



(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
1-0053367
(51)^{2020.01} **H04N 19/124; H04N 19/18; H04N 19/70; H04N 19/132** (13) **B**

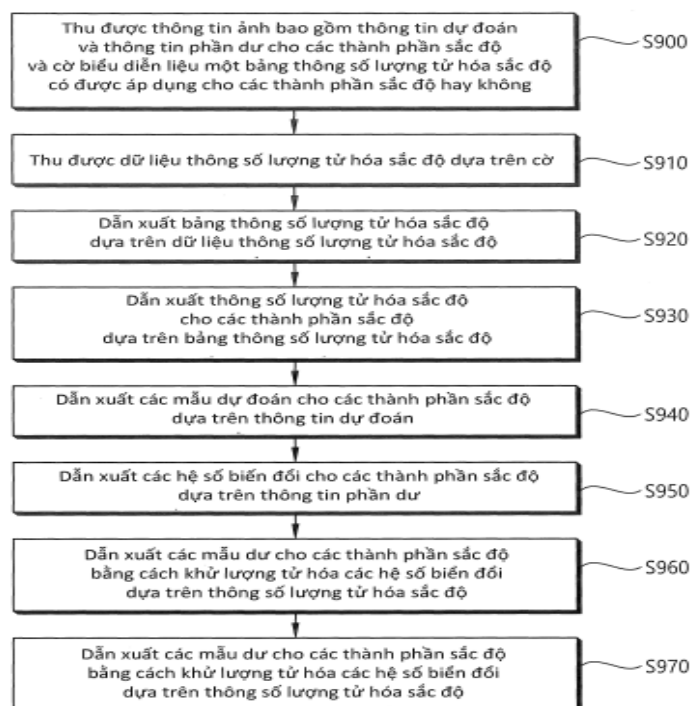
(21) 1-2022-00237 (22) 24/06/2020
(86) PCT/KR2020/008131 24/06/2020 (87) WO/2020/262913 30/12/2020
(30) 62/867,875 28/06/2019 US
(45) 25/11/2025 452 (43) 25/03/2022 408A
(73) LG ELECTRONICS INC. (KR)
128, Yeoui-daero, Yeongdeungpo-gu Seoul 07336, Republic of Korea
(72) PALURI, Seethal (IN); KIM, Seunghwan (KR); ZHAO, Jie (US).
(74) Công ty Luật TNHH T&G (TGVN)

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

(21) 1-2022-00237

(57) Sáng chế này đề cập đến phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã, theo sáng chế, bao gồm các bước: giành được thông tin ảnh bao gồm cờ chỉ báo liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng đối với các thành phần sắc độ hay không, và thông tin dự đoán và thông tin phần dư liên quan tới các thành phần sắc độ; trên cơ sở của cờ, dẫn xuất thông số lượng tử hóa sắc độ cho các thành phần sắc độ; dẫn xuất các mẫu dư cho các thành phần sắc độ trên cơ sở của thông tin phần dư và thông số lượng tử hóa sắc độ; và tạo ra hình ảnh được tái tạo trên cơ sở của các mẫu dự đoán, vốn đã được dẫn xuất trên cơ sở của thông tin dự đoán, và các mẫu dư. Sáng chế cũng đề cập đến phương pháp mã hóa hình ảnh, phương tiện lưu trữ không chuyển tiếp đọc được bởi máy tính và phương pháp truyền dữ liệu cho hình ảnh.

FIG. 9



Lĩnh vực kỹ thuật được đề cập

Sáng chế liên quan đến công nghệ tạo mã hình ảnh, và cụ thể hơn là, đến phương pháp giải mã hình ảnh và thiết bị để tạo mã thông tin ảnh bao gồm dữ liệu thông số lượng tử hóa sắc độ để dẫn xuất bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ trong hệ thống tạo mã hình ảnh.

Tình trạng kỹ thuật của sáng chế

Gần đây, nhu cầu về các ảnh độ phân giải cao, chất lượng cao, chẳng hạn các ảnh độ phân giải cao (High Definition, HD) và các ảnh độ phân giải siêu cao (Ultra High Definition, UHD), đã và đang ngày càng tăng trong các lĩnh vực khác nhau. Vì dữ liệu ảnh này có độ phân giải cao và chất lượng cao, nên lượng thông tin hoặc lượng bit cần được truyền tải so với dữ liệu ảnh kế thừa. Do đó, khi dữ liệu ảnh được truyền bằng các phương tiện như đường băng rộng dùng dây/không dây thông thường, hoặc dữ liệu ảnh được lưu trữ bằng phương tiện lưu trữ hiện có, thì chi phí truyền và chi phí lưu trữ bị tăng lên.

Theo đó, cần có kỹ thuật nén ảnh hiệu quả cao để truyền, lưu trữ, và tái tạo một cách hiệu quả thông tin của các ảnh có độ phân giải cao và chất lượng cao.

Bản chất kỹ thuật của sáng chế

Mục đích kỹ thuật của sáng chế là để đề xuất phương pháp và thiết bị để cải thiện hiệu quả tạo mã hình ảnh.

Mục đích kỹ thuật khác của sáng chế là để cung cấp phương pháp và thiết bị để dẫn xuất thông số lượng tử hóa cho thành phần sắc độ.

Theo phương án, sáng chế đề xuất phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã. Phương pháp bao gồm các bước thu được thông tin ảnh bao gồm thông tin dự đoán và thông tin phần dư cho các thành phần sắc độ và cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, thu được dữ liệu thông số lượng tử hóa sắc độ dựa trên cờ, dẫn xuất bảng thông số lượng tử hóa sắc độ dựa trên dữ liệu thông số lượng tử hóa sắc độ, dẫn xuất thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng thông số lượng tử hóa sắc độ, dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên thông tin dự đoán, dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư, dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi dựa trên thông số lượng tử hóa sắc độ và tạo ra hình ảnh được tái tạo dựa trên các mẫu dự đoán và các mẫu dư, trong đó khi giá trị của cờ là 0, thì dữ liệu thông số lượng tử hóa sắc độ bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Theo phương án, sáng chế đề xuất thiết bị giải mã thực hiện việc giải mã hình ảnh. Thiết bị giải mã bao gồm bộ mã hóa entropi được tạo cấu hình để thu được thông tin ảnh bao gồm thông tin dự đoán và thông tin phần dư cho các thành phần sắc độ và cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, để thu được dữ liệu thông số lượng tử hóa sắc độ dựa trên cờ, bộ dự đoán được tạo cấu hình để dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên thông tin dự đoán, bộ xử lý phần dư được tạo cấu hình để dẫn xuất thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng thông số lượng tử hóa sắc

độ, để dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư, để dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi dựa trên thông số lượng tử hóa sắc độ, và bộ cộng được tạo cấu hình để tạo ra hình ảnh được tái tạo dựa trên các mẫu dự đoán và các mẫu dư, trong đó khi giá trị của c là 0, thì dữ liệu thông số lượng tử hóa sắc độ bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr .

Vẫn theo phương án khác, sáng chế đề xuất phương pháp mã hóa video được thực hiện bởi thiết bị mã hóa. Phương pháp bao gồm các bước dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên việc dự đoán liên ảnh hoặc việc dự đoán trong ảnh, dẫn xuất các mẫu dư cho các thành phần sắc độ dựa trên các mẫu dự đoán, tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ và mã hóa thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ và cờ, trong đó khi giá trị của c là 0, thì dữ liệu thông số lượng tử hóa sắc độ bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr .

Vẫn theo phương án khác nữa, sáng chế đề xuất thiết bị mã hóa video. Thiết bị mã hóa bao gồm bộ dự đoán được tạo cấu hình để dẫn xuất các mẫu dự đoán cho các thành phần sắc độ, bộ xử lý phần dư được tạo cấu hình để dẫn xuất các mẫu dư cho các thành phần sắc độ dựa trên các mẫu dự đoán, để tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ hay không, để tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ, và bộ mã hóa

entrôpi được tạo cấu hình để mã hóa thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ và cờ, trong đó khi giá trị của cờ là 0, thì dữ liệu thông số lượng tử hóa sắc độ bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Theo sáng chế này, bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên cờ biểu diễn liệu cùng một bảng thông số lượng tử hóa sắc độ được sử dụng để dẫn xuất thông số lượng tử hóa cho các thành phần sắc độ, thì hiệu quả tạo mã có thể được cải thiện bằng cách thực hiện việc tạo mã dựa trên thông số lượng tử hóa theo các đặc tính của ảnh.

Theo sáng chế này, bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên dữ liệu lượng tử hóa sắc độ được báo hiệu riêng lẻ hoặc chung cho các thành phần sắc độ, hiệu quả tạo mã có thể được cải thiện bằng cách thực hiện việc tạo mã dựa trên thông số lượng tử hóa theo các đặc tính của ảnh.

Mô tả vắn tắt các hình vẽ

Fig.1 minh họa vắn tắt ví dụ về thiết bị tạo mã video/ảnh mà các phương án của sáng chế có thể áp dụng vào đó.

Fig.2 là giản đồ sơ lược minh họa cấu hình của thiết bị mã hóa video/ảnh mà phương án (các phương án) của sáng chế này có thể được áp dụng vào đó.

Fig.3 là sơ đồ giản lược minh họa cấu hình của thiết bị giải mã video/ảnh mà phương án (các phương án) của sáng chế có thể được áp dụng vào đó.

Fig.4 minh họa ví dụ về phương pháp mã hóa video/ảnh dựa trên việc dự đoán liên ảnh.

Fig.5 minh họa ví dụ về phương pháp giải mã video/ảnh dựa trên việc dự đoán liên ảnh.

Fig.6 là hình vẽ thể hiện sơ lược thủ tục dự đoán liên ảnh.

Fig.7 thể hiện theo cách giản lược phương pháp mã hóa hình ảnh bởi thiết bị mã hóa theo sáng chế này.

Fig.8 thể hiện theo cách giản lược thiết bị mã hóa để thực hiện phương pháp mã hóa hình ảnh theo sáng chế này.

Fig.9 thể hiện theo cách giản lược phương pháp giải mã hình ảnh bởi thiết bị giải mã theo sáng chế này.

Fig.10 thể hiện theo cách giản lược thiết bị giải mã để thực hiện phương pháp giải mã hình ảnh theo sáng chế này.

Fig.11 minh họa sơ đồ dạng cấu trúc của hệ thống phát luồng các nội dung mà sáng chế được áp dụng vào đó.

Mô tả chi tiết sáng chế

Sáng chế có thể được sửa đổi theo nhiều dạng khác nhau, và các phương án cụ thể của sáng chế sẽ được mô tả và được thể hiện trên các hình vẽ. Tuy nhiên, các phương án thực hiện này không nhằm giới hạn sáng chế. Các thuật ngữ được sử dụng trong phần mô tả sau đây là chỉ được sử dụng để mô tả các phương án cụ thể, chứ không nhằm giới hạn sáng chế. Sự trình bày dạng số ít cũng bao gồm sự trình bày dạng số nhiều, miễn là nó được đọc khác nhau một cách rõ ràng. Các từ ngữ như “bao gồm” và “có” là nhằm thể hiện các dấu hiệu, các số lượng, các bước, các hoạt động, các phần tử, các thành phần, hoặc những sự tổ hợp của chúng, mà được sử dụng trong phần mô tả sau đây, là tồn tại, và do đó, cần được hiểu rằng khả năng tồn tại hoặc bổ sung của một hoặc nhiều

dấu hiệu, số lượng, bước, hoạt động, phần tử, thành phần khác, hoặc những sự tổ hợp của chúng, là không bị loại trừ.

Trong khi đó, các phần tử trên các hình vẽ mà được mô tả theo sáng chế là được vẽ độc lập để thuận tiện cho việc mô tả các chức năng cụ thể khác nhau, và không có nghĩa rằng các phần tử này là được thực hiện bởi phần cứng độc lập hoặc phần mềm độc lập. Ví dụ, hai hoặc nhiều phần tử trong số các phần tử là có thể được tổ hợp để tạo thành một phần tử, hoặc một phần tử có thể được phân vùng thành nhiều phần tử. Các phương án mà trong đó các phần tử được kết hợp và/hoặc được chia là thuộc sáng chế chứ không nằm ngoài phạm vi của sáng chế.

Dưới đây, các phương án về sáng chế này sẽ được mô tả chi tiết với việc tham chiếu đến các hình vẽ kèm theo. Bên cạnh đó, các số chỉ dẫn giống nhau được sử dụng để chỉ các phần tử giống nhau trong suốt các hình vẽ, và các phần mô tả giống nhau của các phần tử giống nhau sẽ được lược bỏ.

Fig.1 minh họa vắn tắt ví dụ về thiết bị tạo mã video/ảnh mà các phương án của sáng chế có thể áp dụng vào đó.

Tham chiếu đến Fig.1, hệ thống tạo mã video/ảnh có thể bao gồm thiết bị thứ nhất (thiết bị nguồn) và thiết bị thứ hai (thiết bị nhận). Thiết bị nguồn có thể phân phối thông tin hoặc dữ liệu video/ảnh được mã hóa ở dạng của tệp hoặc phát luồng đến thiết bị nhận thông qua phương tiện lưu trữ dạng số hoặc mạng.

Thiết bị nguồn có thể gồm nguồn video, thiết bị mã hóa, và bộ truyền. Thiết bị nhận có thể bao gồm bộ nhận, thiết bị giải mã, và bộ kết xuất. Thiết bị mã hóa có thể được gọi là thiết bị mã hóa video/ảnh, và thiết bị giải mã có thể được gọi là thiết bị giải mã video/ảnh. Bộ truyền có thể được bao gồm trong thiết bị mã hóa. Bộ nhận có thể

được bao gồm trong thiết bị giải mã. Bộ kết xuất có thể bao gồm bộ phận hiển thị, bộ phận hiển thị có thể được tạo cấu hình như là thiết bị tách rời hoặc thành phần ngoài.

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

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

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

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

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

Sáng chế này liên quan đến việc tạo mã video/ảnh. Ví dụ, các phương pháp/các phương án được bộc lộ trong sáng chế này có thể được áp dụng cho phương pháp được bộc lộ trong chuẩn tạo mã video đa năng (Versatile Video Coding, VVC), chuẩn tạo mã video thiết yếu (Essential Video Coding, EVC), chuẩn audio video 1 (AOMedia Video 1, AV1), chuẩn tạo mã video thế hệ thứ hai (2nd generation of Audio Video Coding standard, AVS2), hoặc chuẩn tạo mã video/ảnh thế hệ tiếp theo (ví dụ, H.267 hoặc H.268, v.v.).

Sáng chế này giới thiệu các phương án khác nhau của việc tạo mã video/ảnh, và các phương án có thể được thực hiện trong sự tổ hợp với nhau trừ khi được đề cập theo cách khác.

Trong sáng chế này, video có thể đề cập đến một loạt các ảnh theo thời gian. Hình ảnh nói chung là đề cập tới đơn vị đang biểu diễn một ảnh trong vùng thời gian cụ thể, và hình ảnh con/lát/ngói là đơn vị cấu thành một phần của hình ảnh trong khi tạo mã. Hình ảnh con/lát/phiến có thể gồm một hoặc nhiều đơn vị cây tạo mã (Coding Tree Unit, CTU). Một hình ảnh có thể gồm có một hoặc nhiều hình ảnh con/lát/phiến. Một hình ảnh có thể gồm có một hoặc nhiều nhóm phiến. Một nhóm phiến có thể bao gồm một hoặc nhiều phiến. Gạch có thể biểu diễn vùng hình chữ nhật của các hàng CTU bên trong phiến trong hình ảnh. Phiến có thể được phân vùng thành nhiều gạch, trong số đó mỗi gạch bao gồm một hoặc nhiều hàng CTU bên trong phiến. Phiến mà không được phân vùng thành nhiều gạch có thể cũng được gọi là gạch. Quét gạch là lập thứ tự các CTU theo trình tự cụ thể đang phân vùng hình ảnh trong đó các CTU được lập thứ tự liên tiếp theo quét mảnh CTU trong gạch, các gạch bên trong phiến được lập thứ tự liên

tiếp theo quét mảnh các gạch của phiến, và các phiến trong hình ảnh được lập thứ tự liên tiếp theo quét mảnh các phiến của hình ảnh. Bên cạnh đó, hình ảnh con có thể biểu diễn vùng hình chữ nhật gồm một hoặc nhiều lát bên trong hình ảnh. Tức là, hình ảnh con chứa một hoặc nhiều lát mà nói chung là bao phủ cả vùng hình chữ nhật của hình ảnh. Phiến là vùng hình chữ nhật của các CTU nằm trong cột phiến cụ thể và hàng ngói cụ thể trong hình ảnh. Cột phiến là vùng hình chữ nhật của các CTU có chiều cao bằng với chiều cao của hình ảnh và chiều rộng được chỉ rõ bằng các phần tử cú pháp trong tập hợp thông số hình ảnh. Hàng phiến là vùng hình chữ nhật của các CTU có chiều cao được chỉ rõ bởi các phần tử cú pháp trong tập hợp thông số hình ảnh và chiều rộng bằng với chiều rộng của hình ảnh. Quét phiến là việc lập thứ tự các CTU theo trình tự cụ thể đang phân vùng hình ảnh trong đó các CTU được lập thứ tự liên tiếp trong quét mảnh CTU trong phiến, trái lại các phiến trong hình ảnh được lập thứ tự liên tiếp trong quét mảnh các phiến của hình ảnh. Lát bao gồm một số gạch nguyên của hình ảnh mà có thể được chứa dành riêng theo đơn vị NAL đơn lẻ. Lát có thể gồm có một số phiến hoàn chỉnh hoặc chỉ trình tự liên tiếp của các gạch hoàn chỉnh của một phiến. Các nhóm phiến và lát có thể được sử dụng theo kiểu thay thế cho nhau trong sáng chế này. Ví dụ, trong sáng chế này, nhóm phiến/phần đầu nhóm phiến có thể được gọi là lát/phần đầu lát.

Điểm ảnh (pixel) hoặc pel có thể nghĩa là đơn vị nhỏ nhất cấu thành một hình ảnh (hoặc ảnh). Ngoài ra, “mẫu” có thể được sử dụng như thuật ngữ tương ứng với điểm ảnh. Mẫu nói chung là có thể biểu diễn 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.

Đơn vị có thể biểu diễn đơn vị cơ sở của việc xử lý ảnh. Đơn vị có thể bao gồm ít nhất một thành phần trong số vùng cụ thể của hình ảnh và thông tin liên quan đến vùng.

Một đơn vị có thể bao gồm một khối độ chói và hai khối sắc độ (ví dụ Cb, cr). Đơn vị có thể được sử dụng theo kiểu thay thế cho nhau theo nghĩa như là khối hoặc khu vực trong một số trường hợp. Trong trường hợp chung, khối $M \times N$ có thể bao gồm các mẫu (hoặc các mảng mẫu) hoặc tập (hoặc mảng) các hệ số biến đổi gồm M cột và N hàng.

Trong sáng chế, “A hoặc B” có thể nghĩa là “chỉ A”, “chỉ B” hoặc “cả A và B”. Nói cách khác, trong sáng chế này, “A hoặc B” có thể được diễn dịch như “A và/hoặc B”. Ví dụ, trong sáng chế này, “A, B hoặc C” nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “tổ hợp bất kỳ của A, B và C”.

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

Trong sáng chế, “ít nhất một thành phần trong số A và B” có thể chỉ “chỉ A”, “chỉ B”, hoặc “cả A và B”. Bên cạnh đó, theo sáng chế này, sự trình bày “ít nhất một thành phần trong số A hoặc B” hoặc “ít nhất một thành phần trong số A và/hoặc B” có thể được diễn dịch giống như “ít nhất một thành phần trong số A và B”.

Bên cạnh đó, trong sáng chế này, “ít nhất một thành phần trong số A, B và C” nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “tổ hợp bất kỳ của A, B và C”. Ngoài ra, “ít nhất một thành phần trong số A, B hoặc C” hoặc “ít nhất một thành phần trong số A, B và/hoặc C” có thể có nghĩa là “ít nhất một thành phần trong số A, B và C”.

Bên cạnh đó, các dấu ngoặc đơn được sử dụng trong sáng chế này có thể nghĩa là “ví dụ”. Cụ thể là, khi “việc dự đoán (dự đoán trong ảnh)” được chỉ báo, thì “dự đoán trong ảnh” có thể được đề xuất như ví dụ của “dự đoán”. Nói cách khác, “dự đoán” trong sáng chế này có thể không được giới hạn ở “dự đoán trong ảnh”, và “dự đoán trong ảnh” có thể được đề xuất như ví dụ của “dự đoán”. Ngoài ra, thậm chí khi “việc dự đoán (dự

đoán trong ảnh)” được thể hiện, thì “dự đoán trong ảnh” có thể được đề xuất như ví dụ của “dự đoán”.

Trong sáng chế này, các đặc điểm kỹ thuật mà được mô tả riêng lẻ trong phạm vi một hình vẽ có thể được thực hiện riêng lẻ hoặc có thể được thực hiện cùng lúc.

Các hình vẽ sau đây được tạo ra để giải thích ví dụ cụ thể của sáng chế. Bởi vì các tên của các thiết bị cụ thể được mô tả trong các hình vẽ hoặc các tên của các tín hiệu/thông điệp/trường cụ thể được biểu thị theo cách ví dụ, nên các đặc điểm kỹ thuật của sáng chế này không được giới hạn ở các tên cụ thể được sử dụng trong các hình vẽ sau đây.

Fig.2 là giản đồ sơ lược minh họa cấu hình của thiết bị mã hóa video/ảnh mà phương án (các phương án) của sáng chế này có thể được áp dụng vào đó. Dưới đây, thiết bị mã hóa video có thể bao gồm thiết bị mã hóa hình ảnh.

Tham chiếu đến Fig.2, thiết bị mã hóa 200 bao gồm bộ phân vùng ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, và bộ mã hóa entropi 240, bộ cộng 250, bộ lọc 260, và bộ nhớ 270. Bộ dự đoán 220 có thể bao gồm bộ dự đoán liên ảnh 221 và bộ dự đoán trong ảnh 222. Bộ xử lý phần dư 230 có thể bao gồm bộ biến đổi 232, bộ lượng tử hóa 233, bộ khử lượng tử hóa 234, và bộ biến đổi ngược 235. Bộ xử lý phần dư 230 có thể còn bao gồm bộ trừ 231. Bộ cộng 250 có thể được gọi là bộ tái tạo hoặc bộ tạo ra khối được tái tạo. Bộ phân vùng ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, bộ mã hóa entropi 240, bộ cộng 250, và bộ lọc 260 có thể được tạo cấu hình bằng ít nhất một thành phần phần cứng (ví dụ, Bộ chip của bộ mã hóa hoặc bộ xử lý) theo phương án. Bên cạnh đó, bộ nhớ 270 có thể bao gồm bộ đệm hình ảnh được giải mã (Decoded Picture Buffer, DPB) hoặc có thể được tạo cấu hình bằng phương tiện lưu trữ số. Thành phần phần cứng có thể còn bao gồm bộ nhớ 270 như thành phần bên trong/bên ngoài.

Bộ phân vùng ảnh 210 có thể phân vùng ảnh đầu vào (hoặc hình ảnh hoặc khung) được nhập vào thiết bị mã hóa 200 vào trong một hoặc nhiều bộ xử lý. Ví dụ, bộ xử lý có thể được gọi là đơn vị tạo mã (Coding Unit, CU). Trong trường hợp này, đơn vị tạo mã có thể được phân vùng theo cách đệ quy theo cấu trúc cây tứ phân nhị phân tam phân (Quad-Tree Binary-Tree Ternary-Tree, QTBTNT) từ đơn vị cây tạo mã (Coding Tree Unit, CTU) hoặc đơn vị tạo mã lớn nhất (Largest Coding Unit, LCU). 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. Trong trường hợp này, ví dụ, 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. Thay vào đó, cấu trúc cây nhị phân có thể được áp dụng trước tiên. Thủ tục tạo mã theo sáng chế có thể được thực hiện dựa trên đơn vị tạo mã cuối cùng mà không được phân vùng tiếp nữa. Trong trường hợp này, đơn vị tạo mã lớn nhất có thể được sử dụng làm đơn vị tạo mã cuối cùng dựa trên hiệu quả tạo mã theo các đặc tính của ảnh, hoặc nếu cần thiết, đơn vị tạo mã có thể được chia theo cách đệ quy thành các đơn vị tạo mã có chiều sâu sâu hơn, và đơn vị lập mã có kích thước tối ưu có thể được sử dụng làm đơn vị tạo mã cuối cùng. Ở đây, thủ tục tạo mã có thể bao gồm thủ tục dự đoán, biến đổi, và tái tạo, mà sẽ được mô tả sau. Theo ví dụ khác, bộ xử lý có thể còn bao gồm đơn vị dự đoán (Prediction Unit, PU) hoặc đơn vị biến đổi (Transform Unit, TU). Trong trường hợp này, đơn vị dự đoán và đơn vị biến đổi có thể được chia tách hoặc phân vùng từ đơn vị tạo mã cuối cùng được đề cập đến ở trên. 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 phần dư từ hệ số biến đổi.

Đơn vị có thể được sử dụng theo kiểu thay thế cho nhau theo nghĩa như là khối hoặc khu vực trong một số trường hợp. Trong trường hợp chung, khối $M \times N$ có thể biểu

diễn tập các mẫu hoặc các hệ số biến đổi bao gồm M cột và N hàng. Mẫu nói chung là có thể biểu diễn điểm ảnh hoặc giá trị của điểm ảnh, có thể biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần độ chói hoặc biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ. Mẫu có thể được sử dụng như thuật ngữ tương ứng với một hình ảnh (hoặc ảnh) cho điểm ảnh hoặc pel.

Trong thiết bị mã hóa 200, tín hiệu dự đoán (khối được dự đoán, mảng mẫu dự đoán) được xuất ra từ bộ dự đoán liên ảnh 221 hoặc bộ dự đoán trong ảnh 222 được trừ từ tín hiệu ảnh đầu vào (khối gốc, mảng mẫu gốc) để tạo ra khối dư tín hiệu phần dư, mảng mẫu dư), và tín hiệu phần dư được tạo ra được truyền đến bộ biến đổi 232. Trong trường hợp này, như được thể hiện, đơn vị để trừ tín hiệu dự đoán (khối được dự đoán, mảng mẫu dự đoán) từ tín hiệu ảnh đầu vào (khối gốc, mảng mẫu gốc) trong bộ mã hóa 200 có thể được gọi là bộ trừ 231. Bộ dự đoán có thể thực hiện việc dự đoán trên khối cần được xử lý (ở dưới đây, được gọi là khối hiện tại) và tạo ra khối được dự đoán bao gồm các mẫu dự đoán cho khối hiện tại. Bộ dự đoán có thể xác định liệu việc dự đoán trong ảnh hay dự đoán liên ảnh được áp dụng trên cơ sở khối hiện tại hay CU. Như được mô tả sau trong phần mô tả về mỗi chế độ dự đoán, bộ dự đoán có thể tạo ra thông tin khác nhau liên quan đến dự đoán, như thông tin chế độ dự đoán, và truyền thông tin được tạo ra đến bộ mã hóa entropi 240. Thông tin về việc dự đoán có thể được mã hóa trong bộ mã hóa entropi 240 và được xuất ra trong dạng luồng bit.

Bộ dự đoán trong ảnh 222 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong hình ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt trong vùng lân cận của khối hiện tại hoặc có thể đặt tách ra theo chế độ dự đoán. Trong việc dự đoán trong ảnh, các chế độ dự đoán có thể bao gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Các chế độ không hướng có thể bao gồm, ví dụ, chế độ DC và

chế độ phẳng. Chế độ có hướng có thể bao gồm, ví dụ, 33 chế độ dự đoán có hướng hoặc 65 chế độ dự đoán có hướng theo cấp độ chi tiết của hướng dự đoán. Tuy nhiên, điều này 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 cài đặt. Bộ dự đoán trong ảnh 222 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.

Bộ dự đoán liên ảnh 221 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bằng vectơ chuyển động trên hình ảnh tham chiếu. Ở đây, để làm giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên ảnh, thông tin chuyển động có thể được dự đoán trong các đơn vị là các khối, các khối con, hoặc các mẫu dựa trên sự tương quan của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể bao gồm vectơ chuyển động và chỉ số hình ảnh tham chiếu. Thông tin chuyển động có thể còn bao gồm thông tin hướng dự đoán liên ảnh (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Trong trường hợp dự đoán liên ảnh, khối lân cận có thể bao gồm khối lân cận không gian có mặt trong hình ảnh hiện tại và khối lân cận thời gian có mặt trong hình ảnh tham chiếu. Hình ảnh tham chiếu chứa khối tham chiếu và hình ảnh tham chiếu bao gồm khối lân cận theo thời gian có thể là giống nhau hoặc khác nhau. Khối lân cận thời gian có thể được gọi là khối tham chiếu được đặt cùng vị trí, CU được đặt cùng vị trí (co-located CU, colCU), và tương tự, và hình ảnh tham chiếu bao gồm khối lân cận thời gian có thể được gọi là hình ảnh được đặt cùng vị trí (collocated Picture, colPic). Ví dụ, bộ dự đoán liên ảnh 221 có thể tạo cấu trúc danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận và tạo ra thông tin chỉ báo ứng viên nào được sử dụng để dẫn xuất vectơ chuyển động và/hoặc chỉ số hình 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 chế độ bỏ qua và chế độ hợp nhất, bộ dự đoán liên ảnh 221 có thể sử dụng thông tin chuyển động của khối lân cận như thông tin chuyển động của khối hiện tại. Trong chế độ bỏ qua, không giống chế độ hợp nhất, tín hiệu phần 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 chỉ báo bằng cách báo hiệu mức chênh lệch vectơ chuyển động.

Bộ dự đoán 220 có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp dự đoán khác nhau được mô tả bên dưới. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh để dự đoán một khối mà còn có thể đồng thời áp dụng cả việc dự đoán trong ảnh và việc dự đoán liên ảnh. Điều này có thể được gọi là dự đoán liên ảnh và dự đoán trong ảnh được tổ hợp (Combined Inter and Intra Prediction, CIIP). Bên cạnh đó, bộ dự đoán có thể được dựa trên chế độ dự đoán trong ảnh sao chép khối (Intra Block Copy, IBC) hoặc chế độ bảng màu để dự đoán khối. Chế độ dự đoán IBC hoặc chế độ bảng màu (palette) có thể được sử dụng cho việc tạo mã hình ảnh/video nội dung của trò chơi hoặc dạng tương tự, ví dụ, tạo mã nội dung màn (Screen Content Coding, SCC). Về cơ bản, IBC thực hiện việc dự đoán trong hình ảnh hiện tại nhưng có thể được thực hiện tương tự cho việc dự đoán liên ảnh trong đó khối tham chiếu được dẫn xuất trong hình ảnh hiện tại. Tức là, IBC có thể sử dụng ít nhất một trong các kỹ thuật dự đoán liên ảnh được mô tả trong sáng chế. Chế độ bảng màu (palette) có thể được xem xét như là ví dụ của việc tạo mã trong ảnh hoặc việc dự đoán trong ảnh. Khi chế độ bảng màu (palette) được áp dụng, giá trị mẫu nằm trong hình ảnh có thể được báo hiệu dựa trên thông tin về bảng của bảng màu (palette) và chỉ số bảng màu (palette).

Tín hiệu dự đoán được tạo ra bởi bộ dự đoán (bao gồm bộ dự đoán liên ảnh 221 và/hoặc bộ dự đoán trong ảnh 222) có thể được sử dụng để tạo ra tín hiệu được tái tạo hoặc để tạo ra tín hiệu dư. Bộ biến đổi 232 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 bao gồm ít nhất một thành phần 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). Ở đây, GBT nghĩa là việc biến đổi thu được từ đồ thị khi thông tin mối quan hệ giữa các điểm ảnh được biểu diễn bằng đồ thị. CNT đề cập đến việc biến đổi được tạo ra dựa trên tín hiệu dự đoán được tạo ra sử dụng tất cả các điểm ảnh được tái tạo từ trước. Bên cạnh đó, quy trình biến đổi có thể được áp dụng cho các khối điểm ảnh hình vuông có cùng kích thước hoặc có thể được áp dụng cho các khối có kích thước có thể thay đổi được khác hình vuông.

Bộ lượng tử hóa 233 có thể lượng tử hóa các hệ số biến đổi và truyền chúng đến bộ mã hóa entropi 240 và bộ mã hóa entropi 240 có thể mã hóa tín hiệu đã lượng tử hóa (thông tin về các hệ số biến đổi đã được lượng tử hóa) và xuất ra luồng bit. Thông tin về các hệ số biến đổi đã được lượng tử hóa có thể được gọi là thông tin phần dư. Bộ lượng tử hóa 233 có thể sắp xếp lại các hệ số biến đổi đã được lượng tử hóa dạng khối thành dạng vectơ một chiều dựa trên thứ tự quét hệ số và tạo ra thông tin về các hệ số biến đổi đã được lượng tử hóa dựa trên các hệ số biến đổi đã được lượng tử hóa ở dạng vectơ một chiều. Thông tin về các hệ số biến đổi có thể được tạo ra. Bộ mã hóa entropi 240 có thể thực hiện các phương pháp mã hóa khác nhau như, ví dụ Golomb hàm số 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 dạng tương tự. Bộ mã hóa entropi 240 có thể mã hóa thông tin cần thiết cho việc tái tạo video/ảnh chứ không phải 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 riêng biệt. Thông tin được mã hóa (ví dụ, thông tin video/ảnh được mã hóa) có thể được truyền hoặc được lưu trữ trong các đơn vị là lớp trừu tượng mạng (Network Abstraction Layer, NAL) ở dạng luồng bit. Thông tin video/ảnh có thể còn bao 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 hình 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/ảnh có thể còn bao gồm thông tin ràng buộc chung. Trong sáng chế này, thông tin và/hoặc các phần tử cú pháp được truyền/được báo hiệu từ thiết bị mã hóa đến thiết bị giải mã có thể được bao gồm trong thông tin video/hình ảnh. Thông tin video/hình ảnh có thể được mã hóa qua thủ tục mã hóa được mô tả ở trên và được bao gồm trong luồng bit. Luồng bit có thể được truyền qua mạng hoặc có thể được lưu trong phương tiện lưu trữ số. Mạng có thể bao gồm mạng quảng bá và/hoặc mạng truyền thông, và phương tiện lưu trữ số có thể bao gồm các phương tiện lưu trữ khác nhau như USB, SD, CD, DVD, Blu-ray, HDD, SSD, và dạng tương tự. Bộ truyền (không được thể hiện) truyền tín hiệu xuất ra từ bộ mã hóa entropi 240 và/hoặc đơn vị lưu trữ (không được thể hiện) đang lưu tín hiệu có thể được bao gồm như thành phần bên trong/bên ngoài của thiết bị mã hóa 200, và theo cách khác, bộ truyền có thể được bao gồm trong bộ mã hóa entropi 240.

Các hệ số biến đổi đã được lượng tử hóa được xuất ra từ bộ lượng tử hóa 233 có thể được sử dụng để tạo ra tín hiệu dự đoán. Ví dụ, tín hiệu phần 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 đối với các hệ số biến đổi đã được lượng tử hóa qua bộ khử lượng tử hóa 234 và bộ biến

đổi ngược 235. Bộ cộng 250 cộng tín hiệu phần dư được tái tạo vào tín hiệu dự đoán được xuất ra từ bộ dự đoán liên ảnh 221 hoặc bộ dự đoán trong ảnh 222 để tạo ra tín hiệu được tái tạo (hình ảnh được tái tạo, khối được tái tạo, mảng mẫu được tái tạo). Nếu không có phần dư nào 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 250 có thể được gọi là bộ tái tạo hoặc bộ tạo ra khối được tái tạo. Tín hiệu được tái tạo được tạo ra có thể được sử dụng cho việc dự đoán trong ảnh khối tiếp theo cần được xử lý trong hình ảnh hiện tại và có thể được sử dụng cho việc dự đoán liên ảnh hình ảnh tiếp theo qua việc lọc như được mô tả bên dưới.

Trong khi đó, phép ánh xạ độ chói cùng với việc định cỡ sắc độ (Luma Mapping With Chroma Scaling, LMCS) có thể được áp dụng trong suốt quá trình mã hóa và/hoặc tái tạo hình ảnh.

Bộ lọc 260 có thể cải thiện chất lượng ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái tạo. Ví dụ, bộ lọc 260 có thể tạo ra hình ảnh được tái tạo được sửa đổi bằng cách áp dụng các phương pháp lọc khác nhau đối với hình ảnh được tái tạo và lưu trữ hình ảnh được tái tạo được sửa đổi trong bộ nhớ 270, cụ thể, DPB của bộ nhớ 270. Các phương pháp lọc khác nhau có thể bao gồm, ví dụ, lọc khử khối, bù tương thích mẫu, bộ lọc vòng lặp tương thích, bộ lọc song phương, và dạng tương tự. Bộ lọc 260 có thể tạo ra thông tin khác nhau liên quan tới việc lọc và truyền thông tin được tạo ra đến bộ mã hóa entropi 240 như sẽ được mô tả sau trong phần mô tả về mỗi phương pháp lọc. Thông tin liên quan đến việc lọc có thể được mã hóa bởi bộ mã hóa entropi 240 và được xuất ra trong dạng luồng bit.

Hình ảnh được tái tạo được sửa đổi mà được truyền đến bộ nhớ 270 là có thể được sử dụng làm hình ảnh tham chiếu ở bộ dự đoán liên ảnh 221. Khi việc dự đoán liên ảnh

được áp dụng qua thiết bị mã hóa, thì việc không khớp dự đoán giữa thiết bị mã hóa 200 và thiết bị giải mã 300 có thể tránh được và hiệu quả mã hóa có thể được cải thiện.

DPB của bộ nhớ DPB 270 có thể lưu hình ảnh được tái tạo được biến đổi để sử dụng như là hình ảnh tham chiếu trong bộ dự đoán liên ảnh 221. Bộ nhớ 270 có thể lưu trữ thông tin chuyển động của khối mà thông tin chuyển động trong ảnh hiện tại được dẫn xuất (hoặc được mã hóa) 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 bộ dự đoán liên ảnh 221 và được sử dụng như là thông tin chuyển động của khối lân cận không gian hoặc thông tin chuyển động của khối lân cận thời gian. Bộ nhớ 270 có thể lưu trữ các mẫu được tái tạo của các khối được tái tạo trong hình ảnh hiện tại và có thể chuyển các mẫu được tái tạo đến bộ dự đoán trong ảnh 222.

Fig.3 là sơ đồ giản lược minh họa cấu hình của thiết bị giải mã video/ảnh mà phương án (các phương án) của sáng chế có thể được áp dụng vào đó.

Tham chiếu đến Fig.3, thiết bị giải mã 300 có thể bao gồm bộ giải mã entropi 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, bộ lọc 350, bộ nhớ 360. Bộ dự đoán 330 có thể bao gồm bộ dự đoán liên ảnh 331 và bộ dự đoán trong ảnh 332. Bộ xử lý phần dư 320 có thể bao gồm bộ khử lượng tử hóa 321 và bộ biến đổi ngược 322. Bộ giải mã entropi 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, và bộ lọc 350 có thể được tạo cấu hình bởi thành phần phần cứng (ví dụ, bộ chip của bộ giải mã hoặc bộ xử lý) theo một phương án. Bên cạnh đó, bộ nhớ 360 có thể bao gồm bộ đệm hình ảnh được giải mã (DPB) hoặc có thể được tạo cấu hình bằng phương tiện lưu trữ số. Thành phần phần cứng có thể còn bao gồm bộ nhớ 360 làm thành phần bên trong/bên ngoài.

Khi luồng bit bao gồm thông tin video/ảnh được nhập vào, thì thiết bị giải mã 300 có thể tái tạo ảnh tương ứng với quy trình trong đó thông tin video/ảnh được xử lý trong

thiết bị mã hóa của Fig.2. Ví dụ, thiết bị giải mã 300 có thể dẫn xuất các đơn vị/các khối dựa trên thông tin liên quan đến sự phân vùng khối được thu từ luồng bit. Thiết bị giải mã 300 có thể thực hiện việc giải mã sử dụng bộ xử lý được áp dụng trong thiết bị mã hóa. Do đó, bộ xử lý việc giải mã có thể đơn vị tạo mã chẳng hạn và đơn vị tạo mã có thể được phân vùng theo cấu trúc cây tứ phân, cấu trúc cây nhị phân và/hoặc cấu trúc cây tam phân từ đơn vị cây tạo mã hoặc đơn vị tạo mã lớn nhất. Một hoặc nhiều đơn vị biến đổi có thể được dẫn xuất từ đơn vị tạo mã. Tín hiệu ảnh được tái tạo được giải mã và xuất ra qua thiết bị giải mã 300 có thể được tái tạo thông qua thiết bị tái tạo.

Thiết bị giải mã 300 có thể nhận tín hiệu mà được xuất ra từ thiết bị mã hóa trên Fig.2 trong dạng luồng bit, và tín hiệu nhận được có thể được giải mã thông qua bộ giải mã entropi 310. Ví dụ, bộ giải mã entropi 310 có thể phân tích cú pháp luồng bit để dẫn xuất thông tin (ví dụ, thông tin video/ảnh) cần thiết cho việc tái tạo ảnh (hoặc tái tạo hình ảnh). Thông tin video/ảnh có thể còn bao gồm thông tin về các tập hợp thông số khác nhau như tập hợp thông số thích ứng (APS), tập hợp thông số hình ảnh (PPS), tập hợp thông số chuỗi (SPS), hoặc tập hợp thông số video (VPS). Bên cạnh đó, thông tin video/ảnh có thể còn bao gồm thông tin ràng buộc chung. Thiết bị giải mã có thể còn giải mã hình ảnh dựa trên thông tin về tập hợp thông số và/hoặc thông tin ràng buộc chung. Thông tin được báo hiệu/thu được và/hoặc các phần tử cú pháp được mô tả ở sau trong sáng chế này có thể được giải mã có thể giải mã thủ tục giải mã và thu được từ luồng bit. Ví dụ, bộ giải mã entropi 310 giải mã thông tin trong luồng bit dựa trên phương pháp tạo mã như tạo mã Golomb hàm số mũ, CAVLC hoặc CABAC, và xuất ra các phần tử cú pháp được yêu cầu cho việc tái tạo ảnh và các giá trị được lượng tử hóa của các hệ số biến đổi cho phần dư. Cụ thể hơn, phương pháp giải mã entropi CABAC có thể nhận ngăn (bin) tương ứng với mỗi phần tử cú pháp trong luồng bit, xác định mô hình ngữ cảnh sử dụng thông tin về phần tử cú pháp mục tiêu giải mã, thông tin về khối

mục tiêu giải mã hoặc thông tin về ký hiệu/bin được giải mã trong pha trước, và thực hiện việc giải mã số học trên bin này bằng cách dự đoán xác suất xuất hiện bin 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/bin được giải mã cho mô hình ngữ cảnh của ký hiệu/bin 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 310 có thể được cấp đến bộ dự đoán (bộ dự đoán liên ảnh 332 và bộ dự đoán trong ảnh 331), và giá trị dư mà việc giải mã entropi được thực hiện trên đó trong bộ giải mã entropi 310, tức là, các hệ số biến đổi được lượng tử hóa và thông tin thông số liên quan, có thể được cấp vào bộ xử lý phần dư 320. Bộ xử lý phần dư 320 có thể dẫn xuất tín hiệu phần dư (khối dư, các mẫu dư, mảng mẫu dư). Bên cạnh đó, thông tin về việc lọc trong số thông tin được giải mã bởi bộ giải mã entropi 310 có thể được cấp đến bộ lọc 350. Trong khi đó, bộ nhận (không được thể hiện) để nhận tín hiệu được xuất ra từ thiết bị mã hóa có thể còn được tạo cấu hình như phần tử bên trong/bên ngoài của thiết bị giải mã 300, hoặc bộ nhận có thể là thành phần của bộ giải mã entropi 310. Trong khi đó, thiết bị giải mã theo sáng chế này có thể được gọi là thiết bị giải mã video/ảnh/hình ảnh, và thiết bị giải mã có thể được phân loại thành bộ giải mã thông tin (bộ giải mã thông tin video/ảnh/hình ảnh) và bộ giải mã mẫu (bộ giải mã mẫu video/ảnh/hình ảnh). Bộ giải mã thông tin có thể bao gồm bộ giải mã entropi 310, và bộ giải mã mẫu có thể bao gồm ít nhất một trong bộ khử lượng tử hóa 321, bộ biến đổi ngược 322, bộ cộng 340, bộ lọc 350, bộ nhớ 360, bộ dự đoán liên ảnh 332, và bộ dự đoán trong ảnh 331.

Bộ khử lượng tử hóa 321 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 321 có thể sắp xếp lại các hệ số biến đổi được lượng tử hóa dưới dạng của loại 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. Bộ khử lượng tử hóa 321 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 lượng tử hóa (ví dụ, thông tin kích thước bước lượng tử hóa) và thu được các hệ số biến đổi.

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

Bộ dự đoán có thể thực hiện việc dự đoán trên khối hiện tại và tạo ra khối được dự đoán bao gồm các mẫu dự đoán cho khối hiện tại. Bộ dự đoán có thể xác định liệu việc dự đoán trong ảnh hoặc việc dự đoán liên ảnh được áp dụng cho khối hiện tại dựa trên thông tin về dự đoán được xuất ra từ bộ giải mã entropi 310 hay không và có thể xác định chế độ dự đoán trong ảnh/liên ảnh cụ thể.

Bộ dự đoán 320 có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp dự đoán khác nhau được mô tả bên dưới. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc dự đoán trong ảnh hoặc dự đoán liên ảnh để dự đoán một khối mà còn có thể áp dụng đồng thời việc dự đoán trong ảnh và dự đoán liên ảnh. Điều này có thể được gọi là dự đoán liên ảnh và dự đoán trong ảnh được tổ hợp (CIIP). Bên cạnh đó, bộ dự đoán có thể được dựa trên chế độ dự đoán trong ảnh sao chép khối (IBC) hoặc chế độ bảng màu để dự đoán khối. Chế độ dự đoán IBC hoặc chế độ bảng màu (palette) có thể được sử dụng cho việc tạo mã hình ảnh/video nội dung của trò chơi hoặc dạng tương tự, ví dụ, tạo mã nội dung màn (SCC). Về cơ bản, IBC thực hiện việc dự đoán trong hình ảnh hiện tại nhưng có thể được thực hiện tương tự cho việc dự đoán liên ảnh trong đó khối tham chiếu được dẫn xuất trong hình ảnh hiện tại. Tức là, IBC có thể sử dụng ít nhất một trong các kỹ thuật dự đoán liên ảnh được mô tả trong sáng chế. Chế độ bảng màu (palette) có thể được xem xét như là ví dụ của việc tạo mã trong ảnh hoặc việc dự đoán trong ảnh.

Khi chế độ bảng màu (palette) được áp dụng, giá trị mẫu nằm trong hình ảnh có thể được báo hiệu dựa trên thông tin về bảng của bảng màu (palette) và chỉ số bảng màu (palette).

Bộ dự đoán trong ảnh 331 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong hình ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt trong vùng lân cận của khối hiện tại hoặc có thể đặt tách ra theo chế độ dự đoán. Trong việc dự đoán trong ảnh, các chế độ dự đoán có thể bao gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Bộ dự đoán trong ảnh 331 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.

Bộ dự đoán liên ảnh 332 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bằng vectơ chuyển động trên hình ảnh tham chiếu. Trong trường hợp này, để làm giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên ảnh, thông tin chuyển động có thể được dự đoán trong các đơn vị là các khối, các khối con, hoặc các mẫu dựa trên mối tương quan của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể bao gồm vectơ chuyển động và chỉ số hình ảnh tham chiếu. Thông tin chuyển động có thể còn bao gồm thông tin hướng dự đoán liên ảnh (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Trong trường hợp dự đoán liên ảnh, khối lân cận có thể bao gồm khối lân cận không gian có mặt trong hình ảnh hiện tại và khối lân cận thời gian có mặt trong hình ảnh tham chiếu. Ví dụ, bộ dự đoán liên ảnh 332 có thể tạo cấu trúc danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận và dẫn xuất vectơ chuyển động của khối hiện tại và/hoặc chỉ số hình 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 bao gồm thông tin chỉ báo chế độ dự đoán liên ảnh cho khối hiện tại.

Bộ cộng 340 có thể tạo ra tín hiệu được tái tạo (hình ảnh được tái tạo, khối được tái tạo, mảng mẫu được tái tạo) bằng cộng tín hiệu phần dư thu được vào tín hiệu dự đoán (khối được dự đoán, mảng mẫu được dự đoán) được xuất ra từ bộ dự đoán (bao gồm bộ dự đoán liên ảnh 332 và/hoặc bộ dự đoán trong ảnh 331). Nếu không có phần dư nào cho khối cần được xử lý, như khi chế độ bỏ qua được áp dụng, thì khối được dự đoán có thể được sử dụng như khối được tái tạo.

Bộ cộng 340 có thể được gọi là bộ tái tạo hoặc bộ tạo khối được tái tạo. Tín hiệu được tái tạo được tạo ra có thể được sử dụng cho việc dự đoán trong ảnh khối tiếp theo cần được xử lý trong hình ảnh hiện tại, có thể được xuất ra qua bước lọc được mô tả ở trên, hoặc có thể được sử dụng cho việc dự đoán liên ảnh hình ảnh tiếp theo.

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

Bộ lọc 350 có thể cải thiện chất lượng ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái tạo. Ví dụ, bộ lọc 350 có thể tạo ra hình ảnh được tái tạo được sửa đổi bằng cách áp dụng các phương pháp lọc khác nhau đối với hình ảnh được tái tạo và lưu trữ hình ảnh được tái tạo được sửa đổi trong bộ nhớ 360, cụ thể, DPB của bộ nhớ 360. Các phương pháp lọc khác nhau có thể bao gồm, ví dụ, lọc khử khối, bù tương thích mẫu, bộ lọc vòng lặp tương thích, bộ lọc song phương, và dạng tương tự.

Hình ảnh được tái tạo (được sửa đổi) được lưu trữ trong DPB của bộ nhớ 360 có thể được sử dụng như hình ảnh tham chiếu trong bộ dự đoán liên ảnh 332. Bộ nhớ 360 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 bộ dự đoán liên ảnh 260 để được sử dụng như thông tin chuyển động của khối lân cận không gian hoặc thông tin chuyển động của khối lân cận thời gian. Bộ nhớ 360 có thể lưu trữ các mẫu được tái tạo của các khối được tái tạo trong hình ảnh hiện tại và chuyển các mẫu được tái tạo đến bộ dự đoán trong ảnh 331.

Trong sáng chế này, các phương án được mô tả trong bộ lọc 260, bộ dự đoán liên ảnh 221, và bộ dự đoán trong ảnh 222 của thiết bị mã hóa 200 có thể là giống như hoặc được áp dụng lần lượt để tương ứng với bộ lọc 350, bộ dự đoán liên ảnh 332, và bộ dự đoán trong ảnh 331 của thiết bị giải mã 300. Việc tương tự cũng có thể áp dụng cho đơn vị 332 và bộ dự đoán trong ảnh 331.

Trong sáng chế này, ít nhất một trong việc lượng tử hóa/lượng tử hóa ngược và/hoặc biến đổi/biến đổi ngược có thể được lược bỏ. Khi việc lượng tử hóa/lượng tử hóa ngược được lược bỏ, thì các hệ số biến đổi được lượng tử hóa có thể được gọi là các hệ số biến đổi. Khi việc biến đổi/biến đổi ngược được lược bỏ, thì các hệ số biến đổi có thể được gọi là các hệ số hoặc các hệ số dư, hoặc có thể vẫn được gọi là các hệ số biến đổi cho sự đồng nhất về sự trình bày.

Trong sáng chế này, hệ số biến đổi được lượng tử hóa và hệ số biến đổi có thể được đề cập đến lần lượt như hệ số biến đổi và hệ số được định cỡ. Trong trường hợp này, thông tin phần dư có thể bao gồm thông tin về hệ số (các hệ số) biến đổi, và thông tin về hệ số (các hệ số) biến đổi có thể được báo hiệu thông qua cú pháp tạo mã phần dư. Các hệ số biến đổi có thể được dẫn xuất dựa trên thông tin dư (hoặc thông tin về hệ số (các hệ số) biến đổi), và các hệ số biến đổi được định cỡ có thể được dẫn xuất bằng cách biến đổi ngược (định cỡ) trên các hệ số biến đổi. Các mẫu dư có thể được dẫn xuất

dựa trên việc biến đổi ngược (biến đổi) trên các hệ số biến đổi được định cỡ. Điều này có thể cũng được áp dụng/được thể hiện trong các phần khác của sáng chế này.

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

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

thì thiết bị mã hoá cũng có thể khử lượng tử/biến đổi ngược các hệ số biến đổi được lượng tử hoá để dẫn xuất khối dư và tạo ra ảnh được tái tạo dựa trên đó.

Việc dự đoán trong ảnh có thể đề cập đến việc dự đoán mà tạo ra các mẫu dự đoán cho khối hiện tại dựa trên các mẫu tham chiếu trong hình ảnh mà khối hiện tại thuộc về (ở dưới đây được gọi là hình ảnh hiện tại). Khi việc dự đoán trong ảnh được áp dụng cho khối hiện tại, thì các mẫu tham chiếu lân cận cần 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 bao gồm mẫu gần kề với biên trái của khối hiện tại có kích thước là $nW \times nH$ và tổng của $2 \times nH$ mẫu gần kề phía dưới cùng-bên trái của khối hiện tại, mẫu gần kề với biên phía trên cùng của khối hiện tại và tổng $2 \times nW$ mẫu gần kề với phía trên cùng-bên phải, và một mẫu gần kề với phía trên cùng-bên trái của khối hiện tại. Thay vào đó, các mẫu tham chiếu lân cận của khối hiện tại có thể bao gồm nhiều cột của các mẫu lân cận phía trên cùng và nhiều hàng của các mẫu tham chiếu bên trái. Bên cạnh đó, các mẫu tham chiếu lân cận của khối hiện tại có thể bao gồm tổng của nH mẫu gần kề với biên bên phải của khối hiện tại có kích thước $nW \times nH$, tổng của nW mẫu gần kề với biên phía dưới cùng của khối hiện tại, và mẫu gần kề phía dưới cùng-bên phải của khối hiện tại.

Tuy nhiên, một số mẫu tham chiếu lân cận trong các mẫu tham chiếu lân cận của khối hiện tại vẫn chưa được giải mã hoặc có thể không khả dụng. Trong trường hợp này, bộ 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. Thay vào đó, các mẫu tham chiếu lân cận cần được sử dụng cho việc dự đoán có được tạo cấu hình thông qua việc nội suy các mẫu khả dụng.

Khi các mẫu tham chiếu lân cận được dẫn xuất, thì (i) mẫu dự đoán có thể được dẫn xuất dựa trên việc tính trung bình hoặc nội suy các mẫu tham chiếu lân cận của khối hiện tại, hoặc (ii) mẫu dự đoán có thể được dẫn xuất dựa trên mẫu tham chiếu tồn tại 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 có thể được tạo ra thông qua việc nội suy mẫu lân cận thứ nhất được đặt theo hướng dự đoán của chế độ dự đoán trong ảnh của khối hiện tại, dựa trên mẫu dự đoán của khối hiện tại, và mẫu lân cận thứ hai được đặt theo hướng ngược với hướng dự đoán, trong số các mẫu tham chiếu lân cận. Trường hợp được mô tả ở trên có thể được gọi là dự đoán trong ảnh nội suy tuyến tính (Linear Interpolation Intra Prediction, LIP). Bên cạnh đó, các mẫu dự đoán sắc độ có thể được tạo ra dựa trên các mẫu độ chói sử dụng mô hình tuyến tính (Linear Model, LM). Trường hợp này có thể được gọi là chế độ LM hoặc chế độ LM thành phần sắc độ (Chroma Component LM Mode, CCLM).

Bên cạnh đó, mẫu dự đoán thời gian của khối hiện tại được dẫn xuất dựa trên các mẫu tham chiếu lân cận được lọc, và mẫu dự đoán của khối hiện tại có thể cũng được dẫn xuất bằng cách cộng theo trọng số mẫu dự đoán thời gian 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 hiện 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 trong số nhiều đường mẫu tham chiếu của khối hiện tại được chọn, và mẫu dự đoán được dẫn

xuất sử dụng mẫu tham chiếu được đặt theo hướng dự đoán trong đường được chọn. Trong trường hợp này, việc mã hóa dự đoán trong ảnh có thể được thực hiện bằng cách chỉ báo (báo 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 nhiều đường tham chiếu hoặc dự đoán trong ảnh dựa trên MRL.

Bên cạnh đó, khối hiện tại được chia thành các phân vùng con thẳng hoặc ngang và việc dự đoán trong ảnh được thực hiện dựa trên chế độ dự đoán trong ảnh giống nhau, nhưng các mẫu tham chiếu lân cận có thể được dẫn xuất và được sử dụng trong các đơn vị là các phân vùng con. Tức là, trong trường hợp này, chế độ dự đoán trong ảnh cho khối hiện tại được áp dụng ngang bằng cho các phân vùng con, nhưng hiệu suất dự đoán trong ảnh có thể được cải thiện trong một số trường hợp bằng cách dẫn xuất và sử dụng các mẫu tham chiếu lân cận trong các đơn vị là các phân vùng con. Phương pháp dự đoán này có thể được gọi là dự đoán trong ảnh dựa trên các phân vùng con trong ảnh (Intra Sub-Partition, ISP).

Các phương pháp dự đoán trong ảnh được mô tả ở trên có thể được gọi là các loại dự đoán trong ảnh cần được phân biệt với chế độ dự đoán trong ảnh. Các loại dự đoán trong ảnh có thể được đề cập đến bằng các thuật ngữ khác nhau như kỹ thuật dự đoán trong ảnh hoặc các chế độ dự đoán trong ảnh bổ sung. Ví dụ, các loại dự đoán trong ảnh (hoặc các chế độ dự đoán trong ảnh bổ sung, v.v.) có thể bao gồm ít nhất một thành phần trong số LIP, PDPC, MRL, và ISP được đề cập đến ở trên. Phương pháp dự đoán trong ảnh chung ngoại trừ loại dự đoán trong ảnh cụ thể như LIP, PDPC, MRL, và ISP có thể được gọi là loại dự đoán trong ảnh thông thường. Loại dự đoán trong ảnh thông thường có thể được áp dụng chung khi loại dự đoán trong ảnh cụ thể ở trên không được áp dụng, và việc dự đoán có thể được thực hiện dựa trên chế độ dự đoán trong ảnh được mô tả ở

trên. Trong khi đó, nếu cần thiết, việc lọc sau xử lý có thể được thực hiện trên mẫu dự đoán được dẫn xuất.

Cụ thể, quy trình dự đoán trong ảnh có thể bao gồm bước xác định chế độ/loại dự đoán trong ảnh, bước dẫn xuất các mẫu tham chiếu lân cận và bước dẫn xuất mẫu dự đoán dựa trên chế độ/loại dự đoán trong ảnh. Bên cạnh đó, nếu cần thiết, bước sau lọc có thể được thực hiện trên mẫu dự đoán được dẫn xuất.

Khi việc dự đoán trong ảnh được áp dụng, chế độ dự đoán trong ảnh được áp dụng cho khối hiện tại có thể được xác định sử dụng chế độ dự đoán trong ảnh của khối lân cận. Ví dụ, thiết bị giải mã có thể chọn một trong các ứng viên chế độ có thể xảy ra nhất (Most Probable Mode, MPM) trong danh sách MPM được dẫn xuất dựa trên các chế độ ứng viên bổ sung và chế độ dự đoán trong ảnh của khối lân cận (ví dụ, khối lân cận bên trái và/hoặc phía trên cùng) của khối hiện tại, hoặc chọn một trong các chế độ dự đoán trong ảnh còn lại không được bao gồm trong các ứng viên MPM (và chế độ phẳng) dựa trên thông tin chế độ dự đoán trong ảnh còn lại. Danh sách MPM có thể được tạo cấu hình để bao gồm hoặc không bao gồm chế độ phẳng như ứng viên. Ví dụ, khi danh sách MPM bao gồm chế độ phẳng như ứng viên, thì danh sách MPM có thể có 6 ứng viên, và khi danh sách MPM không bao gồm chế độ phẳng như ứng viên, thì danh sách MPM có thể có 5 ứng viên. Khi danh sách MPM không bao gồm chế độ phẳng như ứng viên, thì cờ không phẳng (ví dụ, `intra_luma_not_planar_flag`) biểu diễn liệu chế độ dự đoán trong ảnh của khối hiện tại có phải là chế độ phẳng hay không có thể được báo hiệu. Ví dụ, cờ MPM có thể được báo hiệu trước, và chỉ số MPM và cờ không phẳng có thể được báo hiệu khi giá trị của cờ MPM là 1. Ngoài ra, chỉ số MPM có thể được báo hiệu khi giá trị của cờ không phẳng là 1. Ở đây, thực tế là danh sách MPM được tạo cấu hình để không bao gồm chế độ phẳng như ứng viên là ở chỗ chế độ phẳng luôn được xem xét

như MPM chứ không phải là chế độ phẳng không là MPM, do đó, cờ (cờ không phải là phẳng) được báo hiệu đầu tiên để kiểm tra liệu đó là chế độ phẳng hay không.

Ví dụ, việc liệu chế độ dự đoán trong ảnh có được áp dụng khối hiện tại hay không là nằm trong các ứng viên MPM (và chế độ phẳng) hoặc các chế độ còn lại có thể được chỉ báo dựa trên cờ MPM (ví dụ, `intra_luma_mpm_flag`). Cờ MPM có giá trị là 1 có thể chỉ báo là chế độ dự đoán trong ảnh cho khối hiện tại là trong phạm vi các ứng viên MPM (và chế độ phẳng), và cờ MPM có giá trị là 0 có thể chỉ báo là chế độ dự đoán trong ảnh cho khối hiện tại không là trong phạm vi các ứng viên MPM (và chế độ phẳng). Cờ không phẳng (ví dụ, `intra_luma_not_planar_flag`) có giá trị là 0 có thể chỉ báo là chế độ dự đoán trong ảnh cho khối hiện tại là chế độ phẳng, và cờ không phẳng có giá trị là 1 có thể chỉ báo chế độ dự đoán trong ảnh cho khối hiện tại không là chế độ phẳng. Chỉ số MPM có thể được báo hiệu dưới dạng là phần tử cú pháp `mpm_idx` hoặc `intra_luma_mpm_idx`, và thông tin chế độ dự đoán trong ảnh còn lại có thể được báo hiệu dưới dạng là phần tử cú pháp `rem_intra_luma_pred_mode` hoặc `intra_luma_mpm_remainder`. Ví dụ, thông tin chế độ dự đoán trong ảnh còn lại có thể chỉ báo một trong các chế độ dự đoán trong ảnh còn lại không được bao gồm trong các ứng viên MPM (và chế độ phẳng) trong số tất cả các chế độ dự đoán trong ảnh bằng cách đánh chỉ số theo thứ tự của số chế độ dự đoán. Chế độ dự đoán trong ảnh có thể là chế độ dự đoán trong ảnh cho thành phần (mẫu) độ chói. Ở dưới đây, thông tin chế độ dự đoán trong ảnh có thể bao gồm ít nhất một thành phần trong số cờ MPM (ví dụ, `intra_luma_mpm_flag`), cờ không phẳng (ví dụ, `intra_luma_not_planar_flag`), chỉ số MPM (ví dụ, `mpm_idx` hoặc `intra_luma_mpm_idx`), hoặc thông tin chế độ dự đoán trong ảnh còn lại (`rem_intra_luma_luma_mpm_mode` hoặc `intra_luma_mpminder`). Trong sáng chế này, danh sách MPM có thể được đề cập đến bằng các thuật ngữ khác nhau như danh sách ứng viên MPM và `candModeList`.

Khi MIP được áp dụng cho khối hiện tại, thì cờ MPM riêng biệt (ví dụ, `intra_mip_mpm_flag`) cho MIP, chỉ số mpm (ví dụ, `intra_mip_mpm_idx`), và thông tin chế độ dự đoán trong ảnh còn lại (ví dụ, `intra_mip_mpm_remainder`) có thể được báo hiệu, và cờ không phẳng có thể không được báo hiệu.

Nói cách khác, nói chung là, khi việc phân vùng khối cho ảnh được thực hiện, thì khối hiện tại cần được tạo mã và khối lân cận có các đặc tính ảnh tương tự. Do đó, có xác suất cao là khối hiện tại và khối lân cận có chế độ dự đoán trong ảnh giống hoặc tương tự nhau. Theo đó, bộ mã hóa có thể sử dụng chế độ dự đoán trong ảnh của khối lân cận để mã hóa chế độ dự đoán trong ảnh của khối hiện tại.

Ví dụ, thiết bị giải mã/thiết bị mã hóa có thể tạo dựng danh sách các chế độ có thể xảy ra nhất (MPM) cho khối hiện tại. Danh sách MPM có thể được gọi là danh sách ứng viên MPM. Ở đây, MPM có thể đề cập đến các chế độ được sử dụng để cải thiện hiệu quả tạo mã có sự xem xét đến sự tương tự giữa khối hiện tại và các khối lân cận trong khi tạo mã chế độ dự đoán trong ảnh. Như được mô tả ở trên, danh sách MPM có thể được tạo dựng để bao gồm chế độ phẳng, hoặc có thể được tạo dựng để loại trừ chế độ phẳng. Ví dụ, khi danh sách MPM bao gồm chế độ phẳng, thì số lượng ứng viên trong danh sách MPM có thể là sáu. Và, khi danh sách MPM không bao gồm chế độ phẳng, thì số ứng viên trong danh sách MPM có thể là 5.

Bộ mã hóa/bộ giải mã có thể tạo dựng danh sách MPM bao gồm năm hoặc sáu MPM.

Để tạo dựng danh sách MPM, ba loại chế độ, như các chế độ trong ảnh mặc định, các chế độ trong ảnh lân cận, và các chế độ trong ảnh được dẫn xuất, có thể được xem xét.

Đối với các chế độ trong ảnh lân cận, thì hai khối lân cận, tức là, khối lân cận bên trái và khối lân cận phía trên cùng, có thể được xem xét.

Như được mô tả ở trên, nếu danh sách MPM được tạo dựng để không bao gồm chế độ phẳng, thì chế độ phẳng có thể được loại trừ khỏi danh sách, và số ứng viên của danh sách MPM có thể được thiết đặt là năm.

Thêm nữa, chế độ không hướng (hoặc chế độ không góc) trong số các chế độ dự đoán trong ảnh có thể bao gồm chế độ DC dựa trên mức trung bình của các mẫu tham chiếu lân cận của khối hiện tại hoặc chế độ phẳng dựa vào nội suy.

Trong khi đó, khi việc dự đoán liên ảnh được áp dụng, thì bộ dự đoán của thiết bị mã hóa/thiết bị giải mã có thể dẫn xuất các mẫu dự đoán bằng cách thực hiện việc dự đoán liên ảnh trong các đơn vị là các khối. Việc dự đoán liên ảnh có thể được áp dụng khi thực hiện việc dự đoán trên khối hiện tại. Tức là, bộ dự đoán (cụ thể hơn, bộ dự đoán liên ảnh) của thiết bị mã hóa/giải mã có thể dẫn xuất các mẫu dự đoán bằng cách thực hiện việc dự đoán liên ảnh trong các đơn vị là khối. Việc dự đoán liên ảnh có thể biểu diễn rằng dự đoán được dẫn xuất bằng phương pháp phụ thuộc vào các phần tử dữ liệu (ví dụ, các giá trị mẫu hoặc thông tin chuyển động) của hình ảnh (các hình ảnh) ngoài hình ảnh hiện tại. Khi việc dự đoán liên ảnh được áp dụng cho khối hiện tại, thì khối được dự đoán (mảng mẫu dự đoán) cho khối hiện tại có thể được dẫn xuất dựa trên khối tham chiếu (mảng mẫu tham chiếu) được chỉ rõ bằng vectơ chuyển động trên hình ảnh tham chiếu được chỉ báo bằng chỉ số hình ảnh tham chiếu. Trong trường hợp này, để giảm lượng thông tin chuyển động được truyền trong chế độ dự đoán liên ảnh, thông tin chuyển động của khối hiện tại có thể được dự đoán trong các đơn vị là khối, khối con, hoặc mẫu dựa trên sự tương quan của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể bao gồm vectơ chuyển động và chỉ số hình ảnh

tham chiếu. Thông tin chuyển động có thể còn bao gồm thông tin loại dự đoán liên ảnh (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Trong trường hợp áp dụng việc dự đoán liên ảnh, khối lân cận có thể bao gồm khối lân cận không gian mà có mặt trong hình ảnh hiện tại và khối lân cận thời gian mà có mặt trong hình ảnh tham chiếu. Hình ảnh tham chiếu bao gồm khối tham chiếu và hình ảnh tham chiếu bao gồm khối lân cận thời gian có thể là giống nhau hoặc khác với nhau. Khối lân cận thời gian có thể được gọi là tên là khối tham chiếu được đặt cùng vị trí, CU được đặt cùng vị trí (colCU), v.v., và hình ảnh tham chiếu bao gồm khối lân cận thời gian có thể được gọi là hình ảnh được đặt cùng vị trí (colPic). Ví dụ, danh sách ứng viên thông tin chuyển động có thể được tạo cấu hình dựa trên các khối lân cận của khối hiện tại và cờ hoặc thông tin chỉ số chỉ báo ứng viên nào được chọn (được sử dụng) có thể được báo hiệu để dẫn xuất vectơ chuyển động và/hoặc chỉ số hình 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 và ví dụ, trong trường hợp của chế độ bỏ qua và chế độ hợp nhất, thông tin chuyển động của khối hiện tại có thể là giống như thông tin chuyển động của khối lân cận được chọn. Trong trường hợp của chế độ bỏ qua, tín hiệu phần dư có thể không được truyền giống như chế độ hợp nhất. Trong trường hợp của chế độ dự đoán vectơ chuyển động (MVP), vectơ chuyển động của khối lân cận được chọn có thể được sử dụng như bộ dự đoán vectơ chuyển động và mức chênh lệch vectơ chuyển động có thể được báo hiệu. Trong trường hợp này, vectơ chuyển động của khối hiện tại có thể được dẫn xuất bằng việc sử dụng tổng số của bộ dự đoán vectơ chuyển động và mức chênh lệch vectơ chuyển động.

Thông tin chuyển động có thể còn bao gồm thông tin chuyển động L0 và/hoặc thông tin chuyển động L1 theo loại dự đoán liên ảnh (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Vectơ chuyển động hướng L0 có thể được gọi là vectơ chuyển động L0 hoặc MVL0 và vectơ chuyển động hướng L1 có thể được gọi là vectơ chuyển động L1 hoặc

MVL1. Việc dự đoán dựa trên vectơ chuyển động L0 có thể được gọi là dự đoán L0, dự đoán dựa trên vectơ chuyển động L1 có thể được gọi là dự đoán L1, và dự đoán dựa trên cả vectơ chuyển động L0 và vectơ chuyển động L1 có thể được gọi là song dự đoán. Ở đây, vectơ chuyển động L0 có thể chỉ báo vectơ chuyển động liên quan đến danh sách hình ảnh tham chiếu L0 và vectơ chuyển động L1 có thể chỉ báo vectơ chuyển động liên quan đến danh sách hình ảnh tham chiếu L1. Danh sách hình ảnh tham chiếu L0 có thể bao gồm các hình ảnh trước hình ảnh hiện tại theo thứ tự được xuất ra và danh sách hình ảnh tham chiếu L1 có thể bao gồm các hình ảnh kế tiếp với hình ảnh hiện tại theo thứ tự được xuất ra, như các hình ảnh tham chiếu. Các hình ảnh trước có thể được gọi là hình ảnh (tham chiếu) xuôi và các hình ảnh kế tiếp có thể được gọi là hình ảnh (tham chiếu) ngược. Danh sách hình ảnh tham chiếu L0 có thể còn bao gồm các hình ảnh kế tiếp hình ảnh hiện tại theo thứ tự được xuất ra như các hình ảnh tham chiếu. Trong trường hợp này, các hình ảnh trước có thể được đánh chỉ số đầu tiên trong danh sách hình ảnh tham chiếu L0 và sau đó các hình ảnh kế tiếp có thể được lập chỉ số. Danh sách hình ảnh tham chiếu L1 có thể còn bao gồm các hình ảnh trước hình ảnh hiện tại theo thứ tự được xuất ra như các hình ảnh tham chiếu. Trong trường hợp này, các hình ảnh kế tiếp có thể được đánh chỉ số đầu tiên trong danh sách hình ảnh tham chiếu L1 và sau đó các hình ảnh kế tiếp trước được lập chỉ số. Ở đây, thứ tự xuất ra có thể tương ứng với thứ tự đếm thứ tự hình ảnh (Picture Order Count, POC).

Quy trình mã hóa video/ảnh dựa trên việc dự đoán liên ảnh có thể bao gồm theo cách giản lược, ví dụ, như sau.

Fig.4 minh họa ví dụ về phương pháp mã hóa video/ảnh dựa trên việc dự đoán liên ảnh.

Thiết bị mã hóa thực hiện việc dự đoán liên ảnh cho khối hiện tại (S400). Thiết bị mã hóa có thể dẫn xuất chế độ dự đoán liên ảnh và thông tin chuyển động của khối hiện tại và tạo ra các mẫu dự đoán của khối hiện tại. Ở đây, quy trình xác định chế độ dự đoán liên ảnh, quy trình dẫn xuất thông tin chuyển động, và quy trình tạo ra các mẫu dự đoán có thể được thực hiện đồng thời và một quy trình bất kỳ có thể được thực hiện sớm hơn quy trình khác. Ví dụ, đơn vị dự đoán liên ảnh của thiết bị mã hóa có thể bao gồm đơn vị xác định chế độ dự đoán, đơn vị dẫn xuất thông tin chuyển động, và đơn vị dẫn xuất mẫu dự đoán, và đơn vị xác định chế độ dự đoán có thể xác định chế độ dự đoán cho khối hiện tại, đơn vị dẫn xuất thông tin chuyển động có thể dẫn xuất thông tin chuyển động của khối hiện tại, và đơn vị dẫn xuất mẫu dự đoán có thể dẫn xuất các mẫu dự đoán của khối hiện tại. Ví dụ, đơn vị dự đoán liên ảnh của thiết bị mã hóa có thể tìm kiếm khối tương tự với khối hiện tại trong khu vực định trước (khu vực tìm kiếm) các hình ảnh tham chiếu thông qua việc ước lượng chuyển động và dẫn xuất khối tham chiếu trong đó mức chênh lệch so với khối hiện tại là tối thiểu hoặc là bằng hoặc nhỏ hơn tiêu chí định trước. Chỉ số hình ảnh tham chiếu chỉ báo hình ảnh tham chiếu mà khối tham chiếu được định vị tại đó có thể được dẫn xuất dựa trên đó và vectơ chuyển động có thể được dẫn xuất dựa trên mức chênh lệch về vị trí giữa khối tham chiếu và khối hiện tại. Thiết bị mã hóa có thể xác định chế độ được áp dụng cho khối hiện tại trong số các chế độ dự đoán khác nhau. Thiết bị mã hóa có thể so sánh chi phí RD cho các chế độ dự đoán khác nhau và xác định chế độ dự đoán tối ưu cho khối hiện tại.

Ví dụ, khi chế độ bỏ qua hoặc chế độ hợp nhất được áp dụng cho khối hiện tại, thì thiết bị mã hóa có thể tạo cấu hình danh sách ứng viên hợp nhất sẽ được mô tả dưới đây và dẫn xuất khối tham chiếu trong đó mức chênh lệch so với khối hiện tại là tối thiểu hoặc bằng hoặc nhỏ hơn tiêu chí định trước trong số các khối tham chiếu được chỉ báo bởi các ứng viên hợp nhất được bao gồm trong danh sách ứng viên hợp nhất. Trong

trường hợp này, ứng viên hợp nhất liên quan đến khối tham chiếu được dẫn xuất có thể được chọn và thông tin chỉ số hợp nhất chỉ báo ứng viên hợp nhất được chọn có thể được tạo ra và được báo hiệu đến thiết bị giải mã. Thông tin chuyển động của khối hiện tại có thể được dẫn xuất bằng việc sử dụng thông tin chuyển động của ứng viên hợp nhất được chọn.

Theo ví dụ khác, khi chế độ (A)MVP được áp dụng cho khối hiện tại, thì thiết bị mã hóa có thể tạo cấu hình danh sách ứng viên (A)MVP sẽ được mô tả sau và sử dụng vectơ chuyển động của ứng viên mvp được chọn trong số các ứng viên của bộ dự đoán vectơ chuyển động (mvp) được bao gồm trong danh sách ứng viên (A)MVP như mvp của khối hiện tại. Trong trường hợp này, ví dụ, vectơ chuyển động chỉ báo khối tham chiếu được dẫn xuất bằng việc ước lượng chuyển động có thể được sử dụng như vectơ chuyển động của khối hiện tại, và ứng viên mvp có vectơ chuyển động với mức chênh lệch nhỏ nhất so với vectơ chuyển động của khối hiện tại trong số các ứng viên mvp có thể trở thành ứng viên mvp được chọn. Mức chênh lệch vectơ chuyển động (Motion Vector Difference, MVD) mà là mức chênh lệch thu được bằng cách lấy vectơ chuyển động của khối hiện tại trừ đi mvp có thể được dẫn xuất. Trong trường hợp này, thông tin về MVD có thể được báo hiệu đến thiết bị giải mã. Thêm nữa, khi chế độ (A)MVP được áp dụng, thì giá trị của chỉ số hình ảnh tham chiếu có thể được tạo cấu hình như thông tin chỉ số hình ảnh tham chiếu và được báo hiệu riêng biệt đến thiết bị giải mã.

Thiết bị mã hóa có thể dẫn xuất các mẫu dự dựa trên các mẫu được dự đoán (S410). Thiết bị mã hóa có thể dẫn xuất các mẫu dự bằng cách so sánh các mẫu gốc và các mẫu dự đoán của khối hiện tại.

Thiết bị mã hóa mã hóa thông tin ảnh bao gồm thông tin dự đoán và thông tin phần dư (S420). Thiết bị mã hóa có thể xuất ra thông tin ảnh được mã hóa trong dạng luồng

bit. Thông tin dự đoán có thể bao gồm thông tin về thông tin chế độ dự đoán (ví dụ, cờ bỏ qua, cờ hợp nhất hoặc chỉ số chế độ, v.v.) và thông tin về thông tin chuyển động như thông tin liên quan đến thủ tục dự đoán. Thông tin về thông tin chuyển động có thể bao gồm thông tin chọn ứng viên (ví dụ, chỉ số hợp nhất, cờ mvp hoặc chỉ số mvp) mà là thông tin để dẫn xuất vectơ chuyển động. Thêm nữa, thông tin về thông tin chuyển động có thể bao gồm thông tin về MVD và/hoặc thông tin chỉ số hình ảnh tham chiếu. Thêm nữa, thông tin về thông tin chuyển động có thể bao gồm thông tin chỉ báo liệu để áp dụng việc dự đoán L0, dự đoán L1, hay song dự đoán. 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.

Luồng bit được xuất ra có thể được lưu trữ trong phương tiện lưu trữ (số) và được chuyển đến thiết bị giải mã hoặc được chuyển đến thiết bị giải mã thông qua mạng.

Trong khi đó, như được mô tả ở trên, thiết bị mã hóa có thể tạo ra hình ảnh được tái tạo (bao gồm các mẫu được tái tạo và các khối được tái tạo) dựa trên các mẫu tham chiếu và các mẫu dư. Điều này là để dẫn xuất kết quả dự đoán giống như kết quả dự đoán được thực hiện bởi thiết bị giải mã, và theo đó, hiệu quả tạo mã có thể được tăng lên. Theo đó, thiết bị mã hóa có thể lưu trữ hình ảnh tái tạo (hoặc các mẫu tái tạo hoặc các khối tái tạo) trong bộ nhớ và sử dụng hình ảnh tái tạo như hình ảnh tham chiếu. Quy trình lọc trong vòng có thể còn được áp dụng cho hình ảnh tái tạo như được mô tả ở trên.

Quy trình giải mã video/ảnh dựa trên việc dự đoán liên ảnh có thể bao gồm theo cách giản lược, ví dụ, như sau.

Fig.5 minh họa ví dụ về phương pháp giải mã video/ảnh dựa trên việc dự đoán liên ảnh.

Tham chiếu đến Fig.5, thiết bị giải mã có thể thực hiện hoạt động vận hành tương ứng với hoạt động vận hành được thực hiện bởi thiết bị mã hóa. Thiết bị giải mã có thể thực hiện việc dự đoán cho khối hiện tại dựa trên thông tin dự đoán nhận được và dẫn xuất các mẫu dự đoán.

Cụ thể, thiết bị giải mã có thể xác định chế độ dự đoán cho khối hiện tại dựa trên thông tin dự đoán nhận được (S500). Thiết bị giải mã có thể xác định chế độ dự đoán liên ảnh nào được áp dụng cho khối hiện tại dựa trên thông tin chế độ dự đoán trong thông tin dự đoán.

Ví dụ, có thể xác định được liệu chế độ hợp nhất hay chế độ (A)MVP được áp dụng cho khối hiện tại dựa trên cờ hợp nhất. Thay vào đó, một trong số các ứng viên chế độ dự đoán liên ảnh có thể được chọn dựa trên chỉ số chế độ. Các ứng viên chế độ dự đoán liên ảnh có thể bao gồm chế độ bỏ qua, chế độ hợp nhất và/hoặc chế độ (A)MVP hoặc có thể bao gồm các chế độ dự đoán liên ảnh khác nhau mà được mô tả dưới đây.

Thiết bị giải mã dẫn xuất thông tin chuyển động của khối hiện tại dựa trên chế độ dự đoán liên ảnh được xác định (S510). Ví dụ, khi chế độ bỏ qua hoặc chế độ hợp nhất được áp dụng cho khối hiện tại, thì thiết bị giải mã có thể tạo cấu hình danh sách ứng viên hợp nhất được mô tả dưới đây và chọn một ứng viên hợp nhất trong số các ứng viên hợp nhất được bao gồm trong danh sách ứng viên hợp nhất. Ở đây, việc chọn có thể được thực hiện dựa trên thông tin chọn (chỉ số hợp nhất). Thông tin chuyển động của khối hiện tại có thể được dẫn xuất bằng việc sử dụng thông tin chuyển động của ứng viên hợp nhất được chọn. Thông tin chuyển động của ứng viên hợp nhất được chọn có thể được sử dụng như thông tin chuyển động của khối hiện tại.

Theo ví dụ khác, khi chế độ (A)MVP được áp dụng cho khối hiện tại, thì thiết bị giải mã có thể tạo cấu hình danh sách ứng viên (A)MVP sẽ được mô tả sau và sử dụng

véc tơ chuyển động của ứng viên mvp được chọn trong số các ứng viên của bộ dự đoán véctơ chuyển động (mvp) được bao gồm trong danh sách ứng viên (A)MVP như mvp của khối hiện tại. Ở đây, việc chọn có thể được thực hiện dựa trên thông tin chọn (cờ mvp hoặc chỉ số mvp). Trong trường hợp này, MVD của khối hiện tại có thể được dẫn xuất dựa trên thông tin về MVD, và véctơ chuyển động của khối hiện tại có thể được dẫn xuất dựa trên mvp của khối hiện tại và MVD. Thêm nữa, chỉ số hình ảnh tham chiếu của khối hiện tại có thể được dẫn xuất dựa trên thông tin chỉ số hình ảnh tham chiếu. Hình ảnh được chỉ báo bởi chỉ số hình ảnh tham chiếu trong danh sách hình ảnh tham chiếu cho khối hiện tại có thể được dẫn xuất như hình ảnh tham chiếu được tham chiếu cho việc dự đoán liên ảnh khối hiện tại.

Trong khi đó, như được mô tả dưới đây, thông tin chuyển động của khối hiện tại có thể được dẫn xuất mà không cần hình danh sách ứng viên và trong trường hợp này, thông tin chuyển động của khối hiện tại có thể được dẫn xuất theo thủ tục được bộc lộ trong chế độ dự đoán. Trong trường hợp này, cấu hình danh sách ứng viên có thể được lược bỏ.

Thiết bị giải mã có thể tạo ra các mẫu dự đoán cho khối hiện tại dựa trên thông tin chuyển động của khối hiện tại (S520). Trong trường hợp này, hình ảnh tham chiếu có thể được dẫn xuất dựa trên chỉ số hình ảnh tham chiếu của khối hiện tại, và các mẫu dự đoán của khối hiện tại có thể được dẫn xuất sử dụng các mẫu của khối tham chiếu được chỉ báo bởi véctơ chuyển động của khối hiện tại trên hình ảnh tham chiếu. Trong trường hợp này, ở một số trường hợp, thủ tục lọc mẫu được dự đoán cho tất cả hoặc một số mẫu dự đoán của khối hiện tại có thể còn được thực hiện.

Ví dụ, đơn vị dự đoán liên ảnh của thiết bị giải mã có thể bao gồm đơn vị xác định chế độ dự đoán, đơn vị dẫn xuất thông tin chuyển động, và đơn vị dẫn xuất mẫu dự đoán,

và đơn vị xác định chế độ dự đoán có thể xác định chế độ dự đoán cho khối hiện tại dựa trên thông tin chế độ dự đoán nhận được, đơn vị dẫn xuất thông tin chuyển động có thể dẫn xuất thông tin chuyển động (véctor chuyển động và/hoặc chỉ số hình ảnh tham chiếu) của khối hiện tại dựa trên thông tin về thông tin chuyển động nhận được, và đơn vị dẫn xuất mẫu dự đoán có thể dẫn xuất các mẫu được dự đoán của khối hiện tại.

Thiết bị giải mã tạo ra các mẫu dự cho khối hiện tại dựa trên thông tin dự nhận được (S530). Thiết bị giải mã có thể tạo ra các mẫu tái tạo cho khối hiện tại dựa trên các mẫu dự đoán và các mẫu dự và tạo ra hình ảnh tái tạo dựa trên các mẫu tái tạo được tạo ra (S540). Sau đó, thủ tục lọc trong vòng có thể còn được áp dụng cho hình ảnh tái tạo như được mô tả ở trên.

Fig.6 là hình vẽ thể hiện sơ lược thủ tục dự đoán liên ảnh.

Tham chiếu đến Fig.6, như được mô tả ở trên, quy trình dự đoán liên ảnh có thể bao gồm bước xác định chế độ dự đoán liên ảnh, bước dẫn xuất thông tin chuyển động theo chế độ dự đoán được xác định, và bước xử lý dự đoán (tạo ra mẫu dự đoán) dựa trên thông tin chuyển động được dẫn xuất. Quy trình dự đoán liên ảnh có thể được thực hiện bởi thiết bị mã hóa và thiết bị giải mã như được mô tả ở trên. Trong sáng chế này, thiết bị tạo mã có thể bao gồm thiết bị mã hóa và/hoặc thiết bị giải mã.

Tham chiếu đến Fig.6, thiết bị tạo mã xác định chế độ dự đoán liên ảnh cho khối hiện tại (S600). Các chế độ dự đoán liên ảnh khác nhau có thể được sử dụng cho việc dự đoán khối hiện tại trong hình ảnh. Ví dụ, các chế độ khác nhau, như chế độ hợp nhất, chế độ bỏ qua, chế độ dự đoán véctor chuyển động (MVP), chế độ afin, chế độ hợp nhất khối con, chế độ hợp nhất với MVD (Merge With MVD, MMVD), và chế độ dự đoán véctor chuyển động theo lịch sử (Historical Motion Vector Prediction, HMVP) có thể được sử dụng. Chế độ tinh chỉnh véctor chuyển động phía bộ giải mã (Decoder Side

Motion Vector Refinement, DMVR), chế độ phân giải vectơ chuyển động thích ứng (Adaptive Motion Vector Resolution, AMVR), song dự đoán với trọng số cấp CU (Bi-Prediction With CU-Level Weight, BCW), dòng quang song hướng (Bi-Directional Optical Flow, BDOF), và tương tự có thể còn được sử dụng như các chế độ bổ sung. Chế độ afin có thể cũng được gọi là chế độ dự đoán chuyển động afin. Chế độ MVP có thể cũng được gọi là chế độ dự đoán vectơ chuyển động cải tiến (Advanced Motion Vector Prediction, AMVP). Trong sáng chế này, một số chế độ và/hoặc ứng viên thông tin chuyển động được dẫn xuất bằng một số chế độ có thể cũng được bao gồm trong một trong các ứng viên liên quan đến thông tin chuyển động trong các chế độ khác. Ví dụ, ứng viên HMVP có thể được thêm vào vào ứng viên hợp nhất của các chế độ hợp nhất/bỏ qua, hoặc cũng được thêm vào ứng viên.mvp của chế độ MVP. Nếu ứng viên HMVP được sử dụng như ứng viên thông tin chuyển động của chế độ hợp nhất hoặc chế độ bỏ qua, thì ứng viên HMVP có thể được gọi là ứng viên hợp nhất HMVP.

Thông tin chế độ dự đoán chỉ báo chế độ dự đoán liên ảnh của khối hiện tại có thể được báo hiệu từ thiết bị mã hóa đến thiết bị giải mã. Trong trường hợp này, thông tin chế độ dự đoán có thể được bao gồm trong luồng bit và được nhận bởi thiết bị giải mã. Thông tin chế độ dự đoán có thể bao gồm thông tin chỉ số chỉ báo một trong nhiều chế độ ứng viên. Thay vào đó, chế độ dự đoán liên ảnh có thể được chỉ báo thông qua việc báo hiệu phân cấp thông tin cờ. Trong trường hợp này, thông tin chế độ dự đoán có thể bao gồm một hoặc nhiều cờ. Ví dụ, việc liệu có áp dụng chế độ bỏ qua hay không có thể được chỉ báo bằng cách báo hiệu cờ bỏ qua, việc liệu có áp dụng chế độ hợp nhất hay không có thể được chỉ báo bằng cách báo hiệu cờ hợp nhất khi chế độ bỏ qua không được áp dụng, và được chỉ báo rằng chế độ MVP được áp dụng hoặc cờ để phân biệt thêm có thể còn được báo hiệu khi chế độ hợp nhất không được áp dụng. Chế độ afin có thể được báo hiệu như chế độ độc lập hoặc được báo hiệu như chế độ phụ thuộc vào chế

độ hợp nhất hoặc chế độ MVP. Ví dụ, chế độ afin có thể bao gồm chế độ hợp nhất afin và chế độ MVP afin.

Thiết bị tạo mã dẫn xuất thông tin chuyển động cho khối hiện tại (S610). Việc dẫn xuất thông tin chuyển động có thể được dẫn xuất dựa trên chế độ dự đoán liên ảnh.

Thiết bị tạo mã có thể thực hiện việc dự đoán liên ảnh sử dụng thông tin chuyển động của khối hiện tại. Thiết bị mã hóa có thể dẫn xuất thông tin chuyển động tối ưu cho khối hiện tại thông qua thủ tục ước lượng chuyển động. Ví dụ, thiết bị mã hóa có thể tìm kiếm khối tham chiếu tương tự có sự tương quan cao trong các đơn vị là điểm ảnh phân đoạn trong phạm vi tìm kiếm định trước trong hình ảnh tham chiếu bằng cách sử dụng khối gốc trong hình ảnh gốc cho khối hiện tại và dẫn xuất thông tin chuyển động thông qua khối tham chiếu được tìm kiếm. Sự tương tự của khối có thể được dẫn xuất dựa trên mức chênh lệch của các giá trị mẫu dựa trên pha. Ví dụ, sự tương tự của khối có thể được tính toán dựa trên tổng của các mức chênh lệch tuyệt đối (Sum of Absolute Differences, SAD) giữa khối hiện tại (hoặc khuôn mẫu của khối hiện tại) và khối tham chiếu (hoặc khuôn mẫu của khối tham chiếu). Trong trường hợp này, thông tin chuyển động có thể được dẫn xuất dựa trên khối tham chiếu có SAD nhỏ nhất trong khu vực tìm kiếm. Thông tin chuyển động được dẫn xuất có thể được báo hiệu đến thiết bị giải mã theo các phương pháp khác nhau dựa trên chế độ dự đoán liên ảnh.

Thiết bị tạo mã thực hiện việc dự đoán liên ảnh dựa trên thông tin chuyển động cho khối hiện tại (S620). Thiết bị lập mã có thể dẫn xuất mẫu (các mẫu) dự đoán cho khối hiện tại dựa trên thông tin chuyển động. Khối hiện tại bao gồm các mẫu dự đoán có thể được gọi là khối được dự đoán.

Trong khi đó, như được mô tả ở trên, bộ lượng tử hóa của thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng cách áp dụng việc lượng tử hóa cho

các hệ số biến đổi. Bộ khử lượng tử hóa của thiết bị mã hóa 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 hóa.

Nhìn chung, trong việc tạo mã video/ảnh, tỷ lệ lượng tử hóa có thể được thay đổi, và tốc độ nén có thể được sửa đổi sử dụng tỷ lệ lượng tử hóa được thay đổi này. Trong khía cạnh thực hiện, thông số lượng tử hóa (Quantization Parameter, QP) có thể được sử dụng thay vì trực tiếp sử dụng tỷ lệ lượng tử hóa khi xét đến độ phức tạp. 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 sự. Thêm nữa, ví dụ, thông số lượng tử hóa QPY cho thành phần độ chói và thông số lượng tử hóa QPC cho thành phần sắc độ có thể được tạo cấu hình khác nhau.

Trong quy trình lượng tử hóa, hệ số biến đổi C có thể là đầu vào, tỷ lệ lượng tử hóa (Q_{step}) có thể được chia, và hệ số biến đổi được lượng tử hóa C' có thể thu được dựa trên tỷ lệ lượng tử hóa. Trong trường hợp này, tỷ lệ lượng tử hóa có thể được tạo ra dưới dạng số nguyên bằng cách nhân tỷ lệ lượng tử hóa với một thang định cỡ bằng việc xét đến độ phức tạp tính toán, và hoạt động vận hành luân phiên có thể được thực hiện bằng giá trị tương ứng với giá trị thang định cỡ. Thang định cỡ lượng tử hóa có thể được dẫn xuất dựa trên tích số của tỷ lệ lượng tử hóa và giá trị thang định cỡ. Tức là, thang định cỡ lượng tử hóa có thể được dẫn xuất dựa trên QP. Ví dụ, thang định cỡ lượng tử hóa có thể được áp dụng cho hệ số biến đổi C' , và hệ số biến đổi được lượng tử hóa C' có thể được dẫn xuất dựa trên kết quả áp dụng.

Quy trình khử lượng tử hóa là quy trình ngược của quy trình lượng tử hóa. Trong quy trình này, hệ số biến đổi được lượng tử hóa C' có thể được nhân với tỷ lệ lượng tử hóa (Q_{step}), và hệ số biến đổi được tái tạo C'' có thể thu được dựa trên kết quả của phép

nhân. Trong trường hợp này, thang định cỡ cấp độ có thể được dẫn xuất dựa trên thông số lượng tử hóa, thang định cỡ cấp độ có thể được áp dụng cho hệ số biến đổi được lượng tử hóa C' , và hệ số biến đổi được tái tạo C'' có thể được dẫn xuất. Hệ số biến đổi được tái tạo C'' có thể có một số chênh lệch so với hệ số biến đổi thứ nhất C do sự suy hao trong quy trình biến đổi và/hoặc lượng tử hóa. Theo đó, việc khử lượng tử hóa được thực hiện trong thiết bị mã hóa như trong thiết bị giải mã.

Trong khi đó, công nghệ lượng tử hóa đo tần số thích ứng để sửa đổi độ mạnh lượng tử hóa phụ thuộc vào tần số có thể được áp dụng. Công nghệ lượng tử hóa đo tần số thích ứng là phương pháp áp dụng độ mạnh lượng tử hóa theo cách khác nhau cho mỗi tần số. Trong khi lượng tử hóa đo tần số thích ứng, độ mạnh lượng tử hóa cho mỗi tần số có thể được áp dụng theo cách khác nhau nhờ sử dụng ma trận định cỡ lượng tử hóa được định rõ trước. Tức là, quy trình lượng tử hóa/khử lượng tử hóa được đề cập ở trên có thể được thực hiện dựa trên ma trận định cỡ lượng tử hóa. Ví dụ, để tạo ra kích thước của khối hiện tại và/hoặc tín hiệu phần dư của khối hiện tại, ma trận định cỡ lượng tử hóa khác nhau có thể được sử dụng phụ thuộc vào liệu chế độ dự đoán được áp dụng cho khối hiện tại là dự đoán liên ảnh hay 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. Thêm nữa, đối với việc định cỡ thích ứng tần số, thông tin thang định cỡ lượng tử hóa cho mỗi tần số cho ma trận định cỡ lượng tử hóa có thể được tạo dựng/được mã hóa trong thiết bị mã hóa và được báo hiệu đến thiết bị giải mã. Thông tin thang định cỡ lượng tử hóa cho mỗi tần số có thể được gọi là thông tin định cỡ lượng tử hóa. Thông tin thang định cỡ lượng tử hóa cho mỗi tần số có thể bao 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 dữ liệu danh sách định cỡ. Thêm nữa, thông tin thang định cỡ lượng tử hóa cho mỗi tần số có thể bao gồm thông tin cờ biểu diễn liệu dữ liệu

danh sách định cỡ là có mặt hay không. Thay vào đó, nếu dữ liệu danh sách định cỡ được báo hiệu trong cấp cao hơn (ví dụ, SPS), thì thông tin biểu diễn liệu dữ liệu danh sách định cỡ được sửa đổi trong cấp thấp hơn (ví dụ, PPS hoặc phần đầu nhóm phiên, v.v.) của cấp cao hơn, v.v. có thể còn được bao gồm.

Như trong các nội dung được đề cập đến ở trên, việc lượng tử hóa/khử lượng tử hóa có thể được áp dụng cho thành phần độ chói và thành phần sắc độ dựa trên thông số lượng tử hóa.

Các thông số lượng tử hóa cho đơn vị tạo mã có thể được xác định dựa trên thông tin được báo hiệu trong hình ảnh và/hoặc cấp lát. Ví dụ, các thông số lượng tử hóa có thể được dẫn xuất như trong các nội dung được mô tả sau đây.

Ví dụ, thông tin liên quan đến việc dẫn xuất các thông số lượng tử hóa có thể được báo hiệu như trong bảng sau đây thông qua tập hợp thông số chuỗi (SPS).

【Bảng 1】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>bit_depth_luma_minus8</code>	<code>uc(v)</code>
<code>bit_depth_chroma_minus8</code>	<code>uc(v)</code>
<code>...</code>	
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Các ngữ nghĩa của các phân tử cú pháp trong bảng 1 có thể là giống như bảng sau đây.

【Bảng 2】

bit_depth_luma_minus8 specifies the bit depth of the samples of the luma array BitDepth_Y and the value of the luma quantization parameter range offset QpBdOffset_Y as follows:

$$\text{BitDepth}_Y = 8 + \text{bit_depth_luma_minus8}$$

$$\text{QpBdOffset}_Y = 6 * \text{bit_depth_luma_minus8}$$

$\text{bit_depth_luma_minus8}$ shall be in the range of 0 to 8, inclusive.

bit_depth_chroma_minus8 specifies the bit depth of the samples of the chroma arrays BitDepth_C and the value of the chroma quantization parameter range offset QpBdOffset_C as follows:

$$\text{BitDepth}_C = 8 + \text{bit_depth_chroma_minus8}$$

$$\text{QpBdOffset}_C = 6 * \text{bit_depth_chroma_minus8}$$

$\text{bit_depth_chroma_minus8}$ shall be in the range of 0 to 8, inclusive.

Ví dụ, phần tử cú pháp $\text{bit_depth_luma_minus8}$ có thể biểu diễn BitDepth_Y , tức là, chiều sâu bit của các mẫu của mảng độ chói, và QpBdOffset_Y mà là độ bù ngưỡng thông số lượng tử hóa độ chói. Tức là, ví dụ, BitDepth_Y và QpBdOffset_Y có thể được dẫn xuất dựa trên phần tử cú pháp $\text{bit_depth_luma_minus8}$. Ví dụ, BitDepth_Y có thể được dẫn xuất như giá trị thu được bằng cách cộng 8 vào giá trị của phần tử cú pháp $\text{bit_depth_luma_minus8}$. QpBdOffset_Y có thể được dẫn xuất như giá trị thu được bằng cách nhân giá trị của phần tử cú pháp $\text{bit_depth_luma_minus8}$ với 6. Thêm nữa, $\text{bit_depth_luma_minus8}$ có thể là trong phạm vi từ 0 đến 8.

Thêm nữa, ví dụ, phần tử cú pháp $\text{bit_depth_chroma_minus8}$ có thể biểu diễn BitDepth_C , tức là, chiều sâu bit của các mẫu của mảng sắc độ, và QpBdOffset_C , tức là, độ bù phạm vi thông số lượng tử hóa sắc độ. Tức là, ví dụ, BitDepth_C và QpBdOffset_C có thể được dẫn xuất dựa trên phần tử cú pháp $\text{bit_depth_chroma_minus8}$. Ví dụ,

BitDepthc có thể được dẫn xuất như giá trị thu được bằng cách cộng 8 vào giá trị của phần tử cú pháp `bit_depth_chroma_minus8`. `QpBdOffsetc` có thể được dẫn xuất như giá trị thu được bằng cách nhân giá trị của phần tử cú pháp `bit_depth_chroma_minus8` với 6. Thêm nữa, `bit_depth_chroma_minus8` có thể là trong phạm vi từ 0 đến 8.

Thêm nữa, thông tin liên quan đến việc dẫn xuất các thông số lượng tử hóa có thể được báo hiệu như trong bảng sau đây, ví dụ, thông qua tập hợp thông số hình ảnh (hình ảnh parameter set, PPS). Thông tin có thể bao gồm độ bù sắc độ Cb, độ bù sắc độ Cr, độ bù sắc độ kết hợp, và thông số lượng tử hóa ban đầu. Tức là, thông tin có thể bao gồm các phần tử cú pháp cho độ bù sắc độ Cb, độ bù sắc độ Cr, độ bù sắc độ kết hợp, và thông số lượng tử hóa ban đầu.

【Bảng 3】

<code>pic_parameter_set_rbsp() {</code>	Descriptor
<code>pps_pic_parameter_set_id</code>	<code>ue(v)</code>
<code>...</code>	
<code>init_qp_minus26</code>	<code>se(v)</code>
<code>transform_skip_enabled_flag</code>	<code>u(1)</code>
<code>pps_cb_qp_offset</code>	<code>se(v)</code>
<code>pps_cr_qp_offset</code>	<code>se(v)</code>
<code>pps_joint_cbr_qp_offset</code>	<code>se(v)</code>
<code>pps_slice_chroma_qp_offsets_present_flag</code>	<code>u(1)</code>
<code>...</code>	
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Các ngữ nghĩa của các phần tử cú pháp trong bảng 3 có thể là giống như bảng sau đây.

【Bảng 4】

init_qp_minus26 plus 26 specifies the initial value of SliceQp_Y for each slice referring to the PPS. The initial value of SliceQp_Y is modified at the slice layer when a non-zero value of slice_qp_delta is decoded. The value of init_qp_minus26 shall be in the range of $-(26 + \text{QpBdOffset}_Y)$ to +37, inclusive.

pps_cb_qp_offset and **pps_cr_qp_offset** specify the offsets to the luma quantization parameter Qp'_Y used for deriving Qp'_{Cb} and Qp'_{Cr} respectively. The values of pps_cb_qp_offset and pps_cr_qp_offset shall be in the range of -12 to +12, inclusive. When ChromaArrayType is equal to 0, pps_cb_qp_offset and pps_cr_qp_offset are not used in the decoding process and decoders shall ignore their value.

pps_joint_cbr_qp_offset specifies the offset to the luma quantization parameter Qp'_Y used for deriving Qp'_{CbCr} . The value of $\text{pps_joint_cbr_qp_offset}$ shall be in the range of -12 to +12, inclusive. When ChromaArrayType is equal to 0, $\text{pps_joint_cbr_qp_offset}$ is not used in the decoding process and decoders shall ignore its value.

pps_slice_chroma_qp_offsets_present_flag equal to 1 indicates that the $\text{slice_cb_qp_offset}$ and $\text{slice_cr_qp_offset}$ syntax elements are present in the associated slice headers. $\text{pps_slice_chroma_qp_offsets_present_flag}$ equal to 0 indicates that these syntax elements are not present in the associated slice headers. When ChromaArrayType is equal to 0, $\text{pps_slice_chroma_qp_offsets_present_flag}$ shall be equal to 0.

Ví dụ, giá trị thu được bằng cách cộng 26 vào phần tử cú pháp init_qp_minus26 có thể biểu diễn giá trị ban đầu của SliceQp_Y cho mỗi lát mà đề cập đến PPS. Nếu giá trị khác không của slice_qp_delta được giải mã, thì giá trị ban đầu của SliceQp_Y có thể được sửa đổi trong lớp lát. init_qp_minus26 có thể là trong phạm vi từ $-(26 + \text{QpBdOffset}_Y)$ đến +37.

Thêm nữa, ví dụ, các phần tử cú pháp pps_cb_qp_offset và pps_cr_qp_offset có thể biểu diễn các độ bù cho thông số lượng tử hóa độ chói Qp'_Y được sử dụng lần lượt để dẫn xuất Qp'_{Cb} và Qp'_{Cr} . pps_cb_qp_offset và pps_cr_qp_offset có thể là trong phạm vi từ -12 đến +12. Thêm nữa, khi ChromaArrayType là 0, trong quy trình giải mã, pps_cb_qp_offset và pps_cr_qp_offset có thể không được sử dụng, và thiết bị giải mã có thể bỏ qua các giá trị của các phần tử cú pháp.

Thêm nữa, ví dụ, phần tử cú pháp `pps_joint_cbr_qp_offset` có thể biểu diễn độ dịch cho thông số lượng tử hóa độ chói $Qp'Y$ được sử dụng để dẫn xuất $Qp'CbCr$. `pps_joint_cbr_qp_offset` có thể là trong phạm vi từ -12 đến +12. Thêm nữa, khi `ChromaArrayType` là 0, trong quy trình giải mã, `pps_joint_cbr_qp_offset` có thể không được sử dụng, và thiết bị giải mã có thể bỏ qua giá trị của phần tử cú pháp.

Thêm nữa, ví dụ, phần tử cú pháp `pps_slice_chroma_qp_offsets_present_flag` có thể biểu diễn liệu các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` là có mặt trong các phần đầu lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset`. Ví dụ, `pps_slice_chroma_qp_offsets_present_flag` có giá trị là 1 có thể biểu diễn rằng các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` là có mặt trong các phần đầu lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset`. Thêm nữa, ví dụ, `pps_slice_chroma_qp_offsets_present_flag` có giá trị là 0 có thể biểu diễn rằng các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` là không có mặt trong các phần đầu lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset`. Thêm nữa, khi `ChromaArrayType` là 0, trong quy trình giải mã, `pps_slice_chroma_qp_offsets_present_flag` có thể là giống nhau như 0.

Như trong các nội dung được đề cập đến ở trên, các phần tử cú pháp được phân tích cú pháp trong PPS có thể là `init_qp_minus26`, `pps_cb_qp_offset`, `pps_cr_qp_offset`, `pps_joint_cbr_qp_offset`, và `pps_slice_chroma_qp_offsets_present_flag`. Phần tử cú pháp `init_qp_minus26` có thể biểu diễn giá trị ban đầu của `SliceQpY` cho mỗi lát mà đề cập đến PPS. Thêm nữa, các phần tử cú pháp `pps_cb_qp_offset`, `pps_cr_qp_offset`, và `pps_joint_cbr_qp_offset` có thể biểu diễn các độ dịch cho thông số lượng tử hóa độ chói

Qp'Y. Thêm nữa, phần tử cú pháp `pps_slice_chroma_qp_offsets_present_flag` có thể biểu diễn liệu thông số độ bù là có mặt trong phần đầu lát.

Thêm nữa, thông tin liên quan đến việc dẫn xuất các thông số lượng tử hóa có thể được báo hiệu như trong bảng sau đây thông qua phần đầu lát chẳng hạn.

【Bảng 5】

<code>slice_header() {</code>	Descriptor
<code>slice_pic_parameter_set_id</code>	<code>uc(v)</code>
<code>...</code>	
<code>slice_qp_delta</code>	<code>se(v)</code>
<code>if(pps_slice_chroma_qp_offsets_present_flag) {</code>	
<code>slice_cb_qp_offset</code>	<code>se(v)</code>
<code>slice_cr_qp_offset</code>	<code>se(v)</code>
<code>slice_joint_cbr_qp_offset</code>	<code>se(v)</code>
<code>}</code>	
<code>...</code>	
<code>byte_alignment()</code>	
<code>}</code>	

Các ngữ nghĩa của các phần tử cú pháp trong bảng 5 có thể là giống như bảng sau đây.

【Bảng 6】

slice_qp_delta specifies the initial value of Qp_Y to be used for the coding blocks in the slice until modified by the value of **CuQpDeltaVal** in the coding unit layer. The initial value of the Qp_Y quantization parameter for the slice, **SliceQp_Y**, is derived as follows:

$$\text{SliceQp}_Y = 26 + \text{init_qp_minus26} + \text{slice_qp_delta}$$

The value of **SliceQp_Y** shall be in the range of $-QpBdOffset_Y$ to +63, inclusive.

slice_cb_qp_offset specifies a difference to be added to the value of **pps_cb_qp_offset** when determining the value of the Qp'_{Cb} quantization parameter. The value of **slice_cb_qp_offset** shall be in the range of -12 to +12, inclusive. When **slice_cb_qp_offset** is not present, it is inferred to be equal to 0. The value of **pps_cb_qp_offset** + **slice_cb_qp_offset** shall be in the range of -12 to +12, inclusive.

slice_cr_qp_offset specifies a difference to be added to the value of **pps_cr_qp_offset** when determining the value of the Qp'_{Cr} quantization parameter. The value of **slice_cr_qp_offset** shall be in the range of -12 to +12, inclusive. When **slice_cr_qp_offset** is not present, it is inferred to be equal to 0. The value of **pps_cr_qp_offset** + **slice_cr_qp_offset** shall be in the range of -12 to +12, inclusive.

slice_joint_cbr_qp_offset specifies a difference to be added to the value of **pps_joint_cbr_qp_offset** when determining the value of the Qp'_{CbCr} . The value of **slice_joint_cbr_qp_offset** shall be in the range of -12 to +12, inclusive. When **slice_joint_cbr_qp_offset** is not present, it is inferred to be equal to 0. The value of **pps_joint_cbr_qp_offset** + **slice_joint_cbr_qp_offset** shall be in the range of -12 to +12, inclusive.

The derivation process for the luma and chroma quantization parameters begin with the inputs to the process being a luma location, variables specifying the width and height of the current coding block and a variable specifying whether it is a single or a dual tree. The luma, chroma and the joint chroma quantization parameters are denoted as follows: Qp'_Y , Qp'_{Cb} , Qp'_{Cr} and Qp'_{CbCr} .

cu_qp_delta_sign_flag specifies the sign of **CuQpDeltaVal** as follows:

- If **cu_qp_delta_sign_flag** is equal to 0, the corresponding **CuQpDeltaVal** has a positive value.
- Otherwise (**cu_qp_delta_sign_flag** is equal to 1), the corresponding **CuQpDeltaVal** has a negative value.

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

When **cu_qp_delta_abs** is present, the variables **IsCuQpDeltaCoded** and **CuQpDeltaVal** are derived as follows:

$$\text{IsCuQpDeltaCoded} = 1$$

$$\text{CuQpDeltaVal} = \text{cu_qp_delta_abs} * (1 - 2 * \text{cu_qp_delta_sign_flag})$$

The value of **CuQpDeltaVal** shall be in the range of $-(32 + \text{QpBdOffsetY} / 2)$ to $+(31 + \text{QpBdOffsetY} / 2)$, inclusive.

Ví dụ, **slice_qp_delta** có thể biểu diễn giá trị ban đầu của **QpY** cần được sử dụng trong khối tạo mã trong lát cho đến khi nó được sửa đổi bằng giá trị của **CuQpDeltaVal** trong lớp đơn vị tạo mã. Ví dụ, giá trị ban đầu của **QpY** cho lát, **SliceQpY**, có thể được dẫn xuất như $26 + \text{init_qp_minus26} + \text{slice_qp_delta}$. Giá trị của **SliceQpY** có thể là trong phạm vi từ $-\text{QpBdOffsetY}$ đến $+63$.

Thêm nữa, ví dụ, **slice_cb_qp_offset** có thể biểu diễn mức chênh lệch cần được cộng vào giá trị của **pps_cb_qp_offset** khi giá trị của thông số lượng tử hóa **Qp'Cb** được xác định. Giá trị của **slice_cb_qp_offset** có thể là trong phạm vi từ -12 đến $+12$. Thêm nữa, ví dụ, nếu **slice_cb_qp_offset** là không có mặt, thì **slice_cb_qp_offset** có thể được suy ra là 0. Giá trị của **pps_cb_qp_offset** + **slice_cb_qp_offset** có thể là trong phạm vi từ -12 đến $+12$.

Thêm nữa, ví dụ, **slice_cr_qp_offset** có thể biểu diễn mức chênh lệch cần được cộng vào giá trị của **pps_cr_qp_offset** khi giá trị của thông số lượng tử hóa **Qp'Cr** được xác định. Giá trị của **slice_cr_qp_offset** có thể là trong phạm vi từ -12 đến $+12$. Thêm

nữa, ví dụ, nếu `slice_cr_qp_offset` là không có mặt, thì `slice_cr_qp_offset` có thể được suy ra là 0. Giá trị của `pps_cr_qp_offset + slice_cr_qp_offset` có thể là trong phạm vi từ 12 đến +12.

Thêm nữa, ví dụ, `slice_cbr_qp_offset` có thể biểu diễn mức chênh lệch cần được cộng vào giá trị của `pps_cbr_qp_offset` khi giá trị của thông số lượng tử hóa $Qp'CbCr$ được xác định. Giá trị của `slice_cbr_qp_offset` có thể là trong phạm vi từ -12 đến +12. Thêm nữa, ví dụ, nếu `slice_cbr_qp_offset` là không có mặt, thì `slice_cbr_qp_offset` có thể được suy ra là 0. Giá trị của `pps_cbr_qp_offset + slice_cbr_qp_offset` có thể là trong phạm vi từ 12 đến +12.

Quy trình dẫn xuất các thông số lượng tử hóa độ chói và sắc độ có thể được bắt đầu dựa trên thực tế là đầu vào cho quy trình này là vị trí độ chói, thông số để chỉ định chiều rộng và chiều cao của khối tạo mã hiện tại, và thông số để chỉ định cây đơn hoặc cây kép. Trong khi đó, như trong các nội dung được đề cập đến ở trên, thông số lượng tử hóa độ chói, thông số lượng tử hóa sắc độ và thông số lượng tử hóa sắc độ kết hợp có thể được biểu diễn như $Qp'Y$, $Qp'Cb$, $Qp'Cr$ và $Qp'CbCr$.

Trong khi đó, ví dụ, phần tử cú pháp `cu_qp_delta_sign_flag` biểu diễn dấu của `CuQpDeltaVal` có thể được phân tích cú pháp. Ví dụ, `cu_qp_delta_sign_flag` có thể biểu diễn dấu của `CuQpDeltaVal` như sau.

Ví dụ, khi `cu_qp_delta_sign_flag` là 0, thì `CuQpDeltaVal` tương ứng với `cu_qp_delta_sign_flag` có thể có giá trị dương. Thay vào đó, ví dụ, khi `cu_qp_delta_sign_flag` là 1, thì `CuQpDeltaVal` tương ứng với `cu_qp_delta_sign_flag` có thể có giá trị âm. Thêm nữa, nếu `cu_qp_delta_sign_flag` là không có mặt, thì `cu_qp_delta_sign_flag` có thể được suy ra là 0.

Thêm nữa, ví dụ, nếu $cu_qp_delta_abs$ là có mặt, thông số $IsCuQpDeltaCoded$ có thể được dẫn xuất như 1. Thông số $CuQpDeltaVal$ có thể được dẫn xuất như $cu_qp_delta_abs * (1 - 2 * cu_qp_delta_sign_flag)$. $CuQpDeltaVal$ có thể là trong phạm vi từ $-(32 + QpBdOffsetY/2)$ đến $+(31 + QpBdOffsetY/2)$.

Sau đó, ví dụ, thông số lượng tử hóa độ chói QpY có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 1】

$$QpY = ((qP_{Y_PRED} + CuQpDeltaVal + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - QpBdOffset_Y$$

Thêm nữa, nếu $ChromaArrayType$ không là 0 và, $treeType$ là $SINGLE_TREE$ hay $DUAL_TREE_CHROMA$, thì được áp dụng như sau.

- Khi $treeType$ là bằng $DUAL_TREE_CHROMA$, thì thông số QpY có thể được thiết đặt giống với thông số lượng tử hóa độ sáng QpY của đơn vị tạo mã độ chói bao gồm vị trí độ chói $(xCb + cbWidth/2, yCb + cbHeight/2)$.

- Các thông số $qPCb$, $qPCr$ và $qPCbCr$ có thể được dẫn xuất như sau.

【Phương trình 2】

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

Ví dụ, khi $ChromaArrayType$ là 1, thì các thông số $qPCb$, $qPCr$ và $qPCbCr$ có thể được thiết đặt giống với giá trị QpC được chỉ định trong bảng 7 dựa trên các chỉ số qPi lần lượt giống như $qPiCb$, $qPiCr$ và $qPiCbCr$.

【Bảng 7】

qPi	< 30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	> 43
Qp _C	= qPi	29	30	31	32	33	33	34	34	35	35	36	36	37	37	= qPi - 6

Thay vào đó, khi ChromaArrayType không là 1, thì các thông số qPCb, qPCr, và qPCbCr có thể được thiết đặt giống với Min (qPi, 63) dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbC.

- Các thông số lượng tử hóa sắc độ cho thành phần Cb và thành phần Cr, Qp'Cb và Qp'Cr, và thông số lượng tử hóa sắc độ Qp'CbCr cho việc tạo mã Cb-Cr kết hợp có thể được dẫn xuất như sau.

【Phương trình 3】

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Trong khi đó, sáng chế này đề xuất các sơ đồ để cải thiện hiệu quả tạo mã trong quy trình lượng tử hóa/khử lượng tử hóa.

Trong một phương án, sáng chế này đề xuất phương pháp để định rõ và sử dụng bảng ánh xạ lượng tử hóa sắc độ được định rõ bởi người dùng, không phải phương pháp để thu được giá trị thông số lượng tử hóa sắc độ từ giá trị thông số lượng tử hóa độ chói thông qua bảng ánh xạ lượng tử hóa sắc độ được xác định trước trong VVC Draft5 v.7 hiện có khi ChromaArrayType không là 0 (ví dụ, khi ChromaArrayType là 1). Trong phần chỉ rõ VVC (ví dụ, VVC Draft5 v.7), khi qPi (giá trị thông số lượng tử hóa độ chói) được cho, thì Qpc (giá trị thông số lượng tử hóa sắc độ) được dẫn xuất thông qua bảng lượng tử hóa sắc độ định trước (ví dụ, bảng 7), nhưng sáng chế này đề xuất phương pháp để dẫn xuất Qpc từ qPi dựa trên bảng ánh xạ lượng tử hóa sắc độ được xác định mới bởi

người dùng. Theo một phương án của sáng chế này, có phương pháp trong đó giá trị Qpc có thể được dẫn xuất thông qua quan hệ hàm của giá trị qPi được cung cấp, hàm có thể được báo hiệu như cú pháp, như APS, SPS hoặc PPS, thông qua phương pháp chức năng được định rõ bởi người dùng, mà bao gồm các giá trị truyền quan hệ hàm của các phần tử cú pháp định trước, và người dùng xác định sự ánh xạ bảng lượng tử hóa sắc độ dựa trên các giá trị được truyền. Ví dụ, bởi vì giá trị Qpc có thể được dẫn xuất thông qua quan hệ hàm có giá trị qPi, nếu các giá trị phần tử cú pháp biểu diễn hàm tương ứng được truyền, bảng ánh xạ lượng tử hóa sắc độ được định rõ bởi người dùng có thể được dẫn xuất dưới một hình thức, như trong bảng 7.

Trong một phương án, sơ đồ để báo hiệu thông tin về các phần tử cú pháp (Qpc_data) biểu diễn hàm liên quan đến việc ánh xạ lượng tử hóa sắc độ như trong bảng sau đây mà sẽ được mô tả sau trong tập hợp thông số thích ứng (APS).

【Bảng 8】

adaptation_parameter_set_rbsp() {	Descriptor
adaptation_parameter_set_id	u(5)
aps_params_type	u(3)
if(aps_params_type == ALF_APS)	
alf_data(adaptation_parameter_set_id)	
else if(aps_params_type == LMCS_APS)	
lmcs_data()	
else if(aps_params_type == Qpc_APS) // 2	
Qpc_data()	
aps_extension_flag	u(1)
if(aps_extension_flag)	
while(more_rbsp_data())	
aps_extension_data_flag	u(1)
rbp_trailing_bits()	
}	

Tham chiếu đến bảng 8, nếu aps_params_type biểu diễn Qpc_APS, chẳng hạn, khi giá trị của aps_params_type là 2, thì Qpc_data() có thể được báo hiệu.

Các ngữ nghĩa của các phần tử cú pháp trong bảng 8 có thể là giống như bảng sau đây.

【Bảng 9】

adaptation_parameter_set_id provides an identifier for the APS for reference by other syntax elements.

NOTE – APSs can be shared across pictures and can be different in different slices within a picture.

aps_extension_flag equal to 0 specifies that no **aps_extension_data_flag** syntax elements are present in the APS RBSP syntax structure. **aps_extension_flag** equal to 1 specifies that there are **aps_extension_data_flag** syntax elements present in the APS RBSP syntax structure.

aps_extension_data_flag may have any value. Its presence and value do not affect decoder conformance to profiles specified in this version of this Specification. Decoders conforming to this version of this Specification shall ignore all **aps_extension_data_flag** syntax elements.

aps_params_type specifies the type of APS parameters carried in the APS as specified in the Table 2 shown below.

Ví dụ, phần tử cú pháp **adaptation_parameter_set_id** có thể cung cấp bộ định danh của APS được tham chiếu bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp **aps_extension_flag** có thể biểu diễn liệu các phần tử cú pháp **aps_extension_data_flag** là có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp **aps_extension_flag** có giá trị là 1 có thể biểu diễn các phần tử cú pháp **aps_extension_data_flag** là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp **aps_extension_flag** có giá trị là 0 có thể biểu diễn các phần tử cú pháp **aps_extension_data_flag** là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp **aps_extension_data_flag** có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của **aps_extension_data_flag** có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng (profile) được chỉ rõ trong một phiên bản của chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp **aps_extension_data_flag**.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS được bao gồm trong APS như được minh họa trong bảng 10.

【Bảng 10】

aps_params_type	Name of aps_params_type	Type of APS parameters
0	ALF_APS	ALF parameters
1	LMCS_APS	LMCS parameters
2	Qpc_APS	Qpc data parameters
3..7	Reserved	Reserved

Ví dụ, tham chiếu đến bảng 10, khi giá trị của phần tử cú pháp `aps_params_type` là 0, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS là các thông số ALF. Khi giá trị của phần tử cú pháp `aps_params_type` là 1, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS là các thông số LMCS. Khi giá trị của phần tử cú pháp `aps_params_type` là 2, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS là các thông số dữ liệu Qpc. Thông số dữ liệu Qpc có thể biểu diễn thông số dữ liệu lượng tử hóa sắc độ.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất sơ đồ để báo hiệu dữ liệu QpC được định rõ bởi người dùng trong tập hợp thông số hình ảnh (PPS). Như ví dụ để thực hiện sơ đồ được đề xuất trong phương án này, cờ biểu diễn liệu PPS có bao gồm dữ liệu được định rõ bởi người dùng trong SPS hay không có thể được giới thiệu. Tức là, cờ biểu diễn liệu PPS có bao gồm dữ liệu được định rõ bởi người dùng trong SPS hay không, có thể được báo hiệu. Thêm nữa, theo phương án này, dữ liệu được định rõ bởi người dùng có thể được báo hiệu trong PPS. Thay vào đó, dữ liệu được định rõ bởi người dùng có thể được báo hiệu trong phần đầu lát và/hoặc tập phần đầu khác.

Cờ biểu diễn liệu PPS có bao gồm dữ liệu được định rõ bởi người dùng hay không có thể được báo hiệu trong bảng sau đây.

【Bảng 11】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>...</code>	
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể là phần tử cú pháp của cờ được đề cập đến ở trên. Phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu các thông số `Qpc_data()` là có mặt trong cấu trúc cú pháp RBSP PPS hay không. Ví dụ, `Qpc_data_default_flag` là 0 có thể biểu diễn rằng các thông số `Qpc_data()` là không có mặt trong cấu trúc cú pháp RBSP PPS và bảng mặc định được sử dụng để giúp xác định việc lượng tử hóa sắc độ. Trong trường hợp này, bảng mặc định có thể là giống như bảng 7. Thêm nữa, ví dụ, `Qpc_data_default_flag` là 1 có thể biểu diễn rằng các thông số `Qpc_data()` có thể là có mặt trong cấu trúc cú pháp RBSP PPS.

Thêm nữa, dữ liệu được định rõ bởi người dùng được báo hiệu trong PPS theo phương án này có thể là giống như bảng sau đây.

【Bảng 12】

pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
...	
if(Qpc_data_default_flag)	
Qpc_data()	
...	
rbp_trailing_bits()	
}	

Trong khi đó, ví dụ, Qpc_data() có thể bao gồm thông tin cần cho việc dẫn xuất lượng tử hóa sắc độ khi ChromaArrayType là 1.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất cấu trúc linh hoạt cho việc dẫn xuất thông số lượng tử hóa (QP) sắc độ và việc dẫn xuất QP sắc độ được tổ hợp. Phương án này đề xuất sơ đồ để báo hiệu cờ ban đầu biểu diễn liệu chế độ được định rõ bởi người dùng trong đó các thông số biểu diễn hàm được sử dụng để dẫn xuất thông số lượng tử hóa (QP) sắc độ trong SPS và/hoặc PPS có thể được sử dụng là có mặt hay không.

Ví dụ, thông tin cờ được báo hiệu trong cú pháp cấp cao được đề xuất trong phương án này có thể là giống như bảng mà sẽ được mô tả sau.

【Bảng 13】

high_level_syntax_parameter_set {	Descriptor
...	
Qpc_data_present_flag	u(1)
...	
}	

Ví dụ, `Qpc_data_present_flag` có thể biểu diễn liệu các thông số để dẫn xuất thông số lượng tử hóa sắc độ là có mặt trong cấu trúc cú pháp RBSP của cú pháp cấp cao hay không. Ví dụ, `Qpc_data_present_flag` có giá trị là 0 có thể biểu diễn rằng các thông số lượng tử hóa sắc độ là không có mặt trong cấu trúc cú pháp RBSP của cấu trúc cú pháp cấp cao. Thêm nữa, ví dụ, `Qpc_data_present_flag` có giá là 1 có thể biểu diễn rằng các thông số lượng tử hóa sắc độ là có mặt trong cấu trúc cú pháp RBSP của cú pháp cấp cao.

Thay vào đó, phần tử cú pháp `Qpc_data_present_flag` có thể được sử dụng để chỉ báo sơ đồ sử dụng việc dẫn xuất lượng tử hóa sắc độ trong luồng bit. Ví dụ, phần tử cú pháp `Qpc_data_present_flag` có thể biểu diễn công cụ được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hoặc sử dụng chế độ được định rõ bởi người dùng như sau.

Ví dụ, `Qpc_data_present_flag` có thể biểu diễn liệu việc lượng tử hóa sắc độ được định rõ bởi người dùng được sử dụng trong luồng bit hay không. Ví dụ, `Qpc_data_present_flag` có giá trị là 0 có thể biểu diễn liệu việc lượng tử hóa sắc độ được định rõ bởi người dùng là không được sử dụng trong luồng bit. Thêm nữa, ví dụ, `Qpc_data_present_flag` có giá trị là 1 có thể biểu diễn rằng lượng tử sắc độ được định rõ bởi người dùng được sử dụng một mình hay cùng với cờ khác.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất phương án trong đó các thông số lượng tử hóa (QP) sắc độ, tức là, `Qp`Cb`, `Qp`Cr` và `Qp`CbCr` có thể được dẫn xuất sử dụng thông tin được định rõ bởi người dùng được báo hiệu trong một hàm như thế nào. Ví dụ, theo phương án này, dữ liệu biểu diễn hàm để dẫn xuất thông số lượng tử hóa (QP) sắc độ có thể được báo hiệu, và các thông số lượng tử hóa sắc độ có thể được dẫn xuất dựa trên dữ liệu

lượng tử hóa sắc độ. Dữ liệu (hoặc bảng ánh xạ QP được định rõ bởi người dùng) cho việc dẫn xuất thông số lượng tử hóa sắc độ có thể được báo hiệu như trong bảng sau đây.

【Bảng 14】

Qp_C_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
Qp_C_qPi_val[i]	ue(v)
QpOffset_C	ue(v)
}	

Các ngữ nghĩa của các phần tử cú pháp trong bảng 14 có thể là giống như bảng sau đây.

【Bảng 15】

qPi_min_idx specifies the minimum qPi index used in the chroma quantization.

qPi_delta_max_idx specifies the delta value between the Qp_C_min_idx and the maximum qPi index used for the chroma Qp_C derivation. The value of qPiMaxIdx shall be greater than or equal to qPi_min_idx. The maximum index qPiMaxIdx used in Qp_C derivation is derived as follows:

$$qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx$$

Qp_C_qPi_val[i] specifies the Qp_C value for the ith index.

QpOffset_C specifies the offset value to be used the derivation of the Qp_C.

The variable Qp_CIdx[qPi] for qPi with qPi = 0 ... qPiMaxIdx, is derived as follows:

- For qPi < qPi_min_idx, Qp_CIdx[qPi] is set equal to qPi.
- For qPi = qPi_min_idx ... qPiMaxIdx, the following applies:

$$Qp_C_Idx[qPi] = Qp_C_qPi_val[qPi]$$

- For qPi > qPiMaxIdx, Qp_CIdx[qPi] = qPi - QpOffset_C

The value of Qp_C is derived as Qp_CIdx[qPi].

Ví dụ, phần tử cú pháp `qPi_min_idx` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `Qpi_min_idx` và chỉ số `qPi` tối đa được sử dụng trong khi dẫn xuất sắc độ `Qpc`. Giá trị của `qPiMaxIdx` có thể là lớn hơn hoặc bằng `qPi_min_idx`. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qpc` có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 4】

$$qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx$$

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_val[i]` có thể biểu diễn giá trị `QpC` cho chỉ số thứ `i` (`i`-th).

Thêm nữa, ví dụ, phần tử cú pháp `QpOffsetC` có thể biểu diễn giá trị bù được sử dụng cho việc dẫn xuất `QpC`.

Thêm nữa, ví dụ, thông số `QpCIdx[qPi]` cho `qPi` có thể được dẫn xuất như sau. Trong trường hợp này, `qPi` có thể là từ 0 đến `qPiMaxIdx`.

- Khi `qPi < qPi_min_idx`, thì `QpCIdx[qPi]` có thể được thiết đặt giống với `qPi`.
 - Khi `qPi = qPi_min_idx...qPiMaxIdx`, thì `QpCIdx[qPi]` có thể được thiết đặt giống với `QpC_qPi_val[qPi]`.
 - Khi `qPi > qPiMaxIdx`, thì `QpCIdx[qPi]` có thể được thiết đặt như `qPi - QpOffsetC`.
- Sau đó, giá trị của `QpC` có thể được dẫn xuất như `QpCIdx[qPi]`.

Ví dụ, theo phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được mô tả trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 16】

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp \parallel yTmp] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - QpBdOffset_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as specified in-Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively clause 7.x.x
- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Tham chiếu đến bảng 16, quy trình dẫn xuất cho thông số lượng tử hóa độ chói và sắc độ có thể được bắt đầu dựa vào thực tế là đầu vào cho quy trình là vị trí độ chói (x_{Cb} , y_{Cb}), thông số $cbWidth$ và $cbHeight$ chỉ định chiều rộng và chiều cao của khối tạo mã hiệu tại, và thông số $treeType$ chỉ định cây đơn hoặc cây kép. Trong khi đó, như trong

các nội dung được mô tả ở trên, thông số lượng tử hóa độ chói và thông số lượng tử hóa sắc độ có thể được biểu diễn như Qp'Y, Qp'Cb và Qp'Cr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất ví dụ trong đó cờ bên trong như SPS có chế độ được định rõ bởi người dùng hoặc chế độ mặc định, thì các phần tử cú pháp vốn có thể được sử dụng để điều khiển việc dẫn xuất thông số lượng tử hóa được sử dụng. Ví dụ về phần tử cú pháp vốn có thể được sử dụng để dẫn xuất thông số lượng tử hóa có thể là giống như trong các bảng sau đây. Trong khi đó, cấu trúc của các phần tử cú pháp không được giới hạn ở cấu trúc được minh họa trong các bảng sau đây, ví dụ.

【Bảng 17】

pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
...	
Qpc_data_default_flag	u(1)
...	

【Bảng 18】

slice_header() {	Descriptor
slice_pic_parameter_set_id	ue(v)
...	
if(!Qpc_data_default_flag)	
slice_Qp _C _aps_id	u(5)
byte_alignment()	
}	

【Bảng 19】

adaptation_parameter_set_rbsp() {	Descriptor
adaptation_parameter_set_id	u(5)
aps_params_type	u(3)
if(aps_params_type == ALF_APS)	
alf_data(adaptation_parameter_set_id)	
else if(aps_params_type == LMCS_APS)	
lmcs_data()	
else if(aps_params_type == Qpc_APS) // 2	
Qpc_data()	
aps_extension_flag	u(1)
if(aps_extension_flag)	
while(more_rbsp_data())	
aps_extension_data_flag	u(1)
rbps_trailing_bits()	
}	

Ví dụ, phần tử cú pháp Qpc_data_default_flag có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng cho việc dẫn xuất thông số lượng tử hóa hay không. Ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng cho việc dẫn xuất thông số lượng tử hóa sắc độ. Trong trường hợp này, bảng mặc định có thể là giống như bảng 7. Thêm nữa, nếu phần tử cú pháp Qpc_data_default_flag là không có mặt, thì phần tử cú pháp Qpc_data_default_flag có thể được suy ra là 1.

Trong khi đó, nếu chế độ được định rõ bởi người dùng được sử dụng, thì phần đầu lát, nhóm phiên/phần đầu, hoặc phần đầu thích hợp khác tương ứng có thể được sử dụng để báo hiệu ID APS. Ví dụ, như trong bảng 18, phần tử cú pháp biểu diễn ID APS thông qua phần đầu lát có thể được báo hiệu.

Ví dụ, phần tử cú pháp `slice_Qpc_aps_id` có thể biểu diễn `adaptation_parameter_set_id` của APS Qpc được đề cập đến bởi lát. `TemporalId` của đơn vị NAL APS Qpc có `adaptation_parameter_set_id`, như `slice_Qpc_aps_id`, có thể là nhỏ hơn hoặc bằng `TemporalId` của đơn vị NAL lát được tạo mã. Nếu nhiều APS Qpc có `adaptation_parameter_set_id` có cùng giá trị được đề cập đến bởi hai hoặc nhiều lát của cùng hình ảnh, thì nhiều APS Qpc có `adaptation_parameter_set_id` có cùng giá trị có thể có cùng nội dung.

Thêm nữa, cấu trúc APS mà chuyển dữ liệu lượng tử hóa sắc độ, vốn được đề xuất trong phương án này, có thể là giống như bảng 19.

Ví dụ, phần tử cú pháp `adaptation_parameter_set_id` có thể cung cấp bộ định danh của APS được đề cập đến bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_flag` có thể biểu diễn liệu các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp `aps_extension_flag` có giá trị là 1 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp `aps_extension_flag` có giá trị là 0 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_data_flag` có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của `aps_extension_data_flag` có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng (profile) được chỉ rõ trong một phiên bản của chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp `aps_extension_data_flag`.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS được bao gồm trong APS như được minh họa trong bảng 10.

Qpc_data() được bọc lộ trong bảng 19 có thể được báo hiệu như bảng sau đây.

【Bảng 20】

QpC_data () {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
QpC_prec_minus1	ue(v)
QpC_init_val	ue(v)
for (i = qPi_min_idx + 1; i <= qPiMaxIdx; i++)	
QpC_qPi_delta_val[i]	ue(v)
QpOffsetC	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp qPi_delta_max_idx có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi tối đa được sử dụng trong khi dẫn xuất sắc độ Qpc. Giá trị của qPiMaxIdx có thể là lớn hơn hoặc bằng qPi_min_idx. Ví dụ, chỉ số tối đa qPiMaxIdx được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp Qpc_prec_minus1 có thể biểu diễn số lượng bit được sử dụng cho việc biểu diễn cú pháp lms_delta_abs_cw[i]. Giá trị của Qpc_prec_minus1 có thể là trong phạm vi từ 0 đến BitDepthY – 2.

Thêm nữa, ví dụ, phần tử cú pháp QpC_init_val có thể biểu diễn giá trị QpC tương ứng với qPi_min_idx.

Thêm nữa, ví dụ, phần tử cú pháp QpC_qPi_delta_val[i] có thể biểu diễn delta của giá trị QpC cho chỉ số thứ i.

Thêm nữa, ví dụ, phần tử cú pháp QpOffsetC có thể biểu diễn giá trị bù được sử dụng cho việc dẫn xuất Qpc.

Ví dụ, thông số QpCIdx[qPi] for qPi có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là từ 0 đến qPiMaxIdx.

- Khi $qPi < qPi_min_idx$, thì QpCIdx[qPi] có thể được thiết đặt giống với qPi.
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì QpCIdx[qPi] có thể được thiết đặt như $Qpc_qPi_delta_val[qPi] + QpCIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, thì QpCIdx[qPi] có thể được thiết đặt như $qPi - QpOffsetC$.

Sau đó, giá trị của QpC có thể được dẫn xuất như QpCIdx[qPi].

Như trong phương án được đề cập ở trên, thông số lượng tử hóa sắc độ, tức là, Qp`Cb, Qp`Cr và Qp`CbCr có thể được dẫn xuất sử dụng thông tin được định rõ bởi người dùng được báo hiệu hoặc bằng mặc định được minh họa trong bảng mặc định, như bảng 7.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 21】

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg - 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as specified in clause 7.x.x
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} respectively
- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 21, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất các phần tử cú pháp mà có thể được sử dụng để điều khiển việc dẫn xuất thông số lượng tử hóa bằng cách chỉ báo chế độ được định rõ bởi người dùng hoặc chế độ mặc định thông qua cờ của SPS. Cụ thể, phương án này đề xuất sơ đồ để báo hiệu các phần tử cú pháp của cấu trúc cú pháp sau đây. Trong khi đó, cấu trúc của các phần tử cú pháp là một ví dụ, và không được giới hạn đến cấu trúc được minh họa trong bảng sau đây.

【Bảng 22】

Qp _C _data () {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
Qp _C _qPi_delta_val[i]	ue(v)
QpOffset _C	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `Qpi_min_idx` và chỉ số `qPi` tối đa được sử dụng trong khi dẫn xuất sắc độ `Qpc`. Giá trị của `qPiMaxIdx` có thể là lớn hơn hoặc bằng `qPi_min_idx`. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qpc` có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_delta_val[i]` có thể biểu diễn delta của giá trị `QpC` cho chỉ số thứ `i`.

Thêm nữa, ví dụ, phần tử cú pháp `QpOffsetC` có thể biểu diễn giá trị bù được sử dụng trong khi dẫn xuất `Qpc`, như các nội dung được đề cập đến ở trên.

Như trong phương án được đề cập ở trên, thông số lượng tử hóa sắc độ, tức là, `Qp`Cb`, `Qp`Cr` và `Qp`CbCr` có thể được dẫn xuất sử dụng thông tin được định rõ bởi người dùng được báo hiệu hoặc bằng mặc định, như bảng 7.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 23】

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp} \parallel y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively as described below:

The variable $Qp_CIdx[i]$ is derived as follows:

- For $i < qPi_min_idx$, $Qp_CIdx[i]$ is set equal to qPi .
- For $i = qPi_min_idx \dots qPiMaxIdx$, the following applies:

$$Qp_CIdx[i] = Qp_C_qPi_delta_val[i] + Qp_CIdx[i-1]$$

$$\text{For } i > qPiMaxIdx, Qp_CIdx[i] = qPi - QpOffset_C$$

The value of Qp_C is derived as $Qp_CIdx[qPi]$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 23, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Ví dụ, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất tương tự với giá trị của Qp_C dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} như sau.

Ví dụ, thông số $QpCIIdx[i]$ có thể được dẫn xuất như sau.

- Khi $i < qPi_min_idx$, $QpCIIdx[qPi]$ có thể được thiết đặt giống với qPi .
 - Khi $i = qPi_min_idx \dots qPiMaxIdx$, thì $QpCIIdx[i]$ có thể được thiết đặt như $QpC_qPi_delta_val[i] + QpCIIdx[i-1]$.
 - Khi $i > qPiMaxIdx$, thì $QpCIIdx[i]$ có thể được thiết đặt như $qPi - QpOffsetC$.
- Sau đó, Qp_C có thể được thiết đặt như $QpCIIdx[i]$.

Thêm nữa, tham chiếu đến bảng 23, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất các phần tử cú pháp cho thông số dẫn xuất lượng tử hóa sắc độ (QpC) trong tập hợp thông số thích ứng (APS). Ví dụ, ID APS có thể được báo hiệu trong phần đầu lát. Thêm nữa, ví dụ, có thể có cờ bên trong tập hợp thông số hình ảnh (PPS) mà biểu diễn liệu bảng mặc định được sử dụng hay bảng được dẫn xuất từ thông tin được báo hiệu trong APS được sử dụng được đề xuất. Thêm nữa, ví dụ, nếu bảng mặc định là không được sử dụng, sơ đồ điều khiển bổ sung để hỗ trợ truy cập vào APS bao gồm dữ liệu QpC có thể được thêm vào phần đầu lát.

Trong khi đó, theo chuẩn video/ảnh hiện có, QP sắc độ có thể được dẫn xuất từ QP độ chói, và có thể được cập nhật bằng độ bù QP được báo hiệu bổ sung. bảng thông số lượng tử hóa sắc độ QpC hiện có có thể là bảng mặc định, như bảng 7.

Phương án này đề xuất bổ sung hàm để báo hiệu thông số lượng tử hóa sắc độ QpC như hàm của chỉ số qPi. APS có thể được sử dụng để hợp nhất các sơ đồ báo hiệu gồm các giá trị QpC.

Ví dụ, APS theo phương án này có thể là giống như bảng sau đây.

【Bảng 24】

<code>adaptation_parameter_set_rbsp() {</code>	Descriptor
<code>adaptation_parameter_set_id</code>	<code>u(5)</code>
<code>aps_params_type</code>	<code>u(3)</code>
<code>if(aps_params_type == ALF_APS)</code>	
<code>alf_data(adaptation_parameter_set_id)</code>	
<code>else if (aps_params_type == LMCS_APS)</code>	
<code>lmcs_data()</code>	
<code>else if (aps_params_type == QpC_APS) // 2</code>	
<code>QpC_data()</code>	
<code>aps_extension_flag</code>	<code>u(1)</code>
<code>if(aps_extension_flag)</code>	
<code>while(more_rbsp_data())</code>	
<code>aps_extension_data_flag</code>	<code>u(1)</code>
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `adaptation_parameter_set_id` có thể cung cấp bộ định danh của APS được đề cập đến bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các thông số APS được bao gồm trong APS như được minh họa trong bảng 10.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_flag` có thể biểu diễn liệu các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp `aps_extension_flag` có giá trị là 1 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp `aps_extension_flag` có giá trị là 0 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_data_flag` có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của `aps_extension_data_flag` có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng (profile) được chỉ rõ trong một phiên bản của

chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp `aps_extension_data_flag`.

`Qpc_data()` được bọc lộ trong bảng 24 có thể được báo hiệu trong bảng sau đây.

【Bảng 25】

<code>QpC_data () {</code>	Descriptor
<code>qPi_min_idx</code>	<code>ue(v)</code>
<code>qPi_delta_max_idx</code>	<code>ue(v)</code>
<code>for (i = qPi_min_idx; i <= qPiMaxIdx; i++)</code>	
<code>QpC_qPi_delta_val[i]</code>	<code>ue(v)</code>
<code>QpCOffsetC_present_flag</code>	<code>u(1)</code>
<code>if(QpCOffsetC_present_flag)</code>	
<code>QpOffsetC</code>	<code>ue(v)</code>
<code>}</code>	

Ví dụ, phần tử cú pháp `qPi_min_idx` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của `qPi_min_idx` có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` tối đa được sử dụng trong khi dẫn xuất sắc độ `Qpc`. Giá trị của `qPiMaxIdx` có thể là lớn hơn hoặc bằng `qPi_min_idx`. Thêm nữa, ví dụ, giá trị của `qPi_delta_max_idx` có thể là trong phạm vi từ 0 đến 63. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qpc` có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_delta_val[i]` có thể biểu diễn mức chênh lệch giữa các giá trị `QpC` cho chỉ số thứ `i`. Mức chênh lệch có thể cũng được gọi là delta.

Thêm nữa, ví dụ, phần tử cú pháp `QpCOffsetC_present_flag` có thể biểu diễn liệu `QpOffsetC` là có mặt trong luồng bit hay không. Ví dụ, `QpCOffsetC_present_flag` có giá trị là 1 có thể biểu diễn rằng `QpOffsetC` là có mặt trong luồng bit. Thêm nữa, ví dụ, `QpCOffsetC_present_flag` có giá trị là 0 có thể biểu diễn rằng `QpOffsetC` là không có mặt trong luồng bit. Khi `QpCOffsetC_present_flag` là không có mặt, thì `QpCOffsetC_present_flag` có thể được suy ra là 0.

Thêm nữa, ví dụ, phần tử cú pháp `QpOffsetC` có thể biểu diễn giá trị bù được sử dụng trong khi dẫn xuất `Qpc`.

Ví dụ, thông số `QpCIIdx[qPi]` for `qPi` có thể được dẫn xuất như sau. Trong trường hợp này, `qPi` có thể là 0 đến 63.

- Khi $qPi < qPi_min_idx$, thì `QpCIIdx[qPi]` có thể được thiết đặt giống với `qPi`.
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì `QpCIIdx[qPi]` có thể được thiết đặt như `QpC_qPi_delta_val[qPi] + QpCIIdx[qPi-1]`.
- Nếu $qPi > qPiMaxIdx$, `QpCOffsetC_present_flag` là 1, thì `QpCIIdx[qPi]` có thể được thiết đặt như $qPi - QpOffsetC$. Nếu `QpCOffsetC_present_flag` không là 1, tức là, nếu `QpCOffsetC_present_flag` là 0, thì `QpCIIdx[qPi]` có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIIdx[qPiMaxIdx])$.

Sau đó, giá trị của `QpC` có thể được dẫn xuất như `QpCIIdx[qPi]`.

Thêm nữa, phương án này đề xuất cờ được báo hiệu như PPS như trong bảng sau đây.

【Bảng 26】

pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
...	
Qpc_data_default_flag	u(1)
...	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng cho việc dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được đề cập đến ở trên được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là giống như bảng 7. Nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy ra là 1.

Thêm nữa, phương án này đề xuất phần tử cú pháp được báo hiệu như phần đầu lát như trong bảng sau đây.

【Bảng 27】

slice_header() {	Descriptor
slice_pic_parameter_set_id	ue(v)
...	
if(!Qpc_data_default_flag)	
slice_Qpc_aps_id	u(5)
...	
byte_alignment()	
}	

Ví dụ, phần tử cú pháp `slice_Qpc_aps_id` có thể biểu diễn `adaptation_parameter_set_id` của APS Qpc được đề cập đến bởi lát. `TemporalId` của đơn vị NAL APS Qpc có `adaptation_parameter_set_id`, như `slice_Qpc_aps_id`, có thể là nhỏ

hơn hoặc giống như TemporalId của đơn vị NAL lát được tạo mã. Nếu nhiều APS Qpc có adaptation_parameter_set_id có cùng giá trị được đề cập đến bởi hai hoặc nhiều lát của cùng hình ảnh, thì nhiều APS Qpc có adaptation_parameter_set_id có cùng giá trị có thể có cùng nội dung.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 28】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 28, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, trong phương án này đề xuất việc dẫn xuất được định rõ bởi người dùng đối với việc lượng tử hóa sắc độ được báo hiệu trong SPS như sau. Ví dụ, phương án này đề xuất việc lượng tử hóa sắc độ được định rõ bởi người dùng (QpC). Ví dụ, cờ SPS có thể biểu diễn liệu bảng mặc định có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không hay các nội dung của bảng cho việc dẫn xuất lượng tử hóa sắc độ được dẫn xuất trong thông tin sẽ được báo hiệu trong SPS.

Ví dụ, phương án này đề xuất sơ đồ để thực hiện việc lượng tử hóa như hàm của chỉ số qPi bằng cách sử dụng các phần tử cú pháp được minh họa trong bảng sau đây.

【Bảng 29】

QpC_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_delta_val[i]	ue(v)
}	

Ví dụ, phần tử cú pháp `qPi_min_idx` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của `qPi_min_idx` có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` tối đa được sử dụng trong khi dẫn xuất sắc độ `Qpc`. Giá trị của `qPiMaxIdx` có thể là lớn hơn hoặc bằng `qPi_min_idx`. Giá trị của `qPi_delta_max_idx` có thể là trong phạm vi từ 0 đến 63. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qpc` có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_delta_val[i]` có thể biểu diễn delta của giá trị `QpC` cho chỉ số thứ `i`.

Ví dụ, thông số `QpCIdx[qPi]` có thể được dẫn xuất như sau.

- Khi $qPi < qPi_min_idx$, thì `QpCIdx[qPi]` có thể được thiết đặt giống với `qPi`.
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì `QpCIdx[qPi]` có thể được thiết đặt như $QpC_qPi_delta_val[qPi] + QpCIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, `QpCIdx[qPi]` có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Theo đó, `QpC` có thể được thiết đặt như `QpCIdx[qPi]`.

Thêm nữa, cờ của SPS biểu diễn liệu bảng mặc định có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không hoặc liệu thông tin được báo hiệu có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không, vốn được đề xuất trong phương án này, có thể là giống như bảng sau đây.

【Bảng 30】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code> sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code> ...</code>	
<code> Qpc_data_default_flag</code>	<code>u(1)</code>
<code> if(!Qp_C_data_default_flag)</code>	
<code> Qp_C_data()</code>	
<code> ...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng cho việc dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy ra là 1.

Ví dụ, theo phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 31】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 &\text{minTbAddrB} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 31, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất việc bổ sung hàm để báo hiệu thông số lượng tử hóa sắc độ QpC như hàm của chỉ số qPi. Ví dụ, có thể có sơ đồ để báo hiệu các phần tử cú pháp cho bảng được định rõ bởi người dùng cho việc dẫn xuất thông số lượng tử hóa trong PPS được đề xuất. Theo đó, sự linh hoạt liên quan đến việc thay đổi bảng được định rõ bởi người dùng và bảng mặc định trong mỗi hình ảnh mà tham chiếu đến PPS có thể được cung cấp.

Các phần tử cú pháp cho bảng được định rõ bởi người dùng được báo hiệu trong PPS được đề xuất trong phương án này có thể là giống như bảng sau đây.

【Bảng 32】

QpC_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_delta_val[i]	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi tối đa được sử dụng trong khi dẫn xuất sắc độ Qpc . Giá trị của $qPiMaxIdx$ có thể là lớn hơn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi từ 0 đến 63. Ví dụ, chỉ số tối đa $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_delta_val[i]$ có thể biểu diễn delta của giá trị QpC cho chỉ số thứ i .

Ví dụ, thông số $QpCIIdx[qPi]$ có thể được dẫn xuất như sau.

- Khi $qPi < qPi_min_idx$, thì $QpCIIdx[qPi]$ có thể được thiết đặt giống với qPi .
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpCIIdx[qPi]$ có thể được thiết đặt như $QpC_qPi_delta_val[qPi] + QpCIIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, $QpCIIdx[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIIdx[qPiMaxIdx])$.

Theo đó, QpC có thể được thiết đặt như $QpCIIdx[qPi]$.

Thêm nữa, cờ của SPS biểu diễn liệu bảng mặc định có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không hoặc liệu thông tin được báo hiệu có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không, vốn được đề xuất trong phương án này, có thể là giống như bảng sau đây.

【Bảng 33】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code> sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code> ...</code>	
<code> Qpc_data_default_flag</code>	<code>u(1)</code>
<code> if(!Qp_C_data_default_flag)</code>	
<code> Qp_C_data()</code>	
<code> ...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Khi `Qpc_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng cho việc dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy ra là 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 34】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to `availableA`. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - `availableA` is equal to `FALSE`.
 - the CTB address `ctbAddrA` of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to `CtbAddrInBs`, where `ctbAddrA` is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to `availableB`. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 &\text{minTbAddrB} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 34, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất chế độ chung trong đó thông số lượng tử hóa sắc độ QpC được dẫn xuất và được báo hiệu.

Dữ liệu thông số lượng tử hóa sắc độ Qpc_data() cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 35】

QpC_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_delta_val[i]	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi tối đa được sử dụng trong khi dẫn xuất sắc độ Qpc . Giá trị của $qPiMaxIdx$ có thể là lớn hơn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi từ 0 đến 63. Ví dụ, chỉ số tối đa $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_delta_val[i]$ có thể biểu diễn delta của giá trị QpC cho chỉ số thứ i .

Ví dụ, thông số $QpCIdx[qPi]$ có thể được dẫn xuất như sau.

- Khi $qPi < qPi_min_idx$, thì $QpCIdx[qPi]$ có thể được thiết đặt giống với qPi .
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được thiết đặt như $QpC_qPi_delta_val[qPi] + QpCIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, $QpCIdx[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Theo đó, QpC có thể được thiết đặt như $QpCIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không hoặc liệu thông tin được báo hiệu có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không. Cờ có thể được báo hiệu thông qua cấu trúc cấp cao, như tập hợp thông số chuỗi (SPS), hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cấu trúc cấp cao có thể là giống như bảng sau đây.

【Bảng 36】

high_level_parameter_set() {	Descriptor
...	
Qpc_data_default_flag	u(1)
if(!Qpc_data_default_flag)	
Qpc_data()	
...	

Ví dụ, phần tử cú pháp Qpc_data_default_flag có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn dữ liệu thông số lượng tử hóa sắc độ Qpc_data() được sử dụng. Khi Qpc_data_default_flag là 0, thì dữ liệu thông số lượng tử hóa sắc độ Qpc_data() có thể được báo hiệu. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng cho việc dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng 7. Thêm nữa, nếu Qpc_data_default_flag là không có mặt, thì Qpc_data_default_flag có thể được suy ra là 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 37】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 &\text{minTbAddrB} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - QpBdOffset_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 37, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất sơ đồ để dẫn xuất bảng thông số lượng tử hóa sắc độ QpC mà không gồm độ bù. Phương án này có thể được đề xuất để được sử dụng cùng với APS hoặc để được sử dụng độc lập. Ví dụ, cấu trúc cú pháp của APS được tích hợp với dữ liệu lượng tử hóa sắc độ có thể là giống như bảng sau đây.

【Bảng 38】

QpC_data () {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_delta_val[i]	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi tối đa được sử dụng trong khi dẫn xuất sắc độ Qpc . Giá trị của $qPiMaxIdx$ có thể là lớn hơn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi từ 0 đến 63. Ví dụ, chỉ số tối đa $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_delta_val[i]$ có thể biểu diễn mức chênh lệch giữa các giá trị QpC cho chỉ số thứ i . Mức chênh lệch có thể cũng được gọi là delta.

Ví dụ, thông số $QpCIdx[qPi]$ có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là 0 đến 63.

- Khi $qPi < qPi_min_idx$, thì $QpCIdx[qPi]$ có thể được thiết đặt giống với qPi .
- $qPi =$ Khi $qPi_min_idx \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được thiết đặt như $QpC_qPi_delta_val[qPi] + QpCIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, $QpCIdx[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Theo đó, QpC có thể được thiết đặt như $QpCIdx[qPi]$.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho thông số lượng tử hóa được báo hiệu.

Ví dụ, phương án đề xuất, như một ví dụ, sơ đồ trong đó delta (hoặc mức chênh lệch) giữa các giá trị QpC liên tục được giới hạn là 1.

Ví dụ, phương án này đề xuất sơ đồ trong đó việc lượng tử hóa sắc độ được định rõ bởi người dùng (QpC) được bao gồm thêm trong chuẩn ảnh/video hiện có. Ví dụ, cờ của tập hợp thông số chuỗi (SPS) được đề xuất trong phương án này có thể biểu diễn

liệu bảng mặc định hiện có được sử dụng cho việc dẫn xuất thông số lượng tử hóa sắc độ hay không hoặc liệu các nội dung của bảng có được dẫn xuất dựa trên thông tin được báo hiệu trong SPS hay không. Sơ đồ thích hợp cho ảnh được tạo mã có thể được chọn bằng cách điều tiết việc lượng tử hóa sắc độ theo phương án này, và hiệu quả tạo mã có thể được cải thiện.

Ví dụ, phương án này đề xuất việc bổ sung hàm để báo hiệu việc lượng tử hóa sắc độ QpC như hàm của chỉ số qPi bằng cách sử dụng các phần tử cú pháp như trong bảng sau đây.

【Bảng 39】

QpC_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_flag[i]	u(1)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp qPi_delta_max_idx có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi tối đa được sử dụng trong khi dẫn xuất sắc độ Qpc. Giá trị của qPiMaxIdx có thể là lớn hơn hoặc bằng qPi_min_idx. Giá trị của qPi_delta_max_idx có thể là trong phạm vi từ 1 đến 63. Ví dụ, chỉ số tối đa qPiMaxIdx được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất giống phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp QpC_qPi_flag[i] có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp QpC_qPi_flag[i]

có thể biểu diễn liệu giá trị QpC thứ i đã tăng thêm 1 hay không so với giá trị QpC thứ $(i-1)$. Ví dụ, QpC_qPi_flag[i] có giá trị là 1 có thể biểu diễn giá trị QpC được tăng thêm 1. QpC_qPi_flag[i] có giá trị là 0 có thể biểu diễn rằng giá trị QpC không được tăng thêm.

Ví dụ, thông số QpCIIdx[qPi] có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là 0 đến 63.

- Khi $qPi < qPi_min_idx$, thì QpCIIdx[qPi] có thể được thiết đặt giống với qPi.
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì QpCIIdx[qPi] có thể được thiết đặt như $QpC_qPi_flag[qPi] + QpCIIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, QpCIIdx[qPi] có thể được thiết đặt như $qPi - (qPiMaxIdx - QpCIIdx[qPiMaxIdx])$.

Theo đó, QpC có thể được thiết đặt như QpCIIdx[qPi].

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không hoặc liệu thông tin được báo hiệu có được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay không. Cờ có thể được báo hiệu thông qua cấu trúc cấp cao, như tập hợp thông số chuỗi (SPS), hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cấu trúc cấp cao có thể là giống như bảng sau đây.

【Bảng 40】

<code>seq_parameter_set_rbsp()</code> {	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
...	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
...	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng cho việc dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Khi `Qpc_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng cho việc dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy ra là 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất thông số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

【Bảng 41】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 &\text{minTbAddrB} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - QpBdOffset_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 41, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo đúng (true) (tức là, ví dụ, khi Qpc_data_default_flag là 1), thì các thông số qPCb, qPCr và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPiCb, qPiCr và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa.

Ví dụ, phương án này đề xuất ví dụ về cấu trúc báo hiệu dữ liệu cho việc dẫn xuất QP sắc độ. Cụ thể, phương án này đề xuất sơ đồ để bổ sung chroma_qp_mapping_flag mà là phân tử cú pháp mới trong SPS. Ví dụ, nếu giá trị chroma_qp_mapping_flag là 0, thì bảng ánh xạ QP sắc độ mặc định có thể được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Thêm nữa, ví dụ, nếu giá trị chroma_qp_mapping_flag là 1, thì các phân tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ có thể được báo hiệu như trong bảng sau đây.

【Bảng 42】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!QpC_data_default_flag)</code>	
<code>qPi_delta_max_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>for (i = 1; i <= qPi_delta_max_idx_minus1 + 1; i++) {</code>	
<code>QpC_qPi_flag[i]</code>	<code>ue(v)</code>
<code>...</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `QpC_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 42 như được mô tả ở trên được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Nếu `QpC_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 42 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, `QpC_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng mặc định trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được suy ra là 1.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_delta_max_idx_minus1` có thể biểu diễn số lượng các điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn phần tử thứ nhất của tập điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn giá trị delta giữa phần tử thứ *i* và phần tử thứ (*i*-1) của tập điểm mà có hàm ánh xạ không được tăng.

Bảng ánh xạ QP sắc độ có thể được dẫn xuất như sau dựa trên dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 42.

Ví dụ, biến `cQpFlatSize` có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 5】

$$cQpFlatSize = qPi_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, biến `cQpFlat[]` có thể được dẫn xuất như trong bảng sau đây.

【Bảng 43】

The variable `cQpFlat[]` is derived as follows:

```
cQpFlat[ 0 ] = qPi_min_idx_minus1 + 1;
for( i = 1; i < cQpFlatSize; i++ ) {
    cQpFlat[ i ] = QpC_qPi_flag [ i ] + cQpFlat[ i - 1 ]
}
```

Sau đó, dựa trên biến `cQpFlatSize` và biến `cQpFlat[]`, bảng ánh xạ QP sắc độ có thể được dẫn xuất như trong bảng sau đây.

【Bảng 44】

```

The chroma QP mapping table cqpMappingTable[] is derived as follows:
cqpMappingTable [ 0 ] = 0;
for( i = 1; i <= maxQP; i++ ) {
    incStep = 1
    for ( j = 0; j < cQpFlatSize; j++){
        if ( i == cQpFlat[ j ] )
            incStep = 0
    }
    cqpMappingTable [ i ] = cqpMappingTable [ i - 1 ] + incStep;

```

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa.

Ví dụ, phương án này đề xuất sơ đồ để bổ sung `chroma_qp_mapping_flag` mà là phần tử cú pháp mới trong SPS. Ví dụ, nếu giá trị `chroma_qp_mapping_flag` là 0, thì bảng ánh xạ QP sắc độ mặc định có thể được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Thêm nữa, ví dụ, nếu giá trị `chroma_qp_mapping_flag` là 1, thì các phần tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ có thể được báo hiệu như trong bảng sau đây.

【Bảng 45】

	Descriptor
<code>seq_parameter_set_rbsp() {</code>	
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>qPi_delta_max_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>for (i = 1; i <= qPi_delta_max_idx_minus1 + 1; i++) {</code>	
<code>Qpc_qPi_idx_minus1[i]</code>	<code>uc(v)</code>
<code>...</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `QpC_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 42 như được mô tả ở trên được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Nếu `QpC_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 42 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, `QpC_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được suy ra là 1.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_delta_max_idx_minus1` có thể biểu diễn số lượng các điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn phần tử thứ nhất của tập điểm mà có bảng ánh xạ không được tăng.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `QpC_qPi_idx_minus1[i]` có thể biểu diễn giá trị delta giữa phần tử thứ *i* và phần tử thứ (*i*-1) của tập điểm mà có hàm ánh xạ không được tăng.

Bảng ánh xạ QP sắc độ có thể được dẫn xuất như sau dựa trên dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 45.

Ví dụ, biến `cQpFlatSize` có thể được dẫn xuất như trong phương trình 5 như được mô tả ở trên.

Thêm nữa, ví dụ, biến `cQpFlat[]` có thể được dẫn xuất như trong bảng sau đây.

【Bảng 46】

```
The variable cQpFlat[] is derived as follows:
cQpFlat[ 0 ] = qPi_min_idx_minus1 + 1;
for( i = 1; i < cQpFlatSize; i++ ) {
    cQpFlat[ i ] = QpC_qPi_idx_minus1[ i ] + 1 + cQpFlat[ i - 1 ]
}
```

Sau đó, dựa trên biến `cQpFlatSize` và biến `cQpFlat[]`, thì bảng ánh xạ QP sắc độ có thể được dẫn xuất. Ví dụ, bảng ánh xạ QP sắc độ có thể được dẫn xuất như trong bảng 44 như được mô tả ở trên.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa.

Ví dụ, phương án này đề xuất sơ đồ để báo hiệu các bảng riêng lẻ cho các thành phần sắc độ tương ứng. Tức là, ví dụ, phương án này đề xuất sơ đồ để báo hiệu các phần tử cú pháp được sử dụng để dẫn xuất các bảng ánh xạ QP sắc độ cho các thành phần sắc độ tương ứng.

Ví dụ, các bảng ánh xạ QP sắc độ cho các thành phần sắc độ tương ứng có thể được dẫn xuất, và các phần tử cú pháp cho các thành phần sắc độ tương ứng có thể được báo hiệu như trong bảng sau đây.

【Bảng 47】

<code>Qp_C_data() {</code>	Descriptor
<code>qp_luma_to_chroma_joint_map_flag</code>	<code>u(1)</code>
<code>for(i = 0; i < (qp_luma_to_chroma_joint_map_flag ? 1 : 3); i++) {</code>	
<code>qPi_min_idx_minus1[i]</code>	<code>uc(v)</code>
<code>qPi_delta_max_idx_minus1[i]</code>	<code>ue(v)</code>
<code>for (j = qPi_min_idx_minus1 + 1 ; j <= qPiMaxIdx; i++)</code>	
<code>Qp_C_qPi_flag[i][j]</code>	<code>u(1)</code>
<code>}</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `qp_luma_to_chroma_joint_map_flag` có thể biểu diễn liệu bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ có được sử dụng cho các thành phần sắc độ Cb, Cr, và CbCr hay không. Tức là, ví dụ, phần tử cú pháp `qp_luma_to_chroma_joint_map_flag` có thể biểu diễn liệu một bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ được áp dụng cho phần dư Cb, phần dư Cr, và phần dư CbCr hay không. Ví dụ, nếu giá trị của `qp_luma_to_chroma_joint_map_flag` là 1, thì bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ chung có thể được sử dụng cho các thành phần sắc độ Cb, Cr, và CbCr, và nếu giá trị của `qp_luma_to_chroma_joint_map_flag` là 0, thì bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ riêng lẻ có thể được sử dụng cho mỗi thành phần sắc độ trong số các thành phần sắc độ Cb, Cr, và CbCr.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị `qPi_min_idx_minus1` có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_delta_max_idx_minus1` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` tối đa được sử dụng cho việc dẫn xuất sắc độ `QpC`. Giá trị `qPiMaxIdx` có thể là bằng

hoặc lớn hơn giá trị qPi_min_idx . Ví dụ, giá trị $qPi_delta_max_idx_minus1$ có thể là trong phạm vi từ 1 đến 63. Chỉ số tối đa $qPiMaxIdx$ được sử dụng cho việc dẫn xuất QpC có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 6】

```
for( i = 0; i < 3; i++) {  
     $qPiMaxIdx[i] = qPi\_min\_idx\_minus1[i] + 1 + qPi\_delta\_max\_idx\_minus1[i] + 1$ 
```

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_flag[j]$ có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $QpC_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị QpC thứ j của thành phần sắc độ thứ i được tăng thêm 1 hay không so với giá trị QpC thứ $(j-1)$. Ví dụ, $QpC_qPi_flag[j]$ bằng 1 có thể biểu diễn rằng giá trị QpC được tăng thêm 1, và $QpC_qPi_flag[j]$ bằng 0 có thể biểu diễn rằng giá trị QpC không được tăng.

Ví dụ, biến $QpCIdx[i][qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là $maxQp$ trong trường hợp giá trị là 0.

- Trong trường hợp $qPi < qPi_min_idx_minus1+1$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình để là giống như qPi .

- Trong trường hợp $qPi = qPi_min_idx_minus1+1 \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, giá trị QpC có thể được dẫn xuất như $QpCIdx[i][qPi]$.

Trong khi đó, theo phương án này, cờ biểu diễn liệu các phần tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ trong SPS được báo hiệu hay bảng mặc định

được sử dụng có thể được báo hiệu. Ví dụ, cờ có thể được báo hiệu như trong bảng sau đây.

【Bảng 48】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code> sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code> ...</code>	
<code> Qpc_data_default_flag</code>	<code>u(1)</code>
<code> if(!Qpc_data_default_flag)</code>	
<code> Qpc_data()</code>	
<code> ...</code>	

Phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 47 như được mô tả ở trên được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ được minh họa trong bảng 47 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. bảng mặc định có thể là giống như bảng mặc định trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy ra là 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 49】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 49 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch tối đa giữa điểm bắt đầu và điểm kết thúc như delta cho QP tối đa. Tức là, ví dụ, theo phương án này, phần tử cú pháp biểu diễn giá trị delta giữa maxQp được sử dụng để dẫn xuất sắc độ QpC và chỉ số qPi tối đa có thể được báo hiệu.

Dữ liệu thông số lượng tử hóa sắc độ QpC_data() cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 50】

QpC_data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
QpC_qPi_flag[i]	u(1)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `maxQp` và chỉ số `qPi` tối đa được sử dụng để dẫn xuất sắc độ `QpC`. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn giá trị `qPi_min_idx`. Giá trị `qPi_delta_max_idx` có thể là trong phạm vi từ 1 đến 63. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng để dẫn xuất `QpC` có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 7】

$$qPiMaxIdx = maxQp - qPi_delta_max_idx$$

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` thứ `i` được tăng thêm 1 hay không so với giá trị `QpC` thứ `(i-1)`. Ví dụ, `QpC_qPi_flag[i]` có giá trị là 1 có thể biểu diễn rằng giá trị `QpC` được tăng thêm 1, và `QpC_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `QpC` không được tăng.

Ví dụ, biến `QpCIdx[qPi]` có thể được dẫn xuất như sau. Ở đây, `qPi` có thể là từ 0 đến 63.

- Trong trường hợp `qPi < qPi_min_idx`, thì `QpCIdx[qPi]` có thể được tạo cấu hình giống như `qPi`.

- Trong trường hợp `qPi = qPi_min_idx...qPiMaxIdx`, thì `QpCIdx[qPi]` có thể được tạo cấu hình như `QpC_qPi_flag[qPi] + QpCIdx[qPi-1]`.

- Trong trường hợp `qPi > qPiMaxIdx`, thì `QpCIdx[qPi]` có thể được tạo cấu hình như `qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])`.

Sau đó, `QpC` có thể được tạo cấu hình như `QpCIdx[qPi]`.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hoặc thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ hay không sẽ được sử dụng. Cờ có thể được báo hiệu thông qua cấu trúc cấp cao, như tập hợp thông số chuỗi (SPS), hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là như trong bảng sau đây.

【Bảng 51】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!QpC_data_default_flag)</code>	
<code>QpC_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `QpC_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` được sử dụng. Nếu `QpC_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `QpC_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được suy ra là 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 52】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg - 1, yQg) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location (xQg - 1, yQg) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (xQg - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= yQg \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg - 1, yQg).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg, yQg - 1) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 52 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch tối đa giữa điểm bắt đầu và điểm kết thúc bằng cách báo hiệu điểm kết thúc như delta cho QP tối đa hoặc như mức chênh lệch giữa điểm bắt đầu và giá trị thu được bằng cách cộng delta vào điểm bắt đầu.

Dữ liệu thông số lượng tử hóa sắc độ QpC_data() cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 53】

QpC_data() {	Descriptor
qPi_min_idx_minus1	ue(v)
is_delta_maxQp	u(1)
qPi_delta_max_idx_minus1	ue(v)
for (i = qPi_min_idx_minus1 + 1; i <= qPiMaxIdx; i++)	
QpC_qPi_flag[i]	u(1)
}	

Ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn chỉ số qPi tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến maxQp.

Thêm nữa, ví dụ, phần tử cú pháp `is_delta_maxQp` có thể biểu diễn liệu chỉ số tối đa `qPiMaxIdx` được dẫn xuất từ giá trị `maxQp`. Ví dụ, `is_delta_maxQp` có giá trị là 1 có thể biểu diễn rằng `qPiMaxIdx` được dẫn xuất từ giá trị `maxQp`. Thêm nữa, ví dụ, `is_delta_maxQp` có giá trị là 0 có thể biểu diễn rằng `qPiMaxIdx` được dẫn xuất từ phần tử cú pháp `qPi_min_idx_minus1`.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_delta_max_idx_minus1` có thể biểu diễn giá trị delta giữa `maxQp` và chỉ số `qPi` tối đa được sử dụng để dẫn xuất sắc độ `QpC`. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Ví dụ, giá trị `qPi_delta_max_idx_minus1` có thể là trong phạm vi từ 1 đến 63. Chỉ số tối đa `qPiMaxIdx` được sử dụng cho việc dẫn xuất `QpC` có thể được dẫn xuất như trong bảng sau đây.

【Bảng 54】

```
index qPiMaxIdx used in QpC derivation is derived as follows:
if(is_delta_maxQp){
    qPiMaxIdx = maxQp - qPi_delta_max_idx
} else {
    qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx
}
```

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` thứ `i` được tăng thêm 1 hay không so với giá trị `QpC` thứ `(i-1)`. Ví dụ, `QpC_qPi_flag[i]` có giá trị là 1 có thể biểu diễn rằng giá trị `QpC` được tăng thêm 1, và `QpC_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `QpC` không được tăng.

Ví dụ, biến `QpCIdx[qPi]` có thể được dẫn xuất như sau. Ở đây, `qPi` có thể là từ 0 đến `maxQp`.

- Trong trường hợp $qPi < qPi_min_idx_minus1+1$, thì $QpCIIdx[qPi]$ có thể được tạo cấu hình giống như qPi .

- Trong trường hợp $qPi = qPi_min_idx_minus1 \dots qPiMaxIdx$, thì $QpCIIdx[qPi]$ có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như $QpCIIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hoặc thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ hay không sẽ được sử dụng. Cờ có thể được báo hiệu thông qua cú pháp cấp cao như tập hợp thông số chuỗi (SPS) hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là như trong bảng sau đây.

【Bảng 55】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>QpC_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `QpC_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử

hóa sắc độ `QpC_data()` được sử dụng. Nếu `QpC_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `QpC_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được suy ra là 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 56】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (CtbLog2SizeY - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 56 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch tối đa giữa điểm bắt đầu và điểm kết thúc bằng cách báo hiệu điểm kết thúc như delta cho QP tối đa hoặc như mức chênh lệch giữa điểm bắt đầu và giá trị thu được bằng cách cộng delta vào điểm bắt đầu.

Dữ liệu thông số lượng tử hóa sắc độ QpC_data() cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 57】

QpC_data() {	Descriptor
qPi_min_idx_minus1	ue(v)
is_delta_maxQp	u(1)
qPi_delta_max_idx_minus1	uc(v)
for (i = qPi_min_idx_minus1 + 1; i <= qPiMaxIdx; i++)	
QpC_qPi_flag[i]	u(1)
}	

Ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn chỉ số qPi tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến maxQp.

Thêm nữa, ví dụ, phần tử cú pháp `is_delta_maxQp` có thể biểu diễn liệu chỉ số tối đa `qPiMaxIdx` được dẫn xuất từ giá trị `maxQp`. Ví dụ, `is_delta_maxQp` có giá trị là 1 có thể biểu diễn rằng `qPiMaxIdx` được dẫn xuất từ giá trị `maxQp`. Thêm nữa, ví dụ, `is_delta_maxQp` có giá trị là 0 có thể biểu diễn rằng `qPiMaxIdx` được dẫn xuất từ phần tử cú pháp `qPi_min_idx_minus1`.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_delta_max_idx_minus1` có thể biểu diễn giá trị delta giữa `maxQp` và chỉ số `qPi` tối đa được sử dụng để dẫn xuất sắc độ `QpC`. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Ví dụ, giá trị `qPi_delta_max_idx_minus1` có thể là trong phạm vi từ 1 đến 63. Chỉ số tối đa `qPiMaxIdx` được sử dụng để dẫn xuất `QpC` có thể được dẫn xuất như trong bảng 54 như được mô tả ở trên.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `QpC` thứ `i` được tăng thêm 1 hay không so với giá trị `QpC` thứ `(i-1)`. Ví dụ, `QpC_qPi_flag[i]` có giá trị là 1 có thể biểu diễn rằng giá trị `QpC` được tăng thêm 1, và `QpC_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `QpC` không được tăng.

Ví dụ, biến `QpCIdx[qPi]` có thể được dẫn xuất như sau. Ở đây, `qPi` có thể là từ 0 đến `maxQp`.

- Trong trường hợp $qPi < qPi_min_idx_minus1 + 1$, thì `QpCIdx[qPi]` có thể được tạo cấu hình giống như `qPi`.

- Trong trường hợp $qPi = qPi_min_idx_minus1 \dots qPiMaxIdx$, thì `QpCIdx[qPi]` có thể được tạo cấu hình như `QpC_qPi_flag[qPi] + QpCIdx[qPi-1]`.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như $QpCIIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hoặc thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ hay không sẽ được sử dụng. Cờ có thể được báo hiệu thông qua cú pháp cấp cao như tập hợp thông số chuỗi (SPS) hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

【Bảng 58】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được

mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được coi như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 59】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (`xCb`, `yCb`) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable `cbWidth` specifying the width of the current coding block in luma samples,
- a variable `cbHeight` specifying the height of the current coding block in luma samples,
- a variable `treeType` specifying whether a single tree (`SINGLE_TREE`) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (`DUAL_TREE_LUMA`) or chroma components (`DUAL_TREE_CHROMA`) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (`xQg`, `yQg`), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions `xQg` and `yQg` are set equal to `CuQgTopLeftX` and `CuQgTopLeftY`, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREL} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables `CuQgTopLeftX` and `CuQgTopLeftY`.

When `treeType` is equal to `SINGLE_TREE` or `DUAL_TREE_LUMA`, the predicted luma quantization parameter qP_{Y_PREL} is derived by the following ordered steps:

1. The variable qP_{Y_PREL} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREL} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREL} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$x_{Tmp} = (x_{Qg} - 1) \gg \text{MinTbLog2SizeY}$$

$$y_{Tmp} = y_{Qg} \gg \text{MinTbLog2SizeY}$$

$$\text{minTbAddrA} = \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}]$$

$$\text{ctbAddrA} = \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY}))$$
 - Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 59 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu chỉ số cho bảng ánh xạ QP sắc độ bằng cách sử dụng hệ ký hiệu minus1 thay vì giá trị thực.

Dữ liệu thông số lượng tử hóa sắc độ QpC_data() cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 60】

QpC_data() {	Descriptor
qPi_min_idx_minus1	ue(v)
qPi_delta_max_idx_minus1	ue(v)
for (i = qPi_min_idx_minus1; i <= qPiMaxIdx; i++)	
QpC_qPi_flag[i]	u(1)
}	

Ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn chỉ số qPi tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp qPi_delta_max_idx_minus1 có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số

qPi tối đa được sử dụng để dẫn xuất sắc độ QpC. Giá trị qPiMaxIdx có thể là bằng hoặc lớn hơn qPi_min_idx. Giá trị qPi_delta_max_idx có thể là trong phạm vi từ 1 đến 63. Ví dụ, chỉ số tối đa qPiMaxIdx được sử dụng để dẫn xuất QpC có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 8】

$$qPiMaxIdx = qPi_min_idx_minus1 + 1 + qPi_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, phần tử cú pháp QpC_qPi_flag[i] có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp QpC_qPi_flag[i] có thể biểu diễn liệu giá trị QpC thứ i được tăng thêm 1 hay không so với giá trị QpC thứ (i-1). Ví dụ, QpC_qPi_flag[i] có giá trị là 1 có thể biểu diễn rằng giá trị QpC được tăng thêm 1, và QpC_qPi_flag[i] có giá trị là 0 có thể biểu diễn rằng giá trị QpC không được tăng.

Ví dụ, biến QpCIdx[qPi] có thể được dẫn xuất như sau. Ở đây, qPi có thể là từ 0 đến 63.

- Trong trường hợp $qPi < qPi_min_idx_minus1 + 1$, thì QpCIdx[qPi] có thể được tạo cấu hình giống như qPi.

- Trong trường hợp $qPi = qPi_min_idx_minus1 + 1 \dots qPiMaxIdx$, thì QpCIdx[qPi] có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì QpCIdx[qPi] có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như QpCIdx[qPi].

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hoặc thông tin được báo

hiệu để dẫn xuất việc lượng tử hóa sắc độ hay không sẽ được sử dụng. Cờ có thể được báo hiệu thông qua cú pháp cấp cao như tập hợp thông số chuỗi (SPS) hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

【Bảng 61】

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>QpC_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` được sử dụng. Nếu `QpC_data_default_flag` là 0, thì dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `QpC_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được coi như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 62】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned}
 x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\
 y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\
 ctbAddrA &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY}))
 \end{aligned}$$
 - Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, $qP_{Y\ B}$ is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y\ A} + qP_{Y\ B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 62 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Phương án này đề xuất sơ đồ trong đó bảng lượng tử hóa sắc độ riêng lẻ được sử dụng cho mỗi thành phần sắc độ trong các thành phần sắc độ.

Dữ liệu thông số lượng tử hóa sắc độ cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 63】

seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
.....	
Qpc_data_default_flag	u(1)
if(!Qpc_data_default_flag){	
sps_separate_qpc_table_flag	u(1)
qPi_cb_min_idx_minus1	u(7)
qPi_cb_delta_max_idx_minus1	u(7)
for(i = qPi_cb_min_idx_minus1+1; i <= qPiMaxIdxCb; i++)	
Qpc_cb_qPi_flag[i]	u(1)
if(sps_separate_qpc_table_flag) {	
qPi_cb_min_idx_minus1	u(7)
qPi_cb_delta_max_idx_minus1	u(7)
for(i = qPi_cr_min_idx_minus1+1; i <= qPiMaxIdxCr; i++)	
Qpc_cr_qPi_flag[i]	u(1)
}	
}	
.....	
}	

Ví dụ, phần tử cú pháp Qpc_data_default_flag có thể biểu diễn liệu bảng thông số lượng tử hóa sắc độ mặc định được sử dụng hay không. Ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng thông số lượng tử hóa sắc độ mặc định được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn rằng bảng thông số lượng tử hóa sắc độ mặc định không được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ. Tức là, ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn rằng bảng thông số lượng tử hóa sắc độ được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ được báo hiệu để dẫn xuất thông số lượng tử hóa sắc độ được sử dụng.

Thêm nữa, ví dụ, phần tử cú pháp `sps_separate_qpc_table_flag` có thể biểu diễn liệu hai bảng QpC riêng lẻ có được sử dụng cho mẫu Cb và mẫu Cr hay không. Tức là, ví dụ, phần tử cú pháp `sps_separate_qpc_table_flag` có thể biểu diễn liệu các bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ riêng lẻ có được sử dụng lần lượt cho phần dư Cb và phần dư Cr hay không. Ví dụ, `sps_separate_qpc_table_flag` có giá trị là 1 có thể biểu diễn rằng các bảng QpC riêng lẻ lần lượt được sử dụng cho mẫu Cb và mẫu Cr, và `sps_separate_qpc_table_flag` có giá trị là 0 có thể biểu diễn rằng một bảng QpC được sử dụng cho mẫu Cb và mẫu Cr.

Trong khi đó, ví dụ, biến `QpCb[i]` có thể biểu diễn bảng QpC được sử dụng cho mẫu Cb. Thêm nữa, ví dụ, biến `QpCr[i]` có thể biểu diễn bảng QpC được sử dụng cho mẫu Cr. Thêm nữa, ví dụ, nếu giá trị của `sps_separate_qpc_table_flag` là 0, thì `QpCr[i]` có thể là giống như `QpCb[i]`. Ở đây, `i` có thể là từ 0 đến 69.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_cb_min_idx_minus1` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng cho thành phần sắc độ Cb. Giá trị `qPi_cb_min_idx_minus1` có thể là trong phạm vi từ 1 đến 69.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 và phần tử cú pháp `qPi_cb_delta_max_idx_minus1` có thể biểu diễn giá trị delta giữa `qPi_cb_min_idx_minus1` và `qPi_cb_delta_max_idx_minus1` lớn nhất được sử dụng để dẫn xuất sắc độ QpC Cb. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Giá trị `qPi_cb_delta_max_idx_minus1` có thể là trong phạm vi từ 1 đến 69. Ví dụ, chỉ số tối đa `qPiMaxIdxcb` được sử dụng để dẫn xuất QpC cho thành phần Cb có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 9】

$$qPiMaxIdxCb = qPi_cb_min_idx_minus1 + 1 + qPi_cb_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, phần tử cú pháp $QpC_cb_qPi_flag[i]$ có thể biểu diễn giá trị delta giữa giá trị QpC thứ i $QpCb[i]$ cho thành phần Cb và giá trị QpC thứ $(i-1)$ $QpCb[i-1]$. Giá trị $QpC_cb_qPi_flag[i]$ có thể là trong phạm vi từ 0 đến 1.

Ví dụ, biến $QpCb[i]$ có thể được dẫn xuất như sau. Ở đây, i có thể là từ 0 đến 69.

- Trong trường hợp là $i = 0.. qPiMaxIdxCb$, thì $QpCb[i]$ có thể được tạo cấu hình giống như i .

- Trong trường hợp là $i = qPi_cb_min_idx_minus1 + 1 + 1.. qPiMaxIdxCb$, thì $QpCb[i]$ có thể được tạo cấu hình như $QpCb[i-1] + QpC_cb_qPi_flag[i]$.

- Trong trường hợp là $i = qPiMaxIdxCb + 1..69$, thì $QpCb[i]$ có thể được tạo cấu hình như $i - deltaEnd$, và $deltaEnd$ có thể được dẫn xuất như $qPiMaxIdxCb - QpCb[qPiMaxIdxCb]$.

Thêm nữa, ví dụ, $qPi_cr_min_idx_minus1$, $qPiMaxIdxCr$, và $QpC_cr_qPi_flag[i]$, vốn là các phần tử cú pháp cho thành phần Cr , có thể có cùng nghĩa như nghĩa của các phần tử cú pháp cho thành phần Cb .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về thông số lượng tử hóa. Ví dụ, phương án này đề xuất sơ đồ để báo hiệu các thông số cho nhiều bảng QP sắc độ. Thêm nữa, phương án này có thể được kết hợp với ít nhất một trong các phương án như được mô tả ở trên. Tức là, ví dụ, các phương án của sáng chế này có thể được áp dụng chung.

Cụ thể, ví dụ, phương án này đề xuất để bao gồm thông số lượng tử hóa sắc độ được định rõ bởi người dùng (QpC) trong tài liệu chỉ rõ VVC. Ví dụ, theo phương án này, cờ của tập hợp thông số chuỗi (Sequence Parameter Set, SPS) có thể biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa sắc độ hay bảng ánh xạ

sắc độ QP được dẫn xuất dựa trên thông tin được báo hiệu trong SPS. Thông qua điều này, thông số lượng tử hóa sắc độ được định rõ bởi người dùng có thể được sử dụng xét đến các đặc tính của ảnh trong khi tạo mã hình ảnh, và hiệu quả lập mã có thể được cải thiện. Thêm nữa, phương án này có thể cung cấp sự linh hoạt thông qua chọn trong đó một bảng được định rõ bởi người dùng được sử dụng cho các thành phần sắc độ và chọn trong đó các bảng được định rõ bởi người dùng riêng biệt được sử dụng cho thành phần Cb và thành phần Cr.

Ví dụ, dữ liệu thông số lượng tử hóa sắc độ `QpC_data()` cho thông số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 64】

<code>QpC_data() {</code>	Descriptor
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_delta_max_idx</code>	<code>ue(v)</code>
<code>for (i = qPi_min_idx; i <= qPiMaxIdx; i++)</code>	
<code> QpC_qPi_flag[i]</code>	<code>u(1)</code>
<code>}</code>	

Ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn chỉ số `qPi` tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị `qPi_min_idx` có thể là trong phạm vi từ 1 đến 69.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` tối đa được sử dụng để dẫn xuất sắc độ QpC. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Giá trị `qPi_delta_max_idx` có thể là trong phạm vi từ 0 đến 69. Ví dụ, chỉ số tối đa `qPiMaxIdx` được sử dụng để dẫn xuất QpC có thể được dẫn xuất như trong phương trình 4 như được mô tả ở trên.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC thứ i được tăng thêm 1 hay không so với giá trị QpC thứ $(i-1)$. Ví dụ, $QpC_qPi_flag[i]$ có giá trị là 1 có thể biểu diễn rằng giá trị QpC được tăng thêm 1, và $QpC_qPi_flag[i]$ có giá trị là 0 có thể biểu diễn rằng giá trị QpC không được tăng.

Ví dụ, biến $QpCIdx[qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là từ 0 đến 69.

- Trong trường hợp $qPi < qPi_min_idx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình giống như qPi .

- Trong trường hợp $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như $QpCIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ hay bảng ánh xạ QP sắc độ được dẫn xuất dựa trên thông tin được báo hiệu được sử dụng. Cờ có thể được báo hiệu thông qua cú pháp cấp cao như tập hợp thông số chuỗi (SPS) hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là như trong bảng sau đây.

【Bảng 65】

seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
...	
Qpc_data_default_flag	u(1)
if(!QpC_data_default_flag)	
QpC_data()	
...	

Ví dụ, phần tử cú pháp QpC_data_default_flag có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, QpC_data_default_flag có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, QpC_data_default_flag có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ QpC_data() được sử dụng. Nếu QpC_data_default_flag là 0, thì dữ liệu thông số lượng tử hóa sắc độ QpC_data() có thể được báo hiệu. Thêm nữa, ví dụ, QpC_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu QpC_data_default_flag là không có mặt, thì QpC_data_default_flag có thể được suy ra là 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 66】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 &\text{minTbAddrB} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 66 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ QpC_data() trong trường hợp sử dụng các bảng được định rõ bởi người dùng cho các thành phần sắc độ tương ứng được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

【Bảng 67】

QpC_data() {	Descriptor
is_separate_chroma_table	u(1)
for (i = 0; i < is_separate_chroma_table ? 2 : 1; i++) {	
qPi_min_idx_minus1[i]	ue(v)
qPi_delta_max_idx[i]	uc(v)
for (j = qPi_min_idx[i]; j <= qPiMaxIdx[i]; j++)	
QpC_qPi_flag[i][j]	u(1)
}	
}	

Ví dụ, phần tử cú pháp is_separate_chroma_table có thể biểu diễn liệu các thông số liên quan đến bảng lượng tử hóa sắc độ riêng biệt được báo hiệu cho thành phần Cb và thành phần Cr. Tức là, ví dụ, phần tử cú pháp is_separate_chroma_table có thể biểu diễn liệu hai bảng ánh xạ thông số lượng tử hóa sắc độ riêng biệt được sử dụng cho thành phần Cb và thành phần Cr. Ví dụ, phần tử cú pháp is_separate_chroma_table có thể biểu diễn liệu các bảng ánh xạ thông số lượng tử hóa độ chói-sắc độ riêng lẻ có được sử dụng

lần lượt cho phần dư Cb và phần dư Cr hay không. Ví dụ, `is_separate_chroma_table` có giá trị là 1 có thể biểu diễn rằng bảng ánh xạ thông số lượng tử hóa sắc độ riêng biệt được báo hiệu cho thành phần Cb và thành phần Cr, và `is_separate_chroma_table` có giá trị là 0 có thể biểu diễn rằng một bảng ánh xạ thông số lượng tử hóa sắc độ được sử dụng cho thành phần Cb, thành phần Cr, và các thành phần CbCr liên kết. Ví dụ, nếu `is_separate_chroma_table` giá trị là 1, thì `Pi_min_idx_minus1[i]`, `qPi_delta_max_idx[i]`, và `QpC_qPi_flag[i][j]` cho thành phần Cb, và `qPi_min_idx_minus1[i]`, `qPi_delta_max_idx[i]`, và `QpC_qPi_flag[i][j]` cho thành phần Cr có thể được báo hiệu. Thêm nữa, ví dụ, nếu giá trị của `is_separate_chroma_table` là 0, thì `qPi_min_idx_minus1[i]`, `qPi_delta_max_idx[i]`, và `QpC_qPi_flag[i][j]` cho thành phần Cb, thành phần Cr, và thành phần CbCr liên kết có thể được báo hiệu.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn chỉ số qPi tối thiểu được sử dụng cho việc lượng tử hóa sắc độ. Giá trị `qPi_min_idx` có thể là trong phạm vi từ 1 đến 69. Biến `qPi_min_idx[i]` có thể được tạo cấu hình giống như giá trị thu được bằng cách cộng 1 vào `qPi_min_idx_minus1[i]`.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx[i]` và chỉ số qPi tối đa được sử dụng để dẫn xuất QpC sắc độ. Giá trị `qPiMaxIdx[i]` có thể là bằng hoặc lớn hơn `qPi_min_idx[i]`. Giá trị `qPi_delta_max_idx` có thể là trong phạm vi từ 0 đến 69. Ví dụ, chỉ số tối đa `qPiMaxIdx[i]` được sử dụng để dẫn xuất QpC có thể được dẫn xuất như trong phương trình sau đây.

【Phương trình 10】

$$qPiMaxIdx[i] = qPi_min_idx[i] + qPi_delta_max_idx_minus1[i] + 1$$

Giá trị `qPiMaxIdx[i]` có thể là bằng hoặc lớn hơn `qPi_min_idx_minus1[i]`.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị QpC thứ j có thành phần sắc độ thứ i được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $QpC_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị QpC thứ j của thành phần sắc độ thứ i được tăng thêm 1 hay không so với giá trị QpC thứ $(j-1)$ hay không. Ví dụ, $QpC_qPi_flag[j]$ có giá trị là 1 có thể biểu diễn rằng giá trị QpC thứ j của thành phần sắc độ thứ i được tăng thêm 1 hay không, và $QpC_qPi_flag[j]$ có giá trị là 0 có thể biểu diễn rằng giá trị QpC thứ j của thành phần sắc độ thứ i không được tăng.

Ví dụ, biến $QpCIdx[i][qPi]$ có thể được dẫn xuất như trong bảng sau đây. Ở đây, qPi có thể là từ 0 đến 69.

【Bảng 68】

```

For i=0; i<is_separate_chroma_table ? 2 : 1, i++
  The variable  $QpCIdx[i][qPi]$  for  $qPi$  with  $qPi = 0 \dots 69$ , is derived as follows:
  - For  $qPi < qPi\_min\_idx[i]$ ,  $QpCIdx[i][qPi]$  is set equal to  $qPi$ .
  - For  $qPi = qPi\_min\_idx[i] \dots qPiMaxIdx[i]$ , the following applies:
       $QpCIdx[i][qPi] = QpC\_qPi\_flag[i][qPi] + QpCIdx[i][qPi - 1]$ 
  - For  $qPi > qPiMaxIdx[i]$ 
       $QpCIdx[i][qPi] = qPi - (qPiMaxIdx[i] - QpCIdx[i][qPiMaxIdx[i]])$ 

  The value of  $QpC$  of the  $i$ th chroma component is derived as  $QpCIdx[i][qPi]$ .

```

Tham chiếu đến bảng 68, nếu giá trị của $is_separate_chroma_table$ là 1, thì dữ liệu thông số lượng tử hóa sắc độ thứ 0 ($i=0$) và dữ liệu thông số lượng tử hóa sắc độ thứ nhất ($i=1$) có thể được báo hiệu. Ở đây, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể là dữ liệu thông số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ thông số lượng tử hóa sắc độ cho thành phần Cb, và dữ liệu thông số lượng tử hóa sắc độ thứ nhất ($i=1$) có thể là dữ liệu thông số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ thông số lượng tử hóa sắc độ cho thành phần Cr.

Thêm nữa, tham chiếu đến bảng 68, nếu giá trị `is_separate_chroma_table` là 0, thì chỉ dữ liệu thông số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể được báo hiệu. Ở đây, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể là dữ liệu thông số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ thông số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr, và thành phần CbCr kết hợp. Tức là, một bảng ánh xạ thông số lượng tử hóa sắc độ có thể được sử dụng cho các thành phần sắc độ.

Thêm nữa, tham chiếu đến bảng 68, `QpCIIdx[i][qPi]` có thể được dẫn xuất như sau.

- Trong trường hợp $qPi < qPi_min_idx[i]$, thì `QpCIIdx[i][qPi]` có thể được tạo cấu hình giống như `qPi`.
- Trong trường hợp $qPi = qPi_min_idx[i] \dots qPiMaxIdx[i]$, thì `QpCIIdx[qPi]` có thể được tạo cấu hình như `QpC_qPi_flag[i][qPi] + QpCIIdx[i][qPi-1]`.
- Trong trường hợp $qPi > qPiMaxIdx$, thì `QpCIIdx[i][qPi]` có thể được tạo cấu hình như $qPi - (qPiMaxIdx[i] - QpCIIdx[i][qPiMaxIdx])$.

Sau đó, giá trị `QpC` có thể được dẫn xuất như `QpCIIdx[i][qPi]`.

Thêm nữa, phương án này đề xuất sơ đồ báo hiệu cờ biểu diễn liệu bảng mặc định có được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hoặc thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ hay không sẽ được sử dụng. Cờ có thể được báo hiệu thông qua cú pháp cấp cao như tập hợp thông số chuỗi (SPS) hoặc tập hợp thông số hình ảnh (PPS). Cờ được báo hiệu thông qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

【Bảng 69】

seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
...	
Qpc_data_default_flag	u(1)
if(!QpC_data_default_flag)	
QpC_data()	
...	

Ví dụ, phần tử cú pháp QpC_data_default_flag có thể biểu diễn liệu chế độ được định rõ bởi người dùng có được sử dụng để dẫn xuất thông số lượng tử hóa hay không. Ví dụ, QpC_data_default_flag có giá trị là 0 có thể biểu diễn rằng chế độ được định rõ bởi người dùng được sử dụng để dẫn xuất thông số lượng tử hóa. Tức là, ví dụ, QpC_data_default_flag có giá trị là 0 có thể biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ QpC_data() được sử dụng. Nếu QpC_data_default_flag là 0, thì dữ liệu thông số lượng tử hóa sắc độ QpC_data() có thể được báo hiệu. Thêm nữa, ví dụ, QpC_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất thông số lượng tử hóa. Bảng mặc định có thể là như trong bảng 7 như được mô tả ở trên. Thêm nữa, nếu QpC_data_default_flag là không có mặt, thì QpC_data_default_flag có thể được suy ra là 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất thông số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

【Bảng 70】

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

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

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index $qPi[i]$ equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến bảng 70 như được mô tả ở trên, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu QpC_data_default_flag là 0), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất dựa trên thông tin được định rõ bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và QpC_data_default_flag biểu diễn ĐÚNG (TRUE) (ví dụ, nếu QpC_data_default_flag là 1), thì các biến qPCb, qPCr, và qPCbCr có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPiCb, qPiCr, và qPiCbCr.

Fig.7 thể hiện theo cách giản lược phương pháp mã hóa hình ảnh bởi thiết bị mã hóa theo sáng chế này. Phương pháp được bộc lộ trong Fig.7 có thể được thực hiện bằng thiết bị mã hóa được thực hiện bằng thiết bị mã hóa được bộc lộ trên Fig.2. Cụ thể, ví dụ, S700 của FIG.7 có thể được thực hiện bởi bộ dự đoán của thiết bị giải mã, S710 đến S730 của Fig.7 có thể được thực hiện bởi bộ xử lý phần dư của thiết bị mã hóa, và S740 có thể được thực hiện bởi bộ mã hóa entropi của thiết bị mã hóa. Thêm nữa, mặc dù không được minh họa, nhưng quy trình để tạo ra các mẫu được tái tạo và hình ảnh được tái tạo dựa trên các mẫu dư và các mẫu dự đoán có thể được thực hiện bởi bộ cộng của thiết bị mã hóa.

Thiết bị mã hóa dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên việc dự đoán liên ảnh hoặc việc dự đoán trong ảnh (S700). Thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên chế độ dự đoán. Tức là, ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán của khối hiện tại cho các thành phần sắc độ dựa trên chế độ dự đoán. Trong trường hợp này, các phương pháp dự đoán khác nhau được bộc lộ trong sáng chế này, như dự đoán liên ảnh hoặc dự đoán trong ảnh, có thể

được áp dụng. Các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr và/hoặc thành phần CbCr liên kết.

Ví dụ, thiết bị mã hóa có thể xác định liệu có thực hiện việc dự đoán liên ảnh hay dự đoán trong ảnh đối với khối hiện tại cho các thành phần sắc độ, và có thể xác định chế độ dự đoán liên ảnh cụ thể hay chế độ dự đoán trong ảnh cụ thể dựa trên chi phí RD. Theo chế độ được xác định, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán cho khối hiện tại.

Thiết bị mã hóa dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên các mẫu dự đoán (S710). Ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán thông qua việc trừ các mẫu gốc và các mẫu dự đoán của khối hiện tại cho các thành phần sắc độ trong hình ảnh hiện tại.

Thiết bị mã hóa tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không (S720). Thiết bị mã hóa có thể xác định liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, và có thể tạo ra cờ.

Ví dụ, thiết bị mã hóa tạo ra và mã hóa cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ dựa trên loại sắc độ. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType được mô tả ở trên. Ví dụ, khi giá trị của loại sắc độ không là 0, thì thiết bị mã hóa có thể tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không. Ví dụ, khi giá trị của loại sắc độ là 1, thì thiết bị mã hóa có thể tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Ở đây, nếu giá trị của loại sắc độ là 0, thì loại sắc độ có thể là định dạng đơn sắc, và nếu giá trị của loại sắc độ là 1, thì loại sắc độ có thể là định dạng 4:2:0. Nếu giá trị của loại sắc độ là 2, thì

loại sắc độ có thể là định dạng 4:2:2, và nếu giá trị của loại sắc độ là 3, loại sắc độ có thể là định dạng 4:4:4. Ví dụ, phần tử cú pháp cho cờ có thể là `qp_luma_to_chroma_joint_map_flag` cờ, `sps_separate_qpc_table_flag`, hoặc `is_separate_chroma_table` được mô tả ở trên.

Ví dụ, nếu giá trị của cờ là 1, thì cờ có thể biểu diễn rằng một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng bảng thông số lượng tử hóa sắc độ riêng lẻ được áp dụng cho mỗi thành phần sắc độ.

Thêm nữa, ví dụ, cờ có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thiết bị mã hóa tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ (S730). Thiết bị mã hóa có thể tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ.

Ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và/hoặc dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Trong khi đó, ví dụ, thiết bị mã hóa có thể tạo ra và mã hóa cờ cho phép CbCr liên kết biểu diễn dữ liệu thông số lượng tử hóa sắc độ liên kết cho thành phần CbCr liên kết là có mặt. Tức là, ví dụ, thiết bị mã hóa có thể xác định dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr là có mặt, và có thể tạo ra và mã hóa cờ cho phép CbCr liên kết. Thêm nữa, ví dụ, thiết bị mã hóa có thể tạo ra và mã hóa cờ cho phép CbCr liên kết biểu diễn dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết là có mặt dựa trên loại sắc độ hay không. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả ở trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị mã hóa có thể tạo ra được cờ cho phép CbCr kết hợp biểu diễn dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Ví dụ, nếu giá trị của loại sắc độ là 1, thì thiết bị mã hóa có thể tạo ra cờ cho phép CbCr kết hợp biểu diễn dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt. Thêm nữa, ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Trong trường hợp này, nếu giá trị của cờ là 0 (tức là, xác định được rằng nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), và giá trị của cờ cho phép CbCr kết hợp là 1 (tức là, xác định được rằng dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ nhất và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ nhất. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ nhất và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ số của bảng thông số lượng tử hóa sắc độ thứ nhất. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` hoặc `qPi_cb_min_idx_minus1` được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ hai và/hoặc các phần tử cú pháp cho các giá trị thông

số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ hai. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ hai và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ số của bảng thông số lượng tử hóa sắc độ thứ hai. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` hoặc `qPi_cb_min_idx_minus1` được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ ba và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ ba. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ ba và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ

số của bảng thông số lượng tử hóa sắc độ thứ ba. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là qPi_min_idx , $qPi_min_idx_minus1[i]$ hoặc $qPi_cb_min_idx_minus1$ được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là $qPi_delta_max_idx$, $qPi_cb_delta_max_idx_minus1$ hoặc $qPi_delta_max_idx[i]$. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là $QpC_qPi_val[i]$, $QpC_cb_qPi_flag[i]$ hoặc $QpC_qPi_flag[i][j]$ được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, khi giá trị của cờ là 1 (tức là, xác định được rằng một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr và thành phần CbCr kết hợp.

Thiết bị mã hóa sẽ mã hóa thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ và cờ (S740). Thiết bị mã hóa có thể mã hóa thông tin ảnh. Thông tin ảnh có thể bao gồm thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ và cờ.

Ví dụ, thiết bị mã hóa có thể tạo ra và mã hóa thông tin dự đoán cho khối hiện tại. Thông tin dự đoán có thể bao gồm thông tin chế độ dự đoán biểu diễn chế độ dự đoán của khối hiện tại cho các thành phần sắc độ. Thông tin ảnh có thể bao gồm thông tin dự đoán.

Thêm nữa, ví dụ, thiết bị mã hóa có thể mã hóa thông tin phần dư cho các mẫu dư. Ví dụ, thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi dựa trên các mẫu dư, và có thể tạo ra thông tin phần dư dựa trên các hệ số biến đổi. Ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu dư được lượng tử hóa bằng cách lượng tử hóa các mẫu dư dựa trên thông số lượng tử hóa sắc độ, dẫn xuất các hệ số biến đổi dựa trên các mẫu dư được lượng tử hóa, và tạo ra và mã hóa thông tin phần dư dựa trên các hệ số biến đổi. Thêm nữa, ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu dư được lượng tử hóa bằng cách lượng tử hóa các mẫu dư dựa trên thông số lượng tử hóa sắc độ, dẫn xuất các hệ số biến đổi bằng cách biến đổi các mẫu dư được lượng tử hóa, và tạo ra và mã hóa thông tin phần dư dựa trên các hệ số biến đổi.

Ví dụ, khi giá trị của c là 0, thì thiết bị mã hóa có thể dẫn xuất các mẫu dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu dư cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu dư cho thành phần Cr bằng cách lượng tử hóa các mẫu dư cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của c là 0, thì thiết bị mã hóa có thể dẫn xuất các mẫu dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu dư cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, có thể dẫn xuất các mẫu dư được lượng tử hóa cho thành phần Cr bằng cách lượng tử hóa các mẫu dư cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và có thể dẫn xuất các mẫu dư được lượng tử hóa cho thành phần CbCr kết hợp bằng cách lượng tử hóa các mẫu dư cho thành phần CbCr kết hợp dựa trên thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví dụ, khi giá trị của c là 1, thì thiết bị mã hóa có thể dẫn xuất các mẫu dư được lượng tử hóa cho các thành phần sắc độ bằng cách lượng tử hóa các mẫu dư cho thành phần sắc độ dựa trên thông số lượng tử hóa sắc độ.

Thông tin hình ảnh có thể gồm thông tin phần dư. Ví dụ, thông tin phần dư có thể bao gồm các phần tử cú pháp cho các hệ số biến đổi của khối sắc độ hiện tại. Ví dụ, các phần tử cú pháp có thể bao gồm các phần tử cú pháp, như `coded_sub_block_flag`, `sig_coeff_flag`, `coeff_sign_flag`, `abs_level_gt1_flag`, `par_level_flag`, `abs_level_gtX_flag`, `abs_remainder`, và/hoặc `coeff_sign_flag`.

Thêm nữa, ví dụ, thiết bị mã hóa có thể mã hóa dữ liệu thông số lượng tử hóa sắc độ và cờ. Thông tin ảnh có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ và cờ.

Thiết bị mã hóa có thể mã hóa thông tin ảnh bao gồm dữ liệu thông số lượng tử hóa sắc độ, cờ và/hoặc cờ cho phép CbCr kết hợp.

Thiết bị mã hóa có thể mã hóa và xuất ra thông tin ảnh trong dạng luồng bit.

Trong khi đó, luồng bit bao gồm thông tin ảnh có thể được truyền đến thiết bị giải mã thông qua mạng hoặc phương tiện lưu trữ (số). Ở đây, mạng có thể bao gồm mạng quảng bá và/hoặc mạng truyền thông, và phương tiện lưu trữ kỹ thuật số có thể bao gồm các loại phương tiện lưu trữ khác nhau như đĩa USB, SD, CD, DVD, đĩa Blu-ray, HDD, và SSD.

Fig.8 thể hiện theo cách giản lược thiết bị mã hóa để thực hiện phương pháp mã hóa hình ảnh theo sáng chế này. Phương pháp được bộc lộ trong Fig.7 có thể được thực hiện bằng thiết bị mã hóa được thực hiện bằng thiết bị mã hóa được bộc lộ trên Fig.8. Cụ thể, ví dụ, bộ dự đoán của thiết bị mã hóa của Fig.8 có thể thực hiện S700, bộ xử lý phần dư của thiết bị mã hóa có thể thực hiện S710 đến S730, bộ mã hóa entropi của thiết bị mã hóa có thể thực hiện S740. Thêm nữa, mặc dù không được minh họa, nhưng quy trình tạo ra các mẫu được tái tạo và hình ảnh được tái tạo dựa trên các mẫu dư và các mẫu dự đoán có thể được thực hiện bởi bộ cộng của thiết bị mã hóa.

Fig.9 thể hiện theo cách giản lược phương pháp giải mã hình ảnh bởi thiết bị giải mã theo sáng chế này. Phương pháp được bộc lộ trong Fig.9 có thể được thực hiện bởi thiết bị giải mã được bộc lộ ở Fig.3. Cụ thể, ví dụ, S900 đến S910 của Fig.9 có thể được thực hiện bởi bộ giải mã entropi của thiết bị giải mã, S940 của Fig.9 có thể được thực hiện bởi bộ dự đoán của thiết bị giải mã, S920 đến S930 và S950 đến S950 của Fig.9 có thể được thực hiện bởi bộ xử lý phần dư của thiết bị giải mã, và S970 của Fig.9 có thể được thực hiện bởi bộ cộng của thiết bị giải mã.

Thiết bị giải mã thu được thông tin ảnh bao gồm thông tin dự đoán và thông tin phần dư cho các thành phần sắc độ và cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không (S900). Thiết bị giải mã có thể thu được thông tin ảnh thông qua luồng bit. Ví dụ, thông tin ảnh có thể bao gồm thông tin về thông số lượng tử hóa sắc độ.

Ví dụ, thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ hay không. Tức là, ví dụ, thông tin ảnh có thể bao gồm cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Ví dụ, thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ dựa trên loại sắc độ. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả ở trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng. Ví dụ, nếu giá trị của loại sắc độ là 1, thì thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ được áp dụng. Ở đây, nếu giá trị của loại sắc độ là 0, loại sắc độ có thể là định dạng đơn sắc, và nếu giá trị của loại sắc độ là 1, thì loại sắc độ có thể là định dạng 4:2:0. Nếu giá trị của loại sắc độ là 2, thì loại sắc độ có thể là định dạng 4:2:2,

và nếu giá trị của loại sắc độ là 3, loại sắc độ có thể là định dạng 4:4:4. Thêm nữa, ví dụ, các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr và/hoặc thành phần CbCr kết hợp. Ví dụ, phần tử cú pháp cho cờ có thể là `qp_luma_to_chroma_joint_map_flag`, `sps_separate_qpc_table_flag`, hoặc `is_separate_chroma_table` được mô tả ở trên.

Ví dụ, nếu giá trị của cờ là 1, thì cờ có thể biểu diễn rằng một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng bảng thông số lượng tử hóa sắc độ riêng lẻ được áp dụng cho mỗi thành phần sắc độ.

Thêm nữa, ví dụ, cờ có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, thông tin ảnh có thể bao gồm thông tin dự đoán và/hoặc thông tin phần dư cho các thành phần sắc độ. Ví dụ, thông tin ảnh có thể bao gồm thông tin dự đoán cho các thành phần sắc độ, thông tin dự đoán có thể bao gồm thông tin chế độ dự đoán. Thông tin chế độ dự đoán có thể biểu diễn liệu dự đoán liên ảnh hay dự đoán trong ảnh được áp dụng cho khối hiện tại cho các thành phần sắc độ. Thêm nữa, ví dụ, thông tin phần dư có thể bao gồm các phần tử cú pháp cho các hệ số biến đổi của khối hiện tại cho các thành phần sắc độ. Ví dụ, các phần tử cú pháp có thể bao gồm các phần tử cú pháp, như `coded_sub_block_flag`, `sig_coeff_flag`, `coeff_sign_flag`, `abs_level_gt1_flag`, `par_level_flag`, `abs_level_gtX_flag`, `abs_remainder`, và/hoặc `coeff_sign_flag`.

Thiết bị giải mã thu được dữ liệu thông số lượng tử hóa sắc độ dựa trên cờ (S910). Thiết bị giải mã có thể thu được dữ liệu thông số lượng tử hóa sắc độ dựa trên cờ.

Ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và/hoặc dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Trong khi đó, ví dụ, thiết bị mã hóa có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr là có mặt. Ví dụ, thông tin ảnh có thể bao gồm cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt. Thêm nữa, ví dụ, thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt dựa trên loại sắc độ. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả ở trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt. Ví dụ, nếu giá trị của loại sắc độ là 1, thì thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt. Thêm nữa, ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Trong trường hợp này, nếu giá trị của cờ là 0 (tức là, cờ biểu diễn nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), và giá trị của cờ cho phép CbCr kết hợp là 1 (tức là, cờ cho phép CbCr kết hợp biểu diễn rằng dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt), dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ nhất và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ nhất. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ nhất và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ số của bảng thông số lượng tử hóa sắc độ thứ nhất. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` hoặc `qPi_cb_min_idx_minus1` được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu thông qua cú pháp cấp

cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ hai và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ hai. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ hai và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ số của bảng thông số lượng tử hóa sắc độ thứ hai. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là qPi_min_idx , $qPi_min_idx_minus1[i]$ hoặc $qPi_cb_min_idx_minus1$ được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là $qPi_delta_max_idx$, $qPi_cb_delta_max_idx_minus1$ hoặc $qPi_delta_max_idx[i]$. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là $QpC_qPi_val[i]$, $QpC_cb_qPi_flag[i]$ hoặc $QpC_qPi_flag[i][j]$ được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ hai có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ ba và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ thứ ba. Tức là, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ thứ ba và/hoặc phần tử cú pháp cho giá trị thông số lượng tử hóa của mỗi chỉ số của bảng thông số lượng tử hóa sắc độ thứ ba. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` hoặc `qPi_cb_min_idx_minus1` được mô tả ở trên. Ngoài ra, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Ngoài ra, phần tử cú pháp cho giá trị thông số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả ở trên. Thêm nữa, ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, dữ liệu thông số lượng tử hóa sắc độ thứ ba có thể được báo hiệu thông qua tập hợp thông số chuỗi (SPS), tập hợp thông số hình ảnh (PPS), phần đầu lát, hoặc tập hợp thông số thích ứng (APS).

Thêm nữa, ví dụ, khi giá trị của cờ là 1 (tức là, khi cờ biểu diễn một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu thông số lượng tử hóa sắc độ có thể bao gồm dữ liệu thông số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr và thành phần CbCr kết hợp.

Thiết bị giải mã dẫn xuất bảng thông số lượng tử hóa sắc độ dựa trên dữ liệu thông số lượng tử hóa sắc độ (S920). Thiết bị giải mã có thể dẫn xuất bảng thông số lượng tử hóa sắc độ dựa trên dữ liệu thông số lượng tử hóa sắc độ. bảng thông số lượng tử hóa sắc độ có thể được gọi là bảng ánh xạ thông số lượng tử hóa sắc độ, hoặc bảng ánh xạ thông số lượng tử hóa được định rõ bởi người dùng.

Ví dụ, như được mô tả ở trên, bảng thông số lượng tử hóa sắc độ có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng thông số lượng tử hóa sắc độ, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng thông số lượng tử hóa sắc độ và/hoặc các phần tử cú pháp cho các giá trị thông số lượng tử hóa của các chỉ số của bảng thông số lượng tử hóa sắc độ. Tức là, ví dụ, bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa.

Ví dụ, nếu giá trị của cờ là 0 (tức là, cờ biểu diễn nhiều bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), bảng thông số lượng tử hóa sắc độ thứ nhất cho thành phần sắc độ Cb có thể được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, thì bảng thông số lượng tử hóa sắc độ thứ hai cho thành phần sắc độ Cr có thể được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, thì bảng thông số lượng tử hóa sắc độ thứ ba cho thành phần sắc độ CbCr có thể được dẫn xuất dựa trên dữ liệu thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, nếu giá trị của cờ là 1 (tức là, cờ biểu diễn một bảng thông số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì bảng thông số lượng tử hóa sắc độ thứ nhất cho các thành phần sắc độ có thể được dẫn xuất dựa trên dữ liệu

thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr, và/hoặc thành phần CbCr kết hợp.

Thiết bị giải mã dẫn xuất thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng thông số lượng tử hóa sắc độ (S930). Thiết bị giải mã có thể dẫn xuất thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng thông số lượng tử hóa sắc độ.

Ví dụ, nếu giá trị của cờ là 0, thì thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ thứ nhất, và thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ thứ hai. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, thì thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ thứ nhất, thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ thứ hai, và thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ thứ ba. Ở đây, thông số lượng tử hóa cho thành phần Cb có thể biểu diễn $QP'Cb$ như được mô tả ở trên, thông số lượng tử hóa cho thành phần Cr có thể biểu diễn $QP'Cr$ như được mô tả ở trên, và thông số lượng tử hóa cho thành phần CbCr kết hợp có thể biểu diễn $QP'CbCr$ như được mô tả ở trên.

Ví dụ, chỉ số cho các thành phần sắc độ (thành phần Cb, thành phần Cr, hoặc thành phần CbCr kết hợp) có thể được dẫn xuất dựa trên thông số lượng tử hóa cho thành phần độ chói, và thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên thông số lượng tử hóa sắc độ cho chỉ số của bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên thông số lượng tử hóa sắc độ cho chỉ số giống

như thông số lượng tử hóa của thành phần độ chói trong bảng thông số lượng tử hóa sắc độ.

Thêm nữa, ví dụ, thông số lượng tử hóa sắc độ (tức là, $QP'Cb$, $QP'Cr$, hoặc $QP'CbCr$) cho các thành phần sắc độ có thể được dẫn xuất bằng cách cộng độ bù vào thông số lượng tử hóa sắc độ (ví dụ, $QPCb$, $QPCr$, hoặc $QPCbCr$) cho chỉ số của bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ (thành phần Cb , thành phần Cr , hoặc thành phần $CbCr$ kết hợp). Độ bù có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn độ bù để dẫn xuất thông số lượng tử hóa cho các thành phần sắc độ.

Thêm nữa, ví dụ, nếu giá trị của c_r là 1, thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của c_r là 1, thì thông số lượng tử hóa sắc độ cho các thành phần sắc độ được dẫn xuất dựa trên một bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Theo đó, thông số lượng tử hóa sắc độ có thể được áp dụng ngang bằng với các thành phần sắc độ.

Ví dụ, chỉ số cho các thành phần sắc độ (thành phần Cb , thành phần Cr , và thành phần $CbCr$ kết hợp) có thể được dẫn xuất dựa trên thông số lượng tử hóa cho thành phần độ chói, và thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên thông số lượng tử hóa sắc độ cho chỉ số của bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên thông số lượng tử hóa sắc độ cho chỉ số giống như thông số lượng tử hóa của thành phần độ chói trong bảng thông số lượng tử hóa sắc độ.

Thêm nữa, ví dụ, thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất bằng cách cộng độ bù vào thông số lượng tử hóa sắc độ cho chỉ số của

bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ. Độ bù có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn độ bù để dẫn xuất thông số lượng tử hóa cho các thành phần sắc độ.

Thiết bị giải mã dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên thông tin dự đoán (S940). Thiết bị giải mã có thể xác định liệu việc dự đoán liên ảnh hay việc dự đoán trong ảnh được áp dụng cho các thành phần sắc độ dựa trên thông tin dự đoán, và có thể thực hiện việc dự đoán dựa trên điều này. Tức là, ví dụ, thiết bị giải mã có thể xác định liệu việc dự đoán liên ảnh hay việc dự đoán trong ảnh được áp dụng cho khối hiện tại cho các thành phần sắc độ dựa trên thông tin dự đoán, và có thể thực hiện việc dự đoán dựa trên điều này.

Ví dụ, thiết bị giải mã có thể dẫn xuất chế độ dự đoán đang được áp dụng cho khối hiện tại cho các thành phần sắc độ dựa trên thông tin dự đoán, và có thể dẫn xuất các mẫu dự đoán của khối hiện tại dựa trên chế độ dự đoán. Ví dụ, nếu việc dự đoán liên ảnh được áp dụng cho khối hiện tại, thì thiết bị giải mã có thể dẫn xuất thông tin chuyển động của khối hiện tại dựa trên thông tin dự đoán được bao gồm trong thông tin ảnh, và có thể dẫn xuất các mẫu dự đoán của khối hiện tại dựa trên thông tin chuyển động. Thêm nữa, ví dụ, nếu việc dự đoán trong ảnh được áp dụng cho khối hiện tại, thì thiết bị giải mã có thể dẫn xuất các mẫu tham chiếu dựa trên các mẫu lân cận của khối hiện tại, và có thể dẫn xuất các mẫu dự đoán của khối hiện tại dựa trên các mẫu tham chiếu và chế độ dự đoán trong ảnh của khối hiện tại. Các mẫu tham chiếu có thể bao gồm các mẫu tham chiếu phía trên và các mẫu tham chiếu bên trái của khối hiện tại. Ví dụ, khi kích thước của khối hiện tại là $N \times N$ và thành phần x và thành phần y của vị trí mẫu phía trên cùng-bên trái của khối hiện tại lần lượt là 0, thì các mẫu tham chiếu bên trái có thể là $p[-$

1][0] đến p[-1][2N-1] và các mẫu tham chiếu phía trên cùng có thể là p[0][-1] đến p[2N-1][-1].

Thiết bị giải mã dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư (S950). Ví dụ, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư nhận được. Thêm nữa, ví dụ, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi dựa trên thông tin phần dư, và có thể dẫn xuất các hệ số biến đổi được biến đổi ngược bằng cách biến đổi ngược các hệ số biến đổi. Các hệ số biến đổi có thể bao gồm các hệ số biến đổi cho thành phần Cb, các hệ số biến đổi cho thành phần Cr, và/hoặc các hệ số biến đổi cho thành phần CbCr kết hợp.

Thiết bị giải mã dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi dựa trên thông số lượng tử hóa sắc độ (S960). Thiết bị giải mã có thể dẫn xuất các mẫu dư dựa trên thông số lượng tử hóa sắc độ.

Ví dụ, nếu giá trị cờ là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho thành phần Cb bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu dư cho thành phần Cr bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị cờ là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho thành phần Cb bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dẫn xuất các mẫu dư cho thành phần Cr bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dẫn xuất các mẫu dư cho thành phần CbCr kết hợp bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần CbCr kết hợp dựa trên thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví

dụ, nếu giá trị còn là 1, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần sắc độ dựa trên thông số lượng tử hóa sắc độ.

Thêm nữa, nếu giá trị còn là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho thành phần Cb bằng cách khử lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu dư cho thành phần Cr bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị còn là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho thành phần Cb bằng cách khử lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần Cb dựa trên thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dẫn xuất các mẫu dư cho thành phần Cr bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dẫn xuất các mẫu dư cho thành phần CbCr kết hợp bằng cách khử lượng tử hóa các hệ số biến đổi cho thành phần CbCr kết hợp dựa trên thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví dụ, nếu giá trị còn là 1, thì thiết bị giải mã có thể dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần sắc độ dựa trên thông số lượng tử hóa sắc độ.

Thiết bị giải mã tạo ra hình ảnh được tái tạo dựa trên các mẫu dư đoán và các mẫu dư (S970). Ví dụ, thiết bị giải mã có thể tạo ra hình ảnh được tái tạo dựa trên các mẫu dư.

Trong khi đó, ví dụ, thiết bị giải mã có thể dẫn xuất các mẫu dư đoán bằng cách thực hiện chế độ dư đoán liên ảnh hoặc chế độ dư đoán trong ảnh cho khối hiện tại dựa

trên thông tin dự đoán nhận được qua luồng bit, và có thể tạo ra các mẫu được tái tạo và/hoặc hình ảnh được tái tạo bằng cách cộng các mẫu dự đoán và các mẫu dư.

Sau đó, nếu cần thiết, để cải thiện chất lượng hình ảnh chủ quan/khách quan, thủ tục lọc trong vòng, như lọc khử khối, SAO và/hoặc thủ tục ALF, có thể được áp dụng cho các mẫu được tái tạo như được mô tả ở trên.

Fig.10 thể hiện theo cách giản lược thiết bị giải mã để thực hiện phương pháp giải mã hình ảnh theo sáng chế này. Phương pháp được bộc lộ trong Fig.9 có thể được thực hiện bởi thiết bị giải mã được bộc lộ ở Fig.10. Cụ thể, ví dụ, bộ giải mã entropi của thiết bị giải mã của Fig.10 có thể thực hiện S900 đến S910 của Fig.9, bộ dự đoán của thiết bị giải mã của Fig.10 có thể thực hiện S940 của Fig.9, bộ xử lý phần dư của thiết bị giải mã của Fig.10 có thể thực hiện S920 đến S930 và S950 đến S960 của Fig.9, và bộ cộng của thiết bị giải mã của Fig.10 có thể thực hiện S970 của Fig.9.

Theo sáng chế này như được mô tả ở trên, để dẫn xuất thông số lượng tử hóa cho các thành phần sắc độ, bảng thông số lượng tử hóa sắc độ có thể được xác định dựa trên cờ biểu diễn liệu cùng một bảng thông số lượng tử hóa sắc độ được sử dụng hay không, và thông qua điều này, hiệu quả tạo mã có thể được cải thiện thông qua cách thực hiện việc tạo mã dựa trên thông số lượng tử hóa theo các đặc tính của ảnh.

Thêm nữa, theo sáng chế này, bảng thông số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên dữ liệu lượng tử hóa sắc độ được báo hiệu riêng lẻ hoặc chung cho các thành phần sắc độ, và thông qua điều này, hiệu quả tạo mã có thể được cải thiện thông qua thực hiện việc tạo mã dựa trên thông số lượng tử hóa theo các đặc tính của ảnh.

Ở phương án được mô tả ở trên, các phương pháp được mô tả dựa trên sơ đồ tiến trình có một loạt các bước hoặc các khối. Sáng chế không được giới hạn ở thứ tự của

các bước hoặc các khối ở trên. Một số bước hoặc khối có thể xuất hiện đồng thời hoặc theo trật tự khác với các bước hoặc các khối khác được mô tả ở trên. Thêm nữa, người có hiểu biết trung bình trong lĩnh vực kỹ thuật này sẽ hiểu rằng các bước được thể hiện trong sơ đồ tiến trình ở trên không phải là riêng biệt và các bước thêm nữa cũng thể được bao gồm, hoặc một hoặc nhiều bước trong các sơ đồ tiến trình làm ví dụ có thể được xóa mà không ảnh hưởng tới phạm vi bảo hộ của sáng chế này.

Các phương án được mô tả trong bản mô tả này có thể được thực hiện trên bộ xử lý, bộ vi xử lý, bộ điều khiển hoặc chip. Ví dụ, các đơn vị chức năng được thể hiện ở mỗi hình vẽ có thể được thực hiện bằng cách được triển khai trên máy tính, bộ xử lý, bộ vi xử lý, bộ điều khiển hoặc chip. Trong trường hợp này, thông tin cho việc triển khai (ví dụ, thông tin về các lệnh) hoặc thuật toán có thể được lưu trữ trong phương tiện lưu trữ số.

Bên cạnh đó, thiết bị giải mã và thiết bị mã hóa mà sáng chế được áp dụng cho có thể được bao gồm trong thiết bị truyền/thu quảng bá đa phương tiện, thiết bị đầu cuối truyền thông di động, thiết bị video chiếu phim gia đình, thiết bị video chiếu phim số, camera giám sát, thiết bị trò chuyện video, thiết bị truyền thông thời gian thực như truyền thông video, thiết bị phát luồng (streaming) di động, phương tiện lưu trữ, máy ghi hình cầm tay, thiết bị cung cấp dịch vụ video theo yêu cầu (Video on Demand, VoD), thiết bị cung cấp dịch vụ nội dung trên nền mạng viễn thông (Over The Top, OTT), thiết bị cung cấp dịch vụ phát luồng Internet, thiết bị video ba chiều (Three Dimensional, 3D), thiết bị video dùng cho các cuộc họp tổ chức qua điện thoại, thiết bị người dùng cho vận tải (ví dụ, thiết bị người dùng cho xe cộ, thiết bị người dùng cho máy bay, thiết bị người dùng cho tàu, v.v.) và thiết bị video y tế và có thể được sử dụng để xử lý các tín hiệu video hoặc các tín hiệu dữ liệu. Ví dụ, thiết bị video của dịch vụ cung cấp nội dung trên

nền mạng viễn thông (Over The Top, OTT) có thể bao gồm bảng điều khiển chơi trò chơi, máy đọc đĩa Blu-ray, ti vi truy cập Internet, hệ thống rạp hát gia đình, điện thoại thông minh, máy tính bảng, đầu ghi video kỹ thuật số (Digital Video Recorder, DVR), và tương tự.

Hơn thế nữa, phương pháp xử lý mà sáng chế được áp dụng cho có thể được tạo ra dưới dạng chương trình mà để được thực thi bởi máy tính, và có thể được lưu trữ trên phương tiện ghi có thể đọc được bằng máy tính. Dữ liệu đa phương tiện mà có cấu trúc dữ liệu theo sáng chế thì cũng có thể được lưu trữ trên phương tiện ghi đọc được bằng máy tính. Phương tiện ghi đọc được bởi máy tính bao gồm tất cả các loại thiết bị lưu trữ mà dữ liệu đọc được bởi máy tính được lưu trong đó. Phương tiện ghi đọc được bằng máy tính này có thể bao gồm BD, thiết bị buýt nối tiếp vạn năng (Universal Serial Bus, USB), ROM, PROM, EPROM, RAM, CD-ROM, băng từ, đĩa mềm, và thiết bị lưu trữ dữ liệu quang học chẳng hạn. Hơn nữa, phương tiện ghi đọc được bởi máy tính chứa phương tiện được thực hiện ở dạng của các sóng bộ mang (ví dụ, truyền qua Internet). Bên cạnh đó, luồng bit được tạo ra bằng phương pháp mã hóa có thể được lưu trong phương tiện ghi đọc được bởi máy tính hoặc có thể được truyền qua các mạng truyền thông có dây hoặc không dây.

Bên cạnh đó, các phương án theo sáng chế có thể được thực hiện với sản phẩm chương trình máy tính bằng các mã chương trình, và các mã chương trình này có thể được thực hiện trên máy tính bởi các phương án theo sáng chế. Các mã chương trình có thể được lưu trữ trong bộ mang vốn có thể đọc được bởi máy tính.

Fig.11 minh họa sơ đồ dạng cấu trúc của hệ thống phát luồng các nội dung mà sáng chế được áp dụng vào đó.

Hệ thống phát luồng nội dung mà phương án thực hiện (các phương án thực hiện) của sáng chế này được áp dụng cho, có thể phần lớn bao 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 dạng 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 tạo mã hoặc phương pháp tạo ra luồng bit mà phương án (các 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 đang khả dụng. Khi người dùng yêu cầu dịch vụ mong muốn từ máy chủ web, thì máy chủ web chuyển nó đến máy chủ phát luồng, và máy chủ phát luồng truyền dữ liệu đa phương tiện tới người dùng. Trong trường hợp này, hệ thống phát luồng các nội dung có thể bao gồm máy chủ điều khiển riêng biệt. Trong trường hợp này, máy chủ điều khiển có chức năng để điều khiển mệnh lệnh/phản hồi giữa các thiết bị trong hệ thống phát luồng các nội dung.

Máy chủ phát luồng có thể nhận các nội dung từ bộ lưu trữ phương tiện và/hoặc máy chủ mã hóa. Ví dụ, 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 định trước.

Các ví dụ về thiết bị người dùng có thể bao gồm điện thoại di động, điện thoại thông minh, máy tính xách tay, thiết bị đầu cuối truyền quảng bá 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 bộ hiển thị gắn trên đầu), các ti vi số, máy tính để bàn, bảng ký hiệu dạng số và tương tự. Mỗi máy chủ trong hệ thống phát luồng các nội dung có thể được vận hành như là máy chủ phân tán, mà trong trường hợp đó dữ liệu nhận được từ mỗi máy chủ có thể được phân tán.

Các yêu cầu bảo hộ được mô tả trong sáng chế này có thể được tổ hợp theo các cách khác nhau. Ví dụ, các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng phương pháp của sáng chế có thể được tổ hợp và được triển khai dưới dạng thiết bị, và các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng thiết bị của sáng chế có thể được tổ hợp và được triển khai dưới dạng phương pháp. Bên cạnh đó, các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng phương pháp của sáng chế và các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng thiết bị có thể được tổ hợp để được triển khai dưới dạng thiết bị, và các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng phương pháp và các dấu hiệu kỹ thuật của các yêu cầu bảo hộ dạng thiết bị có thể được tổ hợp để được triển khai dưới dạng phương pháp.

YÊU CẦU BẢO HỘ

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

thu được thông tin hình ảnh gồm thông tin dự đoán, thông tin phần dư cho các thành phần sắc độ, và cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng hay không;

thu được dữ liệu thông số lượng tử hóa sắc độ dựa trên cờ;

dẫn xuất ít nhất một bảng thông số lượng tử hóa sắc độ dựa trên dữ liệu thông số lượng tử hóa sắc độ;

dẫn xuất ít nhất một thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên ít nhất một bảng thông số lượng tử hóa sắc độ;

dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên thông tin dự đoán;

dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư;

dẫn xuất các mẫu dư cho các thành phần sắc độ bằng cách khử lượng tử hóa các hệ số biến đổi dựa trên ít nhất một thông số lượng tử hóa sắc độ; và

tạo ra ảnh được tái tạo dựa trên các mẫu dự đoán và các mẫu dư,

trong đó cờ bằng 0 biểu diễn rằng bảng thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết được phát tín hiệu.

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

dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên việc dự đoán liên ảnh hoặc việc dự đoán trong ảnh;

dẫn xuất các mẫu dư cho thành phần sắc độ dựa trên các mẫu dự đoán;

tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng hay không;

tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ; và

mã hóa thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ và cờ,

trong đó cờ bằng 0 biểu diễn rằng bảng thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết được phát tín hiệu.

3. Phương tiện lưu trữ đọc được bởi máy tính không chuyển tiếp lưu trữ luồng bit được tạo ra bởi phương pháp, phương pháp bao gồm các bước:

dẫn xuất các mẫu dự đoán cho các thành phần sắc độ dựa trên việc dự đoán liên ảnh hoặc việc dự đoán trong ảnh;

dẫn xuất các mẫu dư cho thành phần sắc độ dựa trên các mẫu dự đoán;

tạo ra cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng hay không;

tạo ra dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ;

mã hóa thông tin dự đoán cho các thành phần sắc độ, thông tin phần dư cho các thành phần sắc độ, dữ liệu thông số lượng tử hóa sắc độ, và cờ; và

tạo ra luồng bit gồm thông tin dự đoán, thông tin phần dư, dữ liệu thông số lượng tử hóa sắc độ, và cờ,

trong đó cờ bằng 0 biểu diễn rằng bảng thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết được phát tín hiệu.

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

thu được luồng bit của thông tin hình ảnh gồm thông tin dự đoán và thông tin phần dư cho các thành phần sắc độ, cờ, và dữ liệu thông số lượng tử hóa sắc độ cho các thành phần sắc độ; và

truyền dữ liệu gồm luồng bit của thông tin hình ảnh gồm thông tin dự đoán, thông tin phần dư, cờ, và dữ liệu thông số lượng tử hóa sắc độ,

trong đó thông tin dự đoán là thông tin cho các mẫu dự đoán cho các thành phần sắc độ, và các mẫu dự đoán cho các thành phần sắc độ được dẫn xuất dựa trên việc dự đoán liên ảnh hoặc việc dự đoán trong ảnh,

trong đó thông tin phần dư là thông tin cho các mẫu dư cho các thành phần sắc độ, và các mẫu dư cho các thành phần sắc độ được dẫn xuất dựa trên các mẫu dự đoán,

trong đó cờ biểu diễn liệu một bảng thông số lượng tử hóa sắc độ có được áp dụng hay không,

trong đó dữ liệu thông số lượng tử hóa sắc độ được tạo ra dựa trên cờ, và

trong đó cờ bằng 0 biểu diễn rằng bảng thông số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng thông số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng thông số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết được phát tín hiệu.

FIG. 1

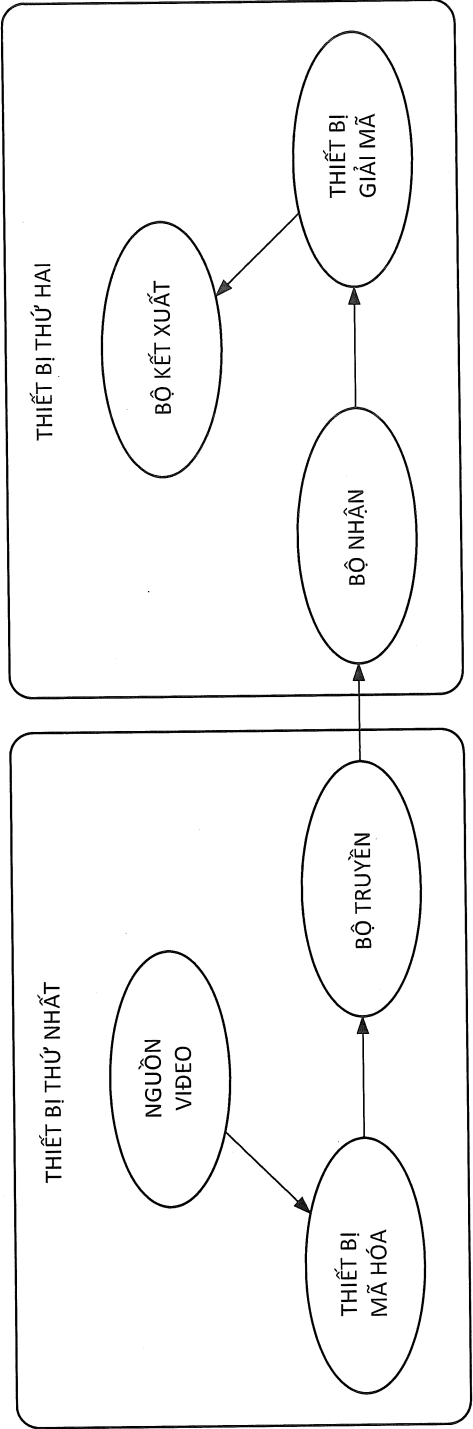


FIG. 2

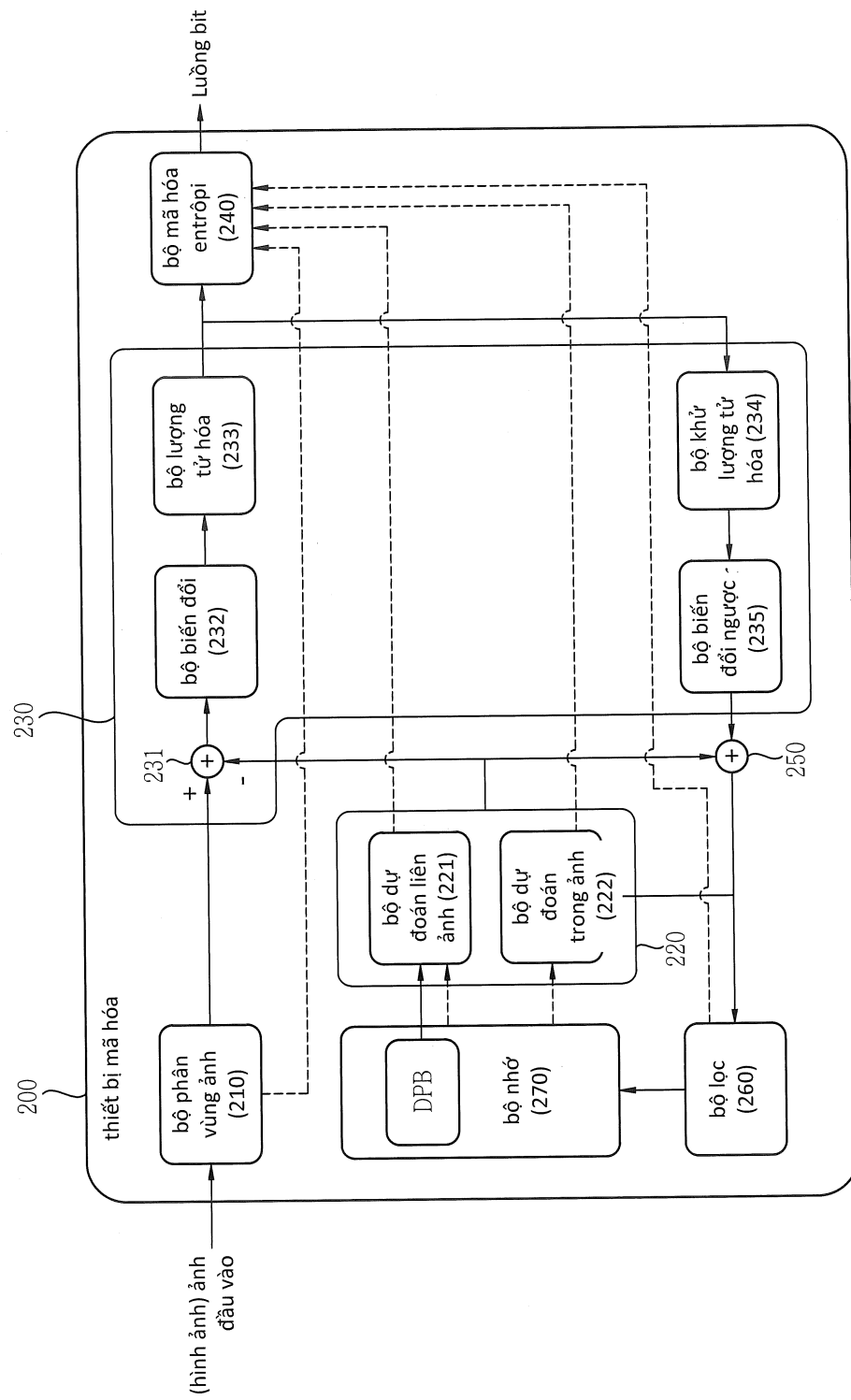


FIG. 3

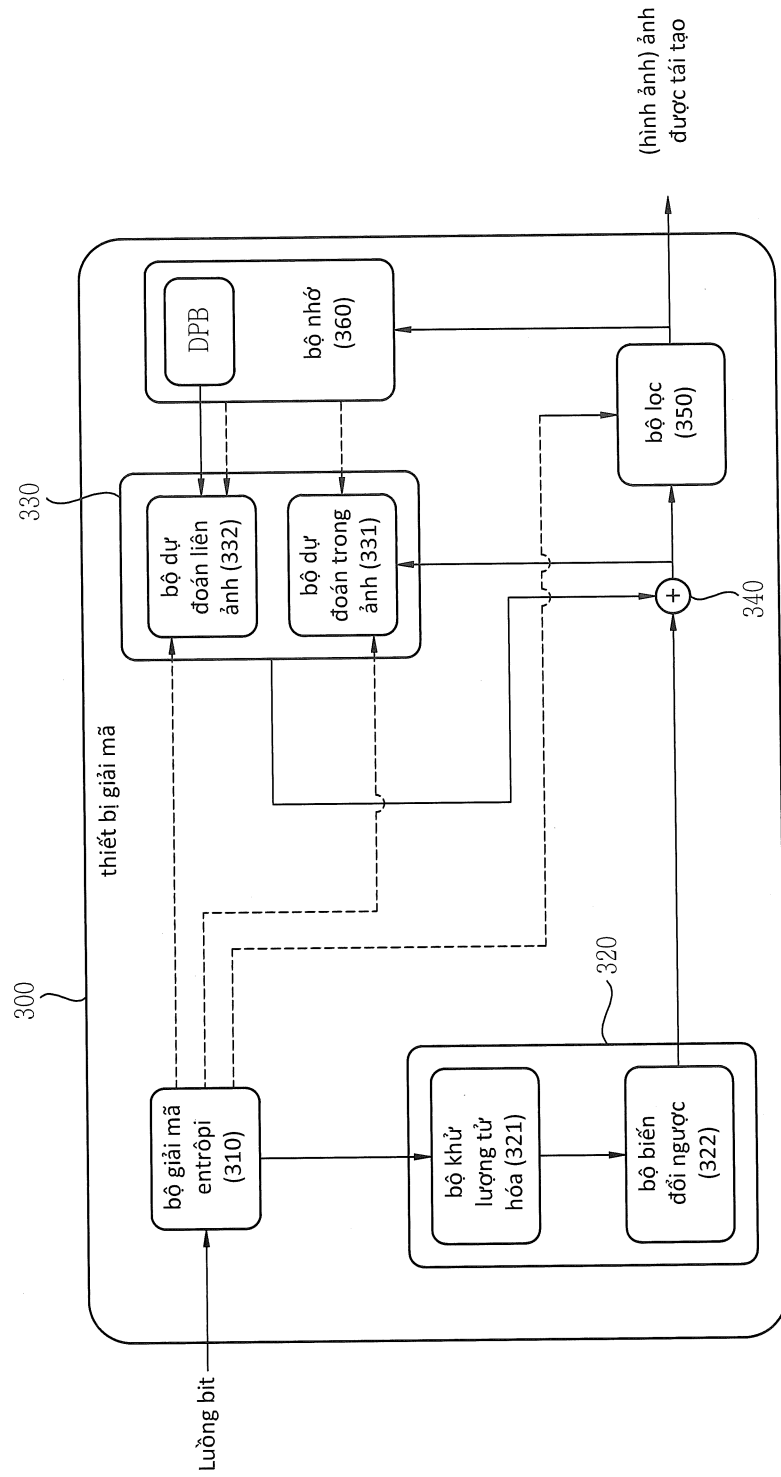


FIG. 4

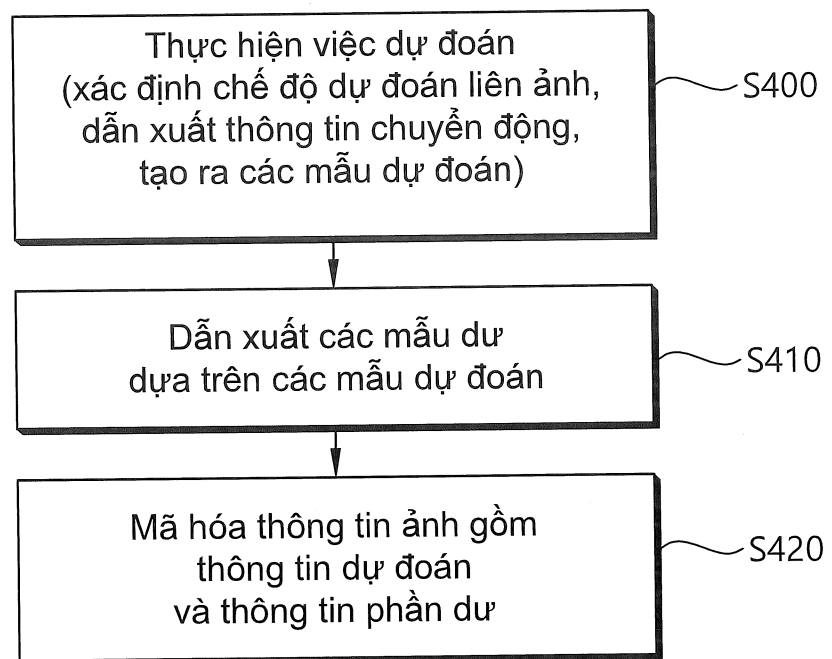


FIG. 5

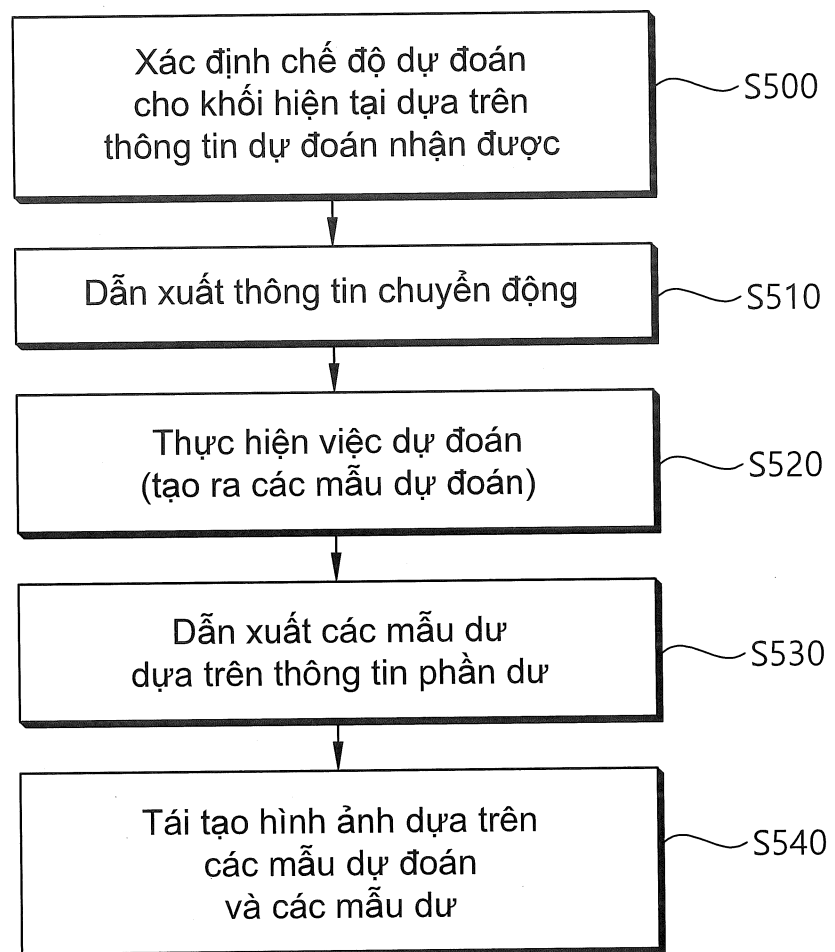


FIG. 6

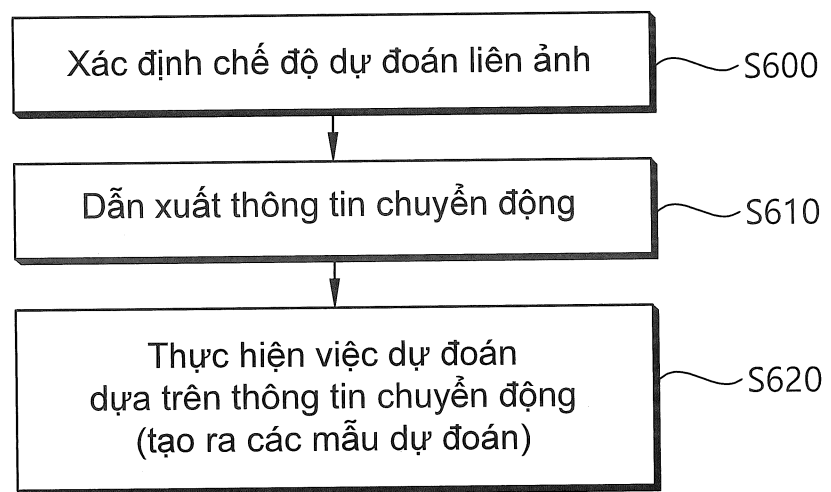


FIG. 7

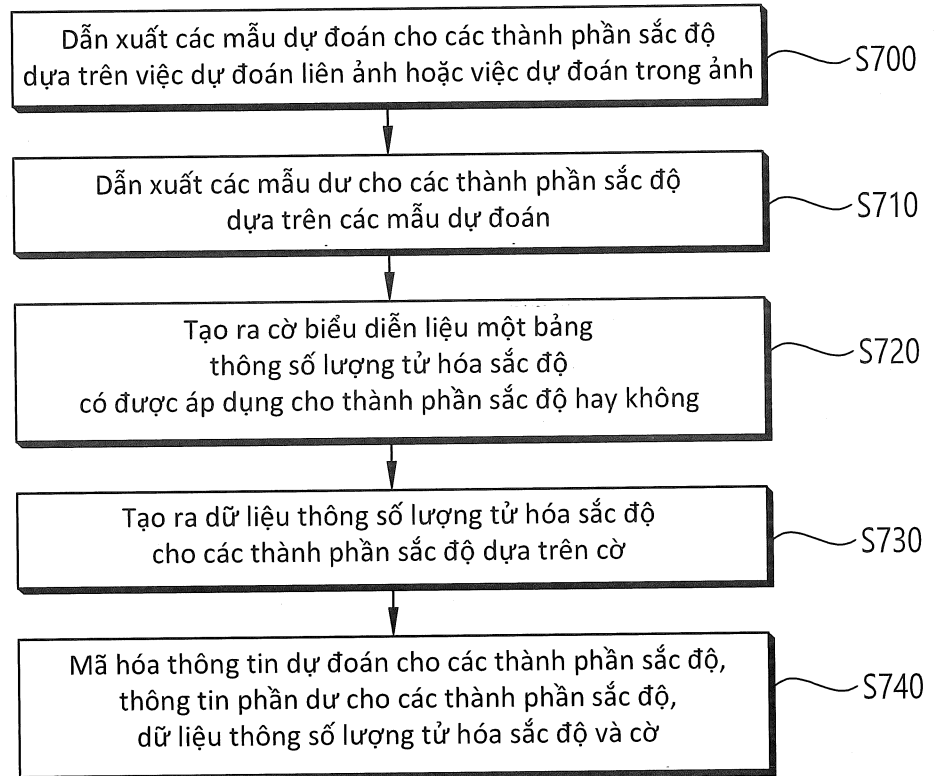


FIG. 8

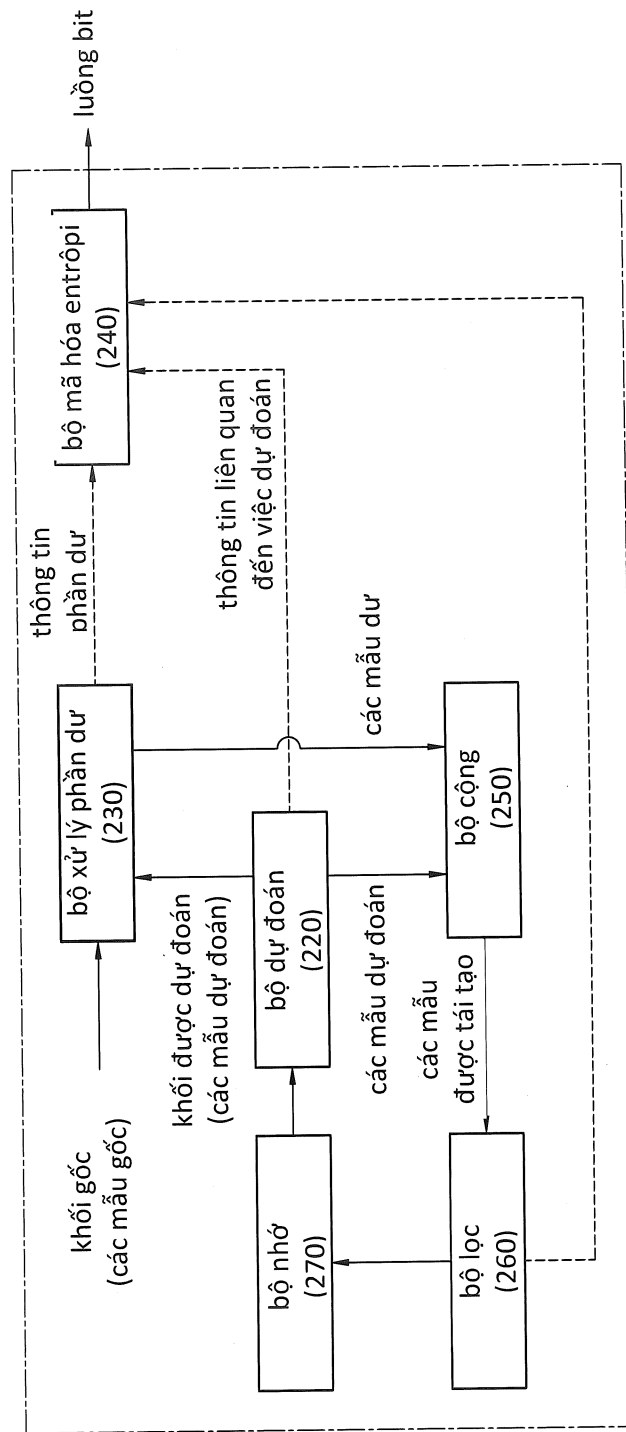


FIG. 9

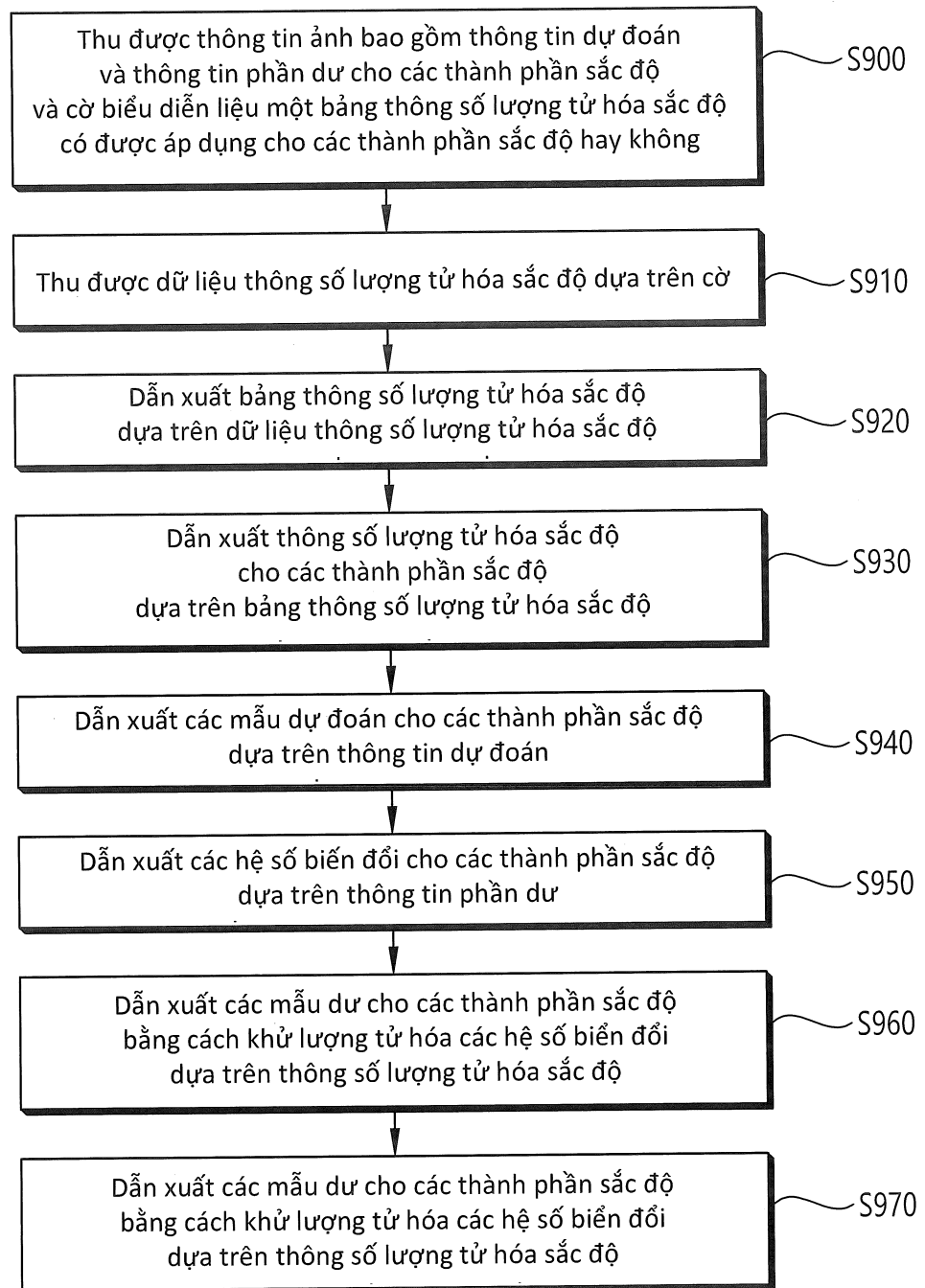


FIG. 10

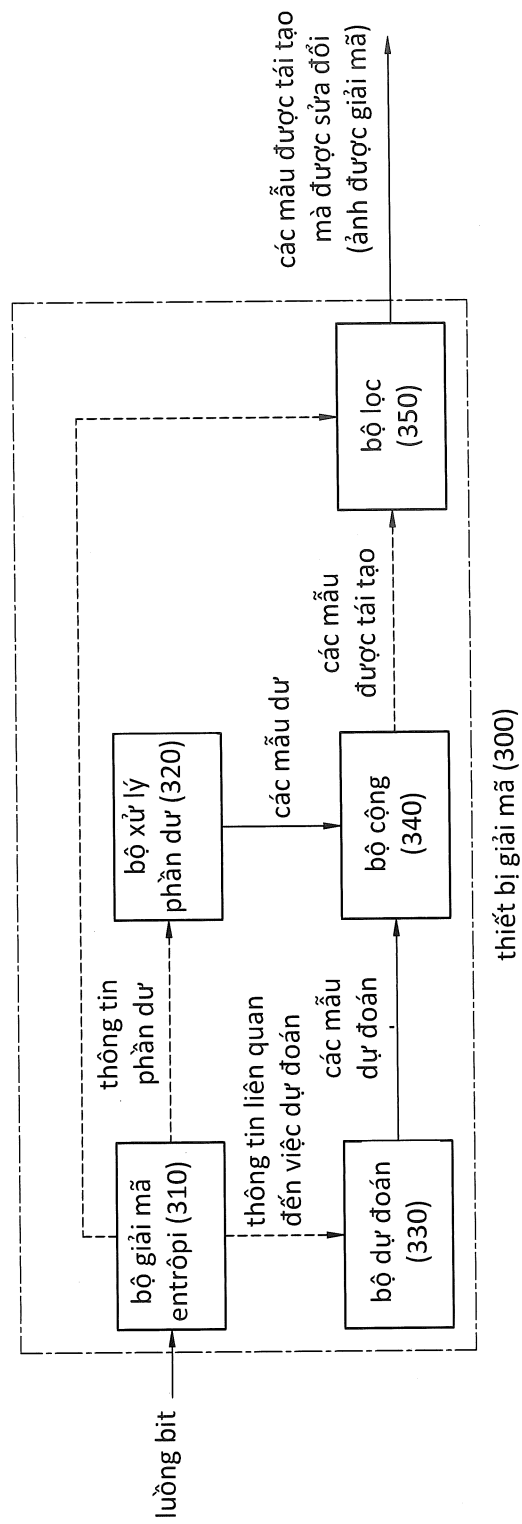


FIG. 11

