



- (12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ 1-0052173
(51)^{2021.01} H04N 19/132; H04N 19/12; H04N 19/18; H04N 19/176; H04N 19/119 (13) B

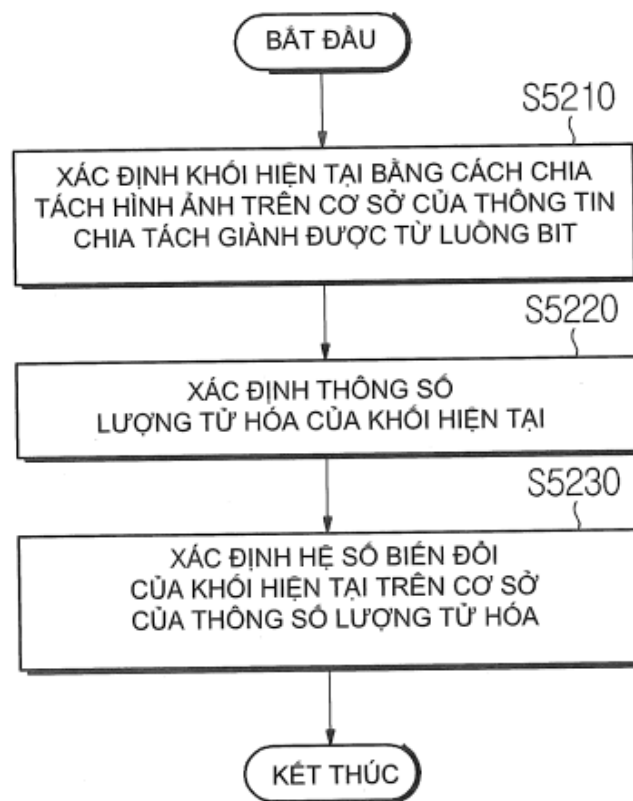
-
- (21) 1-2022-03217 (22) 28/10/2020
(86) PCT/KR2020/014845 28/10/2020 (87) WO2021/086021 06/05/2021
(30) 62/927,113 28/10/2019 US; 62/931,786 06/11/2019 US; 62/939,533 22/11/2019 US;
62/957,128 04/01/2020 US
(45) 27/10/2025 451 (43) 25/07/2022 412A
(73) LG ELECTRONICS INC. (KR)
128, Yeoui-daero, Yeongdeungpo-gu Seoul 07336, Republic of Korea
(72) ZHAO, Jie (KR); KIM, Seung Hwan (KR); HENDRY, Hendry (KR); PALURI,
Seethal (KR).
(74) Công ty Luật TNHH T&G (TGVN)
-

- (54) PHƯƠNG PHÁP MÃ HÓA/GIẢI MÃ HÌNH ẢNH, PHƯƠNG PHÁP TRUYỀN
LUỒNG BIT, VÀ PHƯƠNG TIỆN GHI ĐỌC ĐƯỢC BỞI MÁY TÍNH KHÔNG
CHUYỂN TIẾP

(21) 1-2022-03217

(57) Sáng chế đề cập đến phương pháp và thiết bị mã hóa/giải mã hình ảnh được đề xuất. Phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã hình ảnh, theo sáng chế, bao gồm các bước: xác định khối hiện tại bằng cách chia tách hình ảnh trên cơ sở của thông tin chia tách giành được từ luồng bit; xác định thông số lượng tử hóa của khối hiện tại; và xác định hệ số biến đổi của khối hiện tại trên cơ sở của thông số lượng tử hóa. Sáng chế cũng đề cập đến phương pháp truyền luồng bit và phương tiện ghi đọc được bởi máy tính không chuyển tiếp.

FIG. 52



Lĩnh vực kỹ thuật được đề cập

Sáng chế liên quan đến phương pháp và thiết bị mã hóa/giải mã hình ảnh. Cụ thể hơn là, sáng chế liên quan đến phương pháp và thiết bị mã hóa/giải mã hình ảnh nhờ sử dụng sự biến đổi thích ứng, và phương pháp truyền luồng bit được tạo ra bởi phương pháp/thiết bị mã hóa hình ảnh của sáng chế.

Tình trạng kỹ thuật của sáng chế

Hiện tại, nhu cầu về hình ảnh độ phân giải cao và chất lượng cao như các hình ảnh độ phân giải cao (High Definition, HD) và các hình ảnh độ phân giải siêu cao (Ultra High Definition, UHD) đang tăng trong các lĩnh vực khác nhau. Độ phân giải và chất lượng của dữ liệu hình ảnh được cải thiện, và lượng thông tin được truyền hoặc các bit tăng tương đối khi so sánh với dữ liệu hình ảnh sẵn có. Việc tăng của lượng thông tin hoặc bit được truyền tạo ra chi phí truyền và chi phí lưu trữ.

Theo đó, cần phải có kỹ thuật nén hình ảnh hiệu suất cao hoặc việc truyền, lưu trữ, và tái tạo thông tin một cách hiệu quả trên các hình ảnh có độ phân giải cao và chất lượng cao.

Bản chất kỹ thuật của sáng chế

Vấn đề kỹ thuật

Sáng chế được định hướng để đề xuất phương pháp và thiết bị mã hóa/giải mã hình ảnh với hiệu quả mã hóa/giải mã được cải thiện.

Bên cạnh đó, sáng chế được định hướng để đề xuất phương pháp và thiết bị mã hóa/giải mã hình ảnh để cải thiện hiệu quả mã hóa/giải bằng cách thực hiện sự biến đổi thích ứng.

Bên cạnh đó, sáng chế này được định hướng để đề xuất phương pháp truyền luồng bit được tạo ra bởi phương pháp hoặc thiết bị mã hóa hình ảnh theo sáng chế.

Bên cạnh đó, sáng chế này được định hướng là để đề xuất phương tiện ghi lưu trữ luồng bit được tạo ra bởi phương pháp hoặc thiết bị mã hóa hình ảnh hoặc theo sáng chế.

Bên cạnh đó, sáng chế này được định hướng là để đề xuất phương tiện ghi lưu trữ luồng bit được nhận, được giải mã và được sử dụng để tái tạo hình ảnh bởi thiết bị giải mã hình ảnh theo sáng chế.

Những người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng sẽ hiểu rõ rằng các mục đích kỹ thuật cần đạt được trong sáng chế không bị giới hạn ở các mục đích kỹ thuật được đề cập ở trên và các mục đích kỹ thuật khác vốn không được mô tả ở đây sẽ được hiểu một cách rõ ràng nhờ phần mô tả sau đây.

Giải pháp kỹ thuật

Theo khía cạnh, sáng chế đề xuất phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã hình ảnh, phương pháp gồm các bước: xác định khối hiện tại bằng cách chia tách hình ảnh trên cơ sở của thông tin chia tách giành được từ luồng bit; xác định thông số lượng tử hóa của khối hiện tại; và xác định hệ số biến đổi của khối hiện tại trên cơ sở của thông số lượng tử hóa. Bước xác định của thông số lượng tử hóa có thể được thực hiện trên cơ sở của việc liệu sự biến đổi không gian màu sắc có được áp dụng cho mẫu dư của khối hiện tại hay không, và giá trị của thông số lượng tử hóa có

thể được xác định là có giá trị trong phạm vi mà được thiết lập là từ giới hạn trên được xác định trước đến giới hạn dưới được xác định trước hay không.

Bên cạnh đó, theo khía cạnh, sáng chế đề xuất thiết bị giải mã hình ảnh gồm bộ nhớ và ít nhất một bộ xử lý, trong đó ít nhất một bộ xử lý được tạo cấu hình để xác định khối hiện tại bằng cách chia tách hình ảnh trên cơ sở của thông tin chia tách giành được từ luồng bit, xác định thông số lượng tử hóa của khối hiện tại, và xác định hệ số biến đổi của khối hiện tại trên cơ sở của thông số lượng tử hóa, và bộ xử lý được tạo cấu hình để xác định thông số lượng tử hóa trên cơ sở của việc liệu sự biến đổi không gian màu sắc có được áp dụng cho mẫu dư của khối hiện tại hay không. Ở đây, giá trị của thông số lượng tử hóa có thể được xác định là giá trị trong phạm vi mà được thiết lập là từ giới hạn trên được xác định trước đến giới hạn dưới được xác định trước.

Bên cạnh đó, theo khía cạnh, sáng chế đề xuất phương pháp mã hóa hình ảnh được thực hiện bởi thiết bị mã hóa hình ảnh, phương pháp gồm các bước: xác định khối hiện tại bằng cách chia tách hình ảnh; xác định thông số lượng tử hóa của khối hiện tại; và xác định hệ số biến đổi của khối hiện tại trên cơ sở của thông số lượng tử hóa. Ở đây, bước xác định của thông số lượng tử hóa có thể được thực hiện trên cơ sở của việc liệu sự biến đổi không gian màu sắc có được áp dụng cho mẫu dư của khối hiện tại hay không, và giá trị của thông số lượng tử hóa có thể được xác định là có giá trị trong phạm vi mà được thiết lập là từ giới hạn trên được xác định trước đến giới hạn dưới được xác định trước hay không.

Bên cạnh đó, theo một khía cạnh khác, sáng chế đề xuất phương pháp truyền để truyền luồng bit được tạo ra bởi thiết bị mã hóa hình ảnh hoặc phương pháp mã hóa hình ảnh của sáng chế.

Bên cạnh đó, theo một khía cạnh khác, sáng chế đề xuất phương tiện ghi được được bởi máy tính lưu trữ ở trong nó luồng bit được tạo ra bởi phương pháp mã hóa hình ảnh hoặc thiết bị mã hóa hình ảnh của sáng chế.

Các dấu hiệu được tóm tắt ngắn gọn ở trên đối với sáng chế này chỉ đơn thuần là các khía cạnh làm ví dụ của phần mô tả chi tiết dưới đây của sáng chế này, và không làm hạn chế phạm vi bảo hộ của sáng chế này.

Các ưu điểm của sáng chế

Sáng chế được định hướng để đề xuất phương pháp và thiết bị mã hóa/giải mã hình ảnh với hiệu quả mã hóa/giải mã được cải thiện.

Bên cạnh đó, theo một khía cạnh khác, sáng chế đề xuất phương pháp truyền để truyền luồng bit được tạo ra bởi thiết bị mã hóa hình ảnh hoặc phương pháp mã hóa hình ảnh của sáng chế.

Bên cạnh đó, theo một khía cạnh khác, sáng chế đề xuất phương tiện ghi được được bởi máy tính lưu trữ ở trong nó luồng bit được tạo ra bởi phương pháp mã hóa hình ảnh hoặc thiết bị mã hóa hình ảnh của sáng chế.

Các dấu hiệu được tóm tắt ngắn gọn ở trên đối với sáng chế này chỉ đơn thuần là các khía cạnh làm ví dụ của phần mô tả chi tiết dưới đây của sáng chế này, và không làm hạn chế phạm vi bảo hộ của sáng chế này.

Các ưu điểm của sáng chế

Sáng chế có thể đề xuất phương pháp và thiết bị mã hóa/giải mã hình ảnh với hiệu quả mã hóa/giải mã được cải thiện.

Bên cạnh đó, sáng chế có thể đề xuất phương pháp và thiết bị mã hóa/giải mã hình ảnh có khả năng cải thiện hiệu quả mã hóa/giải mã bằng cách thực hiện sự biến đổi thích ứng.

Bên cạnh đó, sáng chế có thể đề xuất phương pháp truyền luồng bit được tạo ra bởi phương pháp hoặc thiết bị mã hóa hình ảnh hoặc theo sáng chế.

Bên cạnh đó, sáng chế có thể đề xuất phương tiện ghi lưu trữ luồng bit được tạo ra bởi phương pháp hoặc thiết bị mã hóa hình ảnh hoặc theo sáng chế.

Bên cạnh đó, sáng chế có thể đề xuất phương tiện ghi lưu trữ luồng bit được nhận, được giải mã và được sử dụng để tái tạo hình ảnh bởi thiết bị giải mã hình ảnh theo sáng chế.

Người có hiểu biết trung bình trong lĩnh vực kỹ thuật sẽ hiểu rằng có thể đạt được các hiệu quả này nhờ sáng chế, và các hiệu quả này không bị hạn chế vào phần đã được mô tả một cách cụ thể đây và ở trên và các ưu điểm khác của sáng chế này sẽ được hiểu rõ ràng hơn nhờ phần mô tả sau đây.

Mô tả vắn tắt các hình vẽ

Fig.1 là hình thể hiện một cách sơ lược hệ thống tạo mã video, mà phương án của sáng chế có thể áp dụng được vào đó.

Fig.2 là hình thể hiện một cách sơ lược thiết bị mã hóa hình ảnh, mà phương án của sáng chế có thể áp dụng được vào đó.

Fig.3 là hình thể hiện một cách sơ lược thiết bị giải mã hình ảnh, mà phương án của sáng chế có thể áp dụng được vào đó.

Fig.4 là hình thể hiện cấu trúc phân vùng của hình ảnh theo phương án.

Fig.5 là hình thể hiện phương án của loại phân vùng của khối theo cấu trúc cây nhiều loại.

Fig.6 là hình thể hiện cơ chế phát tín hiệu của thông tin chia tách khối trong cấu trúc của cây tứ phân với cây nhiều loại được lồng vào nhau theo sáng chế.

Fig.7 là hình thể hiện phương án mà trong đó CTU được phân vùng thành nhiều CU.

Fig.8 là hình thể hiện các mẫu tham chiếu lân cận theo phương án.

Fig.9 và Fig.10 là các hình thể hiện việc dự đoán trong ảnh theo phương án.

Fig.11 là hình thể hiện sơ đồ khối của CABAC theo phương án để mã hóa một phần tử cú pháp.

Các hình vẽ từ Fig.12 đến Fig.15 là các hình thể hiện việc mã hóa và giải mã entropi theo phương án.

Fig.16 và Fig.17 là các hình thể hiện các ví dụ về các thủ tục giải mã và mã hóa ảnh theo phương án.

Fig.18 là hình thể hiện cấu trúc phân cấp cho hình ảnh được tạo mã theo phương án.

Fig.19 là hình thể hiện phương án của quy trình giải mã mà ACT được áp dụng vào đó.

Fig.20 là hình thể hiện phương án của bảng cú pháp tập hợp thông số chuỗi mà trong đó phần tử cú pháp liên quan đến ACT được phát tín hiệu.

Các hình vẽ từ Fig.21 đến Fig.27 là các hình thể hiện một cách liên tiếp phương án của bảng cú pháp của cơ sở mã hóa mà trong đó phần tử cú pháp liên quan đến ACT được phát tín hiệu.

Fig.28 là hình thể hiện cú pháp cây tạo mã theo phương án.

Fig.29 là hình thể hiện phương pháp mã hóa mẫu dư của BDPCM theo sáng chế.

Fig.30 là hình thể hiện khối dư được lượng tử hóa mà được sửa đổi được tạo ra bằng cách thực hiện BDPCM theo phương án.

Fig.31 là lưu đồ thể hiện thủ tục để mã hóa khối hiện tại bằng cách áp dụng BDPCM trong thiết bị mã hóa hình ảnh theo phương án.

Fig.32 là lưu đồ minh họa thủ tục để tái tạo khối hiện tại bằng cách áp dụng BDPCM trong thiết bị giải mã hình ảnh theo phương án.

Các hình vẽ từ Fig.33 đến Fig.35 là các hình thể hiện một cách sơ lược cú pháp để phát tín hiệu thông tin về BDPCM.

Các hình vẽ từ Fig.36 đến Fig.51 là các hình thể hiện bảng cú pháp để phát tín hiệu phần tử cú pháp ACT theo mỗi phương án riêng lẻ của sáng chế.

Fig.52 là hình thể hiện phương pháp giải mã hình ảnh theo phương án.

Fig.53 là hình thể hiện phương pháp mã hóa hình ảnh theo phương án.

Fig.54 là hình thể hiện hệ thống phát luồng nội dung, mà phương án của sáng chế có thể áp dụng được vào đó.

Mô tả chi tiết sáng chế

Ở dưới đây, các phương án làm ví dụ của sáng chế sẽ được mô tả chi tiết có dựa vào các hình vẽ kèm theo để được triển khai một cách dễ dàng bởi người có hiểu biết

trung bình trong lĩnh vực kỹ thuật tương ứng. Tuy nhiên, sáng chế này có thể được triển khai trong các dạng khác nhau, và không bị hạn chế ở các phương án được mô tả ở đây.

Trong việc mô tả sáng chế này, nếu nó được xác định rằng phần mô tả chi tiết của chức năng hoặc kết cấu đã biết liên quan tạo nên phạm vi của sáng chế này là mơ hồ một cách không cần thiết, thì phần mô tả chi tiết của chúng sẽ được lược bỏ. Trong các hình vẽ, các phần không liên quan tới phần mô tả của sáng chế này sẽ được lược bỏ, và các số chỉ dẫn tương tự được gán cho các phần tương tự.

Trong sáng chế này, khi thành phần “được kết nối”, “được ghép nối” hoặc “được liên kết” tới thành phần khác, thì nó có thể không chỉ gồm quan hệ kết nối trực tiếp mà cũng có thể gồm quan hệ kết nối gián tiếp mà trong đó thành phần xen kẽ là có mặt. Bên cạnh đó, khi thành phần “gồm” hoặc “có” các thành phần khác, thì nó có nghĩa là các thành phần khác có thể còn được gồm, hơn là loại trừ các thành phần khác, trừ khi được tuyên bố theo cách khác.

Trong sáng chế này, các thuật ngữ thứ nhất, thứ hai, v.v. có thể được sử dụng chỉ cho mục đích phân biệt một thành phần với các thành phần khác, và không làm hạn chế thứ tự hoặc mức độ quan trọng của các thành phần trừ khi được tuyên bố theo cách khác. Theo đó, nằm trong phạm vi của sáng chế này, thành phần thứ nhất theo một phương án có thể được gọi là thành phần thứ hai trong một phương án khác, và theo cách tương tự, thành phần thứ hai trong một phương án có thể được gọi là thành phần thứ nhất trong một phương án khác.

Trong sáng chế này, các thành phần được phân biệt với nhau nhằm mô tả một cách rõ ràng mỗi dấu hiệu, và không có nghĩa là các thành phần nhất thiết phải được tách rời. Nghĩa là, nhiều thành phần có thể được tích hợp và được triển khai trong một đơn vị phần cứng hoặc phần mềm, hoặc một thành phần có thể được phân bố và được triển khai

trong nhiều đơn vị phần cứng hoặc phần mềm. Do đó, ngay cả nếu không được tuyên bố theo cách khác, thì các phương án như vậy mà trong đó các thành phần được tích hợp hoặc thành phần được phân bố cũng được gồm trong phạm vi bảo hộ của sáng chế này.

Trong sáng chế này, các thành phần được mô tả trong nhiều phương án không nhất thiết phải có nghĩa là các thành phần thiết yếu, và một số thành phần có thể là các thành phần tùy chọn. Theo đó, phương án gồm có tập hợp con của các thành phần được mô tả trong phương án cũng được gồm trong phạm vi bảo hộ của sáng chế này. Bên cạnh đó, phương án gồm các thành phần khác bên cạnh các thành phần được mô tả trong các phương án khác nhau cũng được gồm trong phạm vi bảo hộ của sáng chế này.

Sáng chế liên quan đến việc mã hóa và giải mã của hình ảnh, và các thuật ngữ được sử dụng trong sáng chế này có thể có nghĩa chung, được sử dụng chung trong lĩnh vực kỹ thuật, mà sáng chế này thuộc vào đó, trừ khi được định rõ theo kiểu mới trong sáng chế này.

Trong sáng chế, “video” có thể nghĩa là tập hợp của một loạt các hình ảnh theo thời gian. “Ảnh” nói chung có nghĩa là cơ sở biểu diễn một hình ảnh trong khoảng thời gian cụ thể, và lát/phiến là cơ sở giải mã cấu thành một phần của ảnh trong việc mã hóa. Một ảnh có thể gồm có một hoặc nhiều lát/phiến. Bên cạnh đó, lát/phiến có thể gồm một hoặc nhiều đơn vị cây tạo mã (Coding Tree Unit, CTU). Một ảnh có thể gồm có một hoặc nhiều lát/phiến. Một ảnh có thể gồm có một hoặc nhiều nhóm phiến. Một nhóm phiến có thể gồm một hoặc nhiều phiến. Viên gạch có thể đề cập đến khu vực hình chữ nhật của các hàng CTU trong phiến ở trong ảnh. Một nhóm phiến có thể gồm một hoặc nhiều phiến. Viên gạch có thể đề cập đến khu vực hình chữ nhật của các hàng CTU trong phiến. Một phiến có thể được phân vùng thành nhiều viên gạch, và mỗi viên gạch

có thể gồm một hoặc nhiều hàng CTU thuộc về phiên. Phiên không được phân vùng thành nhiều viên gạch cũng có thể được xử lý như là viên gạch.

Trong sáng chế này, “điểm ảnh” (pixel) hoặc “pel” có thể có nghĩa là đơn vị nhỏ nhất cấu thành một ảnh (hoặc hình ảnh). Bên cạnh đó, “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 biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần độ chói hoặc chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ.

Trong sáng chế này, “đơn vị” có thể biểu diễn đơn vị cơ bản của việc xử lý hình ảnh. Đơn vị có thể gồm ít nhất một 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 “mảng mẫu”, “khối” hoặc “khu vực” trong một số trường hợp. Trong trường hợp chung, khối $M \times N$ có thể gồm các mẫu (hoặc các mảng mẫu) hoặc tập hợp (hoặc mảng) của các hệ số biến đổi gồm M cột và N hàng.

Trong sáng chế này, “khối hiện tại” có thể có nghĩa là một trong số “khối tạo mã hiện tại”, “đơn vị tạo mã hiện tại”, “khối mục tiêu tạo mã”, “khối mục tiêu giải mã” hoặc “khối mục tiêu xử lý”. Khi việc dự đoán được thực hiện, thì “khối hiện tại” có thể nghĩa là “khối dự đoán hiện tại” hoặc “khối mục tiêu dự đoán”. Khi việc biến đổi (biến đổi ngược)/lượng tử hóa (khử lượng tử hóa) được thực hiện, thì “khối hiện tại” có thể có nghĩa là “khối biến đổi hiện tại” hoặc “khối mục tiêu biến đổi”. Khi việc lọc được thực hiện, thì “khối hiện tại” có thể nghĩa là “khối mục tiêu lọc”.

Bên cạnh đó, trong sáng chế này, “khối hiện tại” có thể có nghĩa là “khối độ chói của khối hiện tại” trừ khi được tuyên bố một cách rõ ràng là khối sắc độ. “Khối sắc độ

của khối hiện tại” có thể được trình bày bằng cách gồm phần mô tả rõ ràng của khối sắc độ, như “khối sắc độ” hoặc “khối hiện tại”.

Trong sáng chế này, thì ký tự “/” và “,” nên được diễn giải là để chỉ ra “và/hoặc”. Ví dụ, cách trình bày “A/B” và “A, B” có thể có nghĩa là “A và/hoặc B.” Hơn nữa, “A/B/C” và “A/B/C” có thể có nghĩa là “ít nhất một thành phần trong số A, B, và/hoặc C.”

Trong sáng chế này, thuật ngữ “hoặc” nên được diễn giải là để chỉ ra “và/hoặc.” Ví dụ, biểu diễn “A hoặc B” có thể bao gồm 1) chỉ “A”, 2) chỉ “B”, và/hoặc 3) cả “A và B”. Nói cách khác, trong sáng chế này, thuật ngữ “hoặc” nên được diễn giải là để chỉ ra “theo cách bổ sung hoặc theo cách thay thế.”

Tổng quan của hệ thống tạo mã video

Fig.1 là hình thể hiện hệ thống tạo mã video theo sáng chế.

Hệ thống tạo mã video theo phương án có thể gồm thiết bị nguồn 10 và thiết bị nhận 20. Thiết bị nguồn 10 có thể phân phối thông tin video và/hoặc hình ảnh được mã hóa hoặc dữ liệu đến thiết bị nhận 20 trong dạng tệp hoặc phát luồng thông qua phương tiện lưu trữ kỹ thuật số hoặc mạng.

Thiết bị nguồn 10 theo phương án có thể gồm bộ tạo ra nguồn video 11, thiết bị mã hóa 12 và bộ truyền 13. Thiết bị nhận 20 theo phương án có thể gồm bộ nhận 21, thiết bị giải mã 12 và bộ kết xuất 23. Thiết bị mã hoá 12 có thể được gọi là thiết bị mã hoá video/hình ảnh, và thiết bị giải mã 12 có thể được gọi là thiết bị giải mã video/hình ảnh. Bộ truyền 13 có thể được gồm trong thiết bị mã hóa 12. Bộ nhận 21 có thể được bao gồm trong thiết bị giải mã 12. Bộ kết xuất 23 có thể gồm bộ hiển thị, bộ hiển thị có thể được tạo cấu hình như là thiết bị tách biệt hoặc thành phần ngoài.

Bộ tạo ra nguồn video 11 có thể giành được video/hình ảnh thông qua quy trình chụp, tổng hợp, hoặc tạo ra video/hình ảnh. Bộ tạo ra nguồn video 11 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, bộ phận lưu trữ video/hình ảnh gồm video/các hình ảnh 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 video/các hình ảnh (theo cách điện tử). 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 12 có thể mã hóa video/hình ảnh đầu vào. Thiết bị mã hóa 12 có thể thực hiện một loạt các thủ tục như là việc dự đoán, biến đổi, và lượng tử hóa cho hiệu quả nén và tạo mã. Thiết bị mã hóa 12 có thể xuất ra dữ liệu được mã hóa (thông tin video/hình ảnh đã được mã hóa) trong dạng luồng bit.

Bộ truyền 12 có thể truyền hình ảnh/thông tin hình ảnh được mã hóa hoặc dữ liệu được xuất ra trong dạng luồng bit đến bộ nhận 21 của thiết bị nhận 20 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 13 có thể gồm phần tử để tạo ra tệp phương tiện thông qua định dạng tệp được xác định trước và có thể gồm thành phần để truyền qua mạng phát rộng/mạng truyền thông. Bộ nhận 21 có thể trích xuất/nhận luồng bit từ phương tiện lưu trữ hoặc mạng và truyền luồng bit đến thiết bị giải mã 12.

Thiết bị giải mã 12 có thể giải mã video/hình ảnh bằng cách thực hiện một loạt các quy trình như là khử lượng tử hóa, biến đổi ngược, và dự đoán tương ứng với hoạt động của thiết bị mã hóa 12.

Bộ kết xuất 23 có thể kết xuất video/hình ảnh được giải mã. Video/hình ảnh được kết xuất có thể được hiển thị thông qua bộ hiển thị.

Tổng quan của thiết bị mã hóa hình ảnh

Fig.2 là hình thể hiện một cách sơ lược thiết bị mã hóa hình ảnh, mà phương án của sáng chế có thể áp dụng được vào đó.

Như được thể hiện trên Fig.2, thiết bị mã hóa hình ảnh 100 có thể gồm bộ phân vùng hình ảnh 110, bộ trừ 115, bộ biến đổi 120, bộ lượng tử hóa 130, bộ khử lượng tử hóa 140, bộ biến đổi ngược 150, bộ cộng 155, bộ lọc 160, bộ nhớ 170, đơn vị dự đoán liên ảnh 180, đơn vị dự đoán trong ảnh 185 và bộ mã hóa entropi 190. Đơn vị dự đoán liên ảnh 180 và đơn vị dự đoán trong ảnh 185 có thể được gọi chung là “đơn vị dự đoán”. Bộ biến đổi 120, bộ lượng tử hóa 130, bộ khử lượng tử hóa 140 và bộ biến đổi ngược 150 có thể được gồm trong bộ xử lý phần dư. Bộ xử lý phần dư có thể còn gồm bộ trừ 115.

Tất cả hoặc ít nhất một số thành phần trong nhiều thành phần tạo cấu hình thiết bị mã hóa hình ảnh 100 có thể được tạo cấu hình bởi một thành phần phần cứng (ví dụ, bộ mã hóa hoặc bộ xử lý) trong một số phương án. Bên cạnh đó, bộ nhớ 170 có thể gồm bộ đệm ảnh được giải mã (Decoded Picture Buffer, DPB) và có thể được tạo cấu hình bởi phương tiện lưu trữ kỹ thuật số.

Bộ phân vùng hình ảnh 110 có thể phân vùng hình ảnh đầu vào (hoặc hình ảnh hoặc khung) được nhập vào thiết bị mã hóa hình ảnh 100 thành một hoặc nhiều đơn vị xử lý. Ví dụ, đơn vị xử lý có thể được gọi là đơn vị tạo mã (Coding Unit, CU). Đơn vị tạo mã có thể giành được bằng cách phân vùng theo kiểu đệ quy đơn vị cây tạo mã (Coding Tree Unit, CTU) hoặc đơn vị tạo mã lớn nhất (Largest Coding Unit, LCU) theo cấu trúc cây tứ phân cây nhị phân cây tam phân (Quad-Tree Binary-Tree Ternary-Tree,

QT/BT/TT). Ví dụ, một đơn vị tạo mã có thể được phân vùng thành nhiều đơn vị tạo mã có độ sâu sâu hơn dựa trên cấu trúc cây tứ phân, cấu trúc cây nhị phân, và/hoặc cấu trúc tam phân. Đối với việc phân vùng của đơn vị tạo mã, thì cấu trúc cây tứ phân có thể được áp dụng trước tiên và cấu trúc cây nhị phân và/hoặc cấu trúc tam phân có thể được áp dụng sau. Thủ tục tạo mã theo sáng chế có thể được thực hiện dựa trên đơn vị tạo mã cuối cùng mà không còn được phân vùng nữa. Đơn vị tạo mã lớn nhất có thể được sử dụng như là đơn vị tạo mã cuối cùng hoặc đơn vị tạo mã có độ sâu sâu hơn, giành được bằng cách phân vùng đơn vị tạo mã lớn nhất có thể được sử dụng làm đơn vị tạo mã cuối cùng. Tại đây, thủ tục tạo mã có thể gồm thủ tục dự đoán, biến đổi, và tái tạo, vốn sẽ được mô tả sau. Theo một ví dụ khác, đơn vị xử lý của thủ tục tạo mã có thể là đơn vị dự đoán (Prediction Unit, PU) hoặc đơn vị biến đổi (Transform Unit, TU). Đơn vị dự đoán và đơn vị biến đổi được chia tách hoặc phân vùng từ đơn vị tạo mã cuối cùng. Đơn vị dự đoán có thể là đơn vị của mẫu dự đoán, và đơn vị biến đổi có thể là đơn vị để dẫn xuất hệ số biến đổi và/hoặc đơn vị để dẫn xuất tín hiệu dư từ hệ số biến đổi.

Đơn vị dự đoán (đơn vị dự đoán liên ảnh 180 hoặc đơn vị dự đoán trong ảnh 185) có thể thực hiện việc dự đoán trên khối cần được xử lý (khối hiện tại) và tạo ra khối được dự đoán gồm các mẫu dự đoán cho khối hiện tại. Đơn vị dự đoán có thể xác định xem liệu việc dự đoán trong ảnh hay việc dự đoán liên ảnh được áp dụng trên cơ sở khối hiện tại hay CU. Đơn vị dự đoán có thể tạo ra thông tin khác nhau liên quan đến việc dự đoán của khối hiện tại và truyền thông tin được tạo ra đến bộ mã hóa entropi 190. Thông tin về việc dự đoán có thể được mã hóa trong bộ mã hóa entropi 190 và được xuất ra trong dạng luồng bit.

Đơn vị dự đoán trong ảnh 185 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt trong khu lân

cận của khối hiện tại hoặc có thể được đặt tách ra theo chế độ dự đoán trong ảnh và/hoặc kỹ thuật dự đoán trong ảnh. Các chế độ dự đoán trong ảnh có thể gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Các chế độ không hướng có thể gồm, ví dụ, chế độ DC và 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 mức độ chi tiết của hướng dự đoán. Tuy nhiên, đây chỉ đơn thuần là ví dụ, nhiều hoặc ít chế độ dự đoán có hướng hơn có thể được sử dụng phụ thuộc vào việc thiết lập. Đơn vị dự đoán trong ảnh 185 có thể xác định chế độ dự đoán được áp dụng cho khối hiện tại bằng cách sử dụng chế độ dự đoán được áp dụng cho khối lân cận.

Đơn vị dự đoán liên ảnh 180 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bởi vectơ chuyển động trên ảnh tham chiếu. Trong trường hợp này, để làm giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên ảnh, thì thông tin chuyển động có thể được dự đoán trong các đơn vị của các khối, các khối con, hoặc các 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 ảnh (dự đoán L0, dự đoán L1, dự đoán đôi, v.v.). Trong trường hợp dự đoán liên ảnh, 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 có thể là giống nhau hoặc khác nhau. Khối lân cận theo thời gian có thể được gọi là khối tham chiếu được đặt cùng một chỗ, CU được đặt cùng một chỗ (co-located CU, colCU), và tương tự. Ảnh tham chiếu gồm khối lân cận theo thời gian có thể được gọi là ảnh được đặt cùng một chỗ (collocated picture, col Pic). Ví dụ, đơn vị dự đoán liên ảnh 180 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 vectơ chuyển động và/hoặc chỉ số ảnh tham chiếu của khối hiện tại. Việc dự đoán liên ảnh có thể được thực hiện dựa trên các chế độ dự đoán khác nhau. Ví dụ, trong trường hợp của chế độ bỏ qua và chế độ hợp nhất, đơn vị dự đoán liên ảnh 180 có thể sử dụng thông tin chuyển động của khối lân cận làm thông tin chuyển động của khối hiện tại. Trong trường hợp của chế độ bỏ qua, không giống như chế độ hợp nhất, tín hiệu dư có thể không được truyền. Trong trường hợp của chế độ dự đoán vectơ chuyển động (Motion Vector Prediction, MVP), vectơ chuyển động của khối lân cận có thể được sử dụng làm bộ dự đoán vectơ chuyển động và vectơ chuyển động của khối hiện tại có thể được phát tín hiệu bằng cách mã hóa sự chênh lệch vectơ chuyển động và bộ chỉ ra cho bộ dự đoán vectơ chuyển động. Sự chênh lệch vectơ chuyển động có thể nghĩa là sự chênh lệch giữa vectơ chuyển động của khối hiện tại và bộ dự đoán vectơ chuyển động.

Đơn vị dự đoán có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp dự đoán và các kỹ thuật dự đoán khác nhau được mô tả dưới đây. Ví dụ, đơn vị dự đoán có thể không chỉ áp dụng việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh mà còn áp dụng một cách đồng thời cả việc dự đoán trong ảnh và việc dự đoán liên ảnh, để dự đoán khối hiện tại. Phương pháp dự đoán để áp dụng một cách đồng thời cả việc dự đoán trong ảnh và việc dự đoán liên ảnh cho việc dự đoán của khối hiện tại có thể được gọi là việc dự đoán liên ảnh và trong ảnh được tổ hợp (Combined Inter And Intra Prediction, CIIP). Bên cạnh đó, đơn vị dự đoán có thể thực hiện việc sao chép khối trong ảnh (Intra block Copy, IBC) cho việc dự đoán của khối hiện tại. Việc sao chép khối trong ảnh có thể được sử dụng cho việc mã hóa hình ảnh/video nội dung của trò chơi hoặc tương tự, ví dụ, việc tạo mã nội dung màn hình (Screen Content Coding, SCC). IBC là phương pháp để dự đoán ảnh hiện tại nhờ sử dụng khối tham chiếu được tái tạo trước đó trong ảnh hiện tại tại vị trí tách ra khỏi khối hiện tại theo khoảng cách được xác định trước. Khi

IBC được áp dụng, thì vị trí của khối tham chiếu ở trong ảnh hiện tại có thể được mã hóa như là véc tơ (véc tơ khối) tương ứng với khoảng cách được xác định trước. IBC về cơ bản thực hiện việc dự đoán trong ảnh hiện tại nhưng cũng có thể được thực hiện theo cách tương tự như việc dự đoán liên ảnh mà trong đó khối tham chiếu được dẫn xuất trong ảnh hiện tại. Nghĩa là, IBC có thể sử dụng ít nhất một kỹ thuật trong số các kỹ thuật dự đoán liên ảnh được mô tả trong sáng chế này.

Tín hiệu dự đoán được tạo ra bởi đơn vị dự đoán có thể được sử dụng để tạo ra tín hiệu được tái tạo hoặc để tạo ra tín hiệu dư. Bộ trừ 115 có thể tạo ra tín hiệu dư (khối dư hoặc mảng mẫu dư) bằng cách trừ tín hiệu dự đoán (khối được dự đoán hoặc mảng mẫu dự đoán) được xuất ra từ đơn vị dự đoán khỏi tín hiệu hình ảnh đầu vào (khối gốc hoặc mảng mẫu gốc). Tín hiệu dư được tạo ra có thể được truyền đến bộ biến đổi 120.

Bộ biến đổi 120 có thể tạo ra các hệ số biến đổi bằng cách áp dụng kỹ thuật biến đổi cho tín hiệu dư. Ví dụ, kỹ thuật biến đổi có thể gồm ít nhất một 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 có điều kiện (Conditionally Non-linear Transform, CNT). Tại đây, GBT có nghĩa là sự biến đổi thu nhận được từ đồ thị khi thông tin mối quan hệ giữa các điểm ảnh được biểu diễn bởi đồ thị. CNT đề cập đến sự biến đổi giành được dựa trên tín hiệu dự đoán được tạo ra nhờ sử dụng tất cả các điểm ảnh được tái tạo trước đó. Bên cạnh đó, quy trình biến đổi có thể được áp dụng cho các khối điểm ảnh hình vuông có cùng kích cỡ hoặc có thể được áp dụng cho các khối có kích cỡ có thể thay đổi được khác hình vuông.

Bộ lượng tử hóa 130 có thể lượng tử hóa các hệ số biến đổi và truyền chúng đến bộ mã hóa entropi 190. Bộ mã hóa entropi 190 có thể mã hóa tín hiệu được lượng tử hóa

(thông tin về các hệ số biến đổi được lượng tử hóa) và xuất ra 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 130 có thể sắp xếp lại các hệ số biến đổi được lượng tử khóa trong dạng khối thành dạng vectơ một chiều dựa trên thứ tự quét hệ số và tạo ra thông tin trên các hệ số biến đổi được lượng tử hóa dựa trên các hệ số biến đổi được lượng tử hóa trong dạng vectơ một chiều.

Bộ mã hóa entrôpi 190 có thể thực hiện các phương pháp mã hóa khác nhau như là, ví dụ, Golomb hàm mũ, tạo mã chiều dài biến đổi thích ứng ngữ cảnh (Context-Adaptive Variable Length Coding, CAVLC), tạo mã số học nhị phân thích ứng ngữ cảnh (Context-Adaptive Binary Arithmetic Coding, CABAC), và tương tự. Bộ mã hóa entrôpi 190 có thể mã hóa thông tin cần thiết cho việc tái tạo video/hình ảnh khác với các hệ số biến đổi được lượng tử hóa (ví dụ, các giá trị của các phần tử cú pháp, v.v.) cùng nhau hoặc theo kiểu tách biệt. Thông tin được mã hóa (ví dụ, thông tin video/hình ảnh được mã hóa) có thể được truyền hoặc lưu trữ trong các đơn vị của các lớp trừu tượng mạng (Network Abstraction Layer, NAL) trong dạng luồng bit. Thông tin video/hình ảnh có thể còn gồm thông tin về các tập hợp thông số khác nhau như tập hợp thông số thích ứng (Adaptation Parameter Set, APS), tập hợp thông số ảnh (Picture Parameter Set, PPS), tập hợp thông số chuỗi (Sequence Parameter Set, SPS), hoặc tập hợp thông số video (Video Parameter Set, VPS). Bên cạnh đó, thông tin video/hình ảnh có thể còn gồm thông tin ràng buộc chung. Thông tin được phát tín hiệu, thông tin được truyền và/hoặc các phần tử cú pháp được mô tả trong sáng chế này có thể được mã hóa thông qua thủ tục mã hóa được mô tả ở trên và được gồm trong luồng bit.

Luồng bit có thể được truyền qua mạng hoặc có thể được lưu trữ trong phương tiện lưu trữ kỹ thuật số. Mạng có thể gồm mạng phát rộng và/hoặc mạng truyền thông, và

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 (không được thể hiện) truyền tín hiệu được xuất ra từ bộ mã hóa entropi 190 và/hoặc đơn vị lưu trữ (không được thể hiện) lưu trữ tín hiệu có thể được gồm dưới dạng thành phần trong/ngoài của thiết bị mã hóa hình ảnh 100. Theo cách thay thế, bộ truyền có thể được cung cấp dưới dạng thành phần của bộ mã hóa entropi 190.

Các hệ số biến đổi được lượng tử hóa mà được xuất ra từ bộ lượng tử hóa 130 có thể được sử dụng để tạo ra tín hiệu dư. Ví dụ, tín hiệu dư (khối dư hoặc các mẫu dư) có thể được tái tạo bằng cách áp dụng việc khử lượng tử hóa và biến đổi ngược cho các hệ số biến đổi được lượng tử hóa thông qua bộ khử lượng tử hóa 140 và bộ biến đổi ngược 150.

Bộ cộng 155 cộng tín hiệu dư được tái tạo vào tín hiệu dự đoán được xuất ra từ đơn vị dự đoán liên ảnh 180 hoặc đơn vị dự đoán trong ảnh 185 để tạo ra tín hiệu được tái tạo (ảnh được tái tạo, khối được tái tạo, mảng mẫu được tái tạo). Nếu không có phần dư cho khối cần được xử lý, như trường hợp mà trong đó chế độ bỏ qua được áp dụng, thì khối được dự đoán có thể được sử dụng làm khối được tái tạo. Bộ cộng 155 có thể được gọi là bộ tái tạo hoặc bộ tạo ra khối được tái tạo. Tín hiệu được tái tạo mà được tạo ra có thể được sử dụng cho việc dự đoán trong ảnh của khối tiếp theo cần được xử lý ở trong ảnh hiện tại và có thể được sử dụng cho việc dự đoán liên ảnh của ảnh tiếp theo thông qua việc lọc như được mô tả dưới đây.

Bộ lọc 160 có thể cải thiện chất lượng hình ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái tạo. Ví dụ, bộ lọc 160 có thể tạo ra ảnh được tái tạo mà được sửa đổi bằng cách áp dụng các phương pháp lọc khác nhau cho ảnh được tái tạo và lưu trữ ảnh được tái tạo mà được sửa đổi trong bộ nhớ 170, cụ thể là, DPB của

bộ nhớ 170. Các phương pháp lọc khác nhau có thể gồm, ví dụ, lọc khử khối, bù thích ứng mẫu, bộ lọc vòng lặp thích ứng, bộ lọc song phương, và tương tự. Bộ lọc 160 có thể tạo ra thông tin khác nhau liên quan đến việc lọc và truyền thông tin được tạo ra đến bộ mã hóa entropi 190 như sẽ được mô tả sau 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 190 và xuất ra trong dạng luồng bit.

Ảnh được tái tạo mà được biến đổi sẽ được truyền đến bộ nhớ 170 có thể được sử dụng làm ảnh tham chiếu trong đơn vị dự đoán liên ảnh 180. Khi việc dự đoán liên ảnh được áp dụng qua thiết bị mã hóa hình ảnh 100, thì sự không khớp của việc dự đoán giữa thiết bị mã hóa hình ảnh 100 và thiết bị giải mã hình ảnh có thể được loại bỏ và hiệu suất mã hóa có thể được cải thiện.

DPB của bộ nhớ 170 có thể lưu trữ ảnh được tái tạo mà được biến đổi để sử dụng như là ảnh tham chiếu trong đơn vị dự đoán liên ảnh 180. Bộ nhớ 170 có thể lưu trữ thông tin chuyển động của khối mà từ 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 hiện tại vừa được tái tạo. Thông tin chuyển động được lưu trữ có thể được truyền đến đơn vị dự đoán liên ảnh 180 và 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 170 có thể lưu trữ các mẫu được tái tạo của các khối được tái tạo trong ảnh hiện tại và có thể truyền các mẫu được tái tạo đến đơn vị dự đoán trong ảnh 185.

Tổng quan của thiết bị giải mã hình ảnh

Fig.3 là hình thể hiện một cách sơ lược thiết bị giải mã hình ảnh, mà phương án của sáng chế có thể áp dụng được vào đó.

Như được thể hiện trên Fig.3, thiết bị giải mã hình ảnh 200 có thể gồm bộ giải mã entropi 210, bộ khử lượng tử hóa 220, bộ biến đổi ngược 230, bộ cộng 235, bộ lọc 240, bộ nhớ 250, đơn vị dự đoán liên ảnh 260 và đơn vị dự đoán trong ảnh 265. Đơn vị dự đoán liên ảnh 260 và đơn vị dự đoán trong ảnh 265 có thể được gọi chung là “đơn vị dự đoán”. Bộ khử lượng tử hóa 220 và bộ biến đổi ngược 230 có thể được gồm trong bộ xử lý phần dư.

Tất cả hoặc ít nhất một số thành phần trong nhiều thành phần tạo cấu hình thiết bị giải mã hình ảnh 200 có thể được tạo cấu hình bởi một thành phần phần cứng (ví dụ, bộ giải mã hoặc bộ xử lý) theo phương án. Bên cạnh đó, bộ nhớ 250 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.

Thiết bị giải mã hình ảnh 200, vốn đã nhận luồng bit gồm thông tin video/hình ảnh, có thể tái tạo hình ảnh bằng cách thực hiện quy trình tương ứng với quy trình được thực hiện bởi thiết bị mã hóa hình ảnh 100 trên Fig.2. Ví dụ, thiết bị giải mã hình ảnh 200 có thể thực hiện việc giải mã nhờ sử dụng đơn vị xử lý được áp dụng trong thiết bị mã hóa hình ảnh. Vì thế, đơn vị xử lý của việc giải mã có thể là đơn vị tạo mã, ví dụ. Đơn vị tạo mã có thể giành được bằng cách phân vùng đơn vị cây tạo mã hoặc đơn vị tạo mã lớn nhất. Tín hiệu hình ảnh được tái tạo mà được giải mã và được xuất ra thông qua thiết bị giải mã hình ảnh 200 có thể được tái tạo qua thiết bị tái tạo (không được thể hiện).

Thiết bị giải mã hình ảnh 200 có thể nhận tín hiệu được xuất ra từ thiết bị mã hóa trên Fig.2 trong dạng luồng bit. Tín hiệu nhận được có thể được giải mã thông qua bộ giải mã entropi 210. Ví dụ, bộ giải mã entropi 210 có thể phân tích cú pháp luồng bit để dẫn xuất thông tin (ví dụ, thông tin video/hình ảnh) cần thiết cho việc tái tạo hình ảnh (hoặc tái tạo ảnh). Thông tin video/hình ảnh có thể còn gồm thông tin về các tập hợp

thông số khác nhau như là tập hợp thông số thích ứng (APS), tập hợp thông số ảnh (PPS), tập hợp thông số chuỗi (SPS), hoặc tập hợp thông số video (VPS). Bên cạnh đó, thông tin video/hình ảnh có thể còn gồm thông tin ràng buộc chung. Thiết bị giải mã hình ảnh còn có thể giải mã ảnh dựa trên thông tin về bộ thông số và/hoặc thông tin ràng buộc chung. Thông tin được phát tín hiệu/nhận được và/hoặc các phần tử cú pháp được mô tả trong sáng chế 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 210 giải mã thông tin trong luồng bit dựa trên phương pháp tạo mã như là tạo mã Golomb hàm mũ, CAVLC, hoặc CABAC, và xuất ra các giá trị của các phần tử cú pháp cần thiết cho việc tái tạo hình ảnh và các giá trị được lượng tử hóa của các hệ số biến đổi cho phần dư. Một cách cụ thể, phương pháp giải mã entropi CABAC có thể nhận ngán tương ứng với với phần tử cú pháp trong luồng bit, xác định mô hình ngữ cảnh nhờ sử dụng thông tin phần tử cú pháp đích giải mã, thông tin giải mã của khối lân cận và khối mục tiêu giải mã hoặc thông tin về ký hiệu/ngán được giải mã trong giai đoạn trước đó, và thực hiện việc giải mã số học trên ngán bằng cách dự đoán khả năng xuất hiện của ngán theo mô hình ngữ cảnh được xác định, và tạo ra ký hiệu tương ứng với giá trị của mỗi phần tử cú pháp. Trong trường hợp này, phương pháp giải mã entropi CABAC có thể cập nhật mô hình ngữ cảnh bằng cách sử dụng thông tin về ký hiệu/ngán được giải mã cho mô hình ngữ cảnh của ký hiệu/ngán tiếp theo sau khi xác định mô hình ngữ cảnh. Thông tin liên quan đến việc dự đoán trong số thông tin được giải mã bởi bộ giải mã entropi 210 có thể được cung cấp cho đơn vị dự đoán (đơn vị dự đoán liên ảnh 260 và đơn vị dự đoán trong ảnh 265), và giá trị dư mà việc giải mã entropi được thực hiện trên đó trong bộ giải mã entropi 210, nghĩa là, các hệ số biến đổi được lượng tử hóa và thông tin thông số liên quan, có thể được nhập vào trong bộ khử lượng tử hóa 220. Bên cạnh đó, thông tin về việc lọc trong số thông tin được giải mã bởi bộ giải mã entropi 210 có thể được cấp đến bộ lọc 240. Trong khi đó, bộ nhận (không

được thể hiện) để nhận tín hiệu được xuất ra từ thiết bị mã hóa hình ảnh có thể còn được tạo cấu hình làm thành phần trong/ngoài của thiết bị giải mã hình ảnh 200, hoặc bộ nhận có thể là thành phần của bộ giải mã entropi 210.

Trong khi đó, thiết bị giải mã hình ảnh theo sáng chế này có thể được gọi là thiết bị giải mã video/hình ảnh/ảnh. Thiết bị giải mã hình ảnh 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 210. Bộ giải mã mẫu có thể gồm ít nhất một thành phần trong số bộ khử lượng tử hóa 220, bộ biến đổi ngược 230, bộ cộng 235, bộ lọc 240, bộ nhớ 250, đơn vị dự đoán liên ảnh 260 hoặc đơn vị dự đoán trong ảnh 265.

Bộ khử lượng tử hóa 220 có thể khử lượng tử hóa các hệ số biến đổi được lượng tử hóa và xuất ra các hệ số biến đổi. Bộ khử lượng tử hóa 220 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 trong thiết bị mã hóa hình ảnh. Bộ khử lượng tử hóa 220 có thể thực hiện việc khử lượng tử hóa trên các hệ số biến đổi được lượng tử hóa bằng cách sử dụng thông số khử lượng tử hóa (ví dụ thông tin kích cỡ bước lượng tử hóa) và thu được các hệ số biến đổi.

Bộ biến đổi ngược 230 có thể biến đổi ngược các hệ số biến đổi để thu tín hiệu dư (khối dư, mảng mẫu dư).

Đơn vị dự đoán có thể thực hiện việc dự đoán trên khối hiện tại và tạo ra khối được dự đoán gồm các mẫu dự đoán cho khối hiện tại. Đơn vị dự đoán có thể xác định xem liệu việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh có được áp dụng cho khối hiện tại dựa trên thông tin trên đầu ra dự đoán từ bộ giải mã entropi 210 và có thể xác định chế độ dự đoán trong ảnh/liên ảnh cụ thể (kỹ thuật dự đoán).

Điều này giống như những điều đã được mô tả trong đơn vị dự đoán của thiết bị mã hóa hình ảnh 100 mà đơn vị dự đoán có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp (các kỹ thuật) dự đoán khác nhau vốn sẽ được mô tả sau.

Đơn vị dự đoán trong ảnh 265 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong ảnh hiện tại. Phần mô tả của đơn vị dự đoán trong ảnh 185 được áp dụng một cách tương tự cho đơn vị dự đoán trong ảnh 265.

Đơn vị dự đoán liên ảnh 260 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bởi vectơ chuyển động trên ảnh tham chiếu. Trong trường hợp này, để làm giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên ảnh, thì thông tin chuyển động có thể được dự đoán trong các đơn vị của các khối, các khối con, hoặc các 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 ảnh (dự đoán L0, dự đoán L1, dự đoán đôi, v.v.). Trong trường hợp dự đoán liên ảnh, 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, đơn vị dự đoán liên ảnh 260 có thể tạo cấu hình danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận và dẫn ra vectơ chuyển động của khối hiện tại và/hoặc chỉ số ảnh tham chiếu dựa trên thông tin chọn ứng viên nhận được. Việc dự đoán liên ảnh 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 dự đoán liên ảnh cho khối hiện tại.

Bộ cộng 235 có thể tạo ra tín hiệu được tái tạo (ảnh được tái tạo, khối được tái tạo, mảng mẫu được tái tạo) bằng cách cộng tín hiệu dự thu được vào tín hiệu dự đoán (khối

được dự đoán, mảng mẫu được dự đoán) được xuất ra từ đơn vị dự đoán (gồm đơn vị dự đoán liên ảnh 260 và/hoặc đơn vị dự đoán trong ảnh 265). Nếu không có phần dư nào cho khối cần được xử lý, như là trong trường hợp của chế độ bỏ qua được áp dụng, thì khối được dự đoán có thể được sử dụng như khối được tái tạo. Phần mô tả của bộ cộng 155 là có thể áp dụng được một cách tương tự cho bộ cộng 235. Bộ cộng 235 có thể được gọi là bộ tái tạo hoặc bộ phân tạo ra khối được tái tạo. Tín hiệu được tái tạo mà được tạo ra có thể được sử dụng cho việc dự đoán trong ảnh của khối tiếp theo cần được xử lý ở trong ảnh hiện tại và có thể được sử dụng cho việc dự đoán liên ảnh của ảnh tiếp theo thông qua việc lọc như được mô tả dưới đây.

Bộ lọc 240 có thể cải thiện chất lượng hình ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái tạo. Ví dụ, bộ lọc 240 có thể tạo ra ảnh được tái tạo mà được sửa đổi bằng cách áp dụng các phương pháp lọc khác nhau cho ảnh được tái tạo và lưu trữ ảnh được tái tạo mà được sửa đổi trong bộ nhớ 250, cụ thể là, DPB của bộ nhớ 250. Các phương pháp lọc khác nhau có thể gồm, ví dụ, lọc khử khối, bù thích ứng mẫu, bộ lọc vòng lặp thích ứng, bộ lọc song phương, và tương tự.

Ảnh được tái tạo (được biến đổi) mà được lưu trữ trong DPB của bộ nhớ 250 có thể được sử dụng làm ảnh tham chiếu trong đơn vị dự đoán liên ảnh 260. Bộ nhớ 250 có thể lưu trữ thông tin chuyển động của khối mà thông tin chuyển động trong hình ảnh hiện tại được dẫn xuất (hoặc được giải mã) từ đó và/hoặc thông tin chuyển động của các khối trong hình ảnh mà đã được tái tạo. Thông tin chuyển động được lưu trữ có thể được truyền đến đơn vị dự đoán liên ảnh 260 để được tận dụng làm 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 250 có thể lưu trữ các mẫu được tái tạo của các khối được tái tạo trong ảnh hiện tại và truyền các mẫu được tái tạo đến đơn vị dự đoán trong ảnh 265.

Trong sáng chế, các phương án được mô tả trong bộ lọc 160, đơn vị dự đoán liên ảnh 180, và đơn vị dự đoán trong ảnh 185 của thiết bị mã hóa hình ảnh 100 có thể được áp dụng một cách tương tự hoặc tương ứng cho bộ lọc 240, đơn vị dự đoán liên ảnh 260, và đơn vị dự đoán trong ảnh 265 của thiết bị giải mã hình ảnh 200.

Tổng quan của việc phân vùng hình ảnh

Phương pháp tạo mã video/hình ảnh theo sáng chế có thể được thực hiện dựa trên cấu trúc phân vùng hình ảnh như sau. Một cách cụ thể, các thủ tục dự đoán, xử lý phần dư (biến đổi (ngược), (khử) lượng tử hóa, v.v.), tạo mã phân tử cú pháp, và lọc, vốn sẽ được mô tả sau, có thể được thực hiện dựa trên CTU, CU (và/hoặc TU, PU) được dẫn xuất dựa trên cấu trúc phân vùng hình ảnh. Hình ảnh có thể được phân vùng trong các đơn vị khối và thủ tục phân vùng khối có thể được thực hiện trong bộ phân vùng hình ảnh 110 của thiết bị mã hóa. Thông tin liên quan đến việc phân vùng có thể được mã hóa bởi bộ mã hóa entropi 190 và được truyền đến thiết bị giải mã hình ảnh trong dạng luồng bit. Bộ giải mã entropi 210 của thiết bị giải mã hình ảnh có thể dẫn xuất khối cấu trúc phân vùng của ảnh hiện tại dựa trên thông tin liên quan đến việc phân vùng được thu nhận từ luồng bit, và dựa trên điều này, có thể thực hiện một loạt các thủ tục (ví dụ, việc dự đoán, xử lý phần dư, việc tái tạo khối/ảnh, việc lọc trong vòng, v.v.) cho việc giải mã hình ảnh.

Các ảnh có thể được phân vùng thành chuỗi các đơn vị cây tạo mã (các CTU). Fig.4 thể hiện ví dụ mà trong đó ảnh được phân vùng thành các CTU. CTU có thể tương ứng với khối cây tạo mã (CTB). Theo cách thay thế, CTU có thể gồm khối cây tạo mã của các mẫu độ chói và hai khối cây tạo mã của các mẫu sắc độ tương ứng. Ví dụ, đối với ảnh mà chứa ba mảng mẫu, thì CTU có thể gồm khối $N \times N$ của các mẫu độ chói và hai khối tương ứng của các mẫu sắc độ.

Tổng quan của việc phân vùng CTU

Như được mô tả ở trên, đơn vị tạo mã có thể giành được bằng cách phân vùng theo kiểu đệ quy đơn vị cây tạo mã (Coding Tree Unit, CTU) hoặc đơn vị tạo mã lớn nhất (Largest Coding Unit, LCU) theo cấu trúc cây tứ phân cây nhị phân cây tam phân (Quad-Tree Binary-Tree Ternary-Tree, QT/BT/TT). Ví dụ, CTU có thể được phân vùng trước tiên bởi các cấu trúc cây tứ phân. Sau đây, các nút lá của cấu trúc cây tứ phân có thể được phân vùng thêm nữa bởi cấu trúc cây nhiều loại.

Việc phân vùng theo cây tứ phân có nghĩa là CU hiện tại (hoặc CTU) được phân vùng thành bốn phần bằng nhau. Bằng cách phân vùng theo cây tứ phân, thì CU hiện tại có thể được phân vùng thành bốn CU có cùng chiều rộng và cùng chiều cao. Khi CU hiện tại không còn được phân vùng nữa thành cấu trúc cây tứ phân, thì CU hiện tại tương ứng với nút lá của cấu trúc cây tứ phân. CU tương ứng với nút lá của cấu trúc cây tứ phân có thể không còn được phân vùng nữa và có thể được sử dụng làm đơn vị tạo mã cuối cùng được mô tả ở trên. Theo cách thay thế, CU tương ứng với nút lá của cấu trúc cây tứ phân có thể được phân vùng thêm nữa bởi cấu trúc cây nhiều loại.

Fig.5 là hình thể hiện loại phân vùng của khối theo cấu trúc cây nhiều loại. Việc chia tách theo cấu trúc cây nhiều loại có thể gồm hai loại chia tách theo cấu trúc cây nhị phân và hai loại chia tách theo cấu trúc cây tam phân.

Hai loại chia tách theo cấu trúc cây nhị phân có thể gồm việc chia tách nhị phân dọc (SPLIT_BT_VER) và việc chia tách nhị phân ngang (SPLIT_BT_HOR). Việc chia tách nhị phân dọc (SPLIT_BT_VER) có nghĩa là CU hiện tại được chia tách thành hai phần bằng nhau theo hướng dọc. Như được thể hiện trên Fig.4, bằng cách chia tách nhị phân dọc, thì hai CU có cùng chiều cao như CU hiện tại và có chiều rộng bằng một nửa chiều rộng của CU hiện tại có thể được tạo ra. Việc chia tách nhị phân ngang

(SPLIT_BT_HOR) có nghĩa là CU hiện tại được chia tách thành hai phần bằng nhau theo hướng ngang. Như được thể hiện trên Fig.5, bằng cách chia tách nhị phân ngang, thì hai CU có chiều cao bằng một nửa chiều cao của CU hiện tại và có cùng chiều rộng như CU hiện tại có thể được tạo ra.

Hai loại chia tách theo cấu trúc cây tam phân có thể gồm việc chia tách tam phân dọc (SPLIT_TT_VER) và việc chia tách tam phân ngang (SPLIT_TT_HOR). Trong việc chia tách tam phân dọc (SPLIT_TT_VER), thì CU hiện tại được chia tách theo hướng dọc tại tỉ lệ 1:2:1. Như được thể hiện trên Fig.5, bằng cách chia tách tam phân dọc, thì hai CU có cùng chiều cao như CU hiện tại và có chiều rộng bằng 1/4 chiều rộng của CU hiện tại và CU có cùng chiều cao như CU hiện tại và có chiều rộng bằng một nửa chiều rộng của CU hiện tại có thể được tạo ra. Trong việc chia tách tam phân ngang (SPLIT_TT_HOR), thì CU hiện tại được chia tách theo hướng ngang tại tỉ lệ 1:2:1. Như được thể hiện trên Fig.4, bằng cách chia tách tam phân ngang, thì hai CU có chiều cao bằng 1/4 chiều cao của CU hiện tại và có cùng chiều rộng như CU hiện tại và CU có chiều cao bằng một nửa chiều cao của CU hiện tại và có cùng chiều rộng như CU hiện tại có thể được tạo ra.

Fig.6 là hình thể hiện cơ chế phát tín hiệu của thông tin chia tách khối trong cấu trúc của cây tứ phân với cây nhiều loại được lồng vào nhau theo sáng chế.

Ở đây, CTU được xử lý như là nút rễ của cây tứ phân, và được phân vùng lần đầu tiên bởi cấu trúc cây tứ phân. Thông tin (ví dụ, qt_split_flag) chỉ ra xem liệu việc chia tách cây tứ phân được thực hiện đối với CU hiện tại (CTU hoặc nút (QT_node) của cây tứ phân) có được phát tín hiệu hay không. Ví dụ, khi qt_split_flag có giá trị thứ nhất (ví dụ, “1”), thì CU hiện tại có thể được phân vùng theo cây tứ phân. Bên cạnh đó, khi qt_split_flag có giá trị thứ hai (ví dụ, “0”), thì CU hiện tại không được phân vùng theo

cây tứ phân, mà trở thành nút lá (QT_leaf_node) của cây tứ phân. Sau đó, mỗi nút lá cây tứ phân có thể được phân vùng thêm nữa thành các cấu trúc cây nhiều loại. Nghĩa là, nút lá của cây tứ phân có thể trở thành nút (MTT_node) của cây nhiều loại. Trong cấu trúc cây nhiều loại, cờ thứ nhất (ví dụ, Mtt_split_cu_flag) được phát tín hiệu để chỉ ra xem liệu nút hiện tại có được phân vùng theo cách bổ sung hay không. Nếu nút tương ứng được phân vùng theo cách bổ sung (ví dụ, nếu cờ thứ nhất là 1), thì cờ thứ hai (ví dụ, Mtt_split_cu_vertical_flag) có thể được phát tín hiệu để chỉ ra hướng chia tách. Ví dụ, hướng chia tách có thể là hướng dọc nếu cờ thứ hai là 1 và có thể là hướng ngang nếu cờ thứ hai là 0. Sau đó, cờ thứ ba (ví dụ, Mtt_split_cu_binary_flag) có thể được phát tín hiệu để chỉ ra xem liệu loại chia tách là loại chia tách nhị phân hoặc loại chia tách tam phân. Ví dụ, loại chia tách có thể là loại chia tách nhị phân khi cờ thứ ba là 1 và có thể là loại chia tách tam phân khi cờ thứ ba là 0. Nút của cây nhiều loại giành được bởi việc chia tách nhị phân hoặc việc chia tách tam phân có thể được phân vùng thêm nữa thành các cấu trúc cây nhiều loại. Tuy nhiên, nút của cây nhiều loại có thể không được phân vùng thành các cấu trúc cây tứ phân. Nếu cờ thứ nhất là 0, thì nút tương ứng của cây nhiều loại không còn được chia tách nữa mà trở thành nút lá (MTT_leaf_node) của cây nhiều loại. CU tương ứng với nút lá của cây nhiều loại có thể được sử dụng làm đơn vị tạo mã cuối cùng được mô tả ở trên.

Dựa trên mtt_split_cu_vertical_flag và mtt_split_cu_binary_flag, thì chế độ chia tách cây nhiều loại (MttSplitMode) của CU có thể được dẫn xuất như được thể hiện trên bảng 1 dưới đây. Trong phần mô tả sau đây, chế độ chia tách cây nhiều loại có thể được gọi là loại chia tách nhiều cây hoặc ngắn gọn là loại chia tách.

[Bảng 1]

MttSplitMode	mtt_split_cu_vertical_flag	mtt_split_cu_binary_flag
--------------	----------------------------	--------------------------

SPLIT_TT_HOR	0	0
SPLIT_BT_HOR	0	1
SPLIT_TT_VER	1	0
SPLIT_BT_VER	1	1

Fig.7 là hình thể hiện ví dụ mà trong đó CTU được phân vùng thành nhiều CU bằng cách áp dụng cây nhiều loại sau khi áp dụng cây tứ phân. Trên Fig.7, các cạnh khối đậm 710 biểu diễn việc phân vùng cây tứ phân và các cạnh còn lại 720 biểu diễn việc phân vùng cây nhiều loại. CU có thể tương ứng với khối tạo mã (CB). Trong phương án, CU có thể gồm khối tạo mã của các mẫu độ chói và hai khối tạo mã của các mẫu sắc độ tương ứng với các mẫu độ chói. Kích cỡ TB hoặc CB (mẫu) thành phần sắc độ có thể được dẫn xuất dựa trên kích cỡ TB hoặc CB (mẫu) thành phần độ chói theo tỷ lệ thành phần theo định dạng màu sắc (định dạng sắc độ, ví dụ, 4:4:4, 4:2:2, 4:2:0 hoặc tương tự) của ảnh/hình ảnh. Trong trường hợp của định dạng màu sắc 4:4:4, thì kích cỡ CB/TB thành phần sắc độ có thể được thiết lập là bằng kích cỡ CB/TB thành phần độ chói. Trong trường hợp của định dạng màu sắc 4:2:2, thì chiều rộng của CB/TB thành phần sắc độ có thể được thiết lập là bằng một nửa chiều rộng của CB/TB thành phần độ chói và chiều cao của CB/TB thành phần sắc độ có thể được thiết lập thành chiều cao của CB/TB thành phần độ chói. Trong trường hợp của định dạng màu sắc 4:2:0, thì chiều rộng của CB/TB thành phần sắc độ có thể được thiết lập là bằng một nửa chiều rộng của CB/TB thành phần độ chói và chiều cao của CB/TB thành phần sắc độ có thể được thiết lập là bằng một nửa chiều cao của CB/TB thành phần độ chói.

Trong phương án, khi kích cỡ của CTU là 128 dựa trên đơn vị mẫu độ chói, thì kích cỡ của CU có thể có kích cỡ từ 128x128 đến 4x4 vốn là cùng kích cỡ như CTU.

Trong một phương án, trong trường hợp của định dạng màu sắc 4:2:0 (hoặc định dạng sắc độ), thì kích cỡ CB sắc độ có thể có kích cỡ từ 64x64 đến 2x2.

Trong khi đó, trong phương án, kích cỡ CU và kích cỡ TU có thể giống nhau. Theo cách thay thế, có thể có nhiều TU trong vùng CU. Kích cỡ TU biểu diễn chung cho kích cỡ khối biến đổi (TB) (mẫu) thành phần độ chói.

Kích cỡ TU có thể được dẫn xuất dựa trên kích cỡ TB có thể chấp thuận được lớn nhất maxTbSize vốn là giá trị được xác định trước. Ví dụ, khi kích cỡ CU lớn hơn maxTbSize , thì nhiều TU (TB) có maxTbSize có thể được dẫn xuất từ CU và việc biến đổi/biến đổi ngược có thể được thực hiện trong các đơn vị của TU (TB). Ví dụ, kích cỡ TB độ chói có thể chấp thuận được lớn nhất có thể là 64x64 và kích cỡ TB sắc độ có thể chấp thuận được lớn nhất có thể là 32x32. Nếu chiều rộng hoặc chiều cao của CB được phân vùng theo cấu trúc cây lớn hơn chiều rộng hoặc chiều cao biến đổi lớn nhất, thì CB có thể được phân vùng một cách tự động (hoặc một cách rõ ràng) cho đến khi giới hạn kích cỡ TB theo các hướng ngang và dọc được thỏa mãn.

Bên cạnh đó, ví dụ, khi việc dự đoán trong ảnh được áp dụng, thì chế độ/loại dự đoán trong ảnh có thể được dẫn xuất trong các đơn vị của CU (hoặc CB) và thủ tục dẫn xuất mẫu tham chiếu lân cận và tạo ra mẫu dự đoán có thể được thực hiện trong các đơn vị của TU (hoặc TB). Trong trường hợp này, có thể có một hoặc nhiều TU (hoặc TB) trong một vùng CU (hoặc CB) và, trong trường hợp này, nhiều Tus hoặc (TBs) có thể chia sẻ cùng chế độ/loại dự đoán trong ảnh.

Trong khi đó, đối với sơ đồ cây tạo mã tứ phân với cây nhiều loại được lồng vào nhau, thì các thông số sau đây có thể được phát tín hiệu dưới dạng các phần tử cú pháp SPS từ thiết bị mã hóa hình ảnh đến thiết bị giải mã. Ví dụ, ít nhất một kích cỡ trong số kích cỡ CTU vốn là thông số biểu diễn kích cỡ nút rễ của cây tứ phân, MinQTSIZE vốn

là thông số biểu diễn kích cỡ nút lá cây tứ phân được chấp thuận tối thiểu, $MaxBtSize$ vốn là thông số biểu diễn kích cỡ nút rễ cây nhị phân được chấp thuận tối đa, $MaxTtSize$ vốn là thông số biểu diễn kích cỡ nút rễ cây tam phân được chấp thuận tối đa, $MaxMttDepth$ vốn là thông số biểu diễn độ sâu phân cấp được chấp thuận tối đa của việc chia tách cây nhiều loại từ nút lá cây tứ phân, $MinBtSize$ vốn là thông số biểu diễn kích cỡ nút lá cây nhị phân được chấp thuận tối thiểu, hoặc $MinTtSize$ vốn là thông số biểu diễn kích cỡ nút lá cây tam phân được chấp thuận tối thiểu được phát tín hiệu.

Theo phương án sử dụng định dạng sắc độ 4:2:0, thì kích cỡ CTU có thể được thiết lập thành các khối độ chói 128×128 và hai khối sắc độ 64×64 tương ứng với các khối độ chói. Trong trường hợp này, $MinOTSize$ có thể được thiết lập thành 16×16 , $MaxBtSize$ có thể được thiết lập thành 128×128 , $MaxTtSize$ có thể được thiết lập thành 64×64 , $MinBtSize$ và $MinTtSize$ có thể được thiết lập thành 4×4 , và $MaxMttDepth$ có thể được thiết lập thành 4. Việc phân vùng cây tứ phân có thể được áp dụng cho CTU để tạo ra các nút lá cây tứ phân. Nút lá cây tứ phân có thể được gọi là nút QT lá. Các nút lá cây tứ phân có thể có kích cỡ từ kích cỡ 16×16 (ví dụ, $MinOTSize$) đến kích cỡ 128×128 (ví dụ, kích cỡ CTU). Nếu nút QT lá là 128×128 , thì nó có thể không được phân vùng theo cách bổ sung thành cây nhị phân/cây tam phân. Việc này là bởi vì, trong trường hợp này, ngay cả nếu được phân vùng, thì nó vẫn vượt quá $MaxBtSize$ và $MaxTtSize$ (ví dụ, 64×64). Trong các trường hợp khác, các nút QT lá có thể được phân vùng thêm nữa thành cây nhiều loại. Do đó, nút QT lá là nút rễ cho cây nhiều loại, và nút QT lá có thể có giá trị độ sâu cây nhiều loại ($mttDepth$) 0. Nếu độ sâu cây nhiều loại đạt đến $MaxMttdepth$ (ví dụ, 4), thì việc phân vùng thêm nữa có thể không được xem xét thêm nữa. Nếu chiều rộng của nút cây nhiều loại bằng $MinBtSize$ và nhỏ hơn hoặc bằng $2 \times MinTtSize$, thì sau đó không có việc phân vùng ngang thêm nữa nào có thể được xem xét. Nếu chiều cao của nút cây nhiều loại bằng $MinBtSize$ và nhỏ hơn hoặc bằng

2xMinTtSize, thì không có việc phân vùng dọc thêm nữa nào có thể được xem xét. Khi việc phân vùng không được xem xét, thì thiết bị mã hóa hình ảnh có thể bỏ qua việc phát tín hiệu của thông tin phân vùng. Trong trường hợp này, thiết bị giải mã hình ảnh có thể dẫn xuất thông tin phân vùng với giá trị được xác định trước.

Trong khi đó, một CTU có thể gồm khối tạo mã cho các mẫu độ chói (ở dưới đây được gọi là “khối độ chói”) và hai khối tạo mã của các mẫu sắc độ tương ứng với nó (ở dưới đây được gọi là “các khối sắc độ”). Mô hình cây tạo mã được mô tả ở trên có thể được áp dụng một cách tương tự hoặc tách biệt cho khối độ chói và khối sắc độ của CU hiện tại. Một cách cụ thể, các khối độ chói và khối sắc độ trong một CTU có thể được phân vùng thành cùng một cấu trúc cây khối và, trong trường hợp này, cấu trúc cây có thể được biểu diễn như là SINGLE_TREE. Theo cách thay thế, các khối độ chói và khối sắc độ trong một CTU có thể được phân vùng thành các cấu trúc cây khối tách biệt, và, trong trường hợp này, cấu trúc cây có thể được biểu diễn như là DUAL_TREE. Nghĩa là, khi CTU được phân vùng thành các cây kép, thì cấu trúc cây khối cho khối độ chói và cấu trúc cây khối cho khối sắc độ có thể được trình bày theo kiểu tách biệt. Trong trường hợp này, cấu trúc cây khối cho khối độ chói có thể được gọi là DUAL_TREE_LUMA, và cấu trúc cây khối cho thành phần sắc độ có thể được gọi là DUAL_TREE_CHROMA. Đối với các nhóm lát/phiến P và B, thì các khối độ chói và khối sắc độ trong một CTU có thể bị giới hạn để có cùng cấu trúc cây tạo mã. Tuy nhiên, đối với các nhóm lát/phiến I, thì các khối độ chói và sắc độ có thể có cấu trúc cây khối tách biệt với nhau. Nếu cấu trúc cây khối tách biệt được áp dụng, thì CTB độ chói có thể được phân vùng thành các CU dựa trên cấu trúc cây tạo mã cụ thể, và CTB sắc độ có thể được phân vùng thành các CU sắc độ dựa trên một cấu trúc cây tạo mã khác. Nghĩa là, điều này có nghĩa là CU trong nhóm lát/phiến I, mà cấu trúc cây khối tách biệt được áp dụng vào đó, có thể gồm khối tạo mã của các thành phần độ chói hoặc các khối

tạo mã của hai thành phần sắc độ và CU của nhóm lát/phiến P hoặc B có thể gồm các khối của ba thành phần màu sắc (thành phần độ chói và hai thành phần sắc độ).

Mặc dù cấu trúc cây tạo mã tứ phân với cây nhiều loại được lồng vào nhau đã được mô tả, nhưng cấu trúc mà trong đó CU được phân vùng không bị hạn chế ở đó. Ví dụ, cấu trúc BT và cấu trúc TT có thể được diễn giải như là khái niệm được gồm trong cấu trúc nhiều cây phân vùng (Multiple Partitioning Tree, MPT), và CU có thể được diễn giải như được phân vùng thông qua cấu trúc QT và cấu trúc MPT. Trong ví dụ mà tại đó CU được phân vùng thông qua cấu trúc QT và cấu trúc MPT, thì phần tử cú pháp (ví dụ, `MPT_split_type`) gồm thông tin về việc nút lá của cấu trúc QT được phân vùng thành bao nhiêu khối và phần tử cú pháp (ví dụ, `MPT_split_mode`) gồm thông tin về hướng nào trong số các hướng dọc và ngang mà nút lá của cấu trúc QT được phân vùng thành sẽ có thể được phát tín hiệu để xác định cấu trúc phân vùng.

Trong một ví dụ khác, CU có thể được phân vùng theo cách khác với cấu trúc QT, cấu trúc BT hoặc cấu trúc TT. Nghĩa là, không giống như việc CU của độ sâu thấp hơn được phân vùng thành $1/4$ CU của độ sâu cao hơn theo cấu trúc QT, thì CU của độ sâu thấp hơn được phân vùng thành $1/2$ CU của độ sâu cao hơn theo cấu trúc BT, hoặc CU của độ sâu thấp hơn được phân vùng thành $1/4$ hoặc $1/2$ CU của độ sâu cao hơn theo cấu trúc TT, thì CU của độ sâu thấp hơn có thể được phân vùng thành $1/5$, $1/3$, $3/8$, $3/5$, $2/3$, hoặc $5/8$ CU của độ sâu cao hơn trong một số trường hợp, và phương pháp phân vùng CU không bị giới hạn thereto.

Cấu trúc khối tạo mã cây tứ phân với cây nhiều loại có thể cung cấp cấu trúc phân vùng khối rất linh hoạt. Bởi vì các loại phân vùng được hỗ trợ trong cây nhiều loại, nên các mẫu hình phân vùng khác nhau có thể có tiềm năng dẫn đến cùng cấu trúc khối tạo mã trong một số trường hợp. Trong thiết bị mã hóa hình ảnh và thiết bị giải mã, bằng

cách giới hạn việc xảy ra của các mẫu hình phân vùng như vậy, nên lượng dữ liệu của thông tin phân vùng có thể được giảm.

Bên cạnh đó, trong việc mã hóa và giải mã của video/hình ảnh theo sáng chế này, thì cơ sở xử lý hình ảnh có thể có cấu trúc phân cấp. Một ảnh có thể được phân chia thành một hoặc nhiều phiến, viên gạch, lát, và/hoặc nhóm phiến. Một lát có thể gồm một hoặc nhiều viên gạch. Một viên gạch có thể gồm một hoặc nhiều hàng CTU trong phiến. Lát có thể gồm các viên gạch của ảnh, trong đó số lượng các viên gạch là số nguyên. Một nhóm phiến có thể gồm một hoặc nhiều phiến. Một nhóm phiến có thể gồm một hoặc nhiều phiến. CTU có thể được phân vùng thành một hoặc nhiều CU. Phiến có thể là khu vực hình tứ giác gồm có các hàng phiến cụ thể và các cột phiến cụ thể gồm có nhiều CTU ở trong ảnh. Nhóm phiến có thể gồm các phiến theo việc quét mảnh phiến ở trong ảnh, trong đó số lượng các phiến là số nguyên. Phần đầu lát có thể mang thông tin/thông số có thể áp dụng được cho lát tương ứng (các khối trong lát). Khi thiết bị mã hóa hoặc thiết bị giải mã có bộ xử lý nhiều lõi, thì các thủ tục mã hóa/giải mã cho phiến, lát, viên gạch, và/hoặc nhóm phiến có thể được thực hiện song song.

Trong sáng chế, các tên hoặc các khái niệm của lát hoặc nhóm phiến có thể được sử dụng theo kiểu thay thế cho nhau. Nghĩa là, nhóm phiến phần đầu có thể được gọi là phần đầu lát. Ở đây, lát có thể có một loại trong số các loại lát gồm lát trong ảnh (I), lát dự đoán (P), và lát dự đoán đôi (B). Đối với các khối trong lát I, thì việc dự đoán liên ảnh có thể không được sử dụng cho việc dự đoán và chỉ có việc dự đoán trong ảnh có thể được sử dụng. Ngay cả trong trường hợp này, thì giá trị mẫu gốc có thể được tạo mã và được phát tín hiệu mà không cần việc dự đoán. Đối với các khối trong lát P, thì việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh có thể được sử dụng. Khi việc dự đoán liên ảnh được sử dụng, thì chỉ có việc dự đoán đơn có thể được sử dụng. Trong lúc đó,

đối với các khối trong lát B, thì việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh có thể được sử dụng. Khi việc dự đoán liên ảnh được sử dụng, thì việc dự đoán đôi, vốn là mức tối đa, có thể được sử dụng.

Theo đặc điểm (ví dụ, độ phân giải) của hình ảnh video hoặc xem xét đến hiệu quả tạo mã hoặc xử lý song song, thì thiết bị mã hóa có thể xác định phiên/nhóm phiên, viên gạch, lát, và các kích cỡ đơn vị tạo mã lớn nhất và nhỏ nhất. Bên cạnh đó, thông tin về việc này hoặc thông tin để dẫn xuất việc này có thể được gồm trong luồng bit.

Thiết bị giải mã có thể thu nhận thông tin chỉ ra xem liệu CTU trong phiên/nhóm phiên, viên gạch, lát, hoặc phiên của ảnh hiện tại có được phân vùng thành nhiều đơn vị tạo mã hay không. Thiết bị mã hóa và thiết bị giải mã chỉ phát tín hiệu thông tin như vậy theo điều kiện cụ thể, nhờ đó tăng hiệu quả mã hóa.

Phần đầu lát (cú pháp phần đầu lát) có thể gồm thông tin/thông số có thể áp dụng được chung cho lát. APS (cú pháp APS) hoặc PPS (cú pháp PPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho một hoặc nhiều ảnh. SPS (cú pháp SPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho một hoặc nhiều chuỗi. VPS (cú pháp VPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho nhiều lớp. DPS (cú pháp DPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho toàn bộ video. DPS có thể gồm thông tin/thông số liên quan đến tổ hợp của các chuỗi video được tạo mã (Coded Video Sequence, CVS).

Bên cạnh đó, ví dụ, thông tin về việc phân vùng và cấu hình của phiên/nhóm phiên/viên gạch/lát có thể được tạo dựng tại giai đoạn mã hóa thông qua cú pháp mức cao, và được truyền trong dạng luồng bit đến thiết bị giải mã.

Tổng quan của việc dự đoán trong ảnh

Ở dưới đây, việc dự đoán trong ảnh được thực hiện bởi thiết bị mã hóa và thiết bị giải mã được mô tả ở trên sẽ được mô tả chi tiết Việc dự đoán trong ảnh có thể đề cập đến việc dự đoán mà tạo ra các mẫu dự đoán cho khối hiện tại trên cơ sở của các mẫu tham chiếu trong ảnh (ở dưới đây, là ảnh hiện tại) mà khối hiện tại thuộc về.

Việc này sẽ được mô tả với sự tham khảo đến Fig.8. Khi việc dự đoán trong ảnh được áp dụng cho khối hiện tại, thì các mẫu tham chiếu lân cận cần để được sử dụng cho việc dự đoán trong ảnh của khối hiện tại có thể được dẫn xuất. Các mẫu tham chiếu lân cận của khối hiện tại có thể gồm: tổng cộng $2 \times nH$ mẫu gồm các mẫu 811 liên kề với ranh giới bên trái của khối hiện tại có kích cỡ là $nW \times nH$ và các mẫu 812 lân cận với phần dưới cùng bên trái; tổng cộng là $2 \times nH$ mẫu gồm các mẫu 821 liên kề với ranh giới trên cùng của khối hiện tại và các mẫu 822 lân cận với phần trên cùng bên phải; và một mẫu 831 lân cận với phần trên cùng bên trái của khối hiện tại. Theo cách thay thế, các mẫu tham chiếu lân cận của khối hiện tại có thể gồm các cột của các mẫu lân cận trên cùng và các hàng của các mẫu lân cận bên trái.

Bên cạnh đó, các mẫu tham chiếu lân cận của khối hiện tại có thể gồm tổng cộng là nH mẫu liên kề với ranh giới bên phải của khối hiện tại có kích cỡ $nW \times nH$, tổng cộng là nW mẫu liên kề với ranh giới dưới cùng của khối hiện tại và mẫu lân cận với phần dưới cùng bên phải của khối hiện tại.

Tuy nhiên, một số mẫu trong số các mẫu tham chiếu lân cận của khối hiện tại chưa được giải mã hoặc có thể không khả dụng. Trong trường hợp này, thiết bị giải mã có thể tạo dựng các mẫu tham chiếu lân cận cần được sử dụng cho việc dự đoán, bằng cách thay thế các mẫu không khả dụng bằng các mẫu khả dụng. Theo cách thay thế, các mẫu

tham chiếu lân cận cần được sử dụng cho việc dự đoán có thể được tạo dựng thông qua việc nội suy của các mẫu khả dụng.

Khi các mẫu tham chiếu lân cận được dẫn xuất, thì (i) mẫu dự đoán có thể được dẫn xuất trên cơ sở của trung bình hoặc việc nội suy của các mẫu tham chiếu lân cận của khối hiện tại, và (ii) mẫu dự đoán có thể được dẫn xuất trên cơ sở của mẫu tham chiếu có mặt trong hướng (dự đoán) cụ thể đối với mẫu dự đoán, trong số các mẫu tham chiếu lân cận của khối hiện tại. Trường hợp (i) có thể được gọi là chế độ không có hướng hoặc chế độ không có góc, và trường hợp (ii) có thể được gọi là chế độ có hướng hoặc chế độ có góc. Bên cạnh đó, mẫu dự đoán cũng có thể được tạo ra thông qua phép nội suy với mẫu lân cận thứ hai và mẫu lân cận thứ nhất mà được đặt theo hướng ngược lại với hướng dự đoán của chế độ dự đoán trong ảnh của khối hiện tại trên cơ sở của mẫu dự đoán của khối hiện tại, trong số các mẫu tham chiếu lân cận. Trường hợp được mô tả ở trên có thể được gọi là việc dự đoán trong ảnh nội suy tuyến tính (Linear Interpolation Intra Prediction, LIP). Bên cạnh đó, các mẫu dự đoán sắc độ cũng có thể được tạo ra trên cơ sở của các mẫu độ chói bằng cách sử dụng mô hình tuyến tính. Trường hợp này có thể được gọi là chế độ LM. Bên cạnh đó, mẫu dự đoán tạm thời của khối hiện tại có thể được dẫn xuất trên cơ sở của các mẫu tham chiếu lân cận được lọc, và mẫu dự đoán của khối hiện tại có thể được dẫn xuất bằng cách lấy tổng có trọng số mẫu dự đoán tạm thời và ít nhất một mẫu tham chiếu được dẫn xuất theo chế độ dự đoán trong ảnh trong số các mẫu tham chiếu lân cận sẵn có, tức là, các mẫu tham chiếu lân cận chưa được lọc. Trường hợp được mô tả ở trên có thể được gọi là việc dự đoán trong ảnh phụ thuộc vào vị trí (Position Dependent Intra Prediction, PDPC). Bên cạnh đó, đường mẫu tham chiếu có độ chính xác dự đoán cao nhất có thể được lựa chọn trong số nhiều đường mẫu tham chiếu của khối hiện tại để dẫn xuất mẫu dự đoán bằng cách sử dụng mẫu tham chiếu được đặt theo hướng dự đoán trong đường tương ứng. Tại thời

điểm này, việc mã hóa dự đoán trong ảnh có thể được thực hiện bằng cách chỉ ra (phát tín hiệu) đường mẫu tham chiếu được sử dụng đến thiết bị giải mã. Trường hợp được mô tả ở trên có thể được gọi là việc dự đoán trong ảnh đường nhiều tham chiếu (Multi-Reference Line, MRL) hoặc việc dự đoán trong ảnh dựa trên MRL. Bên cạnh đó, khối hiện tại có thể được phân chia thành các phân vùng con dọc hoặc ngang để thực hiện việc dự đoán trong ảnh dựa trên cùng chế độ dự đoán trong ảnh, và các mẫu tham chiếu lân cận có thể được dẫn xuất trên cơ sở từng phân vùng con và được sử dụng. Nghĩa là, trong trường hợp này, chế độ dự đoán trong ảnh cho khối hiện tại được áp dụng một cách tương tự cho các phân vùng con, và mẫu tham chiếu lân cận có thể được dẫn xuất trên cơ sở từng phân vùng con và được sử dụng, nhờ đó tăng hiệu suất dự đoán trong ảnh trong một số trường hợp. Phương pháp dự đoán này có thể được gọi là các phân vùng con trong ảnh (Intra Sub-Partition, ISP) hoặc việc dự đoán trong ảnh dựa trên ISP. Các phương pháp dự đoán trong ảnh này có thể được gọi là các loại dự đoán trong ảnh, được phân biệt với các chế độ dự đoán trong ảnh (ví dụ, chế độ DC, chế độ phẳng, và chế độ có hướng). Các loại dự đoán trong ảnh có thể được gọi theo các thuật ngữ khác nhau như là các sơ đồ dự đoán trong ảnh hoặc các chế độ dự đoán trong ảnh bổ sung. Ví dụ, các loại dự đoán trong ảnh (hoặc chế độ dự đoán trong ảnh bổ sung) có thể gồm ít nhất một thành phần được lựa chọn từ nhóm gồm LIP, PDPC, MRL, và ISP mà được mô tả ở trên. Phương pháp dự đoán trong ảnh chung ngoại trừ các loại dự đoán trong ảnh cụ thể, như là LIP, PDPC, MRL, và ISP có thể được gọi là loại dự đoán trong ảnh bình thường. Loại dự đoán trong ảnh bình thường có thể đề cập đến trường hợp mà trong đó các loại dự đoán trong ảnh cụ thể không được áp dụng, và việc dự đoán có thể được thực hiện trên cơ sở của các chế độ dự đoán trong ảnh được mô tả ở trên. Trong lúc đó, khi cần thiết, thì việc lọc sau có thể được thực hiện trên mẫu dự đoán được dẫn xuất.

Một cách cụ thể, thủ tục dự đoán trong ảnh có thể gồm bước xác định chế độ/loại dự đoán trong ảnh, bước dẫn xuất mẫu tham chiếu lân cận, và bước dẫn xuất mẫu dự đoán dựa trên chế độ/loại dự đoán trong ảnh. Bên cạnh đó, khi cần thiết, thì việc lọc sau có thể được thực hiện trên mẫu dự đoán được dẫn xuất.

Trong lúc đó, bên cạnh các loại dự đoán trong ảnh được mô tả ở trên, việc dự đoán trong ảnh có trọng số tuyến tính afin (Affine Linear Weighted Intra Prediction, ALWIP) có thể được sử dụng. ALWIP có thể được gọi là việc dự đoán trong ảnh có trọng số tuyến tính (Linear Weighted Intra Prediction, LWIP) hoặc việc dự đoán trong ảnh ma trận có trọng số hoặc việc dự đoán trong ảnh dựa trên ma trận (Matrix based Intra Prediction, MIP). Khi MIP được áp dụng cho khối hiện tại, thì các mẫu dự đoán cho khối hiện tại có thể được dẫn xuất bằng cách i) sử dụng các mẫu tham chiếu lân cận phải chịu thủ tục tính trung bình, ii) thực hiện thủ tục nhân ma trận-véc tơ, và thực hiện thêm nữa iii) thủ tục nội suy ngang/dọc khi cần thiết. Các chế độ dự đoán trong ảnh được sử dụng cho MIP có thể khác với các chế độ dự đoán trong ảnh được sử dụng trong việc dự đoán trong ảnh LIP, PDPC, MRL, hoặc ISP, hoặc trong việc dự đoán trong ảnh bình thường. Các chế độ dự đoán trong ảnh cho MIP có thể được gọi là các chế độ dự đoán trong ảnh MIP, các chế độ dự đoán MIP, hoặc các chế độ MIP. Ví dụ, các ma trận và các phần bù khác nhau được sử dụng trong phép nhân ma trận-véc tơ có thể được thiết lập theo các chế độ dự đoán trong ảnh cho MIP. Ở đây, ma trận có thể được gọi là ma trận có trọng số (MIP), và phần bù có thể được gọi là véc tơ phần bù (MIP) hoặc véc tơ bias (MIP). Phương pháp MIP chi tiết sẽ được mô tả sau.

Thủ tục tái tạo khối dựa trên việc dự đoán trong ảnh và đơn vị dự đoán trong ảnh trong thiết bị mã hóa có thể gồm một cách sơ lược, ví dụ, phần được mô tả dưới đây. Bước S910 có thể được thực hiện bởi đơn vị dự đoán trong ảnh 185 của thiết bị mã hóa.

Bước S920 có thể được thực hiện bởi bộ xử lý phần dư mà gồm ít nhất một được lựa chọn từ nhóm gồm bộ trừ 115, bộ biến đổi 120, bộ lượng tử hóa 130, bộ khử lượng tử hóa 140, và bộ biến đổi ngược 150 của thiết bị mã hóa. Một cách cụ thể, bước S920 có thể được thực hiện bởi bộ trừ 115 của thiết bị mã hóa. Trong bước S930, thông tin dự đoán có thể được dẫn xuất bởi bộ dự đoán trong ảnh 185 và có thể được mã hóa bởi bộ mã hóa entropi 190. Trong bước S930, thông tin phần dư có thể được dẫn xuất bởi bộ xử lý phần dư 230 và có thể được mã hóa bởi bộ mã hóa entropi 190. Thông tin phần dư là thông tin về các mẫu dư. Thông tin phần dư có thể gồm thông tin về các hệ số biến đổi được lượng tử hóa cho các mẫu dư. Như được mô tả ở trên, các mẫu dư có thể được dẫn xuất bởi các hệ số biến đổi thông qua bộ biến đổi 120 của thiết bị mã hóa, và các hệ số biến đổi có thể được dẫn xuất làm các hệ số biến đổi được lượng tử hóa thông qua bộ lượng tử hóa 130. Thông tin về các hệ số biến đổi được lượng tử hóa có thể được mã hóa bởi bộ mã hóa entropi 190 thông qua thủ tục tạo mã phần dư.

Thiết bị mã hóa có thể thực hiện việc dự đoán trong ảnh trên khối hiện tại trong bước S910. Thiết bị mã hóa dẫn xuất chế độ/loại dự đoán trong ảnh cho khối hiện tại, xuất các mẫu tham chiếu lân cận của khối hiện tại, và tạo ra các mẫu dự đoán trong khối hiện tại trên cơ sở của chế độ/loại dự đoán trong ảnh và các mẫu tham chiếu lân cận. Ở đây, các thủ tục xác định chế độ/loại dự đoán trong ảnh, dẫn xuất các mẫu tham chiếu lân cận, và tạo ra các mẫu dự đoán có thể được thực hiện một cách đồng thời, hoặc một thủ tục bất kỳ có thể được thực hiện trước các thủ tục khác. Ví dụ, mặc dù không được thể hiện, nhưng đơn vị dự đoán trong ảnh 185 của thiết bị mã hóa hình ảnh có thể gồm đơn vị xác định chế độ/loại dự đoán trong ảnh, đơn vị dẫn xuất mẫu tham chiếu và/hoặc đơn vị dẫn xuất mẫu dự đoán. Đơn vị xác định chế độ/loại dự đoán trong ảnh có thể xác định chế độ/loại dự đoán trong ảnh cho khối hiện tại, đơn vị dẫn xuất mẫu tham chiếu có thể dẫn xuất các mẫu tham chiếu lân cận của khối hiện tại, và đơn vị dẫn xuất mẫu

dự đoán có thể dẫn xuất các mẫu dự đoán của khối hiện tại. Trong lúc đó, khi thủ tục lọc mẫu dự đoán, vốn sẽ được mô tả sau, được thực hiện, thì đơn vị dự đoán trong ảnh 185 có thể còn gồm bộ lọc mẫu dự đoán. Thiết bị mã hóa có thể xác định chế độ/loại được áp dụng cho khối hiện tại trong số nhiều chế độ/loại dự đoán trong ảnh. Thiết bị mã hóa có thể so sánh các chi phí RD cho các chế độ/loại dự đoán trong ảnh và có thể xác định chế độ/loại dự đoán trong ảnh tối ưu cho khối hiện tại.

Trong lúc đó, thiết bị mã hóa có thể thực hiện thủ tục lọc mẫu dự đoán. Việc lọc mẫu dự đoán có thể được gọi là việc lọc sau. Nhờ thủ tục lọc mẫu dự đoán, nên một số hoặc tất cả các mẫu dự đoán có thể được lọc. Trong một số trường hợp, thủ tục lọc mẫu dự đoán có thể được lược bỏ.

Thiết bị mã hóa có thể tạo ra các mẫu dự đoán cho khối hiện tại trên cơ sở của các mẫu dự đoán (được lọc) trong bước S920. Thiết bị mã hóa có thể so sánh các mẫu dự đoán trong các mẫu gốc của khối hiện tại trên cơ sở của pha, và có thể dẫn xuất các mẫu dự đoán.

Thiết bị mã hóa có thể mã hóa hình ảnh thông tin gồm thông tin (thông tin dự đoán) về việc dự đoán trong ảnh và thông tin phần dư trên các mẫu dự đoán trong bước S930. Thông tin dự đoán có thể gồm thông tin về chế độ dự đoán trong ảnh và thông tin về loại dự đoán trong ảnh. Thiết bị mã hóa có thể xuất ra thông tin hình ảnh được mã hóa trong dạng luồng bit. Luồng bit đầu ra có thể được truyền đến thiết bị giải mã thông qua phương tiện lưu trữ hoặc mạng.

Thông tin phần dư có thể gồm cú pháp tạo mã phần dư, vốn sẽ được mô tả sau. Thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng cách biến đổi/lượng tử hóa các mẫu dự đoán. Thông tin phần dư có thể gồm thông tin về các hệ số biến đổi được lượng tử hóa.

Trong lúc đó, thiết bị mã hóa có thể tạo ra ảnh được tái tạo (gồm các mẫu được tái tạo và khối được tái tạo). Để đạt được điều này, thì thiết bị mã hóa có thể thực hiện việc khử lượng tử hóa/biến đổi ngược trên các hệ số biến đổi được lượng tử hóa và dẫn xuất các mẫu dư (được sửa đổi). Lý do để thực hiện việc khử lượng tử hóa/việc biến đổi ngược sau việc biến đổi/việc lượng tử hóa của các mẫu dư là để dẫn xuất các mẫu dư that mà giống như các mẫu dư được dẫn xuất bởi thiết bị giải mã như được mô tả ở trên. Thiết bị mã hoá có thể tạo ra khối được tái tạo gồm các mẫu được tái tạo cho khối hiện tại, trên cơ sở của các mẫu dự đoán và các mẫu dư (được sửa đổi). Trên cơ sở của khối được tái tạo, thì ảnh được tái tạo cho ảnh hiện tại có thể được tạo ra. Như đã mô tả trên đây, thủ tục lọc trong vòng lặp có thể còn được áp dụng cho ảnh được tái tạo.

Thủ tục giải mã video/hình ảnh dựa trên việc dự đoán trong ảnh và đơn vị dự đoán trong ảnh trong thiết bị giải mã có thể gồm một cách sơ lược, ví dụ, phần được mô tả dưới đây. Thiết bị giải mã có thể thực hiện hoạt động tương ứng với hoạt động được thực hiện bởi thiết bị mã hóa.

Các bước S1010 đến S1030 có thể được thực hiện bởi đơn vị dự đoán trong ảnh 265 của thiết bị giải mã. Thông tin dự đoán trong bước S1010 và thông tin phần dư trong bước S1040 có thể được thu nhận từ luồng bit bởi bộ giải mã entropi 210 của thiết bị giải mã. Bộ xử lý phần dư gồm bộ khử lượng tử hóa 220 hoặc bộ biến đổi ngược 230 của thiết bị giải mã hoặc cả hai đều có thể dẫn xuất các mẫu dư cho khối hiện tại trên cơ sở của thông tin phần dư. Một cách cụ thể, bộ khử lượng tử hóa 220 của bộ xử lý phần dư có thể thực hiện việc khử lượng tử hóa trên cơ sở của các hệ số biến đổi được lượng tử hóa được dẫn xuất trên cơ sở của thông tin phần dư, và có thể dẫn xuất các hệ số biến đổi. Bộ biến đổi ngược 230 của bộ xử lý phần dư có thể thực hiện việc biến đổi ngược

trên các hệ số biến đổi và có thể dẫn xuất các mẫu dư cho khối hiện tại. Bước S1050 có thể được thực hiện bởi bộ cộng 235 hoặc bộ tái tạo của thiết bị giải mã.

Một cách cụ thể, thiết bị giải mã có thể dẫn xuất chế độ/loại dự đoán trong ảnh cho khối hiện tại trên cơ sở của thông tin dự đoán nhận được (thông tin chế độ/loại dự đoán trong ảnh) trong bước S1010. Thiết bị giải mã có thể dẫn xuất các mẫu tham chiếu lân cận của khối hiện tại trong bước S1020. Thiết bị giải mã có thể tạo ra các mẫu dự đoán trong khối hiện tại trên cơ sở của chế độ/loại dự đoán trong ảnh và các mẫu tham chiếu lân cận trong bước S1030. Trong trường hợp này, thiết bị giải mã có thể thực hiện thủ tục lọc mẫu dự đoán. Việc lọc mẫu dự đoán có thể được gọi là việc lọc sau. Nhờ thủ tục lọc mẫu dự đoán, nên một số hoặc tất cả các mẫu dự đoán có thể được lọc. Trong một số trường hợp, thủ tục lọc mẫu dự đoán có thể được lược bỏ.

Thiết bị giải mã có thể tạo ra các mẫu dư cho khối hiện tại trên cơ sở của thông tin phần dư nhận được. Thiết bị giải mã có thể tạo ra các mẫu được tái tạo cho khối hiện tại trên các cơ sở của các mẫu dự đoán và các mẫu dư, và có thể dẫn xuất khối được tái tạo gồm các mẫu được tái tạo trong bước S1040. Trên cơ sở của khối được tái tạo, thì ảnh được tái tạo cho ảnh hiện tại có thể được tạo ra. Như đã mô tả trên đây, thủ tục lọc trong vòng lặp có thể còn được áp dụng cho ảnh được tái tạo.

Ở đây, mặc dù không được thể hiện, nhưng đơn vị dự đoán trong ảnh 265 của thiết bị giải mã hình ảnh có thể gồm đơn vị xác định chế độ/loại dự đoán trong ảnh, đơn vị dẫn xuất mẫu tham chiếu và đơn vị dẫn xuất mẫu dự đoán. Đơn vị xác định chế độ/loại dự đoán trong ảnh có thể xác định chế độ/loại dự đoán trong ảnh cho khối hiện tại trên cơ sở của chế độ/loại dự đoán trong ảnh thông tin được thu nhận từ bộ giải mã entropi 210. Đơn vị dẫn xuất mẫu tham chiếu có thể dẫn xuất các mẫu tham chiếu lân cận của khối hiện tại. Đơn vị dẫn xuất mẫu dự đoán có thể dẫn xuất các mẫu dự đoán của khối

hiện tại. Trong lúc đó, khi thủ tục lọc mẫu dự đoán được mô tả ở trên được thực hiện, thì đơn vị dự đoán trong ảnh 265 có thể còn gồm bộ lọc mẫu dự đoán.

Thông tin chế độ dự đoán trong ảnh có thể gồm, ví dụ, thông tin cờ (ví dụ, `intra_luma_mpm_flag`) chỉ ra xem liệu chế độ khả thi nhất (Most Probable Mode, MPM) có được áp dụng cho khối hiện tại hay không hoặc chế độ còn lại có được áp dụng hay không. Khi MPM được áp dụng cho khối hiện tại, thì thông tin chế độ dự đoán có thể còn gồm thông chỉ chỉ số (ví dụ, `intra_luma_mpm_idx`) chỉ ra một ứng viên trong số các ứng viên chế độ dự đoán trong ảnh (các ứng viên MPM). Các ứng viên chế độ dự đoán trong ảnh (các ứng viên MPM) có thể được tạo dựng làm danh sách ứng viên MPM hoặc danh sách MPM. Bên cạnh đó, khi MPM không được áp dụng cho khối hiện tại, thì thông tin chế độ dự đoán trong ảnh có thể còn gồm thông tin chế độ còn lại (ví dụ, `intra_luma_mpm_remainder`) chỉ ra một chế độ trong số các chế độ dự đoán trong ảnh còn lại ngoại trừ các ứng viên chế độ dự đoán trong ảnh (các ứng viên MPM). Thiết bị giải mã có thể xác định chế độ dự đoán trong ảnh của khối hiện tại trên cơ sở của thông tin chế độ dự đoán trong ảnh. Danh sách MPM tách biệt có thể được tạo dựng cho MIP được mô tả ở trên.

Bên cạnh đó, thông tin loại dự đoán trong ảnh có thể được hiện thực hóa trong các dạng khác nhau. Ví dụ, thông tin loại dự đoán trong ảnh có thể gồm thông tin chỉ số loại dự đoán trong ảnh chỉ ra một loại trong số các loại dự đoán trong ảnh. Theo một phương án khác, thông tin loại dự đoán trong ảnh có thể gồm ít nhất một thông tin trong số thông tin đường mẫu tham chiếu (ví dụ, `intra_luma_ref_idx`) chỉ ra xem liệu MRL có được áp dụng cho khối hiện tại hay không và đường mẫu tham chiếu nào được sử dụng khi MRL được áp dụng cho khối hiện tại, thông tin cờ ISP (ví dụ, `intra_subpartitions_mode_flag`) để chỉ ra xem liệu ISP có được áp dụng cho khối hiện

tại hay không, thông tin loại ISP (ví dụ, `intra_subpartitions_split_flag`) chỉ ra loại phân vùng của các phân vùng con khi ISP được áp dụng, thông tin còn chỉ ra xem liệu PDCP có được áp dụng hay không, hoặc thông tin còn chỉ ra xem liệu LIP có được áp dụng hay không. Bên cạnh đó, thông tin loại dự đoán trong ảnh có thể gồm còn MIP chỉ ra xem liệu MIP có được áp dụng cho khối hiện tại hay không.

Thông tin chế độ dự đoán trong ảnh và/hoặc thông tin loại dự đoán trong ảnh có thể được mã hoá/được giải mã thông qua phương pháp tạo mã được mô tả trong sáng chế này. Ví dụ, thông tin chế độ dự đoán trong ảnh và/hoặc thông tin loại dự đoán trong ảnh có thể được mã hoá/giải mã thông qua việc tạo mã entropi (ví dụ, CABAC, CAVLC) trên cơ sở của mã nhị phân bị cắt cụt (rice).

Việc lượng tử hóa/khử lượng tử hóa

Như được mô tả ở trên, bộ lượng tử hoá của thiết bị mã hoá có thể dẫn xuất các hệ số biến đổi được lượng tử hoá bằng cách áp dụng việc lượng tử hoá cho các hệ số biến đổi. Bộ khử lượng tử hóa của thiết bị mã hoá hoặc bộ khử lượng tử hóa của thiết bị giải mã có thể dẫn xuất các hệ số biến đổi bằng cách áp dụng việc khử lượng tử hóa cho các hệ số biến đổi được lượng tử hoá.

Trong việc mã hóa và giải mã của video/hình ảnh tĩnh, tỷ lệ lượng tử hóa có thể được thay đổi, và tỷ lệ lượng tử hóa được thay đổi có thể được sử dụng để điều chỉnh tỷ lệ nén. Từ góc độ hiện thực hóa, xem xét đến độ phức tạp, thì thông số lượng tử hóa (QP) có thể được sử dụng thay vì sử dụng tỷ lệ lượng tử hóa một cách trực tiếp. Ví dụ, các thông số lượng tử hóa có các giá trị nguyên từ 0 đến 63 có thể được sử dụng, và mỗi giá trị thông số lượng tử hóa có thể tương ứng với tỷ lệ lượng tử hóa thực tế. Bên cạnh đó, thông số lượng tử hóa QPY cho thành phần độ chói (mẫu độ chói) và thông số lượng tử hóa QPC cho thành phần sắc độ (mẫu sắc độ) có thể được thiết lập để khác nhau.

Trong quy trình lượng tử hóa, hệ số biến đổi C có thể được nhập vào và được chia cho tỷ lệ lượng tử hóa Q_{step} , và trên cơ sở của việc này, thì hệ số biến đổi được lượng tử hóa C' có thể được thu nhận. Trong trường hợp này, xem xét đến độ phức tạp tính toán, thì tỷ lệ lượng tử hóa có thể được nhân với thang định cỡ để ở trong dạng số nguyên, và phép toán dịch chuyển có thể được thực hiện bởi giá trị tương ứng với giá trị định cỡ. Dựa trên phép nhân của tỷ lệ lượng tử hóa và giá trị định cỡ, thì thang định cỡ lượng tử hóa có thể được dẫn xuất. Nghĩa là, thang định cỡ lượng tử hóa có thể được dẫn xuất theo QP. Thang định cỡ lượng tử hóa có thể được áp dụng cho hệ số biến đổi C , và trên cơ sở của việc này, thì hệ số biến đổi được lượng tử hóa C' có thể được dẫn xuất.

Quy trình khử lượng tử hóa là quy trình ngược của quy trình lượng tử hoá. Hệ số biến đổi được lượng tử hóa C' có thể được nhân với tỷ lệ lượng tử hóa Q_{step} , và trên cơ sở của việc này, thì hệ số biến đổi được tái tạo C'' có thể được thu nhận. Trong trường hợp này, thang định cỡ mức có thể được dẫn xuất theo thông số lượng tử hóa. Thang định cỡ mức có thể được áp dụng cho hệ số biến đổi được lượng tử hóa C' , và trên cơ sở của việc này, thì hệ số biến đổi được tái tạo C'' có thể được dẫn xuất. Hệ số biến đổi được tái tạo C'' có thể khác một chút với hệ số biến đổi gốc C bởi vì sự mất mát trong quy trình biến đổi và/hoặc lượng tử hóa. Do đó, thiết bị mã hóa có thể thực hiện quy trình khử lượng tử hóa theo cùng cách thức như thiết bị giải mã.

Trong lúc đó, công nghệ lượng tử hóa có trọng số tần số thích ứng mà trong đó độ mạnh lượng tử hóa được điều chỉnh theo tần số có thể được áp dụng. Công nghệ lượng tử hóa có trọng số tần số thích ứng là phương pháp áp dụng các độ mạnh lượng tử hóa mà biến thiên từ tần số sang tần số. Trong việc lượng tử hóa có trọng số tần số thích ứng, thì các độ mạnh lượng tử hóa biến thiên từ tần số sang tần số có thể được áp dụng bằng cách sử dụng ma trận định cỡ lượng tử hóa được định rõ trước. Nghĩa là, quy trình lượng

tử hóa/khử lượng tử hóa được mô tả ở trên có thể được thực hiện thêm nữa dựa trên cơ sở của ma trận định cỡ lượng tử hóa. Ví dụ, các ma trận định cỡ lượng tử hóa khác nhau có thể được sử dụng theo kích cỡ của khối hiện tại và/hoặc liệu chế độ dự đoán được áp dụng cho khối hiện tại để tạo ra tín hiệu dư của khối hiện tại là việc dự đoán liên ảnh hay việc dự đoán trong ảnh. Ma trận định cỡ lượng tử hóa có thể được gọi là ma trận lượng tử hóa hoặc ma trận định cỡ. Ma trận định cỡ lượng tử hóa có thể được định rõ trước. Bên cạnh đó, đối với việc định cỡ thích ứng tần số, thì thông tin định cỡ lượng tử hóa tần số cụ thể cho ma trận định cỡ lượng tử hóa có thể được tạo dựng/được mã hóa bởi thiết bị mã hóa, và có thể được phát tín hiệu đến thiết bị giải mã. Thông tin định cỡ lượng tử hóa tần số cụ thể có thể được gọi là thông tin định cỡ lượng tử hóa. Thông tin định cỡ lượng tử hóa tần số cụ thể có thể gồm dữ liệu danh sách định cỡ (`scaling_list_data`). Ma trận định cỡ lượng tử hóa (được sửa đổi) có thể được dẫn xuất dựa trên cơ sở của dữ liệu danh sách định cỡ. Bên cạnh đó, thông tin định cỡ lượng tử hóa tần số cụ thể có thể gồm thông tin cờ có mặt chỉ ra xem liệu dữ liệu danh sách định cỡ có mặt hay không. Theo cách thay thế, khi dữ liệu danh sách định cỡ được phát tín hiệu tại mức cao (ví dụ, SPS) còn được gồm là thông tin chỉ ra xem liệu dữ liệu danh sách định cỡ có được sửa đổi trong mức thấp hơn (ví dụ, PPS hoặc phần đầu nhóm phiên, v.v.) hay không.

Việc biến đổi/biến đổi ngược

Như được mô tả ở trên, thiết bị mã hóa có thể dẫn xuất khối dư (các mẫu dư) trên cơ sở của khối (các mẫu dự đoán) được dự đoán thông qua việc dự đoán trong ảnh/liên ảnh/IBC, và có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng cách áp dụng việc biến đổi và việc lượng tử hóa cho các mẫu dư được dẫn xuất. Vì được gồm trong cú pháp tạo mã phần dư, nên thông tin (thông tin phần dư) về các hệ số biến đổi được

lượng tử hóa có thể được mã hóa và được xuất ra trong dạng luồng bit. 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 cách thực hiện việc giải mã. Thiết bị giải mã có thể dẫn xuất các mẫu dư thông qua việc khử lượng tử hóa/việc biến đổi ngược trên cơ sở của các hệ số biến đổi được lượng tử hóa. Như được mô tả ở trên, hoặc việc lượng tử hóa/việc khử lượng tử hóa hoặc việc biến đổi/biến đổi ngược hoặc cả hai đều có thể được lược bỏ. Khi việc biến đổi/biến đổi ngược được lược bỏ, thì hệ số biến đổi có thể được gọi là hệ số hoặc hệ số dư hoặc có thể vẫn được gọi là hệ số biến đổi cho sự đồng nhất về diễn đạt. Việc liệu việc biến đổi/biến đổi ngược có được lược bỏ hay không sẽ có thể được phát tín hiệu trên cơ sở của cờ bỏ qua biến đổi (ví dụ, `transform_skip_flag`). Giá trị thứ nhất (ví dụ, 0) của `transform_skip_flag` có thể chỉ ra rằng việc liệu việc biến đổi có được lược bỏ hay không sẽ được xác định bởi một phần tử cú pháp khác. Giá trị thứ hai (ví dụ, 1) của `transform_skip_flag` có thể chỉ ra việc bỏ qua biến đổi (ví dụ, bỏ qua).

Việc biến đổi/biến đổi ngược có thể được thực hiện trên cơ sở của nhân (các nhân) biến đổi. Ví dụ, sơ đồ lựa chọn nhiều biến đổi (Multiple Transform Selection, MTS) để thực hiện việc biến đổi/biến đổi ngược sẽ có thể được áp dụng. Trong trường hợp này, một số tập hợp nhân biến đổi trong số nhiều tập hợp nhân biến đổi có thể được lựa chọn và được áp dụng cho khôi hiện tại. Nhân biến đổi có thể được gọi bằng các thuật ngữ khác nhau, như là ma trận biến đổi, loại biến đổi, v.v. Ví dụ, tập hợp nhân biến đổi có thể đề cập đến tổ hợp của nhân biến đổi hướng dọc (nhân biến đổi dọc) và nhân biến đổi hướng ngang (nhân biến đổi ngang).

Việc biến đổi/biến đổi ngược có thể được thực hiện trên cơ sở từng CU hoặc TU. Nghĩa là, việc biến đổi/biến đổi ngược có thể được áp dụng cho các mẫu dư trong CU

hoặc các mẫu dư trong TU. Kích cỡ CU và kích cỡ TU có thể giống nhau, hoặc nhiều TU có thể có mặt trong khu vực CU. Trong lúc đó, kích cỡ CU có thể đề cập chung đến kích cỡ CB (mẫu) thành phần độ chói. Kích cỡ TU có thể đề cập chung đến kích cỡ TB (mẫu) thành phần độ chói. Kích cỡ TB hoặc CB (mẫu) thành phần sắc độ có thể được dẫn xuất trên cơ sở của kích cỡ TB hoặc CB (mẫu) thành phần độ chói theo tỷ lệ thành phần theo định dạng màu sắc (định dạng sắc độ, ví dụ, 4:4:4, 4:2:2, 4:2:0, hoặc tương tự). Kích cỡ TU có thể được dẫn xuất trên cơ sở của maxTbSize . Ví dụ, khi kích cỡ CU lớn hơn maxTbSize , thì nhiều TU (TB) có maxTbSize có thể được dẫn xuất từ CU và việc biến đổi/biến đổi ngược có thể được thực hiện trên cơ sở từng TU (TB). maxTbSize có thể được xem xét trong việc xác định liệu các loại dự đoán trong ảnh khác nhau, như là ISP, có được áp dụng hay không. Thông tin về maxTbSize có thể được xác định trước. Theo cách thay thế, thông tin về maxTbSize có thể được tạo ra và được mã hóa bởi thiết bị mã hóa và được phát tín hiệu đến thiết bị mã hóa.

Việc tạo mã entropi

Như đã mô tả ở trên với sự tham khảo đến Fig.2, một số hoặc tất cả thông tin video/hình ảnh có thể được mã hóa entropi bởi bộ mã hóa entropi 190. Một số hoặc tất cả thông tin video/hình ảnh được mô tả với sự tham khảo đến Fig.3 có thể được giải mã entropi bởi bộ giải mã entropi 310. Trong trường hợp này, thông tin video/hình ảnh có thể được mã hóa/được giải mã trên cơ sở từng phần tử cú pháp. Trong sáng chế này, việc mã hóa/giải mã của thông tin có thể gồm việc mã hóa/giải mã bởi phương pháp được mô tả trong đoạn này.

Fig.11 thể hiện sơ đồ khối của CABAC để mã hóa một phần tử cú pháp. Trong quy trình mã hóa của CABAC, trước tiên, khi tín hiệu đầu vào là phần tử cú pháp không phải là giá trị nhị phân, thì tín hiệu đầu vào được biến đổi thành giá trị nhị phân thông qua sự

nhị phân hóa. Khi tín hiệu đầu vào đã là giá trị nhị phân, thì tín hiệu đầu vào sự nhị phân hóa đi vòng qua. Ở đây, mỗi số nhị phân 0 hoặc 1 cấu thành giá trị nhị phân có thể được gọi là ngăn. Ví dụ, trong trường hợp mà trong đó dãy ngăn (dãy ngăn) sau sự nhị phân hóa là 110, thì mỗi thành phần trong số 1, 1, và 0 sẽ được gọi là ngăn. Ngăn (các ngăn) cho một phần tử cú pháp có thể đề cập đến giá trị của phần tử cú pháp.

Các ngăn được nhị phân hóa có thể được nhập vào công cụ tạo mã thông thường hoặc công cụ tạo mã đi ngang qua. Công cụ tạo mã thông thường có thể chỉ định mô hình ngữ cảnh mà áp dụng giá trị xác suất cho ngăn tương ứng, và có thể mã hóa ngăn trên cơ sở của mô hình ngữ cảnh được chỉ định. Sau khi thực hiện việc tạo mã trên mỗi ngăn, thì công cụ tạo mã thông thường có thể cập nhật mô hình xác suất cho ngăn. Các ngăn được tạo mã theo cách này có thể được gọi là các ngăn được tạo mã ngữ cảnh. Công cụ tạo mã đi vòng qua có thể lược bỏ thủ tục ước lượng xác suất của ngăn đầu vào và thủ tục để cập nhật mô hình xác suất được áp dụng cho ngăn sau khi tạo mã. Trong trường hợp của công cụ tạo mã đi vòng qua, ngăn đầu vào được tạo mã bằng cách áp dụng sự phân bố xác suất đồng nhất (ví dụ, 50:50) thay vì việc chỉ định ngữ cảnh, nhờ đó cải thiện tốc độ tạo mã. Các ngăn được tạo mã theo cách này có thể được gọi là các ngăn đi vòng qua. Mô hình ngữ cảnh có thể được chỉ định và được cập nhật cho mỗi ngăn cần được tạo mã ngữ cảnh (được tạo mã thông thường), và có thể được chỉ ra dựa trên cơ sở của `ctxIdx` hoặc `ctxInc`. `ctxIdx` có thể được dẫn xuất dựa trên cơ sở của `ctxInc`. Một cách cụ thể, ví dụ, chỉ số ngữ cảnh (`ctxIdx`) chỉ ra mô hình ngữ cảnh cho mỗi ngăn trong số các ngăn được tạo mã thông thường có thể được dẫn xuất làm tổng của phần gia tăng chỉ số ngữ cảnh (`ctxInc`) và phần bù chỉ số ngữ cảnh (`ctxIdxOffset`). Ở đây, `ctxInc` biến thiên từ ngăn sang ngăn có thể được dẫn xuất. `ctxIdxOffset` có thể được biểu diễn bởi giá trị thấp nhất của `ctxIdx`. Giá trị thấp nhất của `ctxIdx` có thể được gọi là giá trị ban đầu (`initValue`) của `ctxIdx`. `ctxIdxOffset` là giá trị được sử dụng chung cho việc

phân biệt với các mô hình ngữ cảnh cho các phần tử cú pháp khác, và mô hình ngữ cảnh cho một phần tử cú pháp có thể được phân biệt/được dẫn xuất trên cơ sở của ctxinc.

Trong thủ tục mã hóa entropi, thì xác định được việc liệu có thực hiện việc mã hóa thông qua công cụ tạo mã thông thường hoặc thực hiện việc mã hóa thông qua công cụ tạo mã đi vòng qua, và đường dẫn tạo mã có thể được chuyển đổi. Việc giải mã entropi có thể thực hiện theo cùng quy trình như việc mã hóa theo thứ tự đảo ngược.

Việc tạo mã entropi được mô tả ở trên có thể được thực hiện như trên Fig.12 và Fig.13, ví dụ. Tham khảo đến Fig.12 và Fig.13, thiết bị mã hóa (bộ mã hóa entropi) có thể thực hiện thủ tục tạo mã entropi trên 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 dự đoán liên ảnh/trong ảnh, thông tin chế độ dự đoán trong ảnh, và thông tin chế độ dự đoán liên ảnh), thông tin phân dư, thông tin liên quan đến việc lọc trong vòng, hoặc có thể gồm các phần tử cú pháp khác nhau liên quan với chúng. Việc tạo mã entropi có thể được thực hiện trên cơ sở từng phần tử cú pháp. Các bước S1210 đến S1220 trên Fig.12 có thể được thực hiện bởi bộ mã hóa entropi 190 của thiết bị mã hóa trên Fig.2 được mô tả ở trên.

Thiết bị mã hóa có thể thực hiện sự nhị phân hóa trên phần tử cú pháp mục tiêu trong bước S1210. Ở đây, sự nhị phân hóa có thể được dựa trên các phương pháp nhị phân hóa khác nhau, như là quy trình nhị phân hóa rice bị cắt cụt và quy trình nhị phân hóa chiều dài cố định, và phương pháp nhị phân hóa cho phần tử cú pháp mục tiêu có thể được định rõ trước. Thủ tục nhị phân hóa có thể được thực hiện bởi đơn vị nhị phân hóa 191 trong bộ mã hóa entropi 190.

Thiết bị mã hóa có thể thực hiện việc mã hóa entropi trên phần tử cú pháp mục tiêu trong bước S1220. Thiết bị mã hóa có thể thực hiện việc mã hóa dựa trên việc tạo mã

thông thường (dựa trên ngữ cảnh) hoặc dựa trên việc tạo mã đi vòng qua trên dãy ngăn của phần tử cú pháp mục tiêu trên cơ sở của sơ đồ tạo mã entropi như là việc tạo mã số học thích ứng ngữ cảnh (Context-Adaptive Arithmetic Coding, CABAC) hoặc việc tạo mã chiều dài có thể biến đổi được thích ứng ngữ cảnh (Context-Adaptive Variable Length Coding, CAVLC). Đầu ra có thể được gồm trong luồng bit. Thủ tục mã hóa entropi có thể được thực hiện bởi bộ xử lý mã hóa entropi 192 trong bộ mã hóa entropi 190. 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.

Tham khảo đến Fig.14 và Fig.15, 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 dự đoán liên ảnh/trong ảnh, thông tin chế độ dự đoán trong ảnh, và thông tin chế độ dự đoán liên ảnh), thông tin phân dư, thông tin liên quan đến việc lọc trong vòng, hoặc có thể gồm các phần tử cú pháp khác nhau liên quan với chúng. Việc tạo mã entropi có thể được thực hiện trên cơ sở từng phần tử cú pháp. Các bước S1410 đến S1420 có thể được thực hiện bởi bộ giải mã entropi 210 của thiết bị giải mã trên Fig.3 được mô tả ở trên.

Thiết bị giải mã có thể thực hiện sự nhị phân hóa trên phần tử cú pháp mục tiêu trong bước S1410. Ở đây, sự nhị phân hóa có thể được dựa trên các phương pháp nhị phân hóa khác nhau, như là quy trình nhị phân hóa rice bị cắt cụt và quy trình nhị phân hóa chiều 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 dãy ngăn khả dụng (các ứng viên dãy ngăn) cho các giá trị khả dụng của phần tử cú pháp mục tiêu thông qua thủ tục

nhị phân hóa. Thủ tục nhị phân hóa có thể được thực hiện bởi đơn vị nhị phân hóa 211 trong bộ mã hóa entropi 210.

Thiết bị giải mã có thể thực hiện việc giải mã entropi trên phần tử cú pháp mục tiêu trong bước S1420. Trong khi thiết bị giải mã giải mã và phân tích cú pháp một cách tuần tự mỗi ngăn cho phần tử cú pháp mục tiêu từ bit (các bit) đầu vào trong luồng bit, và có thể so sánh dãy ngăn được dẫn xuất với các ngăn khả dụng cho phần tử cú pháp. Khi dãy ngăn được dẫn xuất giống với một dãy ngăn trong số các dãy ngăn khả dụng, thì giá trị tương ứng với dãy ngăn có thể được dẫn xuất làm giá trị của phần tử cú pháp. Nếu không thì bit tiếp theo trong luồng bit được phân tích cú pháp thêm nữa và thủ tục được mô tả ở trên sẽ được thực hiện một lần nữa. Thông qua quy trình này, bit bắt đầu hoặc bit kết thúc cho thông tin cụ thể (phần tử cú pháp cụ thể) trong luồng bit không được sử dụng, nhưng các bit có chiều dài có thể thay đổi được sẽ được sử dụng để phát tín hiệu thông tin. Thông qua việc này thì tương đối ít bit hơn sẽ được chỉ định cho giá trị thấp, nhờ đó tăng hiệu quả tạo mã tổng thể.

Thiết bị giải mã có thể thực hiện việc giải mã dựa trên ngữ cảnh hoặc dựa trên sự đi vòng qua trên mỗi ngăn trong dãy ngăn từ luồng bit trên cơ sở của sơ đồ tạo mã entropi, như là CABAC hoặc CAVLC. Thủ tục giải mã entropi có thể được thực hiện bởi bộ xử lý giải mã entropi 212 trong bộ giải mã entropi 210. Luồng bit có thể gồm các loại thông tin khác nhau cho việc giải mã hình ảnh/video như được mô tả ở trên. 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.

Trong sáng chế này, bảng (bảng cú pháp) gồm các phần tử cú pháp có thể được sử dụng để biểu diễn việc phát tín hiệu của thông tin từ thiết bị mã hóa đến thiết bị giải mã. Thứ tự của các phần tử cú pháp trong bảng gồm các phần tử cú pháp được sử dụng trong

sáng chế này có thể đề cập đến thứ tự phân tích cú pháp của các phần tử cú pháp từ luồng bit. Thiết bị mã hóa có thể tạo dựng và mã hóa bảng cú pháp để các phần tử cú pháp được phân tích cú pháp theo thứ tự phân tích cú pháp bởi thiết bị giải mã. Thiết bị giải mã có thể phân tích cú pháp và giải mã các phần tử cú pháp của bảng cú pháp từ luồng bit theo thứ tự phân tích cú pháp, và vì thế có thể thu nhận các giá trị của các phần tử cú pháp.

Thủ tục tạo mã hình ảnh/video chung

Trong việc tạo mã hình ảnh/video, các ảnh cấu thành hình ảnh/video có thể được mã hóa/được giải mã theo thứ tự giải mã hàng loạt. Thứ tự ảnh tương ứng với thứ tự đầu ra của các ảnh được giải mã có thể được thiết lập khác với thứ tự giải mã, và trên cơ sở của việc này, thì việc dự đoán xuôi cũng như hướng ngược có thể được thực hiện khi việc dự đoán liên ảnh được thực hiện.

Fig.16 thể hiện ví dụ về thủ tục giải mã ảnh sơ lược mà phương án (các phương án) của sáng chế này có thể áp dụng được vào đó. Trên Fig.16, bước S1610 có thể được thực hiện bởi bộ giải mã entropi 210 của thiết bị giải mã được mô tả ở trên với sự tham khảo đến Fig.3. Bước S1620 có thể được thực hiện bởi đơn vị dự đoán gồm đơn vị dự đoán trong ảnh 265 và đơn vị dự đoán liên ảnh 260. Bước S1630 có thể được thực hiện bởi bộ xử lý phần dư gồm bộ khử lượng tử hóa 220 và bộ biến đổi ngược 230. Bước S1640 có thể được thực hiện bởi bộ cộng 235. Bước S1650 có thể được thực hiện bởi bộ lọc 240. Bước S1610 có thể gồm thủ tục giải mã thông tin được mô tả trong sáng chế này. Bước S1620 có thể gồm thủ tục dự đoán liên ảnh/trong ảnh được mô tả trong sáng chế này. Bước S1630 có thể gồm thủ tục xử lý phần dư được mô tả trong sáng chế này. Bước S1640 có thể gồm thủ tục tái tạo khối/ảnh được mô tả trong sáng chế này. Bước S1650 có thể gồm thủ tục lọc trong vòng được mô tả trong sáng chế này.

Tham khảo đến Fig.16, thủ tục giải mã ảnh có thể gồm một cách sơ lược, như được mô tả ở trên với sự tham khảo đến Fig.3, thủ tục giành được thông tin hình ảnh/video từ luồng bit (thông qua việc giải mã) trong bước S1610, thủ tục tái tạo ảnh trong các bước S1620 đến S1640, và thủ tục lọc trong vòng cho ảnh được tái tạo trong bước S1650. Thủ tục tái tạo ảnh có thể được thực hiện trên cơ sở của các mẫu dự đoán và các mẫu dư mà được thu nhận thông qua thủ tục dự đoán liên ảnh/trong ảnh trong bước S1620 và xử lý phần dư trong bước S1630 (khử lượng tử hóa và biến đổi ngược của các hệ số biến đổi được lượng tử hóa) được mô tả trong sáng chế này. Ảnh được tái tạo mà được sửa đổi có thể được tạo ra thông qua thủ tục lọc trong vòng cho ảnh được tái tạo sẽ được tạo ra thông qua thủ tục tái tạo ảnh. Ảnh được tái tạo mà được sửa đổi có thể được xuất ra dưới dạng ảnh được giải mã, và có thể được lưu trữ trong bộ đệm ảnh được giải mã hoặc bộ nhớ 250 của thiết bị giải mã mà được sử dụng sau làm ảnh tham chiếu trong thủ tục dự đoán liên ảnh khi ảnh được giải mã. Trong một số trường hợp, thủ tục lọc trong vòng có thể được lược bỏ. Trong trường hợp này, ảnh được tái tạo có thể được xuất ra dưới dạng ảnh được giải mã, và có thể được lưu trữ trong bộ đệm ảnh được giải mã hoặc bộ nhớ 250 của thiết bị giải mã mà được sử dụng sau làm ảnh tham chiếu trong thủ tục dự đoán liên ảnh khi ảnh được giải mã. Như được mô tả ở trên, thủ tục lọc trong vòng trong bước S1650 có thể gồm thủ tục lọc khử khối, thủ tục bù thích ứng mẫu (SAO), thủ tục lọc vòng thích ứng (ALF), và/hoặc thủ tục lọc song phương. Một số thủ tục hoặc tất cả các thủ tục này có thể được lược bỏ. Bên cạnh đó, một hoặc một số thủ tục trong số thủ tục lọc khử khối, thủ tục bù thích ứng mẫu (SAO), thủ tục lọc vòng thích ứng (ALF) và thủ tục lọc song phương có thể được áp dụng một cách tuần tự, hoặc tất cả các thủ tục đều có thể được áp dụng một cách tuần tự. Ví dụ, thủ tục lọc khử khối có thể được áp dụng cho ảnh được tái tạo, và sau đó thủ tục SAO có thể được thực hiện. Theo cách thay thế, ví dụ, thủ tục lọc khử khối có thể được áp dụng cho ảnh được tái tạo, và sau đó thủ tục

ALF có thể được thực hiện. Việc này có thể được thực hiện theo cùng cách thức như trong thiết bị mã hóa.

Fig.17 thể hiện ví dụ về thủ tục mã hóa ảnh sơ lược mà phương án (các phương án) của sáng chế này có thể áp dụng được vào đó. Trên Fig.17, bước S1710 có thể được thực hiện bởi đơn vị dự đoán gồm đơn vị dự đoán trong ảnh 185 hoặc đơn vị dự đoán liên ảnh 180 của thiết bị mã hóa được mô tả ở trên với sự tham khảo đến Fig.2. Bước S1720 có thể được thực hiện bởi bộ xử lý phần dư gồm bộ biến đổi 120 và/hoặc bộ lượng tử hóa 130. Bước S1730 có thể được thực hiện bởi bộ mã hóa entropi 190. Bước S1710 có thể gồm thủ tục dự đoán liên ảnh/trong ảnh được mô tả trong sáng chế này. Bước S1720 có thể gồm thủ tục xử lý phần dư được mô tả trong sáng chế này. Bước S1730 có thể gồm thủ tục mã hóa thông tin được mô tả trong sáng chế này.

Tham khảo đến Fig.17, thủ tục mã hóa ảnh có thể gồm một cách sơ lược, như được mô tả ở trên với sự tham khảo đến Fig.2, thủ tục để mã hóa thông tin (ví dụ, thông tin dự đoán, thông tin phần dư, và thông tin phân vùng) cho việc tái tạo ảnh và xuất ra thông tin trong dạng luồng bit, thủ tục để tạo ra ảnh được tái tạo cho ảnh hiện tại, và thủ tục (tùy chọn) để áp dụng việc lọc trong vòng cho ảnh được tái tạo. Thiết bị mã hóa có thể dẫn xuất các mẫu dư (được sửa đổi) từ các hệ số biến đổi được lượng tử hóa thông qua bộ khử lượng tử hóa 140 và bộ biến đổi ngược 150, và có thể tạo ra ảnh được tái tạo trên cơ sở của các mẫu dự đoán mà được xuất ra trong bước S1710 và các mẫu dư (được sửa đổi). Ảnh được tái tạo mà được tạo ra có thể giống với ảnh được tái tạo mà được tạo ra bởi thiết bị giải mã được mô tả ở trên. Thủ tục lọc trong vòng có thể được thực hiện trên ảnh được tái tạo để tạo ra ảnh được tái tạo mà được sửa đổi. Ảnh được tái tạo mà được sửa đổi có thể được lưu trữ trong bộ đệm ảnh được giải mã hoặc bộ nhớ 170. Theo cách tương tự với trường hợp trong thiết bị giải mã, thì ảnh được tái tạo mà được sửa đổi có

thể được sử dụng sau làm ảnh tham chiếu trong thủ tục dự đoán liên ảnh khi ảnh được mã hóa. Như được mô tả ở trên, trong một số trường hợp, một số hoặc tất cả thủ tục lọc trong vòng có thể được lược bỏ. Khi thủ tục lọc trong vòng được thực hiện, thì thông tin (thông số) liên quan đến việc lọc trong vòng có thể được mã hóa bởi bộ mã hóa entrôpi 190 và được xuất ra trong dạng luồng bit. Thiết bị giải mã có thể thực hiện thủ tục lọc trong vòng trên cơ sở của thông tin liên quan đến việc lọc theo cùng cách thức như thiết bị mã hóa.

Thông qua thủ tục lọc trong vòng này, thì các nhiễu, như là thành phần tạo tác khối và thành phần tạo tác tiếng vọng, tạo ra trong quá trình tạo mã hình ảnh/video có thể được giảm, và chất lượng trực quan chủ quan/khách quan có thể được cải thiện. Bên cạnh đó, cả thiết bị mã hóa và thiết bị giải mã đều thực hiện thủ tục lọc trong vòng, để thiết bị mã hóa và thiết bị giải mã có thể dẫn xuất cùng kết quả dự đoán, nên độ tin cậy của việc tạo mã ảnh có thể được tăng, và lượng dữ liệu cần được truyền để tạo mã ảnh có thể được giảm.

Như được mô tả ở trên, thủ tục tái tạo ảnh có thể được thực hiện trong thiết bị giải mã cũng như thiết bị mã hóa. Khối được tái tạo có thể được tạo ra trên cơ sở của việc dự đoán trong ảnh/việc dự đoán liên ảnh trên cơ sở mỗi khối, và ảnh được tái tạo gồm các khối được tái tạo có thể được tạo ra. Khi nhóm ảnh hiện tại/lát/phiên là nhóm ảnh/lát/phiên I, thì các khối được gồm trong nhóm ảnh hiện tại/lát/phiên có thể được tái tạo chỉ trên cơ sở của việc dự đoán trong ảnh. Trong lúc đó, khi nhóm ảnh hiện tại/lát/phiên là nhóm ảnh/lát/phiên P hoặc B, thì các khối được gồm trong nhóm ảnh hiện tại/lát/phiên có thể được tái tạo trên cơ sở của việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh. Trong trường hợp này, việc dự đoán liên ảnh có thể được áp dụng cho một số khối trong nhóm ảnh hiện tại/lát/phiên và việc dự đoán trong ảnh có thể được áp

dụng cho các khối còn lại. Thành phần màu sắc của ảnh có thể gồm thành phần độ chói và thành phần sắc độ. Trừ khi bị giới hạn một cách rõ ràng trong sáng chế này, còn không thì phương pháp và các phương án được đề xuất trong sáng chế này có thể được áp dụng cho thành phần độ chói và thành phần sắc độ.

Ví dụ về việc phân cấp và cấu trúc tạo mã

Video/hình ảnh được tạo mã theo sáng chế này có thể được xử lý theo, ví dụ, việc phân cấp và cấu trúc tạo mã mà được mô tả sau.

Fig.18 là hình thể hiện cấu trúc phân cấp cho hình ảnh được tạo mã. Hình ảnh được tạo mã có thể được phân chia thành lớp tạo mã video (Video Coding Hierarchy, VCL) mà xử lý quy trình giải mã của hình ảnh và chính nó, hệ thống con để truyền và lưu trữ thông tin được mã hóa, và lớp trừu tượng mạng (Network Abstraction Hierarchy, NAL) mà có mặt ở giữa VCL và hệ thống con, và phụ trách chức năng thích ứng mạng.

Trong VCL, dữ liệu VCL gồm dữ liệu hình ảnh được nén (dữ liệu lát) có thể được tạo ra. Theo cách thay thế, tập hợp thông số gồm thông tin, như là tập hợp thông số ảnh (Picture Parameter Set, PPS), tập hợp thông số chuỗi (Sequence Parameter Set, SPS), và tập hợp thông số video (Video Parameter Set, VPS), hoặc thông điệp thông tin tăng cường bổ sung (Supplemental Enhancement Information, SEI) được đòi hỏi theo cách bổ sung cho quy trình giải mã hình ảnh có thể được tạo ra.

Trong NAL, thông tin phân đầu (phần đầu đơn vị NAL) được cộng vào phụ tải chuỗi byte thô (Raw Byte Sequence Payload, RBSP) được tạo ra trong VCL để đơn vị NAL có thể được tạo ra. Ở đây, RBSP đề cập đến dữ liệu lát, tập hợp thông số, và thông điệp SEI được tạo ra trong VCL. Phần đầu đơn vị NAL có thể gồm thông tin loại đơn vị NAL mà được chỉ rõ theo dữ liệu RBSP được gồm trong đơn vị NAL.

Như được thể hiện trên hình vẽ, đơn vị NAL có thể được phân chia thành đơn vị NAL VCL và đơn vị NAL không VCL phụ thuộc vào RBSP được tạo ra trong VCL. Đơn vị NAL VCL có thể đề cập đến đơn vị NAL mà gồm thông tin (dữ liệu lát) về hình ảnh. Đơn vị NAL không VCL có thể đề cập đến đơn vị NAL mà gồm thông tin (tập hợp thông số hoặc thông điệp SEI) được đòi hỏi để giải mã hình ảnh.

Với thông tin phần đầu được đính kèm theo tiêu chuẩn dữ liệu của hệ thống con, thì đơn vị NAL VCL và đơn vị NAL không VCL có thể được truyền qua mạng. Ví dụ, đơn vị NAL có thể được biến đổi trong dạng dữ liệu của tiêu chuẩn được xác định trước, như là định dạng tệp H.266/VVC, iao thức vận chuyển theo thời gian thực (Real-time Transport Protocol, RTP), hoặc phát luồng vận chuyển (Transport Stream, TS), và dữ liệu kết quả có thể được truyền qua các mạng khác nhau

Như được mô tả ở trên, liên quan tới đơn vị NAL, thì loại đơn vị NAL của đơn vị NAL có thể được chỉ rõ theo cấu trúc dữ liệu RBSP được gồm trong đơn vị NAL, và thông tin về loại đơn vị NAL có thể được lưu trữ trong phần đầu đơn vị NAL và được phát tín hiệu.

Ví dụ, phụ thuộc vào việc liệu đơn vị NAL có gồm thông tin (dữ liệu lát) về hình ảnh hay không, thì việc phân loại thành loại đơn vị NAL VCL và loại đơn vị NAL không VCL được tiến hành. Loại đơn vị NAL VCL có thể được phân loại theo đặc điểm và loại của ảnh được gồm trong đơn vị NAL VCL, và loại đơn vị NAL không VCL có thể được phân loại theo loại của tập hợp thông số.

Các loại đơn vị NAL được chỉ rõ theo các loại tập hợp thông số mà các loại đơn vị NAL không VCL gồm, sẽ được liệt kê dưới đây như là ví dụ.

- Đơn vị NAL tập hợp thông số thích ứng (Adaptation Parameter Set, APS): Loại đơn vị NAL gồm APS

- Đơn vị NAL tập hợp thông số giải mã (Decoding Parameter Set, DPS): Loại đơn vị NAL gồm DPS

- Đơn vị NAL tập hợp thông số video (VPS): Loại đơn vị NAL gồm VPS

- Đơn vị NAL tập hợp thông số chuỗi (SPS): Loại đơn vị NAL gồm SPS

- Đơn vị NAL tập hợp thông số ảnh (PPS): Loại đơn vị NAL gồm PPS

Các loại đơn vị NAL được đề cập ở trên có thông tin cú pháp cho loại đơn vị NAL, và thông tin cú pháp có thể được lưu trữ trong phần đầu đơn vị NAL và được phát tín hiệu. Ví dụ, thông tin cú pháp có thể là `nal_unit_type`, và các loại đơn vị NAL có thể được chỉ rõ bởi các giá trị `nal_unit_type`.

Phần đầu lát (cú pháp phần đầu lát) có thể gồm thông tin/thông số có thể áp dụng được chung cho lát. APS (cú pháp APS) hoặc PPS (cú pháp PPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho một hoặc nhiều ảnh. SPS (cú pháp SPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho một hoặc nhiều chuỗi. VPS (cú pháp VPS) có thể gồm thông tin/thông số có thể áp dụng được chung cho nhiều lớp. DPS (cú pháp DPS) có thể gồm thông tin/thông số có thể được áp dụng chung cho toàn bộ video. DPS có thể gồm thông tin/thông số liên quan đến sự móc nối của các chuỗi video được tạo mã (CVS). Trong sáng chế này, cú pháp mức cao (HLS) có thể gồm ít nhất một được lựa chọn từ nhóm gồm cú pháp APS, cú pháp PPS, cú pháp SPS, cú pháp VPS, cú pháp DPS, và cú pháp phần đầu lát.

Trong sáng chế, thông tin hình ảnh/video được mã hóa bởi thiết bị mã hóa và được phát tín hiệu trong dạng luồng bit đến thiết bị giải mã có thể không chỉ gồm thông tin liên quan đến việc phân vùng ảnh, thông tin dự đoán liên ảnh/trong ảnh, thông tin phần dư, thông tin lọc trong vòng cũng như thông tin được gồm trong phần đầu lát, thông tin

được gồm trong phần đầu ảnh, thông tin được gồm trong APS, thông tin được gồm trong PPS, thông tin được gồm trong SPS, thông tin được gồm trong VPS và/hoặc thông tin được gồm trong VPS.

Tổng quan của sự biến đổi màu sắc thích ứng (Adaptive Color Transform, ACT)

Sự biến đổi màu sắc thích ứng (ACT) là công nghệ biến đổi (chuyển đổi) không gian màu sắc để loại bỏ sự chồng lấn không cần thiết giữa các thành phần màu sắc, và đã được sử dụng trong phiên bản mở rộng nội dung màn hình HEVC. Điều này cũng có thể được áp dụng cho VVC.

Trong phần mở rộng nội dung màn hình HEVC (phần mở rộng SCC HEVC), ACT đã được sử dụng để biến đổi một cách thích ứng phần dư dự đoán từ không gian màu sắc sẵn có thành không gian màu sắc YCgCo. Một không gian màu sắc trong số hai không gian màu sắc có thể được lựa chọn theo cách tùy chọn bằng cách phát tín hiệu một cờ ACT đối với mỗi cơ sở biến đổi.

Ví dụ, giá trị thứ nhất (ví dụ, 1) của cờ có thể chỉ ra rằng phần dư của cơ sở biến đổi được mã hóa trong không gian màu sắc gốc. Giá trị thứ hai (ví dụ, 1) của cờ có thể chỉ ra rằng phần dư của cơ sở biến đổi được mã hóa trong không gian màu sắc YCgCo.

Fig.19 là hình thể hiện phương án của quy trình giải mã mà ACT được áp dụng vào đó. Trong phương án trên Fig.19, việc dự đoán được bù chuyển động có thể tương ứng với việc dự đoán liên ảnh trong sáng chế.

Như được thể hiện trên Fig.19, ảnh được tái tạo (hoặc khối được tái tạo, mảng mẫu được tái tạo, mẫu (các mẫu) được tái tạo, tín hiệu được tái tạo) có thể được tạo ra trên cơ sở của giá trị đầu ra dự đoán và giá trị đầu ra phần dư. Ở đây, giá trị đầu ra phần dư có thể là giá trị đầu ra biến đổi ngược. Ở đây, việc biến đổi ngược có thể là việc biến đổi

ngược bình thường. Ở đây, việc biến đổi ngược bình thường có thể là việc biến đổi ngược dựa trên MTS hoặc sự biến đổi không thể tách rời được tần số thấp (Low Frequency Non-Separable Transform, LFNST) ngược.

Ở đây, giá trị đầu ra dự đoán có thể là khối dự đoán, việc dự đoán mảng mẫu, mẫu (các mẫu) dự đoán hoặc tín hiệu dự đoán. Giá trị đầu ra phần dư có thể là khối dư, mảng mẫu dư, mẫu (các mẫu) dư, hoặc tín hiệu dư.

Ví dụ, liên quan tới thiết bị mã hóa, thì quy trình ACT có thể được thực hiện trên các mẫu dư được dẫn xuất trên cơ sở của các mẫu dự đoán. Bên cạnh đó, giá trị đầu ra của quy trình ACT có thể được cung cấp dưới dạng đầu vào của quy trình biến đổi bình thường. Ở đây, quy trình biến đổi bình thường có thể là việc biến đổi dựa trên MTS hoặc LFNST.

Thông tin (thông số) về ACT (ngược) có thể được tạo ra và được mã hóa bởi thiết bị mã hóa, và có thể được truyền đến thiết bị giải mã trong dạng luồng bit.

Thiết bị giải mã có thể thu nhận, phân tích cú pháp, và giải mã thông tin (thông số) liên quan đến ACT (ngược), và có thể thực hiện ACT ngược trên cơ sở của thông tin (thông số) liên quan đến ACT (ngược).

Trên cơ sở của ACT ngược, thì các mẫu dư (hoặc khối dư) (được sửa đổi) có thể được dẫn xuất. Ví dụ, các hệ số (biến đổi) có thể được dẫn xuất bằng cách áp dụng việc khử lượng tử hóa cho các hệ số (biến đổi) được lượng tử hóa. Bên cạnh đó, các mẫu dư có thể được dẫn xuất bằng cách thực hiện việc biến đổi ngược trên các hệ số (biến đổi). Bên cạnh đó, các mẫu dư (được sửa đổi) có thể được thu nhận bằng cách áp dụng ACT ngược cho các mẫu dư. Thông tin (thông số) về ACT (ngược) sẽ được mô tả chi tiết sau.

Trong phương án, hàm biến đổi lỗi được sử dụng trong HEVC có thể được sử dụng làm hàm biến đổi lỗi (nhân biến đổi) cho việc biến đổi không gian màu sắc. Ví dụ, các ma trận cho việc biến đổi xuôi và việc biến đổi ngược như được thể hiện trong các phương trình dưới đây sẽ có thể được sử dụng.

[Phương trình 1]

$$\begin{bmatrix} C'_0 \\ C'_1 \\ C'_2 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 2 & -1 & -1 \\ 0 & -2 & 2 \end{bmatrix} \begin{bmatrix} C_0 \\ C_1 \\ C_2 \end{bmatrix} / 4$$

[Phương trình 2]

$$\begin{bmatrix} C_0 \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} C'_0 \\ C'_1 \\ C'_2 \end{bmatrix}$$

Ở đây, C_0 , C_1 , và C_2 có thể tương ứng với G, B, và R. Ở đây, G biểu thị thành phần màu sắc xanh lục, B biểu thị thành phần màu sắc xanh lam, và R biểu thị thành phần màu sắc đỏ. Bên cạnh đó, C'_0 , C'_1 , và C'_2 có thể tương ứng với Y, C_g , và C_o . Ở đây, Y biểu thị độ chói, C_g biểu thị sắc độ xanh lam, và C_o biểu thị thành phần sắc độ cam.

Bên cạnh đó, để bù cho sự thay đổi phạm vi linh động của phần dư trước và sau việc biến đổi màu sắc, thì việc điều chỉnh QP có thể được áp dụng cho phần dư biến đổi bởi $(-5, -5, -3)$. Các chi tiết của việc điều chỉnh QP sẽ được mô tả sau.

Trong lúc đó, trong các quy trình mã hóa và giải mã theo phương án, khi ACT có thể áp dụng được, thì các giới hạn sau đây có thể được áp dụng.

- Trong trường hợp của việc mã hóa/giải mã cây kép, thì ACT được giải hoạt. Ví dụ, ACT có thể chỉ được áp dụng cho việc mã hóa/giải mã cây đơn.
- Khi việc mã hóa và giải mã ISP được áp dụng, thì ACT có thể được giải hoạt.

- Đối với khối sắc độ mà BDPCM được áp dụng vào đó, thì ACT có thể được giải hoạt. Chỉ đối với khối độ chói mà BDPCM được áp dụng vào đó, thì ACT có thể được kích hoạt.

- Khi việc áp dụng ACT là khả thi, thì CCLM có thể được giải hoạt.

Fig.20 là hình thể hiện phương án của bảng cú pháp tập hợp thông số chuỗi mà trong đó phần tử cú pháp liên quan đến ACT được phát tín hiệu.

Các hình vẽ từ Fig.21 đến Fig.27 là các hình thể hiện một cách liên tiếp phương án của bảng cú pháp của cơ sở mã hóa mà trong đó phần tử cú pháp liên quan đến ACT được phát tín hiệu.

Như được thể hiện trên Fig.20, theo như cờ kích hoạt ACT chỉ ra xem liệu ACT có được kích hoạt trong quy trình giải mã hay không, thì `sps_act_enabled_flag` 2010 có thể được sử dụng.

Giá trị thứ nhất (ví dụ, 0) của `sps_act_enabled_flag` có thể chỉ ra rằng ACT không được sử dụng và cờ `cu_act_enabled_flag` 2110, 2710 chỉ ra xem liệu ACT được áp dụng trong cơ sở mã hóa hay không sẽ không được cung cấp trong cú pháp cho cơ sở mã hóa.

Giá trị thứ hai (ví dụ, 1) của `sps_act_enabled_flag` có thể chỉ ra rằng ACT có thể được sử dụng và `cu_act_enabled_flag` có thể được cung cấp trong cú pháp cho cơ sở mã hóa.

Khi `sps_act_enabled_flag` không được thu nhận trong luồng bit, thì giá trị của `sps_act_enabled_flag` có thể được dẫn xuất làm giá trị thứ nhất (ví dụ, 0).

Bên cạnh đó, như được thể hiện trên Fig.21, theo như các cờ ACT chỉ ra xem liệu phần dư của cơ sở mã hóa hiện tại có được mã hóa trong không gian màu sắc YCgCo hay không, thì `cu_act_enabled_flag` 2110, 2710 có thể được sử dụng.

Giá trị thứ nhất (ví dụ, 0) của `cu_act_enabled_flag` có thể chỉ ra rằng phần dư của cơ sở mã hóa hiện tại được mã hóa trong không gian màu sắc gốc. Giá trị thứ hai (ví dụ, 1) của `cu_act_enabled_flag` có thể chỉ ra rằng phần dư của cơ sở mã hóa hiện tại được mã hóa trong không gian màu sắc YCgCo.

Khi `cu_act_enabled_flag` không được cung cấp trong luồng bit, thì cò có thể được dẫn xuất làm giá trị thứ nhất (ví dụ, 0). Ở đây, không gian màu sắc gốc có thể là không gian màu sắc RGB.

Phương pháp dẫn xuất QP của cơ sở biến đổi bằng cách sử dụng phần bù QP ACT

Trong phương án, trong quy trình định cỡ cho hệ số biến đổi, thì quy trình dẫn xuất của thông số lượng tử hóa và quy trình cập nhật Qp có thể được thực hiện như sau. Ví dụ, quy trình dẫn xuất thông số lượng tử hóa có thể được thực hiện nhờ sử dụng các thông số sau đây.

- Các tọa độ độ chói (`xCb`, `yCb`) chỉ ra các tọa độ tương đối của mẫu độ chói trên cùng bên trái của khối mã hóa hiện tại đối với mẫu độ chói trên cùng bên trái của ảnh hiện tại,

- Biến `cbWidth` chỉ ra chiều rộng của khối mã hóa hiện tại trên cơ sở từng mẫu độ chói,

- Biến `cbHeight` chỉ ra chiều cao của khối mã hóa hiện tại trên cơ sở từng mẫu độ chói

- Biến `treeType` chỉ ra xem liệu cây đơn (`SINGLE_TREE`) hay cây kép được sử dụng để phân vùng nút cây tạo mã hiện tại, và chỉ ra, khi cây kép được sử dụng, thì liệu cây kép là cây kép thành phần độ chói (`DAUL_TREE_LUMA`) hay cây kép thành phần sắc độ (`DAUL_TREE_CHROMA`)

Trong quy trình này, thì thông số lượng tử hóa độ chói $Qp'Y$, các thông số lượng tử hóa sắc độ $Qp'Cb$, $Qp'Cr$, và $Qp'CbCr$ có thể được dẫn xuất.

Biến vị trí độ chói (xQg , yQg) có thể chỉ ra vị trí của mẫu độ chói trên cùng bên trái của nhóm lượng tử hóa hiện tại tương ứng với mẫu trên cùng bên trái của ảnh hiện tại. Ở đây, vị trí ngang xQg và vị trí dọc yQg có thể được thiết lập là bằng các giá trị của biến $CuQgTopLeftX$ và biến $CuQgTopLeftY$, một cách tương ứng. Các biến $CuQgTopLeftX$ và $CuQgTopLeftY$ có thể được định rõ là các giá trị được xác định trước trong cú pháp cây tạo mã như được thể hiện trên Fig.28.

Ở đây, nhóm lượng tử hóa hiện tại có thể là khu vực hình tứ giác ở trong khối cây tạo mã, và có thể chia sẻ cùng giá trị qPY_PRED . Chiều rộng và chiều cao của nó có thể bằng chiều rộng và chiều cao của cây tạo mã nút mà trong đó vị trí mẫu độ chói trên cùng bên trái được chỉ định cho mỗi biến trong số $CuQgTopLeftX$ và $CuQgTopLeftY$.

Khi $treeType$ là $SINGLE_TREE$ hoặc $DUAL_TREE_LUMA$, thì giá trị dự đoán thông số lượng tử hóa độ chói qPY_PRED có thể được dẫn xuất như trong các bước sau đây.

1. Biến qPY_PRED có thể được dẫn xuất như sau.

(Điều kiện 1) Khi một điều kiện bất kỳ trong số các điều kiện sau đây là đúng, thì giá trị của qPY_PRED có thể được thiết lập thành cùng giá trị như $SliceQpY$ (ở đây, $SliceQpY$ chỉ ra giá trị ban đầu của thông số lượng tử hóa QpY cho tất cả các lát trong ảnh, và điều này có thể được thu nhận từ luồng bit). Theo cách thay thế, giá trị của qPY_PRED có thể được thiết lập thành giá trị của thông số lượng tử hóa độ chói QpY của cơ sở mã hóa độ chói cuối cùng của nhóm lượng tử hóa ngay trước đó theo thứ tự giải mã.

- (Điều kiện 1-1) Khi nhóm lượng tử hóa hiện tại là nhóm lượng tử hóa thứ nhất trong lát

- (Điều kiện 1-2) Khi nhóm lượng tử hóa hiện tại là nhóm lượng tử hóa thứ nhất trong phiên

- (Điều kiện 1-3) Khi nhóm lượng tử hóa hiện tại là nhóm lượng tử hóa thứ nhất trong hàng CTB trong phiên và sự đồng bộ hóa được xác định trước xảy ra (ví dụ, khi `entrôpi_coding_sync_enabled_flag` có giá trị là 1)

2. Giá trị của biến qP_{Y_A} có thể được dẫn xuất như sau.

(Điều kiện 2) Khi ít nhất một điều kiện trong số các điều kiện sau đây là đúng, thì giá trị của qP_{Y_A} có thể được thiết lập thành giá trị của qP_{Y_PRED} . Theo cách thay thế, giá trị của qP_{Y_A} có thể được thiết lập thành giá trị của thông số lượng tử hóa độ chói Q_{pY} của cơ sở mã hóa mà gồm độ chói mã hóa khối bao trùm vị trí mẫu độ chói ($x_{Qg}-1$, y_{Qg}).

- (Điều kiện 2-1) Đối với khối được nhận dạng bởi vị trí mẫu (x_{Cb} , y_{Cb}), thì khối được nhận dạng bởi vị trí mẫu ($x_{Qg} - 1$, y_{Qg}) không phải là khối lân cận khả dụng,

- (Điều kiện 2-2) Khi CTB gồm độ chói mã hóa khối bao trùm vị trí mẫu độ chói ($x_{Qg} - 1$, y_{Qg}) là khác với CTB gồm khối mã hóa độ chói hiện tại tại vị trí mẫu độ chói (x_{Cb} , y_{Cb}), ví dụ, khi tất cả điều kiện sau đây đều đúng

- (Điều kiện 2-2-1) Giá trị của ($x_{Qg} - 1$) $>>$ $CtbLog2SizeY$ là khác với giá trị của (x_{Cb}) $>>$ $CtbLog2SizeY$

- (Điều kiện 2-2-2) Giá trị của (y_{Qg}) $>>$ $CtbLog2SizeY$ là khác với giá trị của (y_{Cb}) $>>$ $CtbLog2SizeY$

3. Giá trị của biến qP_{Y_B} có thể được dẫn xuất như sau.

(Điều kiện 3) Khi ít nhất một điều kiện trong số các điều kiện sau đây là đúng, thì giá trị của qP_{Y_B} có thể được thiết lập thành giá trị của qP_{Y_PRED} . Theo cách thay thế, giá trị của qP_{Y_B} có thể được thiết lập thành giá trị của thông số lượng tử hóa độ chói Qp_Y của cơ sở mã hóa mà gồm độ chói mã hóa khối bao trùm vị trí mẫu độ chói ($xQg, yQg-1$).

- (Điều kiện 3-1) Đối với khối được nhận dạng bởi vị trí mẫu (xCb, yCb), khi khối được nhận dạng bởi vị trí mẫu ($xQg, yQg-1$) không phải là khối lân cận khả dụng,

- (Điều kiện 3-2) Khi CTB gồm độ chói mã hóa khối bao trùm vị trí mẫu độ chói ($xQg, yQg-1$) là khác với CTB gồm khối mã hóa độ chói hiện tại tại vị trí mẫu độ chói (xCb, yCb), ví dụ, khi tất cả các điều kiện sau đây đều đúng

- (Điều kiện 3-2-1) Giá trị của (xQg) $>>$ $CtbLog2SizeY$ là khác với giá trị của (xCb) $>>$ $CtbLog2SizeY$

- (Điều kiện 3-2-2) Giá trị của ($yQg - 1$) $>>$ $CtbLog2SizeY$ là khác với giá trị của (yCb) $>>$ $CtbLog2SizeY$

4. Giá trị dự đoán thông số lượng tử hóa độ chói qP_{Y_PRED} có thể được dẫn xuất như sau.

Khi tất cả các điều kiện sau đây đều đúng, thì qP_{Y_PRED} có thể được thiết lập thành thông số lượng tử hóa độ chói Qp_Y của cơ sở mã hóa mà gồm độ chói mã hóa khối bao trùm vị trí mẫu độ chói ($xQg, yQg - 1$).

- (Điều kiện 3-1) Đối với khối được nhận dạng bởi vị trí mẫu (xCb, yCb), thì khối được nhận dạng bởi vị trí mẫu ($xQg, yQg-1$) là khối lân cận khả dụng

- Khi nhóm lượng tử hóa hiện tại là nhóm lượng tử hóa thứ nhất trong hàng CTB trong phiên

Trong lúc đó, khi tất cả các điều kiện sau đây đều không đúng, thì qP_{Y_PRED} có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 3]

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

Biến Q_{pY} có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 4]

$$Q_{pY} = ((qP_{Y_PRED} + CuQpDeltaVal + 64 + 2 * QpBdOffset) \% (64 + QpBdOffset)) - QpBdOffset$$

Ở đây, $CuQpDeltaVal$ chỉ ra sự chênh lệch giữa thông số lượng tử hóa độ chói cho cơ sở mã hóa và giá trị dự đoán của nó. Giá trị của nó có thể được thu nhận từ luồng bit. $QpBdOffset$ chỉ ra phần bù phạm vi thông số lượng tử hóa độ chói và sắc độ. $QpBdOffset$ có thể được thiết lập trước thành hằng số được xác định trước hoặc được thu nhận từ luồng bit. Ví dụ, $QpBdOffset$ có thể được tính toán bằng cách nhân hằng số được xác định trước với giá trị của phần tử cú pháp mà chỉ ra độ sâu bit của mẫu độ chói hoặc sắc độ. Thông số lượng tử hóa độ chói $Q_{p'Y}$ có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 5]

$$Q_{p'Y} = Q_{pY} + QpBdOffset$$

Khi giá trị của biến $ChromaArrayType$ chỉ ra loại mảng sắc độ không phải là giá trị thứ nhất (ví dụ, 0) và $treeType$ là $SINGLE_TREE$ hoặc $DUAL_TREE_CHROMA$, thì việc xử lý sau đây có thể được thực hiện.

- Khi giá trị của `treeType` là `DUAL_TREE_CHROMA`, thì giá trị của biến `QpY` có thể được thiết lập thành cùng giá trị như giá trị của thông số lượng tử hóa độ chói `QpY` của cơ sở mã hóa độ chói bao trùm vị trí mẫu độ chói $(x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2)$.

- Các biến `qPCb`, `qPCr`, và `qPCbCr` có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 6]

$$qP_{Chroma} = \text{Clip3}(-QpBdOffset, 63, QpY)$$

$$qP_{Cb} = \text{ChromaQpTable}[0][qP_{Chroma}]$$

$$qP_{Cr} = \text{ChromaQpTable}[1][qP_{Chroma}]$$

$$qP_{CbCr} = \text{ChromaQpTable}[2][qP_{Chroma}]$$

Các thông số lượng tử hóa sắc độ Qp'_{Cb} và Qp'_{Cr} cho các thành phần Cb và Cr và thông số lượng tử hóa sắc độ Qp'_{CbCr} cho việc tạo mã Cb-Cr liên kết có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 7]

$$Qp'_{Cb} = \text{Clip3}(-QpBdOffset, 63, qP_{Cb} + pps_cb_qp_offset + slice_cb_qp_offset + CuQpOffset_{Cb}) + QpBdOffset$$

$$Qp'_{Cr} = \text{Clip3}(-QpBdOffset, 63, qP_{Cr} + pps_cr_qp_offset + slice_cr_qp_offset + CuQpOffset_{Cr}) + QpBdOffset$$

$$Qp'_{CbCr} = \text{Clip3}(-$$

$$QpBdOffset, 63, qP_{CbCr} + pps_joint_cbr_qp_offset + slice_joint_cbr_qp_offset + \\ CuQpOffset_{CbCr}) + QpBdOffset$$

Trong phương trình ở trên, $pps_cb_qp_offset$ và $pps_cr_qp_offset$ là các phần bù được sử dụng để dẫn xuất Qp'_{Cb} và Qp'_{Cr} , và có thể được thu nhận từ luồng bit cho tập hợp thông số ảnh. $slice_cb_qp_offset$ và $slice_cr_qp_offset$ là các phần bù được sử dụng để dẫn xuất Qp'_{Cb} và Qp'_{Cr} , và có thể được thu nhận từ luồng bit cho phần đầu lát. $CuQpOffset_{Cb}$ và $CuQpOffset_{Cr}$ là các phần bù được sử dụng để dẫn xuất Qp'_{Cb} và Qp'_{Cr} , và có thể được thu nhận từ luồng bit cho cơ sở biến đổi.

Bên cạnh đó, ví dụ, quy trình khử lượng tử hóa cho hệ số biến đổi có thể được thực hiện nhờ sử dụng các thông số sau đây.

- Các tọa độ độ chói ($xTbY$, $yTbY$) tham chiếu đến các tọa độ tương đối của mẫu trên cùng bên trái của khối biến đổi độ chói hiện tại đối với mẫu độ chói trên cùng bên trái của ảnh hiện tại

- Biến $nTbW$ chỉ ra chiều rộng của khối biến đổi
- Biến $nTbH$ chỉ ra chiều cao của khối biến đổi
- Biến $cIdx$ chỉ ra thành phần màu sắc của khối hiện tại

Đầu ra của quy trình này có thể là mảng d của các hệ số biến đổi được định cỡ. Ở đây, kích cỡ của mảng d có thể là $(nTbW) \times (nTbH)$. Các phần tử riêng lẻ cấu thành mảng này có thể được nhận dạng là $d[x][y]$.

Để đạt được điều này, thì thông số lượng tử hóa qP có thể được dẫn xuất như sau. Khi $cIdx$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 8]

$$qP = Qp'_Y$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì việc dẫn xuất có thể được tiến hành như được thể hiện trong phương trình dưới đây.

[Phương trình 9]

$$qP = Qp'_{CbCr}$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 10]

$$qP = Qp'_{cb}$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 11]

$$qP = Qp'_{Cr}$$

Sau đây, thông số lượng tử hóa qP có thể được cập nhật như sau. Bên cạnh đó, các biến rectNonTsFlag và bdShift có thể được dẫn xuất như sau. Ví dụ, khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 0 (ví dụ, khi việc biến đổi của khối biến đổi hiện tại không được bỏ qua), thì việc dẫn xuất có thể được tiến hành như được thể hiện trong phương trình dưới đây.

[Phương trình 12]

$$qP = qP - (cu_act_enabled_flag[xTbY][yTbY] ? 5 : 0)$$

$$rectNonTsFlag = 0$$

bdShift = 10

Theo cách thay thế, khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 1 (ví dụ, việc biến đổi của khối biến đổi hiện tại được bỏ qua), thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 13]

$$qP = \text{Max}(QpPrimeTsMin, qP) - (cu_act_enabled_flag[xTbY][yTbY] ? 5 : 0)$$

$$rectNonTsFlag = ((\text{Log2}(nTbW) + \text{Log2}(nTbH)) \& 1) == 1$$

$$bdShift = \text{BitDepth} + (rectNonTsFlag ? 1 : 0) + ((\text{Log2}(nTbW) + \text{Log2}(nTbH)) / 2) - pic_dep_quant_enabled_flag$$

Ở đây, QpPrimeTsMin có thể chỉ ra giá trị thông số lượng tử hóa tối thiểu được chấp thuận khi chế độ bỏ qua biến đổi được áp dụng. Giá trị này có thể được xác định là hằng số được xác định trước hoặc có thể được dẫn xuất từ phân tử cú pháp của luồng bit liên quan đến nó.

Ở đây, các hậu tố Y, Cb, và Cr có thể biểu thị các thành phần màu sắc G, B, và R trong mô hình màu sắc RGB, hoặc các thành phần màu sắc Y, Cg, và Co trong mô hình màu sắc YCgCo.

Tổng quan của việc điều biến mã xung chênh lệch khối (Block Difference Pulse Code Modulation, BDPCM)

Thiết bị mã hóa hình ảnh và thiết bị giải mã hình ảnh theo phương án có thể thực hiện việc mã hóa vi sai của tín hiệu dư. Ví dụ, thiết bị mã hóa hình ảnh có thể mã hóa tín hiệu dư bằng cách trừ tín hiệu dư đoán khỏi tín hiệu dư của khối hiện tại, và thiết bị giải mã hình ảnh có thể giải mã tín hiệu dư bằng cách cộng tín hiệu dư đoán vào tín hiệu dư của khối hiện tại. Thiết bị mã hóa hình ảnh và thiết bị giải mã hình ảnh theo phương

án có thể thực hiện việc mã hóa vi sai của tín hiệu dư bằng cách áp dụng BDPCM được mô tả dưới đây.

BDPCM theo sáng chế có thể được thực hiện trong miền phần dư được lượng tử hóa. Miền phần dư được lượng tử hóa có thể gồm tín hiệu dư được lượng tử hóa (hoặc hệ số dư được lượng tử hóa), và, khi áp dụng BDPCM, thì sự biến đổi của tín hiệu dư được lượng tử hóa sẽ được bỏ qua. Ví dụ, khi áp dụng BDPCM, thì sự biến đổi của tín hiệu dư có thể được bỏ qua và việc lượng tử hóa có thể được thực hiện. Theo cách thay thế, miền phần dư được lượng tử hóa có thể gồm các hệ số biến đổi được lượng tử hóa.

Trong phương án mà BDPCM áp dụng vào đó, thì thiết bị mã hóa hình ảnh có thể dẫn xuất khối dư của khối hiện tại được dự đoán trong chế độ dự đoán trong ảnh và lượng tử hóa khối dư, nhờ đó dẫn xuất khối dư. Khi chế độ mã hóa vi sai của tín hiệu dư được thực hiện đối với khối hiện tại, thì thiết bị mã hóa hình ảnh có thể thực hiện việc mã hóa vi sai đối với khối dư để dẫn xuất khối dư được sửa đổi. Bên cạnh đó, thiết bị mã hóa hình ảnh có thể mã hóa thông tin chế độ mã hóa vi sai chỉ rõ chế độ mã hóa vi sai của tín hiệu dư và khối dư được sửa đổi, nhờ đó tạo ra luồng bit.

Một cách cụ thể hơn, khi BDPCM áp dụng cho khối hiện tại, thì khối được dự đoán (khối dự đoán) gồm các mẫu được dự đoán của khối hiện tại có thể được tạo ra bởi việc dự đoán trong ảnh. Trong trường hợp này, chế độ dự đoán trong ảnh để thực hiện việc dự đoán trong ảnh có thể được phát tín hiệu thông qua luồng bit và có thể được dẫn xuất dựa trên hướng dự đoán của BDPCM được mô tả dưới đây. Bên cạnh đó, trong trường hợp này, chế độ dự đoán trong ảnh có thể được xác định là một chế độ trong số chế độ dự đoán hướng dọc hoặc chế độ dự đoán hướng ngang. Ví dụ, khi hướng dự đoán của BDPCM là hướng ngang, thì chế độ dự đoán trong ảnh có thể được xác định là chế độ dự đoán hướng ngang, và khối dự đoán của khối hiện tại có thể được tạo ra bởi việc dự

đoán trong ảnh theo hướng ngang. Theo cách thay thế, khi hướng dự đoán của BDPCM là hướng dọc, thì chế độ dự đoán trong ảnh có thể được xác định là chế độ dự đoán hướng dọc, và khối dự đoán của khối hiện tại có thể được tạo ra bởi việc dự đoán trong ảnh theo hướng dọc. Khi áp dụng việc dự đoán trong ảnh theo hướng ngang, thì giá trị của điểm ảnh liền kề với phần bên trái của khối hiện tại có thể được xác định là giá trị mẫu dự đoán của các mẫu được gồm trong hàng tương ứng của khối hiện tại. Khi áp dụng việc dự đoán trong ảnh theo hướng dọc, thì giá trị của điểm ảnh liền kề với phần trên cùng của khối hiện tại có thể được xác định là giá trị mẫu dự đoán của các mẫu được gồm trong cột tương ứng của khối hiện tại. Khi áp dụng BDPCM cho khối hiện tại, thì phương pháp tạo ra khối dự đoán của khối hiện tại có thể được thực hiện một cách tương tự trong thiết bị mã hóa hình ảnh và thiết bị giải mã hình ảnh.

Khi áp dụng BDPCM cho khối hiện tại, thì thiết bị mã hóa hình ảnh có thể tạo ra khối dư gồm các mẫu dư của khối hiện tại, bằng cách trừ mẫu dự đoán khỏi khối hiện tại. Thiết bị mã hóa hình ảnh có thể lượng tử hóa khối dư và sau đó mã hóa sự chênh lệch (hoặc delta) giữa mẫu dư được lượng tử hóa và bộ dự đoán của mẫu dư được lượng tử hóa. Thiết bị giải mã hình ảnh có thể tạo ra khối dư được lượng tử hóa của khối hiện tại, bằng cách thu nhận mẫu dư được lượng tử hóa của khối hiện tại dựa trên bộ dự đoán và sự chênh lệch được tái tạo từ luồng bit. Sau đây, thiết bị giải mã hình ảnh có thể khử lượng tử hóa khối dư được lượng tử hóa và sau đó cộng nó vào khối dự đoán, nhờ đó tái tạo khối hiện tại.

Fig.29 là hình thể hiện phương pháp mã hóa mẫu dư của BDPCM theo sáng chế. Khối dư trên Fig.29 có thể được tạo ra bằng cách trừ khối dự đoán khỏi khối hiện tại trong thiết bị mã hóa hình ảnh. Khối dư được lượng tử hóa trên Fig.29 có thể được tạo ra bằng cách lượng tử hóa khối dư bởi thiết bị mã hóa hình ảnh. Trên Fig.29, $r_{i,j}$ chỉ rõ

giá trị của mẫu dư có các tọa độ (i, j) trong khối hiện tại. Khi kích cỡ của khối hiện tại là $M \times N$, thì giá trị i có thể là từ 0 đến $M-1$. Bên cạnh đó, giá trị j có thể là từ 0 đến $N-1$. Ví dụ, phần dư có thể đề cập đến sự chênh lệch giữa khối gốc và khối dự đoán. Ví dụ, $r_{i,j}$ có thể được dẫn xuất bằng cách trừ giá trị của mẫu dự đoán khỏi giá trị của mẫu gốc có các tọa độ (i, j) trong khối hiện tại. Ví dụ, $r_{i,j}$ có thể là phần dư dự đoán sau khi việc dự đoán trong ảnh ngang hoặc việc dự đoán trong ảnh dọc được thực hiện sử dụng mẫu mà không được lọc từ mẫu ranh giới trên cùng hoặc bên trái. Trong việc dự đoán trong ảnh ngang, thì giá trị của điểm ảnh lân cận bên trái được sao chép dọc theo đường cắt ngang khối dự đoán. Trong việc dự đoán trong ảnh dọc, thì đường lân cận trên cùng được sao chép vào đường riêng lẻ của khối dự đoán.

Trên Fig.29, $Q(r_{i,j})$ đề cập đến giá trị của mẫu dư được lượng tử hóa có các tọa độ (i, j) trong khối hiện tại. Ví dụ, $Q(r_{i,j})$ có thể đề cập đến giá trị được lượng tử hóa của $r_{i,j}$.

Việc dự đoán của BDPCM được thực hiện trên các mẫu dư được lượng tử hóa trên Fig.29, và khối dư được lượng tử hóa mà được sửa đổi R' có kích cỡ $M \times N$ gồm các mẫu dư được lượng tử hóa mà được sửa đổi r' có thể được tạo ra.

Khi hướng dự đoán của BDPCM là hướng ngang, thì giá trị $r'_{i,j}$ của mẫu dư được lượng tử hóa mà được sửa đổi có các tọa độ (i, j) trong khối hiện tại có thể được tính toán như được thể hiện trong phương trình dưới đây.

[Phương trình 14]

$$r'_{i,j} = \begin{cases} Q(r_{i,j}), & i = 0, \quad 0 \leq j \leq (N-1) \\ Q(r_{i,j}) - Q(r_{(i-1),j}), & 1 \leq i \leq (M-1), \quad 0 \leq j \leq (N-1) \end{cases}$$

Như được thể hiện trong phương trình 14, khi hướng dự đoán của BDPCM là hướng ngang, thì giá trị $Q(r_{0,j})$ của mẫu dự được lượng tử hóa được chỉ định như là giá trị $r'_{0,j}$ có các tọa độ $(0, j)$. Giá trị $r'_{i,j}$ có các tọa độ (i, j) khác có thể được dẫn xuất dưới dạng sự chênh lệch giữa giá trị $Q(r_{i,j})$ của mẫu dự được lượng tử hóa có các tọa độ (i, j) và giá trị $Q(r_{i-1,j})$ của mẫu dự được lượng tử hóa có các tọa độ $(i-1, j)$. Nghĩa là, thay vì mã hóa giá trị $Q(r_{i,j})$ của mẫu dự được lượng tử hóa có các tọa độ (i, j) , thì sự chênh lệch được tính toán bằng cách sử dụng giá trị $Q(r_{i-1,j})$ của mẫu dự được lượng tử hóa có các tọa độ $(i-1, j)$ làm giá trị dự đoán được dẫn xuất làm mẫu dự được lượng tử hóa mà được sửa đổi giá trị $r'_{i,j}$ và sau đó giá trị $r'_{i,j}$ được mã hóa.

Khi hướng dự đoán của BDPCM là hướng ngang, thì giá trị $r'_{i,j}$ của mẫu dự được lượng tử hóa mà được sửa đổi có các tọa độ (i, j) trong khối hiện tại có thể được tính toán như được thể hiện trong phương trình dưới đây.

[Phương trình 15]

$$r'_{i,j} = \begin{cases} Q(r_{i,j}), & 0 \leq i \leq (M-1), j = 0 \\ Q(r_{i,j}) - Q(r_{i,j-1}), & 0 \leq i \leq (M-1), 1 \leq j \leq (N-1) \end{cases}$$

Như được thể hiện trong phương trình 15, khi hướng dự đoán của BDPCM là hướng dọc, thì giá trị $Q(r_{i,0})$ của mẫu dự được lượng tử hóa được chỉ định như là giá trị $r'_{i,0}$ có các tọa độ $(i, 0)$. Giá trị $r'_{i,j}$ có các tọa độ (i, j) khác có thể được dẫn xuất dưới dạng sự chênh lệch giữa giá trị $Q(r_{i,j})$ của mẫu dự được lượng tử hóa có các tọa độ (i, j) và giá trị $Q(r_{i,j-1})$ của mẫu dự được lượng tử hóa có các tọa độ $(i, j-1)$. Nghĩa là, thay vì mã hóa giá trị $Q(r_{i,j})$ của mẫu dự được lượng tử hóa có các tọa độ (i, j) , thì sự chênh lệch được tính toán bằng cách sử dụng giá trị $Q(r_{i,j-1})$ của mẫu dự được lượng tử hóa có các tọa độ $(i, j-1)$ làm giá trị dự đoán được dẫn xuất làm mẫu dự được lượng tử hóa mà được sửa đổi giá trị $r'_{i,j}$ và sau đó giá trị $r'_{i,j}$ được mã hóa.

Như được mô tả ở trên, quy trình sửa đổi giá trị mẫu dư được lượng tử hóa hiện tại bằng cách sử dụng giá trị mẫu dư được lượng tử hóa gần kề làm giá trị dự đoán có thể được gọi là việc dự đoán BDPCM.

Cuối cùng, thiết bị mã hóa hình ảnh có thể mã hóa khối dư được lượng tử hóa mà được sửa đổi gồm các mẫu dư được lượng tử hóa mà được sửa đổi và có thể truyền khối kết quả đến thiết bị giải mã hình ảnh. Ở đây, như được mô tả ở trên, khối biến đổi dư được lượng tử hóa mà được sửa đổi sẽ không được thực hiện.

Fig.30 là hình thể hiện các khối dư được lượng tử hóa mà được sửa đổi sẽ được tạo ra bằng cách thực hiện BDPCM của sáng chế.

Trên Fig.30, BDPCM ngang thể hiện khối dư được lượng tử hóa mà được sửa đổi sẽ được tạo ra theo phương trình 14 khi hướng dự đoán của BDPCM là hướng ngang. Bên cạnh đó, BDPCM dọc thể hiện khối dư được lượng tử hóa mà được sửa đổi sẽ được tạo ra theo phương trình 15 khi hướng dự đoán của BDPCM là hướng dọc.

Fig.31 là lưu đồ thể hiện thủ tục để mã hóa khối hiện tại bằng cách áp dụng BDPCM trong thiết bị mã hóa hình ảnh theo phương án.

Trước tiên, khi khối hiện tại mà là khối mục tiêu mã hóa là đầu vào trong bước S3110, thì việc dự đoán có thể được thực hiện trên khối hiện tại để tạo ra khối dự đoán trong bước S3120. Khối dự đoán trong bước S3120 có thể là khối được dự đoán trong ảnh, và chế độ dự đoán trong ảnh có thể được xác định như được mô tả ở trên. Trên cơ sở của khối dự đoán được tạo ra trong bước S3120, thì khối dư của khối hiện tại có thể được tạo ra trong bước S3130. Ví dụ, thiết bị mã hóa hình ảnh có thể tạo ra khối dư (các giá trị của các mẫu dư) bằng cách trừ khối dự đoán (các giá trị của các mẫu được dự đoán) khỏi khối hiện tại (các giá trị của các mẫu gốc). Ví dụ, bằng cách thực hiện bước S3130, thì khối dư trên Fig.29 có thể được tạo ra. Trên khối dư được tạo ra trong bước

S3130, việc lượng tử hóa có thể được thực hiện trong bước S3140 để tạo ra khối dư được lượng tử hóa, và việc dự đoán BDPCM có thể được thực hiện trên khối dư được lượng tử hóa trong bước S3150. Khối dư được lượng tử hóa được tạo ra dưới dạng kết quả của việc thực hiện bước S3140 có thể là khối dư được lượng tử hóa trên Fig.29. Theo kết quả của việc dự đoán BDPCM trong bước S3150, khối dư được lượng tử hóa mà được sửa đổi trên Fig.30 có thể được tạo ra theo hướng dự đoán. Vì việc dự đoán BDPCM trong bước S3150 đã được mô tả với sự tham khảo đến Fig.29 đến Fig.30, phần mô tả chi tiết của nó sẽ được lược bỏ. Sau đây, thiết bị mã hóa hình ảnh có thể mã hóa khối dư được lượng tử hóa mà được sửa đổi trong bước S3160 để tạo ra luồng bit. Ở đây, việc biến đổi của khối dư được lượng tử hóa mà được sửa đổi có thể được bỏ qua.

Hoạt động BDPCM trong thiết bị mã hóa hình ảnh được mô tả với sự tham khảo đến các hình từ Fig.29 đến Fig.31 có thể được thực hiện theo cách đảo ngược bởi thiết bị giải mã hình ảnh.

Fig.32 là lưu đồ minh họa thủ tục để tái tạo khối hiện tại bằng cách áp dụng BDPCM trong thiết bị giải mã hình ảnh theo phương án.

Thiết bị giải mã hình ảnh có thể thu nhận thông tin (thông tin hình ảnh) được đòi hỏi để tái tạo khối hiện tại từ luồng bit trong bước S3210. Thông tin được đòi hỏi để tái tạo khối hiện tại có thể gồm thông tin (thông tin dự đoán) về việc dự đoán của khối hiện tại, và thông tin (thông tin phần dư) về phần dư của khối hiện tại. Thiết bị giải mã hình ảnh có thể thực hiện việc dự đoán trên khối hiện tại trên cơ sở của thông tin về khối hiện tại, và có thể tạo ra khối dự đoán trong bước S3220. Việc dự đoán trên khối hiện tại có thể là việc dự đoán trong ảnh, và phần mô tả chi tiết của nó giống với phần mô tả được mô tả với sự tham khảo đến Fig.31. Trên Fig.32 đã thể hiện được rằng bước S3220 tạo ra khối dự đoán cho khối hiện tại được thực hiện trước các bước S3230 đến S3250 tạo

ra khối dư của khối hiện tại. Tuy nhiên, không có giới hạn nào được đặt ra. Sau khi khối dư của khối hiện tại được tạo ra, thì khối dự đoán của khối hiện tại có thể được tạo ra. Theo cách thay thế, khối dư của khối hiện tại và khối dự đoán của khối hiện tại có thể được tạo ra một cách đồng thời.

Thiết bị giải mã hình ảnh có thể tạo ra khối dư của khối hiện tại trong bước S3230 bằng cách phân tích cú pháp thông tin phần dư của khối hiện tại từ luồng bit. Khối dư được tạo ra trong bước S3230 có thể là khối dư được lượng tử hóa mà được sửa đổi được thể hiện trên Fig.30.

Thiết bị giải mã hình ảnh có thể tạo ra khối dư được lượng tử hóa trên Fig.29 bằng cách thực hiện việc dự đoán BDPCM trong bước S3240 trên khối dư được lượng tử hóa mà được sửa đổi trên Fig.30. Việc dự đoán BDPCM trong bước S3240 là thủ tục để tạo ra khối dư được lượng tử hóa trên Fig.29 từ khối dư được lượng tử hóa mà được sửa đổi trên Fig.30 vốn tương ứng với quy trình ngược trong bước S3150 được thực hiện bởi thiết bị mã hóa hình ảnh. Ví dụ, khi thông tin chế độ mã hóa vi sai (ví dụ, `bdpcm_flag`) được thu nhận từ luồng bit chỉ ra chế độ mã hóa vi sai mà trong đó việc mã hóa vi sai của hệ số dư được thực hiện khi BDPCM được áp dụng, thì thiết bị giải mã hình ảnh thực hiện việc mã hóa vi sai trên khối dư để dẫn xuất khối dư được sửa đổi. Nhờ sử dụng hệ số dư cần được sửa đổi và hệ số dư dự đoán, thì thiết bị giải mã hình ảnh có thể sửa đổi ít nhất một hệ số dư cần được sửa đổi trong số các hệ số dư trong khối dư. Hệ số dư dự đoán có thể được xác định trên cơ sở của hướng dự đoán được chỉ ra bởi thông tin hướng mã hóa vi sai (ví dụ, `bdpcm_dir_flag`) được thu nhận từ luồng bit. Thông tin hướng mã hóa vi sai có thể chỉ ra hoặc hướng dọc hoặc hướng ngang. Thiết bị giải mã hình ảnh có thể chỉ định giá trị được thu nhận bằng cách cộng hệ số dư cần được sửa đổi và hệ số dư dự đoán, vào vị trí của hệ số dư cần được sửa đổi. Ở đây, hệ số dư dự đoán

có thể là hệ số ở ngay sau và liền kề với hệ số dư cần được sửa đổi, liên quan tới thứ tự theo hướng dự đoán.

Ở dưới đây, việc dự đoán BDPCM trong bước S3240 được thực hiện bởi thiết bị giải mã hình ảnh sẽ được mô tả chi tiết dưới đây. Thiết bị giải mã có thể tính toán mẫu dư được lượng tử hóa $Q(r_{i,j})$ bằng cách thực hiện phép tính toán được thực hiện bởi thiết bị mã hóa theo cách đảo ngược. Ví dụ, khi hướng dự đoán của BDPCM là hướng ngang, thì thiết bị giải mã hình ảnh có thể tạo ra khối dư được lượng tử hóa từ khối dư được lượng tử hóa mà được sửa đổi bằng cách sử dụng phương trình 16.

[Phương trình 16]

$$Q(r_{i,j}) = \sum_{k=0}^i r'_{k,j}, \quad 0 \leq i \leq (M-1), \quad 0 \leq j \leq (N-1)$$

Như được định rõ trong phương trình 16, giá trị $Q(r_{i,j})$ của mẫu dư được lượng tử hóa có các tọa độ (i, j) có thể được tính toán bằng cách cộng dồn các giá trị của các mẫu dư được lượng tử hóa mà được sửa đổi bắt đầu từ các tọa độ $(0, j)$ đến các tọa độ (i, j) .

Theo cách thay thế, nhờ sử dụng phương trình 17 thay vì phương trình 16, nên giá trị $Q(r_{i,j})$ của mẫu dư được lượng tử hóa có các tọa độ (i, j) có thể được tính toán.

[Phương trình 17]

$$Q(r_{i,j}) = \begin{cases} r'_{i,j}, & i = 0, \quad 0 \leq j \leq (N-1) \\ r'_{i,j} + Q(r_{(i-1),j}), & 1 \leq i \leq (M-1), \quad 0 \leq j \leq (N-1) \end{cases}$$

Phương trình 17 là quy trình ngược tương ứng với phương trình 14. Theo phương trình 17, giá trị $Q(r_{0,j})$ của mẫu dư được lượng tử hóa có các tọa độ $(0, j)$ được dẫn xuất làm giá trị $r'_{0,j}$ của mẫu dư được lượng tử hóa mà được sửa đổi có các tọa độ $(0, j)$. $Q(r_{i,j})$ có các tọa độ (i, j) khác được dẫn xuất làm tổng của giá trị $r'_{i,j}$ của mẫu dư được lượng tử hóa mà được sửa đổi có các tọa độ (i, j) và giá trị $Q(r_{i-1,j})$ của mẫu dư được lượng tử

hóa có các tọa độ $(i-1, j)$. Nghĩa là, giá trị chênh lệch $r'_{i,j}$ được cộng dồn bằng cách sử dụng giá trị $Q(r_{i-1,j})$ của mẫu dư được lượng tử hóa có các tọa độ $(i-1, j)$ làm giá trị dự đoán, nhờ đó dẫn xuất mẫu dư giá trị được lượng tử hóa $Q(r_{i,j})$.

Khi hướng dự đoán của BDPCM là hướng dọc, thì thiết bị giải mã hình ảnh có thể tạo ra khối dư được lượng tử hóa từ khối dư được lượng tử hóa mà được sửa đổi bằng cách sử dụng phương trình 18.

[Phương trình 18]

$$Q(r_{i,j}) = \sum_{k=0}^j r'_{i,k}, \quad 0 \leq i \leq (M-1), \quad 0 \leq j \leq (N-1)$$

Như được định rõ trong phương trình 18, giá trị $Q(r_{i,j})$ của mẫu dư được lượng tử hóa có các tọa độ (i, j) có thể được tính toán bằng cách cộng dồn các giá trị của các mẫu dư được lượng tử hóa mà được sửa đổi bắt đầu từ các tọa độ $(i, 0)$ đến các tọa độ (i, j) .

Theo cách thay thế, nhờ sử dụng phương trình 19 thay vì phương trình 18, nên giá trị $Q(r_{i,j})$ của mẫu dư được lượng tử hóa có các tọa độ (i, j) có thể được tính toán.

[Phương trình 19]

$$Q(r_{i,j}) = \begin{cases} r'_{i,j}, & 0 \leq i \leq (M-1), \quad j = 0 \\ r'_{i,j} + Q(r_{i,j-1}), & 0 \leq i \leq (M-1), \quad 1 \leq j \leq (N-1) \end{cases}$$

Phương trình 19 là quy trình ngược tương ứng với phương trình 15. Theo phương trình 19, giá trị $Q(r_{i,0})$ của mẫu dư được lượng tử hóa có các tọa độ $(i, 0)$ được dẫn xuất làm giá trị $r'_{0,j}$ của mẫu dư được lượng tử hóa mà được sửa đổi có các tọa độ $(i, 0)$. $Q(r_{i,j})$ có các tọa độ (i, j) khác được dẫn xuất làm tổng của giá trị $r'_{i,j}$ của mẫu dư được lượng tử hóa mà được sửa đổi có các tọa độ (i, j) và giá trị $Q(r_{i,j-1})$ của mẫu dư được lượng tử hóa có các tọa độ $(i, j-1)$. Nghĩa là, giá trị chênh lệch $r'_{i,j}$ được cộng dồn bằng cách sử

dụng giá trị $Q(r_{i,j-1})$ của mẫu dư được lượng tử hóa có các tọa độ $(i, j-1)$ làm giá trị dự đoán, nhờ đó dẫn xuất mẫu dư giá trị được lượng tử hóa $Q(r_{i,j})$.

Khi khối dư được lượng tử hóa gồm có các mẫu dư được lượng tử hóa được tạo ra bằng cách thực hiện bước S3240 theo phương pháp được mô tả ở trên, thì thiết bị giải mã hình ảnh thực hiện việc khử lượng tử hóa trên khối dư được lượng tử hóa trong bước S3250 để tạo ra khối dư của khối hiện tại. Khi BDPCM được áp dụng, thì việc biến đổi của khối hiện tại được bỏ qua như được mô tả ở trên. Do đó, việc biến đổi ngược của khối dư được khử lượng tử hóa có thể được bỏ qua.

Sau đây, thiết bị giải mã hình ảnh có thể tái tạo khối hiện tại trong bước 3260 trên cơ sở của khối dự đoán được tạo ra trong bước S3220 và khối dư được tạo ra trong bước S3250. Ví dụ, thiết bị giải mã hình ảnh có thể tái tạo khối hiện tại (các giá trị của các khối được tái tạo) bằng cách cộng khối dự đoán (các giá trị của các mẫu được dự đoán) và khối dư (các giá trị của các mẫu dư). Ví dụ, giá trị mẫu được tái tạo có thể được tạo ra bằng cách cộng mẫu được lượng tử hóa mà được khử lượng tử hóa $Q^{-1}(Q(r_{i,j}))$ vào giá trị dự đoán khối trong ảnh. Thông tin chế độ mã hóa vi sai chỉ ra xem liệu BDPCM có được áp dụng cho khối hiện tại hay không có thể được phát tín hiệu thông qua luồng bit. Bên cạnh đó, khi BDPCM được áp dụng cho khối hiện tại, thì thông tin hướng mã hóa vi sai chỉ rõ hướng dự đoán của BDPCM có thể được phát tín hiệu thông qua luồng bit. Khi BDPCM không được áp dụng cho khối hiện tại, thì thông tin hướng mã hóa vi sai có thể không được phát tín hiệu.

Các hình vẽ từ Fig.33 đến Fig.35 là các hình thể hiện một cách sơ lược cú pháp để phát tín hiệu thông tin về BDPCM.

Fig.33 là hình thể hiện cú pháp của tập hợp thông số chuỗi theo phương án để phát tín hiệu thông tin BDPCM. Trong phương án, tất cả các RBSP SPS được gồm trong ít

nhất một đơn vị truy cập (Access unit, AU) có giá trị là 0 làm ID theo thời gian (TemporalId) hoặc được cung cấp thông qua các phương tiện ngoài có thể được thiết lập thành được sử dụng trước khi được tham chiếu trong quy trình giải mã. Bên cạnh đó, đơn vị NAL SPS gồm RBSP SPS có thể được thiết lập là có nuh_layer_id mà giống với nuh_layer_id của đơn vị NAL PPS đề cập đến đơn vị NAL SPS. Trong CVS, tất cả các đơn vị NAL SPS có giá trị sps_seq_parameter_set_id cụ thể có thể được thiết lập là có cùng nội dung. Trong cú pháp seq_parameter_set_rbsp() trên Fig.33, sps_transform_skip_enable_flag, được mô tả ở trên, và sps_bdpcm_enabled_flag, được mô tả sau, được bộc lộ.

Phần tử cú pháp sps_bdpcm_enabled_flag có thể chỉ ra xem liệu đối với đơn vị tạo mã trong ảnh, thì intra_bdpcm_flag có được cung cấp trong cú pháp CU hay không. Ví dụ, giá trị thứ nhất (ví dụ, 0) của sps_bdpcm_enabled_flag có thể chỉ ra rằng đối với đơn vị tạo mã trong ảnh, thì intra_bdpcm_flag không được cung cấp trong cú pháp CU. Giá trị thứ hai (ví dụ, 1) của sps_bdpcm_enabled_flag có thể chỉ ra rằng đối với đơn vị tạo mã trong ảnh, thì intra_bdpcm_flag được cung cấp trong cú pháp CU. Trong lúc đó, khi sps_bdpcm_enabled_flag không được cung cấp, thì giá trị của sps_bdpcm_enabled_flag có thể được thiết lập thành giá trị thứ nhất (ví dụ, 0).

Fig.34 là hình thể hiện phương án của cú pháp để phát tín hiệu việc liệu giới hạn trên BDPCM có được áp dụng hay không. Trong phương án, điều kiện giới hạn được xác định trước trong quy trình mã hóa/giải mã có thể được phát tín hiệu nhờ sử dụng cú pháp general_constraint_info(). Nhờ sử dụng cú pháp trên Fig.34, nên phần tử cú pháp no_bdpcm_constraint_flag chỉ ra xem liệu giá trị của sps_bdpcm_enabled_flag được mô tả ở trên có cần được thiết lập thành 0 hay không có thể được phát tín hiệu. Ví dụ, giá trị thứ nhất (ví dụ, 0) của no_bdpcm_constraint_flag có thể chỉ ra rằng giới hạn như vậy

không được áp dụng. Khi giá trị của `no_bdpcm_constraint_flag` là giá trị thứ hai (ví dụ, 1), thì giá trị của `sps_bdpcm_enabled_flag` có thể bị buộc phải là giá trị thứ nhất (ví dụ, 0).

Fig.35 là hình thể hiện phương án của cú pháp đơn vị tạo mã() để phát tín hiệu thông tin về BDPCM cho đơn vị mã hóa. Như được thể hiện trên Fig.35, các phần tử cú pháp `intra_bdpcm_flag` và `intra_bdpcm_dir_flag` có thể được phát tín hiệu nhờ sử dụng cú pháp `coding_unit()`. Phần tử cú pháp `intra_bdpcm_flag` có thể chỉ ra xem liệu BDPCM được áp dụng cho khối mã hóa độ chói hiện tại có được đặt tại (x0, y0) hay không.

Ví dụ, giá trị thứ nhất (ví dụ, 0) của `intra_bdpcm_flag` có thể chỉ ra rằng BDPCM không được áp dụng cho khối mã hóa độ chói hiện tại. Giá trị thứ hai (ví dụ, 1) của `intra_bdpcm_flag` có thể chỉ ra rằng BDPCM được áp dụng cho khối mã hóa độ chói hiện tại. Bằng cách chỉ ra rằng BDPCM được áp dụng, thì `intra_bdpcm_flag` có thể chỉ ra xem liệu việc biến đổi có được bỏ qua hay không và liệu chế độ dự đoán có độ chói được thực hiện bởi `intra_bdpcm_dir_flag` hay không, vốn sẽ được mô tả sau.

Trong lúc đó, giá trị của biến được mô tả ở trên `BdpcmFlag[ x ][ y ]` có thể được thiết lập thành giá trị của `intra_bdpcm_flag`, đối với $x = x0..x0 + cbWidth - 1$ và $y = y0..y0 + cbHeight - 1$.

Phần tử cú pháp `intra_bdpcm_dir_flag` có thể chỉ ra hướng dự đoán của BDPCM. Ví dụ, giá trị thứ nhất (ví dụ, 0) của `intra_bdpcm_dir_flag` có thể chỉ ra rằng hướng dự đoán BDPCM là hướng ngang. Giá trị thứ hai (ví dụ, 1) của `intra_bdpcm_dir_flag` có thể chỉ ra rằng hướng dự đoán BDPCM là hướng dọc.

Trong lúc đó, giá trị của biến `BdpcmDir[ x ][ y ]` có thể được thiết lập thành giá trị của `intra_bdpcm_dir_flag`, đối với $x = x_0 \dots x_0 + \text{cbWidth} - 1$ và $y = y_0 \dots y_0 + \text{cbHeight} - 1$.

Việc dự đoán trong ảnh trên khối sắc độ

Khi việc dự đoán trong ảnh được thực hiện trên khối hiện tại, thì việc dự đoán trên thành phần khối độ chói (khối độ chói) của khối hiện tại và việc dự đoán trên thành phần khối sắc độ (khối sắc độ) có thể được thực hiện. Trong trường hợp này, chế độ dự đoán trong ảnh cho khối sắc độ có thể được thiết lập theo kiểu tách biệt với chế độ dự đoán trong ảnh cho khối độ chói.

Ví dụ, chế độ dự đoán trong ảnh cho khối sắc độ có thể được chỉ ra trên cơ sở của thông tin chế độ dự đoán sắc độ trong ảnh. Thông tin chế độ dự đoán sắc độ trong ảnh có thể được phát tín hiệu trong dạng phần tử cú pháp `intra_chroma_pred_mode`. Ví dụ, thông tin chế độ dự đoán sắc độ trong ảnh có thể chỉ ra một chế độ trong số các chế độ sau đây: chế độ phẳng, chế độ DC, chế độ dọc, chế độ ngang, chế độ được dẫn xuất (Derived Mode, DM), hoặc chế độ mô hình tuyến tính chéo-thành phần (Cross-Component Linear Model, CCLM). Ở đây, chế độ phẳng có thể đề cập đến chế độ dự đoán trong ảnh 0, chế độ DC có thể đề cập đến chế độ dự đoán trong ảnh 1, chế độ dọc có thể đề cập đến chế độ dự đoán trong ảnh 26, và chế độ ngang có thể đề cập đến chế độ dự đoán trong ảnh 10. DM cũng có thể được gọi là chế độ trực tiếp. CCLM có thể also được gọi là mô hình tuyến tính (Linear Model, LM). Chế độ CCLM có thể gồm một thành phần bất kỳ trong số các thành phần sau đây: `L_CCLM`, `T_CCLM`, và `LT_CCLM`.

Trong lúc đó, DM và CCLM là các chế độ dự đoán trong ảnh phụ thuộc để dự đoán khối sắc độ bằng cách sử dụng thông tin của khối độ chói. DM có thể đề cập đến chế độ

mà trong đó cùng các chế độ dự đoán trong ảnh như chế độ dự đoán trong ảnh cho thành phần độ chói được áp dụng làm chế độ dự đoán trong ảnh cho thành phần sắc độ. Bên cạnh đó, CCLM có thể đề cập đến chế độ dự đoán trong ảnh mà trong đó trogn quy trình tạo ra khối dự đoán cho khối sắc độ, thì các mẫu được tái tạo của khối độ chói được lấy mẫu con, và sau đó các mẫu được dẫn xuất bằng cách áp dụng các thông số CCLM α và β cho các mẫu được lấy mẫu con sẽ được sử dụng làm các mẫu dự đoán của khối sắc độ.

Tổng quan của chế độ mô hình tuyến tính chéo-thành phần (CCLM)

Như được mô tả ở trên, chế độ CCLM có thể được áp dụng cho khối sắc độ. Chế độ CCLM là chế độ dự đoán trong ảnh mà trong đó sự tương liên giữa khối độ chói và khối sắc độ tương ứng với khối độ chói được sử dụng, và được thực hiện bằng cách dẫn xuất mô hình tuyến tính trên cơ sở của các mẫu lân cận của khối độ chói và các mẫu lân cận của khối sắc độ. Bên cạnh đó, mẫu dự đoán của khối sắc độ có thể được dẫn xuất trên cơ sở của mô hình tuyến tính được dẫn xuất và các mẫu được tái tạo của khối độ chói.

Một cách cụ thể, khi chế độ CCLM được áp dụng cho khối sắc độ hiện tại, thì các thông số cho mô hình tuyến tính có thể được dẫn xuất trên cơ sở của các mẫu lân cận được sử dụng cho việc dự đoán trong ảnh của khối sắc độ hiện tại và các mẫu lân cận được sử dụng cho việc dự đoán trong ảnh của khối độ chói hiện tại. Ví dụ, mô hình tuyến tính cho CCLM có thể được biểu diễn trên cơ sở của phương trình dưới đây.

[Phương trình 20]

$$pred_c(i,j) = \alpha \cdot rec_L'(i,j) + \beta$$

Ở đây, $pred_c(i,j)$ có thể đề cập đến mẫu dự đoán có các tọa độ (i,j) của khối sắc độ hiện tại trong CU hiện tại. $rec_L'(i,j)$ có thể đề cập đến mẫu được tái tạo có các tọa độ (i,j)

của khối độ chói hiện tại trong CU. Ví dụ, $rec_L'(i,j)$ có thể đề cập đến mẫu được tái tạo mà được lấy mẫu xuống của khối độ chói hiện tại. Các hệ số mô hình tuyến tính α và β có thể được phát tín hiệu, hoặc có thể được dẫn xuất từ mẫu lân cận.

Việc tạo mã liên kết của các phần dư (CbCr liên kết)

Trong quy trình mã hóa/giải mã theo phương án, thì các phần dư sắc độ có thể được mã hóa/được giải mã cùng nhau. Việc này có thể được gọi là việc tạo mã liên kết của các phần dư hoặc là CbCr liên kết. Việc liệu có áp dụng (kích hoạt) chế độ tạo mã liên kết của CbCr có thể được phát tín hiệu bởi cờ phát tín hiệu chế độ tạo mã liên kết `tu_joint_cbc_residual_flag` mà được phát tín hiệu tại mức của cơ sở biến đổi. Bên cạnh đó, chế độ mã hóa được lựa chọn có thể được dẫn xuất bởi các CBF sắc độ. Cờ `tu_joint_cbc_residual_flag` có thể có mặt khi giá trị của ít nhất một CBF sắc độ cho cơ sở biến đổi là 1. Giá trị phần bù QP sắc độ chỉ ra sự chênh lệch giữa giá trị phần bù QP sắc độ chung được phát tín hiệu cho chế độ mã hóa phần dư sắc độ thông thường và giá trị phần bù QP sắc độ cho chế độ tạo mã liên kết CbCr. Giá trị phần bù QP sắc độ có thể được phát tín hiệu thông qua PPS hoặc phần đầu lát. Giá trị phần bù QP này có thể được sử dụng để điều khiển giá trị QP sắc độ đối với các khối nhờ sử dụng chế độ mã hóa phần dư sắc độ liên kết.

Trong trường hợp mà trong đó chế độ 2, vốn là chế độ mã hóa sắc độ liên kết tương ứng, trong bảng dưới đây được kích hoạt cho cơ sở biến đổi, trong khi việc lượng tử hóa và giải mã của cơ sở biến đổi được thực hiện, thì phần bù QP sắc độ của nó có thể được cộng vào QP sắc độ được dẫn xuất độ chói mục tiêu (QP sắc độ được dẫn xuất độ chói được áp dụng).

Liên quan tới các chế độ khác như là các chế độ 1 và 3 trong bảng dưới đây, thì sắc độ QP có thể được dẫn xuất theo cách thức mà nó được thu nhận cho khối Cb hoặc

Cr chung. Quy trình tái tạo các phần dư sắc độ (resCb và resCr) như vậy từ khối biến đổi có thể được lựa chọn theo bảng dưới đây. Khi chế độ này được kích hoạt, thì một khối dư sắc độ liên kết đơn (resJointC[x][y] trong bảng dưới đây) được phát tín hiệu, và khối dư resCb cho Cb và khối dư resCr cho Cr có thể được dẫn xuất khi xem xét đến thông tin như là tu_cbf_cb, tu_cbf_cr, và CSign là giá trị của dấu được bọc lộn trong phần đầu lát.

Trong thiết bị mã hóa, thành phần sắc độ liên kết có thể được dẫn xuất như sau. Theo chế độ tạo mã liên kết, resJointC{1, 2} có thể được tạo ra theo thứ tự sau đây. Trong trường hợp của chế độ 2 (phần dư đơn với việc tái tạo $C_b = C$, $C_r = C_{Sign} * C$), thì phần dư liên kết có thể được xác định theo phương trình sau đây.

[Phương trình 21]

$$\text{resJointC}[x][y] = (\text{resCb}[x][y] + C_{Sign} * \text{resCr}[x][y]) / 2.$$

Theo cách thay thế, trong trường hợp của chế độ 1 (phần dư đơn với việc tái tạo $C_b = C$, $C_r = (C_{Sign} * C) / 2$), thì phần dư liên kết có thể được xác định theo phương trình sau đây.

[Phương trình 22]

$$\text{resJointC}[x][y] = (4 * \text{resCb}[x][y] + 2 * C_{Sign} * \text{resCr}[x][y]) / 5.$$

Theo cách thay thế, trong trường hợp của chế độ 3 (phần dư đơn với việc tái tạo $C_r = C$, $C_b = (C_{Sign} * C) / 2$), thì phần dư liên kết có thể được xác định theo phương trình sau đây.

[Phương trình 23]

$$\text{resJointC}[x][y] = (4 * \text{resCr}[x][y] + 2 * C_{Sign} * \text{resCb}[x][y]) / 5.$$

[Bảng 2]

tu_cbf_c b	tu_cbf_c r	việc tái tạo của các phần dư Cb và Cr	chế độ
1	0	$\begin{aligned} \text{resCb}[x][y] &= \text{resJointC}[x][y] \\ \text{resCr}[x][y] &= (\text{CSign} * \text{resJointC}[x][y]) \gg 1 \end{aligned}$	1
1	1	$\begin{aligned} \text{resCb}[x][y] &= \text{resJointC}[x][y] \\ \text{resCr}[x][y] &= \text{CSign} * \text{resJointC}[x][y] \end{aligned}$	2
0	1	$\begin{aligned} \text{resCb}[x][y] &= (\text{CSign} * \text{resJointC}[x][y]) \gg 1 \\ \text{resCr}[x][y] &= \text{resJointC}[x][y] \end{aligned}$	3

Bảng ở trên thể hiện việc tái tạo của các phần dư sắc độ. CSign đề cập đến giá trị của dấu +1 hoặc -1 được chỉ rõ trong phần đầu lát. resJointC[][] đề cập đến phần dư được truyền. Trong bảng, các chế độ đề cập đến TuCResMode, vốn sẽ được mô tả sau. Ba chế độ mã hóa sắc độ liên kết trong bảng có thể chỉ được hỗ trợ cho lát I. Đối với các lát P và B, thì chỉ có chế độ 2 có thể được hỗ trợ. Do đó, đối với các lát P và B, thì phần tử cú pháp tu_joint_cbr_residual_flag có thể chỉ được cung cấp khi cả hai giá trị cbf sắc độ (ví dụ, tu_cbf_cb và tu_cbf_cr) đều là 1. Trong lúc đó, trong việc tạo mô hình ngưỡng cảnh của tu_cbf_luma và tu_cbf_cb, thì độ sâu biến đổi có thể được loại bỏ.

Phương án 1: Phương pháp cập nhật QP nhờ sử dụng ACT Qp_offset

Như được mô tả ở trên, việc cập nhật của QP để áp dụng ACT có thể được thực hiện. Việc cập nhật được mô tả ở trên của QP có một vài vấn đề. Ví dụ, khi phương pháp được mô tả ở trên được sử dụng, thì không thể thiết lập các phần bù Qp ACT khác nhau cho các thành phần màu sắc riêng lẻ. Hơn nữa, giá trị qP được dẫn xuất có thể có giá trị âm. Theo đó, phương án sau đây mô tả phương pháp áp dụng việc cắt xén cho giá trị Qp được dẫn xuất trên cơ sở của giá trị phần bù QP ACT của giá trị thành phần màu sắc.

Trong phương án, thông số lượng tử hóa qP có thể được dẫn xuất như sau.

Trước tiên, khi $cIdx$ có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 24]

$$qP = Qp'_Y$$

$$ActQpOffset = 5$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 25]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = 5$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 26]

$$qP = Qp'_{cb}$$

$$ActQpOffset = 3$$

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 27]

$$qP = \text{Max}(0, qP - (\text{cu_act_enabled_flag}[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 28]

$$qP = \text{Max}(0, \text{Max}(QpPrimeTsMin, qP) - (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \\ \text{ActQpOffset} : 0))$$

Trong một phương án khác, khi $\text{transform_skip_flag}[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được cắt xén nhờ sử dụng giá trị của $QpPrimeTsMin$ thay vì 0 như được thể hiện trong phương trình dưới đây.

[Phương trình 29]

$$qP = \text{Max}(QpPrimeTsMin, qP - (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \\ \text{ActQpOffset} : 0))$$

Trong lúc đó, trong một phương án khác, thông số lượng tử hóa qP có thể được dẫn xuất như sau.

Trước tiên, khi $cIdx$ có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 30]

$$qP = Qp'_Y$$

$$\text{ActQpOffset} = 5$$

Theo cách thay thế, khi $\text{TuCResMode}[xTbY][yTbY]$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 31]

$$qP = Qp'_{CbCr}$$

$$\text{ActQpOffset} = 5$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 32]

$$qP = Qp'_{cb}$$

$$ActQpOffset = 5$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 33]

$$qP = Qp'_{cr}$$

$$ActQpOffset = 3$$

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 34]

$$qP = \text{Max}(0, qP - (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 35]

$$qP = \text{Max}(0, \text{Max}(QpPrimeTsMin, qP) - (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Trong một phương án khác, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1, thì qP có thể được cắt xén nhờ sử dụng giá trị của $QpPrimeTsMin$ thay vì 0 như được thể hiện trong phương trình dưới đây.

[Phương trình 36]

$$qP = \text{Max}(QpPrimeTsMin, qP - (\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0))$$

Trong lúc đó, trong một phương án khác nữa, thông số lượng tử hóa qP có thể được dẫn xuất như sau.

Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 37]

$$qP = Qp'_Y$$

$$\text{ActQpOffset} = -5$$

Theo cách thay thế, khi $\text{TuCResMode}[\text{xTbY}][\text{yTbY}]$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 38]

$$qP = Qp'_{CbCr}$$

$$\text{ActQpOffset} = -5$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 39]

$$qP = Qp'_{Cb}$$

ActQpOffset = -5

Theo cách thay thế, khi cIdx có giá trị là 2, thì phân bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 40]

$$qP = Qp'_{Cr}$$

ActQpOffset = -3

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 41]

$$qP = \text{Max}(0, qP + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{ActQpOffset} : 0))$$

Theo cách thay thế, khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 42]

$$qP = \text{Max}(0, \text{Max}(QpPrimeTsMin, qP) + (\text{cu_act_enabled_flag}[xTbY][yTbY] ? \text{ActQpOffset} : 0))$$

Trong lúc đó, trong một phương án khác, khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 1, thì qP có thể được cắt xén nhờ sử dụng giá trị của QpPrimeTsMin thay vì 0 như được thể hiện trong phương trình dưới đây.

[Phương trình 43]

$qP = \text{Max}(QpPrimeTsMin, qP + (cu_act_enabled_flag[xTbY][yTbY] ? \text{ActQpOffset} : 0))$

Trong phần mô tả ở trên, Y, Cb, và Cr có thể biểu thị ba thành phần màu sắc. Ví dụ, trong việc biến đổi ACT, Y có thể tương ứng với C0. Cb có thể tương ứng với C1 hoặc Cg. Bên cạnh đó, Cr có thể tương ứng với C2 hoặc Co.

Bên cạnh đó, các giá trị ACTQpOffset là -5, -5, và -3 cho ba thành phần màu sắc có thể được thay thế bằng các giá trị khác hoặc các biến khác.

Phương án 2: việc phát tín hiệu của việc điều chỉnh phần bù QP cho ACT

Trong phương án được mô tả ở trên, các việc điều chỉnh phần bù QP ACT được cố định ở -5, -5, và -3 cho các thành phần Y, Cg, và Co. Trong phương án này, để cung cấp độ linh hoạt hơn cho phần bù điều chỉnh QP ACT, thì phương pháp phát tín hiệu phần bù QP ACT sẽ được mô tả. Phần bù QP ACT có thể được phát tín hiệu dưới dạng thông số trong PPS.

Trong phương án, qp_offset có thể được phát tín hiệu theo bảng cú pháp trên Fig.36. Các phần tử cú pháp của nó là như sau.

Phần tử cú pháp pps_act_qp_offsets_present_flag có thể chỉ ra xem liệu phần tử cú pháp liên quan đến phần bù QP ACT có mặt trong PPS hay không. Ví dụ, pps_act_qp_offsets_present_flag có thể chỉ ra xem liệu các phần tử cú pháp pps_act_y_qp_offset, pps_act_cb_qp_offset, và pps_act_cr_qp_offset, vốn sẽ được mô tả sau, có được phát tín hiệu dưới dạng PPS hay không.

Ví dụ, giá trị thứ nhất (ví dụ, 0) của pps_act_qp_offsets_present_flag có thể chỉ ra rằng pps_act_y_qp_offset, pps_act_cb_qp_offset, và pps_act_cr_qp_offset không được phát tín hiệu thông qua bảng cú pháp PPS.

Giá trị thứ hai (ví dụ, 1) của `pps_act_qp_offsets_present_flag` có thể chỉ ra rằng `pps_act_y_qp_offset`, `pps_act_cb_qp_offset`, và `pps_act_cr_qp_offset` được phát tín hiệu thông qua bảng cú pháp PPS.

Khi `pps_act_qp_offsets_present_flag` không được cung cấp từ luồng bit, thì `pps_act_qp_offsets_present_flag` có thể được dẫn xuất làm giá trị thứ nhất (ví dụ, 0). Ví dụ, khi cờ (ví dụ, `sps_act_enabled_flag` được phát tín hiệu trong SPS) chỉ ra xem liệu ACT có được áp dụng hay không có giá trị thứ nhất (ví dụ, 0) chỉ ra rằng ACT không được áp dụng, thì `pps_act_qp_offsets_present_flag` có thể bị buộc phải có giá trị thứ nhất (ví dụ, 0).

Khi giá trị của phần tử cú pháp `cu_act_enabled_flag` là giá trị thứ hai (ví dụ, 1) chỉ ra rằng ACT được áp dụng cho cơ sở mã hóa hiện tại, thì các phần tử cú pháp `pps_act_y_qp_offset_plus5`, `pps_act_cb_qp_offset_plus5s`, và `pps_act_cr_qp_offset_plus3` có thể được sử dụng để xác định các phần bù được áp dụng cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, và Cr, một cách tương ứng. Khi các giá trị của `pps_act_y_qp_offset_plus5`, `pps_act_cb_qp_offset_plus5`, và `pps_act_cr_qp_offset_plus3` không có mặt trong luồng bit, thì mỗi giá trị có thể được thiết lập thành 0.

Theo các phần tử cú pháp, giá trị của biến `PpsActQpOffsetY` có thể được xác định là `pps_act_y_qp_offset_plus5 - 5`. Giá trị của biến `PpsActQpOffsetCb` có thể được xác định là `pps_act_cb_qp_offset_plus5 - 5`. Bên cạnh đó, giá trị của biến `PpsActQpOffsetCr` có thể được xác định là `pps_act_cr_qp_offset_plus3 - 3`.

Ở đây, ACT không phải là việc biến đổi trực giao, nên 5, 5, và 3 có thể được áp dụng như là các giá trị phần bù không đổi cần được trừ. Trong phương án, để so khớp luồng bit, thì các giá trị của `PpsActQpOffsetY`, `PpsActQpOffsetCb`, và

PpsActQpOffsetCr có thể có các giá trị nằm trong phạm vi từ -12 đến 12. Bên cạnh đó, theo phương án, bên cạnh 5, 5, và 3, thì các giá trị phần bù Qp có thể được thay thế bằng các giá trị không đổi khác và được sử dụng.

Trong một phương án khác, QP có thể được điều chỉnh nhờ sử dụng phần bù ACT_QP linh hoạt hơn. Phương án sau đây mô tả ví dụ mà trong đó phần bù QP ACT được phát tín hiệu trong luồng bit. Theo đó, phần bù QP ACT có thể có phạm vi phần bù rộng hơn. Do đó, QP được cập nhật nhờ sử dụng phần bù QP ACT có nhiều khả năng là nằm ngoài phạm vi khả dụng, nên cần phải thực hiện việc cắt xén trên các giới hạn trên và dưới cho QP được cập nhật (các phương án chi tiết hơn sẽ được mô tả sau trong các phương án 6 và 7)

Các biến PpsActQpOffsetY, PpsActQpOffsetCb, PpsActQpOffsetCr, và PpsActQpOffsetCbCr chỉ ra các phần bù QP ACT có thể là các giá trị được dẫn xuất nhờ sử dụng các phần bù QP ACT được phát tín hiệu thông qua luồng bit, hoặc các hằng số được thiết lập trước. Đối với hiệu suất luồng bit, thì PpsActQpOffsetY, PpsActQpOffsetCb, PpsActQpOffsetCr, và PpsActQpOffsetCbCr có thể có các giá trị nằm trong phạm vi từ -12 đến +12.

Khi không cần sử dụng giá trị cố định, giá trị của phần bù QP được phát tín hiệu và giá trị của nó có giá trị nằm trong phạm vi từ -12 đến 12, thì cần phải cắt xén giá trị giới hạn trên của giá trị QP được dẫn xuất, bên cạnh việc cắt xén giá trị giới hạn dưới của QP giá trị được dẫn xuất để tránh việc QP có giá trị âm.

Để ngăn chặn việc giá trị của qP có giá trị âm, thì giá trị thấp nhất của qP có thể bị buộc là 0. Theo cách thay thế, giá trị thấp nhất của qP có thể được thiết lập thành giá trị được xác định bởi phần tử cú pháp được phát tín hiệu. Ví dụ, để phát tín hiệu giá trị thấp nhất của qP khi chế độ bỏ qua biến đổi được áp dụng, thì phần tử cú pháp QpPrimeTsMin

chỉ ra giá trị của qP được áp dụng khi chế độ bỏ qua biến đổi được áp dụng sẽ có thể được sử dụng. Giá trị tối đa của qP có thể bị giới hạn ở giá trị tối đa khả dụng (ví dụ, 63) của qP hoặc giá trị khả dụng tối đa qP được xác định theo phần tử cú pháp được phát tín hiệu.

Trong phương án theo như ở trên, thông số lượng tử hóa qP có thể được dẫn xuất như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 44]

$$qP = Qp'_Y$$

$$ActQpOffset = PpsActQpOffsetY$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 45]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = PpsActQpOffsetCbCr$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 46]

$$qP = Qp'_{Cb}$$

$$ActQpOffset = PpsActQpOffsetCb$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 47]

$$qP = Qp'_{cr}$$

$$ActQpOffset = PpsActQpOffsetCr$$

Trong phương án, thông số lượng tử hóa qP có thể được cập nhật như sau. Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 48]

$$qP = Clip3(0, 63, qP - (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 49]

$$qP = Clip3(0, 63, Max(QpPrimeTsMin, qP) - (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Trong một phương án khác, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì giá trị thấp nhất của qP có thể được cắt xén nhờ sử dụng giá trị của $QpPrimeTsMin$ thay vì 0 như được thể hiện trong phương trình dưới đây.

[Phương trình 50]

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

$$qP = Clip3(0, 63, qP - (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

$$qP = \text{Clip3}(\text{QpPrimeTsMin}, 63, qP - \text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0)$$

Trong một phương án khác, thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 51]

$$qP = \text{Clip3}(0, 63 + \text{QpBdOffset}, qP + (\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0))$$

Theo cách thay thế, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 52]

$$qP = \text{Clip3}(0, 63 + \text{QpBdOffset}, \text{Max}(\text{QpPrimeTsMin}, qP) + (\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0))$$

Trong một phương án khác, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1, thì giá trị thấp nhất của qP có thể được cắt xén nhờ sử dụng giá trị của QpPrimeTsMin thay vì 0 như được thể hiện trong phương trình dưới đây.

[Phương trình 53]

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

$$qP = \text{Clip3}(0, 63 + QpBdOffset, qP + (\text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0))$$

Theo cách thay thế, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

$$qP = \text{Clip3}(QpPrimeTsMin, 63 + QpBdOffset, qP + \text{cu_act_enabled_flag}[\text{xTbY}][\text{yTbY}] ? \text{ActQpOffset} : 0)$$

Phương án 3: phương pháp để chấp thuận ACT khi sắc độ BDPCM được thực hiện

Trong phương án, khi BDPCM được áp dụng cho thành phần khối độ chói, thì ACT có thể được áp dụng để mã hóa/giải mã khối. Tuy nhiên, khi BDPCM được áp dụng cho thành phần khối sắc độ, thì ACT bị giới hạn là không được áp dụng để mã hóa/giải mã khối.

Trong lúc đó, ngay cả khi BDPCM được áp dụng cho thành phần khối sắc độ, thì ACT được áp dụng cho khối, nhờ đó cải thiện hiệu quả mã hóa. Fig.37 thể hiện phương án của cấu hình cú pháp để áp dụng ACT ngay cả khi BDPCM được áp dụng cho thành phần khối sắc độ. Như được thể hiện trên Fig.37, bằng cách loại bỏ điều kiện để thu nhận phần tử cú pháp BDPCM cho thành phần sắc độ theo giá trị của $\text{cu_act_enabled_flag}$ chỉ ra xem liệu ACT có được áp dụng cho cơ sở mã hóa hiện tại hay không, thì phần tử cú pháp BDPCM của nó có thể được thu nhận bất kể việc liệu ACT có được áp dụng cho thành phần khối sắc độ hay không, và việc mã hóa BDPCM có thể được thực hiện theo đó.

Phương án 4: phương pháp áp dụng ACT ngay cả khi việc mã hóa/giải mã được thực hiện với CCLM

Cả CCLM và ACT đều được nhằm để loại bỏ việc chồng lấn không cần thiết giữa các thành phần. Có một số phần chồng lấn giữa CCLM và ACT, nhưng ngay cả sau khi áp dụng tất cả các thành phần này, thì không thể loại bỏ hoàn toàn việc chồng lấn giữa các thành phần. Do đó, việc chồng lấn giữa các thành phần có thể được loại bỏ nhiều hơn bằng cách áp dụng CCLM và ACT cùng nhau.

Phương án sau đây mô tả phương án mà trong đó CCLM và ACT được áp dụng cùng nhau. Khi thực hiện việc giải mã, thì thiết bị giải mã có thể áp dụng CCLM trước tiên, và áp dụng ACT. Khi ACT được áp dụng cho cả BDPCM và CCLM cho thành phần sắc độ, thì bảng cú pháp để phát tín hiệu ACT có thể được sửa đổi như được thể hiện trên Fig.38. Theo đó, như được thể hiện trong bảng cú pháp trên Fig.38, trong số các giới hạn để phát tín hiệu phần tử cú pháp liên quan đến `intra_bdpcm_chroma` và `cclm`, `if(!cu_act_enabled_flag)` để phát tín hiệu phần tử cú pháp phụ thuộc vào việc liệu ACT có được áp dụng hay không có thể được loại bỏ khỏi bảng cú pháp.

Phương án 5: phương pháp áp dụng Qp ACT linh hoạt gồm CbCr liên kết

Khi chế độ ACT được áp dụng, thì phần dư dự đoán có thể được biến đổi từ một không gian màu sắc (ví dụ, GBR hoặc YCbCr) thành không gian màu sắc YCgCo. Bên cạnh đó, các phần dư của cơ sở biến đổi có thể được mã hóa trong không gian màu sắc YCgCo. Theo phương án của việc biến đổi lỗi ACT (nhân biến đổi) được sử dụng cho việc biến đổi không gian màu sắc, thì nhân biến đổi sau đây như được mô tả ở trên có thể được sử dụng.

[Phương trình 54]

$$\begin{bmatrix} C'_0 \\ C'_1 \\ C'_2 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 2 & -1 & -1 \\ 0 & -2 & 2 \end{bmatrix} \begin{bmatrix} C_0 \\ C_1 \\ C_2 \end{bmatrix} / 4$$

[Phương trình 55]

$$\begin{bmatrix} C_0 \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} C'_0 \\ C'_1 \\ C'_2 \end{bmatrix}$$

Như được mô tả trong các phương trình ở trên, C_0' , các việc biến đổi C_1' , và C_2' (ở đây, $C_0'=Y$, $C_1'=C_g$, $C_2'=C_o$) không được chuẩn hóa. Ví dụ, chuẩn L2 không có giá trị là 1. Ví dụ, chuẩn L2 của việc biến đổi cho thành phần riêng lẻ có thể có giá trị là khoảng 0,6 cho C_0' và C_1' , và có thể có giá trị là khoảng 0,7 cho C_2' . Ở đây, chuẩn L2 là giá trị được thu nhận là căn bậc hai của tổng của bình phương của các hệ số tương ứng. Ví dụ, $C_0' = 2/4*C_0 + 1/4*C_1 + 1/4*C_2$ có thể được tính toán. Do đó, chuẩn của C_0' có thể được tính toán là căn bậc hai của $(2/4*2/4 + 1/4*1/4 + 1/4*1/4)$. Do đó, chuẩn này có thể được tính toán là căn bậc hai của 6/16, và có thể được tính toán là có giá trị là khoảng 0,6.

Khi việc biến đổi được chuẩn hóa không được áp dụng, thì phạm vi động của thành phần riêng lẻ là đều. Bên cạnh đó, điều này gây ra việc giảm hiệu suất mã hóa của hệ thống nén video chung.

Để bù cho phạm vi động của tín hiệu dư, thì phần bù giá trị QP để bù cho sự thay đổi phạm vi động cho thành phần biến đổi riêng lẻ được truyền, để việc điều chỉnh QP có thể được thực hiện. Ví dụ, phương án này có thể được áp dụng cho phương pháp điều khiển việc điều chỉnh QP chung cho việc biến đổi ACT cũng như CbCr liên kết.

Các thành phần màu sắc riêng lẻ không được mã hóa một cách độc lập, nhưng cùng nhau, để phương pháp được mô tả ở trên trong phương án 3 cho CbCr liên kết có thể gây ra sự thay đổi phạm vi động giữa các thành phần màu sắc riêng lẻ.

Trong các phương pháp mã hóa và giải mã theo phương án, việc điều chỉnh phần bù QP ACT có thể được cố định là -5, vốn có thể được áp dụng tương ứng cho Y, Cg, và Co.

Trong phương án, để cung cấp việc điều khiển Qp linh hoạt cho các thành phần riêng lẻ và jointCbCr, thì có thể chấp thuận sử dụng các phần bù Qp ACT khác nhau cho Y, Cb, Cr, và/hoặc CbCr liên kết. Các giá trị phần bù Qp ACT có thể được xác định trên cơ sở của chỉ số thành phần, và/hoặc CbCr liên kết, và/hoặc chế độ CbCr liên kết.

Để chỉ ra các phần bù Qp ACT, thì ppsActQpOffsetY , ppsActQpOffsetCb , và ppsActQpOffsetCr có thể được sử dụng. Bên cạnh đó, $\text{ppsActQpOffsetCbCr}$ có thể được sử dụng cho phần bù QP ACT của chế độ CbCr liên kết 2 có CBF mà trong đó tất cả các thành phần Cb và Cr có các giá trị khác không. Các giá trị này (ví dụ, ppsActQpOffsetY , ppsActQpOffsetCb , ppsActQpOffsetCr , và $\text{ppsActQpOffsetCbCr}$) có thể được xác định trước là các giá trị được xác định trước, hoặc được phát tín hiệu thông qua luồng bit. Phần bù QP ACT của chế độ CbCr liên kết có thể được thiết lập trong một phương pháp khác hoặc thành một giá trị khác.

Trong phương án, các phần bù Qp ACT của -5, -5, và -3 có thể được sử dụng cho Y, Cb, và Cr, và phần bù Qp ACT của -4 có thể được sử dụng cho CbCr liên kết.

Trong một phương án khác, các phần bù Qp ACT của -5, -4, và -3 có thể được sử dụng cho Y, Cb, và Cr, và phần bù Qp ACT của -3 có thể được sử dụng cho chế độ CbCr liên kết mà trong đó giá trị của tu_cbf_cb không phải là 0.

Trong một phương án khác nữa, phần bù QP ACT của chế độ CbCr liên kết 2 có thể có giá trị bù riêng của nó. Đối với một chế độ CbCr liên kết khác, phần bù QP ACT có thể sử dụng phần bù của thành phần tương ứng. Ví dụ, thì thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 56]

$$qP = Qp'_Y$$

$$ActQpOffset = ppsActQpOffsetY$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 57]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = ppsActQpOffsetCbCr$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 58]

$$qP = Qp'_{Cb}$$

$$ActQpOffset = ppsActQpOffsetCb$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 59]

$$qP = Qp'_{Cr}$$

$ActQpOffset = ppsActQpOffsetCr$

Trong phương án, thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 60]

$$qP = Clip3(0, 63 + QpBdOffset, qP + (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 61]

$$qP = Clip3(QpPrimeTsMin, 63 + QpBdOffset, qP + cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0)$$

Trong một phương án khác, đối với chế độ CbCr liên kết (ví dụ, trong trường hợp của các chế độ 1 và 2) với $tu_cbf_cb \neq 0$, thì phần bù cho CbCr liên kết có thể được xác định nhờ sử dụng $ppsActQpOffsetCb$. Theo cách thay thế, đối với chế độ CbCr liên kết (ví dụ, trong trường hợp của chế độ 3) với $tu_cbf_cb = 0$, thì phần bù cho CbCr liên kết có thể được xác định nhờ sử dụng $ppsActQpOffsetCr$. Ví dụ, phương án được mô tả ở trên có thể được sửa đổi và được áp dụng như sau.

Thông số lượng tử hóa qP có thể được cập nhật như sau. Trước tiên, khi $cIdx$ có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 62]

$$qP = Qp'_Y$$

$$ActQpOffset = ppsActQpOffsetY$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 63]

$$qP = Qp'_{CbCr}$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 64]

$$qP = Qp'_{cb}$$

$$ActQpOffset = ppsActQpOffsetCb$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 65]

$$qP = Qp'_{Cr}$$

$$ActQpOffset = ppsActQpOffsetCr$$

Khi $cIdx$ không có giá trị là 0 và $TuCResMode[xTbY][yTbY]$ không có giá trị là 0, thì phần bù Qp ACT cho chế độ $CbCr$ liên kết có thể được xác định theo mã giả dưới đây.

[Phương trình 66]

if ($TuCResMode[xTbY][yTbY]$ is equal to 1 or 2)

ActQpOffset= ppsActQpOffsetCb;

else

ActQpOffset= ppsActQpOffsetCr;

Trong phương án, thông số lượng tử hóa qP có thể được cập nhật như sau. Khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 67]

$$qP = \text{Clip3}(0, 63 + QpBdOffset, qP + (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$$

Theo cách thay thế, khi transform_skip_flag[xTbY][yTbY][cIdx] có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 68]

$$qP = \text{Clip3}(QpPrimeTsMin, 63 + QpBdOffset, qP + cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0)$$

Trong một phương án khác, bất kể chế độ CbCr liên kết, thì ppsActQpOffsetY được sử dụng khi chỉ số thành phần là Y, ppsActQpOffsetCb được sử dụng khi chỉ số thành phần là Cb, và ppsActQpOffsetCr được sử dụng khi chỉ số thành phần là Cr, nhờ vậy qP có thể được dẫn xuất. Ví dụ, thông số lượng tử hóa qP có thể được dẫn xuất như sau.

Trước tiên, khi cIdx có giá trị là 0, thì phân bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 69]

$$qP = Qp'_Y$$

$$ActQpOffset = ppsActQpOffsetY$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 70]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = (cIdx == 1) ? ppsActQpOffsetCb : ppsActQpOffsetCr$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 71]

$$qP = Qp'_{cb}$$

$$ActQpOffset = ppsActQpOffsetCb$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 72]

$$qP = Qp'_{cr}$$

$$ActQpOffset = ppsActQpOffsetCr$$

Thông số lượng tử hóa qP có thể được cập nhật như sau.

Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 73]

$qP = \text{Clip3}(0, 63 + QpBdOffset, qP + (cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0))$

Theo cách thay thế, khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 74]

$qP = \text{Clip3}(QpPrimeTsMin, 63 + QpBdOffset, qP + cu_act_enabled_flag[xTbY][yTbY] ? ActQpOffset : 0)$

Phương án 6: phương pháp phát tín hiệu phân bù Qp ACT gồm CbCr liên kết

Ở dưới đây, sáng chế mô tả ví dụ mà trong đó phân bù QP ACT được phát tín hiệu thông qua luồng bit để cung cấp độ linh hoạt hơn. Các phân bù QP ACT có thể được phát tín hiệu thông qua SPS, PPS, phần đầu ảnh, phần đầu lát, hoặc các tập hợp phần đầu của các loại khác. Phân bù Qp ACT của CbCr liên kết có thể được phát tín hiệu theo kiểu tách biệt, hoặc có thể được dẫn xuất từ các phân bù QP ACT cho Y, Cb, và Cr.

Để không làm mất đi tính tổng quát, thì Fig.39 thể hiện ví dụ về bảng cú pháp mà trong đó phân bù Qp ACT được phát tín hiệu trong PPS. Như trong phương án của Fig.39, một phân bù Qp ACT có thể được phát tín hiệu cho CbCr liên kết. Các phần tử cú pháp được chỉ ra trong bảng cú pháp trên Fig.39 sẽ được mô tả.

Phần tử cú pháp $pps_act_qp_offsets_present_flag$ có thể chỉ ra xem liệu phần tử cú pháp liên quan đến phân bù QP ACT có mặt trong PPS hay không. Ví dụ, $pps_act_qp_offsets_present_flag$ có thể chỉ ra xem liệu các phần tử cú pháp $pps_act_y_qp_offset_plusX1$, $pps_act_cb_qp_offset_plusX2$, $pps_act_cr_qp_offset_plusX3$, và $pps_act_cbcr_qp_offset_plusX4$, vốn sẽ được mô tả sau, có được phát tín hiệu dưới dạng PPS hay không.

Ví dụ, giá trị thứ nhất (ví dụ, 0) của `pps_act_qp_offsets_present_flag` có thể chỉ ra rằng `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, và `pps_act_cbr_qp_offset_plusX4` không được phát tín hiệu thông qua bảng cú pháp PPS.

Giá trị thứ hai (ví dụ, 1) của `pps_act_qp_offsets_present_flag` có thể chỉ ra rằng `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, và `pps_act_cbr_qp_offset_plusX4` được phát tín hiệu thông qua bảng cú pháp PPS.

Khi `pps_act_qp_offsets_present_flag` không được cung cấp từ luồng bit, thì `pps_act_qp_offsets_present_flag` có thể được dẫn xuất làm giá trị thứ nhất (ví dụ, 0). Ví dụ, khi cờ (ví dụ, `sps_act_enabled_flag` được phát tín hiệu trong SPS) chỉ ra xem liệu ACT có được áp dụng hay không có giá trị thứ nhất (ví dụ, 0) chỉ ra rằng ACT không được áp dụng, thì `pps_act_qp_offsets_present_flag` có thể bị buộc phải có giá trị thứ nhất (ví dụ, 0).

Khi giá trị của phần tử cú pháp `cu_act_enabled_flag` là giá trị thứ hai (ví dụ, 1) chỉ ra rằng ACT được áp dụng cho cơ sở mã hóa hiện tại, thì các phần tử cú pháp `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, và `pps_act_cbr_qp_offset_plusX4` có thể được sử dụng để xác định các phần bù được áp dụng cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, Cr, và thành phần CbCr liên kết, một cách tương ứng. Khi các giá trị của `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, và `pps_act_cbr_qp_offset_plusX4` không có mặt trong luồng bit, thì mỗi giá trị có thể được thiết lập thành 0.

Theo các phân tử cú pháp, các giá trị của các biến PpsActQpOffsetY, PpsActQpOffsetCb, PpsActQpOffsetCr, và PpsActQpOffsetCbCr có thể được xác định như được thể hiện trong phương trình dưới đây.

[Phương trình 75]

$$\text{PpsActQpOffsetY} = \text{pps_act_y_qp_offset_plusX1} - X1$$

$$\text{PpsActQpOffsetCb} = \text{pps_act_cb_qp_offset_plusX2} - X2$$

$$\text{PpsActQpOffsetCr} = \text{pps_act_cr_qp_offset_plusX3} - X3$$

$$\text{PpsActQpOffsetCbCr} = \text{pps_act_cbcr_qp_offset_plusX4} - X4$$

Ở đây, X1, X2, X3, và X4 có thể chỉ ra các giá trị không đổi được xác định trước. Các biến này có thể là các giá trị giống nhau hoặc các giá trị khác nhau, hoặc chỉ một số biến có thể có cùng giá trị.

Trong phương án, để so khớp luồng bit, thì các giá trị của PpsActQpOffsetY, PpsActQpOffsetCb, PpsActQpOffsetCr, và PpsActQpOffsetCbCr có thể bị giới hạn là có các giá trị nằm trong phạm vi từ -12 đến 12.

Theo việc xác định của các biến, thì thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 76]

$$qP = Qp'_Y$$

$$\text{ActQpOffset} = \text{PpsActQpOffsetY}$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 77]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = PpsActQpOffsetCbCr$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 78]

$$qP = Qp'_{cb}$$

$$ActQpOffset = PpsActQpOffsetCb$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 79]

$$qP = Qp'_{cr}$$

$$ActQpOffset = PpsActQpOffsetCr$$

Trong một phương án phát tín hiệu phần bù Qp ACT khác, thì, nhiều phần bù QP ACT có thể được phát tín hiệu cho các chế độ CbCr liên kết khác nhau sẽ được nhận dạng như là chế độ A và chế độ B.

Chế độ CbCr liên kết A có thể đề cập đến chế độ jointCbCr có tu_cbf_cb có giá trị khác không, như là chế độ 1 và chế độ 2 của bảng 2 được mô tả ở trên. Chế độ CbCr liên kết B có thể đề cập đến jointCbCrmode có tu_cbf_cb có giá trị là 0, như là chế độ 3 của bảng 2 được mô tả ở trên. Bảng cú pháp được thay đổi theo đó sẽ được thể hiện trên Fig.40. Các phần tử cú pháp được chỉ ra trong bảng cú pháp trên Fig.40 sẽ được mô tả.

Khi giá trị của phần tử cú pháp `cu_act_enabled_flag` là giá trị thứ hai (ví dụ, 1) chỉ ra rằng ACT được áp dụng cho cơ sở mã hóa hiện tại, thì các phần tử cú pháp `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, `pps_act_cbr_qp_offset_modeA_plusX4`, và `pps_act_cbr_qp_offset_modeB_plusX5` có thể được sử dụng để xác định các phần bù được áp dụng cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, Cr, và thành phần CbCr liên kết, một cách tương ứng. Khi các giá trị của `pps_act_y_qp_offset_plusX1`, `pps_act_cb_qp_offset_plusX2`, `pps_act_cr_qp_offset_plusX3`, `pps_act_cbr_qp_offset_modeA_plusX4`, và `pps_act_cbr_qp_offset_modeB_plusX5` không có mặt trong luồng bit, thì mỗi giá trị có thể được thiết lập thành 0.

Theo các phần tử cú pháp, các giá trị của các biến `PpsActQpOffsetY`, `PpsActQpOffsetCb`, `PpsActQpOffsetCr`, `PpsActQpOffsetCbCrModeA`, và `PpsActQpOffsetCbCrModeB` có thể được xác định như được thể hiện trong phương trình dưới đây.

[Phương trình 80]

$$\text{PpsActQpOffsetY} = \text{pps_act_y_qp_offset_plusX1} - X1$$

$$\text{PpsActQpOffsetCb} = \text{pps_act_cb_qp_offset_plusX2} - X2$$

$$\text{PpsActQpOffsetCr} = \text{pps_act_cr_qp_offset_plusX3} - X3$$

$$\text{PpsActQpOffsetCbCrModeA} = \text{pps_act_cbr_qp_offset_modeA_plusX4} - X4$$

$$\text{PpsActQpOffsetCbCrModeB} = \text{pps_act_cbr_qp_offset_modeB_plusX5} - X5$$

Ở đây, X1, X2, X3, X4, và X5 có thể chỉ ra các giá trị không đổi được xác định trước. Các biến này có thể là các giá trị giống nhau hoặc các giá trị khác nhau, hoặc chỉ

một số biến có thể có cùng giá trị. Trong phương án, để so khớp luồng bit, thì các giá trị của $PpsActQpOffsetY$, $PpsActQpOffsetCb$, $PpsActQpOffsetCr$, $PpsActQpOffsetCbCrModeA$, và $PpsActQpOffsetCbCrModeB$ có thể bị giới hạn là có các giá trị nằm trong phạm vi từ -12 đến 12.

Theo việc xác định của các biến, thì thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 81]

$$qP = Qp'_Y$$

$$ActQpOffset = PpsActQpOffsetY$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 82]

$$qP = Qp'_{CbCr}$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 83]

$$qP = Qp'_{Cb}$$

$$ActQpOffset = PpsActQpOffsetCb$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 84]

$$qP = Qp'_{cr}$$

$$ActQpOffset = PpsActQpOffsetCr$$

Bên cạnh đó, khi cIdx không có giá trị là 0 và TuCResMode[xTbY][yTbY] không có giá trị là 0, thì phần bù Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 85]

$$ActQpOffset = (tu_cbf_cb[xTbY][yTbY])? PpsActQpOffsetCbCrModeA: PpsActQpOffsetCbCrModeB$$

Trong lúc đó, trong một phương án khác, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì ActQpOffset có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 86]

$$ActQpOffset = (tu_cbf_cb[xTbY][yTbY])? (PPsQpOffsetCbCrModeA + slice_act_CbCr_qp_offset_ModeA): (PPsQpOffsetCbCrModeB + slice_act_CbCr_qp_offset_ModeB)$$

Trong một phương án phát tín hiệu phần bù Qp ACT khác, thì, thì chỉ có các phần bù QP ACT cho Y, Cb, và Cr có thể được phát tín hiệu như trong bảng cú pháp trên Fig.41. Phần bù QP ACT cho CbCr liên kết có thể được dẫn xuất từ PpsActQpOffsetY, PpsActQpOffsetCb, và/hoặc PpsActQpOffsetCr.

Trong phương án, phần bù Qp ACT cho CbCr có thể được thiết lập thành giá trị của PpsActQpOffsetCb. Trong một phương án khác, phần bù Qp ACT cho CbCr có thể được thiết lập thành cùng giá trị như PpsActQpOffsetCb trong trường hợp của chế độ CbCr liên kết mà trong đó tu_cbf_cb có giá trị khác không, hoặc có thể được thiết lập

thành cùng giá trị như PpsActQpOffsetCr trong trường hợp của chế độ CbCr liên kết mà trong đó tu_cbf_cb có giá trị là 0. Theo cách thay thế, nó có thể được thiết lập ngược lại.

Fig.41 là hình thể hiện một phương án khác của bảng cú pháp mà trong đó phần bù Qp ACT được phát tín hiệu trong PPS. Theo việc xác định của các phần tử cú pháp trên Fig.41, thì thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 87]

$$qP = Qp'_Y$$

$$ActQpOffset = PpsActQpOffsetY$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 88]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = (cIdx==1)? PpsActQpOffsetCb: PpsActQpOffsetCr$$

Trong lúc đó, trong một phương án khác, giá trị của ActQpOffset có thể được xác định như sau.

[Phương trình 89]

$$ActQpOffset = (tu_cbf_cb[xTbY][yTbY])? PpsActQpOffsetCb: PpsActQpOffsetCr$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 90]

$$qP = Qp'_{cb}$$

$$ActQpOffset = PpsActQpOffsetCb$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 91]

$$qP = Qp'_{cr}$$

$$ActQpOffset = PpsActQpOffsetCr$$

Phương án 7: việc phát tín hiệu của phần bù Qp ACT tại nhiều mức

Trong phương án, phần bù QP ACT có thể được phát tín hiệu tại nhiều mức. Bên cạnh việc phát tín hiệu của phần bù QP ACT tại một mức, như là PPS, như trong phương án 6 ở trên, thì phần bù QP ACT có thể được phát tín hiệu tại mức thấp hơn (ví dụ, phần đầu lát, phần đầu ảnh, hoặc các phần đầu của các loại khác phù hợp cho việc điều khiển Qp).

Ở dưới đây, hai phương án sẽ được mô tả. Fig.42 và Fig.43 thể hiện các ví dụ mà trong đó phần bù QP ACT được phát tín hiệu thông qua phần đầu lát và phần đầu ảnh. Theo cách này, phần bù QP ACT có thể được phát tín hiệu tại nhiều mức.

Ở dưới đây, các phần tử cú pháp được thể hiện trên Fig.42 và Fig.43 sẽ được mô tả. Phần tử cú pháp pps_slice_act_qp_offsets_present_flag có thể chỉ ra xem liệu các phần tử cú pháp slice_act_y_qp_offset, slice_act_cb_qp_offset, slice_act_cr_qp_offset, và slice_act_cbr_qp_offset, vốn sẽ được mô tả sau, có mặt trong phần đầu lát hay không.

Ví dụ, giá trị thứ nhất (ví dụ, 0) của `pps_slice_act_qp_offsets_present_flag` có thể chỉ ra rằng `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` không có mặt trong phần đầu lát.

Ví dụ, giá trị thứ hai (ví dụ, 1) của `pps_slice_act_qp_offsets_present_flag` có thể chỉ ra rằng `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` có mặt trong phần đầu lát.

Các phần tử cú pháp `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` có thể chỉ ra các phần bù cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, Cr, và thành phần CbCr liên kết, một cách tương ứng. Các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` có thể bị giới hạn là có các giá trị nằm trong phạm vi từ -12 đến 12. Khi các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` không có mặt trong luồng bit, thì mỗi giá trị có thể được thiết lập thành 0. Các giá trị của $\text{PpsActQpOffsetY} + \text{slice_act_y_qp_offset}$, $\text{PpsActQpOffsetCb} + \text{slice_act_cb_qp_offset}$, $\text{PpsActQpOffsetCr} + \text{slice_act_cr_qp_offset}$, và $\text{PpsActQpOffsetCbCr} + \text{slice_act_cbr_qp_offset}$ có thể bị giới hạn là cũng có các giá trị nằm trong phạm vi từ -12 đến 12.

Các phương án được sửa đổi khác nhau để phát tín hiệu phần bù QP ACT cho CbCr liên kết tại mức PPS có thể được áp dụng. Ví dụ, một phần bù QP có thể được phát tín hiệu cho CbCr liên kết, nhiều phần bù Qp ACT có thể được phát tín hiệu cho CbCr liên kết của các chế độ khác nhau, hoặc không cần phát tín hiệu phần bù Qp ACT cho CbCr liên kết, phương pháp dẫn xuất phần bù Qp ACT bằng cách sử dụng

ACTQpOffsets cho Y, Cb, và Cr và/hoặc chế độ của jointCbCr có thể được áp dụng khi việc phát tín hiệu thông qua phần đầu lát được thực hiện.

Hai phương án được sửa đổi sẽ được thể hiện trên Fig.44 và Fig.45. Fig.44 thể hiện phương án mà trong đó phần bù Qp ACT được phát tín hiệu trong phần đầu lát. Fig.45 thể hiện một phương án khác mà trong đó phần bù Qp ACT được phát tín hiệu trong phần đầu lát. Trên Fig.45, chỉ có các phần bù Qp ACT for Y, Cb, và Cr có thể được phát tín hiệu, và phần bù QP ACT tại mức lát cho CbCr liên kết có thể được dẫn xuất từ slice_act_y_qp_offset, slice_act_cb_qp_offset, và/hoặc slice_act_cr_qp_offset. Việc này có thể được xác định trên cơ sở của loại chế độ của jointCbCr. Trong phương án, phần bù Qp ACT mức lát cho CbCr có thể được thiết lập thành cùng giá trị như slice_act_cb_qp_offset. Trong một phương án khác, trong trường hợp của chế độ CbCr liên kết có tu_cbf_cb có giá trị khác không, thì phần bù Qp ACT tại mức lát cho CbCr liên kết có thể được thiết lập thành cùng giá trị như slice_act_cb_qp_offset. Bên cạnh đó, trong trường hợp của chế độ jointCbCr có tu_cbf_cb có giá trị là 0, thì phần bù Qp ACT tại mức lát cho CbCr liên kết có thể được thiết lập thành cùng giá trị như slice_act_cr_qp_offset.

Trong lúc đó, trong một phương án khác, phần tử cú pháp có thể được phát tín hiệu trong phần đầu lát hoặc phần đầu ảnh. Để hiện thực hóa điều này, thì việc mã hóa/giải mã có thể được thực hiện như sau.

- Cờ pps_picture_slice_act_qp_offsets_present_flag chỉ ra xem liệu phần bù Qp ACT có mặt trong phần đầu ảnh hoặc phần đầu lát hay không có thể được phát tín hiệu trong PPS.

- Khi ACT có thể áp dụng được và giá trị của pps_picture_slice_act_qp_offsets_present_flag là giá trị thứ hai (ví dụ, 1), thì cờ

`pic_act_qp_offsets_present_flag` chỉ ra xem liệu phần bù Qp ACT có mặt trong phần đầu ảnh hay không sẽ được phát tín hiệu trong phần đầu ảnh. Ở đây, giá trị thứ hai (ví dụ, 1) của `pic_act_qp_offsets_present_flag` có thể chỉ ra rằng các phần bù Qp ACT cho tất cả các lát của ảnh tương ứng với phần đầu ảnh sẽ được cung cấp trong phần đầu ảnh.

- Giá trị thứ nhất (ví dụ, 0) của `pic_act_qp_offsets_present_flag` có thể chỉ ra rằng các phần bù Qp ACT cho tất cả các lát của ảnh tương ứng với phần đầu ảnh sẽ không được cung cấp trong phần đầu ảnh. Ví dụ, khi ACT có thể áp dụng được và giá trị của `pps_picture_slice_act_qp_offsets_present_flag` là giá trị thứ hai (ví dụ, 1) và giá trị của `pic_act_qp_offsets_present_flag` là giá trị thứ nhất (ví dụ, 0), phần bù Qp ACT cho lát có thể được cung cấp trong phần đầu lát.

Fig.46 là hình thể hiện bảng cú pháp của PPS mà trong đó `pps_pic_slice_act_qp_offsets_present_flag` được phát tín hiệu. Phần tử cú pháp `pps_pic_slice_act_qp_offsets_present_flag` có thể chỉ ra xem liệu phần bù Qp ACT có được cung cấp trong phần đầu ảnh và/hoặc phần đầu lát hay không. Ví dụ, giá trị thứ nhất (ví dụ, 0) của `pps_pic_slice_act_qp_offsets_present_flag` có thể chỉ ra rằng phần bù Qp ACT không được cung cấp trong phần đầu ảnh và phần đầu lát. Giá trị thứ hai (ví dụ, 1) của `pps_pic_slice_act_qp_offsets_present_flag` có thể chỉ ra rằng phần bù Qp ACT được cung cấp trong phần đầu ảnh hoặc phần đầu lát. Khi `pps_pic_slice_act_qp_offsets_present_flag` không được cung cấp trong luồng bit, thì giá trị của `pps_pic_slice_act_qp_offsets_present_flag` có thể được xác định là giá trị thứ nhất (ví dụ, 0).

Fig.47 là hình thể hiện bảng cú pháp của phần đầu ảnh để phát tín hiệu phần bù Qp ACT. Phần tử cú pháp `pic_act_qp_offsets_present_flag` có thể chỉ ra xem liệu phần bù Qp ACT có được cung cấp trong phần đầu ảnh hay không. Giá trị thứ nhất (ví dụ, 0) của

`pic_act_qp_offsets_present_flag` có thể chỉ ra rằng phần bù Qp ACT không được cung cấp trong phần đầu ảnh, nhưng trong phần đầu lát. Giá trị thứ hai (ví dụ, 1) của `pic_act_qp_offsets_present_flag` có thể chỉ ra rằng phần bù Qp ACT được cung cấp trong phần đầu ảnh. Khi giá trị của `pic_act_qp_offsets_present_flag` không được cung cấp trong luồng bit, thì giá trị có thể được xác định là 0.

Fig.48 là hình thể hiện bảng cú pháp của phần đầu lát để phát tín hiệu phần bù Qp ACT. Trong bảng cú pháp trên Fig.48, các phần tử cú pháp `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` có thể chỉ ra các phần bù cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, và Cr. Các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` có thể có các giá trị nằm trong phạm vi từ -12 đến 12. Bên cạnh đó, các giá trị của $\text{PpsActQpOffsetY} + \text{slice_act_y_qp_offset}$, $\text{PpsActQpOffsetCb} + \text{slice_act_cb_qp_offset}$, và $\text{PpsActQpOffsetCr} + \text{slice_act_cr_qp_offset}$ có thể bị giới hạn là cũng có phạm vi gồm các giá trị từ -12 đến 12.

Trong lúc đó, trong trường hợp mà trong đó các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, `slice_act_cr_qp_offset`, và `slice_act_cbr_qp_offset` không được cung cấp trong luồng bit, khi giá trị của `pps_pic_slice_act_qp_offsets_present_flag` là giá trị thứ nhất (ví dụ, 0), thì các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, và `slice_act_cr_qp_offset` có thể được xác định là 0. Theo cách thay thế, khi giá trị của `pps_pic_slice_act_qp_offsets_present_flag` là giá trị thứ hai (ví dụ, 1), các giá trị của `slice_act_y_qp_offset`, `slice_act_cb_qp_offset`, và `slice_act_cr_qp_offset` có thể được

xác định là cùng các giá trị như `pps_act_y_qp_offset`, `pps_act_cb_qp_offset`, và `pps_act_cr_qp_offset`, một cách tương ứng.

Trong lúc đó, khi phần bù Qp ACT có mặt trong cả phần đầu lát và phần đầu ảnh, thì giá trị phần bù cuối cùng được sử dụng để dẫn xuất giá trị qP có thể được xác định là giá trị mà được thu nhận bằng cách cộng giá trị phần bù được phát tín hiệu trong PPS và giá trị phần bù được phát tín hiệu trong phần đầu lát hoặc phần đầu ảnh.

Một cách cụ thể hơn, trong phương án, thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi `cIdx` có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 92]

$$qP = Qp'_Y$$

$$ActQpOffset = PPsQpOffsetY + slice_act_y_qp_offset$$

Theo cách thay thế, khi `TuCResMode[ xTbY ][ yTbY ]` có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 93]

$$qP = Qp'_{CbCr}$$

$$ActQpOffset = PPsQpOffsetCbCr + slice_act_CbCr_qp_offset$$

Theo cách thay thế, khi `cIdx` có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 94]

$$qP = Qp'_{cb}$$

$$ActQpOffset = PpsActQpOffsetCb + slice_act_Cb_qp_offset$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì phân bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 95]

$$qP = Qp'_{Cr}$$

$$ActQpOffset = PpsActQpOffsetCr + slice_act_Cr_qp_offset$$

Trong một phương án khác, khi nhiều phân bù Qp ACT cho $CbCr$ liên kết được phát tín hiệu, thì $ctQpOffset$ cho $CbCr$ liên kết có thể được xác định như sau.

Trước tiên, khi $cIdx$ có giá trị là 0, thì phân bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 96]

$$qP = Qp'_Y$$

$$ActQpOffset = PPsQpOffsetY + slice_act_y_qp_offset$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 97]

$$qP = Qp'_{CbCr}$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì phân bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 98]

$$qP = Qp'_{Cb}$$

$$ActQpOffset = PpsActQpOffsetCb + slice_act_Cb_qp_offset$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 99]

$$qP = Qp'_{Cr}$$

$$ActQpOffset = PpsActQpOffsetCr + slice_act_Cr_qp_offset$$

Bên cạnh đó, khi $cIdx$ không có giá trị là 0 và $TuCResMode[xTbY][yTbY]$ không có giá trị là 0, thì phần bù Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 100]

$$ActQpOffset = (tu_cbf_cb[xTbY][yTbY])? (PPsQpOffsetCbCrModeA + slice_act_CbCr_qp_offset_ModeA): (PPsQpOffsetCbCrModeB + slice_act_CbCr_qp_offset_ModeB)$$

Trong một phương án khác nữa, khi phần bù Qp ACT cho CbCr liên kết không được cung cấp, thì qP và $ActQpOffset$ cho các thành phần Y, Cb, và/hoặc Cr được xác định và $ActQpOffset$ cho CbCr liên kết có thể được xác định nhờ sử dụng các phần bù Qp ACT của các thành phần Y, Cb, và/hoặc Cr như sau. Ví dụ, trong phương án được mô tả ở trên, khi $TuCResMode[xTbY][yTbY]$ liên quan đến phương trình 97 có giá trị là 2, thì bước tính toán của qP có thể được thay đổi và được thực hiện như sau.

“Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 101]

$$qP = Qp'_{CbCr}$$

$$\text{ActQpOffset} = (\text{cIdx} == 1) ? (\text{PPsQpOffsetCb} + \text{slice_act_Cb_qp_offset}):$$

$$(\text{PPsQpOffsetCr} + \text{slice_act_Cr_qp_offset})$$

Trong lúc đó, trong một phương án khác, giá trị của ActQpOffset có thể được xác định như trong phương trình dưới đây.

[Phương trình 102]

$$\text{ActQpOffset} = (\text{tu_cbf_cb}[\text{xTbY}][\text{yTbY}]) ? (\text{PPsQpOffsetCb} + \text{slice_act_Cb_qp_offset}): (\text{PPsQpOffsetCr} + \text{slice_act_Cr_qp_offset})$$

Phương án 8: phương pháp phát tín hiệu nhiều tập hợp phần bù Qp ACT

Trong phương án này, phương pháp sử dụng danh sách của các phần bù Qp ACT sẽ được mô tả. Để đạt được điều này, thì việc xử lý sau đây có thể được thực hiện.

a) Nhiều tập hợp của các phần bù Qp ACT có thể được phát tín hiệu trong dạng danh sách trong tập hợp thông số (ví dụ, SPS hoặc PPS). Mỗi tập hợp trong danh sách có thể gồm các phần bù Qp ACT cho các thành phần Y, Cb, Cr, và CbCr liên kết. Để đơn giản, danh sách của các phần bù Qp ACT có thể được phát tín hiệu trong tập hợp thông số giống như tập hợp thông số để phát tín hiệu danh sách của các phần bù Qp sắc độ.

b) Số lượng các tập hợp của các phần bù Qp ACT trong danh sách có thể là giống như số lượng các tập hợp phần bù Qp sắc độ được phát tín hiệu trong PPS.

c) Vì phần bù Qp ACT được sử dụng để dẫn xuất qP cho mỗi cơ sở mã hóa, nên phần bù Qp ACT thuộc về danh sách có chỉ số (ví dụ, `cu_chroma_qp_offset_idx`) cho phần bù Qp sắc độ cho cơ sở mã hóa có thể được sử dụng.

d) Theo phương án thay thế của b) và c), thì phần sau đây có thể được thực hiện.

- Số lượng các tập hợp của các phần bù Qp ACT trong danh sách có thể được phát tín hiệu. Số lượng các tập hợp của các phần bù Qp ACT trong danh sách có thể là khác với số lượng các tập hợp phần bù Qp sắc độ.

- Khi ACT có thể áp dụng được, thì chỉ số chỉ ra chỉ số của phần bù Qp ACT được sử dụng cho cơ sở mã hóa có thể được phát tín hiệu.

Không vượt ra khỏi khái niệm ở trên, cú pháp để phát tín hiệu danh sách của các phần bù Qp ACT có thể được sử dụng như được thể hiện trên Fig.49. Ví dụ, khi `cu_act_enabled_flag` có giá trị là 1, thì `pps_act_y_qp_offset`, `pps_act_cb_qp_offset`, `pps_act_cr_qp_offset`, và `pps_act_cbr_qp_offset` có thể được sử dụng để xác định các phần bù cần được áp dụng cho các giá trị thông số lượng tử hóa qP cho các thành phần độ chói, Cb, Cr, và CbCr liên kết, một cách tương ứng.

Khi các giá trị của `pps_act_y_qp_offset`, `pps_act_cb_qp_offset`, `pps_act_cr_qp_offset`, và `pps_act_cbr_qp_offset` không có mặt, thì mỗi giá trị có thể được dẫn xuất thành 0.

Khi giá trị của `cu_act_enabled_flag` là giá trị thứ hai (ví dụ, 1) và giá trị của `cu_chroma_qp_offset_flag` là giá trị thứ hai (ví dụ, 1), thì `act_y_qp_offset_list[i]`, `act_cb_qp_offset_list[i]`, `act_cr_qp_offset_list[i]`, và `act_cbr_qp_offset_list[i]` có thể được sử dụng để xác định các phần bù được áp dụng cho giá trị các thông số lượng tử hóa qP cho các thành phần độ chói, Cb, Cr, và thành phần CbCr liên kết, một cách tương ứng. Khi các giá trị của `act_y_qp_offset_list[i]`, `act_cb_qp_offset_list[i]`, `act_cr_qp_offset_list[i]`, và `act_cbr_qp_offset_list[i]` không có mặt, thì mỗi giá trị có thể được dẫn xuất thành 0.

Trong phương án này, thì thông số lượng tử hóa qP có thể được xác định như sau. Trước tiên, khi cIdx có giá trị là 0, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 103]

$$qP = Qp'_Y$$

$$\text{ActQpOffset} = \text{pps_act_y_qp_offset} + (\text{cu_chroma_qp_offset_flag}) ? \\ \text{act_y_qp_offset_list}[\text{cu_chroma_qp_offset_idx}] : 0 + \text{slice_act_y_qp_offset}$$

Theo cách thay thế, khi TuCResMode[xTbY][yTbY] có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 104]

$$qP = Qp'_{CbCr}$$

$$\text{ActQpOffset} = \text{pps_act_cbr_qp_offset} + (\text{cu_chroma_qp_offset_flag}) ? \\ \text{act_cbr_qp_offset_list}[\text{cu_chroma_qp_offset_idx}] : 0 + \text{slice_act_cbr_qp_offset}$$

Theo cách thay thế, khi cIdx có giá trị là 1, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 105]

$$qP = Qp'_{Cb}$$

$$\text{ActQpOffset} = \text{pps_act_cb_qp_offset} + (\text{cu_chroma_qp_offset_flag}) ? \\ \text{act_cb_qp_offset_list}[\text{cu_chroma_qp_offset_idx}] : 0 + \text{slice_act_cb_qp_offset}$$

Theo cách thay thế, khi cIdx có giá trị là 2, thì phần bù qP và Qp ACT có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 106]

$$qP = Qp'_{cr}$$

$$ActQpOffset = pps_act_cr_qp_offset + (cu_chroma_qp_offset_flag) ? \\ act_cr_qp_offset_list[cu_chroma_qp_offset_idx] : 0 + slice_act_cr_qp_offset$$

Phương án 9: Phương pháp chuyển đổi không gian màu sắc ACT cần được áp dụng cho cả việc mã hóa không hao tổn và mã hóa có hao tổn

Sự biến đổi giữa các không gian màu sắc dựa trên các ma trận cho sự biến đổi xuôi và sự biến đổi ngược được mô tả ở trên có thể được tổ chức như sau.

[Bảng 3]

Chuyển đổi xuôi: GBR thành YCgCo	Chuyển đổi ngược: YCgCo thành GBR
$Y = ((G \ll 1) + R + B + 2) \gg 2$	$t = Y - C_g$
$C_g = ((G \ll 1) - R - B + 2) \gg 2$	$G = Y + C_g$
$Co = ((R - B) \ll 1) + 2 \gg 2$	$B = t - Co = Y - C_g - Co$
	$R = t + Co = Y - C_g + Co$

Sự biến đổi không thể đạt được việc tái tạo thành trạng thái gốc bởi vì sự hao tổn một số giá trị xảy ra trong việc xử lý của Co và Cg. Ví dụ, khi giá trị mẫu của không gian màu sắc RGB được chuyển đổi thành giá trị mẫu của không gian YCgCocolor và giá trị kết quả được chuyển đổi theo kiểu ngược lại thành giá trị mẫu của không gian màu sắc RGB, thì giá trị của mẫu gốc không được tái tạo hoàn toàn. Do đó, sự biến đổi theo bảng 3 không thể được sử dụng cho việc mã hóa không hao tổn. Cần phải cải thiện thuật toán biến đổi không gian màu sắc để sự hao tổn giá trị mẫu không xảy ra sau khi sự biến đổi không gian màu sắc ngay cả khi việc mã hóa không hao tổn được áp dụng. Các phương án 9 và 10 mô tả thuật toán biến đổi không gian màu sắc có khả năng được áp dụng cho việc mã hóa không hao tổn cũng như việc mã hóa có hao tổn.

Trong phương án được mô tả dưới đây, sáng chế mô tả phương pháp thực hiện ACT bằng cách sử dụng sự biến đổi không gian màu sắc có thể đảo ngược được thành trạng thái gốc, vốn có thể áp dụng được cho việc mã hóa có hao tổn cũng như việc mã hóa không hao tổn. Sự biến đổi không gian màu sắc có thể đảo ngược được này có thể được áp dụng cho các phương pháp mã hóa và giải mã được mô tả ở trên. Phần bù Qp ACT cũng có thể được điều chỉnh cho sự biến đổi không gian màu sắc dưới đây. Sự biến đổi không gian màu sắc theo phương án có thể được thực hiện như được thể hiện trong phương trình dưới đây. Ví dụ, sự biến đổi xuôi từ không gian màu sắc GBR thành không gian màu sắc YCgCo có thể được thực hiện theo phương trình sau đây.

[Phương trình 107]

$$Co = R - B;$$

$$t = B + (Co \gg 1);$$

$$Cg = G - t;$$

$$Y = t + (Cg \gg 1);$$

Bên cạnh đó, sự biến đổi ngược từ không gian màu sắc YCgCo thành không gian màu sắc GBR có thể được thực hiện theo phương trình sau đây.

[Phương trình 108]

$$t = Y - (Cg \gg 1)$$

$$G = Cg + t$$

$$B = t - (Co \gg 1)$$

$$R = Co + B$$

Sự biến đổi giữa không gian màu sắc YCgCo và không gian màu sắc RGB theo các phương trình ở trên có thể đảo ngược được thành trạng thái gốc. Nghĩa là, sự biến đổi không gian màu sắc theo các phương trình hỗ trợ hoàn hảo cho việc tái tạo. Ví dụ, giá trị mẫu được duy trì ngay cả khi sự biến đổi ngược được thực hiện sau khi sự biến đổi xuôi. Theo đó, sự biến đổi không gian màu sắc theo các phương trình có thể được gọi là sự biến đổi màu sắc YCgCo-R có thể đảo ngược được. Ở đây, R có thể là viết tắt cho có thể đảo ngược được (reversible), vốn có nghĩa là đạt được việc tái tạo thành trạng thái gốc. Sự biến đổi YCgCo-R có thể được cung cấp bằng cách tăng các độ sâu bit của Cg và Co lên 1 so với sự biến đổi sẵn có. Nếu điều kiện này được thỏa mãn, thì các loại biến đổi có thể đảo ngược được khác có thể được sử dụng giống như sự chuyển đổi được mô tả ở trên.

Bởi vì sự biến đổi như được thể hiện trong các phương trình ở trên giá trị chuẩn khác với sự chuyển đổi được mô tả ở trên, nên các phần bù Qp ACT cho Y, Cg, và Co có thể được điều chỉnh để bù cho sự thay đổi phạm vi động due thành sự biến đổi không gian màu sắc.

Sáng chế đã mô tả rằng khi sự biến đổi được mô tả ở trên được áp dụng, thì phần bù Qp QCT theo phương án có thể có giá trị là (-5, -5, -5) cho Y, Cg, và Co. Tuy nhiên, khi sự biến đổi có thể đảo ngược được theo phương án này được áp dụng, vì phần bù Qp QCT theo phương án, nên một giá trị khác không phải là (-5, -5, -5) có thể được chỉ rõ. Ví dụ, vì phần bù Qp QCT theo phương án, nên giá trị là (-5, 1, 3) cho Y, Cg, và Co có thể được sử dụng.

Trong một phương án khác, phần bù QP ACT có thể được phát tín hiệu thông qua luồng bit như trong phương án 6 hoặc 7 được mô tả ở trên.

Ví dụ, khi sự biến đổi YCgCo-R được mô tả ở trên đã được sử dụng cùng với phần bù QP ACT là (-5, 1, 3), thì sẽ thấy rằng sự hao tổn mã hóa không có mặt trong môi trường mã hóa có hao tổn (ví dụ, QP 22, 27, 32, 37), như được thể hiện trên hình dưới đây. Bên cạnh đó, sẽ thấy rằng khi ACT được áp dụng, thì hiệu suất mã hóa 5 % còn được thu nhận khi hiện thực hóa việc mã hóa không hao tổn.

[Bảng 4]

Chuỗi	Y	U	V
RGB, TGM 1080p	0,0%	0,2%	0,1%
RGB, TGM 720p	0,2%	-0,1%	0,1%
RGB, Hoạt ảnh	-0,1%	-0,1%	0,0%
RGB, Nội dung hỗn hợp	-0,1%	0,0%	-0,1%
RGB, Nội dung được chụp bởi camera	-0,3%	0,2%	-0,3%
Tổng thể (RGB)	0,0%	0,0%	0,0%

Bản đặc tả kỹ thuật VVC để gồm ma trận ACT được tích hợp có thể được mô tả như bảng dưới đây.

[Bảng 5]

Quy trình sửa đổi phần dư cho các khối nhờ sử dụng việc chuyển đổi không gian màu sắc

Các đầu vào cho quy trình này là:

- biến nTbW chỉ rõ chiều rộng khối,
- biến nTbH chỉ rõ chiều cao khối,
- mảng (nTbW)x(nTbH) của các mẫu dư độ chói r_Y với các phần tử $r_Y[x][y]$,
- mảng (nTbW)x(nTbH) của các mẫu dư sắc độ r_{Cb} với các phần tử $r_{Cb}[x][y]$,
- mảng (nTbW)x(nTbH) của các mẫu dư sắc độ r_{Cr} với các phần tử $r_{Cr}[x][y]$.

Các đầu ra của quy trình là:

- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_Y của các mẫu dư độ chói,
- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_{Cb} của các mẫu dư sắc độ,
- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_{Cr} của các mẫu dư sắc độ.

Các mảng $(nTbW) \times (nTbH)$ của các mẫu dư r_Y , r_{Cb} và r_{Cr} được sửa đổi như sau:

$$tmp = r_Y[x][y] - (r_{Cb}[x][y] \gg 1)$$

$$r_Y[x][y] = tmp + r_{Cb}[x][y]$$

$$r_{Cb}[x][y] = tmp - (r_{Cr}[x][y] \gg 1)$$

$$r_{Cr}[x][y] = r_{Cb}[x][y] + r_{Cr}[x][y]$$

Ví dụ, các mảng mẫu dư $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y , r_{Cb} , và r_{Cr} có thể được cập nhật như sau.

[Phương trình 109]

$$tmp = r_Y[x][y] - (r_{Cb}[x][y] \gg 1)$$

$$r_Y[x][y] = tmp + r_{Cb}[x][y]$$

$$r_{Cb}[x][y] = tmp - (r_{Cr}[x][y] \gg 1)$$

$$r_{Cr}[x][y] = r_{Cb}[x][y] + r_{Cr}[x][y]$$

Phương án 10: Phương pháp hiệu suất ACT để thực hiện nhiều sự biến đổi màu sắc dựa trên việc phát tín hiệu rõ ràng

Trong phương án này, ít nhất một sự biến đổi màu sắc có thể được thực hiện bởi ACT. Sự biến đổi màu sắc nào cần được thực hiện có thể được xác định bởi cờ (các cờ) được phát tín hiệu trong luồng bit. Cờ (các cờ) có thể được phát tín hiệu tại nhiều mức, như là SPS, PPS, phần đầu ảnh, và lát, hoặc các chi tiết có thể nhận dạng được.

Trong phương án, cờ được xác định trước có thể được phát tín hiệu để chỉ ra ACT nào được áp dụng. Ví dụ, khi cờ có giá trị là 1, thì ACT dựa trên sự biến đổi màu sắc có thể đảo ngược được có thể được áp dụng. Khi cờ có giá trị là 0, thì ACT dựa trên sự biến đổi màu sắc không thể đảo ngược được có thể được áp dụng.

Trong một phương án khác, cờ được xác định trước cho ACT có thể được phát tín hiệu để chỉ ra sự biến đổi màu sắc nào được sử dụng. Ví dụ về cú pháp được phát tín hiệu trong SPS được thể hiện trên Fig.50. Phần tử cú pháp trên Fig.50 sẽ được mô tả. Phần tử cú pháp `sps_act_reversible_conversion` có thể chỉ ra xem liệu phương trình biến đổi không có thể đảo ngược được thành trạng thái gốc có được sử dụng hay không. Giá trị thứ nhất (ví dụ 0) của `sps_act_reversible_conversion` có thể chỉ ra rằng ACT sử dụng phương trình biến đổi không thể đảo ngược được thành trạng thái gốc. Giá trị thứ hai (ví dụ, 1) của `sps_act_reversible_conversion` có thể chỉ ra rằng ACT sử dụng phương trình biến đổi có thể đảo ngược được thành trạng thái gốc.

Theo đó, biến `lossyCoding` chỉ ra xem liệu việc mã hóa có hao tổn có được thực hiện hay không có thể được thiết lập như trong phương trình dưới đây.

[Phương trình 110]

```
lossyCoding = ( !sps_act_reversible_conversion)
```

Bằng cách sử dụng cờ `lossyCoding`, thì mã giả để thiết bị giải mã thực hiện sự biến đổi ngược từ YCgCo thành GBR trong quy trình giải mã có thể được trình bày như sau.

[Phương trình 111]

```
If (sps_act_reversible_conversion == 1)
```

```
{
```

```
// YCgCo-R reversible conversion
```

```
t = Y - (Cg>>1)
```

```
G = Cg + t
```

```
B = t - (Co>>1)
```

$$R = Co + B$$

}

else {

$$t = Y - Cg$$

$$G = Y + Cg$$

$$B = t - Co$$

$$R = t + Co$$

}

Theo đó, bản đặc tả kỹ thuật VVC được thể hiện trong bảng 5 của phương án 9 có thể được sửa đổi như được thể hiện trong bảng dưới đây.

[Bảng 6]

Quy trình sửa đổi phần dư cho các khối nhờ sử dụng việc chuyển đổi không gian màu sắc

Các đầu vào cho quy trình này là:

- biến $nTbW$ chỉ rõ chiều rộng khối,
- biến $nTbH$ chỉ rõ chiều cao khối,
- mảng $(nTbW) \times (nTbH)$ của các mẫu dư độ chói r_Y với các phần tử $r_Y[x][y]$,
- mảng $(nTbW) \times (nTbH)$ của các mẫu dư sắc độ r_{Cb} với các phần tử $r_{Cb}[x][y]$,
- mảng $(nTbW) \times (nTbH)$ của các mẫu dư sắc độ r_{Cr} với các phần tử $r_{Cr}[x][y]$.

Các đầu ra của quy trình là:

- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_Y của các mẫu dư độ chói,
- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_{Cb} của các mẫu dư sắc độ,
- mảng $(nTbW) \times (nTbH)$ được sửa đổi r_{Cr} của các mẫu dư sắc độ.

Các mảng $(nTbW) \times (nTbH)$ của các mẫu dư r_Y , r_{Cb} và r_{Cr} được sửa đổi như sau:

- Nếu `sps_act_reversible_conversion` bằng 1, thì các mảng $(nTbW) \times (nTbH)$ của các mẫu dư r_Y , r_{Cb} và r_{Cr} được sửa đổi như sau

$$tmp = r_Y[x][y] - (r_{Cb}[x][y] \gg 1) \quad)$$

$$r_Y[x][y] = tmp + r_{Cb}[x][y] \quad)$$

$$r_{Cb}[x][y] = tmp - (r_{Cr}[x][y] \gg 1) \quad)$$

$$r_{Cr}[x][y] = r_{Cb}[x][y] + r_{Cr}[x][y]$$

- Nếu không thì, các mảng $(nTbW) \times (nTbH)$ của các mẫu dư r_Y , r_{Cb} và r_{Cr} được sửa đổi như sau

$$tmp = r_Y[x][y] - r_{Cb}[x][y] \quad)$$

$$r_Y[x][y] = r_Y[x][y] + r_{Cb}[x][y] \quad)$$

$$r_{Cb}[x][y] = tmp - r_{Cr}[x][y] \quad)$$

$$r_{Cr}[x][y] = tmp + r_{Cr}[x][y]$$

Theo bảng ở trên, quy trình cập nhật phần dư nhờ sử dụng sự biến đổi không gian màu sắc có thể sử dụng các thông số sau đây làm các đầu vào của quy trình này.

- biến $nTbW$ chỉ ra khối chiều rộng

- biến $nTbH$ chỉ ra khối chiều cao

- mảng $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y cho mẫu dư độ chói, gồm có các phần tử $r_Y[x][y]$,

- mảng $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_{Cb} cho mẫu sắc độ dư, gồm có các phần tử $r_{Cb}[x][y]$,

- mảng $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_{Cr} cho mẫu sắc độ dư, gồm có các phần tử $r_{Cr}[x][y]$,

Đầu ra của quy trình này là như sau.

- mảng được cập nhật $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y cho mẫu dư độ chói,

- mảng được cập nhật $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_{Cb} cho mẫu sắc độ dư,

- mảng được cập nhật $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_{Cr} cho mẫu sắc độ dư,

Bằng cách thực hiện quy trình này, thì các mảng mẫu dư $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y , r_{Cb} , và r_{Cr} có thể được cập nhật như sau.

Trước tiên, khi giá trị của `sps_act_reversible_conversion` là giá trị thứ hai (ví dụ, 1), thì các mảng mẫu dư $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y , r_{Cb} , và r_{Cr} có thể được cập nhật như được thể hiện trong phương trình dưới đây.

[Phương trình 112]

$$tmp = r_Y[x][y] - (r_{Cb}[x][y] \gg 1) \quad)$$

$$r_Y[x][y] = tmp + r_{Cb}[x][y] \quad)$$

$$r_{Cb}[x][y] = tmp - (r_{Cr}[x][y] \gg 1) \quad)$$

$$r_{Cr}[x][y] = r_{Cb}[x][y] + r_{Cr}[x][y]$$

Nếu không thì (ví dụ, khi giá trị của `sps_act_reversible_conversion` là giá trị thứ nhất (ví dụ, 0)), thì các mảng mẫu dư $(nTbW) \times (nTbH)$ -được tạo kích cỡ r_Y , r_{Cb} , và r_{Cr} có thể được cập nhật như được thể hiện trong phương trình dưới đây.

[Phương trình 113]

$$tmp = r_Y[x][y] - r_{Cb}[x][y]$$

$$r_Y[x][y] = r_Y[x][y] + r_{Cb}[x][y]$$

$$r_{Cb}[x][y] = tmp - r_{Cr}[x][y]$$

$$r_{Cr}[x][y] = tmp + r_{Cr}[x][y]$$

Sự biến đổi ngược YCgCo và sự biến đổi ngược YCgCo-R có một số sự tương tự. Trong việc chuyển đổi có thể đảo ngược được thành trạng thái gốc, khi Cg và Co được

thay thế bằng $Cg'=Cg \ll 1$ và $Co'=Co \ll 1$, thì sự biến đổi này có thể hoạt động như là việc chuyển đổi ngược có hao tổn. Phương trình sau đây thể hiện phương án của nó.

[Phương trình 114]

$$t = Y - (Cg' \gg 1) = Y - Cg$$

$$G = Cg' + t = Y + Cg$$

$$B = t - (Co' \gg 1) = t - Co = Y - Cg - Co$$

$$R = Co' + B = t + Co = Y - Cg + Co$$

Theo đó, trong phương án thay thế, thay vì duy trì hai sự biến đổi màu sắc, thì chỉ có sự biến đổi có thể đảo ngược được thành trạng thái gốc là có thể được sử dụng. Trong trường hợp mã hóa có hao tổn, thì các thành phần Cg và Co có thể được định cỡ 1/2 lần trong hoạt động của thiết bị mã hóa, và có thể được định cỡ gấp đôi trong hoạt động của thiết bị giải mã. Điều này cho phép một sự biến đổi được tích hợp sẽ được sử dụng ngay cả khi các trường hợp có hao tổn và không hao tổn được hỗ trợ. Bên cạnh đó, có hai ưu điểm bổ sung là độ sâu bit không thay đổi ngay cả khi việc mã hóa có hao tổn đang tiến hành.

[Bảng 7]

Chuyển đổi ngược : YCgCo thành GBR
<pre> If (lossyCoding) { Cg = Cg << 1; Co = Co << 1; } t = Y - (Cg >> 1) G = Cg + t B = t - (Co >> 1) </pre>

$$R = Co + B$$

Trong phương án, cờ (ví dụ, actShiftFlag) chỉ ra sự biến đổi ACT nào được sử dụng sẽ có thể được sử dụng theo cú pháp trên Fig.51. Trong bảng cú pháp trên Fig.51, phần tử cú pháp sps_act_shift_flag có thể chỉ ra xem liệu việc thực hiện dịch chuyển thành phần màu sắc có được áp dụng trong khi ACT được áp dụng hay không. Ví dụ, giá trị thứ nhất (ví dụ, 0) của sps_act_shift_flag có thể chỉ ra rằng việc thực hiện dịch chuyển thành phần màu sắc không được áp dụng trong khi ACT được áp dụng. Giá trị thứ hai (ví dụ, 1) của sps_act_shift_flag có thể chỉ ra rằng việc thực hiện dịch chuyển thành phần màu sắc được áp dụng trong khi ACT được áp dụng. Biến actShiftFlag có thể được thiết lập thành giá trị của sps_act_shift_flag. Mã giả để hiện thực hóa sự biến đổi ngược từ YCgCo thành GBR trong thiết bị giải mã có thể được ghi nhớ sử dụng actShiftFlag, như sau.

[Bảng 8]

Chuyển đổi ngược: YCgCo thành GBR
<pre> If (actShiftFlag) { Cg = Cg << 1; Co = Co << 1; } t = Y - (Cg >> 1) G = Cg + t B = t - (Co >> 1) R = Co + B </pre>

Phương án 11: Phương pháp hiệu suất ACT để thực hiện nhiều sự biến đổi màu sắc nhờ sử dụng việc dẫn xuất của loại biến đổi

Trong phương án, ít nhất một sự biến đổi màu sắc có thể được sử dụng khi thực

hiện ACT. Bên cạnh đó, loại biến đổi màu sắc nào được sử dụng sẽ có thể được dẫn xuất trên cơ sở của thông tin khác của luồng bit.

Trong phương án, hai loại biến đổi ACT có thể khả dụng gồm sự biến đổi ACT có thể đảo ngược được thành trạng thái gốc và sự biến đổi ACT không thể đảo ngược được thành trạng thái gốc. Các loại biến đổi ACT có thể được dẫn xuất bởi các loại biến đổi. Ví dụ, như được nhận dạng bởi biến `tuIsTransformSkip`, khi loại biến đổi là bỏ qua biến đổi, thì sự biến đổi ACT có thể đảo ngược được thành trạng thái gốc sẽ có thể được sử dụng. Nếu không thì (ví dụ, khi loại biến đổi không phải là bỏ qua biến đổi), thì sự biến đổi ACT không thể đảo ngược được thành trạng thái gốc sẽ có thể được sử dụng. Hai loại mã giả có thể được sử dụng..

[Bảng 9]

Chuyển đổi ngược: YCgCo thành GBR
<pre> If (tuIsTransformSkip) { // YCgCo-R reversible conversion t = Y - (Cg>>1) G = Cg + t B = t - (Co>>1) R = Co + B } else { t = Y - Cg G = Y + Cg B = t - Co R = t + Co } </pre>

[Bảng 10]

Chuyển đổi ngược: YCgCo thành GBR

If (tuIsTransformSkip)

{

 Cg = Cg << 1;

 Co = Co << 1;

}

t = Y - (Cg >> 1)

G = Cg + t

B = t - (Co >> 1)

R = Co + B

Trong một phương án khác, các loại biến đổi ACT có thể được xác định trên cơ sở của các giá trị QP. Khi giá trị Qp nhỏ hơn hoặc bằng giá trị ngưỡng được xác định trước (ví dụ, QpPrimeTsMin), thì sự biến đổi ACT có thể đảo ngược được thành trạng thái gốc sẽ có thể được sử dụng. Nếu không thì (ví dụ, khi giá trị Qp vượt quá giá trị ngưỡng được xác định trước), thì sự biến đổi ACT không thể đảo ngược được có thể được sử dụng.

Phương án 12: Phương pháp dẫn xuất QP nhờ sử dụng phân bù QP ACT

Phương án này liên quan đến các phương án 1 và 2 được mô tả ở trên. Trong các phương án 1 và 2, sáng chế đã mô tả rằng Qp'_Y, Qp'_{CbCr}, Qp'_{Cb}, và Qp'_{Cr} được dẫn xuất sẽ được gồm dưới dạng các QP. Các phương pháp được mô tả trong các phương án 1 và 2 chỉnh sửa các giá trị Qp được dẫn xuất bằng cách sử dụng phân bù QP ACT, và áp dụng kỹ thuật cắt xén thiết yếu để các giá trị QP được chỉnh sửa cho hệ số biến đổi định cỡ không nằm ngoài phạm vi hiệu quả.

Phương án này mô tả phương pháp gồm phân bù QP ACT trong quy trình dẫn xuất QP để dẫn xuất Qp'_Y, Qp'_{CbCr}, Qp'_{Cb}, và Qp'_{Cr}. Quy trình dẫn xuất QP đã gồm bước cắt

xén được xác định trước để các giá trị QP được dẫn xuất không nằm ngoài phạm vi hiệu quả. Do đó, việc gồm phần bù QP ACT trong quy trình dẫn xuất QP có thể tránh additional cắt xén bước, làm đơn giản hóa bước dẫn xuất QP cho quy trình định cỡ hệ số biến đổi, và đảm bảo rằng QP cuối cùng không nằm ngoài phạm vi hiệu quả.

Như được mô tả trong phương án được mô tả ở trên, thì phần bù QP ACT có thể được chỉ rõ trước là hằng số, hoặc có thể được phát tín hiệu thông qua luồng bit. Không vượt ra khỏi sự nhất quán, các phần bù QP ACT cho Y, Cb, Cr, và CbCr có thể được mô tả là ppsActQpOffsetY , ppsActQpOffsetCb , ppsActQpOffsetCr , và $\text{ppsActQpOffsetCbCr}$ trong phần mô tả mà được mô tả sau. ppsActQpOffsetY , ppsActQpOffsetCb , ppsActQpOffsetCr , và $\text{ppsActQpOffsetCbCr}$ có thể là các hằng số hoặc các biến có các giá trị nằm trong phạm vi từ -M đến N. Ở đây, mỗi thành phần trong số M và N có thể được thiết lập thành 12 trong trường hợp mã hóa có hao tổn và có thể được thiết lập thành 0 trong trường hợp mã hóa không hao tổn trong phương án. Bên cạnh đó, ít nhất một phần bù QP ACT có thể được dẫn xuất từ một giá trị phần bù QP ACT khác. Ví dụ, $\text{ppsActQpOffsetCbCr}$ có thể được thiết lập thành cùng giá trị như ppsActQpOffsetCb hoặc ppsActQpOffsetCr , trên cơ sở của chế độ jointCbCr .

Việc xử lý giải mã cho việc dẫn xuất QP nhờ sử dụng phần bù QP ACT có thể được thực hiện như được mô tả dưới đây. Trước tiên, trong trường hợp của quy trình dẫn xuất thông số lượng tử hóa, thì các thông số sau đây có thể được dùng cho quy trình này.

- Các tọa độ độ chói (x_{Cb} , y_{Cb}) chỉ ra các tọa độ tương đối của mẫu độ chói trên cùng bên trái của khối mã hóa hiện tại đối với mẫu độ chói trên cùng bên trái của ảnh hiện tại,

- Biến `cbWidth` chỉ ra chiều rộng của khối mã hóa hiện tại trên cơ sở từng mẫu độ chói,

- Biến `cbHeight` chỉ ra chiều cao của khối mã hóa hiện tại trên cơ sở từng mẫu độ chói

- Biến `treeType` chỉ ra xem liệu cây đơn (`SINGLE_TREE`) hoặc cây kép có được sử dụng để phân vùng nút cây tạo mã hiện tại hay không, và chỉ ra, khi cây kép được sử dụng, thì liệu cây kép là cây kép thành phần độ chói (`DAUL_TREE_LUMA`) hay cây kép thành phần sắc độ (`DAUL_TREE_CHROMA`)

Trong quy trình dẫn xuất thông số lượng tử hóa này, thì thông số lượng tử hóa độ chói Qp'_Y và các thông số lượng tử hóa sắc độ Qp'_{Cb} , Qp'_{Cr} , và Qp'_{CbCr} có thể được dẫn xuất.

Sau đây, biến Qp_Y có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 115]

$$Qp_Y = ((qP_{Y_PRED} + CuQpDeltaVal + 64 + 2 * QpBdOffset) \% (64 + QpBdOffset)) - QpBdOffset$$

Thông số lượng tử hóa độ chói Qp'_Y có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 116]

$$actQpOffsetY = cu_act_enabled_flag[xCb][yCb] ? ppsActQpOffsetY : 0$$

$$Qp'_Y = Qp_Y + QpBdOffset + actQpOffsetY$$

$$Qp'_Y = \text{Clip3}(0, 63 + QpBdOffset, Qp'_Y)$$

Khi giá trị của biến ChromaArrayType chỉ ra loại mảng sắc độ không phải là giá trị thứ nhất (ví dụ, 0) và treeType là SINGLE_TREE hoặc DUAL_TREE_CHROMA, thì việc xử lý sau đây có thể được thực hiện.

- Khi giá trị của treeType là DUAL_TREE_CHROMA, thì giá trị của biến Qp_Y có thể được thiết lập thành cùng giá trị như giá trị của thông số lượng tử hóa độ chói Qp_Y của cơ sở mã hóa độ chói bao trùm vị trí mẫu độ chói $(x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2)$.

- Các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 117]

$$qP_{Chroma} = \text{Clip3}(-QpBdOffset, 63, Qp_Y) \quad (1118)$$

$$qP_{Cb} = \text{ChromaQpTable}[0][qP_{Chroma}] \quad (1119)$$

$$qP_{Cr} = \text{ChromaQpTable}[1][qP_{Chroma}] \quad (1120)$$

$$qP_{CbCr} = \text{ChromaQpTable}[2][qP_{Chroma}] \quad (1121)$$

- Các thông số lượng tử hóa sắc độ Qp'_{Cb} và Qp'_{Cr} cho các thành phần Cb và Cr và thông số lượng tử hóa sắc độ Qp'_{CbCr} cho việc tạo mã Cb-Cr liên kết có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 118]

$$\text{actQpOffsetCb} = \text{cu_act_enabled_flag}[x_{Cb}][y_{Cb}] ? \text{ppsActQpOffsetCb} : 0$$

$$\text{actQpOffsetCr} = \text{cu_act_enabled_flag}[x_{Cb}][y_{Cb}] ? \text{ppsActQpOffsetCr} : 0$$

$actQpOffsetCbCr = cu_act_enabled_flag[xCb][yCb] ? ppsActQpOffsetCbCr :$
0

$Qp'_{cb} = Clip3(-QpBdOffset, 63, qP_{cb} + pps_cb_qp_offset + slice_cb_qp_offset + CuQpOffset_{cb} + actQpOffsetCb) + QpBdOffset$

$Qp'_{cr} = Clip3(-QpBdOffset, 63, qP_{cr} + pps_cr_qp_offset + slice_cr_qp_offset + CuQpOffset_{cr} + actQpOffsetCr) + QpBdOffset$

$Qp'_{CbCr} = Clip3(-QpBdOffset, 63, qP_{CbCr} + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset + CuQpOffset_{CbCr} + actQpOffsetCbCr) + QpBdOffset$

Tiếp theo, quy trình khử lượng tử hóa cho hệ số biến đổi có thể được thực hiện, và thông tin sau đây có thể được dùng làm đầu vào cho quy trình này.

- Các tọa độ độ chói ($xTbY$, $yTbY$) tham chiếu đến các tọa độ tương đối của mẫu trên cùng bên trái của khối biến đổi độ chói hiện tại đối với mẫu độ chói trên cùng bên trái của ảnh hiện tại,

- Biến $nTbW$ chỉ ra chiều rộng của khối biến đổi,
- Biến $nTbH$ chỉ ra chiều cao của khối biến đổi,
- Biến $predMode$ chỉ ra chế độ dự đoán của cơ sở mã hóa,
- Biến $cIdx$ chỉ ra thành phần màu sắc của khối hiện tại

Đầu ra của quy trình khử lượng tử hóa cho hệ số biến đổi có thể là mảng d của các hệ số biến đổi được định cỡ. Ở đây, kích cỡ của mảng d có thể là $(nTbW) \times (nTbH)$. Các phần tử riêng lẻ cấu thành mảng này có thể được nhận dạng là $d[x][y]$.

Khi thực hiện quy trình này, thì thông số lượng tử hóa qP có thể được dẫn xuất như sau. Khi $cIdx$ có giá trị là 0, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 119]

$$qP = Qp'_Y$$

Theo cách thay thế, khi $TuCResMode[xTbY][yTbY]$ có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 120]

$$qP = Qp'_{CbCr}$$

Theo cách thay thế, khi $cIdx$ có giá trị là 1, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 121]

$$qP = Qp'_{cb}$$

Theo cách thay thế, khi $cIdx$ có giá trị là 2, thì qP có thể được dẫn xuất như được thể hiện trong phương trình dưới đây.

[Phương trình 122]

$$qP = Qp'_{Cr}$$

Thông số lượng tử hóa qP có thể được cập nhật như sau. Bên cạnh đó, các biến $rectNonTsFlag$ và $bdShift$ có thể được dẫn xuất như sau. Khi $transform_skip_flag[xTbY][yTbY][cIdx]$ có giá trị là 0, thì việc dẫn xuất có thể được tiến hành như được thể hiện trong phương trình dưới đây.

[Phương trình 123]

$$\text{rectNonTsFlag} = (((\text{Log2}(\text{nTbW}) + \text{Log2}(\text{nTbH})) \& 1) == 1) ? 1 : 0$$

$$\text{bdShift} = \text{BitDepth} + \text{rectNonTsFlag} + ((\text{Log2}(\text{nTbW}) + \text{Log2}(\text{nTbH})) / 2) - 5 + \text{pic_dep_quant_enabled_flag}$$

Theo cách thay thế, khi $\text{transform_skip_flag}[\text{xTbY}][\text{yTbY}][\text{cIdx}]$ có giá trị là 1 (ví dụ, sự biến đổi của khối biến đổi hiện tại được bỏ qua), thì việc dẫn xuất có thể được tiến hành như được thể hiện trong phương trình dưới đây.

[Phương trình 124]

$$qP = \text{Max}(\text{QpPrimeTsMin}, qP)$$

$$\text{rectNonTsFlag} = 0$$

$$\text{bdShift} = 10$$

Các phương pháp mã hóa và giải mã

Ở dưới đây, phương pháp mã hóa hình ảnh được thực hiện bởi thiết bị mã hóa hình ảnh và phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã hình ảnh sẽ được mô tả với sự tham khảo đến Fig.52 và Fig.53.

Trước tiên, hoạt động của thiết bị giải mã sẽ được mô tả. Thiết bị giải mã hình ảnh theo phương án có thể gồm bộ nhớ và bộ xử lý. Thiết bị giải mã có thể thực hiện giải mã theo hoạt động của bộ xử lý. Ví dụ, như được thể hiện trên Fig.52, thiết bị giải mã có thể xác định khối hiện tại trong bước S5210 bằng cách chia tách hình ảnh trên cơ sở của thông tin chia tách được thu nhận từ luồng bit. Tiếp theo, thông số lượng tử hóa của khối hiện tại có thể được xác định trong bước S5220. Tiếp theo, hệ số biến đổi của khối hiện tại có thể được xác định trên cơ sở của thông số lượng tử hóa trong bước S5230. Ở đây, thông số lượng tử hóa có thể được xác định trên cơ sở của việc liệu sự biến đổi không gian màu sắc có được áp dụng cho mẫu dư của khối hiện tại hay không. Bên cạnh

đó, giá trị của thông số lượng tử hóa có thể được xác định là có giá trị trong phạm vi mà được thiết lập là từ giới hạn trên được xác định trước đến giới hạn dưới được xác định trước. Ví dụ, giá trị của thông số lượng tử hóa có thể được xác định bằng cách thực hiện việc cắt xén trên thông số lượng tử hóa để có giá trị trong phạm vi này. Giá trị giới hạn dưới được xác định trước của thông số lượng tử hóa có thể là 0. Bên cạnh đó, giá trị giới hạn trên được xác định trước của thông số lượng tử hóa có thể được xác định trên cơ sở của hằng số được xác định trước giá trị (ví dụ, 63). Theo cách thay thế, như được mô tả ở trên, có thể xác định được trên cơ sở của thông số (ví dụ, $QpBdOffset$) chỉ ra phần bù phạm vi thông số lượng tử hóa độ chói và sắc độ. $QpBdOffset$ có thể được thiết lập trước thành hằng số được xác định trước hoặc có thể được thu nhận trên cơ sở của phần tử cú pháp được xác định trong quy trình mã hóa. Ví dụ, $QpBdOffset$ có thể được tính toán bằng cách nhân hằng số được xác định trước với giá trị của phần tử cú pháp mà chỉ ra độ sâu bit của mẫu độ chói hoặc sắc độ.

Thông số lượng tử hóa có thể được xác định trên cơ sở của phần bù thông số lượng tử hóa mà được xác định trên cơ sở của thành phần màu sắc của khối hiện tại. Ví dụ, thông số lượng tử hóa có thể được xác định bằng cách cộng phần bù thông số lượng tử hóa vào thông số lượng tử hóa mà được xác định trên cơ sở của thành phần màu sắc của khối hiện tại. Khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại là thành phần độ chói, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa có thể được xác định là 5. Bên cạnh đó, khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại là thành phần Cb sắc độ, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa có thể được xác định là 1. Bên cạnh đó, khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại

là thành phần Cr sắc độ, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa có thể được xác định là 3.

Bên cạnh đó, mẫu dư của khối hiện tại có thể được xác định trên cơ sở của hệ số biến đổi của khối hiện tại. Ở đây, giá trị của mẫu dư có thể được cập nhật trên cơ sở của liệu sự biến đổi không gian màu sắc được áp dụng. Ví dụ, khi sự biến đổi không gian màu sắc được áp dụng, thì giá trị của mẫu dư có thể được cập nhật bằng cách thực hiện sự biến đổi không gian màu sắc của mẫu dư với hàm biến đổi không gian màu sắc có thể đảo ngược được thành giá trị của mẫu gốc. Ví dụ, khi sự biến đổi không gian màu sắc được áp dụng, thì giá trị của mẫu dư có thể được cập nhật trên cơ sở của giá trị mẫu mà được thu nhận bằng cách áp dụng phép toán dịch chuyển sang phải cho thành phần giá trị mẫu dư sắc độ.

Ở dưới đây, hoạt động của thiết bị mã hóa sẽ được mô tả. Thiết bị mã hóa hình ảnh theo phương án có thể gồm bộ nhớ và bộ xử lý. Thiết bị mã hóa có thể thực hiện việc mã hóa theo hoạt động của bộ xử lý theo cách thức tương ứng với việc giải mã bởi thiết bị giải mã. Ví dụ, như được thể hiện trên Fig.53, thiết bị mã hóa có thể xác định khối hiện tại bằng cách chia tách hình ảnh trong bước S5310. Tiếp theo, thiết bị mã hóa có thể xác định thông số lượng tử hóa của khối hiện tại trong bước S5320. Tiếp theo, thiết bị mã hóa có thể xác định hệ số biến đổi của khối hiện tại trên cơ sở của thông số lượng tử hóa trong bước S5330. Ở đây, bước xác định của thông số lượng tử hóa có thể được thực hiện trên cơ sở của việc liệu sự biến đổi không gian màu sắc có được áp dụng cho mẫu dư của khối hiện tại hay không. Bên cạnh đó, giá trị của thông số lượng tử hóa có thể bị giới hạn ở có giá trị trong phạm vi mà được thiết lập là từ giới hạn trên được xác định trước đến giới hạn dưới được xác định trước.

Ở đây, thiết bị mã hóa có thể áp dụng việc cắt xén cho thông số lượng tử hóa sao cho giá trị của thông số lượng tử hóa có giá trị nằm trong phạm vi được xác định trước. Ở đây, giá trị giới hạn dưới của phạm vi được xác định trước có thể là 0. Bên cạnh đó, giá trị giới hạn trên của phạm vi được xác định trước có thể là giá trị giới hạn trên của độ chói và thông số lượng tử hóa sắc độ. Giá trị giới hạn trên của độ chói và thông số lượng tử hóa sắc độ có thể được xác định trên cơ sở của độ sâu bit của mẫu.

Trong lúc đó, để xác định thông số lượng tử hóa, thì thiết bị mã hóa có thể xác định thông số lượng tử hóa trên cơ sở của thành phần màu sắc của khối hiện tại, và có thể xác định phần bù thông số lượng tử hóa trên cơ sở của thành phần màu sắc của khối hiện tại. Bên cạnh đó, thiết bị mã hóa có thể thiết lập lại thông số lượng tử hóa bằng cách sử dụng phần bù thông số lượng tử hóa, nhờ đó xác định thông số lượng tử hóa.

Ở đây, thiết bị mã hóa có thể cộng phần bù thông số lượng tử hóa vào thông số lượng tử hóa, nhờ đó thiết lập lại thông số lượng tử hóa.

Ở đây, khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại là thành phần độ chói, thì giá trị của phần bù thông số lượng tử hóa có thể được xác định là -5. Bên cạnh đó, khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại là thành phần Cb sắc độ, thì giá trị của phần bù thông số lượng tử hóa có thể được xác định là 1. Bên cạnh đó, khi sự biến đổi không gian màu sắc được áp dụng cho mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại là thành phần Cr sắc độ, thì giá trị của phần bù thông số lượng tử hóa có thể được xác định là 3.

Trong lúc đó, mẫu dư của khối hiện tại được xác định trên cơ sở của hệ số biến đổi của khối hiện tại, và giá trị của mẫu dư có thể được cập nhật trên cơ sở của liệu sự biến đổi không gian màu sắc được áp dụng. Khi sự biến đổi không gian màu sắc được áp

dụng, thì giá trị của mẫu dư có thể được cập nhật trên cơ sở của giá trị mẫu mà được thu nhận bằng cách áp dụng phép toán dịch chuyển sang phải cho thành phần giá trị mẫu dư sắc độ.

Phương án áp dụng

Dù các phương pháp làm ví dụ của sáng chế này được mô tả ở trên được biểu diễn như là một loạt các hoạt động để tạo sự rõ ràng cho việc mô tả, nhưng nó không nhằm mục đích hạn chế thứ tự mà trong đó các bước được thực hiện, và các bước có thể được thực hiện đồng thời hoặc theo thứ tự khác nếu cần thiết. Để triển khai phương pháp trong sáng chế này, thì các bước được mô tả ở trên có thể còn gồm các bước khác, có thể gồm các bước còn lại ngoại trừ một số bước trong số các bước, hoặc có thể gồm các bước bổ sung ngoại trừ một số bước.

Trong sáng chế này, thiết bị mã hóa hình ảnh hoặc thiết bị giải mã hình ảnh thực hiện hoạt động (bước) được xác định trước có thể thực hiện hoạt động (bước) xác nhận điều kiện thực thi hoặc tình huống của hoạt động (bước) tương ứng. Ví dụ, nếu nó xác định được rằng hoạt động được xác định trước được thực hiện khi điều kiện được xác định trước được thỏa mãn, thì thiết bị mã hóa hình ảnh hoặc thiết bị giải mã hình ảnh có thể thực hiện hoạt động được xác định trước sau khi xác định xem liệu điều kiện được xác định trước có được thỏa mãn hay không.

Các phương án khác nhau của sáng chế này không liệt kê tất cả các tổ hợp khả thi và nhằm để mô tả các khía cạnh biểu diễn của sáng chế này, và các vấn đề được mô tả trong các phương án khác nhau có thể được áp dụng một cách độc lập hoặc trong kết hợp của hai hoặc nhiều hơn hai thành phần.

Các phương án khác nhau của sáng chế này có thể được triển khai trong phần cứng, phần sụn, phần mềm, hoặc tổ hợp của chúng. Trong trường hợp triển khai sáng chế này

bởi phần cứng, thì sáng chế này có thể được triển khai với các mạch tích hợp ứng dụng cụ thể (Application Specific Integrated Circuit, ASIC), các bộ xử lý tín hiệu số (Digital Signal Processor, DSP), các thiết bị xử lý tín hiệu số (Digital Signal Processing Device, DSPD), các thiết bị logic lập trình được (Programmable Logic Device, PLD), các mảng cổng lập trình được bằng trường (Field Programmable Gate Array, FPGA), các bộ xử lý chung, các bộ điều khiển, các bộ vi điều khiển, các bộ vi xử lý, v.v.

Bên cạnh đó, thiết bị giải mã hình ảnh và thiết bị mã hóa hình ảnh mà phần bộc lộ này được áp dụng vào đó có thể được gồm trong thiết bị truyền và nhận phát rộng đa phương tiện, thiết bị đầu cuối truyền thông di động, thiết bị video chiếu phim tại nhà, thiết bị video chiếu phim kỹ thuật số, camera theo dõi, thiết bị thoại video, thiết bị liên lạc theo thời gian thực như liên lạc video, thiết bị phát luồng di động, thiết bị lưu trữ, máy quay video, thiết bị cung cấp dịch vụ video theo yêu cầu (Video on Demand, VoD), thiết bị video trên nền mạng viễn thông (Over The Top, OTT), thiết bị cung cấp dịch vụ phát luồng Internet, thiết bị video ba chiều (Three-Dimensional, 3D), thiết bị thực tế ảo, thiết bị thực tế tăng cường (augmented reality), thiết bị gọi điện thoại video, thiết bị đầu cuối của các phương tiện vận tải (ví dụ, thiết bị đầu cuối của xe (gồm xe ô tô), thiết bị đầu cuối của máy bay, và thiết bị đầu cuối của tàu thủy, v.v.), thiết bị video y tế, và tương tự, và có thể được sử dụng để xử lý tín hiệu video hoặc tín hiệu dữ liệu. Ví dụ, các thiết bị video OTT có thể gồm bộ điều khiển trò chơi, thiết bị phát Blu-ray, TV truy cập Internet, hệ thống rạp hát tại gia, điện thoại thông minh, PC bảng, và thiết bị ghi video kỹ thuật số (Digital Video Recorder, DVR), hoặc tương tự.

Fig.54 là hình thể hiện hệ thống phát luồng nội dung, mà phương án của sáng chế có thể áp dụng được vào đó.

Như được thể hiện trên Fig.54, hệ thống phát luồng nội dung mà phương án (các phương án) của sáng chế này được áp dụng với, có thể phần lớn gồm máy chủ mã hóa, máy chủ phát luồng, máy chủ web, bộ phận lưu trữ phương tiện, thiết bị người dùng, và thiết bị nhập vào đa phương tiện.

Máy chủ mã hóa nén các nội dung được nhập vào từ các thiết bị đầu vào đa phương tiện như điện thoại thông minh, camera, máy quay video, v.v. thành dữ liệu kỹ thuật số để tạo ra luồng bit và truyền luồng bit tới máy chủ phát luồng. Theo một ví dụ khác, khi thiết bị đầu vào đa phương tiện như là điện thoại thông minh, camera, máy ghi hình cầm tay, v.v. trực tiếp tạo ra luồng bit, thì máy chủ mã hóa có thể được lược bỏ.

Luồng bit có thể được tạo ra bởi phương pháp mã hóa hình ảnh hoặc thiết bị mã hóa hình ảnh mà phương án của sáng chế này được áp dụng vào đó, và máy chủ phát luồng có thể tạm thời lưu trữ luồng bit trong quy trình truyền hoặc nhận luồng bit.

Máy chủ phát luồng truyền dữ liệu đa phương tiện đến thiết bị người dùng dựa trên yêu cầu người dùng thông qua máy chủ web, và máy chủ web đóng vai trò là phương tiện để thông báo cho người dùng về các dịch vụ. Khi người dùng yêu cầu dịch vụ mong muốn từ máy chủ web, thì máy chủ web có thể phân phối nó đến máy chủ phát luồng, và máy chủ phát luồng có thể truyền dữ liệu đa phương tiện đến người dùng. Trong trường hợp này, hệ thống phát luồng nội dung có thể gồm máy chủ điều khiển tách biệt. Trong trường hợp này, máy chủ điều khiển có vai trò điều khiển lệnh/hỏi đáp giữa các thiết bị trong hệ thống phát luồng nội dung.

Máy chủ phát luồng có thể nhận nội dung từ bộ phận lưu trữ phương tiện và/hoặc máy chủ mã hóa. Ví dụ, khi nội dung được nhận từ máy chủ mã hóa, thì nội dung có thể được nhận theo thời gian thực. Trong trường hợp này, để cung cấp dịch vụ phát luồng

tron tru, thì máy chủ phát luồng có thể lưu trữ luồng bit trong thời gian được xác định trước.

Các ví dụ của 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, thiết bị trợ giúp số cá nhân (Personal Digital Assistant, PDA), thiết bị chơi đa phương tiện di động (Portable Multimedia Player, PMP), thiết bị định vị, PC dạng phiên, các PC dạng bảng, các máy tính siêu di động, các thiết bị đeo được (ví dụ, các đồng hồ thông minh, các kính thông minh, các thiết bị hiển thị gắn trên đầu), các tivi kỹ thuật số, các máy tính để bàn, bảng ký hiệu kỹ thuật số, và tương tự.

Mỗi máy chủ trong hệ thống phát luồng nội dung có thể được hoạt động như là máy chủ phân tán, mà trong trường hợp đó dữ liệu nhận được từ mỗi máy chủ có thể được phân tán.

Phạm vi của sáng chế này gồm phần mềm hoặc các lệnh thực thi được bởi máy (ví dụ, hệ điều hành, ứng dụng, phần sụn, chương trình, v.v.) để cho phép thực hiện các hoạt động theo các phương pháp của các phương án khác nhau cần được thực thi trên thiết bị hoặc máy tính, phương tiện lưu trữ phi chuyển tiếp đọc được bởi máy tính có phần mềm hoặc các lệnh này được lưu trên đó và thực thi được trên thiết bị hoặc máy tính.

Khả năng áp dụng trong công nghiệp

Các phương án của sáng chế này có thể được sử dụng để mã hóa hoặc giải mã hình ảnh.

YÊU CẦU BẢO HỘ

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

xác định thông số lượng tử hóa của khối hiện tại; và

dẫn xuất các hệ số biến đổi của khối hiện tại dựa trên thông số lượng tử hóa;

trong đó thông số lượng tử hóa được xác định dựa trên phần bù thông số lượng tử hóa,

trong đó phần bù thông số lượng tử hóa được dẫn xuất dựa trên sự biến đổi không gian màu sắc cho các mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại, và

trong đó giá trị của thông số lượng tử hóa được cắt xén để lớn hơn hoặc bằng giá trị cụ thể, giá trị cụ thể được thiết lập dựa trên cờ bỏ qua biến đổi của khối hiện tại, và

trong đó cờ bỏ qua biến đổi được thu nhận dựa trên thành phần màu sắc của khối hiện tại.

2. Phương pháp theo điểm 1,

trong đó giá trị của thông số lượng tử hóa được cắt xén để nhỏ hơn hoặc bằng giá trị trên được xác định trước.

3. Phương pháp theo điểm 1,

trong đó cờ bỏ qua biến đổi là cho sự biến đổi từ miền tần số thành miền không gian.

4. Phương pháp theo điểm 3,

trong đó dựa trên cờ bỏ qua biến đổi của khối hiện tại chỉ ra sự biến đổi từ miền tần số thành miền không gian được xác định dựa trên một phần tử cú pháp khác, thì giá trị dưới được xác định trước được thiết lập là 0.

5. Phương pháp theo điểm 3,

trong đó dựa trên cờ bỏ qua biến đổi của khối hiện tại chỉ ra sự biến đổi từ miền tần số thành miền không gian được bỏ qua, thì giá trị dưới được xác định trước được thiết lập dựa trên giá trị tối thiểu của thông số lượng tử hóa cho chế độ bỏ qua biến đổi.

6. Phương pháp theo điểm 1,

trong đó dựa trên thành phần màu sắc của khối hiện tại là thành phần Cr sắc độ và việc biến đổi không gian màu sắc cho các mẫu dư của khối hiện tại được áp dụng, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa được thiết lập là 3.

7. Phương pháp theo điểm 1,

trong đó dựa trên thành phần màu sắc của khối hiện tại là thành phần độ chói và việc biến đổi không gian màu sắc cho các mẫu dư của khối hiện tại được áp dụng, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa được thiết lập là 5.

8. Phương pháp theo điểm 1,

trong đó dựa trên việc biến đổi không gian màu sắc cho các mẫu dư của khối hiện tại được áp dụng, thì việc trừ dựa trên giá trị của phần bù thông số lượng tử hóa dựa trên thành phần màu sắc của khối hiện tại là thành phần Cr sắc độ và giá trị của phần bù thông số lượng tử hóa dựa trên thành phần màu sắc của khối hiện tại là thành phần Cb sắc độ là 2.

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

xác định thông số lượng tử hóa của khối hiện tại; và
mã hóa hệ số biến đổi của khối hiện tại dựa trên thông số lượng tử hóa;
trong đó thông số lượng tử hóa được xác định dựa trên phần bù thông số lượng tử
hóa,

trong đó phần bù thông số lượng tử hóa được dẫn xuất dựa trên sự biến đổi không
gian màu sắc cho các mẫu dư của khối hiện tại và thành phần màu sắc của khối hiện tại,
và

trong đó giá trị của thông số lượng tử hóa được cắt xén để lớn hơn hoặc bằng giá
trị cụ thể, giá trị cụ thể được thiết lập dựa trên sự biến đổi cho khối hiện tại, và

trong đó sự biến đổi được xác định dựa trên thành phần màu sắc của khối hiện tại.

10. Phương pháp theo điểm 9,

trong đó giá trị của thông số lượng tử hóa được cắt xén để nhỏ hơn hoặc bằng giá
trị trên được xác định trước.

11. Phương pháp theo điểm 9,

trong đó sự biến đổi là từ miền không gian thành miền tần số và được mã hóa như
cờ bỏ qua biến đổi.

12. Phương pháp theo điểm 11,

trong đó giá trị dưới được xác định trước được thiết lập là 0 hoặc giá trị tối thiểu
của thông số lượng tử hóa cho chế độ bỏ qua biến đổi.

13. Phương pháp theo điểm 9,

trong đó dựa trên thành phần màu sắc của khối hiện tại là thành phần Cr sắc độ và việc biến đổi không gian màu sắc cho các mẫu dư của khối hiện tại được áp dụng, thì giá trị tuyệt đối của phần bù thông số lượng tử hóa được thiết lập là 3.

14. Phương pháp truyền luồng bit được tạo ra bởi phương pháp mã hóa hình ảnh theo điểm 9.

15. Phương tiện ghi đọc được bởi máy tính không chuyển tiếp lưu trữ luồng bit được tạo ra bởi phương pháp mã hóa hình ảnh theo điểm 9.

FIG. 1

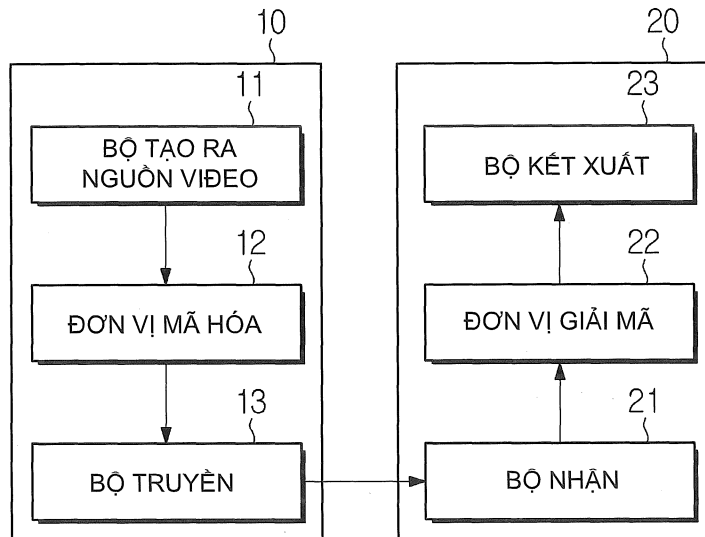


FIG. 2

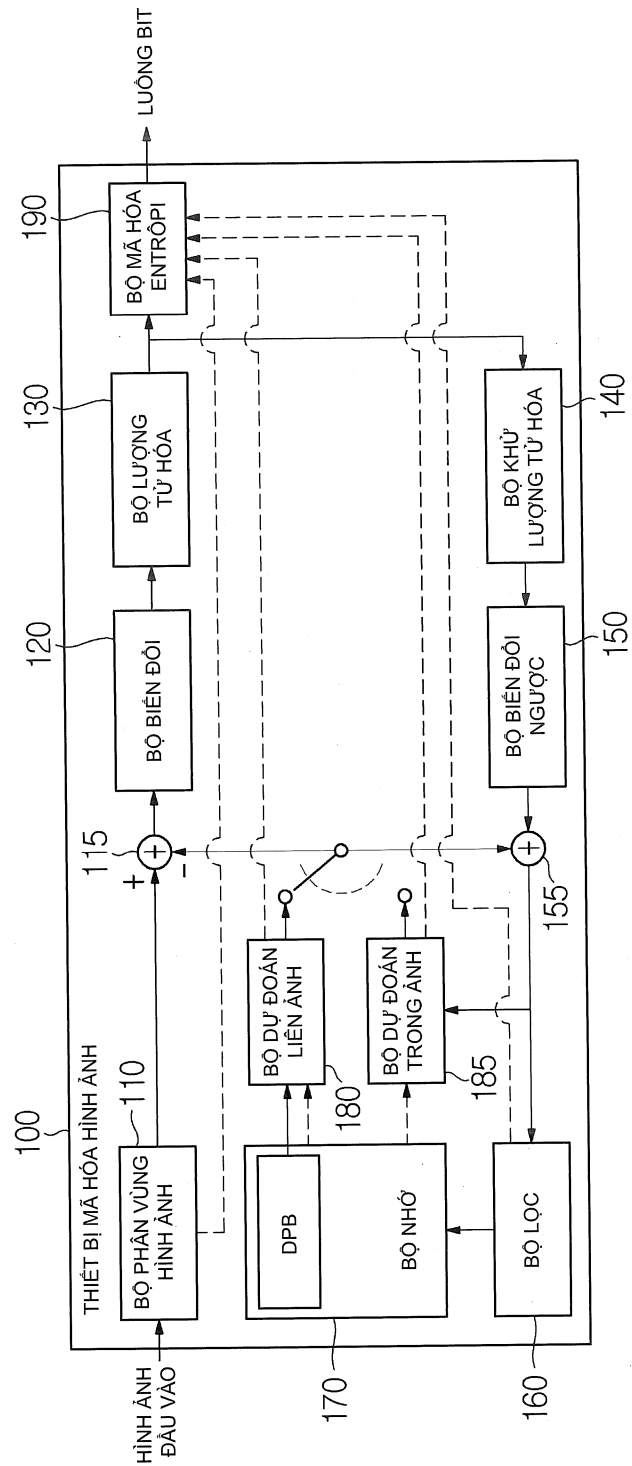


FIG. 3

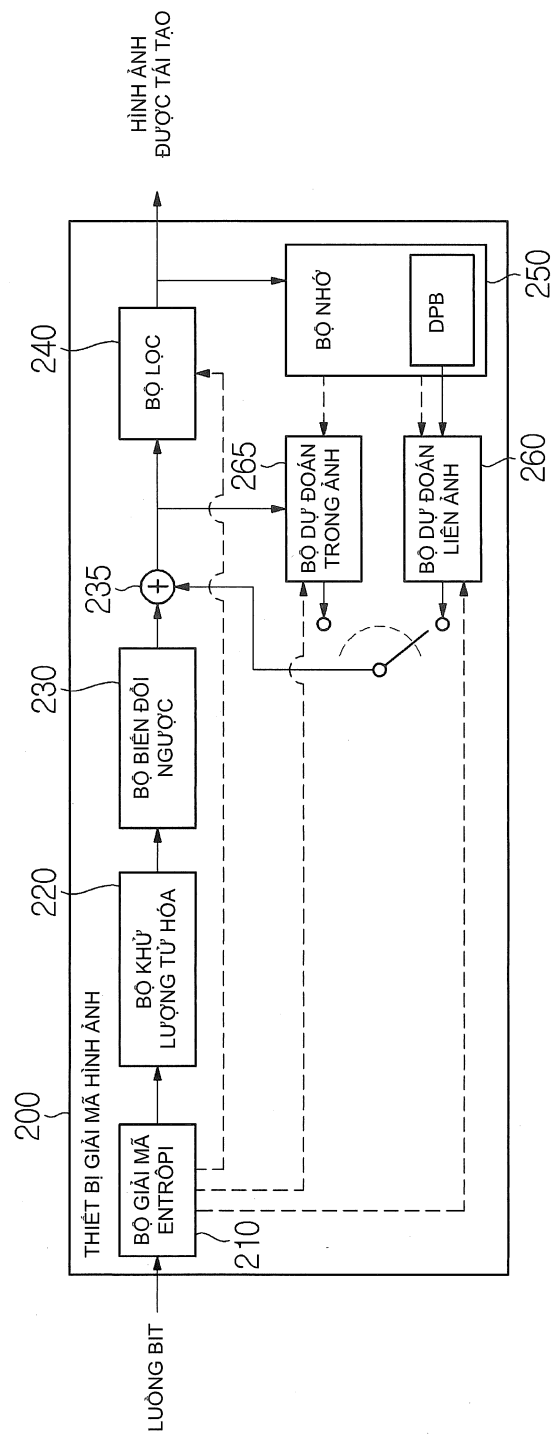


FIG. 4

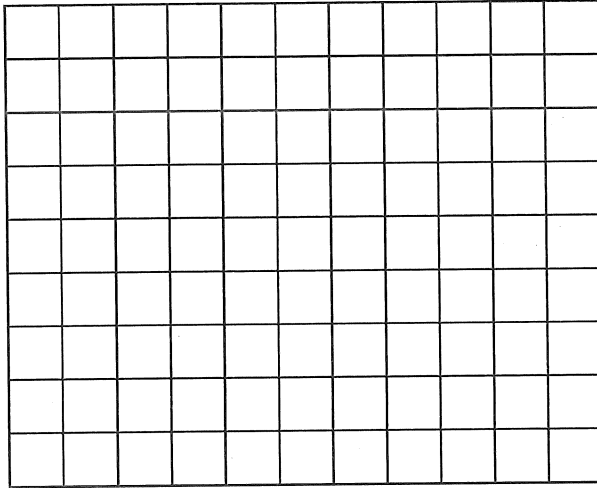


FIG. 5

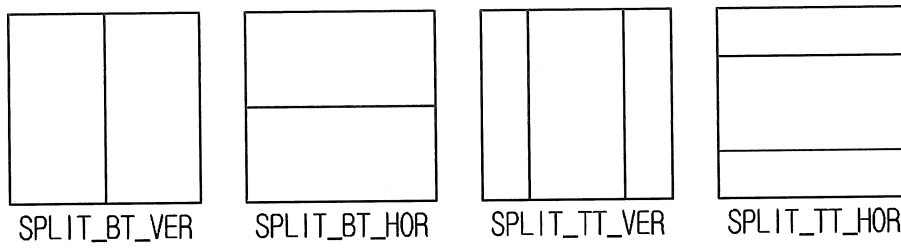


FIG. 7

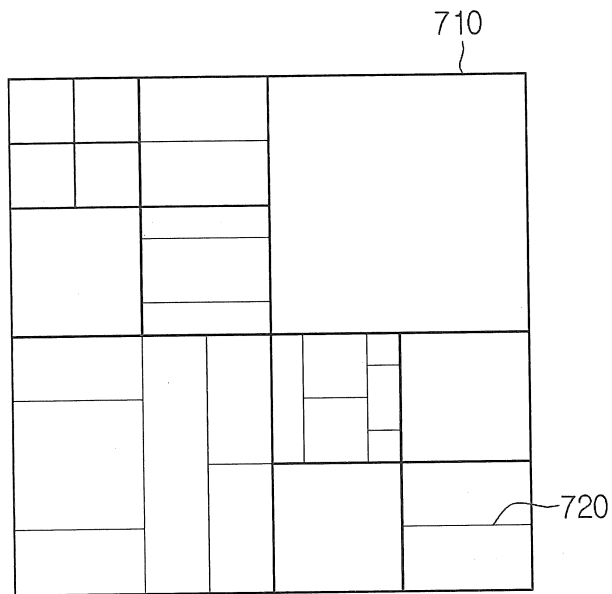


FIG. 8

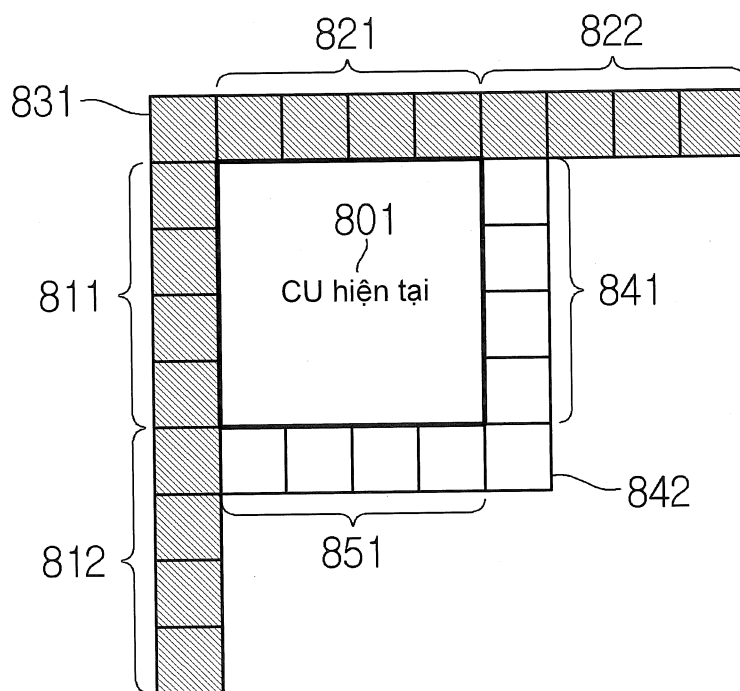


FIG. 9

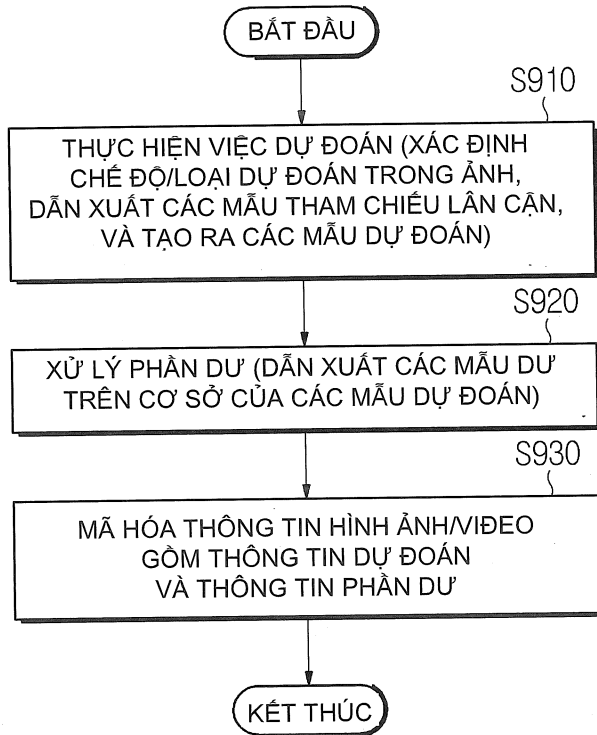


FIG. 10

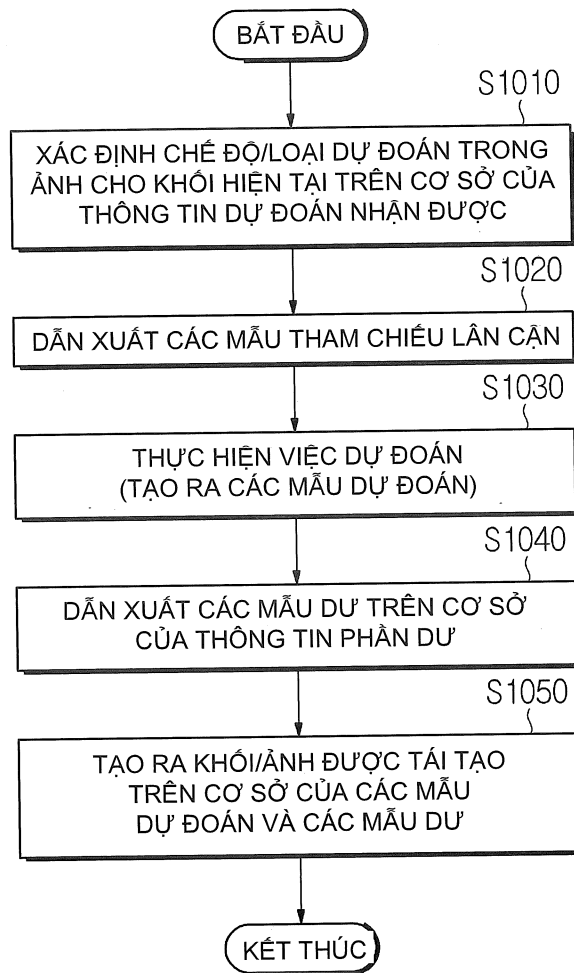


FIG. 11

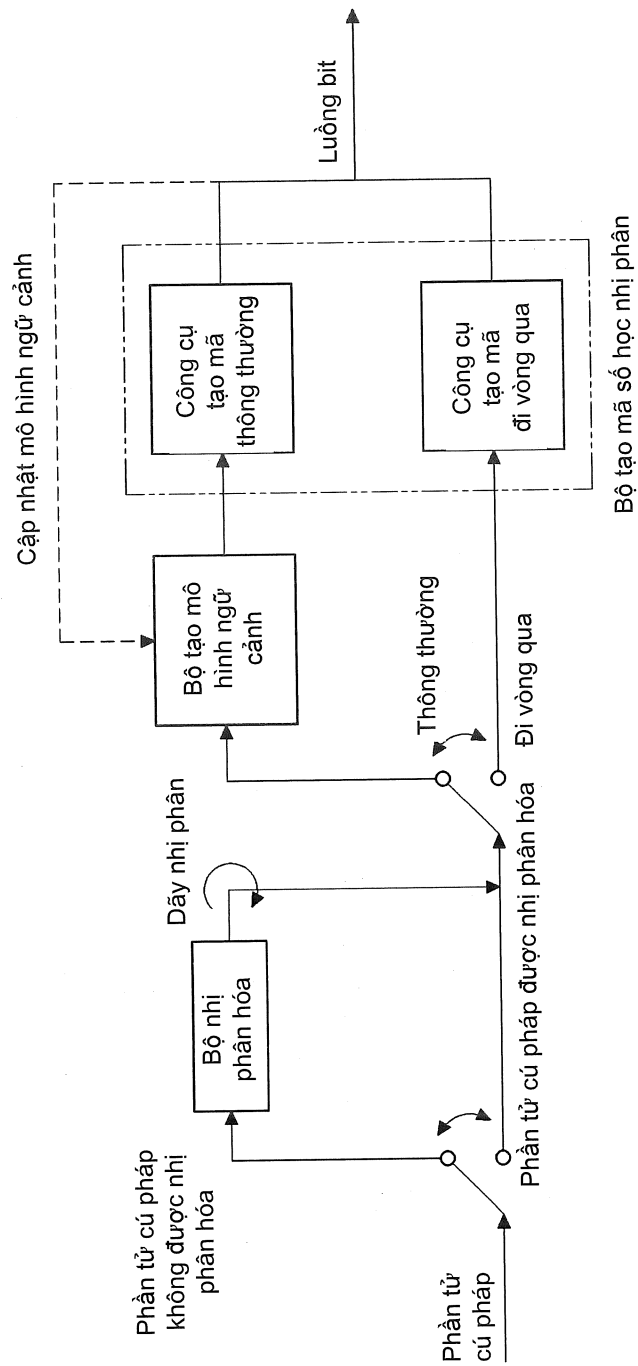


FIG. 12

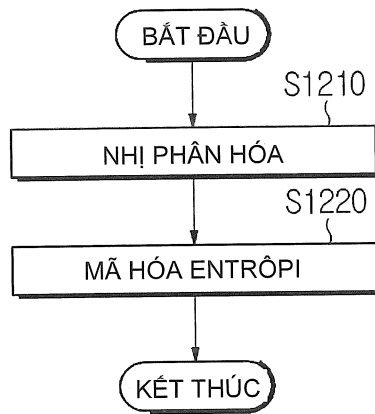


FIG. 13

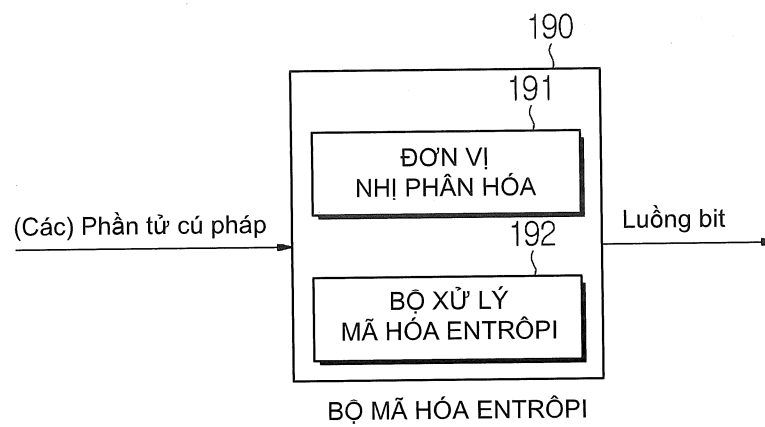


FIG. 14

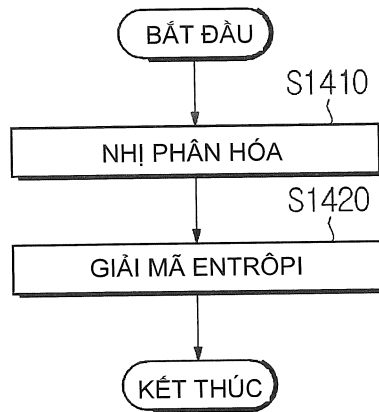


FIG. 15

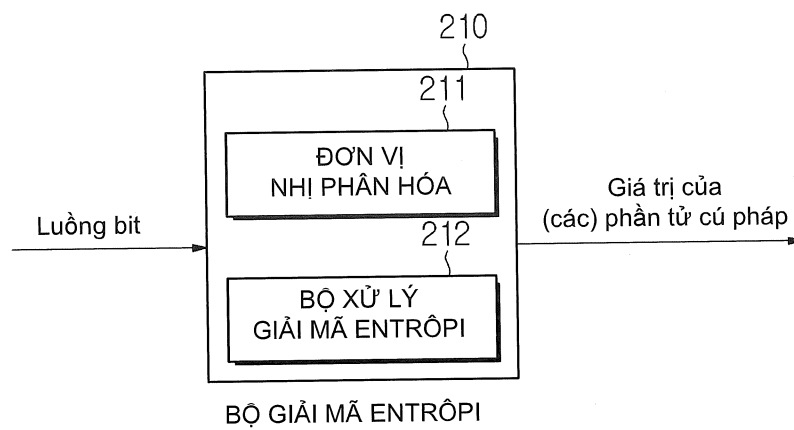


FIG. 16

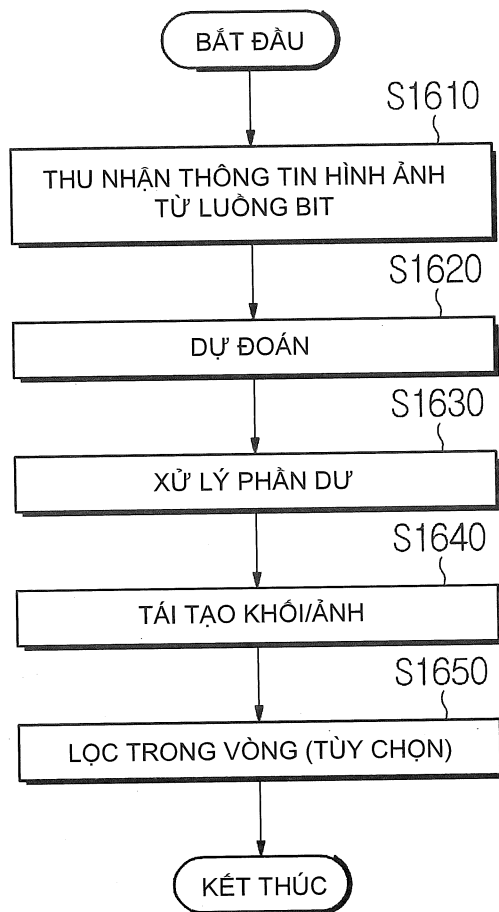


FIG. 17

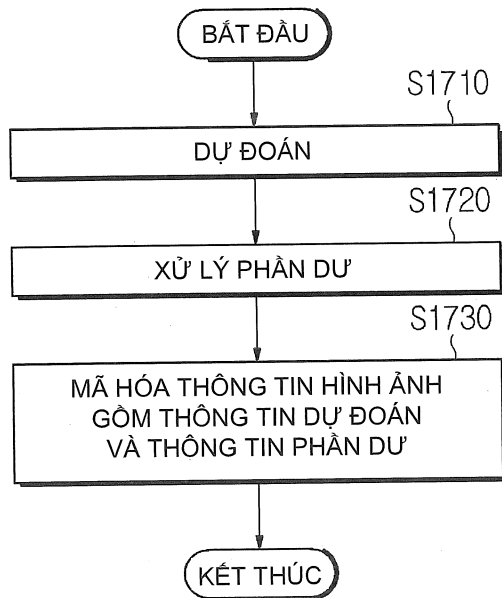


FIG. 18

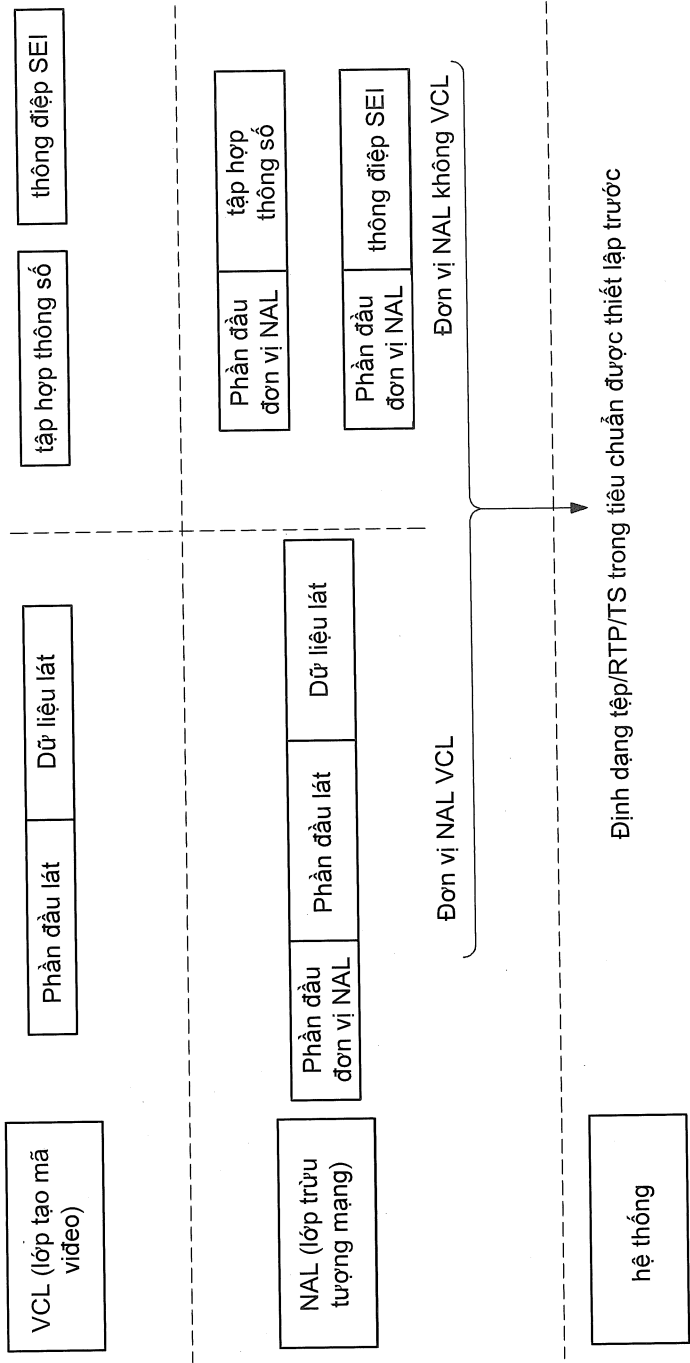


FIG. 19

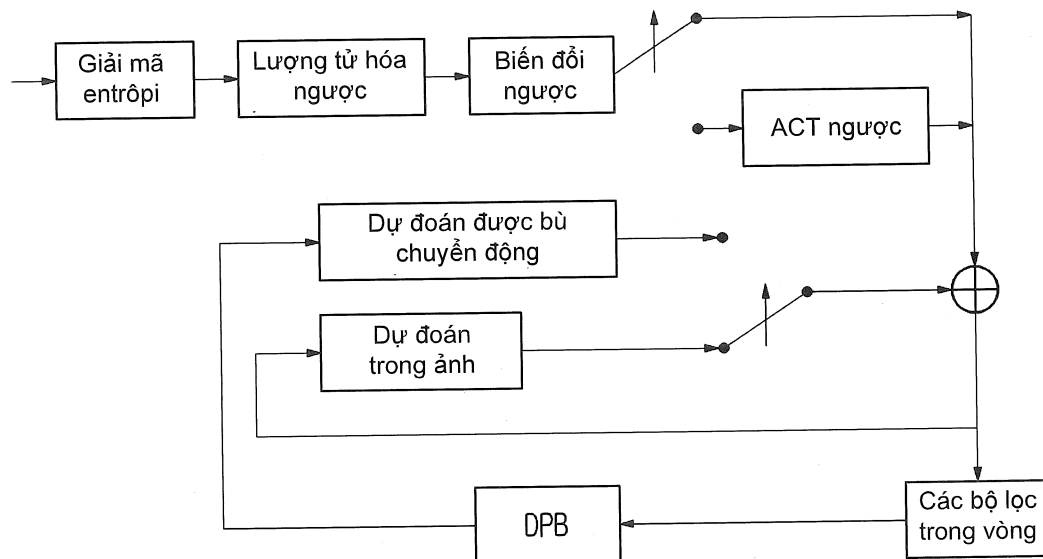


FIG. 20

seq_parameter set rbsp() {	Descriptor
sps_decoding_parameter set id	u(4)
.....	
if(chroma_format_idc == 3) {	
sps_palette_enabled_flag	u(1)
sps_act_enabled_flag	u(1)
}	

2010

FIG. 21

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
chType = treeType == DUAL_TREE_CHROMA ? 1 : 0	
if(slice_type != I sps_ibc_enabled_flag sps_palette_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA && !(((cbWidth == 4 && cbHeight == 4) modeType == MODE_TYPE_INTRA) && !sps_ibc_enabled_flag))	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] == 0 && slice_type != I && !(cbWidth == 4 && cbHeight == 4) && modeType == MODE_TYPE_ALL)	
pred_mode_flag	ae(v)
if(((slice_type == I && cu_skip_flag[x0][y0] == 0) (slice_type != I && (CuPredMode[chType][x0][y0] != MODE_INTRA (cbWidth == 4 && cbHeight == 4 && cu_skip_flag[x0][y0] == 0)))) && cbWidth <= 64 && cbHeight <= 64 && modeType != MODE_TYPE_INTER && sps_ibc_enabled_flag && treeType != DUAL_TREE_CHROMA)	
pred_mode_ibc_flag	ae(v)
if(CuPredMode[x0][y0] == MODE_INTRA && sps_palette_enabled_flag && cbWidth <= 64 && cbHeight <= 64 && cu_skip_flag[x0][y0] == 0 && modeType != MODE_TYPE_INTER)	
pred_mode_plt_flag	ae(v)
}	
if(CuPredMode[chType][x0][y0] == MODE_INTRA && sps_act_enabled_flag && treeType == SINGLE_TREE)	
cu_act_enabled_flag	ae(v)

2110—

FIG. 22

if(CuPredMode[chType][x0][y0] == MODE_INTRA	
CuPredMode[chType][x0][y0] == MODE_PLT) {	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if(pred_mode_plt_flag) {	
if(treeType == DUAL_TREE_LUMA)	
palette_coding(x0, y0, cbWidth, cbHeight, 0, 1)	
else /* SINGLE_TREE */	
palette_coding(x0, y0, cbWidth, cbHeight, 0, 3)	
} else {	
if(sps_bdpcm_enabled_flag &&	
cbWidth <= MaxTsSize && cbHeight <= MaxTsSize)	
intra_bdpcm_luma_flag	ae(v)
if(intra_bdpcm_luma_flag)	
intra_bdpcm_luma_dir_flag	ae(v)
else {	
if(sps_mip_enabled_flag)	
intra_mip_flag [x0][y0]	ae(v)
if(intra_mip_flag[x0][y0]) {	
intra_mip_transposed [x0][y0]	ae(v)
intra_mip_mode [x0][y0]	ae(v)
} else {	
if(sps_mrl_enabled_flag && ((x0 % CtbSizeY) > 0))	
intra_luma_ref_idx [x0][y0]	ae(v)
if(sps_ism_enabled_flag && intra_luma_ref_idx[x0][y0] == 0 &&	
(cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY) &&	
(cbWidth * cbHeight > MinTbSizeY * MinTbSizeY) && !cu_act_enabled_flag)	
intra_subpartitions_mode_flag [x0][y0]	ae(v)
if(intra_subpartitions_mode_flag[x0][y0] == 1)	
intra_subpartitions_split_flag [x0][y0]	ae(v)
if(intra_luma_ref_idx[x0][y0] == 0)	
intra_luma_mpm_flag [x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0]) {	
if(intra_luma_ref_idx[x0][y0] == 0)	
intra_luma_not_planar_flag [x0][y0]	ae(v)
if(intra_luma_not_planar_flag[x0][y0])	
intra_luma_mpm_idx [x0][y0]	ae(v)
} else	
intra_luma_mpm_remainder [x0][y0]	ae(v)
}	
}	
}	
}	

FIG. 23

if (treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) && ChromaArrayType != 0) {	
if(pred_mode_plt_flag && treeType == DUAL_TREE_CHROMA)	
palette_coding(x0, y0, cbWidth / SubWidthC, cbHeight / SubHeightC, 1, 2)	
else {	
if(!cu_act_enabled_flag) {	
if(cbWidth <= MaxTsSize && cbHeight <= MaxTsSize && sps_bdpcm_chroma_enabled_flag) {	
intra_bdpcm_chroma_flag	ae(v)
if(intra_bdpcm_chroma_flag)	
intra_bdpcm_chroma_dir_flag	ae(v)
} else {	
if(CclmEnabled)	
cclm_mode_flag	ae(v)
if(cclm_mode_flag)	
cclm_mode_idx	ae(v)
else	
intra_chroma_pred_mode	ae(v)
}	
}	
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
if(cu_skip_flag[x0][y0] == 0)	
general_merge_flag[x0][y0]	ae(v)
if(general_merge_flag[x0][y0])	
merge_data(x0, y0, cbWidth, cbHeight, chType)	
else if(CuPredMode[chType][x0][y0] == MODE_IBC) {	
mvd_coding(x0, y0, 0, 0)	
if(MaxNumIbcMergeCand > 1)	
mvp_l0_flag[x0][y0]	ae(v)
if(sps_amvr_enabled_flag && (MvdL0[x0][y0][0] != 0 MvdL0[x0][y0][1] != 0))	
amvr_precision_idx[x0][y0]	ae(v)

FIG. 24

} else {	
if(slice_type == B)	
inter_pred_idc[x0][y0]	ae(v)
if(sps_affine_enabled_flag && cbWidth >= 16 && cbHeight >= 16) {	
inter_affine_flag[x0][y0]	ae(v)
if(sps_affine_type_flag && inter_affine_flag[x0][y0])	
cu_affine_type_flag[x0][y0]	ae(v)
}	
if(sps_smvd_enabled_flag && !mvd_l1_zero_flag &&	
inter_pred_idc[x0][y0] == PRED_BI &&	
!inter_affine_flag[x0][y0] && RefIdxSymL0 > -1 && RefIdxSymL1 > -1)	
sym_mvd_flag[x0][y0]	ae(v)
if(inter_pred_idc[x0][y0] != PRED_L1) {	
if(NumRefIdxActive[0] > 1 && !sym_mvd_flag[x0][y0])	
ref_idx_l0[x0][y0]	ae(v)
mvd_coding(x0, y0, 0, 0)	
if(MotionModelIdc[x0][y0] > 0)	
mvd_coding(x0, y0, 0, 1)	
if(MotionModelIdc[x0][y0] > 1)	
mvd_coding(x0, y0, 0, 2)	
mvp_l0_flag[x0][y0]	ae(v)
} else {	
MvdL0[x0][y0][0] = 0	
MvdL0[x0][y0][1] = 0	
}	
if(inter_pred_idc[x0][y0] != PRED_L0) {	
if(NumRefIdxActive[1] > 1 && !sym_mvd_flag[x0][y0])	
ref_idx_l1[x0][y0]	ae(v)
if(mvd_l1_zero_flag && inter_pred_idc[x0][y0] == PRED_BI) {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
MvdCpL1[x0][y0][0][0] = 0	
MvdCpL1[x0][y0][0][1] = 0	
MvdCpL1[x0][y0][1][0] = 0	
MvdCpL1[x0][y0][1][1] = 0	
MvdCpL1[x0][y0][2][0] = 0	
MvdCpL1[x0][y0][2][1] = 0	

FIG. 25

} else {	
if(sym_mvd_flag[x0][y0]) {	
MvdL1[x0][y0][0] = -MvdL0[x0][y0][0]	
MvdL1[x0][y0][1] = -MvdL0[x0][y0][1]	
} else	
mvd_coding(x0, y0, 1, 0)	
if(MotionModelIdc[x0][y0] > 0)	
mvd_coding(x0, y0, 1, 1)	
if(MotionModelIdc[x0][y0] > 1)	
mvd_coding(x0, y0, 1, 2)	
mvp l1 flag [x0][y0]	ae(v)
}	
} else {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
}	
if((sps_amvr_enabled_flag && inter_affine_flag[x0][y0] == 0 && (MvdL0[x0][y0][0] != 0 MvdL0[x0][y0][1] != 0 MvdL1[x0][y0][0] != 0 MvdL1[x0][y0][1] != 0)) (sps_affine_amvr_enabled_flag && inter_affine_flag[x0][y0] == 1 && (MvdCpL0[x0][y0][0][0] != 0 MvdCpL0[x0][y0][0][1] != 0 MvdCpL1[x0][y0][0][0] != 0 MvdCpL1[x0][y0][0][1] != 0 MvdCpL0[x0][y0][1][0] != 0 MvdCpL0[x0][y0][1][1] != 0 MvdCpL1[x0][y0][1][0] != 0 MvdCpL1[x0][y0][1][1] != 0 MvdCpL0[x0][y0][2][0] != 0 MvdCpL0[x0][y0][2][1] != 0 MvdCpL1[x0][y0][2][0] != 0 MvdCpL1[x0][y0][2][1] != 0))) {	
amvr flag [x0][y0]	ae(v)
if(amvr_flag[x0][y0])	
amvr precision idx [x0][y0]	ae(v)
}	
if(sps_bcw_enabled_flag && inter_pred_idc[x0][y0] == PRED_BI && luma_weight_10_flag[ref_idx_10 [x0][y0]] == 0 && luma_weight_11_flag[ref_idx_11 [x0][y0]] == 0 && chroma_weight_10_flag[ref_idx_10 [x0][y0]] == 0 && chroma_weight_11_flag[ref_idx_11 [x0][y0]] == 0 && cbWidth * cbHeight >= 256)	
bcw idx [x0][y0]	ae(v)
}	
}	

FIG. 26

if(CuPredMode[chType][x0][y0] != MODE_INTRA && !pred_mode_plt_flag && general_merge_flag[x0][y0] == 0)	
cu_cbf	ae(v)
if(cu_cbf) {	
if(CuPredMode[chType][x0][y0] == MODE_INTER && sps_sbt_enabled_flag && !ciip_flag[x0][y0] && !MergeTriangleFlag[x0][y0]) {	
if(cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY) {	
allowSbtVerH = cbWidth >= 8	
allowSbtVerQ = cbWidth >= 16	
allowSbtHorH = cbHeight >= 8	
allowSbtHorQ = cbHeight >= 16	
if(allowSbtVerH allowSbtHorH allowSbtVerQ allowSbtHorQ)	
cu_sbt_flag	ae(v)
}	
if(cu_sbt_flag) {	
if((allowSbtVerH allowSbtHorH) && (allowSbtVerQ allowSbtHorQ))	
cu_sbt_quad_flag	ae(v)
if((cu_sbt_quad_flag && allowSbtVerQ && allowSbtHorQ) (!cu_sbt_quad_flag && allowSbtVerH && allowSbtHorH))	
cu_sbt_horizontal_flag	ae(v)
cu_sbt_pos_flag	ae(v)
}	
}	

FIG. 27

2710	if(sps_act_enabled_flag && CuPredMode[chType][x0][y0] != MODE_INTRA && treeType == SINGLE_TREE)	
	cu_act_enabled_flag	ae(v)
	LfstDcOnly = 1	
	LfstZeroOutSigCoeffFlag = 1	
	MtsZeroOutSigCoeffFlag = 1	
	transform_tree(x0, y0, cbWidth, cbHeight, treeType, chType)	
	lfstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC : ((IntraSubPartitionsSplitType == ISP_VER_SPLIT) ? cbWidth / NumIntraSubPartitions : cbWidth)	
	lfstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC : ((IntraSubPartitionsSplitType == ISP_HOR_SPLIT) ? cbHeight / NumIntraSubPartitions : cbHeight)	
	if(Min(lfstWidth, lfstHeight) >= 4 && sps_lfst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] Min(lfstWidth, lfstHeight) >= 16) && Max(cbWidth, cbHeight) <= MaxTbSizeY) {	
	if((IntraSubPartitionsSplitType != ISP_NO_SPLIT LfstDcOnly == 0) && LfstZeroOutSigCoeffFlag == 1)	
	lfst_idx	ae(v)
	}	
	if(treeType != DUAL_TREE_CHROMA && lfst_idx == 0 && transform_skip_flag[x0][y0][0] == 0 && Max(cbWidth, cbHeight) <= 32 && IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT && cu_sbt_flag == 0 && MtsZeroOutSigCoeffFlag == 1 && tu_cbf_luma[x0][y0]) {	
	if((CuPredMode[chType][x0][y0] == MODE_INTER && sps_explicit_mts_inter_enabled_flag) (CuPredMode[chType][x0][y0] == MODE_INTRA && sps_explicit_mts_intra_enabled_flag))	
	mts_idx	ae(v)
	}	
	}	

FIG. 28

coding_tree(x0, y0, cbWidth, cbHeight, qgOnY, qgOnC, cbSubdiv, cqtDepth, mttDepth, depthOffset, partIdx, treeTypeCurr, modeTypeCurr) {	Descriptor
if((allowSplitBtVer allowSplitBtHor allowSplitTtVer allowSplitTtHor allowSplitQt) && (x0 + cbWidth <= pps_pic_width_in_huma_samples) && (y0 + cbHeight <= pps_pic_height_in_huma_samples))	
split cu flag	ae(v)
if(pps_cu_qp_delta_enabled_flag && qgOnY && cbSubdiv <= CuQpDeltaSubdiv) {	
IsCuQpDeltaCoded = 0	
CuQpDeltaVal = 0	
CuQgTopLeftX = x0	
CuQgTopLeftY = y0	
}	
...	...

FIG. 29

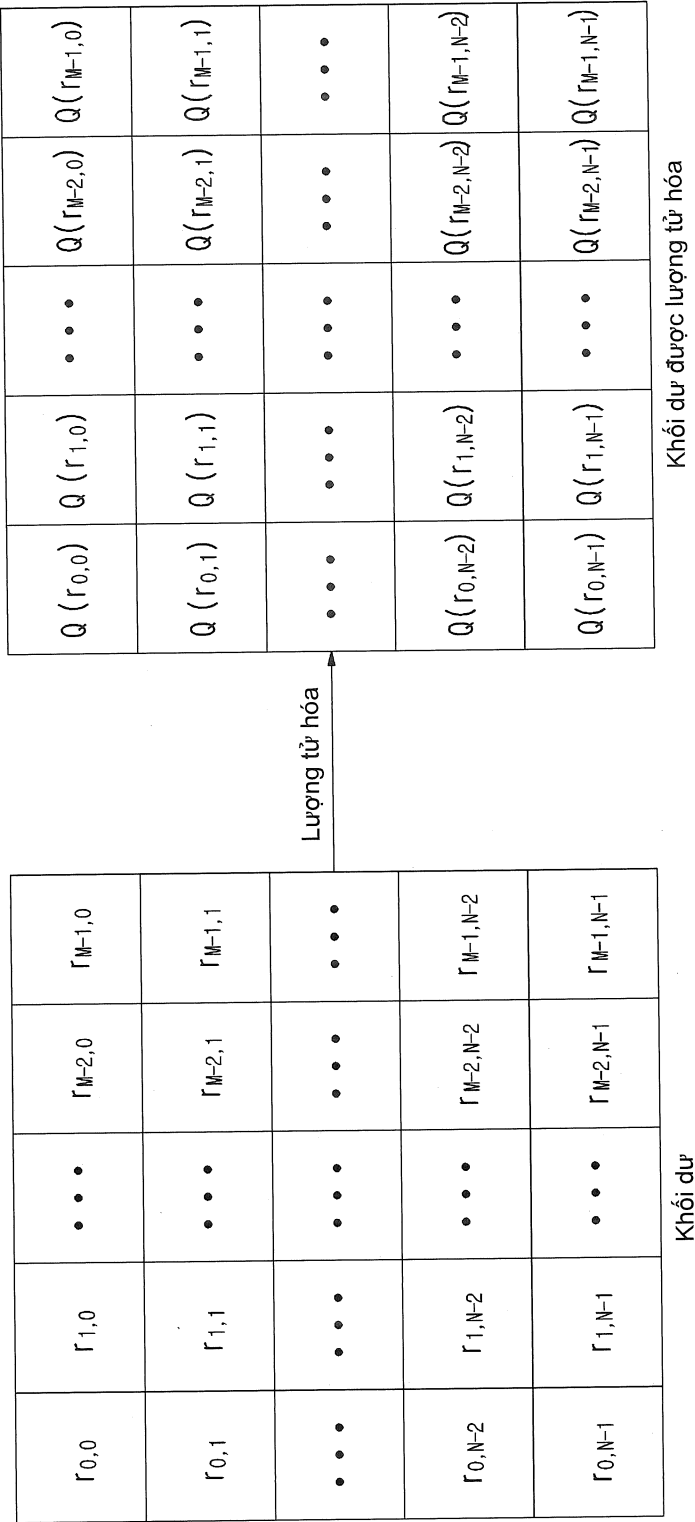


FIG. 30

$Q(r_{0,0})$	$Q(r_{1,0})$ $-Q(r_{0,0})$	$\dots$	$Q(r_{M-2,0})$ $-Q(r_{M-3,0})$	$Q(r_{M-1,0})$ $-Q(r_{M-2,0})$
$Q(r_{0,1})$	$Q(r_{1,1})$ $-Q(r_{0,1})$	$\dots$	$Q(r_{M-2,1})$ $-Q(r_{M-3,1})$	$Q(r_{M-1,1})$ $-Q(r_{M-2,1})$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$Q(r_{0,N-2})$	$Q(r_{1,N-2})$ $-Q(r_{0,N-2})$	$\dots$	$Q(r_{M-2,N-2})$ $-Q(r_{M-3,N-2})$	$Q(r_{M-1,N-2})$ $-Q(r_{M-2,N-2})$
$Q(r_{0,N-1})$	$Q(r_{1,N-1})$ $-Q(r_{0,N-1})$	$\dots$	$Q(r_{M-2,N-1})$ $-Q(r_{M-3,N-1})$	$Q(r_{M-1,N-1})$ $-Q(r_{M-2,N-1})$

BDPCM ngang

$Q(r_{0,0})$	$Q(r_{1,0})$ $-Q(r_{0,0})$	$\dots$	$Q(r_{M-2,0})$ $-Q(r_{M-3,0})$	$Q(r_{M-1,0})$ $-Q(r_{M-2,0})$
$Q(r_{0,1})$	$Q(r_{1,1})$ $-Q(r_{0,1})$	$\dots$	$Q(r_{M-2,1})$ $-Q(r_{M-3,1})$	$Q(r_{M-1,1})$ $-Q(r_{M-2,1})$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$Q(r_{0,N-2})$	$Q(r_{1,N-2})$ $-Q(r_{0,N-2})$	$\dots$	$Q(r_{M-2,N-2})$ $-Q(r_{M-3,N-2})$	$Q(r_{M-1,N-2})$ $-Q(r_{M-2,N-2})$
$Q(r_{0,N-1})$	$Q(r_{1,N-1})$ $-Q(r_{0,N-1})$	$\dots$	$Q(r_{M-2,N-1})$ $-Q(r_{M-3,N-1})$	$Q(r_{M-1,N-1})$ $-Q(r_{M-2,N-1})$

BDPCM dọc

FIG. 31

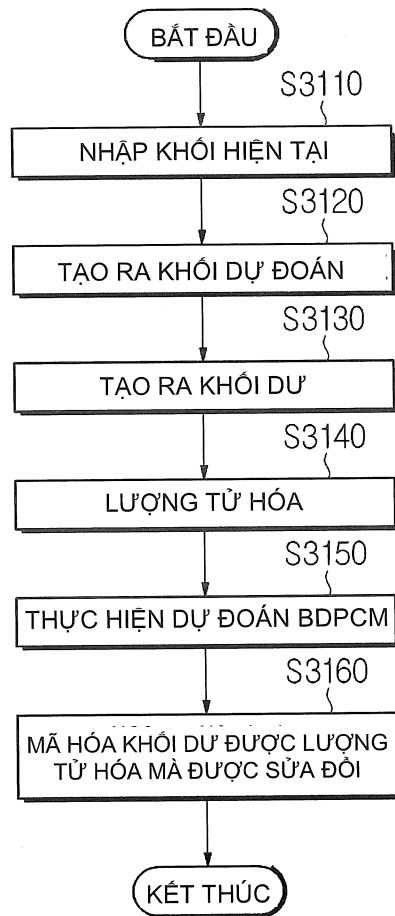


FIG. 32

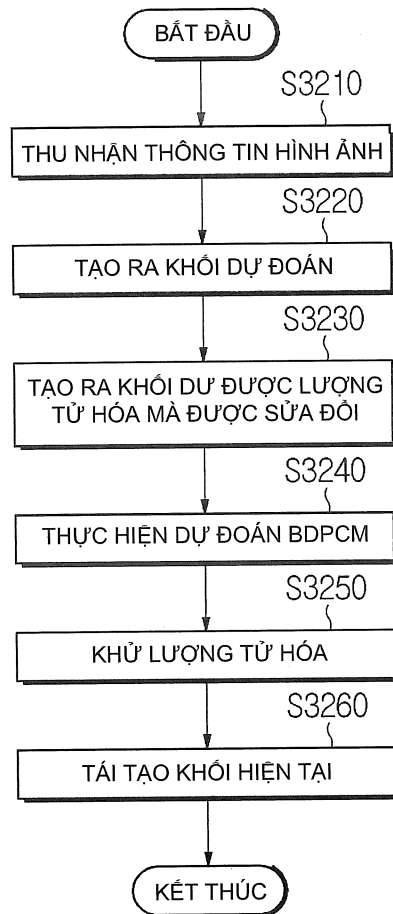


FIG. 33

seq_parameter_set_rbsp() {	Descript or
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)
(....)	
}	

FIG. 34

general_constraint_info() {	Descript or
(...)	
no_transform_skip_constraint_flag	u(1)
no_bdpcm_constraint_flag	u(1)
(...)	
}	

FIG. 35

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
chType = treeType == DUAL_TREE_CHROMA ? 1 : 0	
if(slice_type != I sps_ibc_enabled_flag sps_palette_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA && !((cbWidth == 4 && cbHeight == 4) modeType == MODE_TYPE_INTRA) && !sps_ibc_enabled_flag))	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] == 0 && slice_type != I && !(cbWidth == 4 && cbHeight == 4) && modeType == MODE_TYPE_ALL)	
pred_mode_flag	ae(v)
if(((slice_type == I && cu_skip_flag[x0][y0] == 0) (slice_type != I && (CuPredMode[chType][x0][y0] != MODE_INTRA (cbWidth == 4 && cbHeight == 4 && cu_skip_flag[x0][y0] == 0)))) && cbWidth <= 64 && cbHeight <= 64 && modeType != MODE_TYPE_INTER && sps_ibc_enabled_flag && treeType != DUAL_TREE_CHROMA)	
pred_mode_ibc_flag	ae(v)
if(((slice_type == I (cbWidth == 4 && cbHeight == 4) sps_ibc_enabled_flag) && CuPredMode[x0][y0] == MODE_INTRA) (slice_type != I && !(cbWidth == 4 && cbHeight == 4) && !sps_ibc_enabled_flag && CuPredMode[x0][y0] != MODE_INTRA)) && sps_palette_enabled_flag && cbWidth <= 64 && cbHeight <= 64 && cu_skip_flag[x0][y0] == 0 && modeType != MODE_INTER)	
pred_mode_plt_flag	ae(v)
}	
if(CuPredMode[chType][x0][y0] == MODE_INTRA CuPredMode[chType][x0][y0] == MODE_PLT) {	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
if(pred_mode_plt_flag) {	
if(treeType == DUAL_TREE_LUMA)	
palette_coding(x0, y0, cbWidth, cbHeight, 0, 1)	
else /* SINGLE_TREE */	
palette_coding(x0, y0, cbWidth, cbHeight, 0, 3)	
} else {	
if(sps_bdpcm_enabled_flag && cbWidth <= MaxTsSize && cbHeight <= MaxTsSize)	
intra_bdpcm_flag	ae(v)
if(intra_bdpcm_flag)	
intra_bdpcm_dir_flag	ae(v)
(...)	

FIG. 36

pic_parameter_set_rbsp() {	Descriptor
.....	
pps_act_qp_offsets_present_flag	u(1)
pps_act_y_qp_offset_plus5	se(v)
pps_act_cb_qp_offset_plus5	se(v)
pps_act_cr_qp_offset_plus3	se(v)
...	
}	

FIG. 37

Coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
.....	
if((treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) && ChromaArrayType != 0) {	
if(pred mode plt flag && treeType == DUAL_TREE_CHROMA)	
palette_coding(x0, y0, cbWidth / SubWidthC, cbHeight / SubHeightC, 1, 2)	
else {	
— if(!cu_act_enabled_flag) {	
if(cbWidth <= MaxTsSize && cbHeight <= MaxTsSize && sps_bdpcm_chroma_enabled_flag) {	
intra_bdpcm_chroma_flag	ae(v)
if(intra_bdpcm_chroma_flag)	
intra_bdpcm_chroma_dir_flag	ae(v)
} else if(!cu_act_enabled_flag) {	
if(CclmEnabled)	
cclm_mode_flag	ae(v)
if(cclm_mode_flag)	
cclm_mode_idx	ae(v)
else	
intra_chroma_pred_mode	ae(v)
}	
— }	
}	
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	

FIG. 38

Coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
.....	
if((treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) && ChromaArrayType != 0) {	
if(pred mode plt flag && treeType == DUAL_TREE_CHROMA)	
palette_coding(x0, y0, cbWidth / SubWidthC, cbHeight / SubHeightC, 1, 2)	
else {	
— if(!cu_act_enabled_flag) {	
if(cbWidth <= MaxTsSize && cbHeight <= MaxTsSize && sps_bdpcm_chroma_enabled_flag) {	
intra_bdpcm_chroma_flag	ae(v)
if(intra_bdpcm_chroma_flag)	
intra_bdpcm_chroma_dir_flag	ae(v)
} else {	
if(CclmEnabled)	
cclm_mode_flag	ae(v)
if(cclm_mode_flag)	
cclm_mode_idx	ae(v)
else	
intra_chroma_pred_mode	ae(v)
}	
— }	
}	
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	

FIG. 39

pic parameter set rbsp() {	Descriptor
.....	
pps act qp offsets present flag	u(1)
if(pps act qp offsets present flag) {	
pps act y qp offset plusX1	se(v)
pps act cb qp offset plusX2	se(v)
pps act cr qp offset plusX3	se(v)
pps act cbr qp offset plusX4	se(v)
}	
...	
}	
...	

FIG. 40

pic parameter set rbsp() {	Descriptor
.....	
pps act qp offsets present flag	u(1)
if(pps act qp offsets present flag) {	
pps act y qp offset plusX1	se(v)
pps act cb qp offset plusX2	se(v)
pps act cr qp offset plusX3	se(v)
pps act cbr qp offset modeA plusX4	se(v)
pps act cbr qp offset modeB plusX5	se(v)
}	
...	
}	

FIG. 41

pic parameter set rbsp() {	Descriptor
.....	
pps act qp offsets present flag	u(1)
if(pps act qp offsets present flag) {	
pps act y qp offset plusX1	se(v)
pps act cb qp offset plusX2	se(v)
pps act cr qp offset plusX3	se(v)
}	
...	
}	

FIG. 42

pic parameter set rbsp() {	Descriptor
.....	
pps slice act qp offsets present flag	u(1)
...	
...	

FIG. 43

slice header rbsp() {	Descriptor
.....	
if(pps slice act qp offsets present flag) {	
slice act y qp offset	se(v)
slice act cb qp offset	se(v)
slice act cr qp offset	se(v)
if(sps joint cbr enabled flag)	
slice act cbr qp offset	se(v)
...	
}	
}	

FIG. 44

slice header rbsp() {	Descriptor
.....	
if(pps slice act qp offsets present flag) {	
slice act y qp offset	se(v)
slice act cb qp offset	se(v)
slice act cr qp offset	se(v)
if(sps joint cbr enabled flag) {	
slice act cbr qp offset modeA	se(v)
slice act cbr qp offset modeB	se(v)
}	
...	
}	
}	

FIG. 45

slice header rbsp() {	Descriptor
.....	
if(pps slice act qp offsets present flag) {	
slice act y qp offset	se(v)
slice act cb qp offset	se(v)
slice act cr qp offset	se(v)
}	
...	
}	

FIG. 46

pic parameter set rbsp() {	Descriptor
pps pic slice act qp offsets present flag	u(1)

FIG. 47

picture header rbsp() {	Descriptor
if(pps pic slice act qp offsets present flag) {	
pic act qp offsets present flag	u(1)
if(pic act qp offsets present flag) {	
slice act y qp offset	se(v)
slice act cb qp offset	se(v)
slice act cr qp offset	se(v)
}	
}	
}	

FIG. 48

slice header rbsp() {	Descriptor
iff(pps pic slice act qp offsets present flag && !pic act qp offsets present flag) {	
slice act y qp offset	se(v)
slice act cb qp offset	se(v)
slice act cr qp offset	se(v)
}	
}	

FIG. 49

pic parameter set rbsp() {	Descriptor
...	
pps cb qp offset	se(v)
pps cr qp offset	se(v)
pps joint cbr qp offset present flag	u(1)
iff(pps joint cbr qp offset present flag)	
pps joint cbr qp offset value	se(v)
pps slice chroma qp offsets present flag	u(1)
pps cu chroma qp offset list enabled flag	u(1)
iff(pps cu chroma qp offset list enabled flag) {	
chroma qp offset list len minus1	ue(v)
for(i = 0; i <= chroma qp offset list len minus1; i++) {	
cb qp offset list[i]	se(v)
cr qp offset list[i]	se(v)
iff(pps joint cbr qp offset present flag)	
joint cbr qp offset list[i]	se(v)
}	
}	
pps act qp offsets present flag	u(1)
iff(pps act qp offsets present flag)	
pps act y qp offset	se(v)
pps act cb qp offset	se(v)
pps act cr qp offset	se(v)
iff(pps joint cbr qp offset present flag)	
pps act cbr qp offset	se(v)
pps cu act qp offset list enabled flag	u(1)
iff(pps cu act qp offset list enabled flag) {	
for(i = 0; i <= chroma qp offset list len minus1; i++) {	
act y qp offset list[i]	se(v)
act cb qp offset list[i]	se(v)
act cr qp offset list[i]	se(v)
iff(pps joint cbr qp offset present flag)	
act cbr qp offset list[i]	se(v)
}	
}	
}	
...	
}	

FIG. 50

seq parameter set rbsp() {	Descriptor
...	
sps act reversible conversion	u(1)

FIG. 51

seq parameter set rbsp() {	Descriptor
...	
sps act shift flag	u(1)
...	

FIG. 52

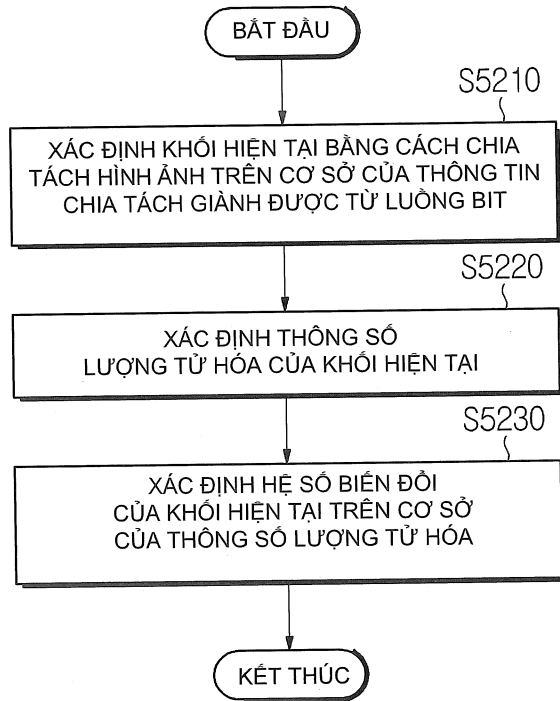


FIG. 53

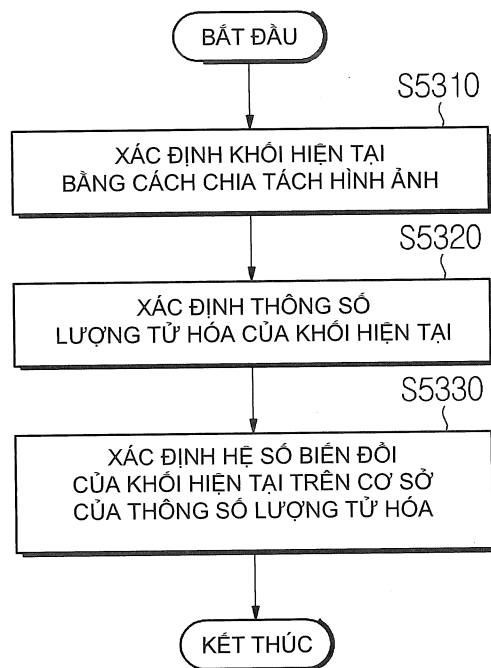


FIG. 54

