in the bitstream or inferred.

The variable `CopyAboveIndicesFlag[ xC ][ yC ]` equal to 1 specifies that the palette index is copied from the palette index in the row above (horizontal scan) or left column (vertical scan). `CopyAboveIndicesFlag[ xC ][ yC ]` equal to 0 specifies that the palette index is explicitly coded in the bitstream or inferred.

The variable `PaletteIndexMap[ xC ][ yC ]` specifies a palette index, which is an index to the array represented by `CurrentPaletteEntries`. The value of `PaletteIndexMap[ xC ][ yC ]` shall be in the range of 0 to `MaxPaletteIndex`, inclusive.

The variable `adjustedRefPaletteIndex` is derived as follows:

```
adjustedRefPaletteIndex = MaxPaletteIndex + 1
if( PaletteScanPos > 0 ) {
    xcPrev = x0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][0]
    ycPrev = y0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][1]
    if( CopyAboveIndicesFlag[ xcPrev ][ ycPrev ] == 0 )
        adjustedRefPaletteIndex = PaletteIndexMap[xcPrev][ycPrev]    (179)
    else {
        if( !palette_transpose_flag )
            adjustedRefPaletteIndex = PaletteIndexMap[ xC ][ yC - 1 ]
        else
            adjustedRefPaletteIndex = PaletteIndexMap[ xC - 1 ][ yC ]
    }
}
```

When `CopyAboveIndicesFlag[ xC ][ yC ]` is equal to 0, the variable `CurrPaletteIndex`

is derived as follows:

```
if( CurrPaletteIndex >= adjustedRefPaletteIndex )
    CurrPaletteIndex++
```

(180)

palette_escape_val specifies the quantized escape coded sample value for a component.

The variable `PaletteEscapeVal[ cIdx ][ xC ][ yC ]` specifies the escape value of a sample for which `PaletteIndexMap[ xC ][ yC ]` is equal to `MaxPaletteIndex` and `palette_escape_val_present_flag` is equal to 1. The array index `cIdx` specifies the colour component.

It is a requirement of bitstream conformance that `PaletteEscapeVal[ cIdx ][ xC ][ yC ]` shall be in the range of 0 to $(1 \ll \text{BitDepth}) - 1$, inclusive.

Tham khảo đến Bảng 4 và Bảng 5 ở trên, trong trường hợp mà chế độ bảng màu được áp dụng cho khối hiện tại (nghĩa là, đơn vị lập mã hiện tại), cú pháp lập mã bảng màu (ví dụ, `palette_coding()`) như trong Bảng 4 ở trên có thể được phân tích cú pháp/được báo hiệu.

Ví dụ, bảng của bảng màu có thể tạo cấu hình dựa trên thông tin mục nhập bảng màu. Thông tin mục nhập bảng màu có thể gồm các phần tử cú pháp, chẳng hạn như `palette_predictor_run`, `num_signalled_palette_entries`, và `new_palette_entries`.

Ngoài ra, bản đồ chỉ số bảng màu cho khối hiện tại có thể tạo cấu hình dựa trên thông tin chỉ số bảng màu. Thông tin chỉ số bảng màu có thể gồm các phần tử cú pháp, chẳng hạn như `num_palette_indices_minus1`, `palette_idx_idc`, và `palette_transpose_flag`. Dựa trên thông tin chỉ số bảng màu như được mô tả ở trên, bản đồ chỉ số bảng màu (ví dụ, `PaletteIndexMap`) có thể được tạo cấu hình bằng việc dẫn xuất chỉ số bảng màu (ví dụ, `PaletteIndexIdc`) cho các mẫu trong khối hiện tại trong khi lưu thông theo hướng quét qua (hướng thẳng hoặc hướng ngang).

Ngoài ra, dựa trên bản đồ chỉ số bảng màu, giá trị mẫu cho mục nhập bảng màu trong bảng của bảng màu có thể được dẫn xuất, và các mẫu được xây dựng lại của khối

hiện tại có thể được tạo ra dựa trên giá trị mẫu được ánh xạ trên mục nhập bảng màu.

Ngoài ra, trong trường hợp mà mẫu có giá trị thoát có mặt (nghĩa là, trong trường hợp mà giá trị của `palette_escape_val_present_flag` là 1) trong khối hiện tại, giá trị thoát cho khối hiện tại có thể được dẫn xuất dựa trên thông tin thoát. Thông tin thoát có thể gồm các phần tử cú pháp, chẳng hạn như `palette_escape_val_present_flag` và `palette_escape_val`. Ví dụ, dựa trên thông tin giá trị thoát được lượng tử hóa (ví dụ, `palette_escape_val`), giá trị thoát cho mẫu được lập mã thoát trong khối hiện tại có thể được dẫn xuất. Các mẫu được xây dựng lại của khối hiện tại có thể được tạo ra dựa trên giá trị thoát.

Trong khi đó, trong quy trình mã hóa/giải mã, kỹ thuật điều biến được lập mã xung chên lệch khối hoặc điều biến mã xung delta dựa trên khối (BDPCM) có thể được sử dụng. BDPCM có thể được gọi tên là việc điều biến mã xung delta dựa trên khối phần dư (residual block-based delta pulse code modulation, RDPCM) được lượng tử hóa.

Trong trường hợp mà khối được dự đoán bằng việc áp dụng BDPCM, các mẫu được xây dựng lại có thể được sử dụng để dự đoán các hàng hoặc các cột của khối từng dòng một. Trong trường hợp này, mẫu tham chiếu được sử dụng có thể là mẫu không được lọc. Hướng BDPCM có thể thể hiện xem việc dự đoán hướng thẳng hay hướng ngang đã được sử dụng. Nghĩa là, trong trường hợp mà BDPCM được áp dụng, việc dự đoán thẳng hoặc việc dự đoán ngang có thể được lựa chọn làm hướng BDPCM, và việc dự đoán có thể được thực hiện trong hướng BDPCM. Lỗi dự đoán có thể được lượng tử hóa trong miền không gian, và mẫu có thể được xây dựng lại bằng việc cộng lỗi dự đoán được khử lượng tử hóa cho việc dự đoán (nghĩa là, mẫu dự đoán). Lỗi dự đoán có thể nghĩa là phần dư. Miền phần dư được lượng tử hóa BDPCM có thể được đề xuất như sự thay thế của BDPCM, và hướng dự đoán hoặc việc báo hiệu có thể bằng với BDPCM được áp dụng cho miền không gian. Nghĩa là, phần dư có thể được xây dựng lại thông qua việc khử lượng tử hóa sau khi các hệ số lượng tử hóa chính chúng được xây dựng như việc điều biến mã xung delta (DPCM) thông qua miền phần dư được lượng tử hóa

BDPCM. Theo đó, miền phần dư được lượng tử hóa BDPCM có thể được sử dụng như có nghĩa rằng đầu lập mã phần dư áp dụng DPCM. Miền phần dư được lượng tử hóa được sử dụng sau đây được thu nhận bằng việc lượng tử hóa phần dư được dẫn xuất dựa trên việc dự đoán mà không được biến đổi, và có nghĩa là miền cho mẫu phần dư được lượng tử hóa. Ví dụ, miền phần dư được lượng tử hóa có thể gồm phần dư được lượng tử hóa (hoặc hệ số phần dư được lượng tử hóa) mà cho nó việc bỏ qua biến đổi được áp dụng, nghĩa là, việc biến đổi được bỏ qua đối với mẫu phần dư, nhưng việc lượng tử hóa được áp dụng cho nó. Ngoài ra, ví dụ, miền phần dư được lượng tử hóa có thể gồm hệ số biến đổi được lượng tử hóa.

Như được mô tả ở trên, BDPCM có thể được áp dụng cho miền phần dư được lượng tử hóa, miền phần dư được lượng tử hóa có thể gồm phần dư được lượng tử hóa (hoặc hệ số phần dư được lượng tử hóa), và trong trường hợp này, việc bỏ qua biến đổi có thể được áp dụng đối với phần dư. Nghĩa là, trong trường hợp mà BDPCM được áp dụng, việc biến đổi có thể được bỏ qua và việc lượng tử hóa có thể được áp dụng đối với mẫu phần dư. Ngoài ra, miền phần dư được lượng tử hóa có thể gồm hệ số biến đổi được lượng tử hóa. Cờ biểu diễn xem BDPCM có thể được ứng dụng hay không có thể được báo hiệu ở mức tần số (SPS), và cờ như vậy có thể được báo hiệu chỉ trong trường hợp được báo hiệu rằng chế độ bỏ qua biến đổi là có thể ở SPS. Cờ có thể được gọi là cờ được cho phép BDPCM hoặc cờ được cho phép SPS BDPCM.

Trong trường hợp áp dụng BDPCM, việc dự đoán nội có thể được thực hiện đối với toàn bộ các khối bởi bản sao mẫu theo hướng dự đoán (ví dụ, dự đoán dọc hoặc dự đoán ngang) tương tự với hướng dự đoán nội. Phần dư mà là giá trị chênh lệch giữa các khối gốc và khối dự đoán có thể được lượng tử hóa thông qua việc bỏ qua việc biến đổi, và giá trị delta, nghĩa là, giá trị chênh lệch giữa phần dư được lượng tử hóa và bộ dự đoán theo hướng ngang hoặc dọc (nghĩa là, phần dư được lượng tử hóa theo hướng ngang hoặc dọc), có thể được lập mã.

Nếu BDPCM là có thể ứng dụng, kích thước CU có thể bằng với hoặc nhỏ hơn

MaxTsSize (kích thước khối bỏ qua biến đổi tối đa) cho mẫu độ chói, và trong trường hợp mà CU được lập mã thông qua việc dự đoán nội, thông tin cờ có thể được truyền ở mức CU. Thông tin cờ có thể được gọi là cờ BDPCM. Ở đây, MaxTsSize có thể nghĩa là kích thước khối tối đa cho chế độ bỏ qua biến đổi để được cho phép. Thông tin cờ có thể chỉ ra xem việc lập mã nội điển hình được áp dụng hay BDPCM được áp dụng. Nếu BDPCM được áp dụng, cờ hướng dự đoán BDPCM chỉ ra xem hướng dự đoán là việc dự đoán ngang hay việc dự đoán thẳng có thể được truyền. Cờ hướng dự đoán BDPCM có thể được gọi là cờ hướng BDPCM. Sau đó, khối có thể được dự đoán thông qua quy trình dự đoán nội ngang hoặc dọc điển hình sử dụng mẫu tham chiếu không được lọc. Ngoài ra, phần dư có thể được lượng tử hóa, và giá trị chênh lệch giữa phần dư được lượng tử hóa và bộ dự đoán, ví dụ, giữa các phần dư đã được lượng tử hóa trong các vị trí xung quanh theo hướng ngang hoặc dọc theo hướng dự đoán BDPCM có thể được lập mã.

Trong khi đó, như được mô tả ở trên, thông tin (phần tử cú pháp) trong bảng cú pháp được bộc lộ trong sáng chế có thể được gồm trong thông tin hình ảnh/video, và có thể tạo cấu hình/được mã hóa bởi thiết bị mã hóa để được chuyển tới thiết bị giải mã dưới dạng luồng bit. Thiết bị giải mã có thể phân tích cú pháp/giải mã thông tin (phần tử cú pháp) trong bảng cú pháp tương ứng. Thiết bị giải mã có thể thực hiện quy trình giải mã (dự đoán, quy trình phần dư (dựa trên việc bỏ qua biến đổi), BDPCM, và việc lập mã bảng màu) cho khối hiện tại dựa trên thông tin được giải mã.

Sau đây, trong sáng chế, sơ đồ hiệu quả cho việc phân tích cú pháp/báo hiệu phần tử cú pháp có sự phụ thuộc được đề xuất đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và/hoặc phần tử cú pháp mức cao liên quan đến việc lập mã bảng màu. Nghĩa là, theo một phương án của sáng chế, trong suốt việc lập mã video/hình ảnh, việc liệu có thực hiện việc lập mã hay không có thể được phân loại theo sự phụ thuộc và sự không phụ thuộc của thông tin mà được sử dụng hoàn toàn cần thiết hoặc bổ sung trong việc thực hiện việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu,

và do đó việc lập mã hiệu quả có thể được thực hiện.

Trong việc lập mã video, việc chuyển dịch công cụ lập mã có thể được xác định trong cú pháp mức cao (high-level syntax, HLS) cụ thể. Trong trường hợp VVC trong kỹ thuật liên quan, thông tin cờ về các công cụ lập mã tương ứng có thể được xác định trong SPS. Ngoài ra, trong VVC, việc tiêu chuẩn hóa đang được tiến hành theo hướng có sự độc lập giữa các tập cú pháp mức cao tương ứng (ví dụ, tập thông số video (VPS), tập thông số trình tự (SPS), tập thông số ảnh (PPS), tập thông số thích ứng (APS), tập thông số giải mã (DPS), và phần đầu lát). Theo đó, trong tập cú pháp mức cao trong đó cờ của công cụ lập mã có mặt, nhiều phần tử cú pháp có sự phụ thuộc có mặt. Trong (các) phương án của sáng chế, phương pháp để phân tích cú pháp/báo hiệu phần tử cú pháp mức cao có sự phụ thuộc theo việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được đề xuất.

Như một phương án, sáng chế đề xuất phương pháp lưu các bit được truyền bằng việc cho phép phần tử cú pháp có sự phụ thuộc xác định xem có thực hiện việc phân tích cú pháp/báo hiệu theo điều kiện phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi hay không. Như một ví dụ, phương pháp phân tích cú pháp phần tử cú pháp mức cao có sự phụ thuộc phụ thuộc vào việc liệu có sử dụng việc bỏ qua biến đổi bởi cờ (được cho phép) bỏ qua biến đổi hay không được đề xuất.

Ví dụ, do các phần tử cú pháp phụ thuộc vào việc lập mã dựa trên việc bỏ qua biến đổi, có cờ (được cho phép) bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`), và thông tin về việc liệu có áp dụng BDPCM hay không (ví dụ, `sps_bdpcm_enabled_flag`). Như một ví dụ, nếu giá trị của cờ (được cho phép) bỏ qua biến đổi được xác định là 1, cờ liên quan hoặc các phần tử cú pháp thông tin nên cần được truyền, trong khi nếu giá trị của cờ (được cho phép) bỏ qua biến đổi được xác định là 0, các phần tử cú pháp ngoại trừ cờ (được cho phép) bỏ qua biến đổi phần tử cú pháp có thể không được truyền.

Nghĩa là, phương pháp truyền phần tử cú pháp mức cao được đề xuất, mà phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không, chẳng hạn như thông tin thông số lượng tử hóa tối thiểu cho khối bỏ qua biến đổi và việc liệu có áp dụng BDPCM hay không, trong suốt việc bỏ qua biến đổi theo giá trị của cờ (được cho phép) bỏ qua biến đổi trong cú pháp mức cao HLS (ví dụ, VPS, SPS, PPS, APS, DPS, và phần đầu lát). Ngoài ra, phương pháp được đề xuất không bị giới hạn ở các phần tử cú pháp được đề cập trong phương án này, nhưng có thể gồm tất cả các phần tử cú pháp mức cao được xác định trong tập cú pháp mức cao mà có sự phụ thuộc phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không và gồm cờ (được cho phép) bỏ qua biến đổi.

Như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi có thể được xác định trong tập cú pháp mức cao, và có thể được xác định trong tập thông số trình tự (SPS) như trong phương án trong Bảng 6 dưới đây.

[Bảng 6]

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>(...)</code>	
<code>sps_transform_skip_enabled_flag</code>	<code>u(1)</code>
<code>if(sps_transform_skip_enabled_flag) {</code>	
<code>sps_bdpcm_enabled_flag</code>	<code>u(1)</code>
<code>min_qp_prime_ts_minus4</code>	<code>ue(v)</code>
<code>}</code>	
<code>(...)</code>	
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Ngoài ra, ví dụ, ngữ nghĩa của phần tử cú pháp cho phương án được mô tả ở trên trong số các phần tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 7 dưới đây.

[Bảng 7]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$$

Tham khảo đến Bảng 6 và Bảng 7 ở trên, các phần tử cú pháp liên quan tới việc bỏ qua biến đổi có thể được xác định trong SPS, và có thể gồm các phần tử cú pháp của **sps_transform_skip_enabled_flag**, **sps_bdpcm_enabled_flag**, và **min_qp_prime_ts_minus4**.

Phần tử cú pháp **sps_transform_skip_enabled_flag** có thể thể hiện xem liệu việc bỏ qua biến đổi có được cho phép hay không dựa trên việc liệu giá trị của nó là 0 hay 1. Ví dụ, nếu giá trị của **sps_transform_skip_enabled_flag** là 1, nó có thể thể hiện rằng việc bỏ qua biến đổi được cho phép, và trong trường hợp này, **transform_skip_flag** có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị biến đổi. Ở đây, phần tử cú pháp **transform_skip_flag** có thể thể hiện xem việc biến đổi có thể được áp dụng cho khối biến đổi được kết hợp tương ứng hay không. Nếu giá trị của **sps_transform_skip_enabled_flag** là 0, nó có thể thể hiện rằng việc bỏ qua biến đổi là không được cho phép, và trong trường hợp này, **transform_skip_flag** có thể không được

phân tích cú pháp/được báo hiệu trong cú pháp đơn vị biến đổi. Nói cách khác, dựa trên cờ được cho phép bỏ qua biến đổi `sps_transform_skip_enabled_flag`, có thể được biểu diễn xem `transform_skip_flag` có mặt trong cú pháp đơn vị biến đổi hay không.

Phần tử cú pháp `sps_bdpcm_enabled_flag` có thể thể hiện xem BDPCM có được cho phép hay không dựa trên việc liệu giá trị của nó là 0 hay 1. Ví dụ, nếu giá trị của `sps_bdpcm_enabled_flag` là 1, nó có thể thể hiện rằng BDPCM được cho phép, và trong trường hợp này, `intra_bdpcm_flag` (hoặc `intra_bdpcm_luma_flag` và `intra_bdpcm_chroma_flag`) có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã cho đơn vị lập mã nội. Ở đây, `intra_bdpcm_flag` phần tử cú pháp có thể thể hiện xem BDPCM có được áp dụng cho khối lập mã hiện tại hay không. Nếu giá trị của `sps_bdpcm_enabled_flag` là 0, nó có thể thể hiện rằng BDPCM là không được cho phép, và trong trường hợp này, `intra_bdpcm_flag` (hoặc `intra_bdpcm_luma_flag` và `intra_bdpcm_chroma_flag`) có thể không được phân tích cú pháp/được báo hiệu trong cú pháp đơn vị lập mã cho đơn vị lập mã nội. Nói cách khác, có thể được biểu diễn xem `intra_bdpcm_flag` (hoặc `intra_bdpcm_luma_flag` và `intra_bdpcm_chroma_flag`) có mặt trong cú pháp đơn vị lập mã hay không dựa trên cờ được cho phép BDPCM `sps_bdpcm_enabled_flag`.

Phần tử cú pháp `min_qp_prime_ts_minus4` có thể thể hiện thông số lượng tử hóa được cho phép tối thiểu được cho phép cho chế độ bỏ qua biến đổi. Ví dụ, dựa trên phần tử cú pháp `min_qp_prime_ts_minus4`, giá trị thông số lượng tử hóa tối thiểu (ví dụ, `QpPrimeTsMin`) trong chế độ bỏ qua biến đổi có thể được dẫn xuất. Dựa trên thông số lượng tử hóa tối thiểu trong chế độ bỏ qua biến đổi, thông số lượng tử hóa được sử dụng trong quy trình định cỡ (quy trình khử lượng tử hóa) có thể được dẫn xuất. Ngoài ra, hệ số biến đổi được định cỡ (hệ số biến đổi được khử lượng tử hóa) có thể được dẫn xuất bằng việc thực hiện quy trình định cỡ (quy trình khử lượng tử hóa) cho khối hiện tại dựa trên thông số lượng tử hóa, và dựa trên điều này, mẫu phần dư của khối hiện tại có thể được dẫn xuất.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi. Ví dụ, như được bộc lộ trong Bảng 6 và Bảng 7 ở trên, dựa trên giá trị của còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong SPS, phần tử cú pháp `min_qp_prime_ts_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu cho khối bỏ qua biến đổi trong chế độ bỏ qua biến đổi và phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_minus4` và phần tử cú pháp `sps_bdpcm_enabled_flag` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, phần tử cú pháp `min_qp_prime_ts_minus4` và phần tử cú pháp `sps_bdpcm_enabled_flag` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, như một phương án, sáng chế đề xuất phương pháp lưu các bit được truyền bằng việc cho phép phần tử cú pháp có sự phụ thuộc xác định xem có thực hiện việc phân tích cú pháp/báo hiệu theo điều kiện phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi hay không. Như một ví dụ, được đề xuất là phương pháp phân tích cú pháp phần tử cú pháp mức cao có sự phụ thuộc phụ thuộc vào việc liệu việc bỏ qua biến đổi bởi còn (được cho phép) bỏ qua biến đổi có được sử dụng hay không.

Ví dụ, như các phần tử cú pháp phụ thuộc vào việc lập mã dựa trên việc bỏ qua biến đổi, có còn (được cho phép) bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin về kích thước ứng dụng bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`), và thông tin về việc

liệu BDPCM có được áp dụng hay không (ví dụ, `sps_bdpcm_enabled_flag`). Như một ví dụ, nếu giá trị của cờ (được cho phép) bỏ qua biến đổi được xác định là 1, cờ liên quan hoặc các phần tử cú pháp thông tin nên cần được truyền, trong khi nếu giá trị của cờ (được cho phép) bỏ qua biến đổi được xác định là 0, các phần tử cú pháp ngoại trừ cờ (được cho phép) bỏ qua biến đổi phần tử cú pháp có thể không được truyền.

Nghĩa là, phương pháp truyền phần tử cú pháp mức cao được đề xuất, mà phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không, chẳng hạn như thông tin về kích thước tối đa của ứng dụng bỏ qua biến đổi, thông tin thông số lượng tử hóa tối thiểu, và việc liệu có áp dụng BDPCM hay không trong suốt việc bỏ qua biến đổi, theo giá trị của cờ (được cho phép) bỏ qua biến đổi trong cú pháp mức cao HLS (ví dụ, VPS, SPS, PPS, APS, DPS, và phần đầu lát). Ngoài ra, phương pháp được đề xuất không bị giới hạn ở các phần tử cú pháp được đề cập trong phương án này, nhưng có thể gồm tất cả các phần tử cú pháp mức cao được xác định trong tập cú pháp mức cao mà có sự phụ thuộc phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không và gồm cờ (được cho phép) bỏ qua biến đổi.

Như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi có thể được xác định trong tập cú pháp mức cao, và có thể được xác định trong tập thông số trình tự (SPS) như trong phương án của Bảng 8 dưới đây. Tuy nhiên, thông tin kích thước khối tối đa cho việc bỏ qua biến đổi mà được xác định trong tập thông số ảnh (PPS) trong kỹ thuật liên quan có thể được xác định mới trong SPS để tránh sự phụ thuộc giữa các HLS, và điều này có thể được trình bày như trong Bảng 8 dưới đây.

[Bảng 8]

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag) {	
sps_bdpcm_enabled_flag	u(1)
min_qp_prime_ts_minus4	ue(v)
log2_transform_skip_max_size_minus2	ue(v)
}	
(...)	
rbp_trailing_bits()	
}	

Ngoài ra, ví dụ, ngữ nghĩa của phần tử cú pháp cho phương án được mô tả ở trên trong số các phần tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 9 dưới đây.

[Bảng 9]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$$

log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.

When not present, the value of **log2_transform_skip_max_size_minus2** is inferred to be equal to 0.

The variable **MaxTsSize** is set equal to $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.

Tham khảo đến Bảng 8 và Bảng 9 ở trên, các phần tử cú pháp liên quan tới việc bỏ qua biến đổi có thể được xác định trong SPS, và có thể gồm các phần tử cú pháp của **sps_transform_skip_enabled_flag**, **sps_bdpcm_enabled_flag**, **min_qp_prime_ts_minus4**, và **log2_transform_skip_max_size_minus2**.

Ở đây, do các phần tử cú pháp của **sps_transform_skip_enabled_flag**, **sps_bdpcm_enabled_flag**, và **min_qp_prime_ts_minus4** đã được giải thích chi tiết trong

Bảng 6 và Bảng 7 ở trên, trong phương án này, phần giải thích chi tiết của chúng sẽ được bỏ qua để thuận tiện cho việc giải thích.

Phần tử cú pháp `log2_transform_skip_max_size_minus2` có thể thể hiện kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi. Trong trường hợp này, phần tử cú pháp `log2_transform_skip_max_size_minus2` có thể là trong khoảng từ 0 đến 3. Ví dụ, như được bộc lộ trong Bảng 9 ở trên, kích thước khối tối đa (ví dụ, `MaxTsSize`) được sử dụng trong chế độ bỏ qua biến đổi có thể được dẫn xuất dựa trên việc tính toán chẳng hạn như $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi. Ví dụ, như được bộc lộ trong Bảng 8 và Bảng 9 ở trên, trong SPS, phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), phần tử cú pháp `min_qp_prime_ts_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu cho khối bỏ qua biến đổi trong chế độ bỏ qua biến đổi, và phần tử cú pháp `log2_transform_skip_max_size_minus2` biểu diễn kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, các phần tử cú pháp của `sps_bdpcm_enabled_flag`, `min_qp_prime_ts_minus4`, và `log2_transform_skip_max_size_minus2` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, các phần tử cú pháp của `sps_bdpcm_enabled_flag`, `min_qp_prime_ts_minus4`, và `log2_transform_skip_max_size_minus2` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, như một phương án, sáng chế đề xuất phương pháp lưu các bit được truyền bằng việc cho phép phần tử cú pháp có sự phụ thuộc để xác định xem có thực hiện việc phân tích cú pháp/báo hiệu hay không theo điều kiện phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và phần tử cú pháp mức cao liên quan đến việc lập mã bảng màu. Như một ví dụ, được đề xuất là phương pháp phân tích cú pháp phần tử cú pháp mức cao có sự phụ thuộc bởi cờ (được cho phép) bỏ qua biến đổi và/hoặc cờ (được cho phép) lập mã bảng màu.

Ví dụ, như các phần tử cú pháp phụ thuộc vào việc lập mã dựa trên việc bỏ qua biến đổi, có cờ (được cho phép) bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin về kích thước ứng dụng bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`), và thông tin về việc liệu BDPCM có được áp dụng hay không (ví dụ, `sps_bdpcm_enabled_flag`). Ngoài ra, như được mô tả ở trên, do giá trị thoát cũng không được thay đổi trong suốt việc lập mã bảng màu, thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi có thể được sử dụng trong việc thực hiện việc lượng tử hóa. Theo đó, cờ (được cho phép) lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) cho việc lập mã dựa trên chế độ bảng màu và thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) trong suốt việc bỏ qua biến đổi có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 1, cờ liên quan hoặc thông tin các phần tử cú pháp nên cần được truyền, trong khi nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 0, các phần tử cú pháp ngoại trừ phần tử cú pháp cờ tương ứng có thể không được truyền.

Nghĩa là, phương pháp truyền phần tử cú pháp mức cao được đề xuất, mà phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không, chẳng hạn như thông tin về kích thước tối đa của ứng dụng bỏ qua biến đổi, thông tin thông số lượng tử hóa

tối thiểu trong suốt việc bỏ qua biến đổi, và việc liệu có áp dụng BDPCM hay không, hoặc việc liệu có thực hiện việc lập mã bảng màu theo giá trị của cờ (được cho phép) bỏ qua biến đổi và/hoặc giá trị của cờ (được cho phép) lập mã bảng màu trong cú pháp mức cao (ví dụ, VPS, SPS, PPS, APS, DPS, và phần đầu lát) hay không.

Ví dụ, (i) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu được xác định là 1, các phần tử cú pháp tương ứng với tập hợp của các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (ii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 1, và cờ (được cho phép) lập mã bảng màu là 0, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi có thể được phân tích cú pháp. (iii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 0, và cờ (được cho phép) lập mã bảng màu là 1, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (iv) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu là 0, các phần tử cú pháp mức cao khác có sự phụ thuộc vào hai công cụ lập mã có thể không được phân tích cú pháp.

Thứ tự phân tích cú pháp của các phần tử cú pháp được đề cập trong phương án này không được giới hạn cụ thể, và trong trường hợp mà việc liệu có thực hiện việc phân tích cú pháp hay không được xác định theo sự phụ thuộc giữa các phần tử cú pháp, chúng được coi là trùng hợp với nhau. Ngoài ra, phương pháp được đề xuất không bị giới hạn ở các phần tử cú pháp được đề cập trong phương án này, mà có thể có sự phụ thuộc phụ thuộc vào việc liệu việc bỏ qua biến đổi hay việc lập mã bảng màu được thực hiện, và có thể gồm tất cả các phần tử cú pháp mức cao được xác định trong tập cú pháp mức cao gồm cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu.

Như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi và/hoặc việc lập mã dựa trên chế độ bảng màu có thể được xác định

trong tập cú pháp mức cao, và có thể được xác định trong tập thông số trình tự (SPS) như trong phương án của Bảng 10 dưới đây.

[Bảng 10]

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
(...)	
}	

Ngoài ra, ví dụ, ngữ nghĩa của phần tử cú pháp cho phương án được mô tả ở trên trong số các phần tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 11 dưới đây.

[Bảng 11]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$$

sps_palette_enabled_flag equal to 1 specifies that **pred_mode_plt_flag** may be present in the coding unit syntax. **sps_palette_enabled_flag** equal to 0 specifies that **pred_mode_plt_flag** is not present in the coding unit syntax. When **sps_palette_enabled_flag** is not present, it is inferred to be equal to 0.

7

Tham khảo đến Bảng 10 và Bảng 11 ở trên, trong SPS, các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được xác định, và có thể gồm các phần tử cú pháp của **sps_transform_skip_enabled_flag**, **sps_bdpcm_enabled_flag**, **sps_palette_enabled_flag**, và **min_qp_prime_ts_minus4**.

Ở đây, do các phần tử cú pháp của **sps_transform_skip_enabled_flag**, **sps_bdpcm_enabled_flag**, và **min_qp_prime_ts_minus4** đã được giải thích chi tiết trong Bảng 6 đến Bảng 9 ở trên, trong phương án này, để thuận tiện cho việc giải thích, phần giải thích chi tiết của chúng sẽ được bỏ qua.

Phần tử cú pháp `sps_palette_enabled_flag` có thể thể hiện xem việc lập mã bảng màu (nghĩa là, chế độ dự đoán bảng màu) có được cho phép hay không dựa trên việc liệu giá trị của nó là 0 hay 1. Ví dụ, nếu giá trị của `sps_palette_enabled_flag` là 1, nó có thể thể hiện rằng việc lập mã bảng màu được cho phép, và trong trường hợp này, `pred_mode_plt_flag` có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã. Ở đây, phần tử cú pháp `pred_mode_plt_flag` có thể thể hiện xem chế độ bảng màu có thể được sử dụng cho đơn vị lập mã hiện tại hay không. Nếu giá trị của `sps_palette_enabled_flag` là 0, nó có thể thể hiện rằng việc lập mã bảng màu là không được cho phép, và trong trường hợp này, `pred_mode_plt_flag` có thể không được phân tích cú pháp/được báo hiệu trong cú pháp đơn vị lập mã. Nói cách khác, dựa trên cờ được cho phép lập mã bảng màu `sps_palette_enabled_flag`, có thể thể hiện xem `pred_mode_plt_flag` có mặt trong cú pháp đơn vị lập mã hay không.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 10 và Bảng 11 ở trên, trong SPS, phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, phần tử cú pháp `sps_bdpcm_enabled_flag` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, phần tử cú pháp `sps_bdpcm_enabled_flag` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, điều kiện phụ thuộc có thể được xác định trong mối quan hệ với phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến

đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 10 và Bảng 11 ở trên, trong SPS, phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể được phân tích cú pháp/được báo hiệu dựa trên phần tử cú pháp `chroma_format_idc`. Như một ví dụ, nếu giá trị của phần tử cú pháp `chroma_format_idc` là 3, phần tử cú pháp `sps_palette_enabled_flag` có thể được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 10 và Bảng 11 ở trên, trong SPS, phần tử cú pháp `min_qp_prime_ts_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu cho chế độ bỏ qua biến đổi có thể có sự phụ thuộc dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`). Như một ví dụ, trong trường hợp mà giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_minus4` có thể được phân tích cú pháp/được báo hiệu.

Ngoài ra, như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi và/hoặc việc lập mã dựa trên chế độ bảng màu có thể được xác định trong tập cú pháp mức cao, và như trong phương án của Bảng 12 dưới đây, có thể được xác định trong tập thông số trình tự (SPS). Tuy nhiên, thông tin kích thước khối tối đa cho việc bỏ qua biến đổi mà được xác định trong tập thông số ảnh (PPS) trong kỹ thuật liên quan có thể được xác định mới trong SPS để tránh sự phụ thuộc giữa các HLS, và điều này có thể được trình bày như trong Bảng 12 dưới đây.

[Bảng 12]

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag) {	
sps_bdpcm_enabled_flag	u(1)
log2_transform_skip_max_size_minus2	ue(v)
}	
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
(...)	
}	

Ngoài ra, ví dụ, ngữ nghĩa của phần tử cú pháp cho phương án được mô tả ở trên trong số các phần tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 13 dưới đây.

[Bảng 13]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$$

log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.

When not present, the value of **log2_transform_skip_max_size_minus2** is inferred to be equal to 0.

The variable **MaxTsSize** is set equal to $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.

sps_palette_enabled_flag equal to 1 specifies that **pred_mode_plt_flag** may be present in the coding unit syntax. **sps_palette_enabled_flag** equal to 0 specifies that **pred_mode_plt_flag** is not present in the coding unit syntax. When **sps_palette_enabled_flag** is not present, it is inferred to be equal to 0.

Tham khảo đến Bảng 12 và Bảng 13 ở trên, trong SPS, các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được xác định, và có thể gồm các phần tử cú pháp của **sps_transform_skip_enabled_flag**,

`sps_bdpcm_enabled_flag`, `log2_transform_skip_max_size_minus2`,
`sps_palette_enabled_flag`, và `min_qp_prime_ts_minus4`.

Ở đây, do các phần tử cú pháp của `sps_transform_skip_enabled_flag`, `sps_bdpcm_enabled_flag`, `log2_transform_skip_max_size_minus2`, `sps_palette_enabled_flag`, và `min_qp_prime_ts_minus4` đã được mô tả chi tiết trong Bảng 6 đến Bảng 11 ở trên, trong phương án này, để thuận tiện cho việc giải thích, phần giải thích chi tiết của chúng sẽ được bỏ qua.

Như được bộc lộ trong phương án của Bảng 12 và Bảng 13 ở trên, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 12 và Bảng 13 ở trên, trong SPS, phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), và phần tử cú pháp `log2_transform_skip_max_size_minus2` biểu diễn kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, các phần tử cú pháp của `sps_bdpcm_enabled_flag` và `log2_transform_skip_max_size_minus2` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, các phần tử cú pháp của `sps_bdpcm_enabled_flag` và `log2_transform_skip_max_size_minus2` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới

việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 12 và Bảng 13 ở trên, phần tử cú pháp `min_qp_prime_ts_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu cho chế độ bỏ qua biến đổi dựa trên các giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể có sự phụ thuộc trong SPS. Như một ví dụ, trong trường hợp mà giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_minus4` có thể được phân tích cú pháp/được báo hiệu.

Trong khi đó, nguồn hoặc ảnh/hình ảnh được lập mã có thể gồm mảng thành phần độ chói, và trong một số trường hợp, có thể còn gồm hai mảng thành phần sắc độ (cb, cr), nghĩa là, một điểm ảnh của ảnh/hình ảnh có thể gồm mẫu độ chói và mẫu sắc độ (cb, cr).

Định dạng màu sắc có thể thể hiện định dạng cấu hình của thành phần độ chói và thành phần sắc độ (cb, cr), và có thể được gọi là thành phần sắc độ. Định dạng màu sắc (hoặc định dạng sắc độ) có thể được xác định trước, hoặc có thể được báo hiệu thích ứng. Ví dụ, định dạng sắc độ có thể được báo hiệu dựa trên ít nhất một trong số `chroma_format_idc` và `separate_colour_plane_flag` như trong Bảng 14 dưới đây.

[Bảng 14]

<code>chroma_format_idc</code>	<code>separate_colour_plane_flag</code>	Chroma format	SubWidthC	SubHeightC
0	0	Monochrome	1	1
1	0	4:2:0	2	2
2	0	4:2:2	2	1
3	0	4:4:4	1	1
3	1	4:4:4	1	1

Tham khảo đến Bảng 14 ở trên, trong việc lấy mẫu đơn sắc, chỉ có một mảng mẫu mà được xem trên danh nghĩa là mảng độ chói.

Trong việc lấy mẫu 4:2:0, mỗi trong số hai mảng sắc độ có một nửa chiều cao

và một nửa chiều rộng của mảng độ chói.

Trong việc lấy mẫu 4:2:2, mỗi trong số hai mảng sắc độ có cùng chiều cao và một nửa chiều rộng của mảng độ chói.

Trong việc lấy mẫu 4:4:4, phần sau đây có thể được áp dụng theo giá trị của `separate_colour_plane_flag`.

- Nếu giá trị của `separate_colour_plane_flag` là 0, mỗi trong số hai mảng sắc độ có cùng chiều cao và cùng chiều rộng của mảng độ chói.

- Mặt khác, nếu giá trị của `separate_colour_plane_flag` là 1, ba dạng mặt phẳng màu sắc có thể được xử lý một cách tách biệt như các ảnh được lấy mẫu đơn sắc.

`SubWidthC` và `SubHeightC` có thể thể hiện tỷ lệ giữa mẫu độ chói và mẫu sắc độ. Ví dụ, nếu `chroma_format_idc` là 3, định dạng sắc độ là 4:4:4, và trong trường hợp này, trong trường hợp mà chiều rộng của khối mẫu độ chói là 16, chiều rộng của khối mẫu sắc độ tương ứng có thể là $16/\text{SubWidthC}$. Nói chung, cú pháp liên quan đến mẫu sắc độ và luồng bit có thể được phân tích cú pháp chỉ trong trường hợp mà loại mảng sắc độ (ví dụ, `chromaArrayType`) là khác 0.

Ngoài ra, như một phương án, sáng chế đề xuất phương pháp lưu các bit được truyền bằng việc cho phép phân tử cú pháp có sự phụ thuộc để xác định xem có thực hiện việc phân tích cú pháp/báo hiệu hay không theo điều kiện phụ thuộc đối với phân tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và phân tử cú pháp mức cao liên quan đến việc lập mã bảng màu. Như một ví dụ, phương pháp phân tích cú pháp phân tử cú pháp mức cao có sự phụ thuộc bởi cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu được đề xuất.

Ví dụ, như các phân tử cú pháp phụ thuộc vào việc lập mã dựa trên việc bỏ qua biến đổi, có cờ (được cho phép) bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin về kích thước của ứng dụng bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), thông tin thông số lượng tử hóa tối

thiếu trong suốt việc bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`), và thông tin về việc liệu có áp dụng BDPCM hay không (ví dụ, `sps_bdpcm_enabled_flag`). Ngoài ra, như được mô tả ở trên, do giá trị thoát cũng không được thay đổi trong suốt việc lập mã bảng màu, thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi có thể được sử dụng trong việc thực hiện việc lượng tử hóa.

Như được biểu diễn trong phương án được mô tả ở trên, nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 1, cờ liên quan hoặc thông tin các phần tử cú pháp nên cần được truyền, trong khi nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 0, các phần tử cú pháp ngoại trừ các phần tử cú pháp cờ tương ứng có thể không được truyền. Nghĩa là, phương pháp truyền phần tử cú pháp mức cao được đề xuất, mà phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không, chẳng hạn như thông tin thông số lượng tử hóa tối thiểu trong suốt việc bỏ qua biến đổi hoặc việc lập mã bảng màu, và việc liệu có áp dụng BDPCM hay không, theo giá trị của cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu, hoặc xem có thực hiện việc lập mã bảng màu trong cú pháp mức cao (ví dụ, VPS, SPS, PPS, APS, DPS, và phần đầu lát) hay không.

Ví dụ, (i) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu được xác định là 1, các phần tử cú pháp tương ứng với tập hợp của các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (ii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 1, và cờ (được cho phép) lập mã bảng màu là 0, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi có thể được phân tích cú pháp. (iii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 0, và cờ (được cho phép) lập mã bảng màu là 1, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (iv) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ

(được cho phép) lập mã bảng màu là 0, các phần tử cú pháp mức cao khác có sự phụ thuộc vào hai công cụ lập mã có thể không được phân tích cú pháp.

Thứ tự phân tích cú pháp của các phần tử cú pháp được đề cập trong phương án này không được giới hạn cụ thể, và trong trường hợp mà việc liệu có thực hiện việc phân tích cú pháp hay không được xác định theo sự phụ thuộc giữa các phần tử cú pháp, chúng được coi là trùng hợp với nhau. Ngoài ra, phương pháp được đề xuất không bị giới hạn ở các phần tử cú pháp được đề cập trong phương án này, mà có thể có sự phụ thuộc phụ thuộc vào việc liệu việc bỏ qua biến đổi hay việc lập mã bảng màu được thực hiện, và có thể gồm tất cả các phần tử cú pháp mức cao được xác định trong tập cú pháp mức cao gồm cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu.

Như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi và/hoặc việc lập mã dựa trên chế độ bảng màu có thể được xác định trong tập cú pháp mức cao, và có thể được xác định trong tập thông số trình tự (SPS) như trong phương án của Bảng 15 dưới đây.

[Bảng 15]

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_palette_enabled_flag)	
min_qp_prime_ts_chroma_minus4	ue(v)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_luma_minus4	ue(v)
(...)	
}	

Ngoài ra, ví dụ, ngữ nghĩa của phân tử cú pháp cho phương án được mô tả ở trên trong số các phân tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 16 dưới đây.

[Bảng 16]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

min_qp_prime_ts_luma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the luma component as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_luma_minus4}$$

min_qp_prime_ts_chroma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the chroma component as follows:

$$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_chroma_minus4}$$

sps_palette_enabled_flag equal to 1 specifies that **pred_mode_plt_flag** may be present in the coding unit syntax. **sps_palette_enabled_flag** equal to 0 specifies that **pred_mode_plt_flag** is not present in the coding unit syntax. When **sps_palette_enabled_flag** is not present, it is inferred to be equal to 0.

Tham khảo đến Bảng 15 và Bảng 16 ở trên, trong SPS, các phân tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được xác định, và có

thể gồm các phần tử cú pháp của `sps_transform_skip_enabled_flag`, `sps_bdpcm_enabled_flag`, `sps_palette_enabled_flag`, `min_qp_prime_ts_luma_minus4`, và `min_qp_prime_ts_chroma_minus4`.

Ở đây, do các phần tử cú pháp của `sps_transform_skip_enabled_flag`, `sps_bdpcm_enabled_flag`, và `sps_palette_enabled_flag` đã được giải thích chi tiết trong Bảng 6 đến Bảng 11 ở trên, trong phương án này, để thuận tiện cho việc giải thích, phần giải thích chi tiết của chúng sẽ được bỏ qua.

Như được bộc lộ trong phương án của Bảng 15 và Bảng 16 ở trên, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 15 và Bảng 16 ở trên, trong SPS, phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, phần tử cú pháp `sps_bdpcm_enabled_flag` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, phần tử cú pháp `sps_bdpcm_enabled_flag` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 15 và Bảng 16 ở trên, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu trong chế độ bỏ qua biến đổi cho thành phần sắc độ dựa trên giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể có

sự phụ thuộc trong SPS. Như một ví dụ, nếu giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 0, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 15 và Bảng 16 ở trên, phần tử cú pháp `min_qp_prime_ts_luma_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu trong chế độ bỏ qua biến đổi cho thành phần độ chói dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể có sự phụ thuộc trong SPS. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_luma_minus4` có thể được phân tích cú pháp/được báo hiệu.

Ngoài ra, như một phương án, sáng chế đề xuất phương pháp lưu các bit được truyền bằng việc cho phép phần tử cú pháp có sự phụ thuộc để xác định xem có thực hiện việc phân tích cú pháp/báo hiệu hay không theo điều kiện phụ thuộc đối với phần tử cú pháp mức cao liên quan đến việc bỏ qua biến đổi và phần tử cú pháp mức cao liên quan đến việc lập mã bảng màu. Như một ví dụ, phương pháp phân tích cú pháp phần tử cú pháp mức cao có sự phụ thuộc bởi cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu được đề xuất.

Ví dụ, như các phần tử cú pháp phụ thuộc vào việc lập mã dựa trên việc bỏ qua biến đổi, cờ (được cho phép) bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin về kích thước của ứng dụng bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), thông tin thông số lượng tử hóa tối thiểu trong suốt việc bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`), và thông tin về việc liệu có áp dụng BDPCM hay không (ví dụ, `sps_bdpcm_enabled_flag`). Ngoài ra, như được mô tả ở trên, do giá trị thoát cũng không được thay đổi trong suốt việc lập mã bảng màu, thông tin thông số lượng tử hóa tối thiểu cho việc bỏ qua biến đổi có thể được sử dụng trong việc thực hiện việc lượng tử hóa.

Như được biểu diễn trong phương án được mô tả ở trên, nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 1, cờ liên quan hoặc thông tin các phần tử cú pháp nên cần được truyền, trong khi nếu giá trị của cờ (được cho phép) bỏ qua biến đổi hoặc cờ (được cho phép) lập mã bảng màu được xác định là 0, các phần tử cú pháp ngoại trừ các phần tử cú pháp cờ tương ứng có thể không được truyền. Nghĩa là, phương pháp truyền phần tử cú pháp mức cao được đề xuất, mà phụ thuộc vào việc liệu có thực hiện việc bỏ qua biến đổi hay không, chẳng hạn như thông tin về kích thước tối đa của ứng dụng bỏ qua biến đổi, thông tin thông số lượng tử hóa tối thiểu trong suốt việc bỏ qua biến đổi hoặc việc lập mã bảng màu và việc liệu có áp dụng BDPCM hay không, theo giá trị của cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu, hoặc xem có thực hiện việc lập mã bảng màu trong cú pháp mức cao (ví dụ, VPS, SPS, PPS, APS, DPS, và phần đầu lát) hay không.

Ví dụ, (i) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu được xác định là 1, các phần tử cú pháp tương ứng với tập hợp của các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (ii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 1, và cờ (được cho phép)

lập mã bảng màu là 0, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) bỏ qua biến đổi có thể được phân tích cú pháp. (iii) Trong trường hợp mà cờ (được cho phép) bỏ qua biến đổi được xác định là 0, và cờ (được cho phép) lập mã bảng màu là 1, các phần tử cú pháp phụ thuộc vào cờ (được cho phép) lập mã bảng màu có thể được phân tích cú pháp. (iv) Trong trường hợp mà cả cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu là 0, các phần tử cú pháp mức cao khác có sự phụ thuộc vào hai công cụ lập mã có thể không được phân tích cú pháp.

Thứ tự phân tích cú pháp của các phần tử cú pháp được đề cập trong phương án này không được giới hạn cụ thể, và trong trường hợp mà việc liệu có thực hiện việc phân tích cú pháp hay không được xác định theo sự phụ thuộc giữa các phần tử cú pháp, chúng được coi là trùng hợp với nhau. Ngoài ra, phương pháp được đề xuất không bị giới hạn ở các phần tử cú pháp được đề cập trong phương án này, mà có thể có sự phụ thuộc phụ thuộc vào việc liệu việc bỏ qua biến đổi hay việc lập mã bảng màu được thực hiện, và có thể gồm tất cả các phần tử cú pháp mức cao được xác định trong tập cú pháp mức cao gồm cờ (được cho phép) bỏ qua biến đổi và cờ (được cho phép) lập mã bảng màu.

Như được mô tả ở trên, các phần tử cú pháp liên quan tới việc lập mã dựa trên việc bỏ qua biến đổi và/hoặc việc lập mã dựa trên chế độ bảng màu có thể được xác định trong tập cú pháp mức cao, và có thể được xác định trong tập thông số trình tự (SPS) như trong phương án của Bảng 17 dưới đây. Trong phương án này, thông tin về kích thước tối đa của việc bỏ qua biến đổi được xác định trong tập thông số ảnh (PPS) trong kỹ thuật liên quan có thể được xác định mới trong SPS để tránh sự phụ thuộc giữa các HLS, và phương pháp thực hiện việc phân tích cú pháp/báo hiệu dựa trên sự phụ thuộc của việc bỏ qua biến đổi và phần tử cú pháp liên quan đến việc lập mã bảng màu được sử dụng trước đó được đề xuất.

[Bảng 17]

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>(...)</code>	
<code>sps_transform_skip_enabled_flag</code>	<code>u(1)</code>
<code>if(sps_transform_skip_enabled_flag) {</code>	
<code>sps_bdpcm_enabled_flag</code>	<code>u(1)</code>
<code>log2_transform_skip_max_size_minus2</code>	<code>ue(v)</code>
<code>}</code>	
<code>(...)</code>	
<code>if(chroma_format_idc == 3)</code>	
<code>sps_palette_enabled_flag</code>	<code>u(1)</code>
<code>(...)</code>	
<code>if(sps_palette_enabled_flag)</code>	
<code>min_qp_prime_ts_chroma_minus4</code>	<code>ue(v)</code>
<code>(...)</code>	
<code>if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)</code>	
<code>min_qp_prime_ts_luma_minus4</code>	<code>ue(v)</code>
<code>(...)</code>	
<code>}</code>	

Ngoài ra, ví dụ, ngữ nghĩa của phần tử cú pháp cho phương án được mô tả ở trên trong số các phần tử cú pháp của cú pháp SPS có thể được trình bày như trong Bảng 18 dưới đây.

[Bảng 18]

sps_transform_skip_enabled_flag equal to 1 specifies that **transform_skip_flag** may be present in the transform unit syntax. **sps_transform_skip_enabled_flag** equal to 0 specifies that **transform_skip_flag** is not present in the transform unit syntax.

sps_bdpcm_enabled_flag equal to 1 specifies that **intra_bdpcm_flag** may be present in the coding unit syntax for intra coding units. **sps_bdpcm_enabled_flag** equal to 0 specifies that **intra_bdpcm_flag** is not present in the coding unit syntax for intra coding units. When not present, the value of **sps_bdpcm_enabled_flag** is inferred to be equal to 0.

log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.

When not present, the value of **log2_transform_skip_max_size_minus2** is inferred to be equal to 0.

The variable **MaxTsSize** is set equal to $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.

min_qp_prime_ts_luma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the luma component as follows:

$$\text{QpPrimeTsMin} = 4 + \text{min_qp_prime_ts_luma_minus4}$$

min_qp_prime_ts_chroma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the chroma component as follows:

$$\text{QpPrimeTsMin} = 4 + \text{min_qp_prime_ts_chroma_minus4}$$

`sps_palette_enabled_flag` equal to 1 specifies that `pred_mode_plt_flag` may be present in the coding unit syntax. `sps_palette_enabled_flag` equal to 0 specifies that `pred_mode_plt_flag` is not present in the coding unit syntax. When `sps_palette_enabled_flag` is not present, it is inferred to be equal to 0.

Tham khảo đến Bảng 17 và Bảng 18 ở trên, trong SPS, các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu có thể được xác định, và có thể gồm các phần tử cú pháp của `sps_transform_skip_enabled_flag`, `sps_bdpcm_enabled_flag`, `log2_transform_skip_max_size_minus2`, `sps_palette_enabled_flag`, `min_qp_prime_ts_luma_minus4`, và `min_qp_prime_ts_chroma_minus4`.

Ở đây, do các phần tử cú pháp của `sps_transform_skip_enabled_flag`, `sps_bdpcm_enabled_flag`, `log2_transform_skip_max_size_minus2`, `sps_palette_enabled_flag`, `min_qp_prime_ts_luma_minus4`, và `min_qp_prime_ts_chroma_minus4` đã được giải thích chi tiết trong Bảng 6 đến Bảng 11 ở trên, trong phương án này, để thuận tiện cho việc giải thích, phần giải thích chi tiết của chúng sẽ được bỏ qua.

Như được bộc lộ trong phương án của Bảng 17 và Bảng 18 ở trên, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 17 và Bảng 18 ở trên, trong SPS, phần tử cú pháp `sps_bdpcm_enabled_flag` biểu diễn xem BDPCM có được cho phép hay không dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và phần tử cú pháp `log2_transform_skip_max_size_minus2` biểu diễn kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi có thể có sự phụ thuộc. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ,

`sps_transform_skip_enabled_flag`) là 1, các phần tử cú pháp của `sps_bdpcm_enabled_flag` và `log2_transform_skip_max_size_minus2` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, các phần tử cú pháp của `sps_bdpcm_enabled_flag` và `log2_transform_skip_max_size_minus2` có thể không được phân tích cú pháp/được báo hiệu. Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 17 và Bảng 18 ở trên, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu trong chế độ bỏ qua biến đổi cho thành phần sắc độ dựa trên giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể có sự phụ thuộc trong SPS. Như một ví dụ, nếu giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` có thể được phân tích cú pháp/được báo hiệu. Ngoài ra, nếu giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 0, phần tử cú pháp `min_qp_prime_ts_chroma_minus4` có thể không được phân tích cú pháp/được báo hiệu.

Ngoài ra, trong SPS, các phần tử cú pháp có sự phụ thuộc có thể được xác định đối với phần tử cú pháp cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) trong số các phần tử cú pháp liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu. Ví dụ, như được bộc lộ trong Bảng 17 và Bảng 18 ở trên, phần tử cú pháp `min_qp_prime_ts_luma_minus4` biểu diễn thông tin thông số lượng tử hóa tối thiểu trong chế độ bỏ qua biến đổi cho thành phần độ chói dựa trên giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc phần tử cú pháp cờ được cho phép lập mã

bảng màu (ví dụ, `sps_palette_enabled_flag`) có thể có sự phụ thuộc trong SPS. Như một ví dụ, nếu giá trị của cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của cờ được cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, phần tử cú pháp `min_qp_prime_ts_luma_minus4` có thể được phân tích cú pháp/được báo hiệu.

Hình vẽ sau đây đã được chuẩn bị để giải thích ví dụ chi tiết của sáng chế. Do tên gọi của thiết bị chi tiết hoặc thuật ngữ chi tiết hoặc tên gọi (ví dụ, cú pháp/tên của phần tử cú pháp) được mô tả trong hình vẽ được thể hiện ví dụ, các dấu hiệu kỹ thuật của sáng chế không bị giới hạn ở tên gọi chi tiết được sử dụng trong hình vẽ. Fig.10 và Fig.11 minh họa một cách sơ lược phương pháp mã hóa video/hình ảnh và ví dụ của các thành phần liên quan theo (các) phương án của sáng chế.

Fig.10 và Fig.11 minh họa một cách sơ lược phương pháp mã hóa video/hình ảnh và ví dụ của các thành phần liên quan theo (các) phương án của sáng chế.

Phương pháp được bộc lộ trên Fig.10 có thể được thực hiện bởi thiết bị mã hóa 200 được bộc lộ trên Fig.2 hoặc Fig.11. Ở đây, thiết bị mã hóa 200 được bộc lộ trên Fig.11 thể hiện sơ lược thiết bị mã hóa 200 được bộc lộ trên Fig.2. Cụ thể là, các bước S1000 đến S1010 của Fig.10 có thể được thực hiện bởi bộ xử lý phân dư 230 được bộc lộ trên Fig.2, và bước S1020 của Fig.10 có thể được thực hiện bởi bộ mã hóa entropi 240 được bộc lộ trên Fig.2. Ngoài ra, mặc dù không được minh họa, phương pháp dẫn xuất mẫu dự đoán có thể được thực hiện bởi bộ dự đoán 220 của thiết bị mã hóa 200, quy trình tạo ra mẫu được xây dựng lại cho khối hiện tại và ảnh được xây dựng lại dựa trên mẫu phân dư và mẫu dự đoán cho khối hiện tại có thể được thực hiện bởi bộ cộng 250 của thiết bị mã hóa 200, và quy trình mã hóa thông tin dự đoán cho khối hiện tại có thể được thực hiện bởi bộ mã hóa entropi 240 của thiết bị mã hóa 200. Ngoài ra, phương pháp được bộc lộ trên Fig.10 có thể gồm các phương án được mô tả ở trên của sáng chế cần được thực hiện. Theo đó, tham chiếu đến Fig.10, việc giải thích chi tiết của các nội dung trùng lặp của các phương án được mô tả ở trên sẽ được bỏ qua hoặc được đơn giản

hóa.

Tham khảo đến Fig.10, thiết bị mã hóa có thể xác định xem có áp dụng việc biến đổi cho khối hiện tại dựa trên thông tin cho phép bỏ qua biến đổi hay không (S1000).

Như một phương án, thiết bị mã hóa trước tiên có thể xác định chế độ dự đoán cho khối hiện tại, và có thể dẫn xuất các mẫu dự đoán. Ví dụ, thiết bị mã hóa có thể xác định xem thực hiện việc dự đoán liên hay việc dự đoán nội đối với khối hiện tại, và cũng có thể xác định chế độ dự đoán liên cụ thể hoặc chế độ dự đoán nội cụ thể dựa trên chi phí RD. Ngoài ra, thiết bị mã hóa có thể xác định xem thực hiện việc dự đoán đối với khối hiện tại dựa trên chế độ CIIP, chế độ IBC, chế độ BDPCM, hay chế độ bảng màu. Thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán cho khối hiện tại bằng việc thực hiện việc dự đoán theo chế độ dự đoán được xác định. Trong trường hợp này, các phương pháp dự đoán khác nhau được bộc lộ trong sáng chế, chẳng hạn như việc dự đoán liên hoặc việc dự đoán nội, có thể được áp dụng. Ngoài ra, thiết bị mã hóa có thể tạo ra và mã hóa thông tin (ví dụ, thông tin chế độ dự đoán) liên quan tới việc dự đoán được áp dụng cho khối hiện tại.

Ngoài ra, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư bằng việc so sánh các mẫu dự đoán với các mẫu gốc cho khối hiện tại với nhau. Thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi thông qua quy trình biến đổi cho các mẫu phần dư. Trong trường hợp này, thiết bị mã hóa có thể xác định xem có áp dụng việc biến đổi cho khối hiện tại hay không xem xét đến hiệu quả lập mã. Nghĩa là, thiết bị mã hóa có thể xác định xem việc biến đổi có được áp dụng cho các mẫu phần dư của khối hiện tại hay không.

Ví dụ, thiết bị mã hóa có thể xác định xem có áp dụng việc biến đổi hoặc chế độ bỏ qua biến đổi cho khối hiện tại (các mẫu phần dư) hay không dựa trên thông tin cho phép bỏ qua biến đổi.

Như được mô tả ở trên, thông tin cho phép bỏ qua biến đổi có thể là thông tin về việc liệu việc bỏ qua biến đổi có được cho phép hay không, và như được bộc lộ trong

Bảng 6 đến Bảng 18, có thể được biểu diễn dưới dạng phần tử cú pháp `sps_transform_skip_enabled_flag`. Ví dụ, nếu giá trị của `sps_transform_skip_enabled_flag` là 1, nó có thể thể hiện rằng việc bỏ qua biến đổi được cho phép, và trong trường hợp này, `transform_skip_flag` có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị biến đổi. Ở đây, phần tử cú pháp `transform_skip_flag` có thể thể hiện xem việc biến đổi có thể được áp dụng cho khối biến đổi được kết hợp hay không. Nếu giá trị của `sps_transform_skip_enabled_flag` là 0, nó có thể thể hiện rằng việc bỏ qua biến đổi là không được cho phép, và trong trường hợp này, `transform_skip_flag` có thể không được phân tích cú pháp/được báo hiệu trong cú pháp đơn vị biến đổi. Thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) có thể được gồm trong SPS, và có thể được báo hiệu tới thiết bị giải mã. Nghĩa là, dựa trên việc giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) được gồm trong SPS là 1, cú pháp đơn vị biến đổi có thể gồm cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`). Trong trường hợp này, nếu giá trị của cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) được gồm trong cú pháp đơn vị biến đổi là 1, chế độ trong đó việc biến đổi không được áp dụng (chế độ bỏ qua biến đổi) có thể được thực hiện cho khối hiện tại. Ngoài ra, nếu giá trị của cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) được gồm trong cú pháp đơn vị biến đổi là 0, việc biến đổi có thể được áp dụng cho khối hiện tại.

Ví dụ, nếu giá trị của thông tin cho phép bỏ qua biến đổi là 1 (nghĩa là, đối với thông tin cho phép bỏ qua biến đổi biểu diễn mà việc bỏ qua biến đổi được cho phép), thiết bị mã hóa có thể xác định xem có áp dụng việc biến đổi cho khối hiện tại hay không. Nghĩa là, thiết bị mã hóa có thể tạo ra thông tin (cờ bỏ qua biến đổi) về việc liệu có áp dụng việc biến đổi cho khối hiện tại hay không dựa trên việc giá trị của thông tin cho phép bỏ qua biến đổi là 1, và có thể báo hiệu cờ bỏ qua biến đổi thông qua cú pháp đơn vị biến đổi. Trong trường hợp này, nếu việc biến đổi không được áp dụng cho khối hiện tại (nghĩa là, trong trường hợp chế độ bỏ qua biến đổi), thiết bị mã hóa có thể tạo ra cờ

bỏ qua biến đổi mà giá trị của nó là 1, và có thể gồm điều này trong cú pháp đơn vị biến đổi. Ngoài ra, trong trường hợp áp dụng việc biến đổi cho khối hiện tại, thiết bị mã hóa có thể tạo ra cờ bỏ qua biến đổi mà giá trị của nó là 0, và có thể gồm điều này trong cú pháp đơn vị biến đổi.

Thiết bị mã hóa có thể tạo ra thông tin phần dư cho khối hiện tại dựa trên việc liệu có áp dụng việc biến đổi hay không (S1010).

Như một phương án, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư của khối hiện tại, và có thể tạo ra thông tin phần dư bằng việc áp dụng việc biến đổi hoặc việc bỏ qua biến đổi cho các mẫu phần dư của khối hiện tại dựa trên việc liệu có áp dụng việc biến đổi hay không. Ví dụ, đối với các mẫu phần dư cho khối hiện tại mà giá trị cờ bỏ qua biến đổi của nó là 1, thiết bị mã hóa có thể áp dụng chế độ bỏ qua biến đổi. Trong trường hợp này, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư của khối hiện tại như các hệ số biến đổi. Ngoài ra, đối với các mẫu phần dư cho khối hiện tại mà giá trị cờ bỏ qua biến đổi của nó là 0, thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi bằng việc thực hiện việc biến đổi. Thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng việc thực hiện quy trình lượng tử hóa dựa trên các hệ số biến đổi được dẫn xuất thông qua việc bỏ qua biến đổi hoặc việc biến đổi. Thiết bị mã hóa có thể tạo ra thông tin phần dư dựa trên các hệ số biến đổi được lượng tử hóa.

Ở đây, thông tin phần dư có thể là thông tin được tạo ra thông qua quy trình biến đổi và/hoặc lượng tử hóa, và có thể là thông tin về các hệ số biến đổi được lượng tử hóa, và ví dụ, có thể gồm thông tin về thông tin giá trị của các hệ số biến đổi được lượng tử hóa, thông tin vị trí, kỹ thuật biến đổi, nhân biến đổi, và thông số lượng tử hóa.

Thiết bị mã hóa có thể mã hóa thông tin hình ảnh (hoặc thông tin video) (S1020).

Ở đây, thông tin hình ảnh có thể gồm thông tin phần dư. Ngoài ra, thông tin hình ảnh có thể gồm thông tin (ví dụ, thông tin chế độ dự đoán) liên quan tới việc dự đoán được sử dụng để dẫn xuất các mẫu dự đoán. Ngoài ra, thông tin hình ảnh có thể gồm

thông tin liên quan tới việc bỏ qua biến đổi, ví dụ, thông tin cho phép bỏ qua biến đổi và thông tin cờ bỏ qua biến đổi. Nghĩa là, thông tin hình ảnh có thể gồm các loại thông tin khác nhau được dẫn xuất trong quy trình mã hóa, và thông tin hình ảnh gồm các loại thông tin khác nhau này có thể được mã hóa.

Ngoài ra, thông tin hình ảnh có thể gồm các loại thông tin khác nhau theo (các) phương án được mô tả ở trên trong sáng chế, và có thể gồm thông tin được bộc lộ trong ít nhất một trong số các Bảng 1 đến 18 như được mô tả ở trên.

Ví dụ, thông tin hình ảnh có thể gồm tập thông số trình tự (SPS). SPS có thể gồm thông tin liên quan đến việc bỏ qua biến đổi và thông tin liên quan đến việc lập mã bảng màu. Như một ví dụ, thông tin liên quan đến việc bỏ qua biến đổi có thể gồm thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`), thông tin về kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), và thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`). Ngoài ra, như một ví dụ, thông tin liên quan đến việc lập mã bảng màu có thể gồm thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) và thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`).

Ngoài ra, ví dụ, như được mô tả ở trên, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`).

Như một ví dụ, SPS có thể tạo cấu hình để phân tích cú pháp/báo hiệu thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) về việc liệu có cho phép BDPCM hay không dựa trên giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ,

`sps_transform_skip_enabled_flag`). Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) có thể được gồm trong SPS, và thông tin (ví dụ, `sps_bdpcm_enabled_flag`) có thể được phân tích cú pháp/được báo hiệu từ SPS. Ngoài ra, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) có thể không được phân tích cú pháp/được báo hiệu từ SPS.

Ngoài ra, dựa trên giá trị của thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) trong SPS, thông tin cờ BDPCM (ví dụ, `intra_bdpcm_flag`) về việc liệu có áp dụng BDPCM cho khối hiện tại hay không) có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã. Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) là 1, thông tin cờ BDPCM (ví dụ, `intra_bdpcm_flag`) có thể được gồm trong cú pháp đơn vị lập mã, và thông tin (ví dụ, `intra_bdpcm_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã. Ngoài ra, nếu giá trị của thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) là 0, thông tin cờ BDPCM (ví dụ, `intra_bdpcm_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã.

Ngoài ra, như một ví dụ, SPS có thể tạo cấu hình để phân tích cú pháp/báo hiệu thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi dựa trên giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`). Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi có thể được gồm trong SPS, và thông tin (ví dụ, `log2_transform_skip_max_size_minus2`) có thể được phân tích cú pháp/được báo hiệu từ SPS. Ngoài ra, nếu giá trị của thông tin

cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi có thể không được phân tích cú pháp/được báo hiệu từ SPS.

Ngoài ra, như một ví dụ, dựa trên giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) được xác định trong SPS, thông tin cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) về việc liệu có áp dụng việc bỏ qua biến đổi cho khối hiện tại hay không có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị biến đổi. Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) có thể được gồm trong cú pháp đơn vị biến đổi, và thông tin (ví dụ, `transform_skip_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị biến đổi. Ngoài ra, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị biến đổi.

Ngoài ra, ví dụ, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS như được mô tả ở trên, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép lập mã bảng màu về việc liệu việc lập mã bảng màu có được cho phép hay không (ví dụ, `sps_palette_enabled_flag`). Như một ví dụ, dựa trên giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) được xác định trong SPS, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) về việc liệu có áp dụng việc lập mã bảng màu (chế độ dự đoán bảng màu) cho khối hiện tại hay không có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã. Trong trường hợp này, nếu giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) có thể được gồm trong cú pháp đơn vị

lập mã, và thông tin (ví dụ, `pred_mode_plt_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã. Ngoài ra, nếu giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 0, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã.

Ngoài ra, ví dụ, như được mô tả ở trên, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`).

Như một ví dụ, trong SPS, dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`), thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi có thể được phân tích cú pháp/được báo hiệu. Nói cách khác, dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được gồm trong SPS, và chỉ trong trường hợp mà điều kiện nêu trên được thỏa mãn, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được phân tích cú pháp/được báo hiệu.

Ở đây, như được mô tả ở trên, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể là thông tin liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi, và dựa trên điều này, thông số lượng tử hóa cho khôi hiện tại có thể được dẫn xuất.

Ví dụ, trong trường hợp mà chế độ bỏ qua biến đổi được áp dụng cho khôi hiện tại, thông số lượng tử hóa cho khôi hiện tại có thể được dẫn xuất dựa trên thông tin thông

số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`), và các hệ số biến đổi được lượng tử hóa có thể được dẫn xuất bằng việc thực hiện quy trình lượng tử hóa dựa trên thông số lượng tử hóa.

Ngoài ra, ví dụ, trong trường hợp mà chế độ lập mã bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho giá trị thoát của khối hiện tại có thể được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`). Trong trường hợp này, giá trị thoát được lượng tử hóa (ví dụ, `palette_escape_val`) có thể được dẫn xuất bằng việc áp dụng thông số lượng tử hóa cho giá trị thoát của khối hiện tại dựa trên thông số lượng tử hóa. Quy trình trong đó chế độ lập mã bảng màu được áp dụng có thể được thực hiện như được bộc lộ trong Bảng 4 và Bảng 5 nêu trên.

Như được mô tả ở trên, thông tin hình ảnh gồm các loại thông tin khác nhau có thể được mã hóa và được xuất dưới dạng luồng bit. Luồng bit có thể được truyền tới thiết bị giải mã thông qua mạng hoặc phương tiện lưu trữ (kỹ thuật số). Ở đây, mạng có thể gồm mạng phát rộng và/hoặc mạng truyền thông, và phương tiện lưu trữ kỹ thuật số có thể gồm phương tiện lưu trữ khác nhau, chẳng hạn như USB, SD, CD, DVD, Blu-ray, HDD, và SSD.

Fig.12 và Fig.13 minh họa một cách sơ lược phương pháp giải mã video/hình ảnh và ví dụ của các thành phần liên quan theo (các) phương án của sáng chế.

Phương pháp được bộc lộ trên Fig.12 có thể được thực hiện bởi thiết bị giải mã 300 được bộc lộ trên Fig.3 hoặc Fig.13. Ở đây, thiết bị giải mã 300 được bộc lộ trên Fig.13 thể hiện sơ lược thiết bị giải mã 300 được bộc lộ trên Fig.3. Cụ thể là, bước S1200 của Fig.12 có thể được thực hiện bởi bộ giải mã entropi 310 được bộc lộ trên Fig.3, các bước từ S1210 đến S1220 của Fig.12 có thể được thực hiện bởi bộ xử lý phần dư 320 được bộc lộ trên Fig.3, và bước S1230 của Fig.12 có thể được thực hiện bởi bộ cộng 340 được bộc lộ trên Fig.3. Ngoài ra, mặc dù không được minh họa, quy trình nhận thông tin dự đoán cho khối hiện tại có thể được thực hiện bởi bộ giải mã entropi 310 của thiết bị giải mã 300, và phương pháp dẫn xuất mẫu dự đoán của khối hiện tại có thể

được thực hiện bởi bộ dự đoán 330 của thiết bị giải mã 300. Ngoài ra, phương pháp được bộc lộ trên Fig.12 có thể gồm các phương án được mô tả ở trên của sáng chế cần được thực hiện. Theo đó, tham chiếu đến Fig.12, việc giải thích chi tiết của các nội dung trùng lặp của các phương án được mô tả ở trên sẽ được bỏ qua hoặc được đơn giản hóa.

Tham khảo đến Fig.12, thiết bị giải mã có thể nhận thông tin hình ảnh (hoặc thông tin video) từ luồng bit (S1200).

Như một phương án, thiết bị giải mã có thể dẫn xuất thông tin (ví dụ, thông tin video/hình ảnh) cần thiết cho việc xây dựng lại hình ảnh (hoặc việc xây dựng lại ảnh) bằng việc phân tích cú pháp luồng bit. Trong trường hợp này, thông tin hình ảnh có thể gồm thông tin phần dư, và thông tin phần dư có thể gồm thông tin giá trị của các hệ số biến đổi được lượng tử hóa, thông tin vị trí, kỹ thuật biến đổi, nhân biến đổi, và thông tin thông số lượng tử hóa. Ngoài ra, thông tin hình ảnh có thể gồm thông tin liên quan đến việc dự đoán (ví dụ, thông tin chế độ dự đoán). Ngoài ra, thông tin hình ảnh có thể gồm thông tin liên quan tới việc bỏ qua biến đổi, ví dụ, thông tin cho phép bỏ qua biến đổi và thông tin cờ bỏ qua biến đổi. Nghĩa là, thông tin hình ảnh có thể gồm các loại thông tin khác nhau cần thiết trong quy trình giải mã, và có thể được giải mã dựa trên phương pháp lập mã, chẳng hạn như lập mã Golomb hàm số mũ, CAVLC, hoặc CABAC.

Ngoài ra, thông tin hình ảnh có thể gồm các loại thông tin khác nhau theo các phương án được mô tả ở trên của sáng chế, và có thể gồm thông tin được bộc lộ trong ít nhất một trong số các Bảng 1 đến 18 như được mô tả ở trên.

Ví dụ, thông tin hình ảnh có thể gồm tập thông số trình tự (SPS). SPS có thể gồm thông tin liên quan đến việc bỏ qua biến đổi và thông tin liên quan đến việc lập mã bảng màu. Như một ví dụ, thông tin liên quan đến việc bỏ qua biến đổi có thể gồm thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`), thông tin cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`), thông tin về kích thước khối tối đa được sử dụng trong chế độ bỏ qua biến đổi (ví dụ, `log2_transform_skip_max_size_minus2`), và thông tin thông số lượng tử hóa tối thiểu

liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`). Ngoài ra, như một ví dụ, thông tin liên quan đến việc lập mã bảng màu có thể gồm thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) và thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi (ví dụ, `min_qp_prime_ts_minus4`).

Ngoài ra, ví dụ, như được mô tả ở trên, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`).

Như một ví dụ, SPS có thể tạo cấu hình để phân tích cú pháp/báo hiệu thông tin còn được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) về việc liệu có cho phép BDPCM hay không dựa trên giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`). Trong trường hợp này, nếu giá trị của thông tin còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin còn được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) có thể được gồm trong SPS, và thông tin (ví dụ, `sps_bdpcm_enabled_flag`) có thể được phân tích cú pháp/được báo hiệu từ SPS. Ngoài ra, nếu giá trị của thông tin còn được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin còn được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) có thể không được phân tích cú pháp/được báo hiệu từ SPS.

Ngoài ra, dựa trên giá trị của thông tin còn được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) trong SPS, thông tin còn BDPCM (ví dụ, `intra_bdpcm_flag`) về việc liệu có áp dụng BDPCM cho khối hiện tại hay không) có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã. Trong trường hợp này, nếu giá trị của thông tin còn được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) là 1, thông tin còn BDPCM (ví dụ, `intra_bdpcm_flag`) có thể được gồm trong cú pháp đơn vị lập mã,

và thông tin (ví dụ, `intra_bdpcm_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã. Ngoài ra, nếu giá trị của thông tin cờ được cho phép BDPCM (ví dụ, `sps_bdpcm_enabled_flag`) là 0, thông tin cờ BDPCM (ví dụ, `intra_bdpcm_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã.

Ngoài ra, như một ví dụ, SPS có thể tạo cấu hình để phân tích cú pháp/báo hiệu thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi dựa trên giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`). Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi có thể được gồm trong SPS, và thông tin (ví dụ, `log2_transform_skip_max_size_minus2`) có thể được phân tích cú pháp/được báo hiệu từ SPS. Ngoài ra, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin về kích thước khối tối đa (ví dụ, `log2_transform_skip_max_size_minus2`) được sử dụng trong chế độ bỏ qua biến đổi có thể không được phân tích cú pháp/được báo hiệu từ SPS.

Ngoài ra, như một ví dụ, dựa trên giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) được xác định trong SPS, thông tin cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) về việc liệu có áp dụng việc bỏ qua biến đổi cho khối hiện tại hay không có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị biến đổi. Trong trường hợp này, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, thông tin cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) có thể được gồm trong cú pháp đơn vị biến đổi, và thông tin (ví dụ, `transform_skip_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị biến đổi. Ngoài ra, nếu giá trị của thông tin cờ được cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 0, thông tin cờ bỏ qua biến đổi (ví

dụ, `transform_skip_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị biến đổi.

Ngoài ra, ví dụ, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS như được mô tả ở trên, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép lập mã bảng màu về việc liệu việc lập mã bảng màu có được cho phép hay không (ví dụ, `sps_palette_enabled_flag`). Như một ví dụ, dựa trên giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) được xác định trong SPS, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) về việc liệu có áp dụng việc lập mã bảng màu (chế độ dự đoán bảng màu) cho khối hiện tại hay không có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị lập mã. Trong trường hợp này, nếu giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) có thể được gồm trong cú pháp đơn vị lập mã, và thông tin (ví dụ, `pred_mode_plt_flag`) có thể được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã. Ngoài ra, nếu giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 0, thông tin cờ chế độ dự đoán bảng màu (ví dụ, `pred_mode_plt_flag`) có thể không được phân tích cú pháp/được báo hiệu từ cú pháp đơn vị lập mã.

Ngoài ra, ví dụ, như được mô tả ở trên, trong số thông tin liên quan tới việc bỏ qua biến đổi và/hoặc việc lập mã bảng màu được gồm trong SPS, thông tin có sự phụ thuộc có thể được xác định đối với thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`).

Như một ví dụ, trong SPS, dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`), thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) liên quan tới thông số lượng tử hóa được cho phép

tối thiểu cho chế độ bỏ qua biến đổi có thể được phân tích cú pháp/được báo hiệu. Nói cách khác, dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được gồm trong SPS, và chỉ trong trường hợp mà điều kiện nêu trên được thỏa mãn, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được phân tích cú pháp/được báo hiệu.

Ở đây, như được mô tả ở trên, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể là thông tin liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi, và dựa trên điều này, thông số lượng tử hóa cho khối hiện tại có thể được dẫn xuất.

Ví dụ, trong trường hợp mà chế độ bỏ qua biến đổi được áp dụng cho khối hiện tại, thông số lượng tử hóa cho khối hiện tại có thể được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`), và hệ số biến đổi được khử lượng tử hóa (hệ số biến đổi được định cỡ) có thể được dẫn xuất bằng việc thực hiện quy trình khử lượng tử hóa (quy trình định cỡ) dựa trên thông số lượng tử hóa. Dựa trên hệ số biến đổi được khử lượng tử hóa, mẫu phần dư của khối hiện tại có thể được dẫn xuất.

Ngoài ra, ví dụ, trong trường hợp mà chế độ lập mã bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho giá trị thoát của khối hiện tại có thể được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`). Trong trường hợp này, giá trị thoát của khối hiện tại có thể được dẫn xuất bằng việc thực hiện việc khử lượng tử hóa (quy trình định cỡ) dựa trên thông số lượng tử hóa. Dựa trên giá trị thoát, mẫu được xây dựng lại của khối hiện tại có thể được tạo ra. Quy trình trong đó chế độ lập mã bảng màu được áp dụng có thể được thực hiện như được bộc lộ trong Bảng 4 và Bảng 5 nêu trên.

Thiết bị giải mã có thể xác định xem có áp dụng việc biến đổi cho khối hiện tại

hay không dựa trên thông tin cho phép bỏ qua biến đổi (S1210).

Như một phương án, nếu thông tin hình ảnh gồm thông tin cho phép bỏ qua biến đổi được nhận, thiết bị giải mã có thể xác định xem áp dụng chế độ biến đổi hay bỏ qua biến đổi cho khối hiện tại dựa trên thông tin cho phép bỏ qua biến đổi.

Như được mô tả ở trên, thông tin cho phép bỏ qua biến đổi có thể là thông tin về việc liệu việc bỏ qua biến đổi có được cho phép hay không, và có thể được biểu diễn dưới dạng phần tử cú pháp `sps_transform_skip_enabled_flag` như được bộc lộ trong Bảng 6 đến Bảng 18. Ví dụ, nếu giá trị của `sps_transform_skip_enabled_flag` là 1, nó có thể thể hiện rằng việc bỏ qua biến đổi được cho phép, và trong trường hợp này, `transform_skip_flag` có thể được phân tích cú pháp/được báo hiệu thông qua cú pháp đơn vị biến đổi. Ở đây, phần tử cú pháp `transform_skip_flag` có thể thể hiện xem việc biến đổi có thể được áp dụng cho khối biến đổi được kết hợp hay không. Nếu giá trị của `sps_transform_skip_enabled_flag` là 0, nó có thể thể hiện rằng việc bỏ qua biến đổi là không được cho phép, và trong trường hợp này, `transform_skip_flag` có thể không được phân tích cú pháp/được báo hiệu trong cú pháp đơn vị biến đổi. Thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) có thể được gồm trong SPS, và có thể được báo hiệu từ thiết mã hóa đến thiết bị giải mã. Nghĩa là, dựa trên việc giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) được gồm trong SPS là 1, cú pháp đơn vị biến đổi có thể gồm cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`). Trong trường hợp này, nếu giá trị của cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) được gồm trong cú pháp đơn vị biến đổi là 1, chế độ trong đó việc biến đổi không được áp dụng (chế độ bỏ qua biến đổi) có thể được thực hiện cho khối hiện tại. Ngoài ra, nếu giá trị của cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`) được gồm trong cú pháp đơn vị biến đổi là 0, việc biến đổi có thể được áp dụng cho khối hiện tại.

Ví dụ, nếu giá trị của thông tin cho phép bỏ qua biến đổi là 1 (nghĩa là, đối với thông tin cho phép bỏ qua biến đổi biểu diễn mà việc bỏ qua biến đổi được cho phép),

thiết bị giải mã có thể xác định xem có áp dụng việc biến đổi cho khối hiện tại hay không.

Thiết bị giải mã có thể dẫn xuất mẫu phân dư dựa trên việc liệu có áp dụng việc biến đổi hay không và thông tin phân dư (S1220).

Như một phương án, thiết bị giải mã có thể nhận thông tin hình ảnh gồm thông tin phân dư. Như được mô tả ở trên, thông tin phân dư có thể gồm thông tin giá trị của các hệ số biến đổi được lượng tử hóa, thông tin vị trí, kỹ thuật biến đổi, nhân biến đổi, và thông tin thông số lượng tử hóa. Thiết bị giải mã có thể dẫn xuất các hệ số biến đổi được lượng tử hóa cho khối hiện tại dựa trên thông tin hệ số biến đổi được lượng tử hóa được gồm trong thông tin phân dư, và có thể dẫn xuất các hệ số biến đổi dựa trên các hệ số biến đổi được lượng tử hóa. Ngoài ra, thiết bị giải mã có thể dẫn xuất các mẫu phân dư dựa trên các hệ số biến đổi.

Ví dụ, nếu giá trị của thông tin cho phép bỏ qua biến đổi là 1 (nghĩa là, đối với thông tin cho phép bỏ qua biến đổi biểu diễn mà việc bỏ qua biến đổi được cho phép), thiết bị giải mã có thể thu nhận thông tin về xem có áp dụng việc biến đổi cho khối hiện tại hay không (cờ bỏ qua biến đổi) từ cú pháp đơn vị biến đổi. Trong trường hợp này, thiết bị giải mã có thể dẫn xuất các mẫu phân dư dựa trên thông tin cờ bỏ qua biến đổi. Ví dụ, chế độ bỏ qua biến đổi có thể được áp dụng cho khối hiện tại mà giá trị của cờ bỏ qua biến đổi của nó là 1, và trong trường hợp này, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi như các mẫu phân dư của khối hiện tại. Ngoài ra, việc biến đổi có thể được áp dụng cho khối hiện tại mà giá trị của cờ bỏ qua biến đổi của nó là 0, và trong trường hợp này, thiết bị giải mã có thể dẫn xuất các mẫu phân dư của khối hiện tại thông qua việc biến đổi ngược của các hệ số biến đổi.

Ngoài ra, đối với khối hiện tại mà giá trị của cờ bỏ qua biến đổi của nó là 1 (nghĩa là, chế độ bỏ qua biến đổi), thiết bị giải mã có thể dẫn xuất thông số lượng tử hóa được sử dụng trong quy trình khử lượng tử hóa dựa trên thông tin thông số lượng tử hóa tối thiểu. Ngoài ra, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi được khử lượng tử hóa bằng việc thực hiện quy trình khử lượng tử hóa dựa trên thông số lượng tử hóa,

và có thể dẫn xuất các mẫu phần dư dựa trên các hệ số biến đổi được khử lượng tử hóa.

Ở đây, như được mô tả ở trên, thông tin thông số lượng tử hóa tối thiểu có thể là thông tin liên quan tới thông số lượng tử hóa được cho phép tối thiểu cho chế độ bỏ qua biến đổi, và có thể được gồm trong thông tin hình ảnh (ví dụ, SPS) dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) và/hoặc thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`). Ví dụ, dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi (ví dụ, `sps_transform_skip_enabled_flag`) là 1, hoặc giá trị của thông tin cho phép lập mã bảng màu (ví dụ, `sps_palette_enabled_flag`) là 1, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được gồm trong SPS. Nghĩa là, chỉ trong trường hợp mà điều kiện nêu trên được thỏa mãn, thông tin thông số lượng tử hóa tối thiểu (ví dụ, `min_qp_prime_ts_minus4`) có thể được phân tích cú pháp/được báo hiệu.

Thiết bị giải mã có thể tạo ra các mẫu được xây dựng lại dựa trên các mẫu phần dư (S1230).

Như một phương án, thiết bị giải mã có thể xác định xem thực hiện việc dự đoán liên hay việc dự đoán nội cho khối hiện tại dựa trên thông tin dự đoán (ví dụ, thông tin chế độ dự đoán) được gồm trong thông tin hình ảnh, và có thể dẫn xuất các mẫu dự đoán cho khối hiện tại bằng việc thực hiện việc dự đoán theo việc xác định. Ngoài ra, thiết bị giải mã có thể tạo ra các mẫu được xây dựng lại dựa trên các mẫu dự đoán và các mẫu phần dư. Trong trường hợp này, thiết bị giải mã có thể sử dụng trực tiếp các mẫu dự đoán như các mẫu được xây dựng lại theo chế độ dự đoán, hoặc có thể tạo ra các mẫu được xây dựng lại bằng việc cộng các mẫu phần dư với các mẫu dự đoán. Ngoài ra, thiết bị giải mã có thể dẫn xuất khối được xây dựng lại hoặc ảnh được xây dựng lại dựa trên các mẫu được xây dựng lại. Sau đó, nếu cần thiết, thiết bị giải mã có thể áp dụng thủ tục lọc trong vòng lặp, chẳng hạn như thủ tục lọc khử khối và/hoặc SAO, cho ảnh được xây dựng lại để cải thiện chất lượng ảnh chủ quan/khách quan như được mô tả ở trên.

Trong các phương án được mô tả ở trên, các phương pháp được giải thích trên

cơ sở của các lưu đồ bởi chuỗi các bước hoặc các khối, nhưng sáng chế không bị giới hạn ở thứ tự của các bước, và bước nhất định có thể được thực hiện theo thứ tự hoặc bước khác với được mô tả ở trên, hoặc đồng thời với các bước khác. Ngoài ra, có thể được hiểu bởi người có trình độ trung bình trong lĩnh vực rằng các bước được thể hiện trong lưu đồ không phải là bắt buộc, và bước khác có thể được kết hợp hoặc một hoặc nhiều bước của lưu đồ có thể được loại bỏ mà không ảnh hưởng đến phạm vi của sáng chế.

Phương pháp theo sáng chế có thể được thực hiện dưới dạng phần mềm, và thiết bị mã hóa và/hoặc thiết bị giải mã theo sáng chế có thể được gồm trong thiết bị mà thực hiện việc xử lý hình ảnh, chẳng hạn như TV, máy tính, điện thoại thông minh, hộp thu tín hiệu truyền hình, và thiết bị hiển thị.

Khi các phương án của sáng chế được thực hiện bởi phần mềm, phương pháp được đề cập ở trên có thể được thực hiện bởi môđun (quy trình hoặc chức năng) mà thực hiện chức năng được đề cập ở trên. Môđun có thể được lưu trữ trong bộ nhớ và được thực thi bởi bộ xử lý. Bộ nhớ có thể được lắp đặt phía trong hoặc phía ngoài bộ xử lý và có thể được kết nối với bộ xử lý thông qua các phương tiện đã biết khác nhau. Bộ xử lý có thể gồm mạch tích hợp ứng dụng cụ thể (Application-Specific Integrated Circuit, ASIC), các bộ chip khác, mạch logic, và/hoặc thiết bị xử lý dữ liệu. Bộ nhớ có thể gồm bộ nhớ chỉ đọc (Read-Only Memory, ROM), bộ nhớ truy cập ngẫu nhiên (Random Access Memory, RAM), bộ nhớ tác động nhanh, thẻ nhớ, phương tiện lưu trữ, và/hoặc thiết bị lưu trữ khác. Nói cách khác, các phương án theo sáng chế có thể được thực hiện và được thực thi trên bộ xử lý, bộ vi xử lý, bộ điều khiển, hoặc chip. Ví dụ, các đơn vị chức năng được minh họa trên các hình vẽ tương ứng có thể được thực hiện và được thực thi trên máy tính, bộ xử lý, bộ vi xử lý, bộ điều khiển, hoặc chip. Trong trường hợp này, thông tin về việc thực hiện (ví dụ, thông tin về các lệnh) hoặc các thuật toán có thể được lưu trữ trong phương tiện lưu trữ kỹ thuật số.

Ngoài ra, thiết bị giải mã và thiết bị mã hóa mà cho nó sáng chế được áp dụng

có thể được gồm trong bộ thu phát phát rộng đa phương tiện, đầu cuối truyền thông di động, thiết bị video điện ảnh tại nhà, thiết bị video điện ảnh kỹ thuật số, camera giám sát, thiết bị trò chuyện video, và thiết bị truyền thông thời gian thực chẳng hạn như truyền thông video, thiết bị phát luồng di động, phương tiện lưu trữ, máy quay video, nhà cung cấp dịch vụ video theo yêu cầu (video on demand, VoD), thiết bị cung cấp dịch vụ nội dung trên nền mạng viễn thông (Over The Top, OTT), thiết bị cung cấp dịch vụ tạo luồng Internet, thiết bị video 3D, thiết bị thực tế ảo (Virtual Reality, VR), thiết bị thực tế tăng cường (Augment Reality, AR), thiết bị video điện thoại hình ảnh, đầu cuối phương tiện (ví dụ, đầu cuối phương tiện (gồm phương tiện tự động), đầu cuối máy bay, hoặc đầu cuối tàu), và thiết bị video y tế; và có thể được sử dụng để xử lý tín hiệu hoặc dữ liệu hình ảnh. Ví dụ, thiết bị video OTT có thể gồm máy chơi trò chơi, máy đọc phát Blu-ray, TV được kết nối Internet, hệ thống rạp hát gia đình, điện thoại thông minh, PCT máy tính bảng, và đầu ghi video kỹ thuật số (Digital Video Recorder, DVR).

Ngoài ra, phương pháp xử lý mà cho nó sáng chế được áp dụng có thể được tạo ra dưới dạng chương trình được thực thi bởi máy tính và có thể được lưu trữ trong phương tiện ghi đọc được bởi máy tính. Dữ liệu đa phương tiện có cấu trúc dữ liệu theo sáng chế cũng có thể được lưu trữ trong phương tiện ghi đọc được bởi máy tính. Phương tiện ghi đọc được bởi máy tính gồm tất cả các loại thiết bị lưu trữ và các thiết bị lưu trữ được phân phối trong đó dữ liệu đọc được bởi máy tính được lưu trữ. Phương tiện ghi đọc được bởi máy tính có thể gồm, ví dụ, đĩa Blu-ray (BD), bus nối tiếp vạn năng (universal serial bus, USB), ROM, PROM, EPROM, EEPROM, RAM, CD-ROM, băng từ, đĩa mềm, và thiết bị lưu trữ dữ liệu quang. Phương tiện ghi đọc được bởi máy tính cũng gồm phương tiện được triển khai dưới dạng sóng mang (ví dụ, truyền qua Internet). Ngoài ra, luồng bit được tạo ra bởi phương pháp mã hóa có thể được lưu trữ trong phương tiện ghi đọc được bởi máy tính hoặc được truyền thông qua mạng truyền thông có dây hoặc không dây.

Ngoài ra, (các) phương án của sáng chế có thể được triển khai dưới dạng sản

phẩm chương trình máy tính dựa trên mã chương trình, và mã chương trình có thể được thực thi trên máy tính theo (các) phương án của sáng chế. Mã chương trình có thể được lưu trữ trên vật mang đọc được bởi máy tính.

Fig.14 biểu diễn ví dụ của hệ thống phát luồng nội dung mà cho nó phương án của sáng chế có thể được áp dụng.

Tham khảo đến Fig.14, hệ thống phát luồng nội dung mà cho nó các phương án của sáng chế được áp dụng thường có thể gồm máy chủ mã hóa, máy chủ phát luồng, máy chủ web, bộ lưu trữ phương tiện, thiết bị người dùng, và thiết bị nhập đa phương tiện.

Máy chủ mã hóa thực hiện chức năng nén tới dữ liệu kỹ thuật số các nội dung được nhập từ các thiết bị đầu vào đa phương tiện, chẳng hạn như điện thoại thông minh, camera, máy quay video và tương tự, để tạo ra luồng bit, và để truyền nó vào máy chủ phát luồng. Theo một ví dụ khác, trong trường hợp trong đó thiết bị đầu vào đa phương tiện, chẳng hạn như, điện thoại thông minh, camera, máy quay video hoặc tương tự, trực tiếp tạo ra luồng bit, máy chủ mã hóa có thể được bỏ qua.

Luồng bit có thể được tạo ra bằng phương pháp mã hóa hoặc phương pháp tạo ra luồng bit mà cho nó các phương án của sáng chế được áp dụng. Và máy chủ phát luồng có thể lưu trữ tạm thời luồng bit trong quy trình truyền hoặc nhận luồng bit.

Máy chủ phát luồng truyền dữ liệu đa phương tiện tới thiết bị người dùng trên cơ sở của yêu cầu của người dùng thông qua máy chủ web, mà đóng vai trò là phương tiện mà thông tin cho người dùng về việc dịch vụ là gì. Khi người dùng yêu cầu dịch vụ mà người dùng muốn, máy chủ web truyền yêu cầu tới máy chủ phát luồng, và máy chủ phát luồng truyền dữ liệu đa phương tiện tới người dùng. Theo đó, hệ thống phát luồng nội dung có thể gồm máy chủ điều khiển tách biệt, và trong trường hợp này, máy chủ điều khiển đóng vai trò điều khiển các lệnh/các phản hồi giữa thiết bị tương ứng trong hệ thống phát luồng nội dung.

Máy chủ phát luồng có thể nhận các nội dung từ bộ lưu trữ phương tiện và/hoặc máy chủ mã hóa. Ví dụ, trong trường hợp các nội dung được nhận từ máy chủ mã hóa, các nội dung có thể được nhận trong thời gian thực. Trong trường hợp này, máy chủ phát luồng có thể lưu trữ luồng bit trong khoảng thời gian được xác định trước để cung cấp dịch vụ phát luồng trơn tru.

Ví dụ, thiết bị người dùng có thể gồm điện thoại di động, điện thoại thông minh, máy tính xách tay, thiết bị đầu cuối phát rộng kỹ thuật số, trợ lý kỹ thuật số cá nhân (personal digital assistant, PDA), máy phát đa phương tiện cầm tay (portable multimedia player, PMP), bộ điều hướng, PC dạng tấm, PCT máy tính bảng, máy tính ultrabook, thiết bị đeo được (ví dụ, đầu cuối kiểu đồng hồ (đồng hồ thông minh), đầu cuối kiểu kính (kính thông minh), bộ hiển thị gắn trên đầu (head mounted display, HMD)), TV kỹ thuật số, máy tính để bàn, biên báo kỹ thuật số hoặc tương tự.

Mỗi trong số các máy chủ trong hệ thống phát luồng nội dung có thể được thao tác như máy chủ được phân phối, và trong trường hợp này, dữ liệu được nhận bởi mỗi máy chủ có thể được xử lý theo cách được phân phối.

Các điểm yêu cầu bảo hộ được mô tả tại đây có thể được tổ hợp theo các cách khác nhau. Ví dụ, các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ dạng phương pháp của sáng chế có thể được tổ hợp và được triển khai dưới dạng thiết bị, và các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ dạng thiết bị của sáng chế có thể được tổ hợp và được triển khai dưới dạng phương pháp. Ngoài ra, các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ phương pháp của sáng chế và các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ thiết bị có thể được kết hợp để được thực hiện dưới dạng thiết bị, và các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ phương pháp và các dấu hiệu kỹ thuật của các điểm yêu cầu bảo hộ thiết bị của sáng chế có thể được kết hợp và được thực hiện dưới dạng phương pháp.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã, phương pháp này bao gồm các bước:

thu nhận thông tin hình ảnh gồm thông tin phần dư và thông tin cho phép bỏ qua biến đổi từ luồng bit;

thu nhận thông tin cờ bỏ qua biến đổi dựa trên giá trị của thông tin cho phép bỏ qua biến đổi;

xác định liệu việc bỏ qua biến đổi có được áp dụng cho khối hiện tại hay không dựa trên thông tin cờ bỏ qua biến đổi;

dẫn xuất mẫu phần dư dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không và thông tin phần dư; và

tạo ra mẫu được xây dựng lại dựa trên mẫu phần dư,

trong đó thông tin hình ảnh gồm thông tin cho phép lập mã bảng màu,

trong đó dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi hoặc thông tin cho phép lập mã bảng màu, thông tin hình ảnh ngoài ra còn gồm thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được phép tối thiểu cho chế độ bỏ qua biến đổi,

trong đó dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi là 1 hoặc giá trị của thông tin cho phép lập mã bảng màu là 1, thông tin thông số lượng tử hóa tối thiểu được gồm trong thông tin hình ảnh, và trong đó dựa trên điều kiện rằng cả giá trị của thông tin cho phép bỏ qua biến đổi và giá trị của thông tin cho phép lập mã bảng màu là 0, thông tin thông số lượng tử hóa tối thiểu không được gồm trong thông tin hình ảnh,

trong đó dựa trên việc bỏ qua biến đổi được áp dụng cho khối hiện tại hoặc chế độ bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho khối hiện tại được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu, và

trong đó dựa trên việc bỏ qua biến đổi được áp dụng cho khối hiện tại, mẫu phần dư được dẫn xuất dựa trên thông số lượng tử hóa, và dựa trên việc chế độ bảng màu được áp dụng cho khối hiện tại, giá trị thoát được dẫn xuất dựa trên thông số lượng tử hóa.

2. Phương pháp mã hóa hình ảnh được thực hiện bởi thiết bị mã hóa, phương pháp này bao gồm các bước:

xác định liệu việc bỏ qua biến đổi có được áp dụng cho khối hiện tại hay không dựa trên thông tin cho phép bỏ qua biến đổi;

tạo ra thông tin cờ bỏ qua biến đổi dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không;

tạo ra thông tin phần dư trên khối hiện tại dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không; và

mã hóa thông tin hình ảnh gồm thông tin cho phép bỏ qua biến đổi và thông tin phần dư,

trong đó thông tin hình ảnh gồm thông tin cho phép lập mã bảng màu,

trong đó dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi hoặc thông tin cho phép lập mã bảng màu, thông tin hình ảnh ngoài ra còn gồm thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được phép tối thiểu cho chế độ bỏ qua biến đổi,

trong đó dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi là 1 hoặc giá trị của thông tin cho phép lập mã bảng màu là 1, thông tin thông số lượng tử hóa tối thiểu được gồm trong thông tin hình ảnh, và trong đó dựa trên điều kiện rằng cả

giá trị của thông tin cho phép bỏ qua biến đổi và giá trị của thông tin cho phép lập mã bảng màu là 0, thông tin thông số lượng tử hóa tối thiểu không được gồm trong thông tin hình ảnh, và

trong đó dựa trên việc bỏ qua biến đổi được áp dụng cho khối hiện tại hoặc chế độ bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho khối hiện tại được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu.

3. Phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp lưu trữ luồng bit được tạo ra bởi phương pháp, phương pháp này bao gồm các bước: thực hiện việc xác định liệu việc bỏ qua biến đổi có được áp dụng cho khối hiện tại hay không dựa trên thông tin cho phép bỏ qua biến đổi, tạo ra thông tin cờ bỏ qua biến đổi dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không, tạo ra thông tin phần dư trên khối hiện tại dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không, và tạo ra luồng bit bằng việc mã hóa thông tin hình ảnh gồm thông tin cho phép bỏ qua biến đổi và thông tin phần dư,

trong đó thông tin hình ảnh gồm thông tin cho phép lập mã bảng màu,

trong đó dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi hoặc thông tin cho phép lập mã bảng màu, thông tin hình ảnh ngoài ra còn gồm thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được phép tối thiểu cho chế độ bỏ qua biến đổi,

trong đó dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi là 1 hoặc giá trị của thông tin cho phép lập mã bảng màu là 1, thông tin thông số lượng tử hóa tối thiểu được gồm trong thông tin hình ảnh, và trong đó dựa trên điều kiện rằng cả giá trị của thông tin cho phép bỏ qua biến đổi và giá trị của thông tin cho phép lập mã bảng màu là 0, thông tin thông số lượng tử hóa tối thiểu không được gồm trong thông tin hình ảnh, và

trong đó dựa trên việc bỏ qua biến đổi được áp dụng cho khối hiện tại hoặc chế độ bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho khối hiện tại được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu.

4. Phương pháp truyền dữ liệu cho hình ảnh, phương pháp truyền này bao gồm các bước:

thu nhận luồng bit, trong đó luồng bit được tạo ra bằng cách thực hiện việc xác định liệu việc bỏ qua biến đổi có được áp dụng cho khối hiện tại hay không dựa trên thông tin cho phép bỏ qua biến đổi, tạo ra thông tin cờ bỏ qua biến đổi dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không, tạo ra thông tin phần dư trên khối hiện tại dựa trên việc liệu việc bỏ qua biến đổi có được áp dụng hay không, và tạo ra luồng bit bằng việc mã hóa thông tin hình ảnh gồm thông tin cho phép bỏ qua biến đổi và thông tin phần dư; và

truyền dữ liệu bao gồm luồng bit,

trong đó thông tin hình ảnh gồm thông tin cho phép lập mã bảng màu,

trong đó dựa trên ít nhất một trong số thông tin cho phép bỏ qua biến đổi hoặc thông tin cho phép lập mã bảng màu, thông tin hình ảnh ngoài ra còn gồm thông tin thông số lượng tử hóa tối thiểu liên quan tới thông số lượng tử hóa được phép tối thiểu cho chế độ bỏ qua biến đổi,

trong đó dựa trên điều kiện rằng giá trị của thông tin cho phép bỏ qua biến đổi là 1 hoặc giá trị của thông tin cho phép lập mã bảng màu là 1, thông tin thông số lượng tử hóa tối thiểu được gồm trong thông tin hình ảnh, và trong đó dựa trên điều kiện rằng cả giá trị của thông tin cho phép bỏ qua biến đổi và giá trị của thông tin cho phép lập mã bảng màu là 0, thông tin thông số lượng tử hóa tối thiểu không được gồm trong thông tin hình ảnh, và

trong đó dựa trên việc bỏ qua biến đổi được áp dụng cho khối hiện tại hoặc chế độ bảng màu được áp dụng cho khối hiện tại, thông số lượng tử hóa cho khối hiện tại được dẫn xuất dựa trên thông tin thông số lượng tử hóa tối thiểu.

FIG. 1

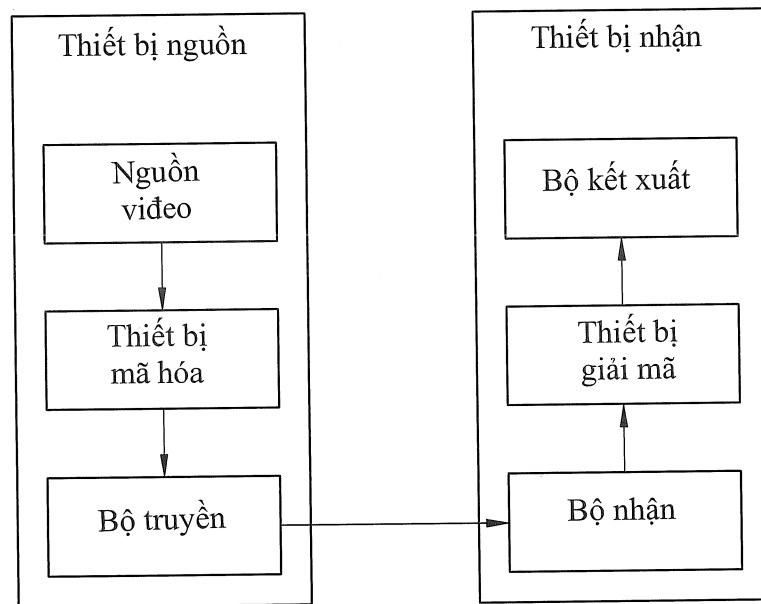


FIG. 2

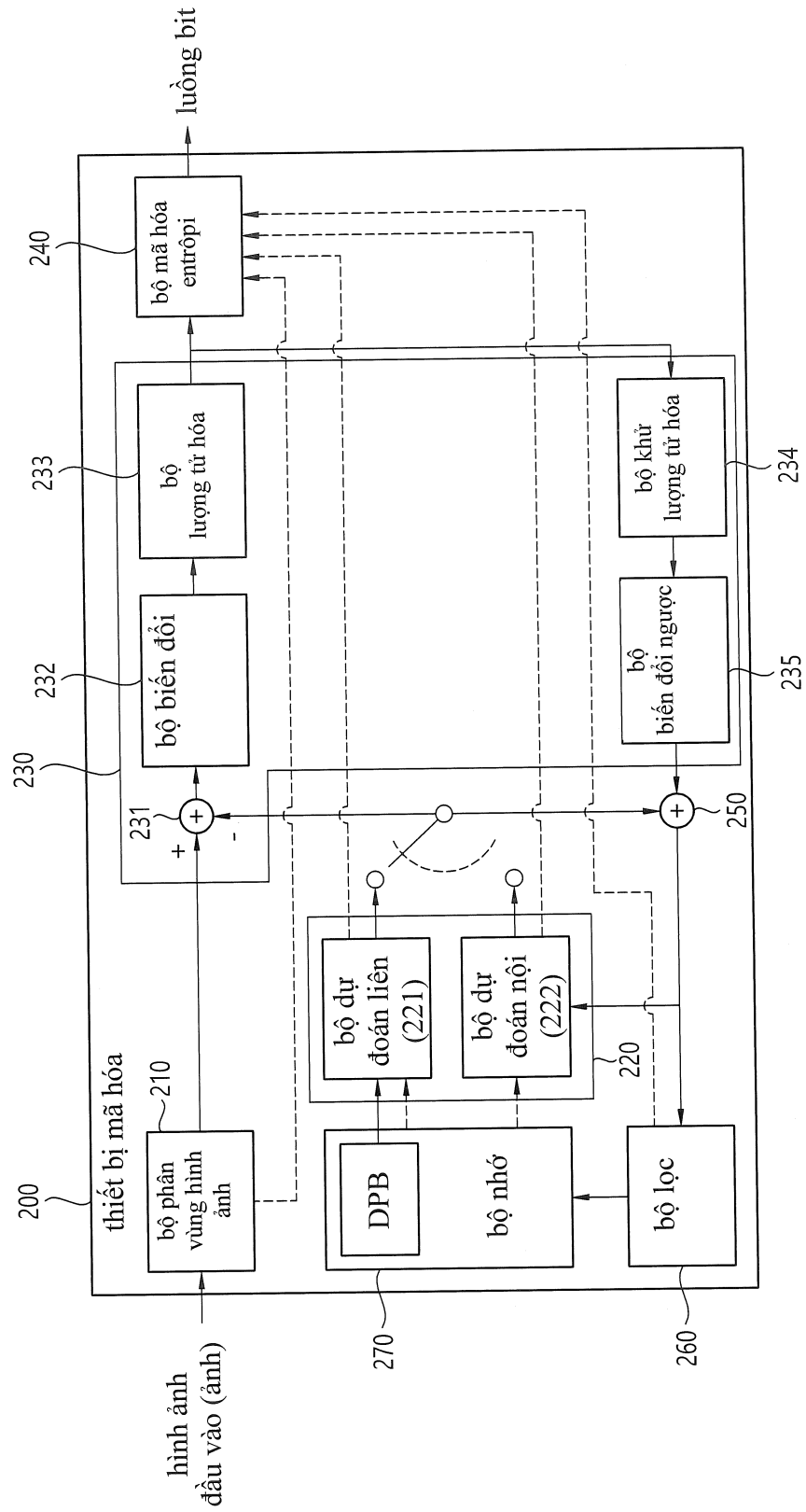


FIG. 3

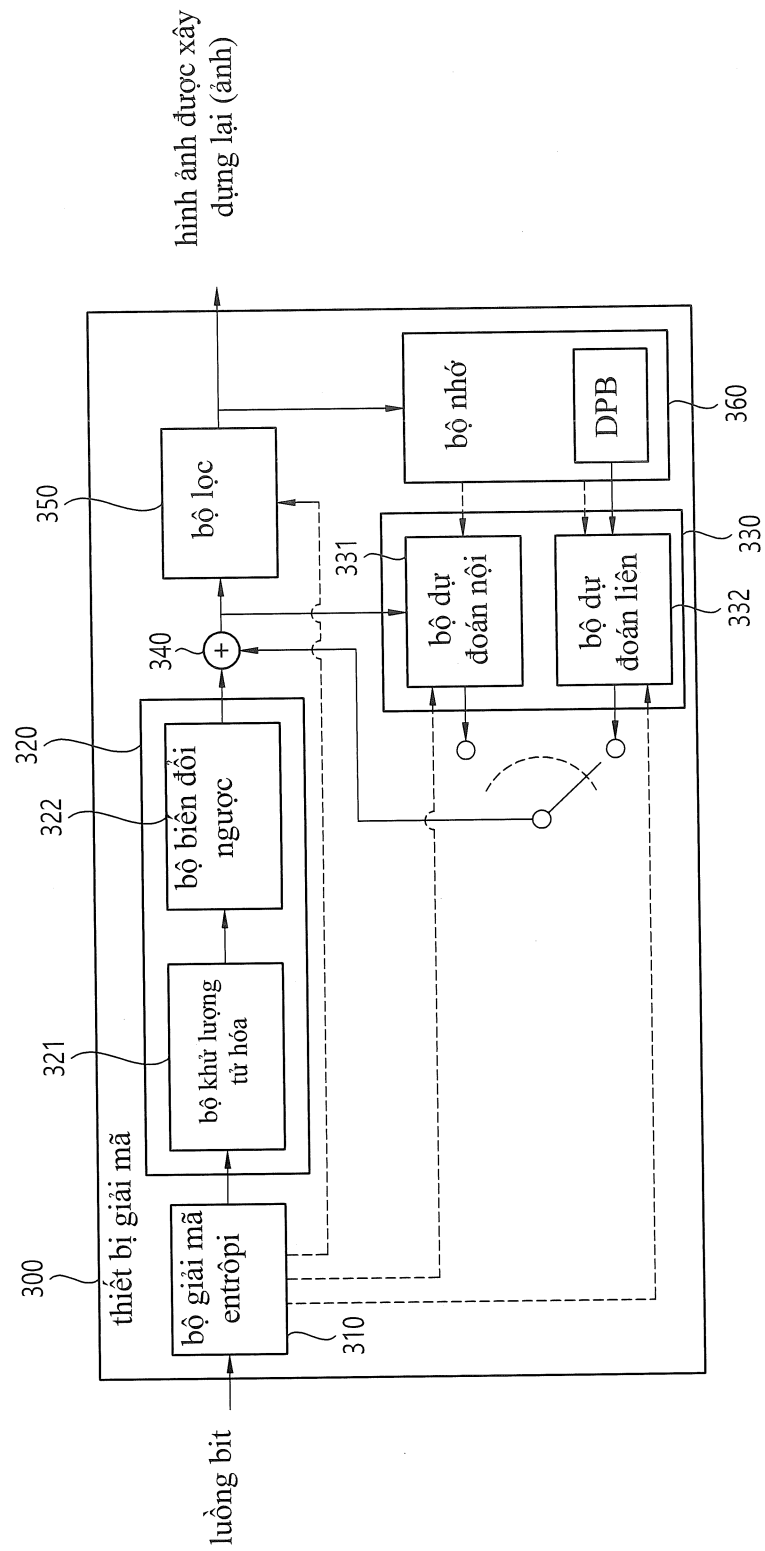


FIG. 4

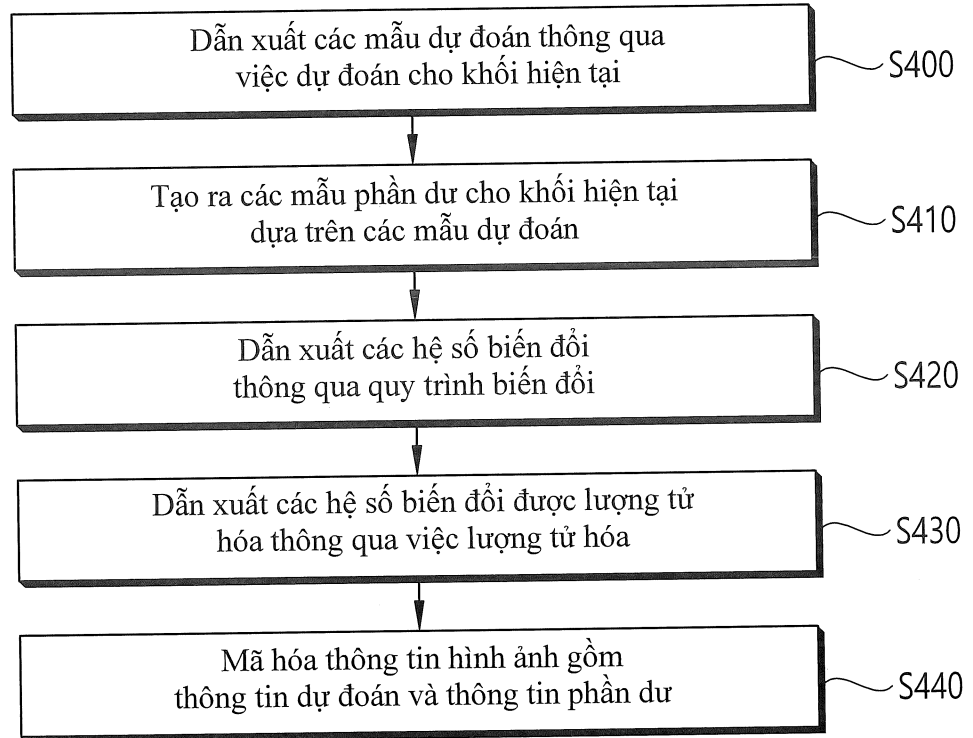


FIG. 5

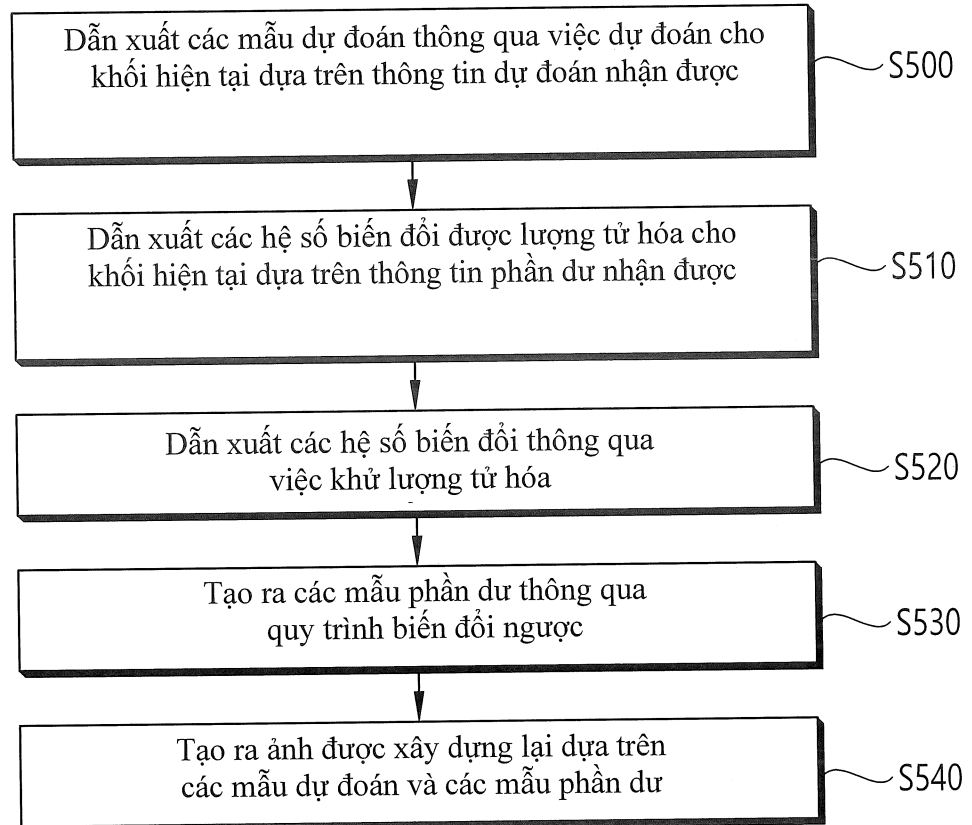


FIG. 6

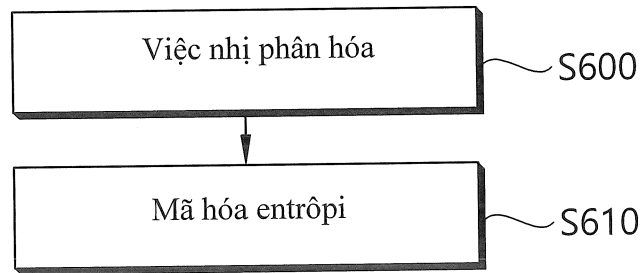


FIG. 7

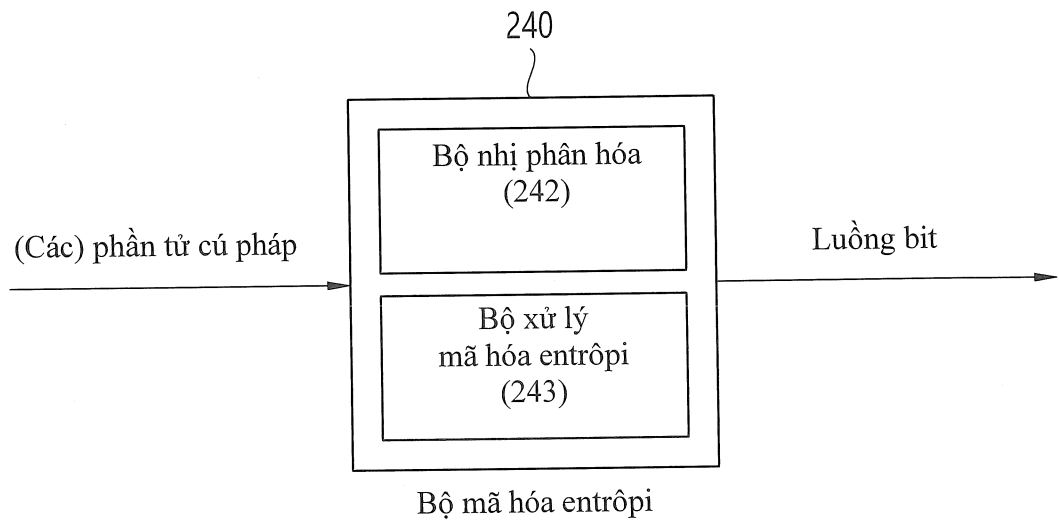


FIG. 8

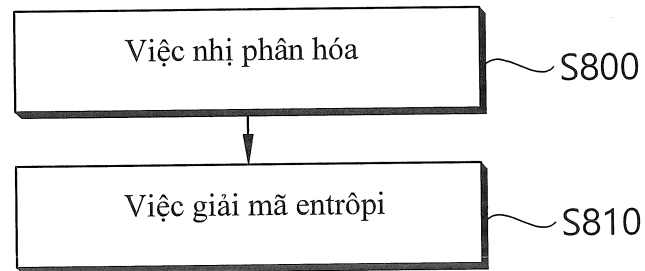


FIG. 9

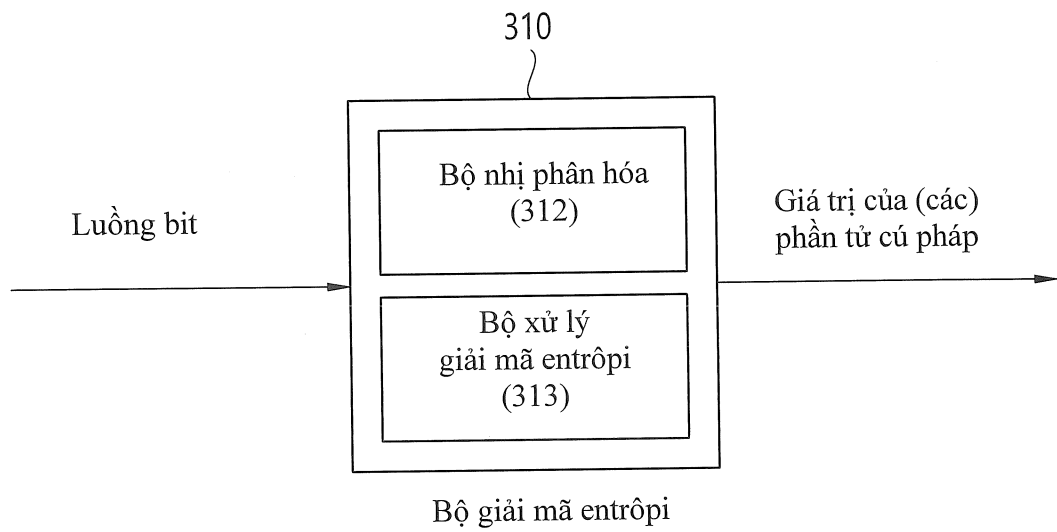


FIG. 10

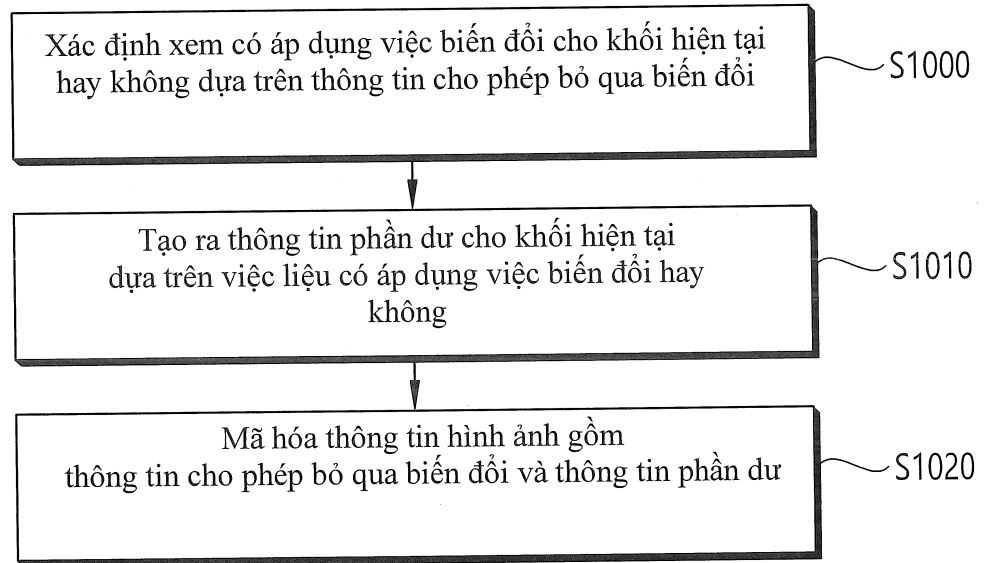
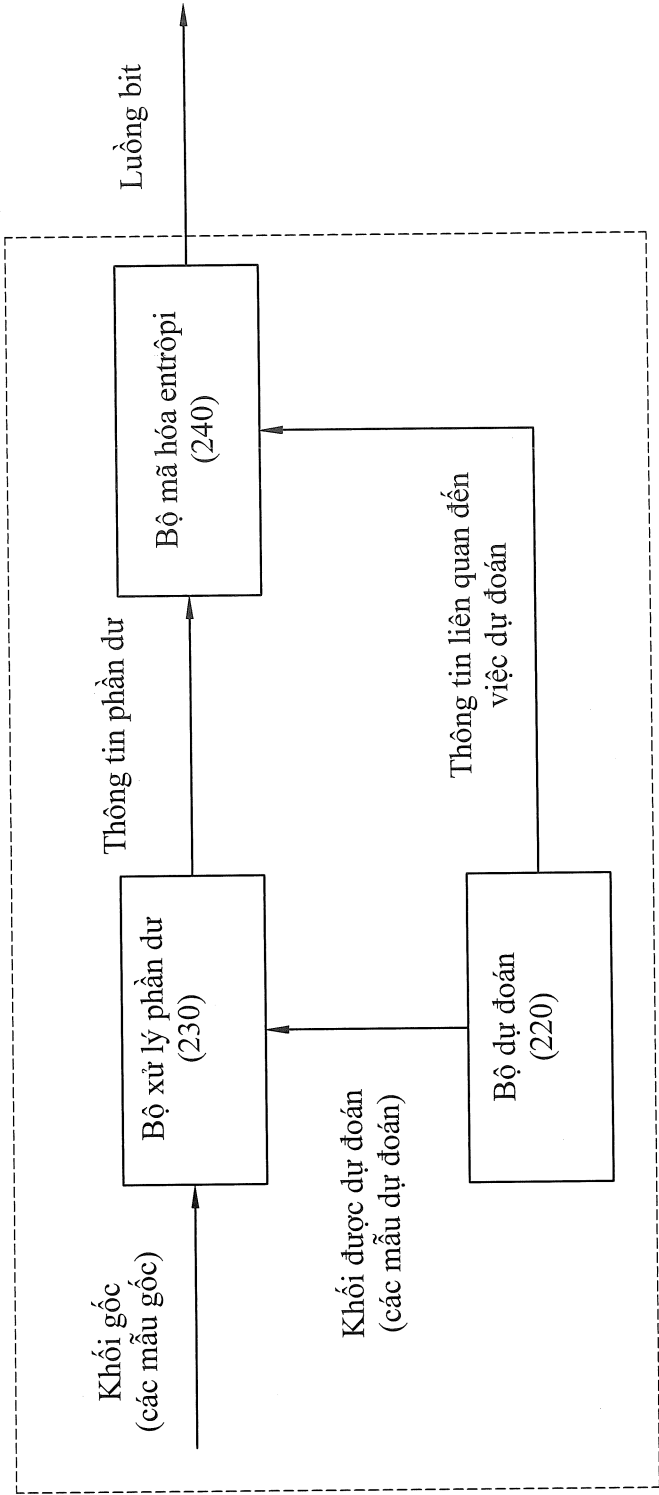


FIG. 11



Thiết bị mã hóa (200)

FIG. 12

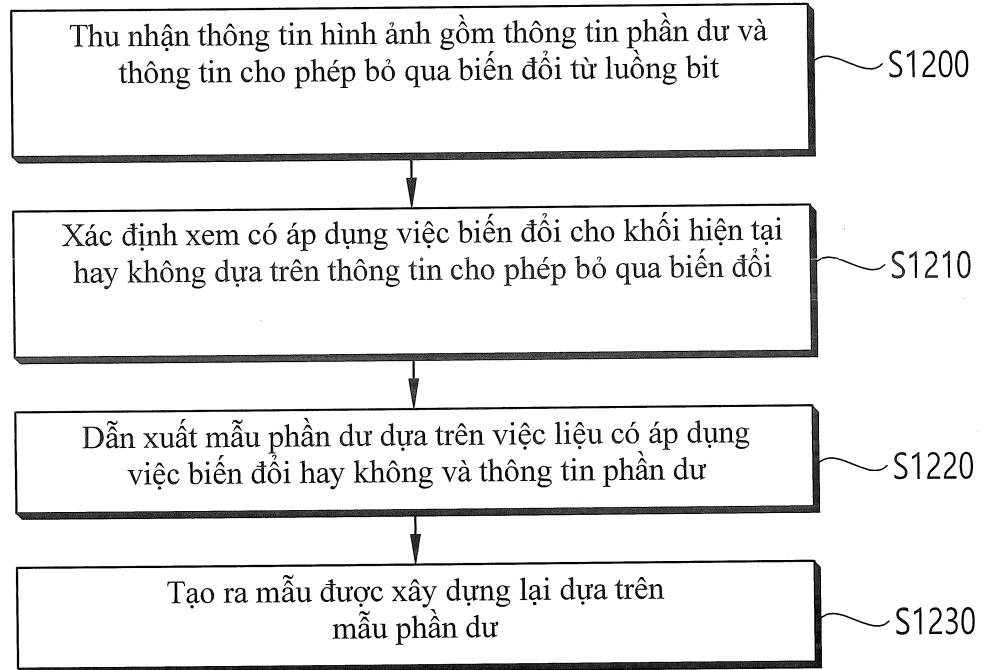


FIG. 13

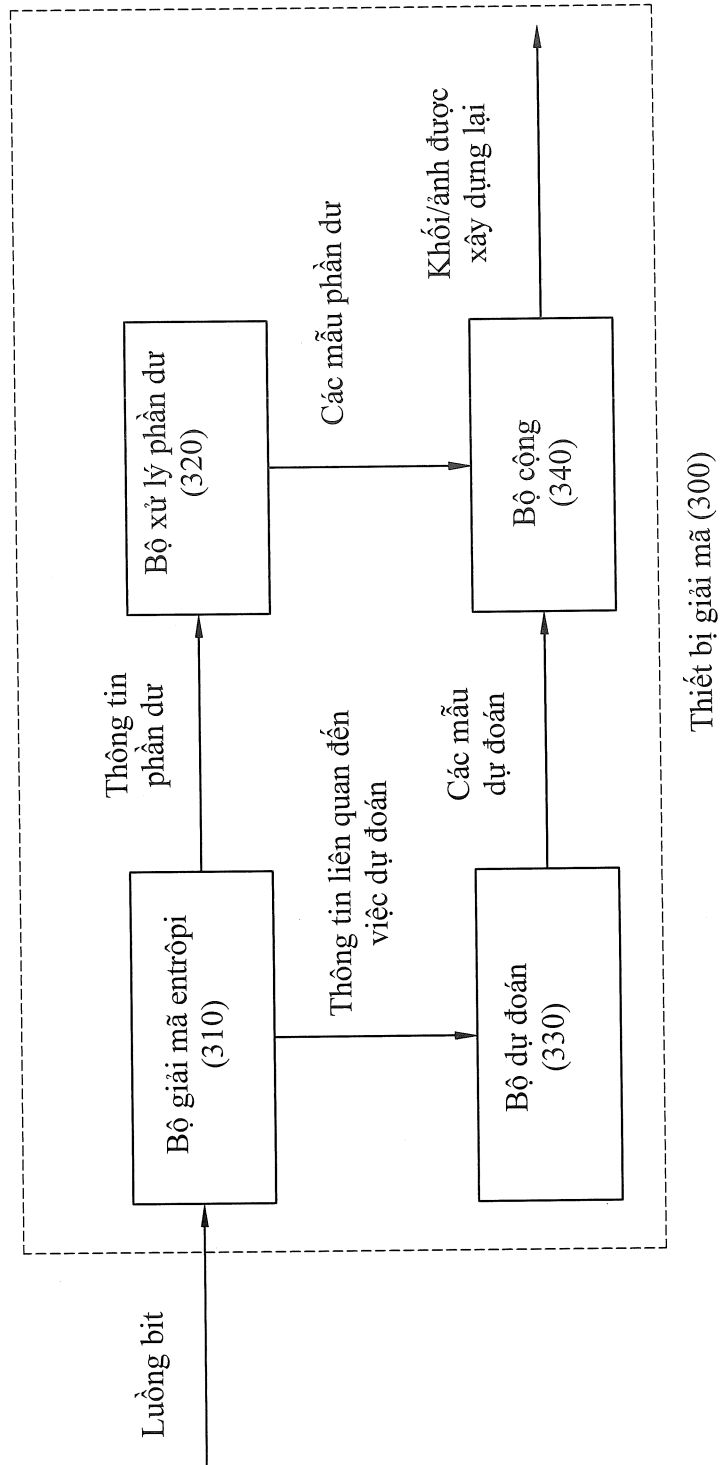


FIG. 14

